

UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA



TESIS DOCTORAL

CONTRIBUCIONES EN LA CREACIÓN Y APLICABILIDAD DE JUEGOS
SERIOS (SERIOUS GAMES) EN EL ÁMBITO DEL GEOPOSICIONAMIENTO

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Víctor Manuel Pérez Colado

DIRECTORES

Baltasar Fernández Manjón

Iván Martínez Ortiz

Madrid, 2022

UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA



TESIS DOCTORAL

**CONTRIBUCIONES EN LA CREACIÓN Y APLICABILIDAD DE JUEGOS
SERIOS (SERIOUS GAMES) EN EL ÁMBITO DEL GEOPOSICIONAMIENTO**

**MEMORIA PARA OPTAR AL GRADO DE DOCTOR EN EL
PROGRAMA DE DOCTORADO EN INGENIERÍA INFORMÁTICA**

PRESENTADA POR

Víctor Manuel Pérez Colado

DIRECTORES

**Baltasar Fernández Manjón
Iván Martínez Ortiz**

Madrid, 2022



U N I V E R S I D A D
COMPLUTENSE
M A D R I D

**DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS
PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR**

D. **Víctor Manuel Pérez Colado**, estudiante en el **Programa de Doctorado en Ingeniería Informática**, de la Facultad de Informática de la Universidad Complutense de Madrid, como autor de la tesis presentada para la obtención del título de Doctor titulada:

Contribuciones a la creación y aplicabilidad de los juegos serios (serious games) en el ámbito del geoposicionamiento

y dirigida por Baltasar Fernández Manjón e Iván Martínez Ortiz

DECLARO QUE:

La tesis es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, de acuerdo con el ordenamiento jurídico vigente, en particular, la Ley de Propiedad Intelectual (R.D. legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, modificado por la Ley 2/2019, de 1 de marzo, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), en particular, las disposiciones referidas al derecho de cita.

Del mismo modo, asumo frente a la Universidad cualquier responsabilidad que pudiera derivarse de la autoría o falta de originalidad del contenido de la tesis presentada de conformidad con el ordenamiento jurídico vigente.

En Madrid, a 6 de mayo de 2022

Fdo.: Víctor Manuel Pérez Colado

“Nunca uses a un humano para hacer el trabajo de una máquina.”

Agente Smith – The Matrix

Agradecimientos

Siempre me ha resultado difícil escribir este tipo de agradecimientos porque no puedo evitar tener la sensación de olvidarme de personas a las que debería de estar agradecido. Por eso, quiero dedicar este primer párrafo a agradecer a todas las personas que en general se interesaron por mi tesis, quisieron aportar su granito con ideas o que simplemente me dieron ese empujón que necesitaba para seguir adelante: lo hemos conseguido.

Quería agradecer a toda la gente del aula 16 y a los demás doctorandos con los que he coincidido: Joaquín, Pablo, Jesús, Ali, Marta, Miguel, Dani; y a Luisma, que al rechazar su plaza en la beca pude tomarla yo para crear esta tesis. Con vosotros me he inspirado y he podido ver que no todo son *papers* y experimentos.

A Cristian, Gorka, Irene, Frank, Max y Sora, que me han escuchado una y mil veces, ya sea dando vueltas en círculos alrededor de la facultad o dando vueltas por toda Madrid, pero en cualquier caso rompiendo las barreras de mi mente.

A Leandro, Alma, Rosa, David y Julieta que han colaborado aportado ideas, utilizado o revisado alguna parte de uAdventure y a Piotr por su prototipo del editor. A los alumnos de los *workshops* especialmente a Alejandro, Ana, Georgi, Miguel, Álvaro y Dany. También a Luis, a Ignacio y Lourdes que nos abrieron las puertas en la ESD.

A Sandra y Maca, pues juntos diseñamos El Señor De Los Arbustos, le dimos una y mil vueltas al juego y sus personajes, y estuvieron al pie del cañón para ayudarme en los experimentos. Y a Jesús, que sin él no habría sido posible en primera instancia.

A Toni, Cris y Manu, compañeros del grupo e-UCM que me han dado tanto apoyo como han podido y de forma incansable, luchando tantos días hasta altas horas... No sólo habéis revisado mis artículos, sino que me habéis ayudado a aprender tanto con vuestra guía y ejemplo que ahora siento que soy mucho mejor investigador gracias a vosotros. A Manu quiero dedicarle una línea más en especial, porque a pesar de no haber sido director en esta tesis he sentido muchas veces que era un director más, implicándose y dándome consejos de los que me hacían romper los esquemas e ir siempre más allá. Eternamente agradecido, compañeros.

A Iván “Koro-sensei” y Balta “The Collejator”, mis directores e intérpretes *checoslovaco-español*, con los que he podido sentirme arropado desde el principio, explorando todas las ideas que el tiempo nos ha permitido. Desde el primer día me habéis hecho saber que

puedo llegar muy lejos (todavía recuerdo aquella reunión en primero de carrera con Iván para preguntarle qué carrera era mejor y el día que Balta nos cogió a mi hermano y a mí por banda porque “había oído que éramos muy buenos”). Siempre me habéis recibido con la mano tendida, aunque me equivocara una y otra vez. Gracias sinceramente por haber hecho esta tesis posible.

A *Rejvel* por haberme apoyado durante todo este tiempo, haciendo que la vida sea mucho más colorida e interesante y no teniendo miedo a sentarte delante de mi ordenador a probar uAdventure o a leer cualquier texto por enrevesado que fuera.

Y por supuesto a mi familia, que me han dado todo, cada día, incondicionalmente. Habéis sido mi sustento, tanto por alimentar mi fuego interno cada día con vuestro apoyo y vuestro ánimo, como por bajarme de las nubes de miedo para ponerme de nuevo los pies en la tierra. Y a ti, Iván, quiero darte las gracias puesto que uAdventure fue un día tu proyecto hasta que lo dejaste en mis manos y es por ello por lo que hoy tengo esta tesis. Gracias de corazón.

Sobre este documento

Este trabajo de tesis doctoral es una recopilación de publicaciones de acuerdo con lo expuesto en la Normativa de Desarrollo del Real Decreto 99/2011, del 28 de Enero (BOE 10/02/2011), por el que se regulan las enseñanzas oficiales de doctorado en la Universidad Complutense de Madrid¹ ².

A continuación, se enumeran los artículos presentados:

- ☰ Víctor Manuel Pérez-Colado, Iván José Pérez-Colado, Manuel Freire-Morán, Iván Martínez-Ortiz, and Baltasar Fernández-Manjón. (2019) **Simplifying the creation of adventure serious games with educational-oriented features**, *Educational Technology and Society*, 22(3), 32–46.
- ☰ Víctor Manuel Pérez-Colado, Iván José Pérez-Colado, Iván Martínez-Ortiz, Manuel Freire-Morán, and Baltasar Fernández-Manjón. (2017). **Simplifying location-based serious game authoring**. In *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM 2017)*. Association for Computing Machinery, New York, NY, USA, Article 45, 1–9. <https://doi.org/10.1145/3144826.3145395>
- ☰ Víctor Manuel Pérez-Colado, Dan Cristian Rotaru, Manuel Freire-Morán, Iván Martínez-Ortiz, and Baltasar Fernández-Manjón. (2018). **Learning analytics for location-based serious games**. In *2018 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, 1192-1200 <https://doi.org/10.1109/educon.2018.8363365>
- ☰ Víctor Manuel Pérez-Colado, Iván José Pérez-Colado, Manuel Freire-Morán, Iván Martínez-Ortiz, and Baltasar Fernández-Manjón. (2021). **A Tool Supported Approach for Teaching Serious Game Learning Analytics**. In *2021 IEEE Frontiers in Education Conference (FIE)*, IEEE, 1–8. <https://doi.org/10.1109/FIE49875.2021.9637062>

¹ <https://www.boe.es/eli/es/rd/2011/01/28/99/con>

² <https://edotorado.ucm.es/normativa>

- ☰ Víctor Manuel Pérez-Colado, Iván José Pérez-Colado, Iván Martínez-Ortiz, Manuel Freire-Morán, and Baltasar Fernández-Manjón. (2021b). **Extending Narrative Serious Games Using Ad-Hoc Mini-games**. In W. Zhou & M. Yi (Eds.), *Advances in Web-Based Learning – ICWL 2021. ICWL 2021. Lecture Notes in Computer Science, vol 13103*. Springer, Cham. (pp. 63–74). https://doi.org/10.1007/978-3-030-90785-3_6
- ☰ Víctor Manuel Pérez-Colado, Iván José Pérez-Colado, Iván Martínez-Ortiz, Manuel Freire-Morán, and Baltasar Fernández-Manjón. (2021a) **Democratizing Game Learning Analytics for Serious Games**. In: de Rosa F., Marfisi Schottman I., Baalsrud Hauge J., Bellotti F., Dondio P., Romero M. (eds) *Games and Learning Alliance. GALA 2021*. Lecture Notes in Computer Science, vol 13134. Springer, Cham. (pp. 164–173) https://doi.org/10.1007/978-3-030-92182-8_16

De acuerdo con la normativa, este documento incluye un índice con los títulos de los capítulos, un estudio del estado del arte, una descripción de los objetivos propuestos para esta tesis y una discusión sobre el contenido incluido en los artículos presentados.

Adicionalmente se presenta un capítulo con conclusiones y posibles líneas futuras de investigación relacionada con el tema.

Finalmente, se incluye una bibliografía con las referencias de los artículos mencionados en esta tesis doctoral.

Resumen

En los últimos años se ha investigado cómo utilizar videojuegos para propósitos más allá del mero entretenimiento. Estos videojuegos se conocen como juegos serios, y, aplicados a la enseñanza, ofrecen un enfoque que a la vez que resulta atractivo para los estudiantes también tiene ventajas para educadores. Por ejemplo, que los alumnos pueden aplicar su conocimiento, experimentando y aprendiendo de sus errores o que permiten la simulación de procesos reales de forma personalizada y segura para cada estudiante. Con el auge de los dispositivos móviles además su aplicabilidad se ha ampliado, ya que es posible utilizar los dispositivos de los propios estudiantes y los juegos serios pueden expandirse a nuevos ámbitos como, por ejemplo, las actividades al aire libre para un aprendizaje contextualizado en el mundo real. Este es el caso de los juegos geoposicionados. Este formato de juego ha recibido un importante impulso a su popularidad gracias a Pokémon GO, un juego mundialmente aclamado donde la mecánica principal del juego es la exploración de elementos virtuales superpuestos sobre espacios del mundo real. Pese al potencial y los beneficios que ofrecen los juegos serios, su uso en la educación es todavía limitado. Esto puede ser debido a distintas razones como, por ejemplo, que el docente sienta que pierde el control de la clase al utilizarlos, que es difícil analizar si los estudiantes han sacado algún provecho de la sesión de juego o que, en muchos casos, falta una validación formal de su efectividad.

El desarrollo de un videojuego es un proceso habitualmente complejo y costoso, de modo que los docentes suelen ver limitadas sus posibilidades a seleccionar juegos serios ya existentes, aunque no encajen completamente con sus necesidades educativas. Los entornos de autoría son un tipo de herramienta que permite rebajar el peaje que es necesario pagar para crear un juego serio, aliviando la necesidad de conocimientos técnicos, normalmente, centrándose en un tipo concreto de juego a desarrollar. El uso de estos entornos de autoría por parte de los docentes abre todo un abanico de posibilidades, permitiendo que el docente pueda crear sus propios juegos o, el caso más habitual, que sería adaptar juegos existentes a sus necesidades educativas. Todo esto con un coste limitado, disponiendo además de características educativas diseñadas específicamente para facilitar y mejorar la experiencia de los docentes. En los últimos años la necesidad de evidencias sobre el proceso y resultado del aprendizaje cuando se usa tecnología ha llevado a la recolección y análisis de datos del proceso educativo en lo que se ha denominado analíticas de aprendizaje. Partiendo de esta idea, ha surgido un nuevo campo

de investigación que en inglés se ha denominado *Game Learning Analytics*, con el objetivo no sólo de poder facilitar el seguimiento de los alumnos mientras juegan, sino también aplicar estas evidencias durante todo el ciclo de vida del juego. Es decir, desde el proceso de desarrollo, pasando por su despliegue y uso, y hasta el objetivo más ambicioso que es poder usar dicha información para validar el juego científicamente. No obstante, la inclusión de analíticas de aprendizaje y su explotación es un proceso complejo, que requiere conocimientos especializados, que no tiene habitualmente un desarrollador de juegos y menos un docente. Por tanto, al introducir una nueva tecnología educativa hay que dar herramientas de ayuda a todos los implicados en el proceso.

El uso de entornos de autoría permite simplificar el proceso al incluir una serie de analíticas básicas por defecto, que pueden utilizarse como punto de partida para obtener una idea general acerca de la actividad de los estudiantes en el juego. Posteriormente se pueden complementar con analíticas específicas personalizadas que puede definir el autor durante la creación del juego. Otras de las dificultades que habitualmente se encuentran los docentes son la complejidad del despliegue de los juegos y la integración con las otras herramientas educativas que ya utilizan. Este no es un problema nuevo ya que ha sido abordado previamente en otros campos como el e-learning mediante la aplicación de estándares y especificaciones técnicas educativas. Como parte de este trabajo de tesis se analizan cómo los estándares pueden ser aplicados en el contexto del desarrollo de los juegos serios para facilitar su aplicación.

Todos los enfoques y modelos presentados se han ejemplificado en esta tesis en un sistema de autoría denominado uAdventure. uAdventure tiene como objetivo simplificar la creación y el uso real de juegos geoposicionados con soporte narrativo en entornos educativos. Este entorno, recoge toda la experiencia del grupo e-UCM con un sistema previo denominado e-Adventure, y lo extiende para ofrecer una experiencia mucho más rica. Aporta mucha más libertad y flexibilidad a través del geoposicionamiento y de potenciar capacidades educativas concretas en sus juegos mediante el uso de minijuegos y extensiones. Este modelo de juegos se complementa con la incorporación de analíticas de aprendizaje por defecto y de múltiples mecanismos de integración que permiten a los educadores tener una experiencia más confiable e integrada.

El uso y aplicación de este entorno de autoría a través de la experimentación con usuarios realizado en la presente tesis ha permitido validar el enfoque. Es decir, probar que este

tipo de soluciones simplifican la creación de juegos serios a usuarios sin experiencia previa, favoreciendo la construcción de analíticas confiables y democratizado su uso por no expertos y, finalmente, simplificando el propio despliegue de los juegos en entornos reales. En definitiva, este enfoque ayuda a mitigar las barreras que se encuentran los docentes al utilizar juegos serios de modo que puede ayudar a su generalización en la educación.

Palabras clave: Juegos serios, entornos de autoría, juegos geoposicionados, analíticas de aprendizaje

Abstract

In recent years, research has been conducted on how to use video games for purposes beyond mere entertainment. These video games are known as serious games, and applied to teaching, they offer an approach that is attractive to students but also has advantages for educators. For example, students can apply their knowledge, experimenting and learning from their mistakes, or they allow the simulation of real processes in a personalized and safe way for each student. With the rise of mobile devices, their applicability has also expanded, since it is possible to use the students' own devices and serious games can be expanded to new areas such as, for example, outdoor activities for contextualized learning in the real world. This is the case of geopositioned games. This game format has received a major boost in popularity thanks to Pokémon GO, a globally acclaimed game where the main game mechanic is the exploration of virtual elements superimposed on real-world spaces. Despite the potential and benefits of serious games, their use in education is still limited. This may be due to different reasons such as, for example, that the teacher feels that he/she loses control of the class when using them, that it is difficult to analyze whether the students have benefited from the game session or that, in many cases, there is a lack of formal validation of their effectiveness.

The development of a video game is usually a complex and costly process, so that teachers are often limited to selecting existing serious games, even if they do not completely fit their educational needs. Authoring environments are a type of tool that allows lowering the toll that needs to be paid to create a serious game, alleviating the need for technical knowledge, usually by focusing on a specific type of game to be developed. The use of these authoring environments by teachers opens up a range of possibilities, allowing teachers to create their own games or, in the most common case, to adapt existing games to their educational needs. All this at a limited cost, with educational features specifically designed to facilitate and improve the teachers' experience. In recent years the need for evidence on the process and outcome of learning when using technology has led to the collection and analysis of data from the educational process in what has been called learning analytics. Based on this idea, a new field of research has emerged, called Game Learning Analytics, with the aim of not only being able to facilitate the monitoring of students while they play, but also to apply this evidence throughout the life cycle of the game. That is, from the development process, through its deployment and use, and up to the most ambitious objective, which is to be able to use this information to validate the

game scientifically. However, the inclusion of learning analytics and its exploitation is a complex process, which requires specialized knowledge that a game developer, let alone a teacher, does not usually have. Therefore, when introducing a new educational technology, it is necessary to provide tools to help all those involved in the process.

The use of authoring environments simplifies the process by including a set of basic default analytics, which can be used as a starting point to get a general idea about the students' activity in the game. These can then be supplemented with specific custom analytics that can be defined by the author during the creation of the game. Other difficulties commonly encountered by teachers are the complexity of game deployment and integration with other educational tools they already use. This is not a new problem as it has been previously addressed in other fields such as e-learning through the application of educational standards and technical specifications. As part of this thesis work, we analyze how standards can be applied in the context of serious game development to facilitate their application.

All the approaches and models presented have been exemplified in this thesis in an authoring system called uAdventure. uAdventure aims to simplify the creation and actual use of geopositioned games with narrative support in educational environments. This environment takes all the experience of the e-UCM group with a previous system called e-Adventure, and extends it to offer a much richer experience. It provides much more freedom and flexibility through geolocation and the enhancement of specific educational skills in its games through the use of mini-games and extensions. This gaming model is complemented by the incorporation of default learning analytics and multiple integration mechanisms that allow educators to have a more reliable and integrated experience.

The use and application of this authoring environment through experimentation with users carried out in this thesis has validated the approach. That is, proving that this type of solutions simplify the creation of serious games to users without previous experience, favoring the construction of reliable analytics and democratizing their use by non-experts and, finally, simplifying the actual deployment of the games in real environments. Ultimately, this approach helps to mitigate the barriers that teachers encounter when using serious games in a way that can help their generalization in education.

Keywords: serious games, authoring tools, location-based games, learning analytics.

Contenido

RESUMEN	XI
ABSTRACT	XV
CAPÍTULO 1. INTRODUCCIÓN Y MOTIVACIÓN.....	1
1.1. Una historia sobre juegos serios geoposicionados	1
1.2. Motivación.....	3
1.3. Línea de investigación	6
CAPÍTULO 2. ESTADO DEL ARTE	9
2.1. Videojuegos y juegos serios	9
2.2. Creación de juegos serios	15
2.3. Juegos geoposicionados.....	23
2.4. Creación de juegos geoposicionados	32
2.5. Analíticas de aprendizaje.....	35
2.6. Analíticas de aprendizaje y estándares: xAPI.....	40
2.7. A modo de conclusión	43
CAPÍTULO 3. OBJETIVOS	45
3.1. Desarrollo del entorno de autoría	47
3.2. Desarrollo de una extensión para juegos geoposicionados	50
3.3. Implementación de capacidades de integración de los juegos en procesos educativos	51
3.4. Desarrollo de una extensión para géneros híbridos y minijuegos	54
3.5. Validación de la aplicabilidad de uAdventure y de los juegos desarrollados	55
CAPÍTULO 4. DISCUSIÓN INTEGRADORA Y CONTRIBUCIONES.....	59
4.1. Desarrollo de la herramienta de autoría.....	59
4.1.1. <i>Recuperación y adaptación de antiguas funcionalidades</i>	63
4.1.2. <i>Nuevos editores y mejoras visuales</i>	64
4.1.3. <i>Simplificación de procesos de configuración y exportación</i>	65
4.1.4. <i>Mejoras de extensibilidad y extensión de analíticas de aprendizaje</i>	66
4.2. Desarrollo de extensión para juegos geoposicionados	67
4.2.1. <i>Modelo, jugabilidad y autoría</i>	67
4.2.2. <i>Aplicabilidad de juegos geoposicionados</i>	71
4.3. Implementación de capacidades de integración de los juegos en procesos educativos	74
4.4. Desarrollo de una extensión para géneros híbridos y minijuegos	79
4.5. Validación de la aplicabilidad de uAdventure y de los juegos desarrollados	82

4.5.1. Pruebas de la herramienta uAdventure.....	82
4.5.2. Pruebas y casos de estudio con juegos creados con uAdventure.....	89
4.5.3. Otros experimentos como parte de TFGs.....	98
CAPÍTULO 5. CONCLUSIONES Y TRABAJO FUTURO	101
5.1. Conclusiones y principales aportaciones.....	101
5.2. Trabajo futuro.....	106
CAPÍTULO 6. ARTÍCULOS PRESENTADOS.....	111
6.1. Simplifying the creation of adventure serious games with educational-oriented features.....	111
6.1.1. Cita completa.....	111
6.1.2. Resumen original de la publicación.....	111
6.2. Simplifying location-based serious game authoring	126
6.2.1. Cita completa.....	126
6.2.2. Resumen original de la publicación.....	126
6.3. Learning analytics for location-based serious games	136
6.3.1. Cita completa.....	136
6.3.2. Resumen original de la publicación.....	136
6.4. A Tool Supported Approach for Teaching Serious Game Learning Analytics.	146
6.4.1. Cita completa.....	146
6.4.2. Resumen original de la publicación.....	146
6.5. Extending Narrative Serious Games Using Ad-Hoc Mini-games.....	155
6.5.1. Cita completa.....	155
6.5.2. Resumen original de la publicación.....	155
6.6. Democratizing Game Learning Analytics for Serious Games	168
6.6.1. Cita completa.....	168
6.6.2. Resumen original de la publicación.....	168
REFERENCIAS	179

Índice de figuras

Figura 1. Juegos serios: The Oregon Trail (1971, izquierda) y America’s Army (2002, derecha)	10
Figura 2. Componentes de los juegos serios. (fuente: doorendoor design.com)	12
Figura 3. Juegos de aventura (en orden): Life is Strange, The Witness, El Prof. Layton y The Secret of Monkey Island.....	14
Figura 4. Motor de juegos Unity (en orden): Vista por defecto, editor de nodos y ejemplos de assets en la Unity Asset Store.....	17
Figura 5. Ventana principal del editor de e-Adventure y ventana de configuración de objeto de aprendizaje.....	21
Figura 6. Posibilidades del aprendizaje móvil ubicuo o m/u-learning. (autor: Espinoza L. et. al).....	23
Figura 7. Primeros juegos móviles inmersivos. Photopet (izquierda) utiliza la cámara y Zombies Run! (derecha) utiliza el GPS.....	24
Figura 8. Juegos geoposicionados (en orden, Harry Potter Wizards Unite, Pokémon GO, Pikmin Bloom y Jurassic World Alive) y sus mecánicas más comunes: el mapa aumentado (izquierda) y la realidad aumentada (derecha).....	27
Figura 9. Estudiantes juegan al juego geoposicionado “RouteMe!” en el campus universitario utilizando smartphones (Fuente: Lemcke et al., 2015)	29
Figura 10. Juegos serios geoposicionados en la investigación: Frequency1550, MobileGame, Gaius’ Day in Egnathia, RouteMe, QuestInSitu y Tag and Seek.....	32
Figura 11. Ciclo de obtención de resultados a través de un juego serio. (autor: Sacha Panic).....	37
Figura 12. Ciclo de obtención de resultados a través de un juego serio. (fuente: e-ucm.es)	38
Figura 13. El Learning Analytics Model (LAM) y su implementación a través del Learning Analytics System (LAS). Un ejemplo de LAS es la arquitectura TLA de xAPI. (fuente: e-ucm.es)	40
Figura 14. Elementos principales de una traza xAPI.....	42
Figura 15. Esquema del proceso de desarrollo de la tesis.	46
Figura 16. Nuevo editor de escenas de uAdventure con inspector.....	64
Figura 17. Elementos del modelo de juegos geoposicionados de uAdventure y ejemplos en los juegos geoposicionados Pokémon GO (izquierda) y El Señor De Los Arbustos (derecha; hecho en uAdventure).....	68
Figura 18. Editor para las escenas basadas en mapas aumentados de los juegos geoposicionados.	70
Figura 19. Dos mapas de calor representando las distintas trazas de un jugador, incluyendo sus diferentes movimientos (izquierda) y acciones geoposicionadas (derecha).	73
Figura 20. Fotos tomadas durante la realización de workshops para enseñar a realizar juegos serios con el apoyo de uAdventure en la ESD y en la UCM.	84

Figura 21. Collage promocional para el juego serio "El Señor De Los Arbustos".	92
Figura 22. Realización del primer experimento con usuarios con el juego serio El Señor De Los Arbustos.	94
Figura 23. Los diferentes minijuegos de la mansión paranormal. Las imágenes numeradas muestran: 1) la interacción dentro del formato aventura; 2) una pantalla de opciones; 3) minijuego formas y colores; 4) minijuego hacking ético; 5) minijuego anillas locas; 6) minijuego electricista; 7) minijuego laberinto de números y 8) retroalimentación al usuario en forma de estrellas tras completar los minijuegos.	95
Figura 24. El gráfico muestra de forma relativa la cantidad de pasos utilizados en cada nivel de dificultad de los diferentes minijuegos. El círculo interior representa el minijuego concreto mientras que el círculo exterior identifica del 1 al 3 el nivel de dificultad siendo 3 el mayor.	97
Figura 25. Estudiantes juegan al juego geoposicionado Guerra Civil UCM en sus móviles en el campus de la Universidad Complutense de Madrid.	98

Estructura de la tesis

Esta tesis doctoral utiliza el formato de recopilación de publicaciones. Esta recopilación se ha integrado en el capítulo 6, al final de esta tesis. En los capítulos previos se proporciona la línea de investigación que se ha desarrollado, ofreciendo un contexto y un hilo conductor entre las diferentes contribuciones que se presentan.

El documento se estructura de la siguiente forma:

- Capítulo 1. Introducción y motivación
- Capítulo 2. Estado del arte
- Capítulo 3. Objetivos y planteamiento de la tesis
- Capítulo 4. Discusión integradora y contribuciones
- Capítulo 5. Conclusiones y trabajo futuro
- Capítulo 6. Artículos presentados

Al final del documento se incluyen las referencias bibliográficas completas que se han utilizado en este documento.

Capítulo 1. Introducción y motivación

1.1. Una historia sobre juegos serios geoposicionados

Hoy Ana y Javi están muy contentos porque les han dejado llevarse el móvil a clase. Harán una excursión al Real Jardín Botánico y les hará falta usarlo para hacer una actividad. Su profesora Mariola, “la de ciencias naturales”, les había dicho unos días antes que había descubierto una herramienta con la que hacer su propia “búsqueda del tesoro” educativa al estilo de Pokémon GO y que no quería contarles más para no reventar la sorpresa. “Pokémon GO a mí no me gusta, yo soy más de Pikmin Bloom... ¡Has visto qué personajes más monos!”, dice Javi a Ana. “Pues a mí me gusta más Pokémon GO. Aún después de 5 años sigo jugando porque desde que aprendí las tablas de tipos no he hecho más que sacar monedas ganando gimnasios y ahora tengo todos los accesorios... ¡Mira esta gorra con orejas de Pikachu!”, respondió Ana. “Chic@s, ya vamos a llegar, si alguien tiene poca batería en el móvil que me lo diga en la puerta del autobús y le doy una batería externa”.

Al llegar al lugar, Mariola les dice a sus estudiantes que accedan a la web del instituto. “Entrad en el aula virtual y justo debajo de los PDFs de ayer os he puesto un enlace para empezar con la aventura de hoy”. Ana, con su iPhone de última generación es la más rápida de tod@s y al hacer clic en el enlace su móvil empieza a cargar un juego llamado Zombis contra Plantas. “¡Cómo mola profe!” replicó Ana, “Es que ahora que vais tod@s con los móviles vais a parecer Zombis” respondió Mariola entre risas. Mientras que las demás compañeras y compañeros abrían el juego en sus móviles, Ana se adelantó y comenzó a jugar por su cuenta. Su avatar, en el centro del mapa, le recordó al de Pokémon GO, aunque en lugar de Pokémon, un Zombi apareció a su lado. “Cereeebr... Eh, ¡espera! tú eres de los míos...” dijo el zombi. “Necesito que investigues el Real Jardín Botánico porque he oído que existe una cura para el virus Zombi pero sólo las plantas conocen el secreto” continuó el zombi.

Al ver que sus estudiantes comenzaban a tocar las pantallas de los móviles y se iban dibujando sonrisas en sus caras, Mariola miraba sorprendida como sus estudiantes estaban cautivados y casi hasta hipnotizados con la actividad. “Si supieran que he metido un tema entero de biología en el juego no sé si estarían tan felices” pensó Mariola, mientras echaba la vista unas semanas atrás cuando conoció la herramienta. “Anda, mira, dice que no hace falta programar...” pensó. Como Mariola era una profesora de ciencias naturales este dato

fue uno de los que la hizo comenzar a experimentar con la herramienta, y casi sin darse cuenta, en menos de una semana ya tenía lista la primera versión de la actividad. Pensó “y con esto ya quedan definidas las misiones que tienen que hacer mis alumnos y así, además, desde la *app* del profe podré ver qué tal van y saber quién ha terminado las actividades principales”, reflexionó Mariola, “Ellos investigando por su cuenta y yo sabiendo lo que realmente están haciendo. No se me va a descontrolar ni Iván que es un despistado y vive en su mundo paralelo.”

“¡He encontrado una planta que dice que mata los virus!” se escuchó exclamar a Jaime a lo lejos. Ana y sus amigos ya estaban recorriendo el Jardín Botánico y al oír a Jaime se acercaron al lugar. “*Echinacea purpurea es una planta medicinal con propiedades antivirales... Sí, seguro que servirá...*”. Al presionar el botón de recolectar Jaime se enfrenta al minijuego de recolección en el que deberá podar la planta adecuadamente para no dañarla. “¡Sí, lo conseguí, ya tengo la medicina curativa!” dijo Ana. “Yo he utilizado Saúco porque además de ser antiviral decía que es un árbol asociado a la muerte y el Zombi me ha dado un logro especial” continuó Ana, mostrándose orgullosa. “A mí se me acaba de bloquear el móvil”, dijo Jaime. “No te preocupes, sólo reinicia el juego, la partida se guarda sola”, respondió Ana mientras continuaban con la investigación.

Mientras tanto, Mariola utiliza la *app* en su móvil con la que controla la actividad de sus estudiantes y su localización en todo momento. “¡Iván! Menos mal que la *app* me ha avisado que habías salido del área de juego, poco más y te vas de vuelta a casa... ¡¿Es que un cartel gigante en medio de la pantalla no te parece suficiente para darte la vuelta?!” dijo Mariola a Iván. “Además, Iván, aquí dice que vas muy rezagado, tus compañer@s ya han preparado la primera medicina y pronto se van a enfrentar al enemigo final. ¡Céntrate!” dijo Mariola a Iván mientras lo acompañaba al punto donde continuar.

Al terminar la actividad, los estudiantes estaban dichosos y alborotados. “Javi... ¡Cómo has hecho para conseguir más puntos que yo!” dijo Ana. “Pues porque mientras tú te dabas prisa en acabar yo estaba explorando y he completado todas las investigaciones de plantas que había... Me encantan las plantas, ¿sabes?”, respondió Javi. “Estoy deseando ver las fotos que has sacado, Javi” comentó Mariola. Al llegar al instituto, desde su ordenador Mariola entró al aula virtual a consultar los resultados finales de la actividad. “A ver, a ver... Vaya, sí, las notas que propone el juego tienen más o menos sentido... Ana tiene un 8 porque completó las tareas principales... Javi un 9,5 porque completó todo el

juego, pero falló una pregunta de la batalla final...” comentaba Mariola mientras leía las notas. “Y al final hasta Iván completó todas las tareas, es casi un milagro, ¡con lo difícil que es lograr que se concentre en alguna actividad!”

Con el éxito de su actividad, Mariola despertó el interés de otros profesores y profesoras, que empezaron a probar la herramienta y colaboraron para crear nuevas e ingeniosas actividades. Diseñar las actividades educativas como juegos era un desafío un poco más complejo de lo esperado, pero de la colaboración entre profesores y las pruebas con alumnos salían cosas interesantes. Hasta el profesor de informática se sintió motivado y se puso a programar para hacerles minijuegos únicos y divertidos con los que completar sus historias. “Sinceramente, no me esperaba que fuera tan sencillo utilizar los juegos en clase; cuando hice mi tesis hacer esto era un completo infierno...”.

1.2. Motivación

Como hemos visto en el escenario de la historia anterior, los videojuegos pueden utilizarse para otros propósitos que van más allá del ocio. De hecho, los videojuegos, han sido utilizados como herramientas para lograr fines tan diversos como, por ejemplo, enseñar, entrenar, abordar prejuicios, o incluso tratar de cambiar el comportamiento de las personas. Se aplican en distintos dominios y sectores tan importantes como la educación (Wilkinson, 2016), el militar (Smith, 2010), la salud (Kato, 2012) o la publicidad (Cauberghe & de Pelsmacker, 2010). Este tipo de juegos se conoce como juegos serios (Abt, 1987), y la investigación asociada ha permitido demostrar científicamente su efectividad para múltiples propósitos como dentro de la enseñanza de todo tipo de materias como las ciencias o la historia (de Freitas, 2018), el desarrollo de habilidades como la resolución de problemas (Fessakis et al., 2016) o el pensamiento computacional (Kazimoglu et al., 2012), y hasta cambiar la actitud de los jugadores a través de la concienciación (Calvo-Morata et al., 2021). Además de su efectividad, los juegos serios también han probado otra serie de beneficios cuando los comparamos con métodos de formación tradicional, especialmente por su capacidad para representar procesos reales de forma controlada (Cheng et al., 2013), motivando a los alumnos y reteniendo su atención (Prensky, 2001b) a la vez que les permiten fomentar la autoevaluación (Squire, 2008).

Con la generalización de los dispositivos móviles en la sociedad, se ha incrementado el interés en investigar cómo pueden utilizarse estos dispositivos de forma efectiva en la

educación, no sólo promoviendo su uso en las aulas (Afreen, 2014), sino ampliando las posibilidades que pueden ofrecer los juegos serios. A diferencia de los juegos serios que se aplican en las plataformas tradicionales (normalmente PCs), los juegos serios para móviles disponen de una mayor libertad para su uso, pudiendo ser utilizados con la filosofía *anytime and anywhere*, es decir, en cualquier momento y en cualquier lugar, o lo que es lo mismo, de forma *ubicua*. Gracias a su flexibilidad, es común que estos juegos puedan usarse en todo tipo de contextos y situaciones. Son especialmente destacables las posibilidades para llevar a cabo actividades al aire libre conectadas con el mundo real tales como actividades de búsqueda del tesoro (Kjeldskov & Paay, 2007) o rutas histórico-culturales (Huizenga et al., 2009). Esto además permite al estudiante aprender directamente desde su entorno (Arango-López et al., 2017) a la vez que se fomenta la actividad física (Althoff et al., 2016; Southerton, 2014). Este es el caso del género de los juegos geoposicionados, una rama de los juegos inmersivos (Montola et al., 2009) cuyo interés ha crecido mucho últimamente gracias al impulso del mundialmente aclamado videojuego Pokémon GO. Aunque ya había existido interés en la investigación en este género de juegos (Benford et al., 2005; Fu & Hwang, 2018; Huizenga et al., 2009), se puede aprovechar este impulso dado por Pokémon GO para ayudar a su difusión y normalización.

Es en este contexto en el que nace la idea de esta tesis. Aunque en la historia anterior hemos visto cómo podría llegar a ser utilizar este tipo de juegos en el entorno educativo, la realidad aún dista mucho de esa idílica historia. De hecho, es destacable que, pese a los múltiples beneficios demostrados de este tipo de actividades con juegos serios, la aplicación de estos juegos en la educación es todavía muy limitado. Los factores que han determinado esta situación son muy diversos (Jean Justice & Ritzhaupt, 2015; Kenny & Gunter, 2011; Torrente et al., 2010). Abarcan factores desde el punto de vista puramente educativo, al ser herramientas cuyo uso está poco normalizado y pueden generar cierta desconfianza en los educadores que no saben exactamente qué hacen sus alumnos mientras juegan. Hasta desde el punto de vista puramente técnico, al ser contenidos difíciles de crear y mantener para educadores y especialmente difíciles de integrar en el proceso educativo. Por ejemplo, esa posible idea de la historia en la que “se abre un enlace y todo está funcionando” es algo hoy en día prácticamente impensable en el caso de usar juegos serios sobre plataformas nativas como Android o iOS.

En los últimos años, el grupo e-UCM había estado investigando en paliar estas dificultades a través del entorno de autoría e-Adventure (Torrente et al., 2010). Los entornos de autoría, en general, ofrecen una interfaz simplificada de creación de juegos serios que permite a los educadores la posibilidad de crear y adaptar juegos serios en base a sus necesidades educativas sin tener profundos conocimientos técnicos (Karoui et al., 2016; Mehm, 2010). La implicación de los educadores aumenta su confianza en el contenido y la confianza de lo que los estudiantes hacen durante la actividad. Además, también es común que estos entornos de autoría incorporen una serie de funcionalidades orientadas a mejorar la experiencia formativa. Por ejemplo, fomentando el uso de metodologías de diseño como el diseño instruccional que potencian la efectividad de los juegos (Kirkle et al., 2005; Vidal et al., 2019) y de mecanismos que facilitan la aplicación de los juegos en entornos reales y su integración en el proceso educativo (como es el caso de los estándares educativos). De este modo, estos entornos logran facilitar tanto tareas más técnicas, como la distribución de los juegos usando plataformas educativas, así como la realización de tareas docentes, poniendo a disposición del educador datos y conclusiones como puntuaciones que se asocian automáticamente a los alumnos en la plataforma educativa.

Pese a los avances logrados con e-Adventure, la construcción de la herramienta con tecnología java supuso que sus funcionalidades fueran quedando progresivamente obsoletas y que esta herramienta no pudiera evolucionar tan rápidamente como el contexto educativo demandaba. Debido a ello, se decidió construir un nuevo sistema sobre Unity, un motor profesional para la creación de juegos. De este modo, se podrían delegar los aspectos más técnicos del desarrollo de videojuegos y centrar el esfuerzo del desarrollo en la mejora e investigación de aspectos educativos que hicieran los juegos más versátiles, confiables y aplicables.

Una de estas líneas de investigación es el estudio de las analíticas de aprendizaje aplicadas a juegos serios también conocidas por su nombre en inglés como *Game Learning Analytics* (Freire et al., 2016). Estas analíticas de aprendizaje aplicadas a juegos serios se sitúan en la intersección entre la analítica de aprendizaje y la analítica de juego, centrándose en monitorizar las interacciones del estudiante con los elementos del juego. Con ellas es posible aumentar la confianza de los profesores al permitirles entender cómo juegan los alumnos, incluso haciendo un seguimiento en tiempo real de la actividad como se narra en la historia, y utilizar herramientas existentes para extraer conclusiones sobre

el proceso de aprendizaje (Alonso-Fernández et al., 2019; Chaudy et al., 2014). Sin embargo, por su dificultad técnica en muchas ocasiones estas analíticas o no se aplican o se usan sólo al final del desarrollo, donde apenas tienen tiempo para madurarlas y validarlas. Esto hace que no sean útiles ni aplicables en entornos realistas en la mayoría de los casos. Si se promueve su uso a través de entornos de autoría, que las simplifiquen e incluso las automaticen, podremos lograr que los educadores puedan interiorizar más su valor y potencial. Así se fomenta el aplicarlas desde las etapas iniciales de diseño del juego serio, logrando en definitiva que sean más útiles en el proceso educativo e incluso durante el desarrollo del propio juego serio. Además, el enfoque basado en un entorno de autoría permite además proporcionar conjunto de analíticas por defecto (limitado), pero que puede servir como punto de partida para entender qué sucede en el juego. Y esto con un esfuerzo muy limitado por parte del creador del juego, cuando además posteriormente se puede complementar añadiendo analíticas específicas el juego concreto para sacar un mayor partido las mismas.

En este contexto, tras haber estado trabajando en el mundo de las tecnologías móviles e inmersivas a través de la realidad aumentada, me uní al grupo de investigación e-UCM para continuar con la investigación en este entorno de autoría. El entorno eAdventure para crear juegos serios de aventura ha sido uno de los proyectos más destacables dentro del grupo e-UCM, con años de investigación y gran aceptación por parte de la comunidad educativa. La idea de aprovechar esta experiencia previa y fusionarla con el tirón de Pokémon GO para continuar la línea de investigación sobre juegos geoposicionados me cautivó. Dada mi experiencia con Unity, pues había realizado mi TFG en el desarrollo de herramientas para simplificar la creación de juegos isométricos, vi la oportunidad de seguir profundizando en las tecnologías inmersivas en las que había estado trabajando recientemente. Así me propuse investigar sobre cómo el género de los juegos geoposicionados y la aplicación de técnicas como las analíticas de aprendizaje podrían contribuir a promover la difusión y normalización de este tipo de juegos.

1.3. Línea de investigación

En base a la motivación, nuestro principal objetivo es simplificar la autoría de juegos serios geoposicionados con soporte narrativo para usuarios no expertos y mejorar su aplicabilidad en entornos educativos. Esta investigación acerca de la posible simplificación de la autoría y mejora de su aplicabilidad se ha realizado a través del

desarrollo y validación del entorno de autoría para juegos serios uAdventure. Como sucesora de e-Adventure, uAdventure hereda la metáfora de autoría narrativa de alto nivel, además de sus características educativas y las funcionalidades que ayudan a los educadores durante el proceso de creación de juegos. Y las evoluciona para explorar la inclusión de nuevos mecanismos que permitan enriquecer la historia narrativa del juego y aprovechar las nuevas capacidades de los dispositivos móviles. Esta base narrativa, supone una base ideal para el desarrollo de juegos geoposicionados, pues la narrativa es una herramienta transversal tanto para motivar a los alumnos, como para guiarlos durante la actividad e introducirles conceptos. A través de esta combinación de géneros se permite a los usuarios crear *aventuras geoposicionadas* en un ambiente de aprendizaje *ubicuo* - a lo que hace referencia la “u” de su nuevo nombre, uAdventure. Para mejorar la aplicabilidad y difusión de los juegos, durante la investigación se explora el uso de analíticas de aprendizaje y la aplicación de estándares educativos con los que aumentar la confianza y se promueve el uso de este tipo de juegos.

Para poder llevar a cabo nuestro objetivo con el que simplificar la autoría y mejorar la aplicabilidad de los juegos serios geoposicionados hemos dividido el objetivo en cinco subobjetivos más específicos:

1. Desarrollo de un entorno de autoría de base narrativa para usuarios no-expertos, extensible, y con soporte básico para analíticas de aprendizaje
2. Desarrollo de una extensión para crear juegos geoposicionados e híbridos, incluyendo además nuevas analíticas para juegos serios geoposicionados
3. Implementación de capacidades en el entorno de autoría para la integración de los juegos en el proceso educativo (e.g. comunicación, soporte de estándares)
4. Desarrollo de una extensión para géneros híbridos y minijuegos orientada a desarrolladores avanzados
5. Validación de la aplicabilidad de uAdventure y de sus juegos mediante su uso docente, el desarrollo de juegos y su prueba en entornos reales

Capítulo 2. Estado del arte

2.1. Videojuegos y juegos serios

Aunque algunas de las primeras aplicaciones de ordenador fueron juegos, tradicionalmente se considera que los videojuegos nacieron en la década de los 70 fomentados por la popularización de los ordenadores domésticos y ofreciendo una nueva forma de entretenimiento (Kent, 2010). Desde entonces, el mercado de los videojuegos no ha parado de crecer hasta convertirse en una de las industrias más importantes y lucrativas del entretenimiento (Setar & MacFarland, 2012), llegando incluso a considerarse un nuevo medio de comunicación audiovisual. No obstante, debido a sus características (e.g. retroalimentación inmediata, motivación, necesidad de aplicar pensamiento lógico y resolución de problemas), los videojuegos pueden tener otro tipo de impacto en el jugador más allá del lúdico. Por ejemplo, en el videojuego Spore³, creado por Maxis Entertainment, el jugador debe hacer evolucionar a su personaje desde su estadio de célula para dotarlo de más capacidades y posibilidades de supervivencia hasta su final ascensión a las estrellas. Aun siendo un juego destinado al entretenimiento, es necesario que los jugadores comprendan los principios básicos de la evolución natural y la biología (e.g. mutaciones) para que los personajes creados sean capaces de alcanzar el espacio. Si bien los conceptos de entretenimiento y videojuegos siempre van de la mano, debido a su versatilidad los videojuegos se han ido expandiendo progresivamente hacia otros ámbitos de la sociedad, como el educativo, la salud o las ciencias. En estos nuevos ámbitos de aplicación donde los videojuegos juegan un papel más allá del aspecto lúdico, reciben el nombre de juegos serios.

El término juego serio fue acuñado por Clark Abt en 1970, y su definición se ha asentado como aquellos juegos cuyo objetivo principal no es únicamente el entretenimiento (Abt, 1987). Originariamente los juegos serios cubrían un abanico muy amplio respecto a su formato, desde juegos físicos como, por ejemplo, los juegos de mesa, hasta los videojuegos digitales, aunque en la actualidad el término se utiliza habitualmente para referirse a videojuegos (Backlund & Hendrix, 2013; Wilkinson, 2016; Zyda, 2005). Si bien existen formas de utilizar juegos comerciales ya existentes con objetivos más allá

³ <http://www.spore.com/>

del entretenimiento, como en el caso ya mencionado del juego Spore, uno de los aspectos diferenciadores entre juego serio y un videojuego tradicional es que los juegos serios habitualmente son concebidos y diseñados con objetivos pedagógicos concretos (Djaouti, Alvarez, & Jessel, 2011). Además del término juego serio también podemos encontrar en la literatura otros términos como juego educativo o juego para enseñar, que han sido utilizados frecuentemente en la investigación independientemente del propósito del juego (Boyle et al., 2016a), e incluso juegos que han sido diseñados como juegos de entretenimiento y que se consideran juegos serios cuando son utilizados con tal propósito (Backlund & Hendrix, 2013).

Dos de los primeros sectores en adoptar los juegos serios fueron el militar y el educativo (Smith, 2010; Wilkinson, 2016). Posteriormente, los juegos serios se han ido incorporando en todo tipo de ámbitos como en la formación en el puesto de trabajo, el ámbito sanitario, la rehabilitación, la difusión cultural, la política o la publicidad y con propósitos tan diversos que van desde el meramente informativo, el educativo o la concienciación hasta su uso para la formación de personal (Backlund & Hendrix, 2013; Djaouti, Alvarez, & Jessel, 2011). Aunque existen juegos serios desde prácticamente los orígenes de los videojuegos, como por ejemplo The Oregon Trail⁴ (1971), un juego para enseñar la colonización americana (**Figura 1**, izquierda), o Lemonade Stand⁵ (1973), enfocado a gestionar negocios, no fue hasta la década del 2000 en la que finalmente despegó y se generalizó el desarrollo de los juegos serios. El juego serio America's Army⁶, un juego para fomentar el reclutamiento del ejército entre los jóvenes



Figura 1. Juegos serios: The Oregon Trail (1971, izquierda) v America's Army (2002, derecha)

⁴ <https://www.visitoregon.com/the-oregon-trail-game-online/>

⁵ [https://archive.org/details/Lemonade Stand 1979 Apple](https://archive.org/details/Lemonade%20Stand%201979%20Apple)

⁶ <https://www.americasarmy.com/>

estadounidenses supuso un hito al ser “el primer juego serio exitoso y bien ejecutado que ganó la atención del público” (**Figura 1**, derecha) (de Freitas, 2018; Djaouti, Alvarez, & Jessel, 2011; Laamarti et al., 2014). Una prueba de ello es que cuenta con 13 millones de usuarios registrados y ha recibido 33 millones de dólares para su desarrollo (de Freitas, 2018).

El juego America’s Army no sólo afianzó el concepto de juego serio, sino que también ayudó a asentar la idea de que los juegos podían ser “tomados en serio” para todo tipo de propósitos. En el ámbito educativo se utiliza el término *Game-Based Learning* para referirse al aprendizaje basado en juegos, donde los juegos (digitales o no) son utilizados en la educación como parte del proceso formativo (Corti, 2006). Más concretamente, cuando los juegos son digitales se habla de *Digital Game-Based Learning* (DGBL). Prensky dice sobre el DGBL que a través del proceso de juego los jugadores aprenden a superar retos mientras resuelven simulaciones bien diseñadas incrementando su motivación y sus logros (Prensky, 2001a). Además, Prensky identifica ocho categorías de juegos: simulación, rol, tutoriales, puzzles, *exergames* (o juegos para hacer ejercicio), juegos de mesa, juegos de género híbrido y la gamificación, que consiste en añadir elementos de juego como pueden ser puntuaciones, vidas o logros, en aplicaciones que no son juegos (Prensky, 2001b). Corti identificaba en 2006 que el aprendizaje basado en juegos tiene el potencial de mejorar las actividades de entrenamiento gracias a su capacidad para generar mayor implicación y motivación en los alumnos, por la posibilidad de representar roles del jugador y por su repetibilidad (Corti, 2006).

Al mismo tiempo que se ha ido generalizando el uso de juegos serios, también ha florecido una nueva área de investigación que ha permitido identificar y estudiar los beneficios que los juegos serios pueden proporcionar. Algunos de los beneficios que podemos encontrar son: mejora de la concentración durante el aprendizaje (Ronimus et al., 2014), mejora en la interiorización de conocimientos a través del uso de roles (Cheng et al., 2013), ofrecer un entorno controlado de competitividad que promueva la motivación (Cagiltay et al., 2015), atraer la atención de los estudiantes (Cheng et al., 2013), favorecer la autoevaluación a través de una retroalimentación inmediata (Squire, 2008), e incluso prevenir el abandono de los estudiantes (Sancho et al., 2012). Además, se ha demostrado que los juegos serios son aplicables a un amplio abanico de actividades de formación. Por ejemplo, Tobias et al. (2011) analiza varios ejemplos de juegos serios que abordan desde aspectos de transferencia de conocimientos y habilidades relacionadas

con entornos reales; impacto en los procesos cognitivos, tales como el pensamiento computacional (Kazimoglu et al., 2012) o la visualización espacial (Ruipérez-Valiente & Kim, 2020), resolución de problemas; y generación de cambios de actitud y concienciación en los jugadores (Calvo-Morata et al., 2021). Gunter et al. (2006) identifican que los juegos serios pueden ser diseñados para ser efectivos en tres dominios: el cognitivo, centrado en el aspecto intelectual la construcción del conocimiento, las ideas y la resolución de problemas; el afectivo, centrado en los intereses, emociones, actitudes o incluso la concienciación; y el psicomotriz, centrado en habilidades motoras y reflejos. De hecho, Egenfeldt-Nielsen (2006) destaca que los juegos serios pueden aplicarse a cualquier método de enseñanza (conductismo, cognitvismo, construccionismo y sociocultural) y existen ejemplos de juegos aplicables en cada uno de estos métodos. Si bien muchos juegos destinados al entretenimiento pueden ser utilizados en el aprendizaje y obtener beneficios, se ha demostrado que la transferencia del aprendizaje con juegos que no fueron diseñados como juegos serios hacia las tareas académicas puede ser limitada (Blumberg et al., 2013). Anastasiadis et al. (2018) concluye que “los juegos diseñados para incluir objetivos educativos y materias específicas de asignaturas tienen un mayor potencial en el proceso de aprendizaje del estudiante, ya que mantienen el interés y la motivación y, en conclusión, son más efectivos y eficientes”.

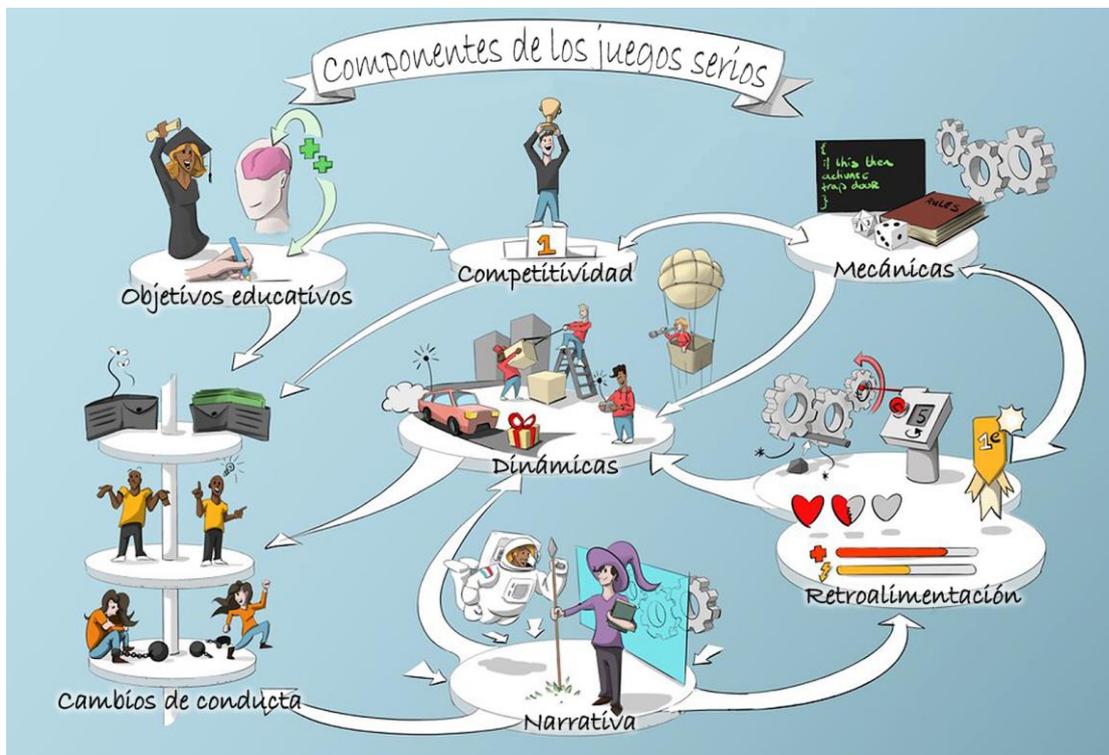


Figura 2. Componentes de los juegos serios. (fuente: doorendoordesign.com)

La gran versatilidad de los juegos reside en su naturaleza para representar situaciones, poniendo al jugador en un rol activo en el que debe de enfrentarse e interactuar con una situación hipotética (Backlund & Hendrix, 2013). Esta interacción en el videojuego puede producirse de formas muy diversas, dependiendo en gran medida del formato del juego, su diseño, sus mecánicas y de la interfaz que proporciona al jugador para ello (**Figura 2**). Por ejemplo, mientras que algunos formatos, como aquellos juegos en tiempo real, son más apropiados para desarrollar habilidades cognitivas tales como reflejos o reconocimiento de patrones (Green & Bavelier, 2015), los juegos que siguen un formato más pausado permitirán al jugador reflexionar, explorar, entender y aplicar sus conocimientos hasta superar la situación (Dickey, 2006). Las mecánicas de interacción que se utilizan en un juego permiten clasificarlos en una taxonomía, comúnmente conocida como género de juego (Aslan, 2016), donde destacan algunos tipos como los de acción, aventura, puzzle, simulación, rol, deporte, peleas y estrategia (Setya Murti et al., 2019).

De entre todos los géneros de juego, los juegos de aventura (**Figura 3**) son especialmente interesantes en el ámbito educativo pues fomentan la reflexión a la vez que ofrecen un buen equilibrio entre la presentación de contenidos y su aplicación para la resolución de problemas (Dickey, 2006). Estos juegos se caracterizan por construirse en torno a una narrativa, donde la historia del juego es su elemento principal, y en la cual normalmente el jugador adoptará un rol concreto. Aunque algunos estudios han encontrado que no hay un impacto directo de la historia en la efectividad del juego (H.-Y. Hsu & Wang, 2010; Wouters et al., 2013), sí se ha demostrado que tienen un impacto directo en la motivación y el rendimiento (Garneli et al., 2017). Los jugadores incrementan su interés ya que la narrativa proporciona un contexto en el que basar sus interacciones y decisiones (Halverson & Owen, 2014) lo que alienta su motivación y rendimiento al perseguir el objetivo de resolver la historia (H.-Y. Hsu & Wang, 2010), incluso perdiendo en algunos casos hasta la noción del tiempo (Dickey, 2005; Rupp et al., 2010). En la historia, el jugador explorará diferentes escenarios donde se encontrará con personajes que pueden representar todo tipo de roles y deberá interactuar con ellos a través de conversaciones para progresar en el juego. Para los juegos de aventura, estos elementos son habitualmente los pilares fundamentales de su jugabilidad, pues dan lugar a un entorno donde el jugador debe tomar decisiones para progresar en la historia.



Figura 3. Juegos de aventura (en orden): Life is Strange, The Witness, El Prof. Layton y The Secret of Monkey Island

En base a los elementos de juego, así como las mecánicas y las formas de la interacción, el género de aventura se subdivide habitualmente en cuatro subgéneros: conversacionales, gráficas, novelas visuales e interactivas. Sin embargo, el subgénero más prolífico y con mayor impacto en el mercado es el de las aventuras gráficas, donde encontramos estilos como los juegos de aventuras narrativas (como los juegos *Life is Strange*⁷ o *The Walking Dead*⁸), aventuras de puzzles (como *El Prof. Layton*⁹ o *The Witness*¹⁰) o los juegos “*point-and-click*” (como *The Secret of Monkey Island*¹¹) entre otros. Por otro lado, otra clasificación interesante basada en la narrativa identifica posibles subgéneros como drama, crimen, fantasía, horror, misterio, ciencia ficción, guerra o espionaje (Grace, 2005).

Como reflexión, elegir el género de aventura gráfica del juego en las etapas iniciales de diseño teniendo en cuenta las mecánicas más apropiadas para lograr los objetivos pedagógicos permitirá obtener una mayor efectividad a la vez que se mejora la experiencia final del jugador. Si bien a priori la elección del género puede parecer que

⁷ <https://lifeisstrange.square-enix-games.com/es/>

⁸ [https://es.wikipedia.org/wiki/The_Walking_Dead_\(serie_de_videojuegos\)](https://es.wikipedia.org/wiki/The_Walking_Dead_(serie_de_videojuegos))

⁹ <https://www.laytonseries.com/naen/>

¹⁰ <http://the-witness.net/>

¹¹ https://es.wikipedia.org/wiki/The_Secret_of_Monkey_Island

limita la creatividad, Aslan y Balci, destacan en su metodología para el diseño de juegos serios que la elección del género no sólo no la limita, sino que es un método recomendado para simplificar el desarrollo y potenciar la creatividad de los desarrolladores (Aslan & Balci, 2015). Además de la elección de un género, Aslan y Balci recomiendan la combinación de géneros ya que puede propiciar la innovación a través de características híbridas. En la siguiente sección analizamos las diferentes posibilidades para la creación de juegos serios.

2.2. Creación de juegos serios

El mercado de los juegos serios es un nicho en crecimiento (Laamarti et al., 2014). Esto se debe a dos motivos: por un lado, el creciente interés científico por estudiar su eficacia en distintos sectores (por ejemplo, el sector militar, que utiliza juegos serios basados en simuladores de vuelo para entrenar pilotos o el caso del juego America's Army); y, por otro lado, al propio auge del mercado de los videojuegos enfocados al entretenimiento. Este auge es especialmente relevante para los juegos serios ya que permite aprovechar las mejoras en el proceso de producción de videojuegos para reducir el tiempo y coste de su desarrollo. Estos procesos se han simplificado mediante herramientas y plataformas gratuitas que permiten reutilizar entornos gráficos, técnicas de desarrollo, motores físicos y mecánicas en la creación de los videojuegos entre otros muchos aspectos. En esta subsección analizaremos las herramientas existentes para el desarrollo de videojuegos, así como las propias herramientas que han surgido dentro del mercado de los juegos serios, explorando sus características y su conveniencia para el desarrollo de juegos serios. También destacaremos aquellas herramientas que han sido identificadas como más relevantes para el desarrollado en la presente tesis doctoral.

Dentro del mercado de videojuegos existen diferentes herramientas para el desarrollo que podemos catalogar como: bibliotecas y motores. Las bibliotecas son componentes que ofrecen funcionalidades con las que simplificar el desarrollo de los videojuegos, como por ejemplo el dibujado de imágenes o modelos 3D. Estas bibliotecas se construyen para ser utilizadas a través de lenguajes de programación o plataformas de desarrollo como

LibGDX ¹² para Java, SDL ¹³ para C, o TreeJS ¹⁴ para HTML5 ¹⁵. Los motores proporcionan una solución integrada para la producción de videojuegos, proporcionando diferentes sistemas que ayuden no sólo a su desarrollo sino también durante el resto del ciclo de vida. Los motores pueden incluir sistemas de renderizado (perspectiva, iluminación, etc.), de físicas, de gestión de interacciones, de publicidad, de pruebas o incluso de analíticas para juegos; y es muy común que incluyan sus propios mecanismos para la creación de mecánicas como lenguajes de scripting o sistemas de programación visual. Dentro de los motores destacamos dos tipos: los motores multipropósito y los motores especializados. Mientras que los motores multipropósito se centran en ofrecer las máximas posibilidades para la calidad visual y la innovación, los motores específicos se centran en ser muy eficaces en el desarrollo de juegos que utilizan géneros o estilos visuales específicos muy frecuentes (como, por ejemplo, el género de juegos de plataformas).

El desarrollo utilizando motores multipropósito se caracteriza por no tener un vínculo específico con géneros o formatos de juego concretos. En general son herramientas de alta calidad (en algunos casos se pueden llegar a utilizar para crear juegos de alto presupuesto llamados informalmente triple A) que reducen los tiempos y costes del desarrollo ofreciendo sistemas reutilizables y configurables para la creación del juego (e.g. sistema de físicas, interfaz de usuario, animación, etc.). Estos motores cuentan con un apartado gráfico flexible y multi género, un sistema de programación orientada a videojuegos que simplifica la creación de mecánicas, incluso en algunos casos proporcionando un entorno visual que simplifica o incluso elimina la necesidad de programar. Estos motores de videojuegos también simplifican el despliegue multiplataforma de modo que el juego desarrollado puede ejecutarse en ordenadores, consolas, móviles y hasta webs. Algunos ejemplos de los más populares de estos motores 3D son Unreal Engine¹⁶, Unity¹⁷, Open 3D Engine¹⁸ (CryEngine) o Godot¹⁹. Además, contribuye a su popularidad que muchas de estas herramientas ofrecen licencias gratuitas,

¹² <https://libgdx.com/>

¹³ <https://www.libsdl.org/>

¹⁴ <https://threejs.org/>

¹⁵ <https://html.spec.whatwg.org/multipage/>

¹⁶ <https://www.unrealengine.com/en-US/>

¹⁷ <https://unity.com/es>

¹⁸ <https://www.o3de.org/>

¹⁹ <https://godotengine.org/>

especialmente apropiadas para los desarrolladores independientes y microestudios, que suele ser el caso más habitual en los desarrollos de juegos serios (Ribeiro et al., 2021). Aunque todos ellos son motores con capacidades muy avanzadas y similares, Unity (**Figura 4**) destaca por haber sido uno de los motores más populares y especializados en dispositivos móviles (según la encuesta de DeltaDNA presentada en el reporte de 2021 de Unity²⁰, este motor es utilizado en el 60% de los juegos gratuitos). Esto lo hace apropiado para juegos serios, pues puede aprovechar la aplicación de políticas como “bring your own device” (Afreen, 2014), a la vez que proporcionan un motor suficientemente optimizado para funcionar en las aulas de informática de los colegios e institutos, que por lo general cuentan con equipos de potencia limitada o incluso están obsoletos (Fidalgo-Neto et al., 2009; Škorić & Milić, 2010). Además, Unity cuenta con una gran capacidad de extensión que permite la creación de herramientas que simplifican

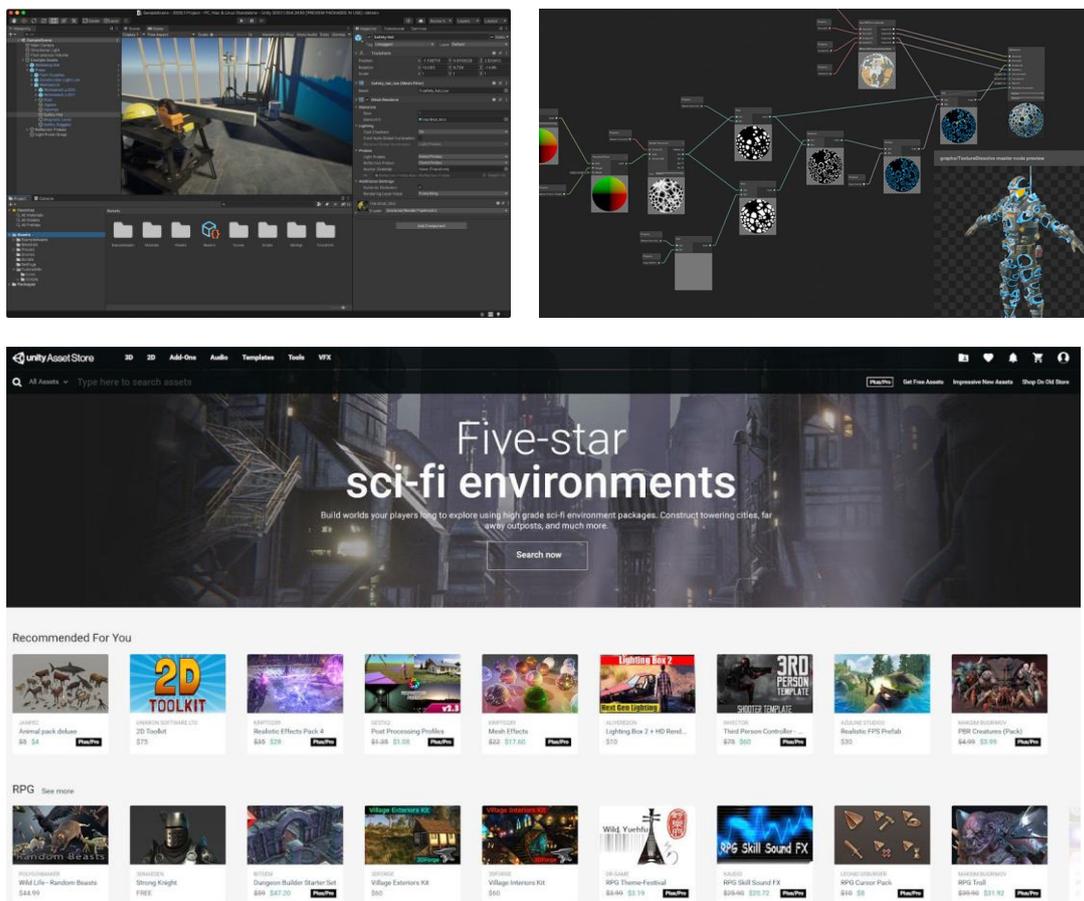


Figura 4. Motor de juegos Unity (en orden): Vista por defecto, editor de nodos y ejemplos de assets en la Unity Asset Store.

²⁰ https://images.response.unity3d.com/Web/Unity/%7B4eb56531-e6aa-492f-8fda-c68ae20af950%7D_2021_Gaming_Report_-_Operate_Solutions.pdf

tareas del desarrollo. En particular, Unity provee a sus desarrolladores de la Unity Asset Store²¹ (**Figura 4**), que contiene todo tipo de extensiones, así como recursos y herramientas que pueden ser utilizados para facilitar la inclusión de contenidos o mecánicas de género concreto. No obstante, aunque herramientas como Unity permiten acortar los tiempos de desarrollo y reducir los costes, el uso de estas herramientas requiere de personal técnico altamente cualificado, pues la mayor parte de las utilidades requieren un profundo conocimiento de Unity para su uso.

Los motores especializados permiten el desarrollo de videojuegos de una tipología concreta. Al ser herramientas especializadas, cuentan con un menor alcance y número de funcionalidades, y como consecuencia de ello tienden a ser más sencillas de utilizar y a proporcionar resultados con una calidad más homogénea. Entre estos motores destacamos dos tipos: aquellos centrados en un tipo de gráficos específico y aquellos centrados en un género concreto de videojuego. Entre los motores de formato concreto podemos encontrar ejemplos tales como GameMaker²² o Cocos2D²³. De entre ellos, GameMaker es un motor especialmente popular (usado para el desarrollo del premiado juego Undertale²⁴) que se centra en el formato 2D, a la vez que provee de mecánicas ya preparadas compatibles con algunos de los géneros propios de este formato como aventuras, puzzles o plataformas. Por otro lado, entre los motores de género concreto podemos encontrar ejemplos en casi todos los géneros, aunque nos centraremos en aquellos de género aventura principalmente. En este género encontramos motores como Adventure Game Studio²⁵, centrado en la creación de aventuras gráficas, RenPy²⁶, centrado en las novelas visuales o Twine²⁷, centrado en la creación de aventuras conversacionales para la web. En estos motores es donde se encuentra el mayor nivel de abstracción, utilizando elementos específicos centrados en el género que simplifiquen la configuración y uso de elementos concretos de juego (como personajes u objetos), ya sea añadiendo una capa de abstracción dentro del scripting del juego o una interfaz visual concreta.

²¹ <https://assetstore.unity.com/>

²² <https://www.yoyogames.com/es/gamemaker>

²³ <https://www.cocos.com/>

²⁴ <https://undertale.com/>

²⁵ <https://www.adventuregamestudio.co.uk/>

²⁶ <https://www.renpy.org/>

²⁷ <https://twinery.org/>

Los motores de videojuegos se utilizan por desarrolladores expertos y estudios de videojuegos para crear juegos serios. Pero su uso por parte de profesionales de la educación es limitado debido a que no son el público objetivo de estas herramientas. Por ello también se desarrollan otro tipo de herramientas enfocadas al ámbito educativo y que entran dentro del sector de las herramientas de autor. Se define herramienta de autor a aquellas herramientas diseñadas para la creación, publicación y gestión de contenidos educativos digitales. Estas herramientas pueden implementar principios de diseño de contenidos educativos (como por ejemplo el diseño instruccional), mecanismos de adaptación o mecanismos de integración para utilizar sus contenidos en la enseñanza (por ejemplo, dando soporte a estándares educativos (Bakhouyi et al., 2017; Lindert & Su, 2016; Papazoglou Papazoglakis, 2013) como SCORM para distribuir los contenidos dentro de una plataforma educativa o LMS) (Ahmad et al., 2020; Mehm, 2010). Sin embargo, a diferencia del prolífico mercado de herramientas para la creación de videojuegos, dentro del mercado de las herramientas de autor existe un número mucho más limitado de herramientas que se orienten en la creación de juegos serios. De hecho, en este mercado, es común que además de las herramientas comerciales se considere el uso de herramientas experimentales desarrolladas dentro del marco de la investigación (desarrolladas por grupos de investigación con dichos fines) (Cowan & Kapralos, 2017).

Algunas de las herramientas de autor comerciales más populares que podemos encontrar son Adobe Captivate²⁸, Articulate Storyline 360²⁹, Gamelearn The Editor³⁰ e ITyStudio³¹. Estas cuatro herramientas comerciales se centran en el género de aventura en el estilo de novela gráfica y cuentan con editores visuales muy simplificados y orientados a ofrecer funcionalidades de alto nivel para reducir los conocimientos técnicos necesarios para utilizarlas. En cuanto a sus características educativas, cuentan con mecanismos de evaluación, así como mecanismos de integración en LMS, como por ejemplo utilizando estándares educativos como SCORM o en algunos casos Experience API (xAPI). Pese a que estas herramientas son muy apropiadas para el desarrollo de juegos serios, en general son muy rígidas y no permiten ampliar los formatos y mecánicas que incorporan, y con

²⁸ <https://www.adobe.com/es/products/captivate.html>

²⁹ <https://articulate.com/360>

³⁰ <https://www.game-learn.com/es/recursos/blog/gamelearn-presenta-el-editor-la-herramienta-de-autoria-que-revolucionara-el-sector-de-la-formacion/>

³¹ <https://itystudio.com/our-product/>

los que poder beneficiarse de otros estilos de juego que no sean el conversacional. Además, para poder extraer los máximos beneficios educativos, la integración con plataformas educativas requiere utilizar habitualmente la plataforma propietaria de dicha compañía, propiciando lo que se conoce como “*vendor lock-in*” (como es el caso de Adobe Captivate y su LMS específico llamado Adobe Captivate Prime³²).

Las herramientas de autor de juegos serios que se crean en el ámbito de la investigación son por lo general, más innovadoras y experimentales. En ellas es común que su construcción tenga como objetivo facilitar la incorporación de nuevos principios de diseño, modelos o estilos de juego enfocados a la educación. Por ejemplo, uno de los ejemplos más recientes es DILEMMA (de Heer et al., 2020), que provee de un modelo para juegos narrativos que motiva a la decisión crítica e integra el uso de analíticas. Puesto que no es el objetivo de esta tesis hacer un análisis extenso de las herramientas de autor narrativas, mencionaremos a continuación los ejemplos que hemos identificado más representativos por el gran volumen de citas que tienen los artículos científicos en donde se presentan y profundizaremos en el caso de e-Adventure, la herramienta que ha inspirado el desarrollo de esta tesis. StoryTec (Göbel et al., 2008) permite la creación de aventuras 3D simplificando el desarrollo con diversos editores visuales que controlan el modelo (como la historia, los escenarios y sus interacciones), sus recursos y su configuración; e incluye funcionalidades específicas para la educación como mecanismos de adaptación al jugador (Göbel, 2017; Mehm et al., 2012). EMERGO (Nadolski et al., 2008) propone una metodología y un conjunto de herramientas para simplificar la autoría de juegos serios narrativos basados en escenarios en primera persona, utilizando una etapa de análisis y diseño *top-down* (de alto a bajo nivel de detalle) para definir estos escenarios utilizando grafos y fomentando el diseño instruccional. En WEEV (Marchiori, Torrente, et al., 2012a), se utilizan también grafos de alto nivel para diseñar la estructura narrativa del juego de forma sencilla y pasar este diseño de forma automatizada a una plantilla compatible con el editor de e-Adventure. Por último, CREATE (Kirkle et al., 2005) es una herramienta que también basa su metodología en el diseño instruccional (bautizándola como SG-IDS) para construir simulaciones narrativas, y también tiene presente el uso de estos grafos.

³² <https://www.adobe.com/es/products/captivateprime.html>

e-Adventure (**Figura 5**) (Torrente et al., 2010) es una herramienta de autoría de juegos de aventura conversacional centrada en el formato 2D “*point-and-click*”. Gracias a que es una herramienta de género muy concreta, los conocimientos necesarios para utilizarla se centran en el uso de un modelo de alto nivel con el que controlar los elementos del juego con propósitos directamente heredados de la metáfora narrativa (Marchiori, Torrente, et al., 2012a). Para ello, e-Adventure incluye editores visuales personalizados para cada tipo de elemento del modelo: personajes, objetos, escenarios o conversaciones. Además de editores para el modelo, e-Adventure incluye un sistema de programación visual dividido en condiciones y efectos. Entre sus capacidades educativas, e-Adventure destaca por centrarse en el formato “*point-and-click*” que además de permitir al estudiante aplicar sus conocimientos con puzzles lógicos y elecciones, favorece las historias no-lineales y la exploración. Otras características educativas de la herramienta incluyen el soporte de estándares educativos, para simplificar la inclusión de los juegos en entornos de e-learning (como IMS CP y SCORM (Bakhouyi et al., 2017)), y mecanismos de adaptación al usuario. Aunque una de las ventajas de e-Adventure en sus inicios era su soporte multiplataforma gracias a que era un motor creado con el lenguaje de programación Java, este hecho ha propiciado el declive de la herramienta debido a la rápida obsolescencia de la plataforma, especialmente en entornos web (Cotroneo et al.,

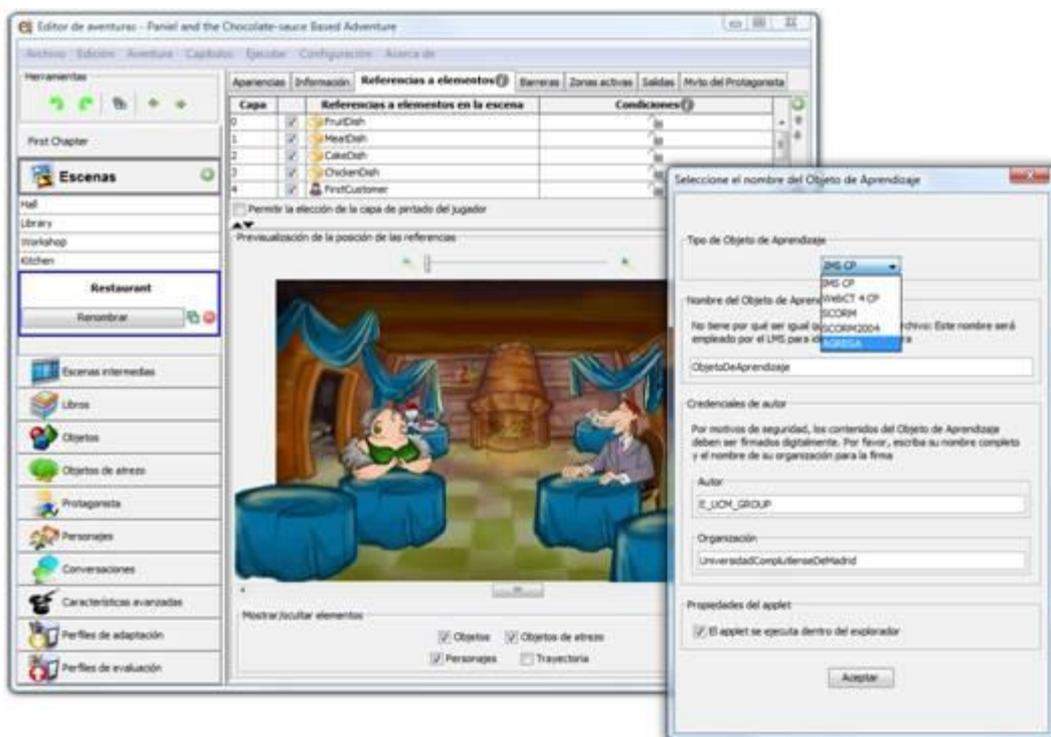


Figura 5. Ventana principal del editor de e-Adventure y ventana de configuración de objeto de aprendizaje.

2007). A este hecho, se suma la complejidad del mantenimiento y desarrollo de la herramienta, con numerosos sistemas propios de motores de juego, editores visuales y mecanismos de integración, en un entorno tan dinámico como es el de la tecnología web y móvil (con varias actualizaciones anuales) lo que hizo que fuera una herramienta difícil de mantener para el pequeño grupo de investigación que la desarrolla.

A pesar de los avances en el desarrollo de los videojuegos y el rápido crecimiento tanto en el mercado de los juegos serios (Djaouti, Alvarez, Jessel, et al., 2011) como en la investigación sobre ellos (Çiftci, 2018) que esto ha conllevado, la adopción de los juegos serios en la educación general es todavía limitada más allá de la investigación o de nichos específicos de mayor capacidad económica (e.g. defensa, negocios, salud) (Jean Justice & Ritzhaupt, 2015; Marchiori, Torrente, et al., 2012a). Los motivos que pueden explicar esta situación son muy diversos, como la complejidad técnica, la falta de fiabilidad, la complejidad y el coste de la validación formal de juegos serios o la dificultad para su integración en el proceso educativo (especialmente en los aspectos de realimentación al estudiante y de soporte de la evaluación de los alumnos para el profesor) (Jean Justice & Ritzhaupt, 2015). Sin embargo, aunque los motivos pueden ser diversos, consideramos que son consecuencia directa de la complejidad intrínseca del desarrollo del juego. Si se quiere conseguir un juego efectivo es necesaria la participación de roles multidisciplinares que aporten tanto conocimientos técnicos como educativos y así poder construir un diseño de juego equilibrado que satisfaga las necesidades educativas y de evaluación del juego (Chaudy & Connolly, 2019). Mientras que el desarrollador se encarga del conocimiento técnico, los educadores son los encargados de aportar la materia y el contenido educativo, así como sus reportes. Además, otros roles como analistas de datos pueden ser necesarios para garantizar la validación de los juegos y su uso como herramientas de evaluación, así como expertos en e-learning pueden favorecer las capacidades del juego para su integración. Debido a estos entornos multidisciplinares, es común que desarrolladores y educadores tengan problemas para ser capaces de transmitir aquellas necesidades o requisitos propios de sus dominios. Por lo que para fomentar la aceptación de los juegos existe una tendencia en la investigación a recomendar la creación de herramientas de autor, pues son las más apropiadas para favorecer que personal no-técnico sea capaz de participar de forma directa en el desarrollo del juego. No obstante, en estas herramientas de autor también es necesario incluir mecanismos que simplifiquen

la aplicación de buenas prácticas de diseño de juegos serios y una mejor integración de éstos en las diferentes etapas de su ciclo de vida.

En la siguiente sección analizaremos el impacto de los dispositivos móviles en el sector de los videojuegos y los juegos serios, que ha propiciado la popularidad de los juegos inmersivos y de la realidad aumentada y la geolocalización (que, a veces, se denominan juegos de realidad extendida). Además, también se analizarán herramientas de autor centradas en los dispositivos móviles y las mejoras que incluyen para solventar algunos de los problemas anteriores y propiciar un mejor ciclo de vida del juego.

2.3. Juegos geoposicionados

Durante los últimos años, las tecnologías móviles se han generalizado en la sociedad y han propiciado nuevas posibilidades a la innovación tecnológica. Con la adopción de los teléfonos móviles inteligentes (*smartphones*), hoy en día son pocos los casos donde la conectividad, la accesibilidad y la innovación no hayan modificado nuestra manera de ver, entender e interactuar con el mundo. Para el sector educativo, las tecnologías móviles han atraído el interés de la investigación con el objetivo de facilitar y mejorar el aprendizaje tanto en la clase como en aquellos contextos más alejados de la enseñanza tradicional (Pachler et al., 2010). Esto supone no sólo que los estudiantes puedan utilizar sus smartphones para acceder a contenidos en cualquier momento y lugar, sino que abre

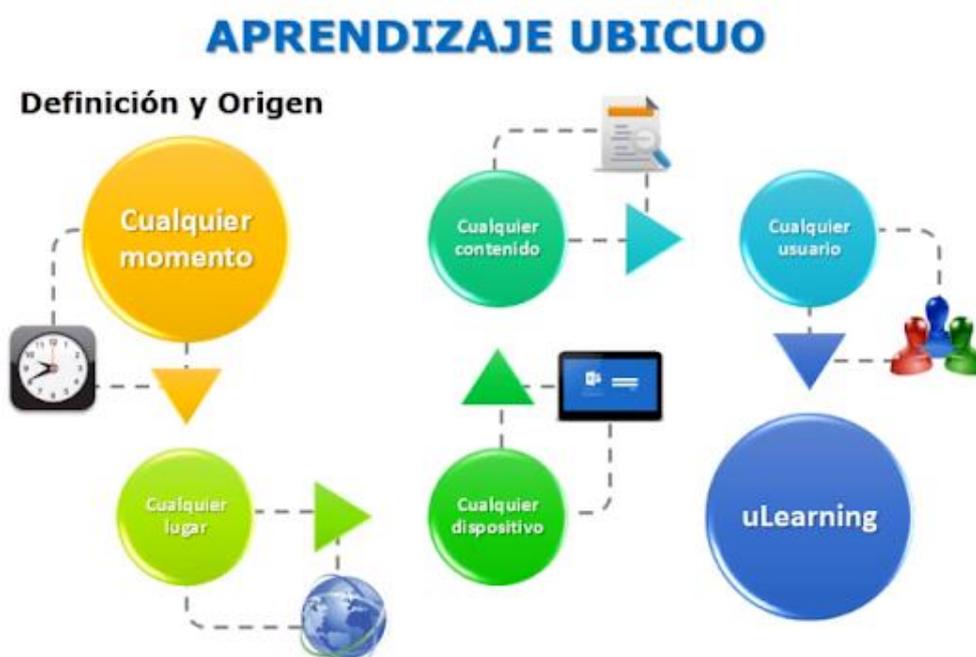


Figura 6. Posibilidades del aprendizaje móvil ubicuo o m/u-learning. (autor: Espinoza L. et. al)

las puertas a nuevos entornos para la enseñanza (**Figura 6**). Según McQuiggan et al. (2015) “el aprendizaje móvil es la experiencia y la oportunidad que ofrece la evolución de las tecnologías educativas. Se trata de un aprendizaje en cualquier lugar y en cualquier momento que permite el acceso instantáneo, y a la carta, a un mundo personalizado repleto de las herramientas y de los recursos que preferimos para crear nuestro propio conocimiento, satisfacer nuestras curiosidades, colaborar con otros y cultivar experiencias que de otro modo serían inalcanzables”. En el aprendizaje móvil (“*mobile learning*”) los estudiantes se alejan de las aulas para participar en actividades formativas más sociales, colaborativas y conectadas, donde prima la relación con el mundo real resultando en uno de los mejores entornos para el aprendizaje (Hansman, 2001). Sin embargo, las tecnologías móviles no pretenden reemplazar el aprendizaje tradicional, sino ser una forma de complementarlo fuera de las aulas que se beneficie de las ventajas de este nuevo formato (Sharples et al., 2010).

A lo largo del tiempo, además del término “*mobile learning*” han aparecido los términos *m-learning* y *u-learning*, que son términos análogos al e-learning para los campos de investigación de la enseñanza móvil y ubicua. Junto con el término “*mobile learning*” comparten tantos aspectos educativos (continuidad entre plataformas, libertad de contexto, aprendizaje autorregulado, etc.) que pueden ser utilizados indistintamente (Pishtari et al., 2020). Desde el punto de vista de investigación, el interés en esta área ha crecido durante los últimos años (Fu & Hwang, 2018) y su adopción está siendo cada vez más amplia en todo tipo de contextos (G. J. Hwang & Wu, 2014; Pimmer et al., 2016). En especial, el campo de los juegos serios sobre dispositivos móviles es una de las áreas con un número creciente de casos y ejemplos de estudio (Kriglstein et al., 2017; Melero



Figura 7. Primeros juegos móviles inmersivos. Photopet (izquierda) utiliza la cámara y Zombies Run! (derecha) utiliza el GPS.

& Hernandez-Leo, 2017; Regal et al., 2018; Volkmar et al., 2018). Para entender los motivos de este creciente interés, debemos entender primero cómo ha evolucionado el sector de los videojuegos móviles, así como los beneficios que estos suponen para la educación.

Los videojuegos móviles (**Figura 7**) hicieron su aparición en la década de los 2000 en los primeros dispositivos móviles para consumidor, tales como teléfonos móviles y computadoras de bolsillo (PDAs), a través de la adaptación de juegos clásicos (como Snake³³, Sonic³⁴ o Arkanoid³⁵) y títulos propios (como Bubble Boost³⁶) que usualmente se inspiraban en el género y el formato de estos juegos clásicos. Sin embargo, la evolución de los juegos móviles ha seguido un curso independiente, pues a la vez que los dispositivos móviles se popularizaron, se iban incorporando en ellos nuevas tecnologías (como WiFi, GPS o Bluetooth) y sensores (como cámaras, acelerómetros o giroscopios) que los hacían cada vez más versátiles para todo tipo de aplicaciones. Estos sensores permitían introducir nuevos estilos de juego innovadores, siendo alguno de los ejemplos más tempranos que podemos encontrar Photopet³⁷ (2004) de Siemens, que hacía uso de la cámara como método de interacción con nuestra mascota virtual, o Pac-Manhattan³⁸ (2004) que utilizaba un sistema experimental de geoposicionamiento en PDAs para simular una partida multijugador de PacMan³⁹. Con la llegada del smartphone a partir de la década del 2010 y el *boom* de la era indie se generalizó que los videojuegos explotaran el potencial tecnológico e innovador que los móviles podían ofrecer. Estos juegos, en origen orientados hacia un público que juega de forma más ocasional (denominados en inglés *casual players*), se caracterizaban por ser conceptos sencillos que hacían uso de las tecnologías presentes en el smartphone para su jugabilidad. Por ejemplo, el juego Zombies Run⁴⁰ utiliza el GPS como método de movimiento, mientras que Trism⁴¹ utiliza el acelerómetro para sus puzles. Gracias a la información disponible en el teléfono móvil, es posible nutrir a los juegos de aspectos provenientes del mundo real, y, en definitiva,

³³ <https://hipertextual.com/2019/02/snake-juego-mas-popular-nokia>

³⁴ https://sonic.fandom.com/wiki/Sonic_the_Hedgehog_Mobile

³⁵ <https://en.wikipedia.org/wiki/Arkanoid>

³⁶ <https://es.phoneky.com/games/?id=j4j5618#gsc.tab=0>

³⁷ <https://es.phoneky.com/games/?id=j4j5619#gsc.tab=0>

³⁸ <https://www.pacmanhattan.com/>

³⁹ <https://pacman.com/en/>

⁴⁰ <https://zombiesrungame.com/>

⁴¹ <https://en.wikipedia.org/wiki/Trism>

del contexto real del jugador, creando así un entorno de juego inmersivo (del inglés *pervasive game*) (Kasapakis & Gavalas, 2015; Montola et al., 2009).

Los juegos inmersivos destacan precisamente por hacer uso de estas variables provenientes del mundo real en su jugabilidad, utilizando el contexto del jugador para su *gameplay*. Kasapakis dijo en 2015 que los juegos inmersivos “dan forma a una excitante y comercialmente prometedora nueva forma de juegos que se construyen sobre la combinación de interfaces híbridas, equipamiento móvil, internet sin cables, sistemas de posicionamiento y tecnologías sensibles al contexto” (Kasapakis & Gavalas, 2015). Poco después, Niantic⁴² lanzó (tras su éxito inicial con Ingress⁴³) el juego inmersivo más popular hasta la fecha, Pokémon GO⁴⁴, que en su primer año registró 232 millones de usuarios y aún en 2020 generó más de 1000 millones de dólares en ingresos superando todos los registros de años anteriores (Iqbal, 2021). Además de haber supuesto toda una revolución en los videojuegos para móvil, Pokémon GO sentó las bases de los juegos que utilizan el geoposicionamiento y la realidad aumentada. Por otro lado, también ha tenido impacto en la investigación debido a la gran cantidad de datos que ofreció sobre los juegos inmersivos y sus implicaciones en el comportamiento físico de los jugadores (Andone et al., 2017; LeBlanc & Chaput, 2017; Wagner-Greene et al., 2017). Dentro de los juegos inmersivos podemos encontrar categorías tan diversas como juegos de resolución de problemas colaborativos, juegos geoposicionados, juegos de realidad aumentada, y juegos que combinan elementos físicos reales y virtuales (Ahlqvist, 2017). El género de juegos geoposicionados es el subgénero más importante para el contexto de esta tesis.

Los juegos geoposicionados es uno de los subgéneros de los juegos inmersivos (Walther, 2005) para dispositivos móviles más populares, con títulos mundialmente exitosos que han atraído el interés de grandes sagas cinematográficas como Harry Potter: Wizards Unite⁴⁵, The Walking Dead: Our World⁴⁶, Jurassic World: Alive⁴⁷, o sagas de videojuegos como Pikmin Bloom⁴⁸ o Minecraft Earth⁴⁹. Por definición, los juegos geoposicionados

⁴² <https://nianticlabs.com/es/>

⁴³ <https://ingress.com/>

⁴⁴ <https://pokemongolive.com/es/>

⁴⁵ <https://www.harrypotterwizardsunite.com/es/>

⁴⁶ <https://www.thewalkingdeadourworld.com/>

⁴⁷ <https://www.jurassicworldalive.com/>

⁴⁸ <https://pikminbloom.com/es/>

⁴⁹ https://es.wikipedia.org/wiki/Minecraft_Earth

(Figura 8) son juegos que utilizan la localización del dispositivo para dar a los jugadores una experiencia contextualizada en el mundo real (la posición y la orientación pasan a ser interacciones del juego); y cuyo punto clave es la mezcla de una experiencia digital y física (Spallazzo & Mariani, 2018). Esta localización puede obtenerse a través de diversas tecnologías tales como GPS, Bluetooth, NFC o códigos QR (Melero & Hernandez-Leo, 2017) e incluso WiFi (Yang & Shao, 2015). Además de utilizar la posición del jugador, los juegos geoposicionados también se caracterizan por utilizar un mapa aumentado en el que los elementos de juego aparecen en lugares específicos del mundo real. A través de sus mecánicas, los jugadores deben ir a lugares concretos, completar retos, responder preguntas e interactuar con objetos virtuales contextualizados. Existiendo, además, la colaboración y la competitividad entre jugadores y equipos (e.g. en Pokémon GO existen tres facciones) (Ribeiro et al., 2021). En este tipo de juegos también están muy presentes los escenarios simulados, con una gran presencia de la realidad aumentada (RA) e incluso la realidad virtual (RV) (Arango-López et al., 2021; Ribeiro et al., 2021). Finalmente, el uso de la narrativa también ha cobrado importancia en este tipo de juegos, pudiendo heredar elementos de juegos de rol, simulación o incluso aventura (Arango-López et al., 2021; López-Arcos et al., 2016; Ribeiro et al., 2021).

En el ámbito de la investigación, el estudio de los juegos geoposicionados ha avanzado de forma similar a la evolución comercial del género. Originariamente las PDAs eran dispositivos comúnmente utilizados con juegos geoposicionados, pero a partir del año

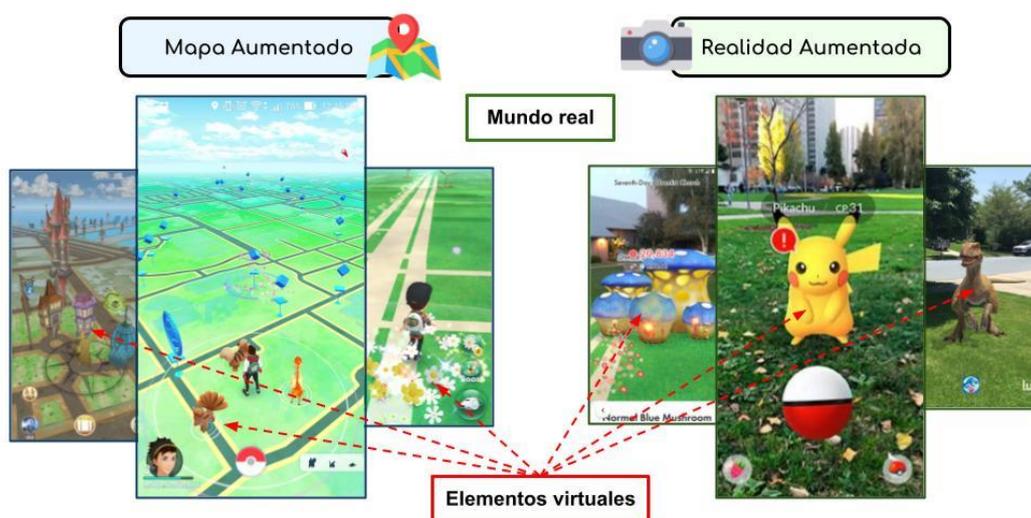


Figura 8. Juegos geoposicionados (en orden, Harry Potter Wizards Unite, Pokémon GO, Pikmin Bloom y Jurassic World Alive) y sus mecánicas más comunes: el mapa aumentado (izquierda) y la realidad aumentada (derecha).

2010 los *smartphones*, *tablets* y dispositivos mixtos (como las gafas Microsoft Hololens⁵⁰) se han convertido en los principales dispositivos utilizados en este tipo de juegos, como se refleja en nueve de cada diez estudios publicados (Chang & Hwang, 2019). Esta tendencia se está generalizando a otros dispositivos como, por ejemplo, los vestibles (*wearables*) o incluso a los robots (Ozdamli & Uzunboylu, 2015). En cuanto a su jugabilidad, la mayoría de los ejemplos que encontramos se centran en el estilo de la “búsqueda del tesoro” o juegos del estilo de seguir el rastro (e.g. *gymkhanas*) (Pánek et al., 2018; Ribeiro et al., 2021; Volkmar et al., 2018). En estos juegos, los jugadores deben utilizar pistas para encontrar, normalmente de forma lineal, determinados puntos de interés, habitualmente con importancia histórica, en los que se ofrecerán determinados contenidos multimedia y en los que deberán completar un reto (por ejemplo, responder un cuestionario, buscar información física del lugar, hacer fotos o videos o incluso completar minijuegos) (Ribeiro et al., 2021). En estos juegos, Karoui et al. (2017) destacan una estructura común recurrente. Esta estructura consiste en una secuencia de unidades de juego que tienen que realizarse en un Punto De Interés (también conocido por sus siglas en inglés, POI). Cada unidad de juego está compuesta de una pista que ayuda a encontrar el POI, un contenido y finalmente una prueba o reto como los previamente mencionados.

Además de los juegos de búsqueda de tesoro, Kjeldskov y Paay (2007) proponen otros cuatro estilos de juego geoposicionado en función del papel que juega la narrativa en ella y su estructura/linealidad: puzzle, dominó, palabras cruzadas (*scrabble*) y caza de mariposas. En los juegos de puzzle, el jugador recibe la historia de forma no-lineal en diferentes ubicaciones y debe utilizar su lógica para ordenarla y resolverla, completando así el juego. Este estilo de juego es apropiado por ejemplo para juegos geoposicionados en museos (Rubino et al., 2015). En el estilo dominó, su estructura narrativa contiene varias tramas lineales que pueden ser recorridas de forma libre a través de las decisiones del jugador (sobre por qué punto empezar o por qué camino seguir, de la misma forma que se amplía el tablero en el juego del dominó) generando una cierta no-linealidad. Además de ser un estilo muy interesante para interiores (pues proporciona una exploración que independientemente de su recorrido tiene un inicio y fin), este estilo es muy conveniente para aportar no-linealidad a los juegos geoposicionados, sin incrementar

⁵⁰ <https://www.microsoft.com/es-es/hololens>

demasiado las posibles rutas físicas en el juego, y optimizando así los tiempos de juego (siempre recomendados por debajo de 1h). El tercer estilo de juego es el estilo *scrabble*, donde los elementos geoposicionados tienen un papel menor en la historia, y son simplemente elementos para resolver un problema mayor. Este sería el caso de Pokémon GO, pues los jugadores recopilan Pokémon y otros elementos en distintas ubicaciones y deberán utilizar su lógica de tipos y utilidades para maximizar sus posibilidades contra retos en gimnasios o para conseguir Pokémon legendarios. Un ejemplo en la investigación, aunque antiguo, sería el videojuego Savannah (Benford et al., 2005), donde los estudiantes son leones y deben sobrevivir alimentándose, utilizando elementos del entorno. Por último, los juegos de estilo “caza de mariposas” son aquellos en los que los eventos de juego no se encuentran enmarcados dentro de una historia con un trasfondo mayor, sino que son simplemente experiencias aisladas de las cuales el jugador extraerá su propio significado. Este es el caso del mundialmente famoso juego geoposicionado “GeoCaching”, en donde los jugadores deben buscar un objeto físico escondido en un lugar público, que contiene un registro y otros objetos que hayan dejado otros jugadores que hayan encontrado el lugar (Neustaedter et al., 2013). Estos dos últimos estilos de juego destacan por su mejor escalabilidad y reusabilidad, al depender menos de una historia y navegación concretas, pero también requieren de una mayor base de usuarios capaces de generar contenido (Lund et al., 2013; Neustaedter et al., 2013) o de un mayor esfuerzo por parte de los desarrolladores para nutrir al juego o a su generación procedural.

Desde el punto de vista educativo, los juegos geoposicionados tienen el potencial de hacer que los estudiantes participen en un aprendizaje auténtico in situ (¡Error! No se encuentra el origen de la referencia.), utilizando juegos de rol y resolviendo problemas complejos del



Figura 9. Estudiantes juegan al juego geoposicionado “RouteMe!” en el campus universitario utilizando smartphones (Fuente: Lemcke et al., 2015)

mundo real (Karoui et al., 2017) y en los que pueden explorarse distintos elementos como el contexto físico, realidades alternativas y mixtas, interacciones sociales, implicaciones temporales y múltiples formatos multimedia entre otros (Arango-López et al., 2017). Al mismo tiempo pueden ayudar a los educadores, al ser herramientas que facilitan el control de la exploración de los usuarios en espacios abiertos (Ribeiro et al., 2021). Además, este tipo de juegos pueden ser aplicados en las aulas y en entornos tan diversos como campus universitarios, museos, edificios históricos, zoos o parques científicos (Chang & Hwang, 2019; Y. C. Hsu et al., 2012) y pueden funcionar con jugadores de todas las edades como podemos encontrar en los diferentes estudios (e.g. niños de primaria, secundaria, universitarios y adultos) (Chang & Hwang, 2019; Ribeiro et al., 2021).

Este potencial de los juegos geoposicionados ha fructificado en numerosos beneficios cuando son aplicados en el marco educativo. Entre estos beneficios, podemos destacar que atraen al jugador y lo mantienen motivado, mejorando el aprendizaje, la diversión (Lloret Irlés et al., 2018; Taborda Mosquera et al., 2021) y satisfacción (Rubino et al., 2015), le ofrecen información contextualizada y real acerca del entorno (Wu et al., 2013), promoviendo la actividad física (Althoff et al., 2016; Southerton, 2014) y favoreciendo el aprendizaje inmersivo y deductivo (Coulter, Klopfer, et al., 2016). En cuanto a las asignaturas en donde se han aplicado más, destacan con diferencia la historia y las ciencias socioculturales (Arango-López et al., 2021) aunque estos juegos se han aplicado también en áreas como ciencia, medicina, geografía, ingeniería, arte o lenguaje entre otros o áreas más innovadoras como el ejercicio físico o el turismo (Ribeiro et al., 2021). Un ejemplo de los beneficios de los juegos geoposicionados para el contexto histórico-cultural es el estudio de Hwang y Chang (2016) que realizó una competición geoposicionada logrando mejorar la identidad cultural local, haciendo el aprendizaje más interesante y con una menor carga cognitiva.

Todos estos beneficios abordan un concepto generalmente asociado a los juegos serios geoposicionados conocido como *authentic learning*. Este concepto representa el hecho de que el aprendizaje se realiza de una manera similar a cuando nos enfrentamos al entorno real (Lucke & Rensing, 2014). Por ejemplo, cuando clasificamos una roca o mineral del suelo, identificamos un comportamiento físico o estudiamos la fauna y flora local. Este tipo de aprendizaje es especialmente enriquecedor para el estudiante, pues al ser un aprendizaje más conectado a la realidad, provoca un aprendizaje más significativo y duradero para el jugador. Por contra, debido a este hecho aflora una de las limitaciones

de los juegos geoposicionados para su replicabilidad, que es su fuerte vinculación con el entorno (especialmente en aquellos donde la narrativa se apoya en éste), haciéndolos muy dependientes a contextos y lugares concretos y haciendo más difícil su reutilización y rentabilidad (Hansman, 2001).

A continuación, analizamos algunos de los ejemplos más relevantes de juegos serios geoposicionados desarrollados en el ámbito de la investigación y que son más informativos para el presente trabajo de tesis doctoral, entre los que se incluyen algunos de los juegos más citados en la literatura (Karoui et al., 2015) y otros que completan el espectro (**Figura 10**). Frequency1550, es un juego que fue utilizado entre 2005 y 2009 centrado en el Amsterdam medieval (Huizenga et al., 2009). Es uno de los primeros juegos en utilizar la historia como elemento de *role-play* para mejorar la inmersión, utilizando el formato de búsqueda de tesoros en su diseño. MobileGame (2005) es un juego que fue utilizado para enseñar el campus a los estudiantes utilizando diferentes misiones cooperativas o competitivas como, por ejemplo, llevar un libro virtual a una biblioteca (Schwabe & Göth, 2005). Gaius' Day in Egnathia (2008) es un juego geoposicionado de rol en el que los jugadores juegan en equipos de 3 a 5 personas y cada jugador tiene un rol diferente de la sociedad Gala. Este juego fue actualizado para smartphones y rebautizado como "Explore!" mejorando su aceptación al incluir elementos 3D (Ardito et al., 2012). Un caso peculiar es el juego RouteMe! (2015) un juego para enseñar conceptos de redes móviles *ad-hoc* en el que los jugadores harán las veces de comunicadores o de nodos intercomunicadores para crear redes e intercambiar mensajes (Lemcke et al., 2015) siendo un ejemplo cercano al estilo scrabble. Otro caso representativo es el de QuestInSitu (2017), un juego simple que sigue el estilo de búsqueda de mariposas, donde los jugadores se encontrarán con pistas que utilizarán para encontrar POIs en la ciudad establecidos por otros jugadores (Melero & Hernandez-Leo, 2017). Este juego destaca porque su contenido depende de su propia comunidad, haciéndolo un juego mucho más escalable. Por último, un ejemplo memorable es Tag and Seek (2014) que utiliza como tecnología de geoposicionamiento exclusivamente NFC (comunicación por proximidad) y es un juego más relajado, orientado al turismo y al público general (Arkenson et al., 2014). Con esta enumeración de juegos se pretende dar una idea de la variedad de las líneas de investigación y aplicación de los juegos serios geoposicionados.



Figura 10. Juegos serios geoposicionados en la investigación: Frequency1550, MobileGame, Gaius' Day in Egnathia, RouteMe, QuestInSitu y Tag and Seek.

2.4. Creación de juegos geoposicionados

Si bien los juegos serios geoposicionados de los que ya hemos hablado han demostrado su eficacia (como Frequency1550 (Huizenga et al., 2009) para enseñar historia de Amsterdam o RouteMe! (Lemcke et al., 2015) para enseñar conceptos de redes *ad-hoc*) y sus múltiples beneficios característicos del género (actividad física, aprendizaje contextualizado en el entorno real, etc.), como ya pasaba con los juegos serios en general, tienen un alto coste de producción. Esto es debido a que son desarrollos específicos hechos para lograr unos objetivos concretos y delimitados (Ribeiro et al., 2021). Además, al ser desarrollos que implican tecnologías complejas y específicas desde un punto de vista técnico, dificultan la participación de expertos del dominio educativo en el diseño y creación de juegos. Karui et al. destacaba en 2016 la necesidad de nuevas herramientas que ayuden a profesores a desarrollar juegos móviles (Karoui et al., 2016). Esto seguía siendo cierto en 2021 como destacan Ribeiro et al., que consideran que esta necesidad se incrementará aún más con la inclusión de nuevas funcionalidades como pueden ser las estrategias de otros géneros de juego o capacidades de realidad aumentada (Ribeiro et al., 2021).

Como resultado de simplificar el uso de tecnologías complejas y facilitar la participación de perfiles educativos en la creación de juegos serios geoposicionados, han surgido diversas iniciativas y herramientas que tratan de paliar esta carencia. Algunas de estas herramientas son:

- MAGIS (Vidal et al., 2019): es un *framework* (o plataforma) construido sobre el motor de videojuegos Unity en el cual se añade una capa de programación de alto nivel que permite la creación de juegos narrativos con ubicación como búsquedas de tesoros. Además, el *framework* incluye una herramienta de autor que automatiza la creación de estos scripts, soporte de realidad aumentada con Vuforia⁵¹ e incluso analíticas automáticas (basadas en la interacción) y personalizables (basadas en el propio scripting del juego), aunque carece de manual que profundice en esto.
- VEDILS (Mota Macías & Ruiz-Rube, 2017): es una herramienta construida sobre App2Inventor⁵² con el objetivo de permitir la creación de juegos serios con capacidades de geoposicionamiento y realidad aumentada. Esta herramienta destaca por el uso de un lenguaje visual para reducir la complejidad de la autoría, además de permitir la recopilación de analíticas de la interacción utilizando Google Fusion Tables⁵³ y MongoDB⁵⁴.
- JEM inventor (Karoui et al., 2017, 2020): es otra herramienta construida sobre App2Inventor que permite crear juegos serios geoposicionados utilizando la estructura de *city tours* (o recorridos turísticos) basada en POIs. Esta herramienta destaca por tener tres niveles diferentes de complejidad de autoría enfocados a diferentes perfiles de menor a mayor conocimiento técnico. En estos perfiles se ocultan parámetros de configuración concretos para reducir la complejidad que se van haciendo visibles en los niveles más avanzados.
- Stqry (antes Mobile Learning Academy): es una herramienta comercial con dos tipos de enfoque en aplicaciones geoposicionadas, las guías, orientadas a crear experiencias geoposicionadas personales, y las visitas, centrados en crear contenidos para explorar en una ciudad. Ambos estilos cuentan con un editor

⁵¹ <https://www.ptc.com/en/products/vuforia>

⁵² <https://appinventor.mit.edu/>

⁵³ https://es.wikipedia.org/wiki/Google_Fusion_Tables

⁵⁴ <https://www.mongodb.com/es>

online con soporte para GPS, Geo-Fences y iBeacons, e incluye analíticas sobre el tráfico de los juegos. Cuenta con una versión gratuita, aunque la mayoría de las funciones son de pago. Aunque esta herramienta no ha sido desarrollada en el contexto de la investigación, se ha utilizado para realizar estudios sobre juegos serios geoposicionados (Ceconello et al., 2015; Edmonds & Smith, 2017).

- TaleBlazer (Medlock-walton, 2013): es una herramienta de autoría online desarrollada en el MIT para juegos geoposicionados con mecánicas de juegos de rol y aventura. Es una de las herramientas con más antigüedad (2012) que siguen siendo usadas hoy en día. Entre sus puntos fuertes destacan el soporte a realidad aumentada (y modelos 3D), su editor visual basado en bloques al estilo de Scratch!⁵⁵ y su buena documentación. Además, es una herramienta desarrollada con la investigación en mente, y ha sido usada para diversos estudios (Cervi-Wilson & Brick, 2018; Fessakis et al., 2016).
- SmartZoos (smartzoos.eu) (Valjataga et al., 2017): una plataforma online y de código abierto para la creación de juegos geoposicionados centrados en zoos originada en Estonia. SmartZoos permite añadir misiones al estilo de juego de “cazando mariposas” utilizando un editor web muy simple. Este hecho ha permitido que uno de los experimentos más recientes de la plataforma utilice a los estudiantes como autores, creando un conjunto de pruebas que sus compañeros deberán jugar, propiciando así no sólo que se aprenda a través de las pruebas sino a través de su creación (Mettis & Valjataga, 2020).

Otras herramientas que merece la pena mencionar por su presencia en la investigación, pero que no son usadas hoy en día son MITAR (Klopfer & Sheldon, 2010; Rosenbaum et al., 2007; Zhang, 2010), desarrollado por el *Scheller Teacher Education Program Lab* y precursor de TaleBlazer (Lehmann, 2013) cuenta con múltiples casos de uso en la literatura científica (como los juegos Environmental Detectives (Klopfer et al., 2002) y Charles River City (Cheung, 2003)), y ARLEARN (Ternier et al., 2012), que fue una de las primeras en utilizar smartphones Android y es la herramienta de autor de este género más citada en el momento de la escritura de esta tesis. Ambas herramientas permiten la creación de juegos serios con uso de geoposicionamiento y realidad aumentada.

⁵⁵ <https://scratch.mit.edu/>

Aunque estas herramientas cubren parte de las necesidades a la hora de desarrollar juegos geoposicionados, Ribeiro et al. también apuntan a la necesidad de aplicar más métodos formales de evaluación sobre juegos geoposicionados (Boyle et al., 2016b; Ribeiro et al., 2021). Para ello, una de las posibilidades más interesantes en la investigación en la actualidad son las analíticas de aprendizaje. A pesar de que MAGIS, VEDILS y Stqry son herramientas que incluyen analíticas, su acercamiento a ellas tiene un enfoque ad-hoc, específico a cada herramienta. El enfoque utilizado por estas herramientas puede limitar su aplicabilidad en el futuro. Un camino centrado en los estándares sería más adecuado para lograr sistematizar las evaluaciones formales y fiables que satisfagan las necesidades de profesores y estudiantes.

2.5. Analíticas de aprendizaje

A pesar de los potenciales y beneficios de los juegos serios, su adopción en la educación es todavía limitada (Barko & Sadler, 2013; Jean Justice & Ritzhaupt, 2015; Kenny & Gunter, 2011). Aunque ya anticipamos algunas posibles causas, como el coste de desarrollo, encontramos además que: por un lado, el propio sistema educativo prima la eficiencia y los buenos resultados y, a menos que una tecnología esté muy probada siendo fiable y sistemática, los profesores serán reticentes a su adopción debido a la desconfianza en la tecnología (Jean Justice & Ritzhaupt, 2015); por otro lado, por su naturaleza puramente tecnológica y su escasa replicabilidad (pues normalmente cada juego lleva un diseño de evaluación personalizado o *ad-hoc*), la complejidad de la integración de los juegos serios en el proceso de enseñanza dificulta en gran medida la evaluación de su eficacia y de su impacto en el proceso de aprendizaje (Muñoz-Cristóbal et al., 2018; Pishtari et al., 2020). Esta problemática de los juegos serios podría abordarse a través de su validación, proporcionando así juegos más confiables, y el uso de mecanismos replicables para su despliegue, proporcionando además a los profesores y estudiantes la capacidad de utilizar los datos de juego para realizar evaluaciones.

En primer lugar, para que esta información sea confiable para los profesores, estos juegos deben seguir un proceso conocido como validación formal que permita verificar científicamente que el juego tiene el impacto deseado en el proceso de aprendizaje. Aunque son muchos los métodos de validación de juegos serios (experimental, quasi-experimental, encuestas/cuestionarios, entrevistas, grupos, observaciones, etc. (Chaudy, 2016)), el método más común de evaluación de juegos serios es el sistema de evaluación

basado en cuestionarios previo y posterior (pre-post) al experimento, donde se establece una correlación entre el uso del juego (y sus posibles resultados) y su efectividad. Para ello se comprueba si ha habido un cambio significativo entre las evaluaciones de la característica deseada realizadas antes y después de su prueba (Alonso-Fernandez et al., 2017). La característica más habitual a la que afecta el juego suele ser el conocimiento del jugador pero puede ser otra como, por ejemplo, la motivación o la concienciación. El uso de cuestionarios pre-post como método de validación, además de ser costoso (requiere una validación formal del instrumento) y de estar limitado al juego para el que se diseña, ofrece poca información a los profesores acerca del proceso de aprendizaje, haciendo que la confianza en ellos sea limitada (Alonso-Fernandez et al., 2017). Por ejemplo, en muchos casos no se da suficiente información sobre el porqué de dicho resultado con el juego.

El problema del despliegue de los juegos serios y su uso para evaluación se resuelve a través de mecanismos de integración. Los estándares educativos ofrecen un mecanismo que además de facilitar esta integración, mejora la aplicabilidad de los juegos al evitar que la solución quede limitada a una plataforma o solución específica. No obstante, para que los estándares educativos puedan ser la solución, los mecanismos que habiliten para la integración de los juegos deben abordar también el aspecto de evaluación, proporcionando una solución que satisfaga las necesidades del proceso educativo. Por ejemplo, uno de los métodos tradicionales para la integración de juegos serios en entornos e-learning se ha realizado a través del estándar SCORM (Papazoglou Papazoglakis, 2013). SCORM es especialmente eficaz en resolver el problema de integración pues es capaz de encapsular los juegos como objetos educativos que se pueden integrar en las plataformas de enseñanza (LMS), recopilando en esta plataforma el resultado final que retorne el juego (Marchiori, Torrente, et al., 2012a).

La evaluación de los estudiantes es una de las bases de la educación, y, según Gipps (2011, publicado originalmente en 1994), consiste en “una amplia gama de métodos para evaluar el rendimiento y los logros de los alumnos, incluidas las pruebas y los exámenes formales, la evaluación práctica y oral, la evaluación en el aula realizada por los profesores y los portafolios”. Visto desde el prisma de los juegos serios, esto significa que, para que un juego serio pueda ser integrado en el proceso educativo, se deben proveer mecanismos a los profesores y estudiantes para que se pueda extraer de los juegos la

información necesaria para realizar un seguimiento de los estudiantes que permita al docente (y si es posible también al alumno) evaluar su uso (**Figura 11**).

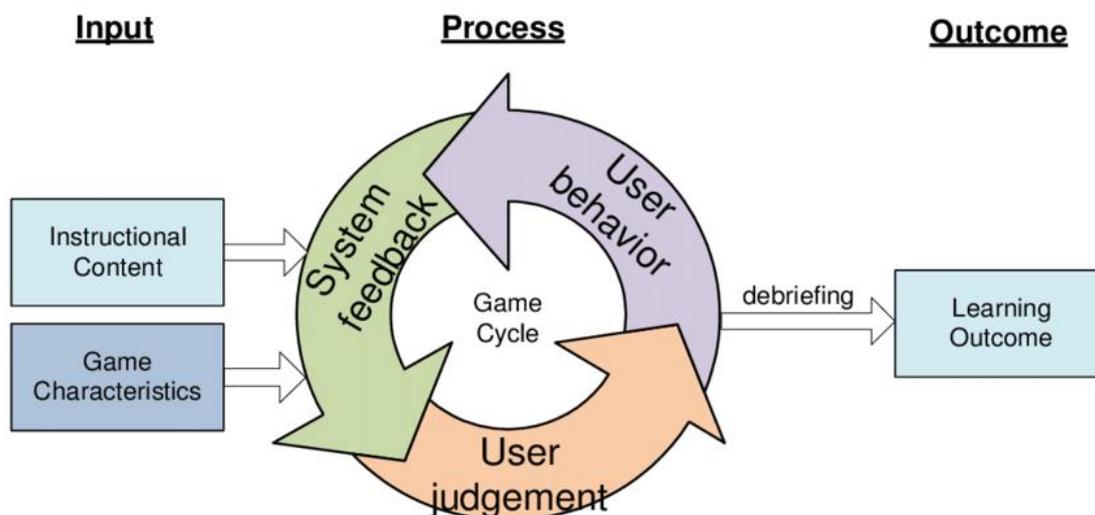


Figura 11. Ciclo de obtención de resultados a través de un juego serio. (autor: Sacha Panic)

Idealmente para que el seguimiento y la evaluación de los estudiantes sean adecuados, según Nicol y Macfarlane-Dick (2006) la información proporcionada por el juego serio debe de ayudar a clarificar el rendimiento, a través de la entrega de datos de calidad, que permita a los estudiantes la autoevaluación y ayuden a alcanzar los rendimientos deseados y motiven el bienestar, y permitan a los profesores ayudar a dar forma a su proceso de enseñanza y al diálogo necesario para orientar a sus estudiantes. No obstante, este objetivo no se cumple para la mayoría de los juegos serios del mercado, pues normalmente sólo disponen de reportes de caja negra que se limitan a mostrar métricas como puntuaciones o tiempos y en los cuales el profesor deberá confiar para integrarlos en su proceso de enseñanza (Marchiori, Torrente, et al., 2012b). Siguiendo con el ejemplo del uso histórico de SCORM para la integración, aunque SCORM supuso un avance en la adopción de los juegos serios (del Blanco et al., 2013; Stănescu et al., 2012), debido a su esquema de caja negra (por sí mismo) es incapaz de permitir una buena evaluación al no ofrecer control suficiente sobre lo que pasa en el interior del juego.

Por tanto, es necesario otra aproximación que ofrezca más información en torno a qué está pasando en el juego y que genere evidencias suficientes para sustentar mejor el proceso de evaluación. La aproximación más prometedora en este sentido y que ha ganado terreno los últimos años en los juegos serios son las analíticas del aprendizaje

(LA, del inglés *Learning Analytics*) (Alonso-Fernández et al., 2019). La sociedad para las *learning analytics* (SoLAR, del inglés *Society for Learning Analytics*) las define como “la medición, la recopilación, el análisis y la comunicación de datos sobre los alumnos y sus contextos, con el fin de comprender y optimizar el aprendizaje y los entornos en los que se produce” (Siemens et al., 2018). Este mejor conocimiento de lo que pasa en el proceso de aprendizaje, permitirá a todos los actores interesados (sobre todo a los profesores) tener más evidencias en las que apoyar el proceso de toma de decisiones del proceso educativo (Calvo-Morata et al., 2019; Persico & Pozzi, 2015). Para los juegos serios, las analíticas de aprendizaje permiten obtener datos (masivos) durante la sesión de juego a través del modelado, trazado y análisis de la interacción (Gibson & de Freitas, 2016) generando datos de la interacción del jugador con el juego como para entender su proceso de juego (e idealmente de su aprendizaje) y, en definitiva, poder inspeccionar el modelo de caja negra del juego. Además de poder utilizarse como método de evaluación del estudiante (Callaghan et al., 2018), las analíticas de aprendizaje también han resultado útiles para procesos de validación de juegos (Calvo-Morata et al., 2021), para predecir la evaluación del estudiante (Alonso-Fernández et al., 2020) e incluso podrían servir para detectar patrones de comportamiento en los jugadores en relación con su adquisición de conocimiento (Schneider & Lemos, 2020).

No obstante, el uso de analíticas en el ámbito de los videojuegos no es nuevo. Existe un área de investigación y aplicación conocida como Analíticas de Juegos (GA, del inglés *Game Analytics*) que está centrada en la captura de datos para usarlos en aspectos relacionados con el impacto del juego desde el punto de vista comercial. Por ejemplo,

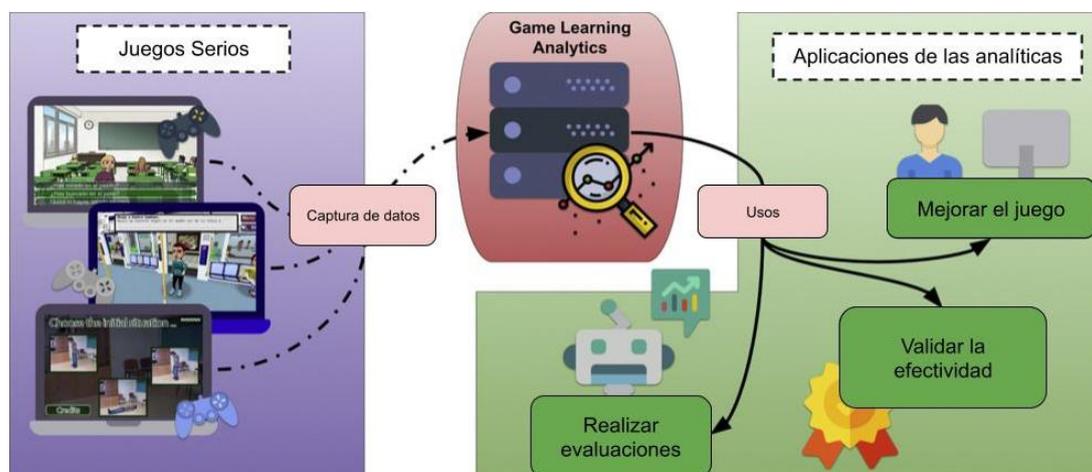


Figura 12. Ciclo de obtención de resultados a través de un juego serio. (fuente: e-ucm.es)

ayudando a la compañía desarrolladora del juego a mantener a los usuarios en el juego, incrementar los ingresos provenientes de cada jugador o simplemente como mecanismo para analizar y depurar problemas en el propio juego.

Recientemente han surgido diferentes iniciativas que combinan las ideas de GA en el contexto de los juegos serios acuñando el término de Analíticas de Aprendizaje para Juegos (GLA, del inglés *Game Learning Analytics*) (Freire et al., 2016). Estas GLA (**Figura 12**) no sólo pueden resultar beneficiosas desde el punto de vista educativo haciendo el proceso de evaluación del aprendizaje más transparente, sino que también pueden utilizarse en otros aspectos del ciclo de vida del juego serio. Por ejemplo, ayudando a mejorar el diseño del juego al hacer aflorar aquellas partes donde los estudiantes se queden atrapados (Freire et al., 2016). No obstante, para sacar el máximo provecho de las GLA, su diseño e integración debe comenzar en las etapas de concepción y diseño del juego, modelando la interacción del jugador y generando datos que contendrán la información necesaria para poder realizar distintos tipos de análisis y relacionarlos con el diseño y el modelo educativo (Hauge et al., 2014; Persico & Pozzi, 2015). Una vez que tenemos la capacidad para poder recopilar los datos, es necesario explotarlos. Para aprovechar las analíticas podemos encontrar dos aproximaciones (no excluyentes) en la literatura: la visualización de los datos y el *data-mining* (en castellano, minería de datos) (Alonso-Fernández et al., 2019; Chaudy et al., 2014). Por un lado, la visualización de los datos busca ofrecer representaciones gráficas de los datos con el objetivo de permitir a los usuarios finales de estas visualizaciones (e.g. científicos de datos, o incluso profesores) extraer conclusiones (Chittaro, 2006) sobre algún aspecto del proceso de aprendizaje. Por otro lado, diseñados los procesos de manera adecuada el *data-mining* permite extraer conclusiones de forma automática analizando los datos recopilados (Chaudy et al., 2014) e incluso descubrir patrones o regularidades presentes en los datos (Kantardzic, 2019). Aplicar cualquiera de las aproximaciones requiere del manejo de los datos y de sistemas normalmente complejos.

Además de la inherente complejidad técnica del GLA, también se suma la habitual ausencia de roles específicos en los estudios de videojuegos capaces de afrontar la tarea de diseñar y mantener estos sistemas, haciendo que la adopción de las analíticas de aprendizaje haya sido limitada. Para mitigarlo, la construcción de un modelo de analíticas de aprendizaje (LAM, del inglés *Learning Analytics Model*) (¡Error! No se encuentra el origen de la referencia.), que relaciona el diseño educativo del juego con los datos de las

analíticas permite, al menos en parte, generalizar el diseño y reutilización de sistemas para analíticas de aprendizaje (Freire et al., 2016). Este modelo identifica las etapas de recolección, agregación, reporte, evaluación y adaptación necesarias para proporcionar soporte a todos aquellos interesados en el ciclo de vida del juego. Además, contribuye a establecer unas expectativas más realistas sobre el proceso y resultado de la aplicación del GLA y sobre todo a establecer una colaboración entre diseñadores, educadores, desarrolladores y analistas (I. Perez-Colado et al., 2018).

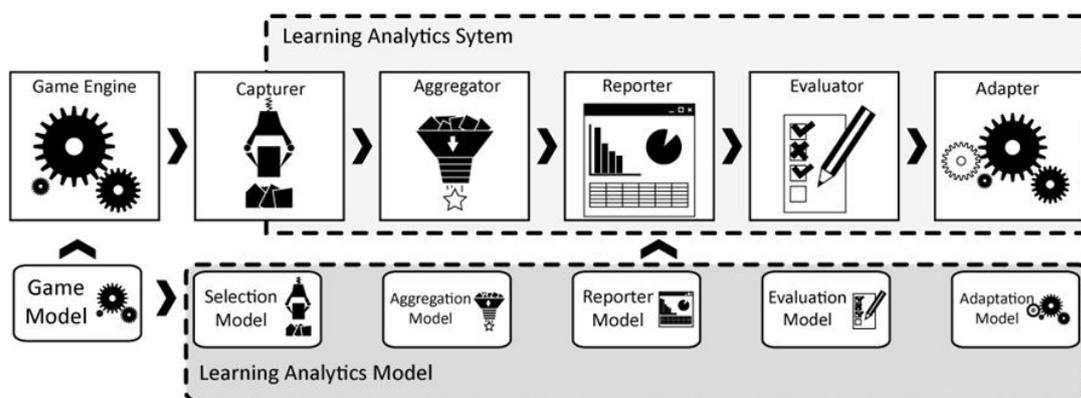


Figura 13. El Learning Analytics Model (LAM) y su implementación a través del Learning Analytics System (LAS). Un ejemplo de LAS es la arquitectura TLA de xAPI. (fuente: e-ucm.es)

2.6. Analíticas de aprendizaje y estándares: xAPI

Para facilitar la construcción del LAM y de las herramientas tecnológicas que lo soportan (herramientas que forman el LAS), otro de los puntos clave es el uso de un formato común de intercambio estandarizado para los datos recopilados. Por un lado, el uso de un formato estandarizado nos permite la integración entre herramientas desarrolladas por diferentes proveedores y, por otro lado, mitigamos problemas previamente mencionados como el “*vendor-lock-in*”. Las iniciativas más prometedoras en esta área son eXperience API (xAPI) y Caliper Analytics (IMS Caliper). Ambas iniciativas proporcionan un formato con el que construir modelos de analíticas basadas en los eventos de la interacción y un protocolo para facilitar la comunicación de estos eventos entre plataformas generadoras y almacenes de datos.

Hoy en día consideramos que la especificación xAPI es uno de los formatos para modelar *learning analytics* más prometedores (Lindert & Su, 2016). Esto es debido a que no sólo

proporciona un formato flexible y sencillo, que se centra en la captura de la interacción, sino a que además se basa en una tecnología muy probada en redes sociales (como es *Activity Stream* (Guy et al., 2011)) y además cuenta con un ecosistema rico que ha surgido a su alrededor. Uno de estos ejemplos de especificación para la construcción de ecosistemas de analíticas de aprendizaje es la *Total Learning Architecture* (TLA) promovida por ADL (Papazoglou Papazoglakis, 2013), que incorpora el diseño de sistemas de analíticas de aprendizaje (LAS) flexibles y adaptables a las necesidades específicas de cada situación. La TLA establece un modelo de madurez de varios niveles para los sistemas en función de sus capacidades y la adopción de sus estándares de interoperabilidad, siendo el primer nivel el nivel básico capaz de capturar y analizar los datos de los estudiantes. En este TLA la pieza central es el *Learning Record Store* (LRS), encargado de recibir las trazas en formato xAPI generadas por los *Learning Record Providers* (LRP) y almacenarlas para que los *Learning Record Consumers* (LRC) puedan acceder a ellas y procesarlas de acuerdo con las siguientes fases del modelo de análisis que proporciona el LAM.

La especificación xAPI surgió como resultado de las carencias identificadas en SCORM, tales como el soporte a las nuevas tecnologías móviles, o la aparición de herramientas altamente interactivas como las simulaciones o los propios juegos serios (Lindert & Su, 2016; Papazoglou Papazoglakis, 2013). Dado que la principal limitación de SCORM es su limitado modelo de intercambio de datos, xAPI aborda esta limitación proporcionando un formato de eventos que pueden enviarse en tiempo real, capaz de representar cualquier evento educativo (Figura 14). Para ello, la especificación define en su formato cuatro elementos principales: un *actor* que realiza la interacción, normalmente el estudiante, vinculado al cliente concreto desde donde se realiza; un *objeto* que recibe la interacción, normalmente un elemento del aprendizaje (tal como un archivo *pdf* o contenido multimedia) o del juego (un botón, objeto o personaje); una *acción* que representa la interacción a través del uso de verbos (e.g. vocabulario controlado); y un *contexto* asociado a la interacción que permite a los sistemas establecer una mejor relación casual y análisis de la traza. Además, las trazas pueden contener un *resultado* en el cual es posible añadir indicadores como si es satisfactorio o no, una *puntuación* o diversas *extensiones* personalizadas. Todas las trazas incluyen una estampación de tiempo (*timestamp*) que facilita realizar los análisis donde se considera el factor tiempo (e.g. duración de una acción o media de tiempo en completar un juego).

Como el formato de trazas xAPI es muy flexible, podría darse la situación en la que cada herramienta utilice la especificación de una manera diferente y, por tanto, limitando el beneficio en la aplicación de esta especificación. En este sentido, la propia especificación define el concepto de *perfil de aplicación* de xAPI, de modo que es posible definir en una comunidad de práctica un vocabulario común, plantillas y patrones de trazas compatibles con la especificación xAPI. Este perfil de aplicación sirve a modo de “lingua franca” en esta comunidad y, finalmente es otro elemento que contribuye a la construcción de LAMs. En este sentido, el grupo e-UCM, en colaboración con ADL, y dentro del contexto de los proyectos europeos del H2020 RAGE y BEACONING que han aglutinado a una muestra relevante de investigadores y de compañías desarrolladoras de juegos serios europeas, han creado un perfil de aplicación para juegos serios (xAPI-SG, del *inglés xAPI Serious Games*) (Serrano-Laguna et al., 2017).

Este perfil de xAPI-SG permite trazar los eventos relevantes del *gameplay* de un juego serio utilizando un modelo conceptual de alto nivel muy cercano a los conceptos usados en el desarrollo de juegos (Serrano-Laguna et al., 2017). Este modelo incluye cuatro categorías principales: *accesibles*, *gameobjects*, *alternativas* y *completables*. Los *accesibles* representan los elementos a los que accede el jugador durante su *gameplay*, incluyendo entre ellos menús, niveles y escenarios. Los *gameobjects* son los objetos del

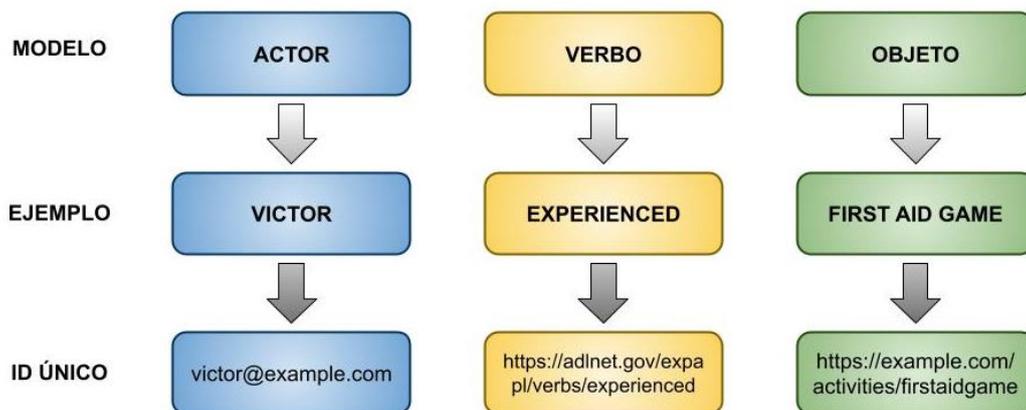


Figura 14. Elementos principales de una traza xAPI.

juego, pudiendo ser personajes, objetos o áreas entre otros; y con los que puede interactuar o ser usados por el jugador. Las *alternativas* representan las elecciones del jugador dentro del juego, como, por ejemplo, los caminos tomados o las respuestas dadas en una conversación. Finalmente, los *completables* representan las diferentes tareas u objetivos del jugador en el juego, así como su estado y progreso. Estos completables tienen tres

estados (inicializado, progresando y completado) y son especialmente útiles para la monitorización, validación y evaluación del juego y los estudiantes. A través del modelo conceptual y del vocabulario proporcionado por este perfil de aplicación xAPI-SG es posible definir un LAM reutilizable y métodos de análisis genéricos para distintos aspectos de los juegos serios de aventura que utilicen el perfil.

Otro de los perfiles más relevantes para el uso de xAPI es CMI5 (Bakhouyi et al., 2017). El perfil de CMI5 proporciona un vocabulario orientado al lanzamiento y comunicación de actividades en sistemas de aprendizaje que implementan xAPI. En este perfil, las actividades son conocidas como unidades asignables (AU, del inglés *Assignable Unit*) y representan unidades de recolección de datos. En el momento de su lanzamiento, CMI5 proporciona a través de protocolos de URL la autenticación del actor, así como el LRS donde se recolectarán los datos. Tras el lanzamiento, CMI5 define una serie de pasos de configuración y, una vez completados, proporciona una serie de mensajes definidos con su vocabulario para comunicar el estado en el que se encuentra la actividad, así como su resultado. Si bien xAPI se ha reconocido como el sucesor de SCORM, realmente es sólo juntamente con CMI5 cuando puede reemplazarlo, al proporcionarle CMI5 la interfaz de comunicación estándar necesaria para el lanzamiento de actividades desde LMS.

Por último, en cuanto a los juegos serios geoposicionados no existe un perfil claro orientado a su uso. Las únicas iniciativas que existen se centran en la definición de la localización del jugador cuando accede a un lugar (Davis, 2020). Para ello, en la sección relativa a las extensiones del contexto de la traza se añaden en formato GeoJSON⁵⁶ los datos de latitud y longitud utilizando los identificadores que xAPI ha reservado⁵⁷. En el caso de querer especificarse cuando un jugador entra a un lugar se utilizará el verbo “*checkin*” identificado en los Activity Streams⁵⁸.

2.7. A modo de conclusión

En este estado del arte hemos explorado los juegos serios, su potencial y beneficios para su uso en la educación, así como las nuevas posibilidades que ofrecen los dispositivos móviles en el área de los juegos serios y cómo pueden aprovecharse desde el punto de

⁵⁶ <https://geojson.org/>

⁵⁷ <http://id.tincanapi.com/extension/latitude> y <http://id.tincanapi.com/extension/longitude> respectivamente.

⁵⁸ <https://activitystrea.ms/schema/1.0/checkin>

vista educativo. Sin embargo, el desarrollo y aplicación actual de los juegos serios en la educación general (especialmente aquellos geoposicionados) sigue siendo limitada y esto se debe en gran medida a su escasa confiabilidad y la dificultad para ser integrados en el proceso de evaluación.

Consideramos que una de las claves que pueden mejorar la generalización de los juegos serios es la mejora y simplificación de su proceso de autoría. Los nuevos motores de videojuegos genéricos han contribuido a mejorar la creación de juegos serios, pero todavía hay mucho margen de mejora. Estos motores siguen teniendo una gran complejidad técnica, que requiere personal bien formado y dificulta el trabajo en equipos multidisciplinares. Además, estos motores no incorporan mecanismos que permitan simplificar la validación de la efectividad del juego ni su integración en el proceso educativo. En general, no es común que haya expertos en tecnologías educativas capaces de usar estos motores de forma eficaz.

Las herramientas de autoría proponen un enfoque más integrado que, por un lado, propicia la participación de roles menos técnicos en el desarrollo al facilitar su comprensión a través de la abstracción de alto nivel a la par que proveen de funcionalidades desarrolladas para mejorar sus capacidades educativas e implantación. En estas herramientas, además, hemos visto cómo al centrarse en géneros específicos de juego permiten a los desarrolladores extraer lo mejor desde el punto de vista educativo de dicho género.

Sin embargo, existen tres factores que podrían mejorarse para aumentar la eficacia de estas herramientas: su mantenibilidad, su versatilidad en el diseño de juego, su integración en el proceso educativo. Con ello, se pretende dar un empuje a su uso a la par que se evita la obsolescencia que caracteriza a todas las herramientas de autoría al tener recursos más limitados para compensar la evolución tecnológica. Este es el ejemplo de e-Adventure, una herramienta demasiado compleja y que por ello ha podido adaptarse a tiempo a los cambios tecnológicos de su plataforma. Mientras que, para mejorar la versatilidad de la herramienta, pueden implementarse nuevos estilos de juego como los juegos geoposicionados, la realidad aumentada o los minijuegos, para mejorar su integración en el proceso educativo será muy interesante la incorporación de analíticas de aprendizaje con xAPI.

Capítulo 3. Objetivos

Como se ha analizado en el capítulo anterior, la tecnología y los videojuegos han evolucionado, ofreciendo nuevas oportunidades y herramientas que pueden ser aplicadas en el contexto educativo mediante la aplicación de los juegos serios. No obstante, como se ha descrito, la creación y aplicación de juegos serios no está exenta de dificultades que deben de ser abordadas durante su proceso de desarrollo e integración en el proceso educativo. En definitiva, esto sucede durante todo el ciclo de vida del juego serio. Más aún, estos problemas y dificultades en el proceso de desarrollo pueden tener un impacto negativo en los propios juegos y, finalmente, en su adopción y generalización en la educación. Creemos que las herramientas de autoría de juego serios pueden paliar los problemas identificados, reduciendo las complejidades del proceso de desarrollo, a la vez que se fomenta la creación de soluciones consistentes y coherentes que contribuirán a mejorar la experiencia de los usuarios y la adopción de los juegos.

Por tanto, el objetivo principal de esta tesis es simplificar el desarrollo de juegos serios geoposicionados aplicables en el contexto educativo. Y este objetivo se concreta con el desarrollo de un entorno de autoría que proporcione una solución integrada. Aunque a priori este objetivo pueda parecer sencillo, puesto que el formato de los juegos geoposicionados es razonablemente simple, en realidad el número de aspectos para tener en cuenta durante el diseño, el desarrollo y la prueba del juego lo hacen complejo de modo que queda fuera del alcance de los educadores. Esta variedad de elementos son los que finalmente determinarán la eficacia del juego serio además de influir tanto en su ciclo de vida como en su aplicabilidad real. Teniendo en cuenta los problemas y limitaciones identificados en el capítulo anterior, este objetivo general se ha dividido en los siguientes objetivos concretos:

1. Desarrollo de un entorno de autoría de base narrativa para usuarios no-expertos, extensible, y con soporte básico para analíticas de aprendizaje
2. Desarrollo de una extensión para crear juegos geoposicionados e híbridos, incluyendo además nuevas analíticas para juegos serios geoposicionados
3. Implementación de capacidades en el entorno de autoría para la integración de los juegos en el proceso educativo (e.g. comunicación, soporte de estándares)
4. Desarrollo de una extensión para géneros híbridos y minijuegos orientada a desarrolladores avanzados

- Validación de la aplicabilidad de uAdventure y de sus juegos mediante su uso docente, el desarrollo de juegos y su prueba en entornos reales

Para facilitar la comprensión del trabajo realizado durante este trabajo de tesis, la **Figura 15** presenta un esquema aproximado del proceso de desarrollo seguido para abordar los objetivos concretos previamente descritos. Este esquema utiliza como base una mezcla entre el desarrollo basado en prototipos y un desarrollo ágil iterativo, donde cada objetivo parte de una implementación básica como prototipo, se posteriormente se desarrolla en una versión completa sobre la que se continúa realizando tanto mantenimiento correctivo como evolutivo con las mejoras (típicamente obtenidas de las pruebas con usuarios). Como podemos ver, a diferencia del resto de los objetivos, el 5º objetivo se ha llevado a cabo durante las diferentes etapas del desarrollo, incluyendo diversos casos de prueba como herramienta docente y con el desarrollo de juegos, con sus correspondientes pilotos y pruebas con usuarios reales que expondremos más adelante.

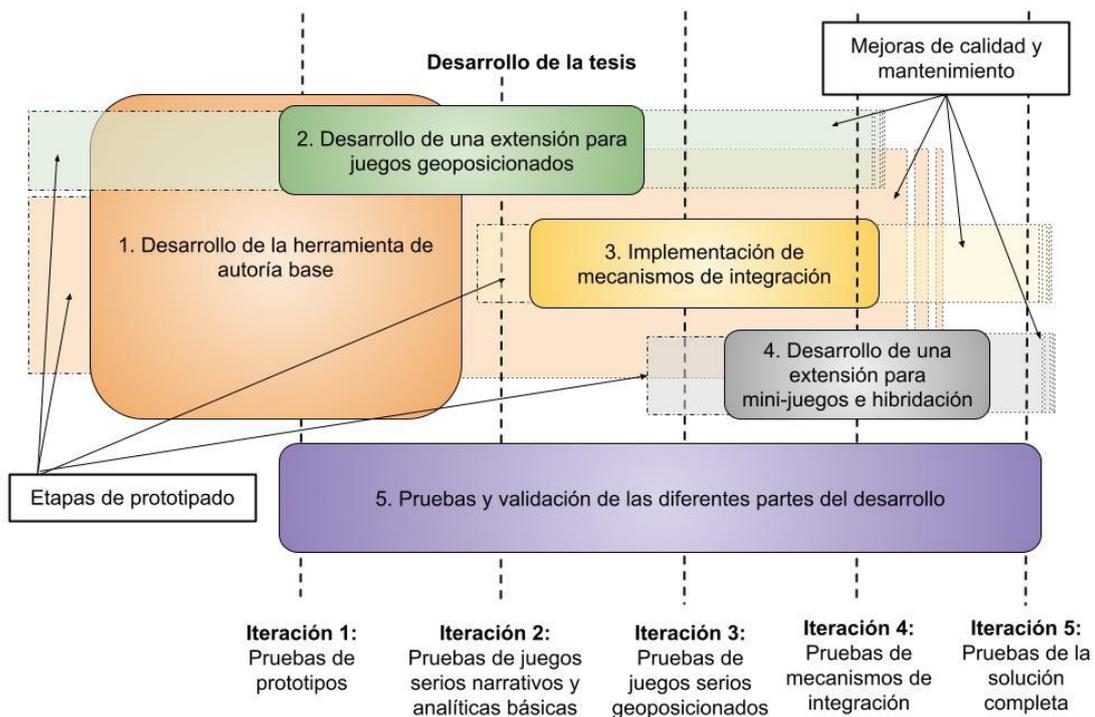


Figura 15. Esquema del proceso de desarrollo de la tesis.

En los siguientes apartados analizaremos en detalle cada uno de los objetivos concretos propuestos. Se abordan tanto los aspectos más conceptuales y de requisitos, como las decisiones de diseño y otras más técnicas que se han tomado en la herramienta finalmente desarrollada.

3.1. Desarrollo del entorno de autoría

El primer objetivo de la tesis es el desarrollo de una herramienta de autoría para juegos serios narrativos se ha visto reflejado en el entorno de creación de juegos denominado uAdventure. Esta herramienta sirve como base sobre la que se han construido las soluciones tecnológicas que abordan el resto de los objetivos concretos planteados en este trabajo de tesis. No obstante, cabe destacar que uAdventure se construye como una evolución de un sistema previo basado en tecnología Java denominado eAdventure. Partiendo del análisis del estado del arte, y con el objetivo de extraer los principales requisitos y características de la herramienta, hemos planteado cinco preguntas para guiar el diseño de la herramienta: ¿quién utilizará la herramienta?, ¿qué género de juego es el más apropiado?, ¿qué flexibilidad va a ofrecer la herramienta (tanto para nuestros propósitos como de cara al futuro de la herramienta)?, ¿cómo vamos integrar los juegos resultantes en el proceso educativo? y ¿quién va a mantener la herramienta cuando finalice su desarrollo?

Para definir el público objetivo nos hemos basado en el proceso típico de desarrollo de los juegos serios y en la experiencia previa con eAdventure. En el desarrollo de juegos serios, según lo expuesto en el estado del arte, es necesaria la colaboración entre expertos del dominio, es decir del tema que aborda el juego, expertos en cómo abordar educativamente ese juego (e.g., profesores) y desarrolladores de juegos para conseguir materializar los objetivos educativos del juego serio (Aslan & Balci, 2015; Marchiori, Torrente, et al., 2012a). De estos tres roles, consideramos que los educadores deben jugar un papel clave ya que deberían participar tanto en el desarrollo como en la aplicación de los juegos serios ya que, a nuestro entender, son los principales actores para la generalización de los juegos en la educación. Por tanto, creemos especialmente relevante proporcionar una herramienta de autoría que oculte los detalles más técnicos del desarrollo del videojuego, simplificando o incluso eliminando los aspectos relacionados con la programación de modo que sea más accesible para ellos (y si no crear el juego desde cero sí que, al menos, fueran capaces de modificarlo o adaptarlo a sus necesidades particulares).

En cuanto al género de sus juegos, se ha destacado en el estado del arte la importancia de la narrativa como uno de los pilares fundamentales en la efectividad y la inmersión de los juegos serios. El género narrativo tiene una serie de ventajas educativas (e.g. promueve

razonamiento frente a ritmo de interacción) y además permiten ser extendidos fácilmente para construir un juego geoposicionado (Arango-López et al., 2021). Por lo que la herramienta tiene como requisito la incorporación de elementos del género narrativo que permitan introducir historias no-lineales.

Por otro lado, y aun partiendo del soporte narrativo, es importante no restringir excesivamente el género e idealmente permitir la extensión de la herramienta (y con ello la introducción de nuevas mecánicas y dinámicas). Como se ha identificado en el estado del arte la mezcla de géneros mejora tanto las capacidades educativas del juego serio como la experiencia del jugador (Aslan & Balci, 2015; Setya Murti et al., 2019). De hecho, esta rigidez a un género concreto es una de las grandes carencias de la mayoría de las herramientas de autoría tanto comerciales como desarrolladas en el ámbito de la investigación. Por tanto, la herramienta tiene como requisito ser fácilmente extensible para permitir que los desarrolladores expertos puedan incorporar nuevas dinámicas y mecánicas de forma sencilla, simplificando el desarrollo de nuevas funcionalidades y sin aumentar significativamente la complejidad de su mantenimiento.

Para responder a la pregunta de la integración de los juegos en el proceso educativo, consideramos que las analíticas de aprendizaje pueden aportar flexibilidad y transparencia de los que carecían otros enfoques apropiados para juegos serios como el uso de SCORM (Lindert & Su, 2016). En este sentido, la herramienta de autoría debe proporcionar tanto un soporte de analíticas básico por defecto, con una intervención mínima por parte del creador de juego serio, así como la introducción de analíticas específicas personalizadas que puedan aprovechar al máximo las capacidades interactivas del juego serio (aunque en este caso se requerirá una mayor intervención por parte del desarrollador del juego).

Finalmente, el mantenimiento tanto de la herramienta desarrollada como de los juegos creados con la herramienta es crucial para incrementar el impacto en el tiempo. Por un lado, para facilitar el mantenimiento y supervivencia de la herramienta es de especial importancia que la herramienta se construya sobre una base sólida que permita a los desarrolladores de la herramienta centrarse en los aspectos de mayor valor añadido que proporciona la herramienta a los profesores. Por otro lado, habitualmente las herramientas y, especialmente, los contenidos educativos desarrollados por los profesores tienen ciclos de vida muy largos. En este sentido, proporcionar a los educadores herramientas que permitan amortizar el esfuerzo invertido en dominar una herramienta y que los juegos

creados puedan utilizarse a lo largo del tiempo es un aspecto muy relevante. Teniendo en cuenta estos dos aspectos, nuestro último requisito es que esta herramienta sea implementada sobre un motor de videojuegos profesional, lo que mejorará la calidad de los juegos, permitirá delegar gran parte de los requisitos técnicos al motor, simplificando la evolución para adaptarse a los cambios en la tecnología y reduciendo en gran medida el esfuerzo en mantenimiento lo que también permitirá prolongar la vida útil tanto de la herramienta desarrollada como de los videojuegos creados.

Como se ha mencionado previamente, aunque uAdventure es un desarrollo completamente nuevo y desde cero se basa en la amplia experiencia adquirida por el grupo e-UCM la herramienta e-Adventure (Torrente et al., 2010), así como en el análisis de las fortalezas y las debilidades de dicha herramienta. De hecho, el nombre de la nueva herramienta de autoría, uAdventure, es una evolución de e-Adventure representando el concepto de la *ubicuidad* (característica de los juegos inmersivos) de la enseñanza que es posible gracias a su gran flexibilidad para adaptarse a nuevos entornos como, por ejemplo, el de los juegos geoposicionados. Reutilizar la metodología de autoría de e-Adventure nos permite partir de un concepto ya probado en el entorno educativo que cumple con nuestros dos primeros requisitos, pues es una herramienta orientada a la simplificación del desarrollo y mantenimiento de juegos narrativos de aventura para los usuarios no-expertos. Esta reimplementación, además de suponer una nueva oportunidad para ofrecer mejores editores en base a las tendencias del mercado, se centra en convertir a e-Adventure en una herramienta más fácil de extender y mantener en el futuro, cumpliendo así otros dos de nuestros requisitos. Además, ha permitido una mejor implementación de nuevas funcionalidades y un mejor soporte de las analíticas de aprendizaje.

Teniendo en cuenta todas estas consideraciones, se ha decidido construir uAdventure sobre el motor de videojuegos Unity, delegando en este motor la evolución y mantenimiento de los sistemas técnicos propios de videojuegos (tales como el renderizado y la conectividad), y adaptando los editores al estilo de Unity como, por ejemplo, los editores basados en nodos y la noción del inspector de objetos, pero simplificando los aspectos y mayor complejidad. Gracias al soporte de Unity y a su arquitectura modular y extensible, uAdventure puede incluir mecánicas no sólo de aventura gráfica, sino de cualquier género, al ser este un motor multipropósito. Para ello, uAdventure incluye mecanismos para ampliar estas mecánicas modificando el formato de juego e incluir nuevas funcionalidades sin perder algunas de sus características

educativas (e.g., analíticas). Otra de las ventajas de desarrollar sobre Unity es que permite un alcance multiplataforma, lo que simplifica el desarrollo en aquellas plataformas disponibles en los centros educativos (Windows⁵⁹, Mac⁶⁰ y Linux⁶¹), y en las más habituales para los juegos inmersivos (Android⁶² e iOS⁶³) e incluye además soporte genérico para tecnologías web (WebGL⁶⁴ sobre HTML5).

Para abordar el soporte de analíticas de aprendizaje utilizaremos xAPI (Freire et al., 2016), el sucesor natural de SCORM, y su perfil para juegos serios xAPI-SG (Serrano-Laguna et al., 2017), que habrá que extender durante el siguiente objetivo para dar soporte a juegos geoposicionados. Para terminar y con el objetivo de permitir nuestro objetivo de desarrollo autónomo y uso por terceros, se desarrollará un manual completo para la herramienta y todas sus funcionalidades. Además, se creará un repositorio básico con juegos desarrollados para poder aprender de ejemplos implementados siguiendo buenas prácticas.

3.2. Desarrollo de una extensión para juegos geoposicionados

El desarrollo de juegos geoposicionados es uno de los principales objetivos para esta tesis. Por ello, una vez completada la herramienta base de autoría uAdventure, se extiende con nuevas funcionalidades para la implementación de juegos geoposicionados. Dentro de este objetivo contamos con dos apartados, la creación de la herramienta de autoría para juegos geoposicionados y la extensión de las analíticas de aprendizaje para cubrir las nuevas necesidades de este género de juegos geoposicionados.

Para el diseño de la parte de autoría, según nuestra revisión del estado del arte, hemos encontrado que la mayor parte de los juegos geoposicionados siguen la forma de la búsqueda de tesoros, aunque existen muchos otros estilos de juego como puzzles, dominós o cazas de mariposas (Kjeldskov & Paay, 2007). El diseño del modelo pretende ser capaz de representar el esquema principal de los juegos de búsqueda de tesoros que incluye pistas, puntos de interés y retos (Karoui et al., 2017). Pero en lugar de optar por un modelo rígido, uAdventure amplía estos conceptos para encajar mejor dentro del género de

⁵⁹ <https://www.microsoft.com/es-es/windows>

⁶⁰ <https://www.apple.com/es/macOS/>

⁶¹ <https://www.linux.org/>

⁶² https://www.android.com/intl/es_es/

⁶³ <https://www.apple.com/es/ios>

⁶⁴ <https://www.khronos.org/webgl/>

aventuras y utilizar la naturaleza flexible de este entorno para permitir también otros estilos de juego. En la revisión también se encontró que el principal formato del escenario de los juegos geoposicionados es el mapa aumentado en el que se incluyen los diferentes elementos tanto reales como virtuales. Estos elementos presentes en el mapa representan los “puntos de interés” identificados en el esquema anterior, pero para ser más flexibles se representan en uAdventure como elementos geoposicionados en general. De este modo, se contempla no sólo cualquier tipo de punto de interés propiamente dicho sino también cualquier elemento relevante para el mapa de juego, tales como edificios y regiones reales, o personajes y objetos virtuales. Por último, para mejorar la jugabilidad, es necesario definir y dar soporte a las diferentes mecánicas para el juego geoposicionado, tanto para cuando este ocurra en exteriores como en interiores, soportando diferentes tipos de posicionamiento para cada caso.

El segundo apartado de esta extensión de juegos geoposicionados es el de proporcionar a estos juegos la posibilidad de incluir trazas de interacción de los usuarios para dar soporte a analíticas de aprendizaje. Sin embargo, como vimos en el estado del arte, no existe un perfil concreto para este tipo de juegos. Es necesario un nuevo perfil para juegos geoposicionados que complemente el modelo genérico para juegos serios (xAPI-SG) descrito anteriormente.

3.3. Implementación de capacidades de integración de los juegos en procesos educativos

Como se identifica en el estado del arte, la integración efectiva de los juegos en los procesos educativos es uno de los principales retos a resolver para mejorar la aceptación de los juegos serios y su generalización en la educación (Jean Justice & Ritzhaupt, 2015). Este proceso no consiste sólo en la distribución y despliegue de los juegos en la clase, sino que idealmente incluye otros aspectos como, por ejemplo, la gestión de la actividad para saber qué es lo que realmente pasa cuando se usa el juego o la recolección de los datos de interacción que permitan evaluar a los alumnos. Además, si se recogen esos datos se puede mejorar todo el ciclo de vida del juego serio permitiendo contrastar los resultados con sus objetivos educativos, probar su eficacia o detectar posibles problemas.

El problema de saber qué está pasando realmente o cómo evaluar los resultados de un juego, tradicionalmente se ha abordado manualmente mediante observación directa o utilizando formularios externos que rellenan los usuarios. Este método se vuelve inviable

conforme la solución escala en volumen de usuarios, siendo costosa y muy propensa a errores debido a lo complejo y farragoso del proceso. Con el asentamiento del e-Learning y los campus virtuales, para el material educativo muchas de estas labores han sido simplificadas como, por ejemplo, la distribución y la gestión de contenidos o la realización de actividades (e.g., cuestionarios). Nosotros consideramos que sería deseable utilizar un enfoque similar para poder simplificar el despliegue y escalado de los juegos serios. En este caso, es un poco más complejo ya que los juegos habitualmente serán actividades externas al LMS, ya que si se quiere conseguir una buena experiencia de juego hay que abandonar el LMS en pro de otras plataformas web o nativas (sobre todo ahora con las restricciones para juegos con tecnología Adobe Flash⁶⁵). Para este tipo de integración, el uso de estándares educativos permite sistematizar la integración ya que ofrece una interfaz de comunicaciones para actividades externas no-propietario que se puede usar para desplegar juegos serios y recolectar sus resultados. Uno de los estándares presentes en todos los LMS es SCORM (Papazoglou Papazoglakis, 2013), que ha sido utilizado para integrar juegos serios, aunque ofrece capacidades muy limitadas pues no es posible abandonar la plataforma web. Además, las limitaciones de comunicación con y desde el juego proporciona pocos datos sobre el desarrollo de la actividad. Otro de los estándares con mayor soporte para la integración de herramientas externas dentro de un LMS es IMS LTI (Bakhouyi et al., 2017), pero no es eficaz al no poder desplegar cualquier tipo de videojuego (en especial juegos nativos) ya que la especificación está pensada para integrar herramientas web.

Como se ha identificado tanto en el capítulo del estado del arte, como en apartados anteriores, las analíticas de aprendizaje son un mecanismo especialmente relevante para abordar el problema de aplicabilidad y confiabilidad en los juegos serios. No obstante, como también se ha descrito, es habitual encontrarse herramientas, protocolos y modelos de datos propietarios para gestionar las analíticas de aprendizaje. Esta situación conlleva que sea necesario utilizar herramientas específicas o que estemos atados a las herramientas concretas que nos proporcione el desarrollador. No obstante, en el campo de los estándares y especificaciones educativas han surgido diferentes iniciativas que vienen a paliar estos problemas. La iniciativa que consideramos más relevante debido a su flexibilidad, repercusión y adopción que está teniendo en diversos ámbitos como, por

⁶⁵ Comunicado del fin de Adobe Flash: <https://www.adobe.com/es/products/flashplayer/end-of-life.html>

ejemplo, el educativo o de la formación en el trabajo, es la especificación xAPI (eXperience API). Al adoptar esta especificación, como con cualquier otro estándar, optamos a todo un ecosistema de herramientas y arquitecturas regulado por dicha especificación que se pueden usar para dar soluciones generalizables y escalables a las analíticas de juegos serios. Además, esta especificación se puede combinar con otras como, por ejemplo, CMI5 (Miller et al., 2021), que está especialmente diseñada para solventar el problema de lanzamiento (inicialización) desde una plataforma LMS de un recurso complejo como es un videojuego. De este modo, la integración es más sistemática y escalable ya que la experiencia educativa siempre comienza y termina en el LMS, aunque se pase al videojuego temporalmente para que los alumnos jueguen. En conclusión, se puede utilizar la especificación CMI5 para simplificar la distribución y despliegue (con comunicación bidireccional) de actividades complejas desde las plataformas educativas, lo que es especialmente adecuado para juegos serios pues da soporte a plataformas nativas y móviles además de entornos web.

La especificación xAPI además nos ofrece otros beneficios desde el punto de vista técnico, permitiéndonos desacoplar la herramienta desde la que se inicia la actividad (el LMS) y el almacén para las analíticas generadas durante la sesión de juego. En la especificación xAPI este almacén, denominado LRS (*Learning Record Store*), podría estar integrado dentro del propio LMS o puede ser un LRS independiente (por ejemplo, un LRS corporativo donde se almacenan todas las analíticas generadas en distintos sistemas). Una vez recopiladas las analíticas en el LRS pueden ser procesadas de manera específica o bien pueden ser procesadas dentro de un modelo de explotación más general como, por ejemplo, el propuesto en la arquitectura TLA (*Total Learning Architecture*) propuesto por ADL para el Ministerio de Defensa de EE. UU. (Alonso-Fernandez et al., 2017; Papazoglou Papazoglakis, 2013). De este modo es posible integrar los juegos serios mediante ecosistemas para analíticas de aprendizaje que se construyan utilizando componentes software genéricos y ya disponibles, reduciendo así su complejidad y coste, y aumentando la fiabilidad. El soporte de estándares permite interoperar e integrar las soluciones desarrolladas en este trabajo de tesis con otras herramientas de terceros que ya son utilizadas por la comunidad educativa y aporta una mayor longevidad al software creado.

No obstante, esta interoperabilidad basada en estándares exige un esfuerzo extra de desarrollo. Se parte de la solución implementada en las primeras fases de la tesis para dar

soporte para la generación y recolección de las trazas xAPI y se extiende para lograr la interoperabilidad. El objetivo es aportar soluciones para la aplicación inmediata de los juegos serios conectados con sistemas de analíticas tanto de propósito específico como los desarrollados en el contexto de los proyectos del H2020 RAGE y BECONING o la herramienta de soporte a la validación de juegos serios SIMVA (I. J. Perez-Colado et al., 2019), como de propósito general para lo que incluye el soporte a LRS genéricos. Este proceso de integración de las analíticas, aunque orientado a desarrolladores debe de ser sencillo, mantenible y robusto y, por supuesto, proporcionar funcionalidades y una buena experiencia de usuario a sus usuarios (e.g., educadores, desarrolladores).

Para mejorar la aplicabilidad de las analíticas para juegos serios, además de implementar la solución dentro del entorno de autoría, también se desarrollará como un componente independiente, con el objetivo de que otros desarrolladores de juegos serios puedan realizarlo como un componente en la plataforma Unity. Esto requiere el estudio de las soluciones existentes (relacionadas tanto con analíticas de aprendizaje en general, las analíticas de juegos en particular) para ofrecer una solución competitiva y eficaz. Este estudio se focaliza en la pieza central del cliente de analíticas, el *tracker* xAPI y en las principales herramientas con las que se conectará, los LRS. Puesto que el enfoque del *tracker* propuesto es el de ofrecer una interfaz de alto nivel, en el que no es necesario tener un amplio conocimiento de los estándares soportados, este *tracker* podría ser utilizado por todo tipo de desarrolladores. Esto posibilita un caso interesante de uso con aquellos desarrolladores que quieran mejorar las funcionalidades de sus juegos y dar el salto a xAPI y a su ecosistema (LRS, CMI5, TLA, etc.) ya que reduce su curva de aprendizaje, los requisitos técnicos necesarios y mitiga en gran medida posibles errores por un conocimiento insuficiente de los estándares. En este caso el *tracker* actúa como un middleware de conexión de datos que apantalla gran parte de los complejos detalles de los estándares.

3.4. Desarrollo de una extensión para géneros híbridos y minijuegos

Otra de las características encontradas en la revisión del estado del arte para mejorar el proceso de diseño y la efectividad de los juegos consiste en la hibridación o mezcla de géneros de juego y la inclusión de minijuegos de propósito específico (Aslan & Balci, 2015). Aplicando esta idea, es posible suplir las carencias del modelo narrativo y añadir

mecánicas nuevas a los juegos que mejoren el proceso de aprendizaje con esos juegos. De hecho, esta hibridación y los minijuegos son especialmente relevantes para los juegos de aventura y los geoposicionados, caracterizados por incluir diferentes puzles (como en el caso de los juegos de aventura de puzles) y retos personalizados (como en las diferentes pruebas de una gymkhana/búsqueda de tesoros geoposicionada).

Si bien conseguir esto es técnicamente posible con uAdventure, pues es una herramienta con una arquitectura extensible para los desarrolladores, el objetivo de esta extensión es simplificar aún más el proceso y permitir que los desarrolladores de juegos puedan complementar los juegos de uAdventure sin tener que conocer en profundidad los detalles de bajo nivel de la herramienta. Para poder aprovechar todas las capacidades de Unity se ha decidido que dichos contenidos híbridos y minijuegos se almacenen como escenas de Unity ya que es un formato amigable para los desarrolladores experimentados en este entorno. Para simplificar su integración, ha sido necesario definir un sistema capaz de intercambiar estas escenas de Unity durante la ejecución del juego de forma que sea transparente al jugador. Por otro lado, para que los desarrolladores puedan además beneficiarse del uso de uAdventure, todas las interfaces de control del motor de uAdventure están disponibles durante estas escenas (e.g., los gestores del estado del juego, las mecánicas narrativas). Por último, y para que estos contenidos híbridos puedan integrarse completamente en el proceso educativo, deben poder generar trazas xAPI como el resto de las escenas de uAdventure. Para ello, se aprovecha la API de alto nivel del *tracker* de analíticas de uAdventure que, como hemos descrito antes, permite a los desarrolladores utilizar xAPI sin conocer las cuestiones técnicas del formato.

3.5. Validación de la aplicabilidad de uAdventure y de los juegos desarrollados

La validación de las diferentes soluciones aportadas en este trabajo de tesis se realiza a través de las pruebas piloto y la experimentación. Esta validación se centra en dos objetivos independientes: por un lado, demostrar que tanto los juegos que se crean con uAdventure como la propia herramienta funciona correctamente y, por otro lado, validar que la metáfora de creación de juegos es flexible y adecuada, de modo que los usuarios sean capaces de crear sus propios juegos con las mecánicas proporcionadas o bien ampliándolas con otras propias si así lo requieren.

Las pruebas piloto con juegos permiten demostrar la utilidad de los juegos creados con uAdventure con usuarios reales. Estos pilotos ponen a prueba la herramienta y sus capacidades para funcionar en diferentes situaciones, con múltiples dispositivos y con un volumen significativo de usuarios, validando así sus capacidades. Estos pilotos permiten validar los distintos objetivos de la tesis y en algunos casos un mismo piloto permite abordar simultáneamente diversas áreas. Por ejemplo, el juego serio First Aid, un videojuego preexistente y previamente validado en el grupo e-UCM (Marchiori, Ferrer, et al., 2012), se usa para probar tanto que uAdventure cumple con las capacidades de jugabilidad de e-Adventure, como para verificar que las analíticas de aprendizaje automáticamente generadas son funcionales. El diseño e implementación desde cero de un juego serio geoposicionado, llamado El Señor De Los Arbustos, ha permitido realizar pruebas con alumnos de distintos niveles (universidad, instituto) para mejorar la jugabilidad y la aplicabilidad del modelo de juegos geoposicionado en el entorno educativo. Este juego, además, ha sido el primero en ser probado con SIMVA, como prueba piloto de la integración de las soluciones en otros entornos. Cabe destacar que el resto de las pruebas con juegos han surgido como resultado del trabajo de nuestros usuarios y sirven para completar la validación del ciclo completo de la herramienta.

Las pruebas de la autoría de la herramienta por usuarios reales se realizarán en dos etapas. En primer lugar, una vez finalizadas las funcionalidades básicas de la herramienta las primeras pruebas con usuarios tienen como objetivo demostrar que las herramientas de autoría son asequibles para usuarios sin experiencia en la herramienta y de forma autónoma, obteniendo así datos que permitirán mejorar la calidad de la herramienta. Para ello, se entregará a los usuarios el software, una guía de inicio rápido y el manual completo con ejemplos y tareas a completar. Esto es necesario, pues, además de probar que en la práctica los usuarios pueden usar uAdventure por sí mismos, permite que, en las próximas pruebas, en formato de *workshop*, se maneje un volumen mayor de usuarios gracias a que la herramienta será más estable y que el aspecto de formación autónoma no sea tan importante (al haberse probado ya), permitiendo así acelerar la formación de los participantes. En segundo lugar, las pruebas en *workshop* tienen el objetivo de conseguir que los usuarios desarrollen juegos serios completos de forma autónoma, permitiendo así una mayor exploración y valoración de todas las posibilidades que permite la herramienta y la creación de un conjunto de juegos creados por usuarios con los que experimentar. Para los cursos se utilizará el formato de *workshop*, donde los estudiantes recibirán una

iniciación guiada y deberán utilizar los conocimientos adquiridos para crear juegos. Estos conocimientos se irán ampliando conforme se sucedan los diferentes objetivos de esta tesis en forma de módulos con temáticas de aventura, geoposicionamiento, analíticas de aprendizaje y minijuegos, y dando lugar al conjunto de juegos que utilizaremos en los experimentos finales.

En el siguiente apartado de esta tesis expondremos los diferentes resultados y contribuciones que hemos logrado con nuestros objetivos, así como entraremos en detalle en las diferentes pruebas que se han realizado.

Capítulo 4. Discusión integradora y contribuciones

En este capítulo se expone el trabajo realizado para abordar tanto el objetivo general como los cinco objetivos concretos de esta tesis propuestos y descritos en detalle en el Capítulo 3. Dado que esta tesis está compuesta por un compendio de publicaciones, este capítulo proporcionará una breve discusión integradora del trabajo realizado, pudiéndose consultar los detalles completos del trabajo en las publicaciones que acompañan este trabajo de tesis doctoral. Este capítulo se divide en cinco apartados, cada uno de los cuales corresponde a uno de los objetivos específicos propuestos para este trabajo de tesis.

4.1. Desarrollo de la herramienta de autoría

Como el objetivo principal de esta tesis es simplificar el desarrollo de juegos serios geoposicionados aplicables en el contexto educativo, la herramienta de autoría de juegos serios es la pieza fundamental sobre la que articular y desarrollar soluciones para abordar el resto de los objetivos propuestos en este trabajo de tesis. Para diseñar este entorno de autoría “base” para juegos serios narrativos, que hemos denominado uAdventure, hemos tomado como referencia la herramienta eAdventure previamente creada en el grupo de investigación e-UCM. Esta herramienta simplifica la creación de juegos y aventuras gráficas narrativas a usuarios no técnicos (ya que no requiere profundos conocimientos de programación), y ha tenido una gran acogida llegando a tener más de 100.000 descargas desde su creación⁶⁶ y siendo utilizada tanto en proyectos de investigación como por parte de profesores para actividades docentes (Cowan & Kapralos, 2017; Marchiori, Torrente, et al., 2012a; Torrente et al., 2010).

En uAdventure, se han tratado de paliar algunas de las limitaciones existentes en eAdventure, como la obsolescencia tecnológica, además de extender el género de juegos disponibles incluyendo los geoposicionados (como veremos en la sección 4.2), y heredando sus puntos fuertes: la amplia experiencia de uso dentro del colectivo docente y la exitosa metáfora de autoría que ha permitido que haya sido utilizada y aplicada en diversos entornos. Más concretamente, la arquitectura de uAdventure se ha diseñado con el objetivo de simplificar el desarrollo y mantenimiento del propio entorno, y flexibilizar la incorporación de nuevas funcionalidades educativas tal y como hemos analizado en el

⁶⁶ La última versión de e-Adventure y las estadísticas pueden encontrarse en: <https://sourceforge.net/projects/e-adventure/>

capítulo anterior. Al desarrollar uAdventure sobre el motor Unity, podemos simplificar el desarrollo y mantenimiento al delegar al máximo los aspectos genéricos relacionados con el desarrollo de videojuegos en el motor. Estos aspectos incluyen la construcción visual de la escena, la creación de mecánicas, la gestión de recursos gráficos y la creación de editores, que Unity no sólo simplifica, sino que lo hace de forma eficiente y dando soporte multiplataforma. Esto nos permite centrar el esfuerzo de desarrollo de uAdventure en los aspectos concretos relacionados con la simplificación del proceso de autoría por parte de no expertos y la integración de características educativas. Construir uAdventure sobre una herramienta que se utiliza a nivel comercial y profesional como Unity ha permitido simplificar su mantenimiento sostenido en el tiempo, y abre la puerta a explorar otras características de Unity que permiten probar nuevos géneros y mejorar la compatibilidad de los juegos creados con uAdventure en diferentes plataformas y dispositivos. Por otro lado, se ha tenido un especial cuidado en mantener la compatibilidad con los juegos previamente desarrollados con eAdventure, permitiendo su conversión automática con el objetivo de, por un lado, prolongar la vida útil de los juegos existentes y, por otro lado, facilitar la transición hacia la nueva herramienta.

El trabajo desarrollado como parte de la tesis se ha inspirado en los prototipos desarrollados por Piotr Marzał (Marszał, 2016) e Iván José Pérez Colado (I. J. Pérez-Colado, 2016) en 2016 para crear una herramienta que pudiera convertir los juegos creados con eAdventure en juegos que pudieran ser interpretados por un motor creado en la plataforma Unity. El objetivo principal, era paliar la obsolescencia de la tecnología Java utilizada para desarrollar el motor de juegos integrado en eAdventure (Cotroneo et al., 2007), y analizar la viabilidad de una migración completa de la plataforma eAdventure a Unity. Como parte de este trabajo se desarrolló un prototipo donde era posible crear nuevos juegos e importar juegos antiguos de e-Adventure para aprovechar Unity para mejorar su calidad y poder distribuirlos en nuevas plataformas, dándoles así una nueva vida. No obstante, este prototipo era muy primitivo, pues además de contener errores no contaba con soporte para muchas de las funcionalidades de e-Adventure como los juegos en tercera persona (donde el jugador está representado por un avatar). Partiendo de esta prueba de concepto, se han llevado a cabo diversas iteraciones hasta dar lugar a la versión actual de uAdventure en la que no solo se ha logrado la paridad con las funcionalidades ofrecidas por eAdventure, sino que se han sentado las bases para establecer una herramienta modular sobre la que desarrollar e integrar nuevas funcionalidades

educativas. Estas nuevas características han permitido desarrollar los otros objetivos de esta tesis doctoral.

La primera iteración del desarrollo de uAdventure consistió en modificar el prototipo de motor de juegos eAdventure sobre Unity, con el objetivo de adaptar y flexibilizar la arquitectura de la herramienta haciéndola más modular y extensible de manera que se pudieran añadir nuevas funcionalidades. Con esta nueva arquitectura se desarrolló una extensión experimental de la herramienta para crear juegos geoposicionados básicos.

Las siguientes iteraciones del desarrollo se centraron en reconstruir la herramienta de autoría también sobre la plataforma Unity, e implementar las características y funcionalidades de eAdventure que no habían sido abordadas en el prototipo original desarrollado en 2016. El trabajo desarrollado en esta etapa se analiza en la publicación Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). **Simplifying the creation of adventure serious games with educational-oriented features.** *Educational Technology and Society*, 22(3), 32–46. aunque esta publicación incluye también aspectos que se tratan en otras secciones que expondremos más adelante.

Para comprender mejor las contribuciones de esta sección expondremos primero de forma sencilla la arquitectura y el modelo de la plataforma. La arquitectura de uAdventure consta de tres partes básicas: el *núcleo*, el *editor* y el *intérprete*. El *núcleo* es el módulo que contiene las funcionalidades básicas para representar tanto los elementos como la lógica asociada a ellos, incluyendo las acciones de interacción del jugador con los elementos de juego y los posibles estados que estos elementos pueden tener. También es el encargado de interpretar los ficheros que usa el entorno para almacenar esta información y actualizarlos en el caso de que sea necesario para que sean compatibles con la versión del entorno que se esté utilizando. Este módulo sirve como base para los otros dos módulos. El *editor* es el módulo que contiene las diferentes interfaces de usuario que se utilizarán para construir la representación del juego y también se encarga de almacenar esta información de forma persistente para su posterior ejecución. Por último, el *intérprete* es el módulo encargado de construir la representación visual del juego y proporcionar los mecanismos de interacción al jugador con los que explorar e interactuar con los diferentes elementos y así progresar en el juego.

El modelo de juego heredado de e-Adventure es un modelo narrativo inspirado en los juegos de aventura “*point and click*”. En él se incluyen *escenarios* (donde ocurre la acción), *personajes* y *objetos* (con los que interactuar) y elementos decorativos llamados *atrezos*; además de otros elementos que contribuyen a crear la narrativa como *conversaciones*, *temporizadores* o *libros*. Al ser un modelo de alto nivel es sencillo de comprender (y de manipular) incluso para personas sin experiencia en desarrollo de juegos pero que entiendan los componentes básicos del género narrativo. Estos escenarios pueden explorarse tanto en *primera* como en *tercera persona*, y para añadir la jugabilidad y los puzzles o desafíos del juego, los diferentes elementos del modelo cuentan con un sistema para definir la interacción básica y con un soporte de programación de alto nivel basado principalmente en *variables*, *condiciones* y *efectos*. El sistema de interacción se basa en diferentes verbos (como coger, hablar o usar) que representarán las diferentes *acciones* disponibles para el jugador. Al realizar las acciones se podrán realizar cambios en el estado del juego o retroalimentar de diferentes formas al jugador utilizando el sistema de *efectos*. Algunos de los efectos que se incluyen son cambiar de escenas, iniciar conversaciones o cambiar las variables del estado del juego. En el estado del juego, además del contexto del jugador y los elementos, contaremos con un sistema de *variables*. Estas variables son accesibles a través del sistema de *condiciones*, que está presente en casi todas las partes del modelo, permitiendo así que éste cambie conforme el jugador interactúa. Otros elementos heredados del modelo de e-Adventure incluyen elementos de “calidad de vida” que simplifican la autoría como *áreas interactivas* (que pueden usarse para añadir interactividad a elementos del fondo), *macros* de efectos y *estados globales*.

Tras los primeros prototipos y con las correspondientes pruebas con usuarios podemos afirmar que las contribuciones de esta tesis han convertido a uAdventure en una herramienta madura que cumple con los objetivos que habíamos planteado en el Capítulo 3. Esta maduración abarca todas las partes del proyecto, incluyendo el rediseño de sus editores, así como el enriquecimiento de la experiencia de juego (*gameplay*) y múltiples mejoras al modelo y sus funcionalidades. Los puntos más importantes del trabajo han consistido en: la recuperación y adaptación de antiguas funcionalidades, la adaptación de los editores visuales, la simplificación de los procesos de configuración y exportación (relativos a Unity), y la mejora de la extensibilidad de la arquitectura para incluir nuevos módulos y funcionalidades.

4.1.1. Recuperación y adaptación de antiguas funcionalidades

Puesto que uAdventure aspira a ser el sucesor natural de e-Adventure, es fundamental dar soporte a sus antiguos usuarios y juegos como parte del desarrollo de la herramienta. Esto permitirá a los usuarios de eAdventure dar el salto a la nueva versión, actualizar sus juegos extendiendo su vida útil, beneficiándose de las nuevas funcionalidades y mejoras del nuevo motor. Además, esto es compatible con que los nuevos usuarios añadan nuevas posibilidades a sus juegos.

Una de las funcionalidades más importantes de e-Adventure era el modo en tercera persona. En este modo, un avatar representa al jugador dentro del juego y puede explorar las escenas de manera condicionada con un sistema de búsqueda de rutas. En su nueva implementación, se ha aprovechado para mejorar el sistema de rutas permitiendo varias rutas alternativas en un mismo escenario y un mejor uso de las áreas de influencia de los objetos (lo que mejora mucho la jugabilidad).

Otras funcionalidades importantes que se han recuperado han sido los libros, efectos como el utilizado para mover personajes o el efecto de esperar, las funcionalidades generales de aventura, incluyendo las ventanas de configuración, la vista general de escenas del juego, las capacidades de guardar, cargar y reiniciar la partida y el sistema de gestión de inventarios. Por ejemplo, con este sistema de inventario, además, se recuperaron las acciones que combinan o entregan objetos, así como las interacciones de “*drag and drop*” convenientes para dispositivos móviles.

Aprovechando las nuevas capacidades tecnológicas disponibles a través de Unity, se han añadido mejoras visuales y de interacción especialmente diseñadas para dispositivos móviles. Entre estas nuevas características se incluye poder acceder al inventario más fácilmente, añadir brillo a los objetos para que llamen la atención del jugador, mover la cámara de la escena arrastrando los dedos o la posibilidad de habilitar *zoom* en las escenas con soporte táctil.

Además de estos cambios, se han mejorado algunas de las metáforas y mecánicas básicas ya existentes. Estas mejoras van desde optimizaciones para hacer un uso más eficiente de los recursos, por ejemplo, durante el proceso de importación de proyectos existentes, hasta nuevas funcionalidades en sistemas como el de conversaciones, permitiendo organizar y distribuir las conversaciones de diversas maneras además de poder incluir imágenes como parte de estas.

4.1.2. Nuevos editores y mejoras visuales

En uAdventure todos los editores han sido reimplementados desde cero para poder ofrecer una experiencia de usuario más moderna y cercana a Unity, pero a la vez sin tener que lidiar con toda la complejidad de un motor profesional de desarrollo de juegos. Queremos hacer especial énfasis en dos de los editores que han sido modificados: el editor de escenas y el editor de conversaciones y efectos.

El nuevo editor de escenas de uAdventure (**Figura 16**) permite un control de la escena más simple, amplio y eficiente sin romper completamente con la experiencia de e-Adventure. Este nuevo editor mejora el manejo de los elementos, utilizando controles más libres e intuitivos, y aumenta la productividad al permitir manipular todas las variables de la escena y sus elementos desde un sólo lugar. Asimismo, este nuevo diseño está inspirado en el editor de Unity, con el objetivo de ofrecer un punto intermedio entre la experiencia de los usuarios que conocen e-Adventure y la de aquellos que conocen Unity. La característica más destacable del mismo es su inspector, en donde se desglosa la información simplificada de cada elemento. Por otro lado, el editor reemplaza el comportamiento por defecto del botón de ejecutar de Unity, haciendo que se ejecuten las escenas de uAdventure, facilitando así la prueba y depuración de los juegos.

Los nuevos editores de conversaciones y efectos pertenecen a la categoría de editores de nodos. En los editores de nodos, los nodos contienen toda la información relevante para

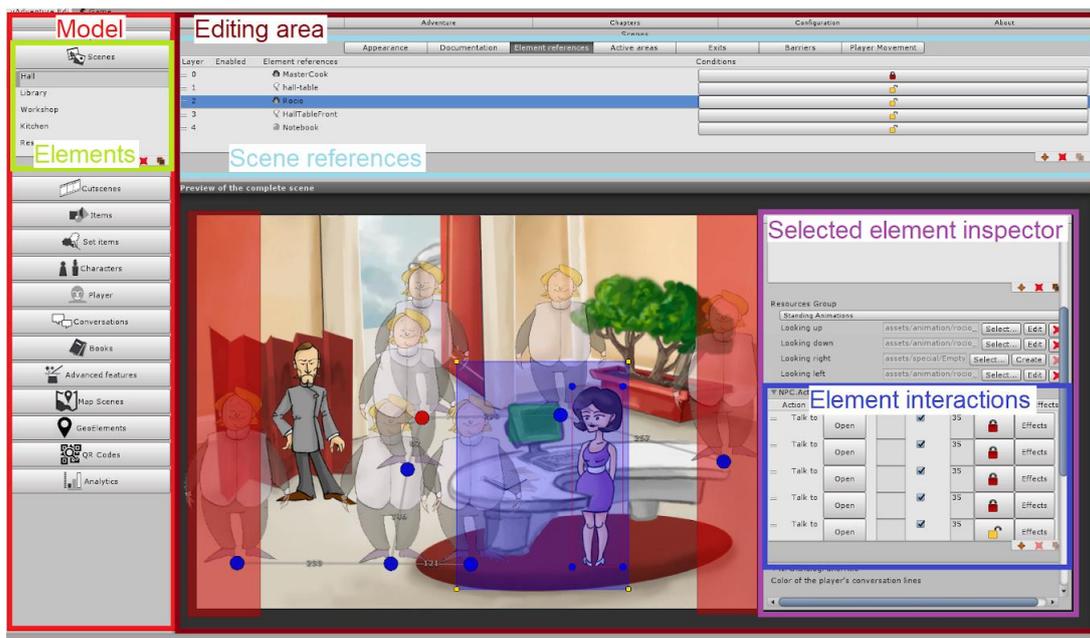


Figura 16. Nuevo editor de escenas de uAdventure con inspector.

la ejecución en ese momento y se conectan a través de líneas direccionales a otros nodos para indicar el orden de ejecución. Este diseño además favorece la creación de configuraciones no-lineales al permitir la libre distribución y exploración de los nodos. Estos editores suponen un acercamiento similar al que encontramos en el mercado como los Shader Graph⁶⁷ de Unity o los Blueprints⁶⁸ de Unreal Engine.

4.1.3. Simplificación de procesos de configuración y exportación

Para maximizar la aplicabilidad de los juegos serios es importante flexibilizar las plataformas en las que se puede distribuir el juego ya que tanto los equipos como los contextos de juego pueden ser muy diversos. Esta necesidad, es uno de los principales motivos por los cuales se ha elegido Unity como base para construir uAdventure, pues al ser una herramienta comercial de amplio uso garantiza soporte en las plataformas actuales (y sus continuas actualizaciones) así como el soporte de otras nuevas plataformas que puedan aparecer a lo largo del tiempo. Dentro de la variedad de plataformas soportadas por Unity, el desarrollo de uAdventure se ha centrado en mantener y verificar la compatibilidad de los juegos creados en las plataformas de PC, móviles y entornos web. Consideramos que son los entornos utilizados más habitualmente en el contexto educativo y, por tanto, de especial relevancia para la consecución de los objetivos de esta tesis.

Si bien Unity permite la construcción de versiones en todas las plataformas mencionadas, el proceso de exportación de los juegos puede ser confuso y propenso a errores debido a la gran cantidad de parámetros de configuración disponibles. Por ello, uAdventure proporciona un exportador simplificado que automatiza parte del proceso de construcción de la aplicación final ofreciendo por defecto los parámetros más relevantes con los valores más habituales. Este exportador es capaz de realizar las configuraciones y ajustes en Unity para poder exportar los elementos en plataformas de sobremesa (Windows, Linux y Mac), móviles (Android e iOS) y web (WebGL). El caso del entorno web es especialmente interesante pues por su ámbito ha requerido más automatizaciones como, por ejemplo, transformar los recursos multimedia a formatos compatibles con la web utilizando la librería de código abierto FFMPEG⁶⁹.

⁶⁷ <https://docs.unity3d.com/Manual/shader-graph.html>

⁶⁸ <https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/Blueprints/>

⁶⁹ <https://www.ffmpeg.org/>

Además, en el momento de la importación de recursos (*assets*) y proyectos, uAdventure configura tanto la calidad como los parámetros de configuración internos de Unity para garantizar el mejor rendimiento y minimizar los errores. Este proceso es especialmente relevante para actualizar antiguos juegos, pues permite optimizar los parámetros de configuración de los grandes volúmenes de *assets* que contienen de forma automática reduciendo drásticamente el tiempo de actualización.

4.1.4. Mejoras de extensibilidad y extensión de analíticas de aprendizaje

Como avanzamos al principio de esta sección, uno de los requisitos del rediseño de uAdventure ha sido que sea fácilmente extensible para soportar nuevas funcionalidades y características de juego. Esto no es sólo útil para la consecución de los objetivos planteados en este trabajo de tesis, sino que también hará mucho más sencillo para desarrolladores crear nuevas extensiones (en inglés, *plugins*) que adapten o permitan evolucionar la herramienta y facilitará la sostenibilidad y mantenibilidad de esta en el futuro.

Esta extensibilidad de uAdventure se ha centrado en simplificar la ampliación de los elementos de juego disponibles y editables a través de la herramienta. Esto incluye nuevos tipos de escenarios, efectos, elementos en el modelo, editores, reacciones a eventos (o *hooks*) y extensiones del *gameplay*. Para facilitar el proceso de integración, la arquitectura hace uso de los sistemas para la extensión del editor de Unity y las características de reflexión presentes en C#⁷⁰ (lenguaje de scripting integrado con Unity), lo que permite detectar y cargar las extensiones de forma automática, sin necesidad de que el desarrollador tenga que conocer o manipular el código fuente de uAdventure para crear extensiones. Otras contribuciones orientadas a simplificar la extensión de uAdventure consisten en la creación de clases genéricas para editores (por ejemplo, pestañas o listas), así como otras utilidades como, por ejemplo, las clases orientadas a simplificar la gestión de datos de guardado provenientes de extensiones.

La primera extensión desarrollada utilizando estas nuevas capacidades de uAdventure, ha sido la extensión para dar soporte a las analíticas de aprendizaje. Como se ha descrito en el estado del arte, con las analíticas de aprendizaje es posible conocer aspectos tales como el estado del juego, las acciones del jugador u otros eventos relevantes, y todos estos datos

⁷⁰ Introducción a scripting en Unity: <https://docs.unity3d.com/Manual/CreatingAndUsingScripts.html>

pueden ser utilizados con múltiples propósitos. Por ejemplo, esta información puede usarse para identificar errores durante el desarrollo del juego, monitorizar la interacción del jugador con el objetivo de analizar el proceso de aprendizaje y hasta ser utilizada como mecanismo de evaluación de los estudiantes. No obstante, para que las analíticas sean efectivas es necesario que se planteen desde las etapas de concepción, diseño y desarrollo del juego para que sean depuradas y validadas conforme se desarrolle el juego serio. Por tanto, además de incluir una serie de analíticas por defecto, que requieren una mínima configuración por parte de los desarrolladores, se ha implementado en uAdventure un editor específico para poder identificar el progreso del jugador en el juego. En este editor se permite manipular elementos *completables* que identifican cuando una tarea se *incia*, como se *progres*a en ella y cuando se *completa*. Para lograrlo se vinculan a estos estados los *hitos* correspondientes dentro del juego, tales como las interacciones con elementos del juego o el estado de sus variables. Estos elementos completables se basan en el concepto de *completable* del perfil de xAPI para juegos serios (xAPI-SG) y permiten configurar la generación de las trazas a modo de plantilla, de forma que, incluso un desarrollador sin conocimiento profundo de analíticas puede aprovecharse de sus ventajas de manera sencilla.

4.2. Desarrollo de extensión para juegos geoposicionados

El desarrollo de una extensión de juegos geoposicionados para uAdventure tiene como objetivo permitir la autoría de juegos inmersivos de género geoposicionado. Esta extensión tiene dos objetivos independientes: la extensión del modelo de juego para incluir geoposicionamiento y sus editores correspondientes (e.g., mapas), y la extensión de las analíticas de aprendizaje para tener en cuenta estos nuevos aspectos geoposicionados. En los siguientes subapartados se analizará esta extensión y su aplicabilidad en el desarrollo de juegos serios geoposicionados.

4.2.1. Modelo, jugabilidad y autoría

Esta nueva extensión de uAdventure aprovecha los mecanismos de extensibilidad de uAdventure para añadir tanto nuevos elementos de juego y mecánicas de interacción, así como nuevos editores en el entorno de autoría. Esta extensión propone un formato de juego geoposicionado orientado a las aventuras, dando lugar al género de aventura geoposicionada. Este trabajo se ha presentado en el artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2017).

Simplifying location-based serious game authoring. In J. M. Dodero, M. S. Ibarra Sáiz, & I. Ruiz Rube (Eds.), *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM)* (pp. 1–9).

Las funcionalidades que añade la extensión a uAdventure para juegos geoposicionados pretenden explotar el potencial inmersivo que permiten los sensores presentes en los teléfonos inteligentes, abarcando los casos más comunes identificados en el estado del arte como, por ejemplo, el uso del geoposicionamiento, la orientación o la cámara. Para ello, en la extensión se expande el modelo de uAdventure y se añaden nuevas mecánicas que aprovechen este género de juego. Dados los diferentes estilos de juegos geoposicionados (Kjeldskov & Paay, 2007), el modelo de juego implementado uAdventure es más flexible que el de los juegos geoposicionados estándar (Karoui et al., 2017). En uAdventure, el escenario principal es el *mapa aumentado*, donde se introducen elementos reales y elementos virtuales llamados *elementos geoposicionados* (**Figura 17**).

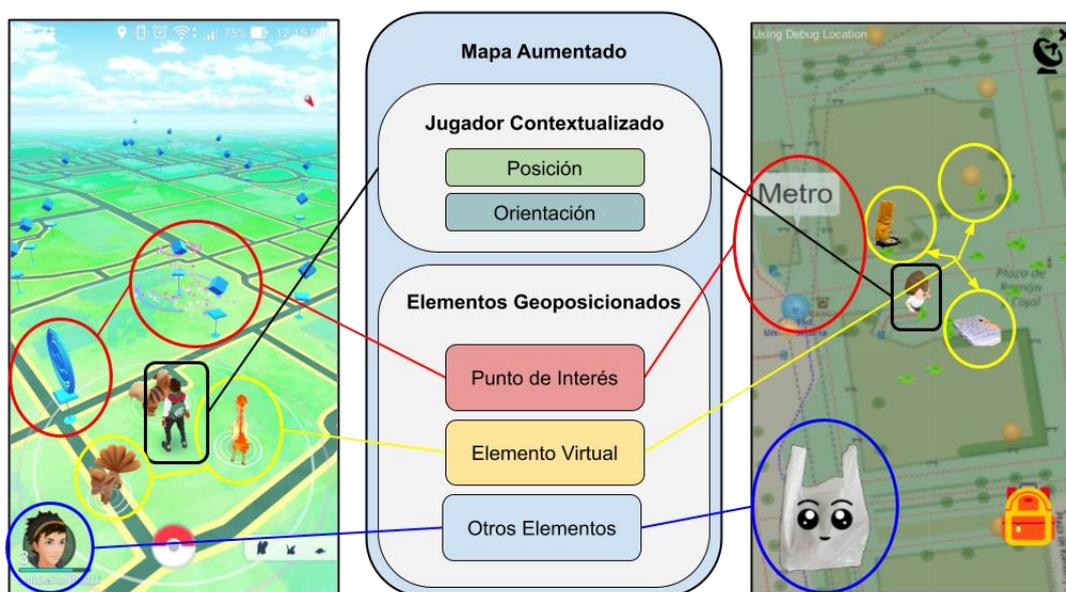


Figura 17. Elementos del modelo de juegos geoposicionados de uAdventure y ejemplos en los juegos geoposicionados Pokémon GO (izquierda) y El Señor De Los Arbustos (derecha; hecho en uAdventure).

Por un lado, el mapa aumentado ha sido construido utilizando como base el proyecto MapzenGO⁷¹, utilizando los mapas de OpenStreetMap⁷² y Carto⁷³, proporcionando un

⁷¹ <https://github.com/brnkhy/MapzenGo>

⁷² <https://www.openstreetmap.org/>

⁷³ <https://carto.com/>

total de 6 estilos diferentes de representación visual. En este mapa, el *avatar* del jugador aparece en el medio y el jugador puede controlar la distancia de la perspectiva para ayudarlo a orientarse. Por otro lado, para introducir los diferentes elementos geoposicionados, los elementos reales se pueden representar utilizando puntos, regiones y caminos, que se basan en el estándar GML (Open Geospatial Consortium, 2010) y se traducen en este al guardar el modelo, mientras que, los elementos virtuales se añaden utilizando el propio modelo de uAdventure (como personajes y objetos) que ahora pueden incluirse en el mapa. Estos elementos virtuales provenientes del modelo narrativo de uAdventure proporcionan la capa de *narrativa* que se utilizará en los juegos para introducir contenidos y dar sentido a las tareas que tiene que realizar el jugador.

Durante la sesión de juego utilizando un mapa aumentado, la interacción se realiza a través del contexto del jugador en el mundo real, teniendo en cuenta su *posición* y *orientación*, desencadenando lo que hemos llamado *acciones geoposicionadas*. Las acciones geoposicionadas utilizan la mecánica de “*área de influencia*” que habitualmente encontramos en los juegos y herramientas geoposicionados (Benford et al., 2005; Mettis & Valjataga, 2020), que se basa en contrastar la posición del jugador con la posición del elemento a interactuar y una distancia máxima (en metros) respecto a esta área. En función de esta distancia, el jugador puede *entrar* o *salir* de la influencia y *mirar* al objeto o hacia una dirección concreta estando en el área de influencia del objeto. Un ejemplo de interacción sería entrar en un edificio (por ejemplo, el edificio de la Facultad de Informática), cuya influencia permitiría abarcar también sus alrededores, disminuyendo así los posibles fallos en el juego debidos a la falta de precisión en la localización. Aparte del desencadenante, estas nuevas acciones son similares al resto de interacciones de uAdventure en cuanto a configuración y uso, simplificando su curva de aprendizaje y permitiendo integrar estas acciones con el resto de las funcionalidades de uAdventure. Siguiendo con el ejemplo anterior, al entrar en el área de influencia de la Facultad de Informática se podría ejecutar una conversación que introdujera una misión a realizar en el lugar a través de una conversación estándar de uAdventure.

Para determinar la posición del jugador se utiliza principalmente la señal GPS del móvil (aunque en algunos dispositivos además puede utilizarse la red de satélites de

GLONASS⁷⁴), pero también es posible localizar al jugador de otros modos en casos donde la cobertura sea mala o necesitemos una mayor precisión. En ese caso, uAdventure proporciona una extensión para códigos QR que podrán leerse desde el propio juego a través de la cámara del móvil, funcionando tanto en exteriores como en interiores. En el caso de la orientación para las acciones en las que hay que mirar hacia algún lugar, esta viene determinada por la brújula del móvil.

Además, el modelo geoposicionado incorpora otras mecánicas, para personalizar la visualización de los elementos del mapa, por ejemplo, ocultándolos hasta que el jugador esté cerca de su influencia (de forma similar a como hace Pokémon GO), haciendo que muestren una pista en el mapa mientras están ocultos o incluso posicionándolos la pantalla de forma estática (por ejemplo, para la creación de interfaces o representar a otros compañeros). Un elemento especialmente relevante es el navegador que ayuda a los jugadores a seguir las rutas esperadas por el juego y proporcionar así un entorno más controlado para profesores y estudiantes (por ejemplo, evitando zonas peligrosas). Otra mecánica para mejorar la inmersión son los cambios de escenas durante la estancia en el área de influencia del elemento, haciendo que estas escenas sólo puedan explorarse durante la permanencia del jugador en el lugar real.

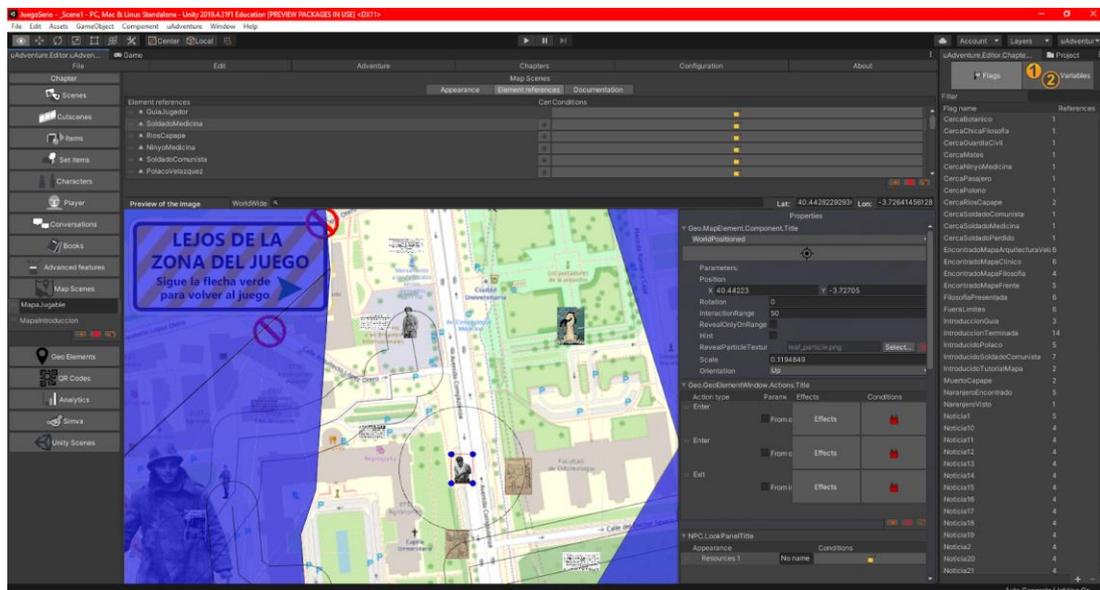


Figura 18. Editor para las escenas basadas en mapas aumentados de los juegos geoposicionados.

⁷⁴ https://www.glonass-iac.ru/spa/about_glonass/

Los editores diseñados para esta extensión se han ido mejorando a lo largo de las diferentes iteraciones de uAdventure realizadas durante la tesis en base a las pruebas con usuarios. En la **Figura 18** puede observarse la versión actual del editor de escenas en mapas. En estos editores, además de poder manipular la colocación y representación de los elementos, también es posible determinar la zona de juego de forma que el mapa se descargará y almacenará en el juego para permitir que pueda jugarse sin necesidad de conexión a internet lo que aumenta las posibilidades de aplicación de los juegos resultantes.

4.2.2. Aplicabilidad de juegos geoposicionados

Uno de los objetivos principales de esta tesis es la aplicabilidad real de los juegos serios, especialmente en términos de geoposicionamiento. El enfoque para su aplicabilidad se basa en la extracción de datos de valor utilizando analíticas de aprendizaje basadas en xAPI para evaluar y contrastar su uso en el proceso educativo. Como vimos en el apartado 4.1 sobre la herramienta de autoría uAdventure incluye soporte de analíticas utilizando xAPI, mediante el perfil específico para juegos serios xAPI-SG presentado por Serrano-Laguna et al., (2016) . El uso de este perfil hace posible la inclusión de analíticas por defecto en las que, sin necesidad de manipular la herramienta, todos los cambios de escena, interacciones con personajes y objetos, y elecciones que se produzcan en nuestros juegos geoposicionados generarán trazas que podrán ser analizadas a posteriori. Sin embargo, este perfil no estaba diseñado para juegos geoposicionados y, por tanto, no es capaz de representar de manera precisa las particularidades de los diferentes elementos geoposicionados, interacciones y mecánicas concretas de este tipo de juegos.

Como avanzábamos en el estado del arte, en términos de geoposicionamiento xAPI no tiene un perfil concreto para este tipo de actividades, aunque existen algunas iniciativas para incluir geolocalización en las trazas, como la propuesta por Julian Davis (Davis, 2020). Por ello, durante las fases iniciales de esta tesis y durante la realización del proyecto europeo H2020 BEACONING, se diseñó y maduró una ampliación del perfil xAPI-SG para juegos geoposicionados. Este perfil ha sido presentado en el artículo Pérez-Colado, V. M., Rotaru, D. C., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2018). **Learning analytics for location-based serious games**. *2018 IEEE Global Engineering Education Conference (EDUCON)*, 2018-April, 1192–1200.

Para diseñar este perfil hemos abordado dos ideas. Por un lado, el perfil tiene que permitirnos añadir el contexto del jugador en las trazas determinado por su *localización* y su *orientación*. Para conseguirlo, utilizamos el mismo procedimiento que el propuesto en (Davis, 2020), añadiendo las coordenadas geográficas (latitud y longitud) en la sección de extensiones del contexto de la traza xAPI utilizando el formato GeoJSON, y además permitimos incluir la orientación del jugador basada en la información de la brújula del dispositivo. Por otro lado, en estas nuevas trazas hemos incrementado el vocabulario a partir de nuestro modelo para juegos geoposicionados y sus acciones. Consideramos que, aun así, es un enfoque flexible aplicable a muchos estilos de juego geoposicionado además de ser un modelo compatible con los juegos de aventura.

Los diferentes objetos permiten identificar los elementos reales asociados a los elementos geoposicionados, pudiendo ser de forma genérica un *place* (en castellano, lugar) o uno de los siguientes valores de alto nivel: *building* (edificio), *urban area* (zona urbana fuera de un edificio), *green area* (zona verde) o *water* (agua). Además de estos lugares, también es posible utilizar la categoría POI (punto de interés) para aquellos elementos concretos del mundo real que se usan en el juego, tales como monumentos, lugares históricos o lugares de referencia (como, por ejemplo, la entrada de una boca de metro). Sobre estos objetos se realizan las acciones geoposicionadas, identificadas por los verbos *enter*, *exit* y *look* (entrar, salir y mirar). Con estos verbos se crean las diferentes trazas que incorporan además la extensión para tener en cuenta la localización en la que se produjo la interacción, y, en el caso de mirar, también se incluye la orientación.

Además de las acciones específicas sobre elementos del juego concreto, trazar la posición del jugador a lo largo del tiempo puede ser útil para entender cómo juegan los jugadores e incluso identificar posibles casos en los que los jugadores caminan en direcciones equivocadas o que van a sitios que no tienen relevancia para las actividades que forman parte del diseño de juego. Con este objetivo, el juego envía trazas de forma periódica con el verbo *move* (moverse) que indican la posición del jugador respecto al mapa o al punto de interés concreto en el que el jugador se encuentre en ese instante.

Por último, también se han añadido trazas que permiten analizar si el jugador está siguiendo los elementos y pistas de navegación que se hayan incluido en el mapa. Estas trazas de navegación se representan a través del objeto *direction* (dirección) y el verbo *follow* (seguir). Para la creación de las trazas es necesario además incluir el conjunto

exacto de direcciones que el jugador tiene a su disposición utilizando para ello la extensión *guide* (guía).

La siguiente **Tabla 1** resume los esquemas para la creación de las trazas geopositionadas que se han descrito en esta sección.

Acción	Actor	Verbo	Objeto	Contexto (extensiones)
Moverse	Jugador	<i>Move</i>	Lugar (*)	<i>Location</i>
Entrar en un lugar	Jugador	<i>Enter</i>	Lugar (*)	<i>Location</i>
Salir de un lugar	Jugador	<i>Exit</i>	Lugar (*)	<i>Location</i>
Mirar un lugar	Jugador	<i>Look</i>	Lugar (*) en especial POIs	<i>Location & Orientation</i>
Seguir	Jugador	<i>Follow</i>	<i>Direction</i>	<i>Location & Guide</i>

Tabla 1. Resumen de los esquemas de las trazas disponibles en el perfil para juegos serios geopositionados. (*) Lugar puede ser representado por cualquiera de los tipos identificados para este propósito.

Las trazas generadas en este perfil permiten nuevos análisis en base al contexto físico del jugador. Estas trazas se pueden visualizar en múltiples formatos, incluyendo visualizaciones genéricas mediante gráficos de barras o diagramas de progresos que son aplicables a las trazas de xAPI del perfil de juegos serios. No obstante, gracias a la extensión de la ubicación, es posible contextualizar estas trazas en un mapa lo que permite

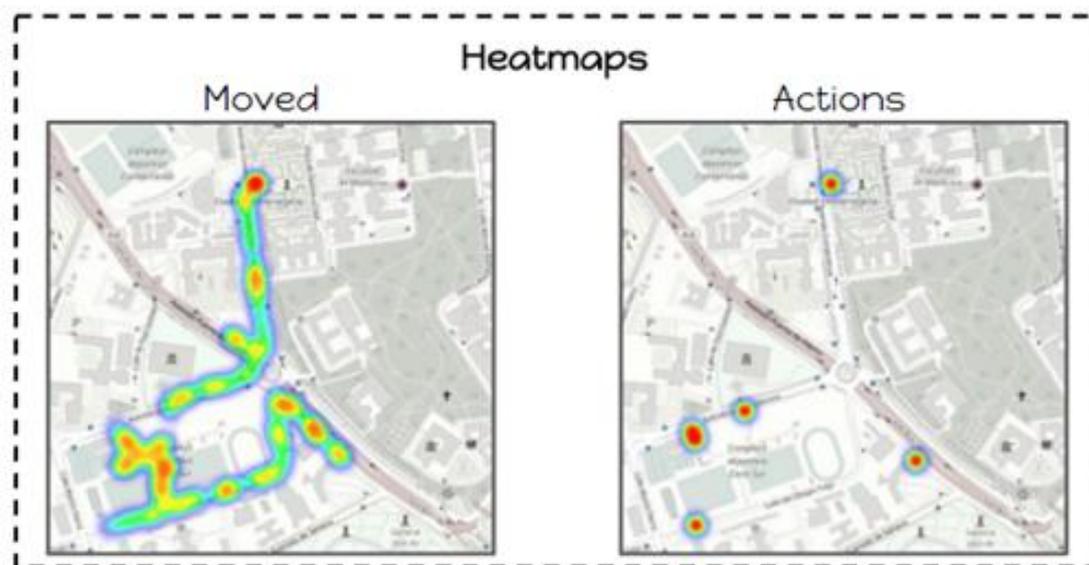


Figura 19. Dos mapas de calor representando las distintas trazas de un jugador, incluyendo sus diferentes movimientos (izquierda) y acciones geopositionadas (derecha).

realizar nuevos análisis a los profesores o investigadores que estudian la información recopilada. Como parte de los proyectos H2020 RAGE y BEACONING se diseñaron visualizaciones específicas aprovechando esta nueva capacidad mediante el uso de mapas de calor (**Figura 19**).

Las características de geoposicionamiento y la explotación de las analíticas que se pueden obtener se describen en mayor profundidad en la publicación Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). **Simplifying the creation of adventure serious games with educational-oriented features**. *Educational Technology and Society*, 22(3), 32–46.

4.3. Implementación de capacidades de integración de los juegos en procesos educativos

Como se estableció en la sección 3.3, dos de las barreras relacionadas con la aplicación de los juegos educativos son: 1) la complejidad técnica del despliegue del propio juego y la integración del mismo dentro del entorno de aprendizaje, habitualmente un LMS o campus virtual, que se utiliza como mecanismo de interacción o soporte del proceso de aprendizaje; y 2) aprovechar el potencial de los juegos serios como mecanismo de evaluación o ayuda a la evaluación de los estudiantes explotando las capacidades que ofrecen las analíticas de aprendizaje. Estas dos barreras han sido los ejes conductores para abordar este objetivo.

Como se ha descrito previamente, durante el desarrollo de la herramienta de autoría, se planteó una extensión capaz de generar analíticas de aprendizaje que podrían ser recolectadas y utilizadas para extraer datos útiles para mejorar el ciclo de vida de los juegos. La intención al abordar este aspecto es además aumentar la confianza de los educadores al usar los juegos serios, al permitirles conocer no sólo los resultados de su aplicación, sino también el proceso que ha llevado al jugador a dichos resultados. Sin embargo, para que esta solución pueda ser aplicable en entornos reales se deben simplificar también otros aspectos técnicos como son el proceso de configuración y el despliegue de los juegos en los entornos educativos.

Para comprender mejor las contribuciones de este objetivo, explicaremos el punto de partida de la extensión de analíticas de aprendizaje de forma resumida. Esta extensión es capaz de generar trazas que representan las interacciones del jugador en juegos serios y

juegos geoposicionados y enviarlas a un almacén de datos utilizando la especificación xAPI como formato de intercambio. Más concretamente, y en el contexto de la especificación xAPI, los juegos serios creados con uAdventure actúan como *Learning Record Providers* (LRP) o generadores de analíticas de aprendizaje, que envían trazas a un *Learning Record Store* (LRS) o almacén para que se puedan analizar posteriormente. Pese a que esta arquitectura propone una versión simplificada del proceso de recolección de trazas, dista de resolver el problema completamente en un entorno educativo real. Por un lado, es necesario poder identificar al jugador que está jugando y, por otro lado, es necesario identificar la actividad educativa y/o el curso dentro del cual se encuentra organizada la sesión o sesiones de juego. Originalmente, esta extensión fue creada basándose en un componente reutilizable denominado *tracker*, desarrollado dentro del marco de los proyectos H2020 RAGE y BEACONING, que se encargaba no sólo de permitir crear trazas xAPI y enviarlas a estas plataformas, sino también de interactuar con otros módulos de gestión de usuarios y de actividades educativas desarrollados como parte de estos proyectos. No obstante, el *tracker* estaba diseñado para estas dos plataformas, de modo que su aplicabilidad en un entorno real educativo diferente, como sería el despliegue e integración del juego serio dentro de un LMS, era limitado.

Partiendo de estas experiencias y manteniendo la compatibilidad con las mismas, se comenzó a desarrollar una versión evolucionada tanto del *tracker* como de la extensión de uAdventure para analíticas. Esta nueva versión tenía el objetivo de que pudiera ser utilizado con un LRS genérico para el almacenamiento de las analíticas y que, dentro de las posibilidades que nos ofrecen los estándares y especificaciones educativas actuales, se pudieran obtener los parámetros y datos de configuración necesarios (credenciales, identidad del jugador, actividad educativa bajo la que se lleva a cabo la sesión de juego, etc.). Estos datos se obtendrían de una manera simplificada y con una mínima configuración / interacción por parte de los profesores que quieran desplegar los juegos dentro de su LMS corporativo. De este modo, estas nuevas capacidades tienen por objetivo mejorar la experiencia del usuario final de uAdventure (y Unity) al utilizar los juegos como fuente de datos de analítica (LRPs), garantizando una libertad de elección del LRS donde almacenar dichas analíticas.

No obstante, el soporte actual en los LMS de los estándares y especificaciones educativas es variado. En particular, pese a que existen numerosos LMS tanto comerciales como de código abierto que declaran soportar estándares, en la actualidad no todos los LMS

integran un LRS nativamente o proporcionan un mecanismo para poder enviar trazas xAPI desde el LMS al LRS. Por otro lado, como ya se ha mencionado previamente, el paso de parámetros para poder utilizar un LRS en una LRP no está cubierta por la propia especificación xAPI. En la actualidad esta necesidad (entre otros casos de uso) ha suscitado el interés y el desarrollo de la especificación CMI5 (Miller et al., 2021), como complemento de la especificación xAPI. Sin embargo, la especificación CMI5 todavía no está suficientemente extendida en los LMS, aunque existen iniciativas como el proyecto CATAPULT⁷⁵ que tiene por objetivo fomentar la adopción de esta nueva especificación.

Por tanto, existen múltiples posibilidades y combinaciones en el uso de especificaciones y estándares técnicos educativos para el despliegue de juegos. Para facilitar las pruebas de despliegue de juegos en este ecosistema tecnológico tan diverso y también para facilitar la validación de los juegos creados con uAdventure, dentro del grupo de investigación se ha desarrollado la herramienta SIMVA (*SIMple VALidator*) para orquestar todo el proceso (I. J. Pérez-Colado et al., 2019). Un análisis de las especificaciones técnicas educativas y de las oportunidades que nos ofrecen a la hora de facilitar el despliegue de juegos y recopilación de analíticas combinando uAdventure y SIMVA se describe en la publicación Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021a). **Democratizing Game Learning Analytics for Serious Games**. *International Conference on Games and Learning Alliance (GALA)*, 164–173. A continuación, expondremos de forma resumida las diferentes funcionalidades que hemos habilitado a través de uAdventure en relación con SIMVA, la adquisición de analíticas y el soporte de estándares.

La primera contribución fue la extensión de SIMVA para uAdventure, con la que permitimos que nuestros usuarios puedan gestionar de forma centralizada el proceso, es decir, extraer valor de las analíticas de aprendizaje proporcionando un sistema real, y simplificando su configuración y el desarrollo de la actividad. SIMVA pretende ser un ecosistema completo de analíticas de aprendizaje orientada a los procesos de validación de juegos serios y es capaz de proporcionar, entre otras funcionalidades: gestión de usuarios, configuración de actividades (incluidos cuestionarios previos y/o posteriores), recolección de analíticas y mecanismos básicos de análisis. Esta integración se ha

⁷⁵ <https://adlnet.gov/projects/cmi5-CATAPULT/>

realizado de forma que el usuario puede utilizar un configurador en tres pasos para preparar su actividad: definir los diferentes cuestionarios pre-post que se quieran realizar, habilitar diferentes características para el almacenamiento de los datos y configurar los usuarios que podrán acceder a la actividad de forma anónima utilizando un sistema de identificadores. SIMVA, además, es capaz de integrarse a través de IMS LTI con los LMS, permitiendo identificar al estudiante y devolver resultados, a la vez que facilita el acceso directo a los análisis y datos recogidos con xAPI a los educadores.

La segunda contribución fue el rediseño del *tracker* para dotarlo de nuevas funcionalidades como las que podemos encontrar en soluciones de analíticas similares en el ámbito comercial y de nuevas capacidades de asincronismo, resiliencia y robustez. En primer lugar, realizamos el estudio de los diferentes *trackers* y LRS disponibles en el mercado (las herramientas analizadas se exponen en el artículo V. M. Pérez-Colado, Pérez-Colado, Martínez-Ortiz, et al., 2021a). Estos *trackers* de terceros normalmente proporcionan una interfaz de programación de bajo nivel que se centra en la construcción y manipulación del modelo de datos asociado a las trazas xAPI y su envío a un LRS. Esto hace que su uso sea más complejo que en nuestra aproximación de alto nivel construida para dar soporte a los conceptos y el vocabulario concreto introducido en el perfil de xAPI para juegos serios. Por otro lado, la mayoría de los *trackers* estudiados contaban con deficiencias tales como: tener una arquitectura síncrona que paraliza el juego durante el envío de trazas; incluir dependencias que imposibilitan su uso en plataformas web; o no implementar soporte completo de la especificación xAPI. El último aspecto mejorable es su fiabilidad, pues no consideran aspectos como su resiliencia, haciéndolos inviables en entornos tan heterogéneos como los educativos donde, por ejemplo, no siempre se puede contar con una conexión de red estable o con un ancho de banda suficiente. El rediseño de nuestro *tracker* se enfoca en ofrecer una solución a los problemas identificados a través de cinco aspectos: un alto nivel de abstracción para sus usuarios, un enfoque orientado a videojuegos (siendo compatible con Unity, asíncrono y con soporte multiplataforma), un completo soporte para LRS y mecanismos de autorización, flexibilidad para poder trabajar en diferentes modos y una construcción resiliente para un buen funcionamiento en diferentes entornos y contextos.

La tercera contribución se centra en simplificar el empaquetamiento y distribución de los juegos de forma estandarizada. Teniendo en cuenta el estado de adopción de las especificaciones de xAPI en los LMS hemos optado por dar soporte al empaquetamiento

usando IMS LOM e IMS CP de forma inmediata. La idea es ir implementando el soporte de CMI5 para que los juegos puedan utilizarse como actividades xAPI en plataformas que ya tengan soporte al estándar o que hagan de mediadoras (como SCORM Cloud⁷⁶) hasta la futura adopción de forma generalizada de CMI5 en LMS. Para dar soporte a IMS LOM e IMS CP se han implementado nuevos asistentes y un exportador que utiliza el soporte de HTML5 y WebGL en Unity para construir objetos educativos que pueden incorporarse de forma sencilla en un LMS. Estos objetos educativos cuentan con soporte de analíticas de aprendizaje, pues durante la reimplementación del *tracker* se habilitó su funcionamiento en WebGL, haciendo posible que se conecten a un LRS, previa configuración manual, aunque sin poder identificar a los usuarios, o a SIMVA, pudiendo en este caso identificar a los usuarios a través del código anónimo proporcionado por la plataforma. Por otra parte, como avance del soporte de CMI5 se ha preparado la herramienta para que pueda recibir la información necesaria a través de URI Protocol⁷⁷, que es el mecanismo utilizado por CMI5 para poder abarcar tanto plataformas web como nativas. Para ello, a través de la creación automatizada de diferentes instaladores y configuradores los juegos son capaces de asociar esquemas únicos de URI que les permiten detectar cuándo lanzarse e interpretar la información que facilite la plataforma para la identificación del usuario y el destino web donde enviar las trazas.

Como resultado final de estas tres contribuciones hemos desarrollado tres modos de funcionamiento del *tracker* y la extensión de analíticas de aprendizaje con SIMVA. En primer lugar, el modo LRS habilita, previa configuración, que el *tracker* pueda conectarse con un LRS y enviarle datos. De este modo el *tracker* es funcional en LMS, pero con la limitación de que no tiene la capacidad de identificar a los usuarios. En segundo lugar, el modo LMS está orientado a poder establecer una comunicación bidireccional con un LMS a través de plataformas mediadoras hasta que los LMS den soporte al estándar. De esta forma puede propagarse la información del usuario hasta el juego utilizando, por ejemplo, una combinación de LTI y CMI5. Por último, el modo SIMVA está orientado a que los usuarios trabajen directamente con la plataforma de validación, proporcionando una experiencia de usuario que simplifica el proceso de configuración y el uso de SIMVA. Este modo permite hacer uso completo de todas las capacidades que ofrece SIMVA, en

⁷⁶ <https://rusticsoftware.com/products/scorm-cloud/>

⁷⁷ Guías y procedimientos de registro de URI Schemes: <https://datatracker.ietf.org/doc/html/rfc7595>

especial, la orquestación y la gestión de experimentos de validación de juegos serios. A través de estos tres modos conseguimos adaptarnos al estado actual de la adopción del estándar xAPI y especificaciones complementarias como CMI5, dando soporte a analíticas de aprendizaje, simplificando la configuración del proceso y ofreciendo la flexibilidad y libertad suficiente a los desarrolladores.

Como mencionamos en nuestra sección de objetivos, esta contribución no sólo es importante para uAdventure, sino que también es útil para que los desarrolladores de Unity puedan disponer de herramientas con las que introducir analíticas de aprendizaje de forma adecuada y simplificada en sus juegos. De hecho, en la sección 4.5.3 exploramos brevemente su uso en el juego Articoding, juego que busca fomentar la programación entre jóvenes.

4.4. Desarrollo de una extensión para géneros híbridos y minijuegos

Durante el proceso de desarrollo de un juego serio la elección del estilo de juego, así como de sus mecánicas y dinámicas específicas, es determinante para su efectividad a la hora de conseguir sus objetivos educativos. Estos estilos de juego vienen determinados en gran medida por la elección del género del juego, que dará un enfoque a estas mecánicas y dinámicas, ayudando además a simplificar el proceso de desarrollo al acotar las posibles opciones. De hecho, el desarrollo de uAdventure se centra en el género de juegos de aventura pues, como hemos visto en el estado del arte, es uno de los géneros que ofrecen un mejor equilibrio entre sus capacidades educativas y la complejidad necesaria para crear los juegos en un entorno de autoría. No obstante, esta elección de género no es limitante, sino que en muchos casos a través de la hibridación (es decir, la incorporación de estilos de juego de otros géneros) es posible mejorar la efectividad del juego. Por ejemplo, se puede lograr más atracción a través de la innovación en el estilo de juego (Aslan & Balci, 2015) al utilizar dinámicas y mecánicas más adaptadas a nuestros objetivos educativos.

La posibilidad de crear juegos híbridos es uno de los objetivos de esta tesis, que motivó el desarrollo de uAdventure utilizando Unity, con el objetivo de facilitar la inclusión de nuevos elementos y estilos de juego. Este aspecto es muy útil de cara al futuro de la herramienta (para su mantenimiento y sus mejoras), pero también abre las puertas a que los desarrolladores puedan aprovecharlo para implementar otras mecánicas o estilos en sus juegos, e incluso que estas nuevas extensiones puedan ser compartidas y reutilizarlas

entre desarrolladores de la comunidad de uAdventure. No obstante, sin un diseño cuidadoso tanto desde el punto de vista de metáfora de integración dentro del proceso de edición / juego, como desde el punto de vista de desarrollo de estos nuevos elementos, lograr dicho objetivo puede ser demasiado costoso para el beneficio a obtener.

Para resolver este problema se ha optado por ofrecer hibridación a través del modelo de minijuegos. Los minijuegos, como su nombre indica, son juegos más cortos y normalmente de baja complejidad que son independientes del juego principal. Normalmente estos minijuegos proporcionan una dinámica completamente distinta al juego original, introduciendo nuevas mecánicas y estilos de juego que pueden pertenecer a otros géneros, aunque pueden guardar relación con el juego principal. Puesto que su modelo se orienta a ser independientes, es posible realizar una integración mucho más simple y encapsulada que si tratamos de ampliar las funcionalidades de la herramienta, siendo además mucho más apropiado para ser utilizado por nuestros usuarios. En el artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021b). **Extending Narrative Serious Games Using Ad-Hoc Mini-games**. In W. Zhou & M. Yi (Eds.), *Advances in Web-Based Learning – ICWL 2021: 20th International Conference, ICWL 2021, Macau, China, November 13–14, 2021, Proceedings* (pp. 63–74) exponemos la contribución que hemos llevado a cabo para realizar su implementación.

El soporte a los minijuegos se ha incluido a través de la creación de una nueva extensión de uAdventure. Esta extensión ha sido construida especialmente para que los desarrolladores que conocen Unity puedan obtener el máximo beneficio del uso de uAdventure, al disponer de todas sus funcionalidades y la libertad absoluta para saltar a Unity cuando lo necesiten. Por ello, la creación de los minijuegos se hace sobre escenas Unity directamente, donde el desarrollador experimentado puede construir las funcionalidades que desee. Cuando el desarrollador lo necesite, puede saltar de forma transparente a estas escenas, siendo sólo necesario establecer una cámara en ellas (cuya configuración se mantiene) que indique el punto de vista para comenzar con el minijuego. En este ámbito, además, el desarrollador dispone de acceso a las principales interfaces de manipulación de uAdventure a través de las cuales puede acceder, por ejemplo, al estado del juego o a la ejecución de efectos sobre el modelo de uAdventure.

Dada su naturaleza independiente, es posible que los minijuegos sean utilizados varias veces durante los juegos, pudiendo variar sutilmente entre sus apariciones, por ejemplo, modificando su complejidad, comportamiento o estilo visual. Por otro lado, para que el uso de minijuegos tenga sentido, al finalizar su ejecución su resultado debe quedar disponible en el estado del juego para que se pueda adaptar la historia principal del juego a dicho resultado. Estas dos afirmaciones en conjunto ponen de manifiesto la necesidad de una comunicación bidireccional entre uAdventure y el minijuego. El enfoque planteado para realizar esta comunicación utiliza los sistemas ya existentes, usando el sistema de *efectos* de uAdventure para introducir los parámetros en el *estado del juego* y en los minijuegos las interfaces (principalmente el *estado del juego*) y los sistemas de *condiciones* y *efectos* para leer las configuraciones y comunicar los resultados. Con ello logramos una comunicación bidireccional evitando incrementar innecesariamente la complejidad del proceso.

En este modelo de comunicación, existe cierta similitud con el esquema de caja negra, donde se introducen las configuraciones y datos iniciales y se extraen los resultados. Este hecho hace que sea especialmente relevante el soporte de analíticas de aprendizaje en los minijuegos para poder conocer las interacciones que realizan los jugadores hasta completarlos. Por ello, una de las interfaces a las que puede acceder el desarrollador es la extensión de analíticas de aprendizaje. Gracias a ello, los desarrolladores que usen uAdventure pueden utilizar nuestro *tracker* de alto nivel para generar analíticas de aprendizaje que ayuden a monitorizar lo que ocurre en los minijuegos.

Desde el punto de vista del desarrollo de los juegos serios, esta integración además encaja con su carácter multidisciplinar. Permite que los expertos no técnicos puedan utilizar uAdventure para dar forma al contenido educativo y delegar a los desarrolladores, de forma además muy encapsulada, la creación de los minijuegos que aborden objetivos educativos específicos. Al encontrarse en un entorno más limitado para los desarrolladores podremos proporcionarles unos requisitos más simples y limitados, y, en consecuencia, reducir los posibles errores creados por un mal entendimiento entre ambos perfiles. Además, este acercamiento también permite que los desarrolladores con experiencia en Unity puedan aprovechar uAdventure para sus juegos y prototipos, reutilizando su modelo como plataforma de desarrollo de sus juegos o incluso dejando a uAdventure en una posición secundaria como una librería o extensión más que ofrece funcionalidades complementarias.

Para concluir las contribuciones de este objetivo hemos puesto en práctica estas funcionalidades como parte de nuestros experimentos de validación de la herramienta y de videojuegos particulares. En la siguiente sección exploraremos estos casos y los resultados obtenidos.

4.5. Validación de la aplicabilidad de uAdventure y de los juegos desarrollados

En esta sección trataremos las diferentes pruebas y experimentos realizados durante esta tesis para evaluar nuestras soluciones. Esta evaluación tiene como objetivo garantizar y probar la efectividad de los resultados con respecto a los objetivos de la tesis. Es decir, evaluar el resultado de la implementación del modelo en uAdventure, así como sus diferentes extensiones y componentes. A diferencia de las otras secciones que se enfocan en un sólo objetivo y sus resultados, en esta sección reunimos las diferentes pruebas que hemos realizado a lo largo del desarrollo del trabajo de tesis. Su objetivo ha sido mejorar tanto el modelo conceptual como la aplicabilidad de la herramienta uAdventure al aplicarla en diferentes situaciones. Se ha tenido en cuenta la experiencia y realimentación obtenida de los usuarios, madurando así la herramienta y los juegos para lograr soluciones más efectivas.

En las siguientes secciones analizaremos el proceso de evaluación, que se ha dividido de forma general entre las pruebas de la herramienta uAdventure y las pruebas de los juegos creados con ella. En general, estas pruebas comienzan como pilotos y amplían sus objetivos progresivamente conforme se ha madurado la herramienta.

4.5.1. Pruebas de la herramienta uAdventure

Las pruebas con usuarios realizadas con uAdventure se han realizado en diversas iteraciones que han incluido workshops y pruebas piloto para la creación de juegos, en los que se ha abordado su ciclo de vida y su autoría utilizando la herramienta como apoyo al aprendizaje. Estas pruebas se han publicado en el artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). *Simplifying the creation of adventure serious games with educational-oriented features*. *Educational Technology and Society*, 22(3), 32–46., que también forma parte de la sección 4.1 de este capítulo. A continuación, se resumen parte de estas pruebas y se

aprovecha para expandir algunas de las pruebas más recientes y juegos realizados como resultado del proceso (y que todavía no se han publicado).

4.5.1.1. Pruebas piloto de uAdventure

Tras finalizar el desarrollo de las funcionalidades necesarias para la creación de aventuras gráficas, se realizaron varias pruebas piloto con el objetivo de validar el desarrollo y como paso previo a unas pruebas a mayor escala. En estas pruebas piloto el objetivo es probar la herramienta uAdventure y verificar la factibilidad del desarrollo autónomo por parte de los usuarios, de modo que se detecten los posibles errores que pudieran entorpecer pruebas a mayor escala. Nuestra metodología consistió en proporcionar a los usuarios además del *software*, una guía de inicio rápido, el manual completo de uAdventure, una serie de ejemplos a desarrollar como objetivo principal y un guion para que pudieran realizar un juego completo de forma autónoma, además de los recursos gráficos (*assets*) necesarios para ello. En las pruebas, los estudiantes debían desarrollar al menos un juego simple y algunos otros ejemplos parciales. Al final se realizó una encuesta que incluía una catalogación del perfil (para tener en cuenta su conocimiento y experiencia previa), diversas cuestiones para comprobar el grado de completitud de las tareas y una encuesta de satisfacción para medir el grado de dificultad encontrado al utilizar cada funcionalidad individual que hubieran utilizado en uAdventure (entre fácil, normal o difícil).

En las pruebas piloto iniciales participaron 10 usuarios de perfiles variados incluyendo 2 usuarios de rama de artes (como diseñadores gráficos), 6 usuarios de la rama de informática y 2 usuarios que no se centraban en ninguna rama técnica. Dentro de los resultados más destacables pudimos observar que, en general, los usuarios habían sido capaces de realizar desarrollos de forma autónoma, completando de media el 95% de los ejemplos y habían puesto en práctica de media un 68% de las funcionalidades disponibles de uAdventure. De estas funcionalidades menos de un 1% fueron identificadas como tareas difíciles. Además, los usuarios nos ayudaron a identificar diversos problemas a mejorar antes de las siguientes pruebas con usuarios.

4.5.1.2. Pruebas como herramienta de apoyo a la enseñanza: analíticas y juegos geoposicionados

Tras la revisión de la herramienta en base a los resultados de las pruebas piloto, las pruebas con usuarios se realizaron en formato de *workshop* para aprender el desarrollo de juegos serios, utilizando uAdventure como herramienta de apoyo a la enseñanza. Estos

workshops variaron en duración y objetivos, siendo los primeros *workshops* más cortos y con objetivos muy acotados, para posteriormente ampliar su duración y ambición conforme se consolidaban las diferentes partes de la herramienta.

Los primeros *workshops* se realizaron durante la última etapa de implementación para lograr nuestro primer objetivo (antes de finalizar la revisión de las analíticas de aprendizaje), centrándose en consolidar las funcionalidades de aventura y probar que a través del proceso de enseñanza asistido con uAdventure los participantes eran capaces de implementar un juego serio básico. En primer lugar, los participantes recibían una lección de una a dos horas en las que se les daba una introducción a los juegos serios y se explicaba la parte de aventura de la herramienta, su modelo narrativo. A continuación, se enseñaba de forma práctica el proceso de diseño e implementación de un juego serio a través de unos ejemplos guiados que los participantes debían implementar de forma autónoma al finalizar. Como resultado, los participantes construyeron un juego serio predefinido y, en los casos en los que la duración lo permitía, podían implementar otros ejemplos más complejos. Durante esta iteración participaron 50 alumnos en la Escuela Superior de Diseño (ESD) y en la UCM (**Figura 20**).

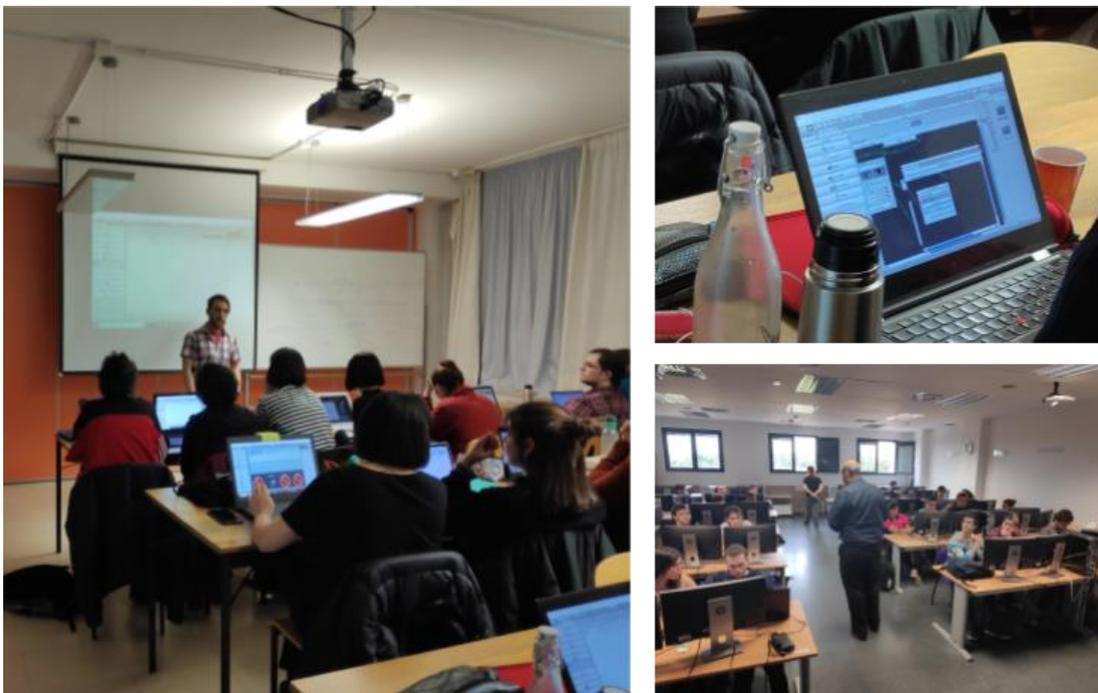


Figura 20. Fotos tomadas durante la realización de *workshops* para enseñar a realizar juegos serios con el apoyo de uAdventure en la ESD y en la UCM.

La segunda iteración de las pruebas de uAdventure como herramienta de apoyo a la enseñanza se realizó tras la finalización de nuestro primer objetivo, incluyendo la extensión de analíticas de aprendizaje. En esta iteración se podían cubrir más aspectos de cara a la aplicabilidad de los juegos serios y su ciclo de vida. Para abordarlos, se usó un enfoque práctico en el que los participantes diseñan un juego serio eligiendo sus propios objetivos educativos y lo implementan usando uAdventure durante varias sesiones con asistencia de los profesores. Una vez que los alumnos habían sido capaces de crear un juego, se les daba una lección sobre analíticas de aprendizaje y su uso a través de la extensión de uAdventure, de modo que los participantes tenían que aplicar estas analíticas a su juego. En esta iteración hubo 32 participantes que desarrollaron 7 juegos con uAdventure en los que utilizaban el modelo de analíticas de aprendizaje.

Una vez que se tenía una versión estable de la herramienta uAdventure incluyendo las analíticas, se pasó a probar su uso para apoyar a la enseñanza en la creación de juegos geoposicionados. En esta iteración se diseñó una nueva lección con ejemplos específicos en los que se abordaban los detalles de este género y su implementación a través de uAdventure. Esta lección además introdujo una serie de buenas prácticas en juegos geoposicionados que habíamos aplicado en algunos de nuestros prototipos de forma experimental como, por ejemplo, incluir un guía en el juego, las rutas con navegación y los elementos coleccionables. Estas buenas prácticas pretenden hacer frente a posibles problemas y situaciones detectadas durante nuestras primeras pruebas con prototipos. Estas pruebas incluyeron a 36 participantes que desarrollaron 8 juegos de aventura y 8 juegos geoposicionados. Fue especialmente interesante el caso de los juegos geoposicionados pues las buenas prácticas introducidas fueron bien acogidas e implementadas por los participantes, que crearon juegos no-lineales y con diferentes finales. En estos *workshops* se detectó que los participantes tendían a crear juegos que, pese a su corta duración, contaban con recorridos demasiado largos, de modo que se incluyó este aspecto en las buenas prácticas. Un detalle que mencionar en esta iteración es que durante esta tercera iteración se preparó el primer prototipo de la extensión de minijuegos. Esto posibilitó que pudiera utilizarse de forma experimental por un grupo de alumnos con experiencia previa en Unity, que crearon el juego serio Vuela para aprender protocolos aeroportuarios. Este caso se expone brevemente dentro del artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021b). **Extending Narrative Serious Games Using Ad-Hoc Mini-games.**

In W. Zhou & M. Yi (Eds.), Advances in Web-Based Learning – ICWL 2021: 20th International Conference, ICWL 2021, Macau, China, November 13–14, 2021, Proceedings (pp. 63–74).

Como conclusión de estas tres iteraciones de pruebas se logró consolidar y madurar uAdventure probando su utilidad para crear juegos, mejorando el aprendizaje del proceso de desarrollo y la aplicabilidad de los juegos serios. En total se desarrollaron 22 juegos de 10-15 minutos de duración y una complejidad narrativa de unas 8 veces mayor a la presente en los ejemplos guiados que se realizan durante la primera lección del workshop (en base a la diferencia cuantitativa de elementos del modelo). Estos talleres prácticos demostraron que los participantes comprendían mejor el proceso de desarrollo de juegos serios y que uAdventure permitía que lo pusieran en práctica.

Las siguientes iteraciones de los *workshops* se desarrollaron tanto en los meses previos como durante la pandemia producida por la COVID-19, lo que nos permitió probar uAdventure en el apoyo a la enseñanza de forma *online*. En esta nueva etapa se sentó la base para la creación de una comunidad virtual en Discord⁷⁸ para que los usuarios puedan recibir asistencia directa de los desarrolladores y antiguos usuarios de uAdventure. Esta comunidad cuenta hoy en día con más de 30 usuarios habituales.

4.5.1.3. Pruebas de la herramienta en cursos: uAdventure, analíticas avanzadas y minijuegos

Tras los primeros *workshops* se consolidaron las contribuciones de nuestro primer y segundo objetivos habiendo además madurado el uso de uAdventure como herramienta de apoyo para enseñar a diseñar y crear juegos serios. Sin embargo, este proceso de enseñanza apenas abordaba el tema de la aplicabilidad de los juegos. En las siguientes iteraciones, se incluyó esta aplicabilidad como parte del ciclo de vida de los juegos serios. Estos *workshops* tienen una mayor duración en su parte práctica, permitiendo así dar una mejor visión de este ciclo de vida abordando la aplicabilidad y ofreciendo a los usuarios más tiempo para el diseño de sus objetivos educativos. De esta forma los participantes pueden crear juegos más ambiciosos y experimentales, e incluso aprovechar el uso de las

⁷⁸ Discord (<https://discord.com/>) es una plataforma para la creación de comunidades online con mensajería y comunicación de voz entre muchas otras funcionalidades. La comunidad de uAdventure es accesible (previo registro) a través del enlace: <https://discord.gg/D2ZWteP>

nuevas funcionalidades como, por ejemplo, la hibridación de juegos a través de minijuegos.

Como se ha descrito previamente, la importancia de las analíticas de aprendizaje es determinante para la futura aplicabilidad de los juegos. Con ellas es posible entender mejor el proceso de aprendizaje en base a las interacciones de los jugadores, aplicando técnicas de visualización y análisis que permitan extraer conclusiones. Sin embargo, en la práctica, en actividades limitadas en el tiempo como los talleres (*workshops*) es muy complejo asimilar el concepto de analíticas de aprendizaje y ponerlas en práctica. Esto implica formar a los participantes tanto en procesos de diseño de analíticas (para que estas sean eficaces en el ciclo de vida de sus juegos) así como en su implementación en los juegos (para que puedan aplicarlas de forma real). No obstante, este proceso puede ser simplificado, pues de cara a la aplicabilidad de las analíticas, conocer y aplicar correctamente el proceso de diseño de las analíticas es más relevante de cara al ciclo de vida del juego que las cuestiones técnicas de su implementación. De hecho, los beneficios de comprender los detalles técnicos pueden obtenerse o mejorarse a través del uso de entornos como uAdventure que los automaticen, simplifiquen y optimicen. Las herramientas simplifican los detalles técnicos, lo que permite invertir más tiempo en el proceso de diseño y buenas prácticas de las analíticas. Centrándonos en estos aspectos mejoramos la comprensión del ciclo de vida del juego.

Para aplicar este modelo de enseñanza, uAdventure, sus extensiones y el *tracker* jugaron un papel fundamental, para crear y aplicar una metodología publicada en el artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2021). **A Tool Supported Approach for Teaching Serious Game Learning Analytics**. *2021 IEEE Frontiers in Education Conference (FIE)*, 1–8. A continuación expondremos de forma resumida la metodología y los resultados de aplicarla en nuestros cursos.

La metodología parte de la experiencia previa obtenida en los *workshops* pero ampliando su enfoque al de un curso completo. En este curso completo se plantean tres partes: una etapa de formación en juegos serios y analíticas de aprendizaje, una etapa de diseño y construcción de juegos serios y una última etapa de aplicabilidad de analíticas. En la primera etapa, los jugadores reciben lecciones teóricas sobre juegos serios y analíticas de aprendizaje, para posteriormente analizar diferentes juegos serios. En la segunda etapa

los participantes aplicarán sus conocimientos para diseñar y crear dos juegos serios, uno de género de aventura y otro geoposicionado, que implementarán con uAdventure una vez recibida dicha formación. Además, debían incluir analíticas de aprendizaje desde el diseño inicial y aprender a usar el *tracker* en uAdventure. La formación se completó con otras clases para el uso de SIMVA y la hibridación utilizando la nueva extensión para minijuegos. Una vez implementados ambos juegos, en la tercera etapa los jugadores deben elegir uno de los juegos y extenderlo para dar soporte a analíticas de aprendizaje. Para poner en práctica las analíticas, los participantes utilizan la extensión de SIMVA para configurar su actividad y disponer de un entorno de pruebas. Finalmente, los participantes probaban estos juegos intercambiándose con los otros grupos, comprobando a través de SIMVA si sus jugadores han podido completar sus juegos, comprendiendo el grado de satisfacción de sus objetivos educativos y detectando posibles errores en sus juegos.

Este uso en los cursos de las analíticas de aprendizaje permitió madurar estas funcionalidades y probar su aplicabilidad con usuarios contribuyendo a probar el tercer objetivo de esta tesis. Esta metodología se probó en las asignaturas de juegos serios (del Grado en Desarrollo de Videojuegos de la UCM) y de una forma mucho más breve en la asignatura de e-Learning (del Máster en Ingeniería Informática, también de la UCM). Como resultado tuvimos 9 y 7 participantes respectivamente, aunque estos últimos no desarrollaron juegos completos. Los participantes del curso de juegos serios consiguieron desarrollar un total de 6 juegos de aventura y geoposicionados, de los cuales 3 fueron ampliados y puestos en práctica en el entorno final de analíticas. Aunque el resultado es muy limitado, en esta puesta en práctica todos los alumnos consiguieron aplicar sus juegos serios, incluidas las analíticas, y con ello comprobar la aplicabilidad de sus juegos y descubrir errores de diseño o implementación analizando las posibles causas.

La mayor duración y cobertura completa del ciclo de vida de los juegos serios a través de uAdventure permitió no sólo probar la efectividad de la metodología sino también que los juegos resultantes adquirieran más calidad y aplicabilidad. Como resultado, se han creado varios juegos entre los que destacan tres, que debido a su madurez y calidad nos ha permitido realizar experimentos con usuarios reales.

El primero de estos juegos es el juego de La Entrevista. La Entrevista es un juego que tiene como objetivo concienciar acerca de la discriminación de las mujeres en trabajos

del ámbito de ciencias (STEM). Este juego destaca por ser un juego con múltiples finales y conciencia a los jugadores poniéndoles como receptores de la discriminación. A diferencia de los otros dos juegos que veremos a continuación, todavía no se han terminado los experimentos con usuarios reales, pero está prevista su prueba con alumnos de un centro educativo catalán durante el mes de marzo y abril de 2022. Los otros dos juegos desarrollados, llamados La Mansión Paranormal y Guerra Civil UCM, ya han sido probados con usuarios reales en sendos casos de estudio. En la siguiente sección se exponen las diferentes pruebas realizadas con juegos creados con uAdventure, incluyendo diversas pruebas piloto (ver 4.5.2.1 y 4.5.2.2), las pruebas realizadas con el juego geopositionado El Señor De Los Arbustos (ver 4.5.2.3) y las pruebas realizadas con los juegos La Mansión Paranormal y Guerra Civil UCM (ver 4.5.2.4).

4.5.2. Pruebas y casos de estudio con juegos creados con uAdventure

Las pruebas con juegos tienen como objetivo validar la aplicabilidad de los juegos serios creados con uAdventure y sus extensiones. Estas pruebas se realizaron en diferentes etapas incluyendo desde experimentos piloto para depuración de la jugabilidad hasta experimentos a mayor escala que permiten validar tanto los juegos como los diferentes mecanismos de integración. Estas pruebas se han hecho con juegos de aventura y con juegos geopositionados. Las mecánicas de juegos de aventura reutilizan la experiencia previa de e-Adventure de modo que las pruebas se orientan más a la aplicabilidad de los mecanismos de integración, mientras que las mecánicas para juegos geopositionados han sido desarrolladas como parte de esta tesis y las pruebas tienen como objetivo tanto validar su efectividad como su aplicabilidad. A continuación, expondremos estas pruebas y los diferentes objetivos concretos de cada una.

4.5.2.1. Prueba piloto de juegos geopositionados: Compluaventuras

Como parte del proceso de validación de la idea seminal de este trabajo de tesis se diseñó e implementó un juego serio geopositionado llamado Compluaventuras que tenía como objetivo validar las diferentes mecánicas del prototipo de la extensión para juegos geopositionados. Este juego se expone en el artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2017). **Simplifying location-based serious game authoring**. In J. M. Doderó, M. S. Ibarra Sáiz, & I. Ruiz Rube (Eds.), *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM)* (pp. 1–9). El juego ponía en práctica

los diferentes tipos de elementos geoposicionados, acciones geoposicionadas, mecánicas (como la navegación) y códigos QR en un juego para enseñar a los usuarios las diferentes instalaciones deportivas de la Universidad Complutense de Madrid (UCM).

La prueba piloto realizada con este juego utilizó la metodología de validación utilizando cuestionarios antes y después del experimento, combinado con la recolección de las diferentes analíticas de geoposicionamiento. El experimento realizado con 5 participantes ofreció resultados prometedores, pudiendo probar su aplicabilidad puesto que los usuarios entendieron las mecánicas y pudieron aplicarlas para completar sus juegos, y su potencial para el aprendizaje al haber mejorado sus conocimientos sobre las instalaciones de la UCM. Por otro lado, también se pudo probar la aplicabilidad de las analíticas a través de su posterior uso en análisis y mapas de calor. Por último, estos experimentos también ayudaron a formar una base de las buenas prácticas, incluyendo el uso de tutoriales, la figura del guía como asistente durante el juego y pusieron en evidencia la necesidad de abordar la ausencia de actividad de juego en los caminos largos. Esta última situación se acabó resolviendo mediante el uso de elementos coleccionables y diferentes personajes secundarios.

4.5.2.2. Prueba piloto de la herramienta base: *First Aid Game*

La aplicabilidad de los juegos de aventura y las analíticas de aprendizaje en uAdventure se validó dentro del experimento publicado en Alonso-Fernández et al., (2020). Puesto que la jugabilidad de los juegos de aventura ya fue probada en e-Adventure, este experimento aprovecha para demostrar las capacidades de la herramienta para extender el ciclo de vida de juegos implementados en e-Adventure, a la vez que pone a su disposición las nuevas funcionalidades educativas como las analíticas de aprendizaje para mejorar la aplicabilidad de los juegos. En este experimento se utilizó el juego serio diseñado por el grupo e-UCM llamado *First Aid Game* (Marchiori, Ferrer, et al., 2012) en el que los jugadores aprenden los diferentes mecanismos de primeros auxilios en formato de aventura “*point and click*”. Sobre este juego se implementaron las analíticas de aprendizaje incluyendo, además de las analíticas preconfiguradas de uAdventure, diferentes identificadores para decisiones y completables para conocer el estado de la actividad. Aunque los resultados de este experimento se enfocaron en extraer el valor de las analíticas, desde el punto de vista de esta tesis los resultados demostraron la aplicabilidad de uAdventure y de la extensión de analíticas de aprendizaje.

Esta prueba piloto permitió además verificar la utilidad de nuestras contribuciones para dar una nueva vida a juegos educativos desarrollados antiguamente. De esta forma se saca aún más provecho a las inversiones realizadas durante el periodo de vida de e-Adventure. Otros juegos actualizados con este propósito además de *First Aid Game* han sido *Fire Protocol* (Moreno-Ger, 2008) y *Checklist* (Moreno-Ger et al., 2008). Como casos anecdóticos también han mostrado interés antiguos usuarios como Anibal Parra Ruiz para actualizar su juego *Piraton fuit la Terre*⁷⁹, o Marta Martín del Pozo para actualizar sus cursos sobre juegos serios utilizando uAdventure (Martín del Pozo, 2014).

4.5.2.3. Experimento con juegos geoposicionados: El Señor De Los Arbustos

A diferencia de los juegos de aventura, las mecánicas diseñadas para juegos geoposicionados sólo habían sido validadas de forma limitada a través del piloto realizado en la etapa de prototipo. Con la revisión de la autoría y las mecánicas surgió la necesidad de realizar un nuevo juego serio con el que probar las capacidades de la herramienta y la aplicabilidad de los juegos generados en ella con usuarios y móviles reales, tanto Android como iOS. Además, con este nuevo juego también se pretendía estudiar la efectividad de los juegos geoposicionados y compararla con otros métodos de enseñanza.

El nuevo juego que se diseñó se llamó El Señor De Los Arbustos y fue desarrollado en colaboración con la Unidad de Campus y Medio Ambiente⁸⁰ del Vicerrectorado de Tecnología y Sostenibilidad de la Universidad Complutense de Madrid. Esta colaboración fue posible gracias al apoyo de Jesús Montero y el trabajo de Sandra Ballesteros y Macarena Lucas que formaron parte como expertos en el tema en las etapas de diseño. Este juego tiene como objetivo concienciar acerca del reciclaje y el concepto de la recogida de basura como “la cuarta R” del reciclaje. Su diseño parte del juego homónimo en físico⁸¹ que había aplicado la unidad en años anteriores, en el que los jugadores recorrían las plazas de la UCM conociendo su historia, flora y fauna mientras realizaban una recogida de basuras que se puntuaba y les permitiría obtener una serie de premios. Puesto que esta actividad se desarrolla al aire libre era posible su adaptación al formato geoposicionado y además proporciona una base con la que comparar los experimentos.

⁷⁹ http://www.educalandia.net/videojuegos_educativos/piraton_fuit_la_terre/

⁸⁰ <https://www.ucm.es/sostenibilidad/contacto>

⁸¹ <http://4rcampusjoven.com/senor-los-arbustos/>



Figura 21. Collage promocional para el juego serio "El Señor De Los Arbustos".

A través de diversas iteraciones de desarrollo se adaptó el juego físico original a través de un juego geoposicionado (**Figura 21**) en el que se hace uso de las funcionalidades disponibles en uAdventure para realizar una recogida virtual de basura. En esta recogida virtual los usuarios recorren la Plaza de Ramón y Cajal de la UCM en busca de diferentes objetos virtuales que descubrirán cuando estén cerca de ellos. Una vez localizados, los deben recoger y tirar en su contenedor adecuado utilizando códigos QR reales distribuidos en los diferentes contenedores. La narrativa juega un papel fundamental en la historia, siendo un elemento natural de los juegos geoposicionados como se identifica en (Arango-López et al., 2021). Su planteamiento es el siguiente: el villano *Basauron*, un monstruo de basura ha venido de otro mundo para acabar con nuestro mundo debido a la gran cantidad de basura que producimos y el mago *Bandalf* nos avisará de que la única solución es el reciclaje, ayudándonos a aprender a identificar los residuos a través de un compañero llamado *Bolsi*, y dándonos un plan para acabar con el villano a través de un combate final sobre reciclaje. A través de la narrativa aprovechamos también para conectar con conceptos de historia, flora y fauna fomentando así la exploración de diferentes puntos de interés (como monumentos o restos de la guerra civil) y conectando con diferentes personajes ilustres de la UCM (Santiago Ramón y Cajal, Severo Ochoa y Margarita Salas) que nos piden que recojamos materiales disponibles en la plaza que nos proporcionarán un bonus en la batalla final.

Para aplicar este juego se diseñaron distintos tipos de experimentos de validación: pilotos iniciales para probar el juego, experimentos para probar su efectividad de forma aislada y otros para compararlo con el método tradicional. Como parte de la metodología de los experimentos se prepararon dos esquemas diferentes que compartían dos cuestionarios (para antes y después) con los que poder validar la efectividad y comparar los resultados de los juegos. Para la creación de estos cuestionarios se utilizaron datos extraídos de la campaña de reciclaje de la Comunidad de Madrid⁸², formularios utilizados en otros experimentos del tema (para lo cual se realizó una pequeña investigación) (do Valle et al., 2004; Ramayah et al., 2012; Sidique et al., 2010; Tiew et al., 2013) y un formulario basado en el estándar ISO-9241/10 para evaluar la calidad de la experiencia (Prümper, 1993). En los casos en los que se utilizara el juego original, se realizó una adaptación del juego que garantizara su replicabilidad, materializando los residuos virtuales en físico a través de códigos que los usuarios debían encontrar y diseñando un pequeño libreto para orientar en la experiencia. Además, los experimentos con el juego geoposicionado también recolectarían analíticas de aprendizaje con el nuevo modelo de analíticas consolidado en nuestro segundo objetivo.

Las primeras pruebas piloto con el juego se centraron en la plataforma Android y nos ayudaron a pulir las mecánicas y la jugabilidad gracias a las confusiones y errores que se detectaron. En total se realizaron tres pruebas piloto hasta que el juego fue lo suficientemente estable, en las cuales tuvimos una acogida muy positiva por parte de los jugadores, aunque algunos no hubieran sido capaces de finalizar el juego. De cara a la herramienta las pruebas piloto permitieron consolidar el funcionamiento de uAdventure en dispositivos móviles, implementando soluciones a funcionalidades problemáticas como la precisión del GPS, la suspensión automática de los dispositivos, las diferentes orientaciones de las cámaras, o las limitaciones de espacio y de rendimiento.

Tras las pruebas piloto se programaron múltiples experimentos con el juego en plataformas Android. Sin embargo, sólo pudo realizarse un experimento (**Figura 22**) que utilizó tanto el juego físico como el juego geoposicionado utilizando la división los jugadores entre Android e iOS para definir los grupos (pues era la división más eficiente al aprovechar todos los usuarios a pesar de posibles sesgos). Sin embargo, debido a

⁸² https://www.ucm.es/data/cont/docs/3-2020-01-29-guia_adultos_residuos.pdf (Depósito legal: M-31295-2017)



Figura 22. Realización del primer experimento con usuarios con el juego serio El Señor De Los Arbustos.

retrasos el experimento no pudo realizarse adecuadamente, aunque algún jugador lograra completar el juego geoposicionado. Un segundo experimento tuvo que ser cancelado debido a las lluvias. Los demás experimentos programados fueron cancelados debido al inicio de la pandemia por el COVID-19, aunque llegó a realizarse un piloto que probó la aplicabilidad del juego de forma online sin utilizar el geoposicionamiento.

En este piloto el objetivo era enseñar a crear juegos geoposicionados utilizando uAdventure como apoyo y jugar a El Señor De Los Arbustos como parte del proceso de formación. Dadas las condiciones del confinamiento, era imposible probar los juegos en su entorno real y por ello, se diseñó un nuevo modo de juego para juegos geoposicionados con objetivos de depuración y demostración. Este nuevo modo está disponible para el uso dentro de la herramienta, pero también puede utilizarse en los juegos previa configuración o incluso puede activarse o desactivarse dentro del juego. En este modo se utilizan las interacciones realizadas sobre el mapa (con puntero o táctiles) para reubicar la posición del jugador. Al ser un estilo de interacción natural fue bien recogido por los participantes del piloto y fue muy útil para depurar los juegos desarrollados durante las últimas aplicaciones de uAdventure como herramienta docente (sección 4.5.1.3).

Debido a la situación que se desarrolló por la pandemia, fue muy complicado organizar experimentos pues, aunque fueran al aire libre seguían poniendo en riesgo la salud de los participantes. En su lugar, se aprovechó para mejorar el soporte para juegos en iOS con el objetivo de la posterior publicación del juego en la App Store. Para ello fue especialmente útil el modo de juego de depuración/demostración permitiendo a los revisores de la App Store probar al juego de forma remota durante el proceso de validación. Además, se actualizó el juego tras la integración de uAdventure con SIMVA para las futuras pruebas.

En la etapa final de la tesis consiguieron programarse dos experimentos con alumnos de bachillerato, aunque sólo uno se llevó a cabo con la participación de 22 alumnos debido una vez más a las lluvias. Aunque los datos recogidos fueron limitados, este experimento permitió asentar el uso de uAdventure en dispositivos móviles, especialmente en iOS donde se detectaron errores críticos en el uso de la memoria que afectaban a dicha plataforma; y probar la integración con SIMVA a pequeña escala, donde también se detectaron diversos errores debidos a la concurrencia.



Figura 23. Los diferentes minijuegos de la mansión paranormal. Las imágenes numeradas muestran: 1) la interacción dentro del formato aventura; 2) una pantalla de opciones; 3) minijuego formas y colores; 4) minijuego hacking ético; 5) minijuego anillas locas; 6) minijuego electricista; 7) minijuego laberinto de números y 8) retroalimentación al usuario en forma de estrellas tras completar los minijuegos.

4.5.2.4. Pruebas con juegos que resultaron del uso de uAdventure por alumnos

Como avanzamos al finalizar la sección 4.5.1, nuestras pruebas de aplicabilidad de la herramienta como apoyo a la enseñanza culminaron dando lugar a diversos juegos que se han usado como parte de las pruebas con la herramienta. Estos juegos son La Mansión Paranormal y Guerra Civil UCM.

La Mansión Paranormal (**Figura 23**) es un juego que aprovecha las funcionalidades de creación de minijuegos específicos para incluir distintos tipos de puzzles que promocionan competencias relacionadas con el pensamiento computacional. Estos minijuegos se apoyan en la narrativa y utilizan el formato de puzzle para desarrollar capacidades de algoritmia, descomposición, abstracción y reconocimiento de patrones. Los cinco minijuegos desarrollados aprovechan las capacidades de Unity construyendo minijuegos con distintas mecánicas: las anillas locas, el electricista, el hacking ético, formas y colores, y el laberinto de números. Estos minijuegos cuentan con varios niveles (que se configuran a través de parámetros en uAdventure) e incluyen analíticas de aprendizaje personalizadas para identificar las diferentes interacciones de los jugadores en los minijuegos y su progreso a través de completables. Además, todos los juegos cuentan con una mecánica de pistas para ayudar en su resolución si el usuario tiene dificultades, así como una puntuación en estrellas que da una idea de lo bien que se ha resuelto el puzzle y trata de fomentar la competitividad de los jugadores.

El experimento llevado a cabo utilizando este juego tenía como objetivo evaluar la perspectiva que tenían los profesores para su utilización en sus clases. Para ello, el diseño experimental contaba con la realización de cuestionarios pre-post que ayudarían a hacer un perfil de los profesores y permitiría que estos opinaran acerca de su utilidad y futura aplicabilidad tras jugarlo, así como la recopilación de analíticas de aprendizaje. Por ello, desde la perspectiva de esta tesis, esta recopilación de analíticas pudo aprovecharse para validar la aplicabilidad de nuestro modelo de analíticas en minijuegos y fue publicado como parte del artículo Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021b). **Extending Narrative Serious Games Using Ad-Hoc Mini-games**. In W. Zhou & M. Yi (Eds.), *Advances in Web-Based Learning – ICWL 2021: 20th International Conference, ICWL 2021, Macau, China, November 13–14, 2021, Proceedings* (pp. 63–74). Como caso de uso de estos datos, pudimos analizar de forma comparativa la complejidad de los juegos y sus diferentes dificultades en base a la cantidad de interacciones realizadas durante su uso (**Figura 24**).

Pasos medios en cada mini-juego según su dificultad



Figura 24. El gráfico muestra de forma relativa la cantidad de pasos utilizados en cada nivel de dificultad de los diferentes mini-juegos. El círculo interior representa el mini-juego concreto mientras que el círculo exterior identifica del 1 al 3 el nivel de dificultad siendo 3 el mayor.

Gracias a este análisis los desarrolladores pudieron comprobar cómo, en algunos casos, la cantidad de interacción no escalaba de modo acorde a la dificultad.

En cuanto al otro juego, Guerra Civil UCM, se trata de un juego geoposicionado en formato de búsqueda de tesoros que trata el tema de la guerra civil en la zona de la UCM. En el juego deberemos encontrar la ubicación real de cinco fotos que fueron tomadas tras la guerra civil. Durante el proceso de investigación, los jugadores hablarán con personajes de ambos bandos, aprendiendo sobre el contexto histórico y el avance y recibiendo pistas que les ayudarán a encontrar los lugares de las fotos. Aunque en la primera versión del juego se planteó la historia, este juego fue revisado y mejorado como parte del trabajo de esta tesis tanto en su contenido como en el control de los jugadores durante la experiencia, puesto que se realizó una prueba piloto que mostró diversas carencias. La revisión del juego mejoró la historia de soporte, incluyendo no-linealidad en entornos más controlados, y perfeccionando el acabado visual y el uso de los elementos coleccionables. Las pruebas piloto con cuatro usuarios tras la revisión probaron que los jugadores podían completar el juego de forma completamente autónoma.

Tras los diferentes pilotos se diseñó un experimento (**Figura 25**) con participantes de perfil universitario proveniente del Grado de Desarrollo de Videojuegos. El objetivo era realizar una prueba a mayor escala con el juego a la vez que se probaba la aplicabilidad de la reciente integración con SIMVA y se usaba como caso ilustrativo para el inicio de los próximos cursos que utilizarían uAdventure.



Figura 25. Estudiantes juegan al juego geoposicionado Guerra Civil UCM en sus móviles en el campus de la Universidad Complutense de Madrid.

4.5.3. Otros experimentos como parte de TFGs

En esta sección exponemos dos últimos casos prácticos en los que se utilizó uAdventure o sus componentes para su realización.

Como parte del trabajo de fin de grado titulado “*Desarrollo y edición de juegos serios con Unity y E-Adventure*” realizado por David Martín-Maldonado Jiménez y Javier Sandoval Ferrandis y dirigido por el Prof. Dr. Manuel Freire Morán se utilizaron las capacidades de extensibilidad de uAdventure para la creación de una nueva extensión orientada a la creación de minijuegos (Martín-Maldonado Jiménez & Sandoval Ferrandis, 2017). Para ello, los alumnos utilizaron como referencia la implementación de la extensión orientada a códigos QR. El resultado de este TFG nos permitió comprender las dificultades que podría experimentar un desarrollador a la hora de implementar una extensión en uAdventure en el contexto de los minijuegos, dando pie al desarrollo de la extensión para minijuegos que se desarrolló posteriormente.

Como parte del trabajo de fin de grado titulado “*Juegos Serios para Promover el Pensamiento Computacional y la Programación*” realizado por Dany Faouaz, Santillana Arturo García Cárdenas y Álvaro Poyatos Morate y dirigido por el Prof. Dr. Baltasar

Fernández Manjón y el Dr. Antonio Calvo Morata se utilizó la extensión diseñada para integrar uAdventure con SIMVA y el *tracker* para analíticas de aprendizaje (Faouaz Santillana et al., 2021). El TFG se centra en el diseño y desarrollo de un juego serio con Unity que utiliza un lenguaje similar a Scratch para resolver puzles que permiten desarrollar el pensamiento computacional. El uso de las herramientas tuvo como objetivo probar la aplicabilidad de las herramientas de forma externa a uAdventure y permitió a los estudiantes desarrollar analíticas de aprendizaje específicas para su juego utilizando la interfaz de alto nivel que proporciona el *tracker*. Los experimentos realizados durante este TFG aplicaron el juego tanto con profesores como con alumnos de ESO y Bachillerato, aunque aún no se ha terminado el análisis sobre las analíticas recopiladas utilizando el *tracker* y la integración con SIMVA.

Capítulo 5. Conclusiones y Trabajo futuro

En este capítulo se presentan las conclusiones y las principales contribuciones de esta tesis doctoral, así como posibles futuras líneas de investigación que han surgido a lo largo del trabajo.

Es importante resaltar que el trabajo desarrollado tiene un importante componente de ingeniería. Para resolver el objetivo científico propuesto, simplificar y mejorar la creación de juegos, se ha propuesto y desarrollado una arquitectura genérica a través de un prototipo que demuestra su factibilidad. Generando un producto software en forma de herramienta de código abierto a utilizar por otros desarrolladores y por usuarios finales que forman parte de la comunidad educativa. La herramienta obtenida es uAdventure. Además, también se proporciona un *tracker* xAPI que permite aplicar algunas de las características presentadas a cualquier juego generado en Unity.

5.1. Conclusiones y principales aportaciones

Esta tesis plantea como principal objetivo simplificar la creación y mejorar la aplicabilidad de los juegos serios, centrándonos especialmente en el género de los juegos geoposicionados con soporte narrativo. Con este propósito se ha propuesto un modelo con una arquitectura extensible que se ha plasmado en el entorno de creación de juegos uAdventure. uAdventure supone una revisión completa de una herramienta anterior denominada e-Adventure, de modo que aprovecha la experiencia previa y solventa los problemas que propiciaron su obsolescencia. Las aportaciones y conclusiones de este trabajo son:

- El diseño y modelo de alto nivel de una herramienta de autoría (ver 4.1) que permite la creación de juegos serios geoposicionados de una manera simple para usuarios no expertos. Este diseño se construye a partir del ya usado en eAdventure con mejoras en las principales características de interfaz y usabilidad, y con la adición de una extensión que permite el uso de mecánicas de geoposicionamiento. Esto se ha llevado a cabo mediante el desarrollo del entorno de autoría uAdventure, utilizando de base el motor de videojuegos Unity, delegando a través de la automatización de procesos requisitos técnicos y características como la exportación a diferentes plataformas (Windows, Mac, Linux, Android, iOS y WebGL). Este diseño y modelo de alto nivel son especialmente relevantes para

usuarios del entorno educativo como, por ejemplo, los docentes que pasan a ser capaces de crear o modificar juegos educativos sin necesidad de conocimientos técnicos (e.g., programación). El entorno de autoría se ha utilizado para desarrollar diversos juegos educativos que ponen en práctica sus capacidades (ver 4.5.2), demostrando, en nuestras pruebas con usuarios, que la herramienta puede ser utilizada por no-expertos de forma autónoma para desarrollar juegos serios geoposicionados con soporte narrativo (ver 4.5.1.1, 4.5.1.2 y 4.5.1.3), y para asistir como herramienta docente en el proceso de enseñanza de los juegos serios (ver 4.5.1.2 y 4.5.1.3). El diseño de la herramienta y los resultados de su aplicación con usuarios han sido publicados en la revista de alto impacto *Educational Technology and Society*.

- Un modelo abstracto y genérico para la creación de juegos geoposicionados (ver 4.2.1) que reduce la rigidez del estilo de juego e incluye soporte narrativo. Con este modelo se fomenta la creatividad de los desarrolladores y la inmersión de los juegos resultantes, pudiendo catalogar los juegos resultantes en el nuevo género de “aventura geoposicionada”. Además, este género permite la aplicación de una serie de buenas prácticas para este tipo de juegos (como la creación de tutoriales o el uso de un guía). Este modelo ha demostrado su eficacia tanto en nuestros desarrollos (ver 4.5.2.1, 4.5.2.3 y 4.5.2.4) como en los desarrollos de nuestros usuarios, que no han encontrado dificultades a la hora de implementar sus ideas de juego con la herramienta (ver 4.5.1). El modelo y su primer ejemplo de aplicabilidad ha sido publicado en la conferencia *Technological Ecosystems for Enhancing Multiculturality*.
- El soporte de analíticas de aprendizaje y la incorporación intrínseca de un modelo de analíticas para juegos serios y juegos geoposicionados (ver 4.1.4 y 4.2.2) que permiten monitorizar las actividades con juegos serios sin incrementar de forma significativa la complejidad del desarrollo. El modelo de analíticas para juegos geoposicionados ha sido publicado en la conferencia *IEEE Global Engineering Education Conference*. Esta solución reduce la complejidad de su adopción permitiendo utilizar las analíticas desde etapas tempranas del desarrollo para probar la efectividad del juego, detectar errores o mejorar su diseño; y favorece la consolidación de los modelos de analíticas de los juegos haciéndolos más útiles y confiables para los usuarios finales del juego serio. La integración de analíticas de

aprendizaje permite monitorizar las actividades con juegos serios geoposicionados (ver ejemplos en 4.2.2, y experimentos en 4.5.2), y que los usuarios que utilizan uAdventure como herramienta docente puedan comprobar la efectividad de sus juegos y mejorarlos en futuras iteraciones (ver 4.5.2.4). El modelo propuesto permite, beneficiarse del uso de estándares educativos como xAPI sin tener conocimientos de éstos y acceder a todo un ecosistema no-propietario para el análisis.

- El diseño de una arquitectura extensible sobre un motor de videojuegos multipropósito (ver 4.1) que facilita tanto su mantenimiento como la incorporación de nuevas funcionalidades, y convierte a uAdventure en una herramienta multipropósito. uAdventure simplifica la colaboración interdisciplinar, puesto la herramienta es accesible para no-expertos y sus capacidades para extenderse, especialmente a través de la extensión de minijuegos (ver 4.4), están orientadas a expertos en el desarrollo de videojuegos. La arquitectura ha demostrado su efectividad, por un lado, al haber sido posible incorporar múltiples extensiones (juegos geoposicionados, códigos QR, analíticas de aprendizaje y minijuegos). Además, en nuestras pruebas con usuarios hemos podido comprobar que los usuarios han podido desarrollar minijuegos de propósito específico utilizando nuestra extensión simplificada (ver 4.5.1.3 y 4.5.2.4) e incluso la arquitectura ha sido usada en un TFG que desarrolló su propia extensión para uAdventure (ver 4.5.3). El modelo de minijuegos ha sido publicado en la *International Conference on Web-Based Learning*.
- Los mecanismos de integración de los juegos serios geoposicionados con analíticas de aprendizaje en procesos educativos para permitir el empaquetamiento, la distribución, la aplicación y la validación de los juegos; y el análisis y la evaluación de las actividades que utilizan estos juegos. El soporte combinado de estándares educativos facilita la aplicación de juegos serios con analíticas de aprendizaje en entornos educativos. Estos estándares se han implementado a través de múltiples componentes (el *tracker* y una extensión), que constituyen software reutilizable con tres modos de funcionamiento (ver 4.3): un modo LRS que recolecta los datos en un contenedor preconfigurado (que es capaz de funcionar en un LMS), un modo LMS que permite lanzar los juegos serios y su configuración dinámica utilizando el estándar CMI5 (que puede utilizarse en

LMS que lo soporten o a través de herramientas mediadoras), y un modo SIMVA, que permite el uso directo con esta plataforma para gestión de experimentos con analíticas de aprendizaje. Hemos podido probar la efectividad del *tracker* en nuestros experimentos, así como la capacidad de simplificar la integración en el modo SIMVA por parte de nuestros usuarios finales. Por otro lado, ambas componentes han sido útiles para acelerar el desarrollo de minijuegos (ver 4.5.2.4) y han permitido el desarrollo como TFG de un juego independiente a la herramienta de autoría (ver 4.5.3). Esta contribución y sus resultados han sido publicados en el congreso internacional *Games and Learning Alliance*, junto con un estudio de las diferentes herramientas alternativas disponibles (*trackers* y LRS) recibiendo la nominación a *Best Paper Award*.

- La metodología de enseñanza utilizando uAdventure como herramienta docente. Esta metodología permite formar a estudiantes en el desarrollo de los juegos serios y en el uso de analíticas de aprendizaje dentro del límite de tiempo de un semestre académico. Esto es posible gracias al alto nivel de abstracción de la herramienta y la implementación intrínseca de las analíticas, que reducen los requisitos técnicos permitiendo enfocar el desarrollo en los aspectos educativos relevantes para el ciclo de vida de los juegos serios. Esta metodología ha sido probada (ver 4.5.1) en diversos workshop y asignaturas universitarias dando como resultado múltiples juegos (de aventura y geoposicionados) que tienen una duración de entre 10 y 15 minutos; y, cuentan con un diseño de analíticas de aprendizaje para validar sus objetivos educativos que los estudiantes han podido probar utilizando la integración en modo SIMVA. Esta contribución se publicó en el congreso *Frontiers In Education*.
- Los juegos y experimentos realizados durante la tesis a través de uAdventure, nos han permitido aprender una serie de lecciones valiosas más allá de probar la validez y eficacia de las contribuciones. En primer lugar, es muy importante realizar un buen diseño del juego para: asegurar que los jugadores entienden el juego y sus objetivos, para lo cual son muy útiles los tutoriales y asistentes; realizar una exploración segura y controlada, evitando demasiada libertad que propicia perder a los jugadores, pero ofreciendo cierta no-linealidad para que los jugadores puedan aplicar sus conocimientos; y evitar distancias largas o complementarlas con coleccionables o personajes secundarios. En segundo lugar,

es esencial la planificación para: conseguir las mejores condiciones climáticas, evitando lluvias o calor extremo, siendo especialmente positivos los meses de marzo por ser un buen término medio y de junio a septiembre poniendo atención al calor; conseguir que la actividad arranque rápidamente, proporcionando instrucciones detalladas de la instalación a los participantes previamente; y evitar los problemas asociados a la práctica “*bring your own device*” debido a dispositivos problemáticos, poniendo a disposición de los participantes baterías externas y dispositivos de reemplazo. Por último, es importante tener alternativas. Esto ha sido especialmente relevante durante la pandemia mundial del COVID-19, pues los experimentos no podían realizarse pese a ser actividades al aire libre, ni podían probarse los juegos desarrollados. Como alternativa, el nuevo modo de juego para juegos geoposicionados utilizando la pantalla táctil permite que se puedan jugar o probar los juegos de forma remota utilizando la pantalla táctil, e incluso es posible intercambiar entre el modo de juego en mitad de la partida para solucionar problemas derivados del geoposicionamiento. Como reflexión final, debido la cantidad de factores que pueden influir en el buen desarrollo del juego y de las actividades que consideramos que los juegos geoposicionados deben ser utilizados de forma esporádica, aunque no creemos que esto limite su potencial, pudiendo introducir y practicar contenidos en un contexto en el que conseguiremos motivar a los participantes.

La implementación de uAdventure ha sido publicada como código abierto en el repositorio de GitHub: <https://github.com/e-ucm/uAdventure>. Junto con la herramienta se han desarrollado un manual, una guía de inicio rápido y una serie de ejemplos con los que aprender a utilizar la herramienta y que han sido validados en nuestras pruebas; así como una serie de contenidos didácticos (como diapositivas o videotutoriales) que han sido utilizadas como parte de la metodología docente. El componente del *tracker* también se ha productificado para poder ser utilizado por desarrolladores de forma independiente, y se distribuye en el repositorio: <https://github.com/e-ucm/unity-tracker>.

Además, hemos desarrollado diversos juegos como parte de esta tesis, algunos de ellos en colaboración con alumnos que participaron en workshops para la creación de juegos serios utilizando uAdventure (ver 4.5.2). Estos juegos forman parte de un repositorio de ejemplos y pueden ser utilizados de forma libre por educadores. Entre ellos podemos destacar dos contribuciones en formato de juegos geoposicionados. “El Señor de los

Arbustos” nació como colaboración con la Unidad de Campus y Medio Ambiente de la UCM para ayudar en la tarea de concienciación del reciclaje y ha sido publicado como código abierto en el repositorio: <https://github.com/Victorma/arbustos-go>. Y “Guerra Civil UCM”, que nació como proyecto de alumnos utilizando uAdventure (dentro de nuestras pruebas como herramienta docente) con el objetivo de enseñar el contexto histórico del frente de la guerra civil en el campus de la Moncloa de la UCM; y se ha publicado como código abierto en el repositorio: <https://github.com/Victorma/juego-guerra>.

5.2. Trabajo futuro

Tras la finalización de la tesis, y aunque consideramos que se han logrado los objetivos planteados, creemos que aún queda trabajo por hacer para ayudar a la adopción de los juegos serios geoposicionados. Consideramos que existen al menos tres líneas de trabajo que pueden desarrollarse de forma paralela: la mejora de la autoría de juegos serios geoposicionados, la simplificación del uso de los juegos serios geoposicionados y la simplificación del proceso de publicación y difusión de los juegos.

Con respecto a la mejora de la autoría de juegos serios geoposicionados se pueden abordar los siguientes puntos:

- Añadir una nueva estructura al modelo de juegos geoposicionados para agrupar y encapsular contenidos. Actualmente, el modelo propuesto para la creación de juegos serios geoposicionados permite una completa libertad para dar forma a la estructura del juego. Pero esta flexibilidad aumenta la complejidad del desarrollo conforme se añaden más elementos y situaciones al juego. Consideramos que la inclusión de “etapas de juego” en el modelo permitiría: dividir de una forma más eficaz las diferentes partes del desarrollo; facilitar la configuración de entornos controlados de exploración; y habilitar una forma natural de guiar al jugador a través del desbloqueo y bloqueo de las diferentes partes del juego.
- Estudiar e integrar en el modelo actual otras tecnologías de geoposicionamiento para hacerlo más preciso y versátil. Estas tecnologías podrían complementar el modelo ya existente y/o formar parte del modelo de los juegos como ya hacen los códigos QR, adaptando el estilo de juego al nuevo tipo de geolocalización. Por ejemplo, podrían asociarse iBeacons Bluetooth a elementos geoposicionados para

detectar si nos encontramos en su área de influencia. Por otro lado, también podrían implementarse técnicas para mejorar la eficacia del geoposicionamiento sin incrementar la complejidad tecnológica del juego. Por ejemplo, podría mejorarse la experiencia del usuario a través de la modificación dinámica de las áreas de influencia (como se expone en (Benford et al., 2005)).

- Integrar el uso de la tecnología de realidad aumentada en el diseño de entorno de autoría propuesto y en la propia herramienta de uAdventure. La realidad aumentada es una característica transversal de los juegos inmersivos (y sus herramientas de autoría (Medlock-walton, 2013; Mota Macías & Ruiz-Rube, 2017; Vidal et al., 2019)) especialmente relevante para juegos geoposicionados. Con ella se permite una interacción más concreta y detallada con el entorno que la que provee el mapa aumentado. Sin embargo, las librerías disponibles para su uso se enfocan en desarrolladores. Con el nuevo modelo se facilitaría el uso de la realidad aumentada, pudiendo incluir funcionalidades como el desarrollo de escenarios en formato de realidad aumentada o el posicionamiento local basado en marcadores de realidad aumentada basados en imágenes reales que eliminarían la necesidad de utilizar recursos físicos externos (e.g., códigos QR impresos).

Con respecto a la simplificación del uso de juegos serios geoposicionados, como trabajo futuro, aún es necesario:

- Estudiar y mejorar la experiencia de usuario y la usabilidad de los juegos serios geoposicionados en entornos web. Hemos presentado en esta tesis diversas formas de exportar y empaquetar los juegos serios para ser usados en entornos web (como por ejemplo utilizando estándares educativos como IMS CP o CMI5). No obstante, al habernos enfocado principalmente en las aplicaciones nativas, este empaquetamiento ha sido utilizado solamente de forma experimental con juegos de aventura. Es necesario estudiar cómo se comportan los juegos geoposicionados en este entorno para mejorar la experiencia del usuario en estos entornos.
- Diseñar un sistema orientado a ayudar en el proceso de despliegue/instalación de los juegos serios geoposicionados en plataformas nativas. Aunque el uso de plataformas nativas parece ser el más apropiado para garantizar la mejor experiencia del jugador, los estándares que hemos identificado sólo simplifican el proceso de lanzamiento de los juegos y el paso de parámetros de configuración.

Esto implica que es necesario que los juegos estén instalados a priori en la plataforma a utilizar. Actualmente su instalación sólo puede hacerse de forma manual, lo que dificulta la realización de experimentos y tiene el potencial de causar errores. El sistema de instalación ayudaría al usuario a desplegar el juego en su dispositivo proporcionándole las instrucciones necesarias y comprobaría que el juego se haya desplegado correctamente antes de su posterior uso.

Con respecto a la simplificación de la publicación y la difusión de los juegos se pueden abordar los siguientes puntos:

- Simplificar el proceso de publicación de los juegos nativos en mercados de dispositivos móviles (Google Play y App Store). Puesto que el proceso de publicación es especialmente complejo (sobre todo en App Store) y se orienta a desarrolladores, podríamos simplificarlo publicando una aplicación intermedia (o *launcher*) que se encargaría de obtener y ejecutar los juegos. De hecho, esta idea ya fue probada en el TFM de Iván José Pérez Colado (I. J. Pérez-Colado, 2016) a través de la creación de una aplicación que se encargaba de leer el modelo del juego y ejecutarlo utilizando el intérprete. Publicando este *launcher* permitiríamos el acceso a plataformas nativas sin necesidad de pasar el proceso de publicación.
- Diseñar un repositorio para la difusión e intercambio de juegos y minijuegos. Este repositorio permitiría a los educadores explorar soluciones existentes de juegos y minijuegos compatibles con la plataforma, pudiendo incorporarlas en sus propios juegos y/o adaptarlas a sus necesidades de forma sencilla. Además, este repositorio también podría usarse para almacenar y proporcionar los datos de los juegos a los *launchers* anteriormente descritos.

Para finalizar, una de las líneas de trabajo directas en esta tesis doctoral y que han quedado pendientes debido fundamentalmente a las restricciones y retrasos impuestas por la pandemia, es la realización de experimentos masivos con los juegos desarrollados: El Señor de los Arbustos GO, La Mansión Paranormal, La Entrevista y Guerra Civil UCM. Se han hecho pruebas iniciales para demostrar la factibilidad y el enfoque, pero con experimentos de mayor volumen con usuarios reales y habida cuenta de toda la adquisición de datos de analíticas incorporadas se podrían obtener resultados de primer nivel internacional. Por otra parte, también se podría estudiar la realización de nuevos

análisis de los datos generados durante las diferentes pruebas piloto y los experimentos iniciales para comprobar algunos supuestos que no se analizaron en esos casos.

Capítulo 6. Artículos presentados

En este capítulo se incluyen los artículos aportados como parte fundamental de esta tesis doctoral.

6.1. Simplifying the creation of adventure serious games with educational-oriented features

6.1.1. Cita completa

Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). **Simplifying the creation of adventure serious games with educational-oriented features**. *Educational Technology and Society*, 22(3), 32–46.

6.1.2. Resumen original de la publicación

Developer-friendly professional authoring tools have greatly simplified entertainment videogame development. However, this simplification had a limited impact on serious games, which require the active collaboration of developers with educators and other stakeholders. To address this issue, we present uAdventure, an easy-to-use game development environment for narrative "point-and-click" graphic adventure games. uAdventure started as a re-implementation of the previously validated eAdventure authoring tool, now built on top of the Unity game platform. The idea is to enjoy the advantages of the Unity professional environment while reducing its complexity to allow non-developers to author serious games. uAdventure is designed to simplify the creation of serious games that include educational-oriented capabilities such as learning analytics without requiring programming knowledge and has been formatively tested with different types of users and in different settings. Initial testing was carried out with a group of heterogeneous users with different computer usage profiles in a vocational environment. In the second round of testing, it was used by students of two different university courses to develop serious games that include educational features such as location-based mechanics and learning analytics. The results of these formative evaluations show that uAdventure can be used as a serious game teaching tool and that it simplifies the creation of serious games with educational features by non-expert authors.

Simplifying the Creation of Adventure Serious Games with Educational-Oriented Features

Pérez-Colado Víctor Manuel^{*}, Pérez-Colado Iván José, Freire-Morán Manuel, Martínez-Ortiz Iván and Fernández-Manjón Baltasar

Software Engineering and Artificial Intelligence Department, Faculty of Informatics, Complutense University of Madrid, Spain // victormp@ucm.es // ivanjper@ucm.es // manuel.freire@fdi.ucm.es // imartinez@fdi.ucm.es // balta@fdi.ucm.es

^{*} Corresponding author

ABSTRACT: Developer-friendly professional authoring tools have greatly simplified entertainment videogame development. However, this simplification had a limited impact on serious games, which require the active collaboration of developers with educators and other stakeholders. To address this issue, we present uAdventure, an easy-to-use game development environment for narrative “point-and-click” graphic adventure games. uAdventure started as a re-implementation of the previously validated eAdventure authoring tool, now built on top of the Unity game platform. The idea is to enjoy the advantages of the Unity professional environment while reducing its complexity to allow non-developers to author serious games. uAdventure is designed to simplify the creation of serious games that include educational-oriented capabilities such as learning analytics without requiring programming knowledge and has been formatively tested with different types of users and in different settings. Initial testing was carried out with a group of heterogeneous users with different computer usage profiles in a vocational environment. In the second round of testing, it was used by students of two different university courses to develop serious games that include educational features such as location-based mechanics and learning analytics. The results of these formative evaluations show that uAdventure can be used as a serious game teaching tool and that it simplifies the creation of serious games with educational features by non-expert authors.

Keywords: Serious game development, Authoring tool, Narrative games, Learning analytics, Location-based games

1. Introduction

Serious Games (SGs) are videogames where the main purpose is not entertainment, with education as one of the most notable examples. There are multiple benefits for students when it comes to learning through SGs and gamified experiences (De Freitas, 2018), among which an increased engagement is especially important. However, although multiple initiatives have demonstrated a positive impact on both educational institutions and companies, SGs are not yet widely used in education (Almeida & Simoes, 2019). The difficulty of formally proving their benefits and return of investment; and to the need to involve other stakeholders (e.g., educators, domain experts) with key responsibilities in their development process are two of the main reasons hindering more widespread adoption of SGs in educational institutions.

Nowadays, there is wide range of tools to create serious games from professional platforms to research tools. At the professional level, we can find all-purpose but complex game platforms such as Unity, or more narrowly-scoped and consequently simpler platforms such as Game Maker Studio. Certain commercial game authoring tools are even accessible to non-programmers, including GameSalad, Genie, or The Training Arcade. Besides, there are also tools created as a result of research projects that were specifically designed as authoring tools for SGs. Some of them are StoryTec (Göbel, Salvatore, Konrad, & Mehm, 2008) that is a story-telling authoring tool or Emergo (Nadolski et al., 2008) that is a scenario-based authoring tool.

The use of genre-specific game engines such as Adventure Game Studio can greatly simplify game development and thus SG development (del Blanco et al., 2012; Marchiori et al., 2012), making their creation accessible for non-programmers and simplifying the participation of non-technical SG stakeholders, among which teachers are particularly important. This paper describes uAdventure (uA), an SG authoring tool implemented on top of the Unity game environment, designed to simplify the creation of serious games that include educational-oriented capabilities such as learning analytics without requiring extensive programming knowledge.

To simplify the SG development process for non-experts, uA focuses on facilitating a high-level authoring metaphor, instead of overwhelming authors with the highly flexible but complex functionalities offered by all-purpose platforms. We achieve this by specializing in the adventure game genre. Specific game genres support different types of learning (Aust, Nitsche, & Pelka, 2014). For example, game-like simulators allow authentic learning through recreating real-life situations (Center for Technology Implementation in Education, 2014), but the specificity and complexity of their mechanics can be expensive to develop and will usually require extensive ad-hoc programming. We chose to focus uA on authoring only “point-and-click” adventure SGs because the genre provides a good balance between power and complexity for creating educational games, which combine a strong narrative with puzzles and problem solving, promoting learning and reflection (Dickey, 2006). After testing uA with different users and in different settings, we consider that uA can be used as a serious game authoring and teaching tool. Additionally, uA also allows non-expert authors to add educational-oriented features, such as analytics, to the SGs that they develop.

The rest of the paper is structured as follows: we first introduce narrative adventure games and eAdventure (eA) the predecessor of uA and its limitations. Then, we introduce uA features that both address eA’s limitations and further simplify adventure SG development for non-experts. Finally, we detail the experiments used to validate and guide the development of uA, discuss the results, and present our conclusions and future work.

2. Narrative adventure games: From eAdventure to uAdventure

Adventure games are story-driven (Rollings & Ernest, 2003) and inherit their main elements from the narrative metaphor (Marchiori et al., 2012). The main elements of adventure games are (Dickey, 2006): *characters*, which are the entities that populate the story and perform the different actions; *objects* that the player must collect or interact with; *scenes*, where the action occurs; and finally, the use of *narrative* and *actions* to encompass narration, conversations, and interactions of characters with their environment. In adventure games, the player advances through the story by exploration and solving logic puzzles, normally involving characters or items (Amory, 2001). These features are especially useful in educational games, as players need to reflect on their knowledge to achieve their in-game goals.

Game stories can be *linear* or *non-linear*. Linear stories follow a strict sequence, while non-linear stories can result in different states (endings, scores, etc.) depending on the players’ choices, allowing them to access a whole “sequence of possibilities considering changing world states” (Spierling, 2009). Non-linearity is ideal for SGs, as players feel that their actions have meaning and understand that they must investigate and use deduction to reach the desired ending. Non-linearity is also very useful from an analytics perspective, as game states, player responses to in-game questions, and past problem-solving actions can be analyzed and used for assessment.

The eAdventure (eA) SG platform (Figure 1) is a previous authoring tool for the creation of “point-and-click” games (Torrente, del Blanco, Marchiori, Moreno-Ger, & Fernandez-Manjon, 2010). The key goals of eA were:

- To provide a multiplatform java-based adventure game authoring tool for non-programmers, and even accessible to users without high levels of computer literacy.
- To allow authors to package games as educational learning objects using different educational standards such as IMS Content Packaging usable in Learning Management Systems (LMS) such as Moodle.
- To allow authors to create games as an assessment tool. eA’s authoring tool included a simple assessment tool based on flags and variables and a reporting tool based on the (limited) capabilities provided by SCORM standard (del Blanco, Marchiori, Torrente, Martínez-Ortiz, & Fernández-Manjón, 2013).

At the time eA was created, game engines and authoring tool licenses were expensive or did not provide then-current features. This situation made a Java-based free and open-source project such as eA a good fit for educators. At the time, applets allowed easy packaging and distribution of SGs using any web-based Learning Management System such as Moodle. The java technology multiplatform approach was initially adequate for the diversity in operating systems and devices, but the fast change in mobile technology made eA maintenance a challenge. Thus, relying on commercial game engines, such as Unity, as a middleware for an SG development tool, ensures cross-platform compatibility and ongoing support as the platforms evolve, and can greatly reduce development and maintenance costs. However, using Unity alone requires significant programming skills, making it inaccessible to non-experts.

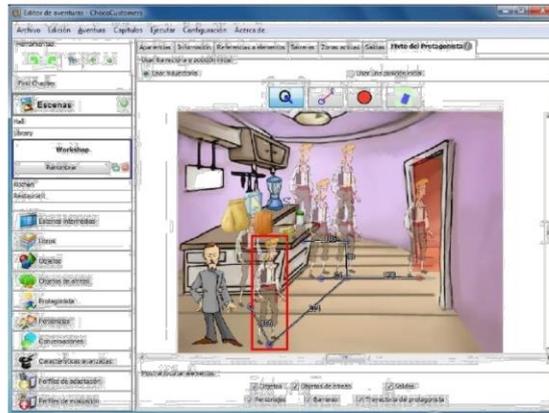


Figure 1. The scene editor in the eAdventure serious game authoring tool

To increase the added value of SGs, and to support their use as assessment tools, we consider that evaluation should be embedded as part of the game in a non-intrusive manner, and if possible, based on the evaluation of players' in-game actions, as opposed to requiring players to answer intrusive post-tests. Assessment capabilities were already included in eA, based on author-specified flags and variables that represented milestones or important events. This kind of assessment was fixed at authoring time, and affected by the limitations of the ADL SCORM data model and of the LMSs that hosted the games.

Since eA was developed, the use of Learning Analytics (LA) has become much more widespread. LA information is usually collected via *traces*, where each trace represents an event produced during the interaction with the educational tool. ADL eXperience API (xAPI) is a new LA specification designed to address the limitations of previous standards such as SCORM, providing an open common structure and an extensible data model that can be adapted by different communities of practice; for example, there are application profiles for SGs, videos, and other media. The inclusion of xAPI can provide more flexible and advanced analytics for SGs, which, for example, need not be completely fixed at game creation and can instead be improved based on experience.

Moreover, technology-consuming behavior has changed, transitioning from the use of single to multiple devices, and from the preference of using PCs and laptops to mobile devices such as tablets and smartphones. Mobile devices, in particular, are preferred by children (Lauricella, Cingel, Blackwell, Wartella, & Conway, 2014) – and also include new sensors and capabilities that can be of value to SGs. In particular, location-based (LB) games are a popular trend in both commercial (Laato, Rauti, Pietarinen, & Laine, 2019) and educational (Xanthopoulos & Xinogalos, 2018) sectors. In terms of game mechanics, location-based games use real-world locations as game elements both outdoors (using the GPS) and indoors (using QRs codes or beacons). From an educational perspective, location-based SGs can offer new learning experiences such as gymkhanas or historical tours that require interacting with the real world to progress in the game.

3. uAdventure

uAdventure (uA) is a SG framework built on top of Unity that reuses our previous experience with eA to simplify the creation of narrative “point-and-click” SGs by non-experts. We decided to build uA on top of Unity to solve eA's major technical problems: uA will always support the latest technologies and platforms, as long as Unity keeps being updated to support them. Reusing most of Unity's subsystems allows us to focus the development on educationally-oriented features that make uA unique, making it more maintainable and less likely to become obsolete (Perez-Colado, Perez-Colado, Martínez-Ortiz, Freire-Moran, & Fernández-Manjon, 2017a). Still, uA maintains the

A new element inspector in uA simplifies element modification: instead of having to navigate to the underlying elements themselves to view or change their properties, the inspector allows authors to directly access them in the scene editor without navigating away from the scene itself. The design of this inspector (Figure 3) is based on the Unity inspector, making it more affordable to users that are already familiar with Unity. We expect it to be useful in three distinct scenarios: where development teams include both educational experts and Unity experts, the latter of which will often have to switch between the different perspectives; when Unity users want to use uA as a prototyping tool; and when uA users want to step into Unity to achieve more complex results. In any case, authors can switch between the uA and Unity interfaces at any time in just a couple of clicks.

To simplify SG content creation by non-experts, uA provides its editors, replacing Unity's default interface with SG-relevant views such as a model editor, context variables and flags, a gameplay window, and a project file explorer. uA also handles some of Unity's default functionalities. For example, when the "play" button is pressed to test the game, the corresponding uA scene is loaded transparently, without forcing users to deal with Unity scenes at all. Conversely, when the game stops, the uA window is reopened. Saving and loading projects work similarly. Also, uA uses the asset importing pipeline to automatically handle asset configurations when users copy assets into their project folders, allowing authors to ignore the technicalities involved in configuring each asset type.

Supporting multiple platforms requires significant tuning to achieve good performance in each of them, for example, in terms of quality or lighting; and is, therefore, a multi-step, knowledge-intensive process in Unity. For uA, we have significantly simplified the generation of executable distributions for each of the target platforms by providing a simplified builder. This builder asks authors to select the platforms where the games are to be played, chooses sensible default values, and creates executable versions of games for each selected platform.

3.1. uA extensibility

Unity, and game engines in general, rapidly evolve as new trends and technologies emerge. To leverage this evolution, uA is easily extensible, allowing the creation of new modules to add support for features ranging from low-level and technical to high-level and educational. uA added two additional educational capabilities to the core features adapted from eA: learning analytics and location-based games. Both features are described in the following sections.

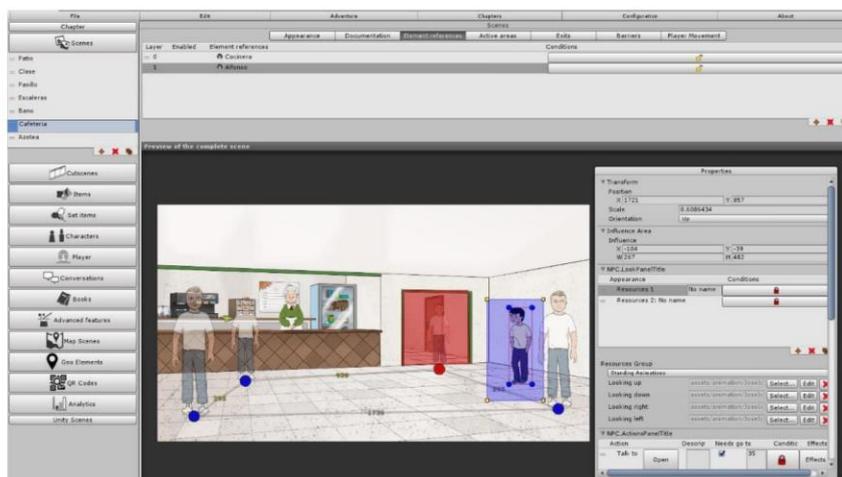


Figure 3. uAdventure' simplified editor for a third-person game, showing several characters in a scene and the Unity-like inspector at the right

3.1.1. Learning analytics extension

The Learning Analytics (LA) educational extension included in uA is far more flexible and powerful eA's assessment system, which was based on the SCORM specification. Using xAPI standards-compliant LA allows us to consider all in-game user actions and to represent in-game relevant facts as analytics data traces. To simplify the application of LA, uA allows easy authoring of a Game Learning Analytics model (Freire, Serrano-Laguna, & Iglesias, 2016), translating in-game events into analytics data transparently. In our implementation, we use the xAPI Serious Games (xAPI-SG) Profile (Serrano-Laguna et al., 2017), which is a close fit for our game model. Since xAPI is event-based, an xAPI-compliant server can gather the different events and states generated while playing through non-linear stories to later reconstruct the entire gameplay. During analysis, traces that provide evidence of in-game actions related to learning objectives are essential to evaluate player knowledge of those objectives. For this reason, authors that wish to use LA correctly must first identify such learning objective-relevant traces in the game and ensure that they are correctly reported via game-process traces to later inform the analysis process.

By having out-of-the-box support for the xAPI-SG profile, and not forcing authors to deal with low-level analytics details directly, uA greatly simplifies the use of LA for novice authors. Most major uA elements, such as scenes or conversations, can generate xAPI statements automatically. To measure game progress and identify educationally-relevant in-game events, we implement *completables*, as constructs that are configurable in a high-level editor (Figure 4) based on milestones. The milestones link in-game element interactions or game-status changes to progress reports for completables; and configure the score calculation based on current game variables or on previous scores.

Analytics data generated from uA games are handled by a *tracker* component embedded into games; and can be sent to an external server and stored locally for later retrieval, an option which is useful when there is no network or network disruptions are to be expected. Consequently, uA includes a section to configure tracker settings, covering both the format used to report the traces and the reporting method, including offline storage, LRS settings, and credentials, or both.

3.1.2. Location-based (LB) games extensions

Two extensions have been developed for both outdoors and indoors positioning: GPS and QR. Both extensions are part of a Location-Based (LB) game mechanics pack (Pérez-Colado, Pérez-Colado, Martínez-Ortiz, Freire-Morán, & Fernández-Manjón, 2017b) to be used in mobile environments. The GPS extension includes functionality for outdoor positioning and a new type of map-based scene, mixing both physical elements (e.g., points-of-interest or regions) and adventure elements (e.g., characters). These elements include a new set of actions for physical areas such as entering, exiting, looking towards certain points (using the mobile compass), or in certain directions. Indoors positioning is handled by an integrated QR code and scanner system. Among other functionalities, the extension includes navigation to guide the player in the scenes and the possibility to trigger the opening of uA scenes based on player location. Lastly, to simplify the development process, we have implemented a location-setting debug window to manage and simulate player location from the editor, allowing games to be tested locally.

To guarantee a good player experience in location-based games, we have identified several best practices. First, a companion-character figure should be placed in the map scenes to assist players (e.g., when they get lost or forget their pending tasks), using conversational menus and setting up navigation if needed. Second, we always recommend using moderate distances in navigation in order not to lose players' interest in the game. This aspect was already pointed out by players of our first location-based pilots (Pérez-Colado et al., 2017b). Another recommendation to keep players' interest is to enhance paths by adding enemies (such as zombies) and/or treats (such as extra points). Lastly, we always recommend using zoned uA scenes to represent indoor environments – where QR codes might, if needed, be placed.

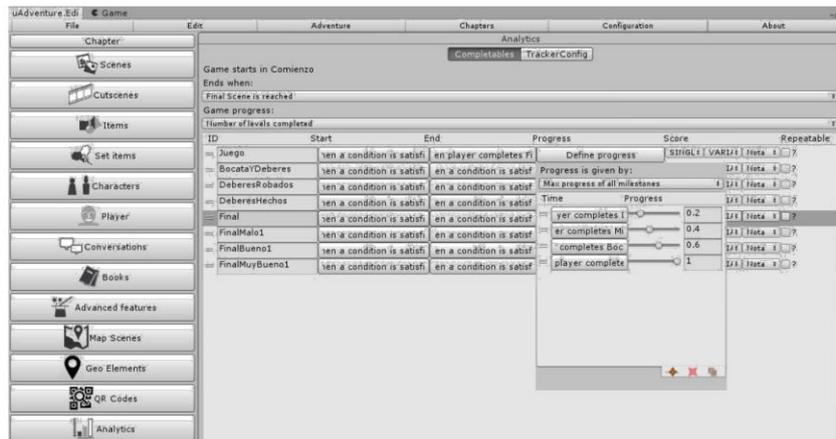


Figure 4. Completables editor in uAdventure’s learning analytics extension, showing a completable’s progress editor open with several completable steps

4. Testing uAdventure in different settings

To study the extent to which uA’s goals have been achieved, we have conducted several tests. The goals of these tests are two-fold: to measure the degree to which uA allows non-experts to author SGs; and to measure and improve software quality towards a full uA 1.0 release, including both the LA and LB extensions. Additionally, these experiments should be understood as pilots guiding the development of uA itself: after each round of testing, we used the results to identify and address bugs and to improve the user experience for subsequent versions.

Testing of uA has been performed in an iterative process divided into two main stages: (i) pilots, testing uA using a small and controlled group of authors with varied profiles, and (ii) use uA in actual undergraduate and graduate courses. The pilot testing helped us to fix bugs and identify possible improvements in uA, while the courses used uA as a teaching tool to introduce different SG aspects, and allowed us to study how users approach SG creation that satisfied different design requirements such as achieving certain learning objectives, using branching and multiple endings, or using geo-localization features. Figure 5 depicts the relationship between the different experiments carried out and the uA features under test.

4.1. uAdventure pilots

We performed a preliminary evaluation with users from different profiles, including both non-technical users (e.g., teachers and artists) and programmers. The purpose of these pilots was to find usability problems in uA and guide the development of uA and its supporting materials (e.g., user’s manual); and also to verify that its users were capable of: (i) developing narrative adventure game examples, (ii) create non-linear stories with multiple endings, (iii) develop their own stories autonomously and (iv) build actual games.

Participants with no prior experience with uA were provided with uA’s user manual and uA itself. The manual introduced the main SG narrative elements, concepts, and features of uA. Then, participants were asked to implement the eleven self-guided examples contained in the manual by themselves. By completing all examples, participants created a simple non-linear story with multiple endings. Finally, participants were also encouraged to develop their own stories to test high-level actions not included in the examples and to build and test their games on the Windows and Android platforms.

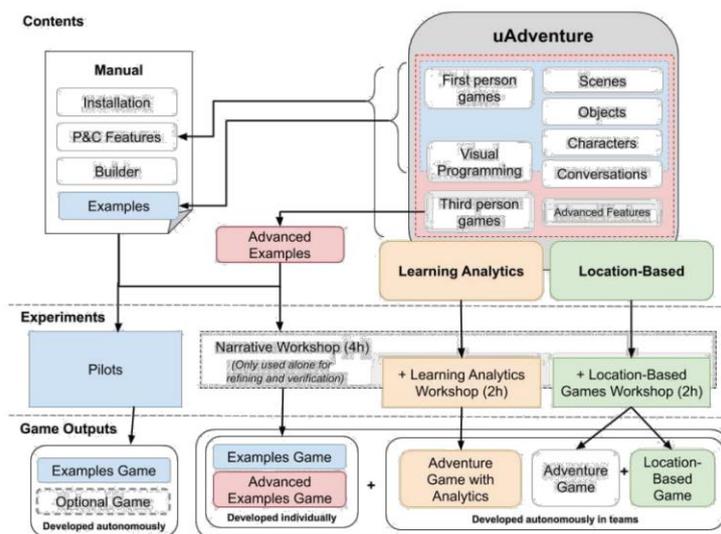


Figure 5. Relationship between different experiments (mid), uA features (top), and outputs (bottom)

After the experiment, a questionnaire was used to build profiles of participants, and to measure the perceived difficulty of each task that they performed into four levels: easy, normal, difficult, or not-attempted. In total, 110 simple high-level user actions (e.g., “create a scene”) were identified to test the main uA functionality (of which 59 were covered in the examples). The profiles questionnaire consisted of a self-classification as a non-technical user, artist, or programmer; and a set of questions regarding the participant’s experience creating stories or narrative games to estimate their “narrative experience” (see Table 1). Finally, participants were also asked about their success and questions while following the examples, their goals, and the degree to which the manual helped them through the learning process.

We first grouped the results by user profiles depending on their answers regarding computer use: artists used the computer mainly for image processing or 3D software, programmers mainly for programming, and the remaining users were classified as non-technical. Then we calculated the average success of the goals and high-level actions. Table 1 shows participant backgrounds, their level of understanding of the manual and uA’s main concepts, and the number of goals and high-level actions they completed. The last section analyzes task complexity and hence, uA’s success in terms of interface simplification.

According to Table 1, usability was high: participants finished 95% of the examples successfully, which was corroborated by the 87,4% of the examples actions completed; the percentage of incomplete or “found hard” high-level actions is correspondingly low, and most high-level actions were rated as simple (~65%). Individually, artists had the lowest success in completing examples: they had 15% more untried high-level actions than the global average. Regarding the goals of allowing users to create non-linear stories and author multiple endings, the results show an average success rate of 70%.

All users reported finding the manual helpful. However, they also reported multiple difficulties using uA. For non-technical users, doubts regarding uA’s model were probably related to their lack of narrative experience. Programmers also reported similar questions. On the other hand, questions regarding effects and conditions are, as expected, more frequent for users with no programming background. Finally, another relation is present between the questions asked by users and how complex they reported to have found the high-level actions. For example, non-technical computer users only reported ~26% of completed actions as simple, compared to 79-72% for artists and

programmers. Finally, participants also reported 29 different issues, misconceptions, and difficulties, of which 27 were considered to have merit and fixed in a subsequent release.

Table 1. Summary of results for different profiles (non-technical, artists and programmers)

	Participants by category			All
	Non-tech.	Artist	Programmer	
User profile				
Number of users	2	2	6	10
Narrative experience (%) ^a	50	100	78	77
Experience with SGs (%) ^b	80	80	70	74
Manual				
Read entire	1	1	4	6
Found helpful	2	2	6	10
Asked questions on model elements	1	0	1	2
Asked questions on effects and conditions	1	1	2	4
Goals				
Examples completed (% , n = 11)	95	86	97	95
Non-linear story created	2	2	3	7
Multiple endings?	2	2	3	7
Have built the game	1	1	2	4
High-level actions (110 in total, in %) ^c				
Completed	65.4	51.3	74.4	68.0
... of which found simple	25.7	79.8	72.1	64.5
... of which found normal	74.3	18.4	26.9	34.6
... of which found hard	0.0	1.7	1.0	0.9
... actions covered in the examples	85.6	72.6	92.9	87.4
... actions out of the examples	43.3	27.6	53.5	46.3
Not completed	0.0	3.6	0.9	1.2
Not needed / tried	34.5	45.0	24.6	30.6

Note. ^a = Avg. between users. Each user value is calculated with the sum of experiences: 0-30% in playing narrative games, 0-40% in creating narrative content (e.g., novels) and 0-30% in creating narrative or serious games.

^b = Avg. between users. Scale from 0% (I do not know what they are) to 100% (I know them, and I have used them).

^c = There were 110 possible high-level actions, such as "select a scene background," that could be performed in the version of uA used in this experiment.

4.2. uAdventure as SG teaching tool in university courses

The previous pilots helped us to identify areas for improvement before attempting more ambitious experiments. By the end of the previous round of pilots, users could generate simple non-linear stories with multiple endings just by following the guided examples in the manual. We then improved both manuals and examples based on the results of the first testing round. However, the examples in the manual were relatively simple, and all participants implemented the same stories: we wanted to test uAdventure on a more realistic scenario, where a new set of participants with no uA knowledge could learn it and explore larger stories, different from each other's, in a more agile development environment.

For the second round of testing, we used uA as part of both undergraduate and graduate courses, dedicating several sessions to first teach the use of the tool, and then allow students to build their games. For this second round, we adopted a workshop-based approach, instead of the self-taught approach of the first round. This new structure offers a faster learning process, where users have a better foundation in both authoring and the serious game development process, and allowed us to carry out experiments to explore more advanced goals in smaller periods when compared to a self-taught approach. However, a workshop-based approach can also lead to false successes due to the influence of the learning process. In the case of uA, both the manual and examples had been already proven to be effective, so a workshop based on the manual's updated contents and examples allow us to start up with a realistic foundation not too different than self-taught users.

After the first round of experiments, we released an updated version (uA Beta in Figure 6), addressing all identified limitations. We also designed a narrative adventure game training workshop (*narrative workshop* for short) to address the most common conceptual issues found in the first round. This workshop requested participants to build more elaborated games and to build the example game described in the manual and used in the initial pilots in just one 2-hour session. It also included a second 2-hour session, where more advanced uA aspects such as third-person games were covered. As shown in Figure 6, the narrative workshop was tested in two courses with a diverse set of participants, including both design school and post-graduate students. The two following workshops included many more sessions and allowed students to games in teams based on their ideas. While student teams worked independently, they still had expert assistance available. Besides, these longer workshops included LA or location-based development as each plugin became ready for testing.

The rest of this section describes the common aspects of the courses where the experiments took place, while the following sections describe the particularities of each of them. All courses took place in the Faculty of Informatics at the Complutense University of Madrid, and therefore participants can be considered programmers following the classification used in the first round of experiments.

In all cases, participants were introduced to the following topics: adventure games, narrative mechanics, uA's narrative model, and features, and learning-objectives-based game design. This last topic was introduced through a demo session implementing a SG based on the examples used in the pilots. After this introductory lesson, participants were asked to implement the example game with a teacher nearby to help with any issues, ensuring that all of them fully completed the example, and closing the initial two-hour session. After this, participants were introduced to the features and examples required to build third-person games, after which they were asked to create a small third-person game example. This second session also lasted two hours.

After these two sessions, participants were asked to form teams of 4-5 people and use uA to implement one or more SGs covering different aspects and features during the next weeks. During the development, additional development-only sessions took place, with uA experts assisting the participants and providing guidance when requested. In general, the expected length of games ranged from 5 to 10 minutes and had to include multiple endings. To implement the game, participants were provided with graphical resources from an existing school-based third-person game; but were also encouraged to create or use external resources or plots.

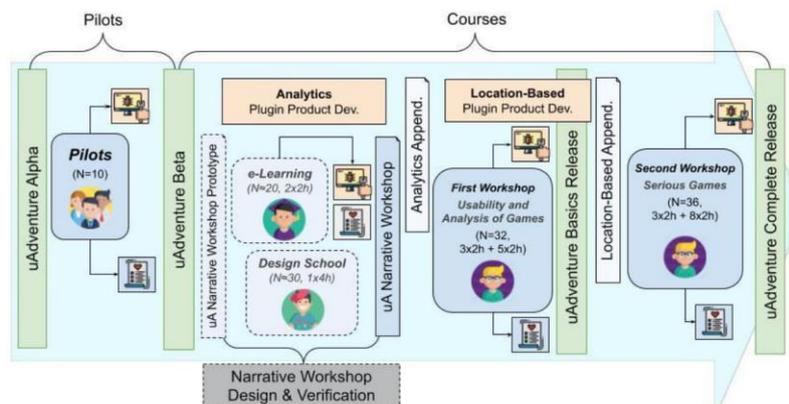


Figure 6. Timeline of the experiments (blue rounded boxes), uA releases (green), and development efforts (orange) and short workshops (dashed borders)

Finally, to improve uA software quality, participants were asked to report and submit issues for any bugs, ideas, or new features that they found or thought of while using uA. Reporting and triaging were performed through the GitHub issue system.

4.2.1. First (LA) workshop: introducing game analytics

The first workshop was designed for the “Usability and Analysis of Games” course for undergraduate students. For this reason, it included an additional two-hour session specializing on the LA extension in addition to the four-hour foundation sessions. The additional lesson introduced LA concepts, and the analytics generated by uA games, including the different traces and formats, completables, and how to configure the tracker.

This workshop focused not only on proving that users could create SGs with learning objectives, but also that they could generate analytic data from the game to whether to understand if players had learned. Hence, participants were also requested to create games with multiple endings and to include a list of analytics traces generated from the game and a document describing how the traces would be used to assess the learning objectives.

4.2.2. Second (LB) workshop: creating narrative serious games with location-based mechanics

The second workshop was designed for the “Serious Games” course for undergraduate students. After the initial four-hour foundation sessions, participants received an additional two-hour lesson covering uA’s location-based features, location-based SG best practices, and LB authoring and debugging. The location-based features included augmented map-based scenes, region-based interactions, QR-based positioning, and navigation. Besides, participants were instructed on the differences in positioning accuracy indoors versus outdoors, the use of companion characters, and guided navigation to help players not to get lost in LB games, and the importance of not asking players to travel long distances with insufficient feedback. Finally, while the workshop teacher demonstrated LB game creation, participants were asked to follow and reproduce all authoring steps on their own copy of uA.

Participants were requested to develop a SG covering uA narrative features, including a simplified game design document (GDD) describing the learning objectives and how uA elements cover them; and to develop a location-based SG using the main location-based mechanics, including a simplified game design document mentioning educational objectives and how location-based elements and mechanics in the game cover each objective. For each game, teams were also asked to create three-minute gameplay videos.

5. Results

Around 50 students participated in the two short workshops (in “e-Learning” and Design School courses) to refine and validate the four-hour foundation sessions explaining basic and advanced uA concepts and use; and 68 students participated in the two main workshops (in the “Usability and Analysis of Games” and “SGs” courses), which also introduced Learning Analytics (LA) and Location-Based games (LB), respectively. In these two workshops, students received 6 hours of lessons each and were divided among 15 teams, with a team size average of 4 people for the first workshop and 5 for the second workshop. Participants in the LA-focused workshop had around 10 hours of labs during which they could contact uA experts, and those in the LB-focused workshop had 16 hours of similar labs.

A total of 22 SGs were created with uA based on each teams’ ideas (5 of them third-person games), 7 of them covering uA’s narrative “point-and-click”, advanced and educational features, 8 covering also uA’s LA features, and 7 focused on location-based mechanics. All teams implemented SGs with a learning objective-based design and multiple endings. The game topics were mostly unique for each team. Since participants in the LA workshop were provided with a set of assets for a school-centered game, they mostly focused on school life, tackling issues such as bullying, violence, drugs, plagiarism, and women in STEM. LB games were, on the other hand, mostly based on historical events in Madrid, featuring kings, writers, and wars.

To analyze the change in game complexity, we counted the different model elements in both the example game and the games created by student teams. Table 2 shows the comparison of these counts, where the first two rows reflect counts of game elements in the example and the average of team-built games. The last row displays average change, as a multiplier.

Table 2. Comparing game complexity in terms of discrete game elements between the example game and games built by teams

Game	Scenes	Cutscenes	Refs.	NPCs	Items	Actions	Conv.	Lines	Cond.	Effects
Example	4	4	12	1	1	5	1	17	14	22
Team Avg.	10	3	47.3	8.5	10.5	30.8	15.5	138	75	227
Increase	×2.5	×0.75	×3.9	×8.5	×10.5	×6.16	×15.5	×8.1	×5.3	×10.3

The complexity of team-created games was, in general, several times greater in all but one game element categories. The “Refs.” column of Table 2 counts element references to NPCs, items and atrezzo (3 in the example, 35 in avg. game), areas (1 in the example, 1.6 in avg. game), and exits (8 in the example, 10.5 in avg. game).

In terms of narrative complexity, games had an average of 15 conversations and 138 dialogue lines per game. This represents an estimated 10-to-15 minutes of gameplay for “point-and-click” adventure games when compared to the duration of plays of the same length, without accounting for replayability – as players are generally interested in exploring the different available endings. Compared to the example game, this is an 8-fold increment in narrative content.

In terms of game and branching complexity, we see a six-fold increment in actions, five-fold in conditions, and ten times more effects than the example game. Resulting games had on average, at least two endings (good and bad), with a few of them, including up to four different endings.

In games with LA, there were 7 completables with 11 milestones per game on average. In location-based games, at least one map-scene was used with 13 geo elements, 5 zone scenes, and an average of 5 navigation paths.

Table 3 provides an overview of software-quality related results: a total of 70 issues were reported, including 47 bugs, 9 enhancements, and 14 new features. Most of the bugs were solved, and many of the enhancements were implemented or planned while the workshops were ongoing. We prioritized bugs, and therefore only a small number of new features were implemented in this period, mostly centered on usability improvements in effects and conversations.

Table 3. Issues reported, accepted, and solved after the second round of testing

Issues	Bugs	Enhancements	New features	Total
Reported	47	9	14	70
Accepted	43	9	13	65
Solved	35	4	3	42

6. Discussion

The experiences described in this paper helped us to conduct a continuous process of improvements for uA. The results of the pilots described in the first testing round, in which almost no participant tried or succeeded in creating their games, helped us to mature the software (e.g., fixing bugs or implementing new features) for the second, and more successful, test pilots.

Participants in the pilots generally achieved the main goal of proving that even users with no prior experience in Unity, and to a lesser extent also non-programmers, can create at least simple serious games examples. However, our results also show the importance of having a prior narrative experience to be more proficient during the authoring process. Also, several users that completed all the assigned tasks reported having problems building non-linear stories or multiple endings, which was one of the purposes of the examples. Therefore, we carefully revised the manual, providing more background on uA’s narrative structure in the introduction, and generally improving uA’s internal help system.

The workshop results for teams developing SGs with uA and assisted by researchers are highly positive. They prove that uA allowed participants to: (i) create longer stories (with up to 15 minutes of gameplay) with more narrative complexity; (ii) work in small teams; (iii) work in more diverse game genres; and (iv) develop serious games using an agile methodology.

These last results were more successful (besides the bug fixing of the tool as a result of the experiment) due to the fact that new features implemented after pilots (and from first to the second workshop). Besides, the longer and guided formative sessions (in contrast to the pilots where participants only had the manual, and they were on their own for few days), the workshop had hours of guided formation and weeks of free work. Last, the planning of the sessions was modified to mitigate the students' difficulties. For instance, we focused more on model the design and the way conditions are planned to prevent mistakes.

Regarding complexity, we have seen in the results a great increase in terms of conversations, lines, effects, and conditions. The example game was already based in a real serious game named Fire Protocol with several endings, and our results show that most of the participants created games that were, on average, eight times more complex than the example in the manual. We found several promising games created by participants, such as the STEM awareness game for girls, and many of the location-based games that were historical-themed and could be joined to become a complete historical tour of Madrid showing very promising results for the location-based games extension.

During the experiments, we noticed that uA elements such as scenes, characters, objects, and conversations were easier to understand than game-state aspects such as conditions and effects. A possible explanation is that during the demonstration sessions more emphasis was done on demonstrating the narrative elements (due to the low experience of participants on these aspects) than on explaining conditions and effects, which were supposed to be more familiar for participants. This last assumption seems to have been wrong, maybe due to the higher cognitive load required to understand and apply the rule-based execution model of conditions and effects.

Since uA allows the development and testing of even early prototypes, it is a good fit for teams using agile development approaches. Most of the enhancements and new features suggested and later implemented were oriented towards improving this workflow, including the usage of dropdown interfaces to make windows simpler; making conversation-editor interfaces more flexible for extending nodes and options; or having more flexible dialogs that can include images to avoid the need for using effects for such simple tasks.

Throughout the tests, participants used uA to quickly prototype their game ideas, easily linking educational value to game elements. This experience highly contrasts with the full Unity experience, which is much more difficult to learn and where SG development requires significantly more effort. In this sense, our integrated experience approach is successful, as it simplifies authoring for participants. Regarding their feedback, they liked the tool and found it very inspiring. Many proposed ideas and new features, but in general, they were satisfied with current features and even impressed by their amount and quality. Artists, in contrast, requested more graphic flexibility, while programmers asked for proper ways to hook with the tool.

An important limitation is that, even if all participants were able to follow the guided example lessons successfully, many of them required expert assistance while authoring games. Having this assistance probably had a positive impact on the success of our results, but also accepts additional interpretations. On the one hand, it is normal for developers to explore different solutions while authoring, and in these cases, expert assistance helped them choose among several possible technical solutions, of which several could have been valid from a player's perspective. On the other hand, by providing this assistance, we were able to find aspects where users require more help (as issue-reporting has shown), and thereby reduce the need for assistance in subsequent experiments. Therefore, while we cannot state the extent to which uA can achieve its goals in unsupervised environments, it is certain that a newer version based on the lessons learned in the latest experiments would be an improvement over previous versions of the tool.

7. Conclusions and future work

This paper presents uAdventure, an SG game authoring tool, and describes how uAdventure was tested in different settings. With uAdventure, non-experts can start developing adventure SGs without having to invest in expensive software or the time to learn general game development. Moreover, with the eAdventure to uAdventure conversion tool, the lifespan of pre-existing eAdventure users can be expanded with minimum effort.

Implemented on top of the well-known Unity game environment, uAdventure allows authors to use high-level concepts (e.g., scenarios, characters, conversations) to develop "point-and-click" adventure games instead of dealing

with the complexity of the full-featured Unity. This way, uAdventure makes use of Unity's advantages in terms of performance, support for emerging game technologies, and ability to package its games for multiple platforms. The uAdventure tool brings together the main narrative elements of "point-and-click" adventure games in a simplified, easy-to-understand model, allowing non-expert game developers to focus on the educational features they need most. To this end, uAdventure hides or replaces most of Unity's user interface, which could be overwhelming for non-experts. Therefore, uAdventure is not just a Unity extension, nor a standalone editor, but an environment that transforms Unity into a "point-and-click" SG editor affordable to non-experts.

The first testing round included a formal evaluation with different kinds of users, yielding promising results for in simple narrative adventure game development. However, its results did not extend to more realistic games, although they were very helpful in improving uA and its ancillary materials. In the second testing round, we tested uAdventure through workshops in four courses, with two short workshops focusing only on narrative features, and two much longer workshops focusing on one extension each: learning analytics and location-based games, respectively, over many more sessions. In these last two workshops, students were asked to create full SGs, with very impressive results: all participants were able to successfully create their games in an autonomous development with expert's assistance, and several were of high quality. The second testing round also serves to improve uAdventure's quality and user experience, as participants identify many issues and enhancements that were considered.

Next steps in the project are focusing on three aspects: first, to verify that the uAdventure games can be actually used in real environments (e.g., schools); second, to test uAdventure with more diverse user profiles in non-assisted environments, and specifically with projects where participants contribute with specific but diverse skillsets, such as artists collaborating with educators and programmers; and third, to create new uAdventure extensions designed to improve the experience of expert Unity developers when using uAdventure as a prototyping tool. To address the first goal, we are currently developing a location-based SG (including learning analytics features) for the vice-rectory of Sustainability and Environment of the Complutense University of Madrid, planned to have a high volume of participants testing uAdventure's features in a complex and diverse environment. For the second aspect, we are planning a hackathon next year using the uAdventure final release in the Designer School of Madrid, oriented at multi-profile teams. Finally, to achieve the third goal, we are currently developing plugins that integrate Unity scenes as mini-games into uAdventure.

Acknowledgment

This work has been partially funded by Regional Government of Madrid (eMadrid P2018/TCS4307), by the Ministry of Education (TIN2017-89238-R), by the European Commission (Erasmus+ IMPRESS 2017-1-NL01-KA203-035259).

References

- Almeida, F., & Simoes, J. (2019). The Role of serious games, gamification and Industry 4.0 tools in the education 4.0 paradigm. *Contemporary Educational Technology, 10*(2), 120–136. doi:10.30935/cet.554469
- Amory, A. (2001). Building an educational adventure game: Theory, design, and lessons. *Journal of Interactive Learning Research, 12*(2), 249-263.
- Aust, R., Nitsche, M., & Pelka, J. (2014). Digital game-based learning and video games in teacher training. Conception, evaluation and results from Leipzig University. *Perspectives of Innovations, Economics and Business, 14*(3), 113–131. doi:10.15208/peib.2014.14
- Center for Technology Implementation in Education. (2014). *Learning with Computer Games and Simulations. American Institutes of Research*. Retrieved from http://www.ctdinstitute.org/sites/default/files/file_attachments/CITEd%20-%20Learning%20with%20Computer%20Games%20and%20Simulations%20FINAL.pdf
- De Freitas, S. (2018). Are games effective learning tools? A Review of educational games. *Educational Technology & Society, 21*(2), 74–84.

6.2. Simplifying location-based serious game authoring

6.2.1. *Cita completa*

Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2017). **Simplifying location-based serious game authoring**. In J. M. Doderó, M. S. Ibarra Sáiz, & I. Ruiz Rube (Eds.), *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM)* (pp. 1–9). <https://doi.org/10.1145/3144826.3145395>

6.2.2. *Resumen original de la publicación*

Pervasive gaming has been a field of exploration for the gaming industry the recent years. With the exceptional success of Pokémon GO and some others such as Ingress users are more than ever ready to experiment with games that are more focused into their actions and their playing contexts. In the serious game industry, the implementation of pervasive games is a very promising way to improve students learning process with a more cohesive and realistic experience. To take advantage of this opportunity we have developed a model for location-based serious games intended to be used along with other game genres such the adventure one. We have implemented this model in the uAdventure (uA) authoring tool, allowing non-experts for the simple creation for location-based adventure games without programming or complex ge positioning knowledge requirements. With uA we address to simplify the creation of gymkhanas, tours or guided visits with the added value of the adventure games. Finally, to improve the quality of the games, we have incorporated learning analytics into the system and tested the platform with a small experiment that showed promising results.

Simplifying location-based serious game authoring

Pérez-Colado Víctor Manuel
Dept. of Software Engineering and AI,
Facultad de Informática,
Universidad Complutense de Madrid
Madrid, Spain
victormp@ucm.es

Pérez-Colado Iván José
Dept. of Software Engineering and AI,
Facultad de Informática,
Universidad Complutense de Madrid
Madrid, Spain
ivanjper@ucm.es

Martínez-Ortiz Iván
Dept. of Software Engineering and AI,
Facultad de Informática,
Universidad Complutense de Madrid
Madrid, Spain
imartinez@ucm.es

Freire-Morán Manuel
Dept. of Software Engineering and AI,
Facultad de Informática,
Universidad Complutense de Madrid
Madrid, Spain
mfreire@ucm.es

Fernández-Manjón Baltasar
Dept. of Software Engineering and AI,
Facultad de Informática,
Universidad Complutense de Madrid
Madrid, Spain
balta@ucm.es

ABSTRACT

Pervasive gaming has been a field of exploration for the gaming industry the recent years. With the exceptional success of Pokémon GO and some others such as Ingress users are more than ever ready to experiment with games that are more focused into their actions and their playing contexts. In the serious game industry, the implementation of pervasive games is a very promising way to improve students learning process with a more cohesive and realistic experience. To take advantage of this opportunity we have developed a model for location-based serious games intended to be used along with other game genres such the adventure one. We have implemented this model in the uAdventure (uA) authoring tool, allowing non-experts for the simple creation for location-based adventure games without programming or complex geospositioning knowledge requirements. With uA we address to simplify the creation of gymkhanas, tours or guided visits with the added value of the adventure games. Finally, to improve the quality of the games, we have incorporated learning analytics into the system and tested the platform with a small experiment that showed promising results.

Keywords

Serious games; location-based; augmented map; GPS; augmented reality; ubiquity; pervasive; xAPI; learning analytics;

1 INTRODUCTION

Serious games market is expected to reach \$5,448.82 Million by 2020 [19]. One of the crucial factors is the consolidation and constant grow of mobile technologies; accentuated by the usage of BYOD (bring your own device) in education and companies. Serious games applications are diverse; using serious games in schools is an innovative way to improve

engagement of students; on the company side, serious games can be applied in the instructive and formative stage, being excellent to simulate corporate processes. For this reason, serious games may come in different formats and genres, focusing more into visuals, storytelling or in-game mechanics depending on the game purpose [5].

Mobile technologies allow for videogames and serious games to use novel mechanics and features; those involve using the context sources from the built-in sensors such as camera, radio sensors, GPS, accelerometer or compass; exploiting the players' ubiquity present in mobile. These capacities have impacted the growth of Augmented Reality (AR), now available for every smartphone user. Pervasive games defined as games that blends its in-game world with the physical world [12] is the super category of AR games and highlights the categories of location-based games, mixed reality games and affective games (games that involve user's physical responses).

Videogames in 2016, and especially pervasive games, have been highlighted by the global Pokémon GO¹ release by Niantic. The impact of Pokémon GO has been relevant in young people; it has had more than 650 million downloads and still has 65 million monthly active users by June 2017². One of the most direct impacts has been detected in players health [1]. Other case of success has been Ingress, the previous game of Niantic. Hence, Pokémon GO's (and Niantic games in general) success gives the community the chance to start developing other kind of games in the Pokémon GO format having more chances to be socially accepted [10].

Pokémon GO is well-known as a AR game; however, it fits even better in the location-based games as its AR factor has been revealed not as its crucial mechanic (many player have deactivated it) being the augmented map (AM) the most

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org

TEEM 2017, October 18–20, 2017, Cádiz, Spain
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5386-1/17/10...\$15.00
<https://doi.org/10.1145/3144826.3145395>

¹ <http://www.pokemongo.com>

² <http://expandedramblings.com/index.php/pokemon-go-statistics/>

important feature. The key difference between AR and AM is that, both focus into overlaying in-game elements in real-world, but the AR focuses in live images from the phone camera, while AM focuses on map information (that comes also from reality). While the AM is a typical feature of the location-based games, AR can be fitted into several categories (location-based, mixed reality, etc.) depending on the end use of the AR inside of the game. For instance, if the game mixes a specific real world location like a park or a museum with AR, it can fit in the location-based, but, if the game uses AR as game environment (like in Playstation's *Invizimals*³) it fits better in the mixed-reality.

Location-based games AM is the in-game main view or scenario based on a map that includes virtual elements on it. The map itself may come in different forms depending on the source: physical maps, based on satellite images and can be increased with height information to create more realistic maps (i.e. Mapbox⁴ SDK); schematic maps, based on an abstraction of the real world information, creating more simple figures and shapes, and often including real world information (such as names, places or POIs); vector maps (in the middle between physical and schematic) include information about landforms, paths, buildings, parks, etc., stored as vector instead of as an image. In addition to the map, there are the in-game elements generally placed in the map, and may be linked to a point or zone.

The AM normally shows an avatar in the center of the map to represent the player; located using satellite. GPS (Global Positioning System), GLONASS and Galileo are positioning systems is based on radio signals emitted by satellites. They can achieve a precision of a few centimeters but it commonly works with a few meters accuracy, decreasing when the signals are not in line of sight (i.e. indoors); and it can provide measures of movement speed, direction and altitude. However, other approaches are possible without using satellites. Close range radio devices such as Wi-Fi's and Beacons can be used to locate the player too [2, 23], and the augmented map might be just a support tool for orientation or might not be present at all. The benefits of this kind of positioning is that works the same indoors and outdoors. In addition, it may be implemented by using QR codes that are cheap and accessible for independent developers.

Developing location-based videogames can be achieved in two ways: (i) creating tailored games that fit exactly the needs using SDK's (i.e. Mapbox, Google Maps, etc.) or game engines (Unity, Unreal, etc.) but require high level knowledge, time, and invest or (ii), using a specific platform for this purpose, intended to simplify the process by providing templates of all the possible tasks this games involve.

There are several platforms in the market available such as DiscoveryAgents⁵, ActionBound⁶ or GooseChase among others. With them, developers can easily create location-based experiences (e.g. gymkhanas, tours, treasure hunts, visits) by including mechanics that require the player to interact with the environment (e.g. reach locations, take pictures or videos, scan QR codes). In general, the experience includes several

customizable tasks (normally sequential) the player must perform during the play. Despite the differences elements they may or not include, most of the platforms include analytics that allow the game developer to track players, inspect their responses and see leaderboards.

All these platforms fit in the serious game market, but there are some specifically developed for teaching, such as DiscoveryAgents, MIT's MITAR and eAdventure for Android. DiscoveryAgents aims to teach players about public parks and natural reserves. An experiment in the Calgary park evaluated with tests reported more fun and better emotions than the players that didn't use the app; and in terms of knowledge they reported similar results to the students that visited the park with the park guide [6]. MITAR platform was developed in 2008 [8, 13] by using AM with PDAs (Personal Digital Assistants) and highly involving narrative content. MITAR has been applied in the learning process in two ways: (i) using its output games; and (ii) using the platform for learners to create games themselves. In the first players had to interact with the playground through their PDAs to acquire information about the environment, talking to virtual characters our using simulated sensors; this helped to achieve a more connected and authentic learning, highlighting players' engagement. In the second approach students had to choose a topic and create a game that their classmates were going to play in the end of the experiment; conclusions shown that students tend to get deeper knowledge to explain it in the game, and they need to experiment themselves the playground to replicate that in the game.

The previous examples revealed that using location-based serious games helps users to better settle down the subject by complementing deducing and experimenting in the real world with the in-game content [4]. In addition, AM also provide an interactive way to teach learners about cartography, orientation and investigation.

As seen in the previous examples, we expect that including more narrative content in location-based content will result in richer and more interesting serious games. eAdventure (eA)⁷, developed as Open Source in Java, was launched in 2007 and allowed the creation of serious 2D point-and-click adventure games eA [21]. A test version was developed for Android intended to use location-based features such as scene loading depending on player's location and QR code but, due to Java issues and project discrepancies, the branch was discarded and the project has been reimplemented in Unity as uAdventure (uA) [16], offering new possibilities for multiplatform. For this purpose, we propose using uA platform as the perfect target to introduce location-based mechanics into it, such as AM. The resulting games will not only fit in the location-based category, but also in the adventure category, resulting in a new game genre we call the location-based-adventure or located-adventure.

In this paper, we propose a generic model for located-adventures and location-based games, exposing all the AM elements, interactions and different positioning methods and

³ <https://www.playstation.com/en-gb/games/invizimals-psp/>

⁴ <https://www.mapbox.com/>

⁵ <https://www.discoveryagents.net/>

⁶ <https://en.actionbound.com/>

⁷ <http://e-adventure.e-ucm.es/>

provide an example implementation. In addition, editors for these features will simplify the task of creation this format of games. Finally, an analytics model will be proposed for all the new interactions to be merged with the already existing analytics xAPI model present in uA.

2 A GENERIC LOCATION-BASED MODEL FOR UADVENTURE

This section exposes the location-based game model as a generic approach to be used by game engines and editors and, in this case, by uA. As location-based games may be approached through different ways (i.e. Wi-Fi or Beacons), this paper will expose models for GPS and QR code location-based.

2.1 Positioning outdoors: GPS based model

Location-based key element is the AM and indeed, is the most important feature while playing outdoors as it both guides the player and provides information.

This AM can be included from an external library (i.e. Google Maps, OpenStreetMaps⁸, Mapbox, etc.) or be self-implemented. Due to maps contain big amounts of data, normally this information is segmented into tiles (regions) of different sizes (levels of detail) based on the zoom property. This segmentation is specified in the TMS (tile map service) specification [7, 20] and transforms tile coordinates to world coordinates; and so the world coordinates are expressed in meters that can be used in most a game engines. For self-implementation cases, we recommend to translate and scale world coordinates to local coordinates to avoid the overflow of the meters, by translating the map center to the world origin and then, use the scaled meters from the center to locate the elements over the map using coordinates. Once the map is ready, to augment it, it is necessary to include game elements. As maps mainly based on coordinates, the most useful classification for AM elements is whether the element is fully based on coordinates or is just an external element with location-based metadata.

Based on this classification, we propose a hybrid model using map-native elements fully based on coordinates (from standards GML and GeoJSON) and game-native external elements extended with location-based features. By combining both types it is possible to create gymkhanas, city tours, treasure hunts, etc. with rich visuals, ensuring engagement and a better cohesive learning. For instance, a city tour has a set of coordinates -map-native elements- connected via navigation with information attached, and a set of graphical elements -game-native elements- in the map representing features.

2.1.1 Map-native elements. Map-native elements are designed to be used inside of a map and are based on coordinates. These elements, from now on called GeoElements, rely on a set of latitude and longitude tuples to be represented in the map and are based in different standard definitions such as Geo-JSON or GML that also uses a set of coordinates to define map elements. Since these standards are very complex to implement due to the wide range of geometries [3, 14], we determine three essential geometries that are enough to implement several use

cases for serious games: POIs (Points Of Interest), Zones and Paths:

1. The POI is a single coordinate element in the map of interest or where the player has to perform an action. Since POIs are just a single coordinate, an influence radius is added for more flexibility about the POIs' range of influence.
2. The Zone is a range of ordered coordinates that determine an area. In-game, a zone is a special ambit where the player could be contextualized; being in or crossing the border of a specific area may be the trigger for an in-game event. Essentially, a zone is a polygon and the GML standard suggests defining them by a single exterior polygon and a set of interior polygons to be subtracted from the original polygon; in turn, being defined in two possible ways, by a linear or shapely ring. For simplicity, we consider the exterior polygon to be enough in most cases. However, a simpler way to determine a polygon roundness is to establish a generic influence radius, being unified with POI's definition.
3. The Path is a range of coordinates, like a Zone except that paths are not filled. Because of this, the path has a starting point and an ending point, determined by its order. In-game, Paths can determine barriers the player should not cross; routes the player must follow; or just informative routes for the player to know (e.g. suggested gravel paths, bike paths or rivers to avoid). The standard GML offers two ways to define Paths: linearly, as LineString, or shapely, as Curves. Due to curves editing complexity, we consider LineStrings are enough. As in the previous cases, Paths also should have an influence area to determine if the player is inside the path or has left it.

All three elements, then, have a geometry and an influence area. Based on these features, we have determined four different actions the player can perform with these elements: enter, exit, look-to and inspect:

1. The enter action is performed when the player crosses the limit of the influence area towards the inside of the geometry, coming from the outside (as the player could be inside at the start of the game).
2. The exit action is performed as the player leaves the influence area (where the player was before).
3. The look-to is an action that involves the use of a compass to determine if the player is facing an specific direction. This action can be performed with elements (i.e. if the player must face a game element such as the center of a polygon or path) or directions (i.e. if the player must look from a location to a desired direction such as a building's side). For the latest, a vector specifying the direction is used. This action can be also configured to be limited to the element influence area.
4. The inspect action is an action to be performed interacting with the element on-screen. This way, a click performed to the element can be used to trigger an event (for instance, clicking into a POI can display information or pictures).

⁸ <https://www.openstreetmap.org/>

This set of actions allows the connection of geographic elements with an interactive game experience fitting location-based game's needs. Nevertheless, the game native elements are also crucial for a better game experience. In the next subsection, we explore their characteristics and potential actions.

2.1.2 Game-native elements. Game-native elements are objects that already exist in the game engine meant to be used in virtual worlds, without any constraint for the gaming context; they include positioning configuration, physical information (for the physics engine) and behaviors to define aspects for each object. For instance, Unity uses `GameObjects`⁹ with Components to represent any game element. Hence, since the augmented map uses its own positioning system based on coordinates, these elements need to be transformed to be used adapting the virtual world position to this coordinates-based position.

The transformations of game-native elements mainly involve translating, scaling and rotating towards their desired position into the AM. Depending on the in-game purpose of the object, these transformations will vary with the possible user interactions. For example, in this AM environment, the elements are fixed in the map, by default, but also there could be other elements such as screen buttons, arrows or indications depending on the user position.

Therefore, we define three different positioning systems: map positioned, based purely in coordinates; radial positioned, based on other element position and other physical measures; and screen positioned, that does not depend on the map:

1. The map based positioning adapts the world coordinate to the AM local position including a scale factor (that works accordingly to the map zoom) and a rotation. To control the interaction, by using the player's distance to the object we can allow or disallow it. In addition, this distance limit can be used to hide and show the element whenever the player is in-range or not, such as Pokémon GO does. We identify this maximum interaction distance as the object influence, being represented radially.
2. The radial based positioning adapts the position of an element nested to another element in the map (e.g. the player avatar). For this, we use three parameters for: distance, angle and an additional "rotate around" parameter (to rotate the element as the player rotates and make the elements completely relative to the parent object). In this case, it is not necessary to block the interaction depending on the influence area as objects are mostly visual or utilities (navigation arrows, pointers, particles, etc.).
3. The screen based positioning adapts the world position to the viewport position. There are two ways to approach this: (i) by reverting the map transformation process, for which it is only necessary to reverse the process described by the TMS, and (ii) by totally avoiding nesting the object inside of the map, and with that, avoid the parent transformations, which for the mainstream game engines, just requires

the object to be set independent, and move with the camera.

The interaction with these elements is made through bypassing the user interaction to them from the map, resulting in these objects acting as they do naturally. For instance, in adventures the elements will provide their look, use, grab, etc. native actions.

Since the nature or behavior of the native object to adapt for the map is a priori unknown, we suggest to create an object wrapper that converts the original object coordinates to each of these map coordinates, avoiding to adapt each object individually.

2.1.3 Nested scene launching. Mixing other serious game genres mechanics with the location-based scheme could improve the learning process by taking the best of each genre. Using game-native elements in the AM is the minimum approach; however, a bigger visual change might be useful for cases where there is a complex task to do in a certain location that may require using a better visual representation of the concepts (e.g. a mini-game, adventure-like or AR scenes).

To do this we propose do nested scene launching; that may happen because of any interaction (location-based or not). As the scene is launched, the game changes its appearance leaving the AM and the player interaction; even so, location-based mechanics may be present in a standard-game scene. For this, we propose that launching a nested scene could be linked to a specific map-native element that retains the scene while the player is in the element and, when the player leaves, the scene switches back to the AM scene.

2.1.4 Navigation. Most native maps include a navigation system for the user to know the best route to reach the destination. This navigation system may come handy for games which purpose may not require the player to know how to navigate or find a location. Since the game author might not want to take care of each navigation point we suggest using game elements as route points.

To launch the navigation system, it requires set of elements to travel and one traveling order. We suggest two different orders: linearly, one by one; or by distance, the closest first (potentially reducing the game playing time). This order is clearly useful for gymkhanas where challenges do not need to be accomplished in a specific order. Also, since each step might involve a task to complete, each element needs indicate if the element is completed as the player reaches it or if it requires a manual completion managed by in-game mechanics. This requirement is especially notable in gymkhanas or treasure hunts, where the player must resolve a clue or complete a challenge to step ahead to the next point.

The navigation system, then, must implement a public interface allowing other game components to configure and control the navigation present in the game. To select or step the current destination, its interface should provide public methods, so each game-element determines if the task has been completed.

There are several ways to approach the navigation system, depending on the game purpose. For instance, games in parks have freeway so the navigation only needs a single arrow to

⁹ <https://docs.unity3d.com/Manual/class-GameObject.html>

determine the direction; however, games in cities may use a street based navigation with distances, turns, etc.

2.2 Positioning indoors: QR code based model

Since location-based games are ubiquitous, they might happen indoors too. However, satellite oriented positioning systems are not so effective in this context [12, 17, 18]. There are several alternatives to achieve a positioning system indoors such as IMU based, optical-based or radio-based. Field tests with QR codes are sustained on the arguments that is an easy, simple and efficient way to determine the players location [11, 12]. In fact, the QR code positioning system is widely used in location-based games as it could be used outdoors or even in combination with GPS to create a better user experience (i.e. hidden clues in treasure hunts).

This model approaches location by assuming that scanning a QR code requires the player to be physically located in the same position as the QR. Since scanning the QR code is part of the game sequence, either (i) the game has to be programmed to expect a QR code at a certain point and continue the sequence after it; or (ii) the QR is able to include the in-game consequences inside the code for the game sequence to advance (i.e. mechanics such as dialogs, image show, or just game state changes). For linear games, the first approach might be easier to design and implement, but, for location-based games, where the game sequence might involve freely affordable challenges, the second approach is better as it allows to implement this nonlinear game sequences as well as the linear sequences that may happen inside of each challenge.

In order to protect the game content associated to the QR code, as well as the linearity required in some cases, we suggest to use identifiers as QR content. This way, the game acts as a mediator to access the content, and can even protect the linearity by only allowing the content access if certain conditions are satisfied (i.e. if the player has passed all the previous tasks inside the challenge).

The other key piece of a QR code model is the QR scanner; there are two possible implementations for it: (i) using an external app and launch the game passing the content to the game; or (ii) using an embed scanner inside of the game.

Using an external scanner app is the easiest approach but it breaks the gameplay as the player has to exit the game to be able to scan the QR. To implement it, it is necessary to define a URI based namespace that tells the OS to launch the proper application and embed it into the QR code content. For instance: an uA URI for QR could be "uA://qr/random-id"). This protocol must be registered into each OS the game is targeted for, ending up making the process not so easy in the end.

Using an internal scanner (by using a library, for example) requires more effort to implement with derived costs of money and time. Nevertheless, users will not have to exit the game to use it, avoiding to reload the game or restarting the services (GPS, accelerometer, etc.). Furthermore, the scanner can be limited to the moments that is necessary, simplifying the gameplay; and can be restricted to scan the QR codes the situation involve.

2.3 Implementing location-based adventures in uAdventure

The described location-based model has been implemented in uA extending its model. A runtime infrastructure that is able to parse the stored model in xml and creates all the requirements for the location-based games has also been implemented.

uA's key elements for authoring are the condition system, that allows to control the game behavior based on the global game state; and the effect system, that allows to visual program the consequences of the user interaction.

2.3.1 Outdoors positioning. The uA game platform has been extended by adding a new scene named MapScene to store all the configuration for the AM and all the elements, being those GeoElements or native elements (NPCs, Items or Atrezzos). Both MapScenes and GeoElements can be configured by using the Map module implemented for the Unity Editor that can represent polygons, areas, points and images, and handle the clicks for creating points or selecting, and the drags to move points or elements.

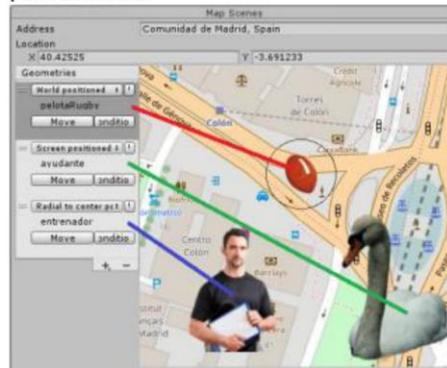


Figure 1: MapScene editor tab, displaying three native elements with all the possible positioning systems.

As uA is a Point and click adventures game engine [16], by extending it with AM it will generate location-based adventures. Hence, the process of creating location-based games based on AM in uA first start with the creation of the native elements (and its possible interactions) that will be added to the MapScene. These elements then, can be added to the MapScene in the MapScene tab (Figure 1); and configured with a specific positioning method (world, radial or screen), that determines how the element is placed, and some of its interaction constraints (such as the influence area).

Secondly, the zones, points and routes, along with their location-based actions (enter, exit, etc.), must be created before added to the AM. To create them, the GeoElement tab (Figure 3) allows creating the different polygons and configuring their influence area.

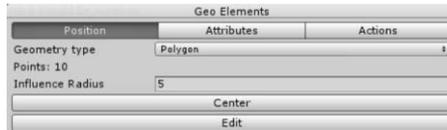


Figure 2: GeoElements main tab. An editor map as the one present in Figure 1 and 4 is present below the "edit" button.

For each GeoElement selected this tab geo-actions section (Figure 3) manages the different location-based actions associated to it. Each action will have a set of conditions to determine its access and a set of effects to determine the consequences.

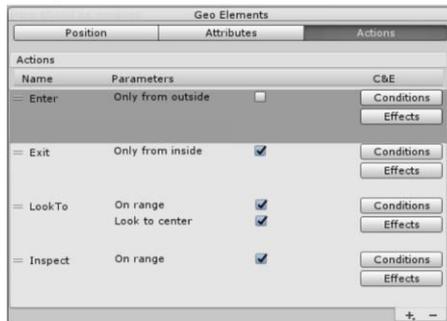


Figure 3: GeoElement editor tab. Top shows the geometry attributes. Bottom shows a set up of all possible location-based actions.



Figure 4: Location-based geometries. The red is POI, the blue is polygon and the purple is a path.

Once all the elements for a specific MapScene are prepared, they can be added into it by selecting them in the MapScene tab (Figure 4). The map scene view shows the editor map, where the elements can be configured. Both map and native elements are stored as a reference with a set of conditions that can be used to show or hide the elements depending on the game state. For instance: there could be a character used to teach the player how to play, but after the tutorial is finished, the character can be hidden by activating a flag after finishing the tutorial. Native

elements will also store their positioning information in the reference.

At the runtime stage, once a MapScene is loaded we instance a map based in tiles (Figure 5) based in the open source AM for Unity project of Baran Kahyaoglu¹⁰. This AM has the potential to work with tile maps based in both graphical and vector information and can be customized depending on each game requirements. However, this map has been modified to include factories for uA elements (both native and location-based) that are instanced in the corresponding tiles using the TMS transformation.

In the map view, the map-elements are displayed by using polygons of different colors with a tooltip with the name above it: POIs are represented with spheres; Paths are horizontally extended and rounded; and polygons are just transformed into meshes using triangulation.

On the other hand, the uA native-elements are wrapped runtime and positioned depending of the selected positioning system. It is especially remarkable that the implementation of the map-based positioning includes the hide and show mechanic based on the player closeness. A special effect using particles with the shape of leaves is displayed once the object is shown, similarly to the Pokémon GO discovery effect.

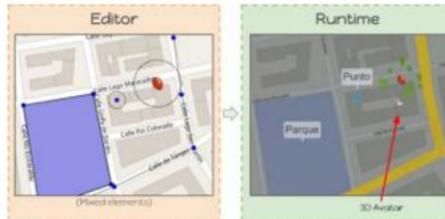


Figure 5: Correspondence of a set of elements in the editor view with the runtime view. Native element on top of the avatar is displaying the discover leaf-based effect.

Scene nesting discussed in the model has been also implemented for map-elements. At the editing stage, a special variant of the trigger scene effect named trigger geolocated scene effect has to be configured by setting the desired GeoElement the scene will be nested to. Once the player leaves the scene runtime, the nested scene is closed and the player is switched back to the previous AM.

Navigation system also described in the model has been implemented inside of the effect system (Figure 6). The navigate effect is used to configure the navigation controller with the desired steps and the order of processing (being by list order or by closeness order). Secondly, for those steps that require specific conditions to be completed, the navigation control effect makes the navigator controller change its current target. At runtime, the navigator has been implemented simply by using a straight arrow that indicates the direction of the destination (Figure 6). Future versions will probably implement the usage of a route system for cities.

¹⁰ <https://github.com/brnkhv/MapzenGo>

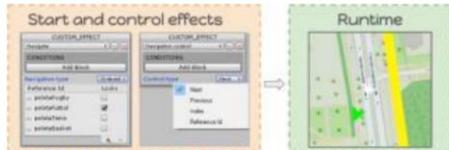


Figure 6: Navigation. In the left, the effect to set up and control the navigation. On the right, the runtime visualization of the navigation arrow.

2.3.2 *Indoors positioning.* As the model described, positioning based on GPS provides a good positioning quality outdoors but is not such convenient for using indoors. uA approaches to solve this by using QRs as described in the model.

The QR tab (Figure 7) inside the uA editor manages the creation and content of each code. Once selected, the tab allows to define the QR content formed by: a line of text that is shown after the code is scanned; the list of effects that will determine the consequences of scanning it; and the set of conditions that restricts the QR scanning that could break the game linearity. This content is explained hereunder.

The sequence of actions that happens after a QR is scanned determines how the game progress after the player scans it. For this, the model showed two approaches for linear and nonlinear situations. In this case, we considered that to afford nonlinear solutions (when needed), each QR contains the in-game effects (provided by the uA effect system) of scanning the code that controls the game progress.

On the other hand, to guarantee that the linear parts inside of the game are locked until the player reaches the point we used the uA condition system. This way, each QR contains the restrictions to be scanned. With this, the QR is ready to be used in the game.



Figure 7: QR code editor tab.

Finally, the QR tab also allows for saving and printing the QR codes from the uA editor, simplifying the QR exporting and reducing the game setup times.

As the model showed, the other key piece to work with QRs is the scanner. As uA is a gaming authoring tool, and we want to provide the best in-game experience and the easiest setup we decided that using an integrated QR scanner was the best

choice. This integrated scanner can be opened in-game by launching a QR prompt effect (Figure 8), that contains a message to show to the player, and the desired limitations for that scanning session. Since it is an effect, it can be linked to an on-screen button, conversation, or whatever the game designer requires.

In-game, the QR scanner shows the message in the top and opens any of the available cameras (preferring the back camera) showing an animated screen reminding a barcode scanner.

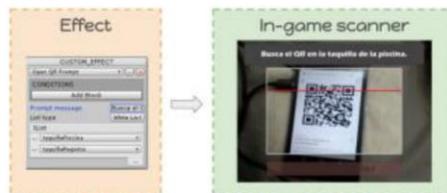


Figure 8: QR scanner. In the left, scanner prompt effect configured with a white list of possible codes. In the right, the in-game view of the scanner with the message on top.

3 LEARNING ANALYTICS IN LOCATION-BASED GAMES

Fitting a serious game in the learning process usually requires an assessment or evaluation process to verify the game has been effective. This can be approached different ways but new trends afford learning analytics (LA) as the next generation move since they can both cover traditional assessment, and be applied to more complex analysis techniques (i.e. datamining) that allows authors and teachers for a better understanding of the learning process.

There are several possible approaches for its implementation, highlighting xAPI¹¹ over all [9, 22]. For clarification, xAPI is a standard for traces that represent what actions or events are happening real-time, in a high-level language, identifying actors, verbs and outcomes. Hence, traces are easy to read and have direct value for LA.

uA implements the xAPI for serious games profile [15] from which are implemented four categories of events: (i) Completable, intended for tracing game progress, tracking each task in the game based on a group of milestones or steps; (ii) Accessible, for tracing the player in-game movement, such as stage changes; (iii) GameObject, for tracing the user's actions with in game elements; and (iv) Alternative, for tracing the player decisions, that are especially valuable for assessment.

In the location-based games ambit, however, the xAPI for Serious Games profile is insufficient as the verbs, terms and extensions are oriented to track in-game elements, missing aspects from the real-world environment actions (movement, looking, access, etc.). Hence hereunder we present the draft of what is going to be a new xAPI profile or location-based applications.

In location-based games we identify two different ambits for actions: (i) player's Position related, that include real-world coordinates information and contextualization about the area

¹¹ <https://www.adlnet.gov/xAPI>

where the player is; and (ii) player's Direction related, that include the different choices the player makes while traveling. An example of the Position ambit could be a trace where the player has Moved to an Urban-Area with the ID Madrid, and with exact coordinates as a Location extension. On the other hand, for the Direction ambit, when the navigation system of the game decides a route that links the starting point with the next point-of-interest, traces could identify whether the player has followed those directions, or has chosen his own path.

Once the system has been integrated with this model the traces must be collected and analyzed. For this purpose, uA uses the RAGE Analytics system designed to work with xAPI traces. In this system, a location-based traces analysis is done to finally generate a set of map-based visualizations, especially useful to rapidly understand how players perform location-based actions.

A more exhaustive analysis of the visualizations can help the teacher to determine, for example, (i) using a heatmap, which of the designed missions are misleading the students to an undesired location, or (ii) using a route-map, if a student has gone straight to the point-of-interest, or he has got confused for a period, and reached the destination with a delay.

4 FIELD TRIAL EXPERIMENT

With the purpose of validating the new uA upgrade for location-based games, we developed a test game that uses most of the features implemented. The experiment purpose is to explain the sport facilities and the usage of the registry in the Moncloa UCM campus having four parts: (i) tutorial for using compass and a navigation to one of the multiple sport facilities; (ii) game of gathering hidden balls in the different fields; (iii) explanation of the swimming pool facility (inside) with QR scan; and (iv) simulation of a document delivery at a registry with QR scan.

The methodology of the experiment was performed in a three-step process: (i) pre-test of knowledge; (ii) game play and analytics collection; and (iii) post-test of knowledge and satisfaction. The development of the game, materials and setup was made in less than a week, including testing.

The trial involved four people and they decided the game together as a single team; in a session that lasted for 51 minutes. As a summary of the results, we detected an improvement on the response rate from a 29% to an 86% of correctness.

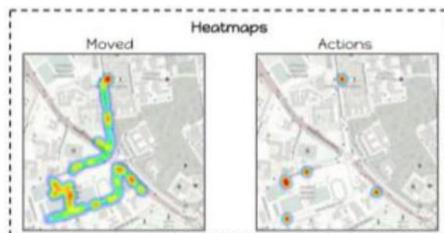


Figure 9: Heatmaps analytics reports. Left shows players route during gameplay. Right shows points where location-based actions were performed.

In the analytics report we could explore the players' movement, actions and behaviors in the location-based heatmap analytics

(Figure 9); and time measures of each task completion, whose results concluded a bad ratio of navigation/challenge time that players also reported in the satisfaction test.

From the satisfaction test the experience got an overall 7.2/10 score. The players' comments in the positive side highlighted the ball collection part, as they enjoyed having to deduce where the balls could be. On the negative side, they highlighted the long navigation periods without anything clear to do, that could be filled with some task about gathering random elements; and absence of a competitive leaderboard.

5 DISCUSSION

Location-based gaming market is growing enormously, specially highlighting Pokémon GO success, creating opportunities to the serious game market to move along and benefit from it. However, as this paper has discussed, there are no platforms oriented to create location-based games adequate for the low-cost serious games. In addition, these platforms also miss the opportunities arising from mixing up aspects from different game genres to better focus the objective of each learning situation (i.e. navigation vs simulation).

In this paper, we present a location-based model designed to be integrated in other serious game authoring tools supporting other game genres; or it also can be used as a model for location-based serious games themselves. This model, that is based on well-known standards aims to be compatible with existing map models, but also creates a layer of interaction especially valuable in location-based serious games (e.g. enter, exit, look-at). Secondly but not less important, this model aims to mix game genres and therefore it allows for the integration of native elements into the map and even nested launch other genre's scenes. Because of this, we've exposed different positioning techniques and behaviors mixed elements should use. Complementing the map representation, navigation systems helps the player to move in the maps and teaches players about orientation and navigation.

This model aims to work both outdoors and indoors, so we showed different approaches highlighting a possible implementation based on QR codes. By the usage of QR codes it is possible to complement the system and provide better game experience. Future work for the uA platform will involve the implementation or location based in Beacons or Wi-Fis too.

Narrative serious game is the genre we attempt to integrate location-based model into. Our authoring tool, uA has been extended to include several tools for authors to easily create map-based elements and mechanics and to integrate the adventure game elements into the location-based model. In fact, the editing tools provide nice visual representations of the AM design and tools such as reverse geocoding services to avoid dealing straight with coordinates.

Finally, to improve the serious game life-cycle, assessment is made by using the novel approach of integrating learning analytics. We've drafted a location-based xAPI profile that can be used for any kind of location-based applications and we've created the proper visualizations for non-experts to understand and evaluate players.

As a conclusion, with our small field test involving real users in a simple game context, we've obtained promising learning results (in terms of knowledge and engagement) and nice receptions from the player's side. Soon, we expect to test this

model and provide more realistic conclusions comparing different learning methodologies. However, at this moment, the uA platform is still in early stage, needing a huge push in terms of robustness and usability, but still we hope this paper will impulse better learning experiences for students, more connected with the real-world and, in the end, more authentic.

6 ACKNOWLEDGMENTS

We want to specially acknowledge Baran Kahyaoğlu for his open source MapzenGO project. This research has been partially financed by the Regional Government of Madrid [eMadrid S2013/ICE-2715], by the Ministry of Education [TIN2013-46149-C2-1-R] and by the European Commission [RAGE H2020-ICT-2014-1-644187, BEACONING H2020ICT-2015-687676].

All the map tiles appearing in figures 1, 5, 6 and 9 belong to OpenStreetMap and are licensed with Creative Commons Attribution-ShareAlike 2.0 licence. More information about OSM copyright can be found at www.openstreetmap.org/copyright.

7 REFERENCES

- [1] Althoff, T. et al. 2016. Influence of Pokémon Go on Physical Activity: Study and Implications. *Journal of Medical Internet Research*. 18, 12 (Dec. 2016), e315. DOI:<https://doi.org/10.2196/jmir.6759>.
- [2] Brassil, J. 2014. Improving Indoor Positioning Accuracy with Dense, Cooperating Beacons. *Procedia Computer Science*. 40, C (2014), 1-8. DOI:<https://doi.org/10.1016/j.procs.2014.10.025>.
- [3] Butler, H. et al. 2016. *The GeoJSON Format*.
- [4] Coulter, B. and Klopfer, E. 2016. Discovering Familiar Places: Learning through Mobile Place-Based Games. *Games, Learning, and Society: Learning and Leading in the Digital Age*. (2016), 327-354. DOI:<https://doi.org/10.1017/CBO9781139031127.025>.
- [5] Dickey, M.D. 2006. Game Design Narrative for Learning: Appropriating Adventure Game Design Narrative Devices of Interactive Learning Environment. *Educational Technology Research and Development*. 54, 3 (2006), 245-263. DOI:<https://doi.org/10.1007/s11423-006-8806-y>.
- [6] DiscoveryAgents 2016. *Calgary Parks and the Agents of Nature App*.
- [7] EPSG:3857: 2010. <http://wiki.openstreetmap.org/wiki/EPSG:3857>.
- [8] Falconi, R.F. 2010. Usability and game design : improving the MITAR Game Editor ; Improving the MITAR Game Editor ; Usability and game design : improving the Massachusetts Institute of Technology Augmented Reality Game Editor. *MIT*. (2010).
- [9] Freire, M. et al. 2016. *Game learning analytics: Learning analytics for serious games*.
- [10] Hobbs, T. 2016. Why Pokémon Go is a game changer for augmented reality and marketers.
- [11] Ilkovičová, L. et al. 2014. Positioning in Indoor Environment using QR Codes. (2014), 117-122.
- [12] Kasapakis, V. 2015. Pervasive gaming: Status, trends and design principles. *Journal of Network and Computer Applications*. 55 SRC-, (2015), 213-236. DOI:<https://doi.org/http://dx.doi.org/10.1016/j.jnca.2015.05.009>.
- [13] Klopfer, E. 2008. *Augmented Learning: Research and Design of Mobile Educational Games*.
- [14] Open Geospatial Consortium 2010. *gml:AbstractGeometry*.
- [15] Perez-colado, I.J. et al. 2017. Integrating learning analytics into a game authoring tool. *International Conference on Web-based Learning* (2017).
- [16] Perez-Colado, I.J. et al. 2017. uAdventure: The eAdventure reboot. *IEEE Education Engineering EDUCON 2017 Conference (en prensa)* (Athens, 2017).
- [17] PETERSON, B. et al. 2001. Measuring GPS signals indoors. *Proceedings of the Institute of Navigation's ION GPS-2001*. (2001), 615-624.
- [18] Schneider, D. 2013. New indoor navigation technologies work where gps can't. *ieee spectrum*.
- [19] Serious Game Market worth \$5,448.82 Million by 2020: 2014. <http://www.marketsandmarkets.com/PressReleases/serious-game.asp>.
- [20] Tile Map Service Specification: 2012. http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification.
- [21] Torrente, J. et al. 2010. <e-Adventure>: Introducing educational games in the learning process. *IEEE EDUCON 2010 Conference* (Apr. 2010), 1121-1126.
- [22] Velicanu, A. et al. 2013. INTEGRATING SERIOUS GAMES INTO E-LEARNING PLATFORMS: PRESENT AND FUTURE. *The 9th International Scientific Conference eLearning and software for Education Bucharest, April 25-26, 2013*. i (2013), 380-386.
- [23] Yang, C. and Shao, H. 2015. WiFi-based indoor positioning. *IEEE Communications Magazine*. 53, 3 (Mar. 2015), 150-157. DOI:<https://doi.org/10.1109/MCOM.2015.7060497>.

6.3. Learning analytics for location-based serious games

6.3.1. *Cita completa*

Pérez-Colado, V. M., Rotaru, D. C., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2018). **Learning analytics for location-based serious games**. *2018 IEEE Global Engineering Education Conference (EDUCON)*, 2018-April, 1192–1200. <https://doi.org/10.1109/EDUCON.2018.8363365>

6.3.2. *Resumen original de la publicación*

Pervasive gaming has experimented a huge commercial growth with location-based game successes such as Pokémon GO or Ingress. The serious game industry has an opportunity to take advantage of location-based mechanics to better connect games with the real world, creating more authentic immersive learning environments. Games such as historical tours, story-based exploration, laboratories, or flora explorations can greatly benefit from location-based mechanics. Location-based games usually include an augmented map to provide game context, which combines both traditional game-dependent elements such as avatars, and location-based elements such as areas or points of interest. For players, interacting with location-based elements may involve entering or exiting specific areas; or reaching a certain location and looking in a specific direction. To include standards-based learning analytics for location-based serious games (SGs), we have added support for player movement and location-based interactions to the pre-existing xAPI serious game profile. We have validated this approach through a case-study example that guided players through different sports-related facilities within a large outdoor area. This work have been carried out and it is available as part of the analytics infrastructure used in EU H2020 RAGE and BEACONING serious game projects.

Learning analytics for location-based serious games

Victor M. Perez-Colado, Dan Cristian Rotaru, Manuel Freire, Ivan Martinez-Ortiz, Baltasar Fernandez-Manjon

Dept. of Software Engineering and Artificial Intelligence
Universidad Complutense de Madrid, Facultad de Informática
C/ Profesor Jose Garcia Santesmases, 9 28040 Madrid, Spain
{victormp, drotaru}@ucm.es {manuel.freire, imartinez, balta}@fdi.ucm.es

Abstract — Pervasive gaming has experimented a huge commercial growth with location-based game successes such as *Pokémon GO* or *Ingress*. The serious game industry has an opportunity to take advantage of location-based mechanics to better connect games with the real world, creating more authentic immersive learning environments. Games such as historical tours, story-based exploration, laboratories, or flora explorations can greatly benefit from location-based mechanics. Location-based games usually include an augmented map to provide game context, which combines both traditional game-dependent elements such as avatars, and location-based elements such as areas or points of interest. For players, interacting with location-based elements may involve entering or exiting specific areas; or reaching a certain location and looking in a specific direction. To include standards-based learning analytics for location-based serious games (SGs), we have added support for player movement and location-based interactions to the pre-existing *xAPI* serious game profile. We have validated this approach through a case-study example that guided players through different sports-related facilities within a large outdoor area. This work has been carried out and it is available as part of the analytics infrastructure used in EU H2020 RAGE and BEACONING serious game projects.

Keywords—serious games; location-based games; geolocation; learning analytics; *xAPI*

I. INTRODUCTION

Technology can allow learning to take place in more authentic learning environments, outside of books and classrooms, leveraging the capabilities of current smartphones and tablets. The appeal of location-based games is exemplified by *Pokémon Go* [1], a free-to-play smartphone game that combines fun and outdoor physical activity through exploration, that has achieved both commercial [2] and social success. This approach may even have benefits for previously identified issues such as the so-called “physical activity crisis” [3] and those users with severe social withdrawal [4]. Location-based games have also been proved to support effective learning [5]; and are often combined with augmented reality (AR), which has also shown a positive impact on learning [6].

Developing a standardized Learning Analytics (LA) infrastructure that relates in-game user activities with the learning objectives is the first step towards a cost-effective method of validating location-based SGs. To connect and exchange data between location-based games and such a LA infrastructure requires a standardized method for communicating player interaction data. This paper proposes and describes a method that comprises the specificities of

location-based games, as an extension of the *Experience API* (*xAPI*) standard to support location-based SGs. We have tested the proposed solution in a small case-study experiment, where players had to visit and discover the main outdoor sports facilities at the Complutense University of Madrid. This small experiment provided insights on how to use LA for discovering how players explored and learnt about the different sport facilities. We concluded that the proposed *xAPI* extension is a simple but effective, standards-based and extensible means for coping with the LA data exchange between location-based games and a pre-existing *xAPI* standard-based Learning Analytics infrastructure.

A. Pervasive gaming

The generalization of smartphones and tablets has promoted software that can increasingly adapt to the context of everyday life. Applications that adapt to your location, the time, or your environment are some of the possibilities. Further examples that include sport tracking, location visiting, activity registration, or



Fig. 1 Pervasive games *PokémonGO* (left) and *Ingress* (right). Both games share avatar (red call-outs) and located game-elements (yellow call-outs). Those elements are only interactive when the player is inside of their influence area. In addition, device orientation (blue call-outs) is always present in the game.

live weather. Furthermore, applications can use this context information not only to adapt their content, but also as a primary method of interaction. For example, in navigation-based apps, entertainment recommendations or videogames, the primary input is often contextual. Applications that adapt to context or use it as a primary input are known as pervasive or ubiquitous, as they may be used anytime and anywhere, and they adapt to their context of use.

The gaming industry has been greatly influenced by the opportunities of pervasiveness. Since the first location-aware smartphones, games involving the use of location information have captured the interest of both developers and players. Examples include “Zombies, Run!”, “Ingress” or “Pokémon GO” (Fig 1). Generalizing, these videogames use player location as their main source of interactivity. Based on player positioning, events are triggered when players approach certain locations, enter or leave areas, or even look in specific directions. For instance, the game “Zombies, Run!” provides instructions on how to escape (imaginary) zombies based only on location, and audio indications allow the game to be played even without looking at the screen. However, in most location-aware games an augmented map is used to represent the game situation. The map includes real world features among with virtual elements from the game. *Pokémon GO* is a clear example of the usage of an augmented map (Fig. 1), as the player will be placed in the center of the map, and the Pokémon (collectable pet monsters) will appear in the map when the avatar approaches their game-determined locations. To make this kind of interactions possible, location-based elements have an influence range that reacts to the player location; only being displayed once the player enters the area.

In addition, as mentioned earlier, the sources of context used to create pervasive applications can also include time or even friends’ activity or messages from social networks. This can allow the ubiquitous experience to be further expanded. Pervasive games usually involve social interactions, from simple cases, such as as trading information or valuable elements, to more organized, such as group based events where the players should combine their forces to beat difficult challenges.

Pervasive experiences are also highly related with augmented reality (AR), of which they are a super-set. AR offers a complementary way for understanding the environment and enhance it in real-time as the player focuses on elements of interest. Where AR normally depends just on real-time images from the real world, it can be extended to other pervasive sources as location and orientation, and in such cases the AR experience becomes more authentic as it can behave differently depending on players location. For example, a generic image target for information can be placed in relevant in-game locations, but depending on the player’s location, it will show different information about the surroundings.

Commercially, pervasive gaming has been constantly growing as smartphones evolved. The success of mentioned games such as *Ingress* or *Pokémon GO* have settled up a solid foundation for investment and researching. Furthermore, we expect a huge impulse in pervasive gaming growth, as Warner Bros. have recently announced partnership with *Pokémon GO*

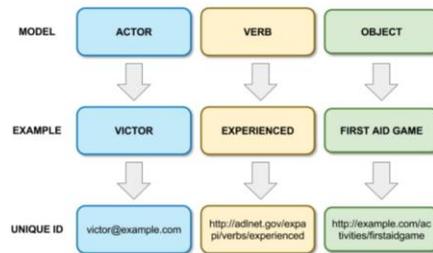


Fig. 2 Experience API data model specification with an example.

creator Niantic towards the creation of a pervasive game for the Harry Potter series [7].

For education, pervasive games offer a more authentic and situated learning, as the experiences occur in the real world and allows for learning to emerge from deduction and experimentation. The game can act as a guide to knowledge, and a verification and extender of the content learnt, including some of the benefits of gamification applied to education [8]. Also, it may require students to cooperate (as in real working environments) to complete the in-game challenges. For instance, in a game based on science environmental study, players could take different roles such as meteorologist, geologist or biologist. As the game progresses they would be required to cooperate with the other roles to mix their specialized knowledge towards a common solution.

1) Location-based mechanics

Pervasive games can leverage many different types of interactivity. In this paper we focus on the benefits of location-based games, leaving other approaches (time, social, etc.) as future work. This subsection explores the general mechanics for location-based games identifying low level concepts and atomic interactions.

In this genre of games, player use their position to interact. Because of this, the target elements are areas, and interactions are based on movement or orientation. As the work of T. Heintz [9] summarizes, the interactions happen against two types of elements: regions of interest (RoI or areas) and points of interest (PoI). The work explores the interactions based on position and orientation. The interaction happens when the player changes his position or orientation over time. Because of this, interactions referring to movement include entering or exiting, and staying inside or outside of an specific area. On the other hand, referring to orientation, the interactions happen when players, which are assumed to hold the device in front of them (so that the player can see what is in front of the device starts), start, stop, or keep looking at a particular location.

Finally, location-based games usually require reaching certain locations. For this purpose, navigation mechanics are usually included, such as reaching a set of points or even passing challenges at each point.

As mentioned, pervasive games can also mix social elements. Interactions with players can be treated as a special case of the previous described interactions (i.e. looking to a player or entering the influence area of a player). Other interactions such as forming groups or sharing game elements are general for any pervasive game, but it's a game-design matter if these mechanics will be location-limited.

Throughout this first section we have introduced the main concepts of pervasive methodologies and location-based gameplay mechanics and their application to SGs. In the next section we briefly describe *Experience API*, which can be extended to address the current lack of a standard for data collection in location-based SGs. Section III proposes such an extension, and in Section IV we analyze the results of applying the proposed solution in an outdoor experiment. Finally, we present a summary with conclusions and future work.

II. EXPERIENCE API

Experience API (xAPI), previously commonly referred to as *Tin Can API*, is a set of specifications for exchanging learning information, specifically to track learning experiences. It is continuously evolving, with the latest revision as of this writing being 1.0.3 (October 2017). *xAPI* is also commonly regarded as the evolution of *SCORM* even if the *xAPI* does not replace *SCORM*. *xAPI* focusses on user tracking in the education experience and not in the packaging or organization of the educational content itself [10] (actually *xAPI* will be used in many cases to track user interaction with *SCORM* content).

Within the *xAPI* specification, data is structured in statements composed of triplets of *noun*, *verb*, and *object*, as depicted in Fig 2. Actor, verb and object are the three main parts of a *statement*, which we will use as synonymous with *trace*, since statements trace the path of a player through a game. First, the *noun* represents an actor defined as anything with the capability to perform actions. Normally, the actor is the player; but any element involved in the application, such as an AI or other players, could potentially be used as an actor. Second, the *object* represents the target of the statement or trace, the element in which the action is being performed. For example, the object could be an in-game element, an area or scenario, or even real elements. Third and last, the *verb* represents the action that is performed. Depending on the *noun* and the *object* there is a set of verbs that can be used to track the different interactions between them. For instance, between an in-game item and the player, the verbs "grab", "use", or "inspect" would certainly make sense.

Optionally, an *xAPI* statement might include a *result* field, which represents the consequences of the corresponding action. The result field is used to convey scores, progress, or a set of extensions. Among the possible set of extensions, some are used to complement the information in the statement. For example, *result* fields can contain timestamps, changes to in-game variables, or secondary objects involved in the interaction.

In general, SGs have a set of specific needs regarding data to be tracked in order to gain useful insights of player behavior. Tracking the set of choices that players take while playing a serious game, or the level of completion of different parts of the game (or the game itself) is not easily achievable with the default *xAPI* specification. These serious-game specific requirements have been successfully fulfilled in the *xAPI* serious-games profile, proposed as an extension of *xAPI* and implemented by the Complutense University of Madrid for the H2020 RAGE project in collaboration with the Advanced Distributed Learning (ADL) initiative [11]. The next subsection explores this profile

A. *xAPI* serious games profile

The *xAPI* schema is especially relevant in SGs, where assessment has traditionally been performed by treating the game as a black-box, which once configured, simply outputs a set of grades. However, to play a serious game, players must perform a long set of interactions before reaching the end. In terms of evaluation, reducing this to a single score discards the potential wealth of information that can be gained by analyzing how the goal was reached. The additional information is also useful to assist the game life-cycle, and can guide improvements to the game layer. For instance, unexpected player behavior may be found, at the educational layer, as the player might not be acquiring the knowledge properly due to the way the information is presented. During the assessment process, the player (and the teacher) can benefit from warnings and alerts triggered by in-game actions. Such real-time analytics can help both understand and correct problems long before the game is over.

The serious-games *xAPI* profile (*xAPI SG*)¹ includes most, if not all, interactions that players can perform inside a serious game. For example, some actions include game-element interactions such as characters or items, scenario changes such as game level competitions, choices that the player may choose when alternatives such as questions are presented, game-variable changes relevant for analysis; and tasks that the player must fulfill in order to complete the game. However, in environments where the real-world context surrounding the player is relevant, such as in location-based games, *xAPI SG* offers no clear way to report environmental information such as location or orientation. Hence, for location-based SGs next subsection introduces the new location-based SG profile, covering new interactions and extensions including physical contextual information will be required.

III. XAPI LOCATION-BASED SERIOUS GAMES PROFILE

SGs that use location-based information require new elements in statements, currently absent from the pre-existing SG *xAPI* profile, in order to gather all the information and events produced during such gameplays. With these new elements it will be possible to analyze the player's interactions towards an understanding of the player behavior and even an assessment process. For instance, if a location-based serious game requires the player to navigate towards a specific location, but the player

¹ <http://www.adlnet.org/a-serious-games-profile-for-xapi>

never reaches this location and fails the game, having player positions included in traces will help both game developers and the teachers to identify and correct the cause, either by helping students or by changing the game. In addition, the more location-based context we introduce in each trace, the more potentially-valuable information will be available, via analytics, to improve how pervasive games are applied. Nevertheless, not only the contextual information is important, but also the actions that are derived from the location-based interaction with real world elements such as areas, buildings, parks or points of interest.

Location-based interactions are based on the player's physical actions performed towards these locations, including approaching or looking at them. This is especially relevant in location-based SGs, as one of the benefits of this games is the possibility to make the player deduce information from the real-world. The most cost-effective way to guarantee this is happening is to trace if the player has performed this interaction accordingly, such as looking at a monument, building or plant.

A. Extensions

Location-based extensions offer the possibility to include new contextual information, in particular the (i) location, (ii) orientation and (iii) navigation context of players, in any *xAPI* trace. These extensions are included in most of the cases when a location-based action is performed, and so will be available for use in later analysis to understand how players behave during their playthroughs. Furthermore, by including location-based extensions into the *xAPI SG* statements these traces can be promoted into a location-based analysis, covering the analysis of multiple mechanics in location-based games. For instance, using an in-game item, such as a sensor, is a SG statement, but in a location-based environment, it will result in different outputs depending on player's location. Hence by enriching the *xAPI SG* statement including the location extension makes possible to identify which information the player has obtained.

The *location* extension describes the global position of the player, expressed by latitude and longitude, in the "*GeoPoint*"² format. By including this extension in the traces, we can distinguish between the locations where interaction occurs (i.e. if the player opens the help menu, is it because he is lost?). This extension is especially relevant for location-based interactions as explained later in this paper.

The *orientation* extension holds the compass direction towards which the player's device is pointing, which corresponds to the player's heading in the common scenario where players look at the screen of their devices while holding them in front of their faces. It is expressed with a number in degrees that can have a decimal part (where 0° represents magnetic north, 90° points towards the east, and so on). The orientation extension adds a second layer of context which is especially important in pervasive games based on deduction, where players heading off in the wrong direction can easily explain errors in answering related questions. In addition, in

traces where the action implies looking, knowing the direction being looked at is especially relevant.

Finally, the third extension, the *guide* describes the navigational context being used to guide the player in the current moment. The guide extension is only used when a player is following navigational advice. The format is a list of directions and/or coordinates to follow. The guide extension is useful to detect when the player is misbehaving in terms of following the navigator's indications, or even perform late analysis debugging.

B. Actors and Objects

Describing location-based interactions requires defining new actors and objects for *xAPI* statements. The current proposal is the result of analyzing *GML* [12] and *GeoJSON* [13], and the location-related formats used in APIs of services such as *OpenStreetMap* [14] and *Google Maps* [15]. In these standards and APIs we have found several levels of area descriptions, such as terrain-like features, construction type, or even lower levels such as utilities. While any of location-based interactions could be used within traces, we will limit ourselves to a small subset of specific high-level elements: (i) buildings, for constructions; (ii) urban areas for any place that is not inside of a building but in a city; (iii) green zones for parks, forested areas, seaside areas and nature in general; and (iv) water for natural or artificial water-covered areas. This set of 4 high-level descriptors is enough for common location-based zone actions, and can be refined with additional optional descriptors when needed.

However, not all location-based actions are zone-based. In particular, a very important concept is that of "Point of Interest" (*PoI*), which can represent any important location relevant to the game, such as a monument, a preserved tree, or just the place where an interaction should take place. *PoIs* directly represent important locations that are therefore more relevant to analysis: looking at a *PoI* has an implied meaning for the system.

Since this is generic, sometimes the element is difficult to categorize or the categorization is not truly relevant for the specific application. In this case the *place* type can be used as a generic type to represent any location or area. This type is particularly important for cases where the players add locations to the game collaboratively, because it avoids forcing them to perform more fine-grained classification and therefore simplifies this type of collaboration.

While the previously discussed types provide a simple and compact coverage of location-based context, other types might be added in the future to provide increased descriptive accuracy regarding functionality, quality or shape. Social interactions during location-based gameplays will use the *xAPI SG player* type as both actor or target, and might include extensions with information about the user (i.e. the username).

Finally, in addition to the previous location types, location-based interactions can also include the usage of directions. The

²

<https://www.elastic.co/guide/en/elasticsearch/reference/current/geo-point.html>

direction type is intended for use in traces that indicate the intention of the player according to directions.

C. Verbs

In this section we describe verbs for location-based interactions based on the previously presented extensions and objects. In terms of interaction, new traces using these verbs allow actions such as moving, entering and exiting, looking, and following directions to be described in a standard way for pervasive games.

1) Moving

The basic behavior in a location-based environment is movement of actors. Whenever any actor changes its location a trace should be emitted indicating the new actor position, using the *moved* verb, specifying the coordinates by including the *location* extension. Since the movement is a reflexive action (what is moved is the actor), *where* it finalizes (when the trace is sent) is used as *target*. As mentioned in previous subsection, several kinds of categories could be used to identify this target, but we recommend using any of the previously-described area types. Optionally, the *orientation* extension could be added too, increasing the precision of the movement, and making it possible to make better predictions for real-time analysis.

2) Entering and exiting

When moving, there is the chance of crossing the limit of any of the relevant game locations. This limit can be either the boundaries of the region, or the influence area of a point of interest. For both cases, when the player crosses the limit there are two possible verbs that identify the player's final position with respect to it: if an actor that was out of the limit and then goes inside has *entered*, while one that goes from inside to outside has *exited*. In both cases, the object of the trace is the element that is referred to, which should be of any place type. As for extensions, the location is enough to determine the point where the limits have been crossed.

In-game, entering and exiting limits is one of the main location-based mechanics as it can derivate into different game events. For instance, entering an enemy limit could start a battle with the enemy. In cases where crossing the limits changes the game-state, *xAPI SGs* extensions are used to indicate the consequences of this crossing, such as changes in variables [16].

There are some special considerations to append to enter/exit traces. First, places might overlap in the real world. When this is the case, actors can enter several places in a row without leaving any of them. This implies that the analysis cannot expect an order in entering/exiting unless specific conditions, regarding to real world place overlapping, are set in the analysis. The second consideration is that the player could enter or leave limits without explicitly crossing them. This might happen when opening the game or application after entering; or when the device location cannot be determined, due to weak signal, interference, or location being turned off momentarily within the device. While the trace could eventually be sent later (for example once location information is again available), trace consumers must be aware of the possibility of missing location-related events.

Finally, there is the case of games that can only be launched within particular areas, for example within a classroom when first enabled by instructors. In this particular scenario, while the entered trace would be implicit, exited traces would likely result highly relevant for the analytics system.

3) Looking

When playing a location-based game, orientation is very important in the interaction with the real world. This allow to know if players have points of interest (*POIs*) in their line of sight (*LoS*). For this purpose, the *looked* verb is used to trace when an actor looks towards a relevant element (normally, towards its center to within a specified angular degree of tolerance). In this case, the *orientation* extension must be included to verify that the actor is indeed looking at the object, and, since the direction is subtle to the position, the *location* extension must also be present.

Additional care must be taken when dealing with *POI* occlusion. For example, within a painting gallery, paintings can only be viewed from certain directions, as they will otherwise be occluded by intervening walls and the fact that the canvas is only painted on one side. To detect this situation, a combination of being inside the viewing area of the *POI* and looking at the *POI* itself is required.

4) Following directions

When playing a pervasive game, it is very frequent to require the player to go to specific locations. To reach the location, a set of steps might have to be followed to help the player reach the destination. This is represented with the verb *followed* using the type *direction* as the target of the trace. When navigation steps are being followed, the *guide* extension, described in the extensions section, is used to indicate the current programmed navigation towards the destination, helping to discover misbehaviors.

Action	Noun	Verb	Target	Extensions
Moving	Player	Move	Places (*)	Location
Entering place	Player	Enter	Places (*)	Location
Exiting place	Player	Exit	Places (*)	Location
Looking at place	Player	Look	Places (*), especially POIs	Location Orientation
Following navigation indications	Player	Follow	Direction	Location Guide

Table 1. Summary table of all the combinations for traces including verbs and extensions. In blue, traces related to location interactions, in yellow, traces related to navigation. (*) Places represents all the possible types described in B except direction.



Fig 3. High level overview of the input traces going through the analysis system (default and custom) to produce output results.

D. Summary Table

Table 1 summarizes all the verb and trace combinations described in this section.

IV. ANALYSIS AND VALUE EXTRACTION

Displaying location based visualizations in real time requires to extract value from the data. Traces formatted following the *xAPI* location-based profile are sent to a backend server to be analyzed. There are two main parts interacting with each other in a *Learning Analytics (LA)* infrastructure: (1) client-side, where data is generated and (2) server-side, where data is collected, analyzed and stored. Initially, traces are generated in the mobile devices (client-side) on which the game is being played. During play, traces are sent over the network to a backend service that performs multiple actions on the traces (server-side). The different actions performed by the server-side are to collect the data, transform the data to facilitate later analysis, and then store the results waiting for the data to be retrieved and displayed. Furthermore, the results may be retrieved by an additional service and displayed in forms of an analytics dashboard composed of multiple visualizations.

A. Learning Analytics Infrastructure

LA infrastructures typically need to support types of users. We propose the use of, at least, the roles of developer, teacher and student. Each of these roles has specific requirements that increase the overall complexity of the system [17]. Furthermore, the insights that analytics can uncover are strongly limited by the existence of game-specific analyses and visualizations. We propose two different types of analysis, one generic, and therefore game-independent, and another customized to specific games (see Fig. 3.). In generic analyses, the data is collected and transformed in a game-independent manner. Due to the lack of game-specific input, the results of this analysis require an additional layer of interpretation by the user of the system (student, teacher, developer) when displayed. This is because automatic inference for each specific game is only possible when game-specific analysis models are provided to the system. Analysis models to create custom analyses are provided by the

teachers and game developers to allow assessment and to help the system to obtain deeper and more meaningful game-specific results.

Our system provides a default analysis that automatically obtains game-independent results from *xAPI*-formatted traces, without any additional game-specific input. Finally, as an extension of the infrastructure, custom analyses are supported via the possibility of (1) defining additional game-specific inputs in form of assessment by the teacher and (2) creating additional analysis that infer conclusions from these assessments.

B. Visualizations

One of the endpoints of the value extraction process is the creation of visualizations. For clarification, visualizations are graphical representations of the analyzed data to help the viewer on its interpretation. Thus, these visualizations can have very different formats, from two-dimensional graphics representing scores, to progress bars representing time spent per mission. This diversity of visualizations is a result of the generic purposes of *xAPI* specification. Hence, as a new *xAPI* profile is specified, new visualizations are possible representing this new source of information.

There are two approaches to creating visualizations. On the one hand, given the self-explanatory scheme of *xAPI*, it is possible to establish a series of graphs that represent (or compare) the natural meaning of the traces in a visually interpretable way. These graphs are known as *default* graphs, and can be based on one or more parameters of the analysis results (i.e. showing the player's positions over time). Normally, this type of graphs require an analysis by the interested party to extract a value from them, for example, limiting the cases of the input variables. However, once the educational model of a game is known, it is possible to create visualizations with the objective of representing metrics with educational value. These types of visualizations are known as *custom* visualizations and require the specialized and specific knowledge of the game learning model. For example, even if the player throws different traces of position, only those that occur during a treasure hunt mission are relevant to understanding whether the player has understood the hint that leads to the treasure. In general, we again advocate for

providing simple *defaults* while providing support for *custom* visualizations.

Given the previous description, next subsection explains the new possible location-based *default* and *custom* visualizations.

1) Location-Based visualizations

New visualizations are required to leverage the proposed *xAPI* location-based profile. In particular, data from the location extension from the proposed profile is most naturally displayed on a map. Map-based visualizations represent the location of the students or their behavior. Using the map as a base canvas with real-world information as background, the information can be represented using different colors, areas or graphics. Depending on the time dimension, different visualizations can be created.

When the time is not used, heat maps are recommended to represent the overall amount of activity in each location by changing the color of the active region to warmer colors. Warm colored zones will represent more activity, while colder ones represent sporadic activity. Using heat maps we provide two different *default* visualizations. The first *default* heat-map visualization represents the activity of a student in each location. In this implementation, the heat map visualization uses *OpenStreetMaps* for the back-end graphics and navigation controls, as shown in Fig. 4. The second *default* heat map visualization represents activity of location-based actions (enter, leave and look). For example, by filtering *enter* traces it will be possible to identify main entry points of an area and special of by avoidance of main entrances.

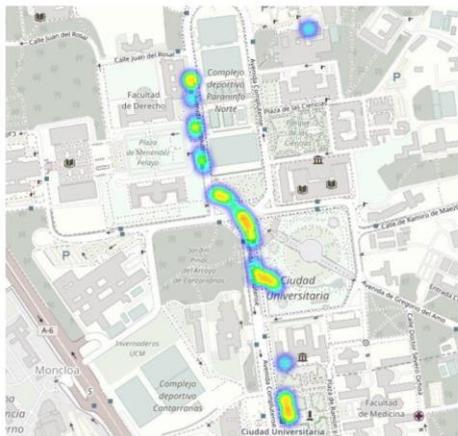


Fig 4. Heat-map of player locations from an experiment where a group of students had to explore the sports-related facilities of the Complutense University of Madrid.

When the time is used, the different locations can be interpreted as a movement with direction. Hence, route based maps can be created using lines connecting the different traced

locations. In this case, *default* visualizations represent the player gameplay route and can be filtered by time intervals.

Besides of map-based visualizations, visualizations of metrics derived from the additional location-based data sent to the server can also be created. For example, developers and instructors may wish to know when players arrive at a certain outdoor area or find a specific item in the map. Furthermore, using the additional verbs proposed in the *xAPI* location-based profile we can generate and display visualizations that describe the behavior of the students in considerable detail. For example, a game could ask its players to explore different type of points of interest within an outdoor area in a specific order. By carefully analyzing the *default* generic map visualizations and filtering by both player and time, teachers could then reconstruct each individual player's performance, annotating the order in which each point of interest was discovered.

In both visualization types, to infer game-specific conclusions about the activity (or metrics) and display it, it is necessary to create *custom* visualizations by providing additional input to the visualization. Together with a *custom* analysis, a *custom* visualization could easily extract this same information and display the results in a much more user-friendly, albeit game-specific visualization; which would however make no sense for games with entirely different goals and mechanics.

C. Assessment

The second endpoint of the analysis is assessment. Assessment complements the visualizations offering a qualitative point of view of students' learning, for example identifying when something is right or wrong. In SGs, assessment can be used as an effective tool to aid the teacher's conclusions about the students' learning [18]. This is also the case in a *Learning Analytics* infrastructure, especially if the assessment process can be automatically inferred by the system when a given set of game-specific inputs are provided by the teachers.

Game-specific information can be provided through multiple inputs to the *Learning Analytics* infrastructure. To begin, a rule-based engine is recommended to be running on top of the default infrastructure that allows teachers to automatically infer conclusions based on different specific parameters of the game. The game-specific parameters must be quantifiable to facilitate the use of the system. Examples include measuring the time spent to achieve a goal, time inside an outdoor area, or the time until a sequence of goals have been successfully achieved.

For the *xAPI* location-based profile itself it is possible to measure different *time* metrics such as the *time* spent in each area and the total completion time. Also, location-based *actions* can be assessed based on the targeted object. However, significant variables sent as game-specific input can be used to assess the students' learning. The variables must be explicitly included by the game developers as extensions to *xAPI* traces, and will generally be *string*, *numeric* or *Boolean* in nature. The assessment rules can then compare the value of the variables with their expected values to give further feedback to the teacher. Assessing on location-based actions is very relevant, as they have an extra meaning of positioning intention. For

instance, it is possible to detect when the player is exiting the game-area and notify the teacher accordingly.

Customized analysis algorithms that automatically produce high-level results can also be used to assess the students. The results generally require game-specific information, and will therefore be of little applicability to other games; but high-level results can often be displayed in simple customized visualizations automatically set up by the developer, such as bar-graphs. In more complex cases, developer must create both the customized analysis and the associated visualizations. This process does not require the intervention of the teacher and can result in higher-value insights because of the additional game-related data available to the customized algorithm that is not present in the default analysis.

Assessment is especially relevant for location-based data, which can combine locations with additional time-related information to assess the behavior over time of the students. This is also relevant to detect misconceptions and allow the use of targeted feedback to improve the learning experience [19]. For instance, if specific students spend too much time in an area where the game design did not expect such lingering, this can be easily uncovered using Heat maps in combination with behavior over time; and can allow teachers to reach out to the students and help them continue with the activity, among other possible feedback.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have described a data model designed for reporting location-related aspects of in-game player interactions in pervasive experiences to a learning analytics infrastructure. The underlying data model is proposed as an extension of a pre-existing *xAPI* (*Experience API*). Its use can cover from SGs to gamified applications, and the model can be further extended to match specific needs. We also describe the consequences and benefits of including the proposed profile in a Learning Analytics infrastructure for learning experiences, allowing their users to gain insights based on the location and behavior of students over time in outdoor areas. Additional layers of customization may provide higher-level insights as well as automatically extract conclusions based on students' learning. This is especially relevant for assessment, which is only possible once teacher-defined assessment rules have been specified.

The *xAPI* location-based profile is designed for pervasive experiences; and therefore, relies to an extent on constant internet connection where traces are sent and received correctly and in a timely manner. If the connection fails or is not present, traces should be stored locally in each device and sent to the server once it is successfully reestablished.

One of the limitations of the proposed solution, and learning analytics in general, is the fact that teachers are responsible of extracting meaning from the insights if no additional layer of customization is provided to link inputs to actual learning.

The proposed *xAPI* location-based profile has been tested in terms of functionality in a small experiment where students had to explore different sports-related facilities in an outdoor area at the Complutense University of Madrid. The experiment methodology included both pre- and post- surveys regarding

knowledge of the subject matter (sports facilities in an academic campus); and gathering of all game-generated traces as sent by player devices during the experiment. To gather these traces, the experiment used a modified version of the analytics infrastructure developed for the H2020 RAGE project; after being tested successfully, the modified infrastructure is available as open-source to any interested parties. The main goal of the experiment was to test the capacities of the standard in terms of functionality, while the game-specific assessment layer received considerably less attention. In this sense, future experiments to validate the assessment possibilities for this standard have yet to be performed. We did, however, obtain insights on the new visualizations and assessment opportunities discussed in this paper.

Experiments to further validate the functionality, usability and assessment capabilities in different contexts are currently being planned as part of the H2020 BEACONING. We wish to highlight an experiment that is, as of this writing, being developed by the *GeoMotion* team, designed to work in a web-based environment (JavaScript framework) that will be carried out in several European cities. Additional experiments are being planned by the e-UCM team to test the capabilities of the location-based model integration in the *uAdventure* platform. In this case, the implementation of the location-based tracing is built on top of the Unity environment (C# framework). We expect to obtain exhaustive feedback on the proposed solution, which will allow us to further iterate and adjust the data model, the default visualizations, and the customized assessment capabilities. Future papers will report on the educational value of the model as the assessment capabilities are validated.

We conclude that the proposed solution provides a simple location-based assessment model that can aid and support other methods, and can be used to complement, but not replace, traditional assessment methods. The potential for assessment is especially high if game-specific information is related with the educational design and used within appropriate custom analyses to gain insight and better understand student behavior.

ACKNOWLEDGMENT

This research has been partially financed by the Regional Government of Madrid [eMadrid S2013/ICE-2715], by the Ministry of Education [TIN2013-46149-C2-1-R, TIN2017-89238-R] and by the European Commission [RAGE H2020-ICT-2014-1-644187, BEACONING H2020ICT-2015-687676, Erasmus+ IMPRESS 2017-1-NL01-KA203-035259].

REFERENCES

- [1] A. de Souza e Silva, "Pokémon Go as an HRG: Mobility, sociability, and surveillance in hybrid spaces," *Mob. Media Commun.*, vol. 5, no. 1, pp. 20–23, 2017.
- [2] F. Mäyrä, "Pokémon GO: Entering the Ludic Society," *Mob. Media Commun.*, vol. 5, no. 1, pp. 47–50, 2017.
- [3] A. G. LeBlanc and J. P. Chaput, "Pokémon Go: A game changer for the physical inactivity crisis?," *Preventive Medicine*, vol. 101, pp. 235–237, 2017.
- [4] M. Tateno, N. Skokauskas, T. A. Kato, A. R. Teo, and A. P. S. Guerrero, "New game software (Pokémon Go) may help youth with severe social

- withdrawal, hikikomori," *Psychiatry Research*, vol. 246, pp. 848–849, 2016.
- [5] C. M. Chen and Y. N. Tsai, "Interactive location-based game for supporting effective English learning," in *Proceedings - 2009 International Conference on Environmental Science and Information Application Technology, ESIAAT 2009*, 2009, vol. 3, pp. 523–526.
- [6] M. Dunleavy and C. Dede, "Augmented reality teaching and learning," ... *Res. Educ. Commun.*, pp. 1–34, 2014.
- [7] Pottermore.com, "New mobile game Harry Potter: Wizards Unite coming soon," *Pottermore.com*, 2017.
- [8] K. Conley and C. Donaldson, "Gamification: The measurement of benefits," in *Gamification in Education and Business*, 2015, pp. 673–688.
- [9] T. Heinz, "Location-based Game Design Pattern Exploration Through Agent-Based Simulation," 2017.
- [10] K. C. Lim, "Using xAPI and Learning Analytics in Education," *Elearning Forum Asia*, no. June 2015, pp. 13–15, 2016.
- [11] Á. Serrano-Laguna, I. Martínez-Ortiz, J. Haag, D. Regan, A. Johnson, and B. Fernández-Manjón, "Applying standards to systematize learning analytics in serious games," *Comput. Stand. Interfaces*, vol. 50, pp. 116–123, 2017.
- [12] C. Portele, S. J. D. Cox, P. Daisey, R. Lake, and A. Whiteside, "OpenGIS® Geography Markup Language (GML) Encoding Standard," *OGC Implement. Specif.*, vol. 07–036, p. viii + 426, 2007.
- [13] H. Butler, M. Daly, A. Doyle, S. Gillies, S. Hagen, and T. Schaub, "The GeoJSON Format," Aug. 2016.
- [14] OpenStreetMaps, "OpenStreetMaps Places," 2017. [Online]. Available: <http://wiki.openstreetmap.org/wiki/Places>.
- [15] Google, "Google Places API," 2017. [Online]. Available: <https://developers.google.com/places/>.
- [16] I. J. Perez-colado, V. M. Perez-colado, M. Freire-moran, I. Martínez-ortiz, and B. Fernandez-manjon, "Integrating learning analytics into a game authoring tool."
- [17] R. Ferguson, "Learning analytics: drivers, developments and challenges," *Int. J. Technol. Enhanc. Learn.*, vol. 4, no. 5/6, p. 304, 2012.
- [18] F. Bellotti, B. Kapralos, K. Lee, P. Moreno-Ger, and R. Berta, "Assessment in and of Serious Games: An Overview," *Adv. Human-Computer Interact.*, vol. 2013, pp. 1–11, 2013.
- [19] Á. Fidalgo-Blanco, M. L. Scin-Echaluce, F. J. García-Peñalvo, and M. Á. Conde, "Using Learning Analytics to improve teamwork assessment," *Comput. Human Behav.*, vol. 47, pp. 149–156, 2015.

6.4. A Tool Supported Approach for Teaching Serious Game Learning Analytics

6.4.1. Cita completa

Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2021). **A Tool Supported Approach for Teaching Serious Game Learning Analytics**. *2021 IEEE Frontiers in Education Conference (FIE)*, 1–8. <https://doi.org/10.1109/FIE49875.2021.9637062>

6.4.2. Resumen original de la publicación

This is an Innovative Practice Full Paper. Serious Game learning analytics can provide insight to improve both the learning process and the lifecycle of games. Due to the complexity and diversity of topics involved, from game design to implementation to tracing the user, teaching game learning analytics to engineering students is challenging. We have created a teaching approach built on an integrated set of tools to minimize the boilerplate setup and configuration typically required when building game learning analytics from disparate modules. Our approach relies on the combination of an authoring environment that eases the creation of adventure and location-based games, a standards-based interaction tracker, and a cloud-based analytics framework. In this environment, students can design and implement serious games with associated analytics models from the very beginning, allowing them to experiment from early on with analytics to improve their games' lifecycles. We have piloted this approach in two university courses for two years, where students prototyped serious games and then used analytics to understand how their users interacted with their games. The use of analytics helped our students to reflect on and refine their designs. While our approach can be applied with any tools, our authoring environment and analytics tools are available as open- source projects to facilitate development, prototyping, or testing of games with analytics.

A Tool Supported Approach for Teaching Serious Game Learning Analytics

Pérez-Colado, Víctor Manuel
Dept. of Software Engineering and Artificial
Intelligence
Complutense University of Madrid
Madrid, Spain
victormp@ucm.es

Pérez-Colado, Iván José
Dept. of Software Engineering and Artificial
Intelligence
Complutense University of Madrid
Madrid, Spain
ivanjper@ucm.es

Freire-Morán, Manuel
Dept. of Software Engineering and Artificial
Intelligence
Complutense University of Madrid
Madrid, Spain
manuel.freire@fdi.ucm.es

Martinez-Ortiz, Iván
Dept. of Software Engineering and Artificial
Intelligence
Complutense University of Madrid
Madrid, Spain
imartinez@fdi.ucm.es

Fernandez-Manjón, Baltasar
Dept. of Software Engineering and Artificial
Intelligence
Complutense University of Madrid
Madrid, Spain
balta@fdi.ucm.es

Abstract—This is an Innovative Practice Full Paper. Serious Game learning analytics can provide insight to improve both the learning process and the lifecycle of games. Due to the complexity and diversity of topics involved, from game design to implementation to tracing the user, teaching game learning analytics to engineering students is challenging. We have created a teaching approach built on an integrated set of tools to minimize the boilerplate setup and configuration typically required when building game learning analytics from disparate modules. Our approach relies on the combination of an authoring environment that eases the creation of adventure and location-based games, a standards-based interaction tracker, and a cloud-based analytics framework. In this environment, students can design and implement serious games with associated analytics models from the very beginning, allowing them to experiment from early on with analytics to improve their games' lifecycles. We have piloted this approach in two university courses for two years, where students prototyped serious games and then used analytics to understand how their users interacted with their games. The use of analytics helped our students to reflect on and refine their designs. While our approach can be applied with any tools, our authoring environment and analytics tools are available as open-source projects to facilitate development, prototyping, or testing of games with analytics.

Keywords—serious games, analytics, teaching, higher education

I. INTRODUCTION

Serious Games (SGs) are videogames where the main purpose is not pure entertainment: it may be to teach, to change an attitude or behavior or to create awareness of a certain issue [1]. However, many, possibly most SGs are "black-boxes", interactive educational contents that, in the best of the cases, only report basic game information such as player's completion status or results (i.e., scores). Moreover, in black-box approaches there is no information about the player's behavior in the game, and the evaluation of the game or the measurement of its actual impact, if any, can only be performed through a

costly, complex and ad-hoc external evaluation process, for example by using pre-post questionnaires.

Game learning analytics (GLA) is the capture and analysis of in-game player interactions to gain insights that can be used to improve the learning process [2]. GLA can also be used to improve the game lifecycle based on the actual evidence obtained from user behavior, instead of relying on simple metrics such as game completion status or end-game scores. From the learning process perspective, GLA can not only prove that players attain the educational objectives of a game; it can also help to identify issues in the learning process. In fact, we can draw a parallelism between GLA and game analytics, which have been used for a long time in the game industry. In Game Analytics, the focus is on game monetization and game quality capturing player's interaction is to analyze game performance, identify game issues and maximize player retention. In GLA, the focus is on the learning process, measuring the effectiveness of the game, the performance of students, and providing feedback for the learning process, such as where and how students get stuck in the game, and even pin-pointing possible measures or improvements.

However, effective use of GLA requires understanding not only of how to generate, collect and analyze in-game player interactions; but also how to integrate GLA as part of the whole lifecycle of SGs, and how to use that information to gain insights that can improve a very broad range of aspects, from the design to the effectiveness of the selected mechanics in specific educational deployments. Due to this breadth of scope, teaching GLA to students is a challenge.

The rest of the paper is structured as follows: Section II presents related work on GLA and SG's authoring tools with built-in analytics support, Section III outlines our proposed teaching approach, Section IV describes our results and, finally Section V provides some conclusions and future lines of work.

II. RELATED WORK

We have not found specific methodologies to improve teaching of game learning analytics (GLA). We therefore present relevant work on GLA and game authoring tools to better contextualize our approach.

A. Game Learning Analytics

Understanding the SG lifecycle is key to effectively learn and apply GLA. This lifecycle includes design, creation, and deployment; and, to fully close the loop, the later analysis and decision-making process to improve the SG. Due to its educational purpose, the SG's main stakeholders are teachers and students, but researchers and developers can also benefit from GLA to guide their decision making. According to Chaudy, Connolly and Hainey [3], analytics are relevant from three perspectives: "the student's (how am I doing?), the teacher's (how are my students doing?) and the researcher's (how are the games used and are they useful?)." Chaudy [4] says that according to Powell and MacNeill [5] there are five potential purposes for LA: for learners to receive feedback on their learning and compare it with others', to predict a learner's need for additional support, for teachers to plan interventions, to improve current courses and develop new ones and for decision making at administration level. Additionally, Alonso-Fernandez et. al. consider that GLA can be used by SGs developers [6] in the game testing, validation and maintenance phases.

We can find some remarkable use-cases that have proven GLA to be useful in the game lifecycle. On its early stages, it can be used for validating games, as in the study from Calvo-Morata where GLA was used to prove the effectiveness of Conectado [7]. Another use case is to provide monitoring and custom assessment, such as in the case of the ENGAGE platform [8], where teachers used GLA to customize how to evaluate the students using the games; and it can even be used to even predict assessment itself as in [9]. These use cases, among other examples in the literature, can help solve the important problem of trust in the effectivity of games in education, making teachers less reluctant to use them [10], [11]. GLA can increase this trust by opening and providing insight into what would otherwise be black-boxes to stakeholders; and by enabling unobtrusive assessment and fostering evidence-based education.

There are two main approaches to GLA according to Chaudy, Connolly and Hainey [3]: information visualization (human-driven) and data mining (machine-driven). Alonso-Fernandez et. al. [6] noted that "the main purpose when analyzing data from SGs is assessment, [...], or visually displaying performance information". They also concluded that the application of GLA information in the game lifecycle can help reduce cost and development complexity and provide a clearer measurement of the impact of SGs.

Despite GLA being a growing field, the usage of GLA is still limited when compared to the number of SGs being developed and studied. In 2014 Chaudy, Connolly and Hainey presented a Games Learning Analytics review showing only 14 relevant studies (including games and frameworks), with the oldest one from 2011 [3]. A review from Connolly et. al. published in 2016 [12] (extended from a 2012 publication [13]) found that only a limited number of educational games provide evidence of their effects. And only a small percentage of SGs included learning

analytics at that point. However, a 2019 review from Alonso-Fernandez et. al. on SGs that applied data science [6] (visualizations, supervised and unsupervised models) to GLA showed that the number of GLA related publications has been steadily growing since 2011.

We consider that the additional complexity required to successfully apply GLA is one of the reasons for its limited usage. To use GLA it is necessary to create a Learning Analytics Model (LAM). This LAM contains the information that is needed to gain insight on the educational objectives and how they are processed, to allow later enable reporting, evaluation, and adaptation of the SG's lifecycle. GLEANER [14] proposes that this LAM can be then executed by a Learning Analytics System (LAS) that implements all the different processing steps, thus applying the LAM. The LAM and LAS are, in certain aspects, similar to Evidence Centered Design (ECD), as promoted in the ADAGE framework [15]; since ECD requires a tailored standardized process to create an analytics model (and system) that can validate game-specific competencies based on the in-game analytics tasks. However, the LAS is independent from any specific game and, in contrast to ECD, LAMs can be made flexible enough to re-use significant fragments, reducing costs and complexity compared to ECD. Reusable LAMs can be very useful, especially when coupled with educational standards and frameworks or authoring tools.

For this reason, Alonso-Fernandez et. al. [6] noted that "authors have pointed out a clear need for specific game learning analytics (GLA), where the use of standards to collect GLA data is desirable, as it allows the creation of open data sets in standard formats, such as xAPI [16], for research purposes, and simplifies results reproducibility and improvement, as well as testing of new techniques and integration of analytics as a module of larger systems". This approach is compatible with other works like the IDEFA framework [17], which works with event-stream data, using an external analytics processor with different levels of event interactions and progression marks.

The xAPI (Experience API) specification for Learning Analytics proposed by ADL is a flexible trace format that aims for flexibility and reusability [18]. This *de-facto* standard has consistently grown interest in both the educational and research communities, with most e-learning platforms supporting it or planning to support it. Traces in xAPI must include an actor, a verb, and an object, providing a flexible and expected format for very different purposes. xAPI not only covers data format aspects, but also defines a protocol to interact with a Learning Record Store (LRS) to submit and query xAPI statements [18]. Although the LRS protocol could be implemented in Learning Management Systems (LMS), it is usually implemented as an independent component, freeing the LMS not only of the task of managing the large amount of data that games can generate in real-time, but also offering a simple way to access the game results for both systematic analysis and assessment. GLA standards-based tools not only allow for a better interoperability, thus lowering costs, but also can simplify the learning analytics teaching by reusing existing components.

In conclusion, these standards can help the grow of GLA – and by some extent the acceptance of SGs – by allowing the creation of reusable LAS that can be constructed from smaller

pieces, coupled with LAMs that can be both used generically for any SG and tweaked to fit specific SG objectives. Game authors can then focus on understanding and adapting a default LAM for their own purposes, using generic tools to add GLAs to their games, and reducing not only costs and complexity, but the learning curve as the GLA community and ecosystem grow.

B. Serious Games Authoring Tools and Learning Analytics

Teaching how to author SGs is also a complex task itself, and the choice of tools is critical for the viability of the course, as students need to develop SGs to fully understand their lifecycle. Although there are plenty of tools and platforms that could be used to develop SGs, in this section we have limited our analysis to those that both simplify development and include GLA.

On one hand we can find commercial tools for developing games such as Unity and Adventure Game Studio that have been proven as effective to develop SGs [19]; however, they usually present a steep learning curve when used to teach GLA. On the other hand, we can find tools like WEEV [11], which are simpler to use but much narrower in scope, and where educators take a more active role in the development process, which is designed to help them feel more comfortable while using educational games and even perform small modifications to better fit the games in their courses. By using a simplified authoring tool, we both reduce the SG authoring learning and creation effort.

There are several commercial SG authoring tools available. For example, ITyStudio¹ is a narrative game engine for use-cases such as behavioral simulations or training scenarios; and has integrated analytics within its own platform. Moreover, some of the most popular commercial educational authoring tools focus on narrative courses creation (not being strictly SGs authoring tools), such as Articulate 360² or Adobe Captivate³. Both have support interoperability standards through SCORM [20], and thus provide very limited analytics. One remarkable case that is especially relevant to the work presented in this paper is GameSalad⁴ for education, used in courses to teach SG authoring to students. However, GameSalad does not include any GLA.

In addition, there several research and open-source game authoring tools. We found that many of those tools either did not provide support for learning analytics, such as the *Storytelling Alice* series [21] LAGARTO [22] or ARLearn [23]; or were discontinued, such as e-Adventure [24], Unigame [25] or Thinking Worlds [26]. We did find three tools that have publications in recent years and are available as open source:

- ENGAGE [27]: an open-source SG platform that supports many game genres and is focused on assessment, supporting quizzes, quests, monitoring of states and use of probabilistic models. With ENGAGE, developers create SGs that teachers can later modify through a web-based interface. In this interface, teachers can customize their assessment and learning analytics dashboards blocks including basic information, scores, interaction, and data mining.

- VEDILS [28]: a web-based application to develop virtual reality games compatible with many different external devices such as EEG sensors or hand trackers. Games can be easily designed using their web-based interface but to configure specific interactions it requires programming knowledge. According to [29] it has basic analytics support based on Google Tables and MongoDB.
- MAGIS [30]: an open-source AR focused engine that uses a domain specific language. It has a custom `@analytics` command to set up and send custom traces to an external server.

Both commercial and research tools offer simpler ways to develop SGs reducing costs and time and providing a more polished end-user experience with smoother learning curves. Commercial tools offer very controlled environments that can be key to guarantee the return on investment of the development of a SG, and provide a simple environment to monitor the learning process by limiting the assessment interoperability and potentially causing a vendor lock-in. In contrast, research and opens-source tools offer a more flexible environment that is normally paired with a more difficult analytics configuration. However, we think that ENGAGE [27] could be used for teaching GLA using our approach, as it can both simplify the game creation and offer a simple analytics and assessment management system, but we consider that a standards-based approach with xAPI is more beneficial in the long run, because it is the *de-facto* standard to represent learning analytics traces and also facilitates the usage and collaboration between special-purpose tools for each of the different aspects related to GLA (e.g. storage, analysis, visualization, etc.). However, we did not find any available integrated tools ready to use for a course on SG with GLA.

III. TEACHING GAME LEARNING ANALYTICS

We have created a teaching approach built on an integrated set of tools, to minimize the boilerplate required to define and use game learning analytics (GLA) in serious games (SGs).

A. The tools

Our approach relies on the combination of an authoring environment that eases the creation of adventure and location-based games, a standards-based interaction tracker, and a cloud-based analytics framework (Fig. 1).

The uAdventure [31] open-source tool helps users with the creation of adventure and location-based SGs by providing simple authoring metaphor that focuses on educational aspects and does not require an extensive programming background. uAdventure is an evolution of a previous validated SG authoring tool called e-Adventure [24]. uAdventure maintains e-Adventure's functionality but built on top of Unity to take the most advantage of a professional, well-supported, and technically sound platform. uAdventure provides default GLA for the created games.

¹ <https://itystudio.com/>

² <https://articulate.com/360>

³ <https://www.adobe.com/es/products/captivate.html>

⁴ <https://gamesalad.com/>

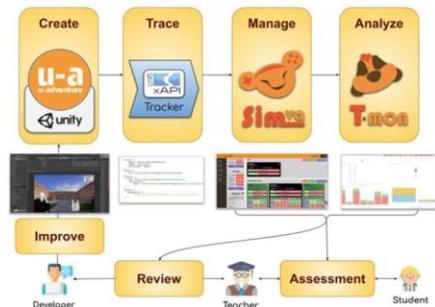


Fig. 1. The tools used in our courses that simplify the serious game (SG) lifecycle and the use of game learning analytics (GLA): uAdventure, xAPI Tracker, Simva and TxMon.

The out-of-the-box GLA uAdventure implements traces for all player related events in both narrative and location-based games and most of the system feedback. These events are traced using standard xAPI traces from the SGs profile [16] and location-based games profile [32]. The traces for narrative games include: player actions with elements, scene changes, conversation start and end, conversation choices, dialog fragment start, end and skipping and conversation choice selection; and game-state variable changes. The traces for location-based games include player interactions with real world elements such as: player movement and entering, exiting, and looking to regions or points of interest; and system events such as the discovery of elements hidden in the map. Additionally, creators can configure *completables* to obtain extra traces to determine the player progress and/or score in the game.

The *completables* system gives the creator a way to measure progress in the game of different things such as quests, stages or competencies that can be related with in-game events. To configure a completable, the user must relate the different progress milestones with specific key points of the player interaction or game state. Additionally, to the user created completables, uAdventure represents the game itself as a completable that will start when the game starts and end once the game ends or all the completables are completed and can be used to measure the overall player progress and score in the game.

To gather and analyze the different traces generated by the uAdventure game, the Simva, "SIMple-VALidator", [33] platform complements the ecosystem to help manage GLA. Simva is an open-source project, that offers an integrated solution that provides storage for xAPI traces but also contributes with SG validation by managing validation studies, including user enrollment, groups management, support for surveys for pre and/or post testing, and providing access to real-time analysis and visualization. In Simva, the user can see user progress and scores, monitor the traces and access analytics dashboards. The Simva platform also includes TxMon, "Traces Monitor", an extensible python-based data science environment that provide additional analysis.

Both tools can be easily connected by using the Simva extension for uAdventure [34], which includes a simple and resilient tracker that implements the xAPI for SGs analytics profile [16], and also simplifies the Simva setup with the use of a step-by-step wizard. Using both uAdventure and Simva, students can design and implement a game with associated analytics in uAdventure and then experiment with Simva to access actual user traces in real-time, learning how these analytics can be used to improve the lifecycles of their games.

B. Course Design

Using this combination of tools our students can focus on implementing the SGs and use the available analytics from early-on and during each of the different phases of the development of their SGs, without having to deal with low-level implementation aspects or GLA infrastructure setup.

The course structure for this approach consists in three parts: i) an introductory part, ii) a SG development part, and iii) a GLA workshop part (Fig. 2).

The introductory part lasts for five weeks (20 hours total) and introduces the students to the SGs theory and design from three perspectives: theory, practice, and experimentation. For the theory, students will learn about SGs, game genres, gamification, educational objectives, game design documents and the basics of GLA. To complement the theory, students will experience with some SGs, analyzing some SGs selected by the teacher and researching their own SGs to show them to their classmates in the class. To contrast all this theory and experimentation, the students receive some real-life advice from industry experts in several masterclasses, describing their experiences in development and use of GLA. Finally, students are divided into groups of three.

After the introduction student groups implement SGs on their own during another five weeks period (20 hours total). This small timespan is one of the reasons to choose a tool with a smooth learning to create the SGs, allowing the students to focus on the design and implementation of their SGs from an educational perspective. uAdventure lets students create games using a high-level interface based on narrative concepts. The tool also includes a user startup guide, a complete in-depth user manual with eleven examples, and several extra examples and step-by-step tutorials. In our course we introduce uAdventure from two points of view: adventure games and location-based games; each including an introduction to the game genre and a guided, 4-hour example.

During this period, students designed and implement at least two SGs prototypes which allow them to experiment with different ideas and scenarios. For this reason, we recommend creating adventure games, since they provide a proven narrative environment that can be applied to many different use cases [35]; and location-based (or augmented reality like) games, which let students develop games that use real-life elements in their teaching. However, any VR/AR/Simulation game would fit very well into this part. For each game, students must create a game design document that includes their learning objectives along with the game plot and puzzles in the specific game format fitting uAdventure's capabilities.

During the final part, students are introduced to GLA concepts and design, including GLA-related tools, in a two-hour session. They must then implement GLA in their games and test those games with their classmates using the tools during the rest of the six remaining weeks (totaling 28 hours). For the concepts, we include GLA design and xAPI traces and how these traces can be used to evaluate the games including xAPI for SGs profile [16] and for location-based games experimental profile [32]. This approach is supported because we use uAdventure as a trace generator along with the tracker and Simva as to collect traces (LRS) and manage experiments.

In uAdventure, both SG and location-based profiles are supported and traces from those kinds of interactions are tracked out-of-the-box or with minimal configuration. Since one of the most important use cases for GLA is to measure progress and scores [14] *completables* are explained and enforced in this stage. uAdventure's completable view allows students to indicate in-game milestones, such as interactions with elements, scenes, or even specific game-states, which represent progress; and optionally to link those milestones to the game's educational objectives.

Once their games are completed, including analytics, students ask their classmates to play, and then review the results using Simva as the supporting GLA framework. This exchange

allows students to access GLA traces in real-time and receive feedback from users with prior experience in SGs. Prior to using Simva to manage experiments, students receive a brief introduction to both the basic Simva interface and the uAdventure-Simva experiment wizard; and participate in a live experimental session to understand how traces are generated in the uAdventure platform. We would also like to test our student's games with actual users in real environments, but this is not always possible, since it depends heavily on schools being interested in deploying the games in their classrooms at a very short notice.

We have piloted this teaching approach in two university courses: the serious game and e-learning modules. The courses are part of the Video Game Development Degree and the MSc. in Computer Science respectively at the Complutense University of Madrid.

IV. USE CASES

A. Previous teaching experiences

Prior to developing Simva, we piloted our approach by using uAdventure for learning analytics in two different workshops (two and four hours respectively) [36]. In these workshops, students were able to configure and extract progress-related traces from their uAdventure games in both xAPI and CSV formats. However, due to the complexity of the previously mentioned configuration process, students were unable to use Learning Analytics dashboards and test their games with either colleagues or actual target users.

With Simva and the step-by-step wizard included in uAdventure, we have solved this issue reducing the complexity of the setup and providing easy access to experiment control panels, LA dashboards and participant access codes.

B. Serious Games module

Our approach has been used in the SGs module for two years. This module is taught in the first semester and in the 2020-2021 academic year had nine students divided into three different groups. It is organized in 28 two-hour sessions, including both theoretical and practical lessons, plus 6 extra unsupervised sessions with teacher support –totaling 68 hours of work.

The introductory part teaches students the basics of SGs, including an introduction, applications, gamification, and game design documents. Then, students had to analyze different cherry-picked SGs (e.g., OregonTrail, BadNews, Foldit) and research on their own SGs to present to their classmates using the CitizenScienceGames.com website as an entry point. To finalize the introduction, a total of three experts in SGs gave them masterclasses with advice, showcasing their own projects.

As previously mentioned, the SGs development part consisted in the implementation of two different SGs per group including a narrative SG and a location-based SG. This implementation phase had a total of six lessons, including: i) a theoretical uAdventure lesson with a live example implementation, ii) a networking and brainstorming session, iii) a formal presentation of their ideas, iv) two to three supported development sessions, and v) a final presentation including a prototype and a video. A total of six different games were

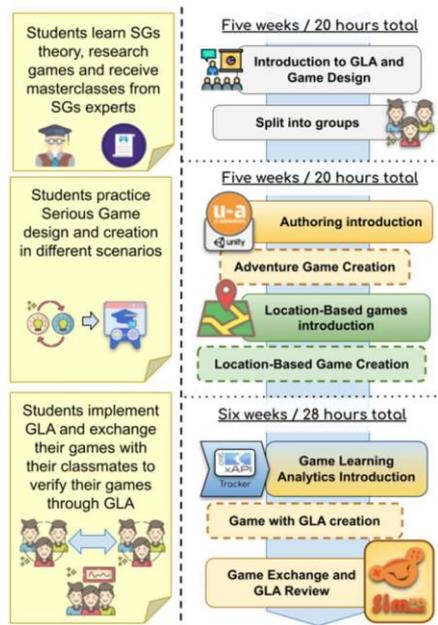


Fig. 2. Our course scheme including three periods: i) theory, research, and expert masterclasses; ii) serious game (SG) authoring of different genres; and iii) Game Learning Analytics (GLA) workshop, including, how to set them up in games, and then testing their SGs and GLA with classmates.

implemented, both narrative games and location-based games. Some of these projects were later extended for the final GLA workshop part.

In the GLA workshop part students extended three of their previous developed prototypes and included GLA using the built-in set of tools in uAdventure and the uAdventure and Simva connection. The three final projects included at least basic analytics. The use completables to track the players overall progress within the game. In general, all the groups created a one-to-one relationship between the narrative chapters and the completables. For instance, all groups added an introduction or tutorial completable and completables for the different puzzles in their games. In addition to the out-of-the-box GLA, one group implemented tailored traces to represent the different states of the puzzles using the xAPI tracker built into uAdventure. Their traces represented initial states and the interaction with each button in the different game puzzles.

The students succeeded in both building SGs with uAdventure including GLA using the Simva platform and then exchanged the games with their classmates. With the use of GLA, the students were able to validate the extent to which: 1) their classmates can complete their games, 2) they can make progress in their playthroughs and, 3) they can complete and score in each game's completables. After this workshop, students analyzed their results and made a presentation showcasing their projects, the data they obtained, and the next steps they would make to their projects. With this last presentation, students consolidated their knowledge of the usability of GLA in the game's lifecycle.

1) Narrative Game Projects

During the first SG development, students implemented purely narrative SGs. This year, the games explored three different topics: i) computational thinking, ii) sexism at work and iii) internet scam awareness for old and disabled people.

The computational thinking game was organized as an escape room. The game tells the story of a mansion where paranormal activity is occurring. The player must solve different mini games that require one or more computational thinking skills including: algorithm design, abstraction, pattern recognition and problem breakdown. It was extended with more detailed analytics for the final module project.

The game about sexism at work consists of a job interview where the player is asked frequent questions that women can be asked (such as pregnancy related ones) and in the end of the game it will be revealed that the player is a woman. As a final project it was extended with a second part represents the first day of work where the player must choose who to befriend and will receive different questions and comments during the day that will raise awareness to the player about sexism.

The last narrative game about internet scams awareness for old and disabled people and is designed as a short game that helps the player identify common signs of scam such as unexpected prize emails, you-have-a-virus ads, or mobile phone subscriptions.

2) Location-Based game projects

During the second SG development, students implemented location based SGs. The games explore three different topics: i) treasure hunts to promote computational thinking, ii) showcasing the collection of different small museums at the college and iii) civil war remains at the college campus.

The treasure hunting with computational thinking puzzles game was based on the same principles as the first computational thinking game but using narrative puzzles instead of mini games. The players had to find a hidden treasure solving four different puzzles in a nearby park.

The college museums are small, but very rich and diverse. This game proposed a gymkhana where, for each museum, players had to answer questions with clues found in the museum, embracing exploration and investigation by searching real-life locations.

The last location-based game about the civil war tells the story of both sides as they clashed in the campus, which was caught in the front line. In the game, the player will explore four different key locations of the battleground seeing pictures and learning about the civil war. To entertain players while they go from one key building to the next, the players will find characters and news from the civil war time. This game was also extended for the final project.

C. E-Learning module

The second experiment was performed over a span of 3 weeks, starting in late 2020, to students of a MSc. course on e-Learning. The 7 enrolled students were introduced to the uAdventure game authoring tool, asked to design a simple conversation-driven adventure game on a topic of their choice, and requested to design a full experiment, including in-game analytics, to validate their games.

This section of the course encompassed 6 sessions in total, with two sessions per week. Weekly sessions were half online, introducing new concepts for 1 hour, and half face-to-face, dedicated to practical tasks, with students working and presenting in pairs using laptops for 2 hours. The sessions in each week were structured as follows: Session 1 introduced SGs and adventure games, with hands-on tutorial creating a serious game using uAdventure. Homework: build a minimal conversational game on a topic of your choice. Session 2 required students to design experiments in SGs, with a Q&A session to iron out problems in the initial games. Homework: design pre-post questionnaires on a topic of your choice to validate your game on that topic. Finally, session 3 was focused on analytics in SGs, connecting the games to Simva-enabled experiments that could be carried out over the winter break. Homework: test each other's games and propose analytics dashboards on which to analyze results from your own games.

Although technical problems and time constraints prevented us from carrying out the experiments envisioned in the 3rd week, the students of the course, divided into 4 groups (3 pairs and one student working on his own), succeeded in building (very simple) analytics-enabled conversational adventure games to teach concepts in 4 different topics, with accompanying experimental designs that were readily uploaded to Simva. The

students did not have to learn how to integrate analytics with their games, configure an analytics server to receive and process them, or worry about identifying game players against an analytics server – as all these steps were automated by Simva; yet the steps were easy to inspect once configured. In this sense, the teacher responsible for the course, which is also one of the authors of the present paper, plans to continue using uAdventure + Simva in future courses.

V. DISCUSSION AND CONCLUSIONS

Teaching serious game analytics can be a challenge due to the amount of knowledge to fit in the courses. By using a tool assisted approach, we simplified much of the boilerplate-but-complex technical aspects allowing for students to focus on the game creation and the use of analytics to better understand the game lifecycle from its early design stages until their testing stages. Automating non-essential parts of a course can greatly improve usability and satisfaction; this approach is especially relevant in courses where programming is needed but not required to understand the main concepts, and it is frequently used in AR and Machine Learning course.

Game learning analytics (GLA) data allows student to contrast the game results and improve it in the different phases. Our results show that students were able to create serious games and use GLA to improve the game quality. For instance, finding educational design mistakes, implementation issues or unexpected user behaviors that hinder game applicability but that were not possible to identify during the quality assurance phase. Therefore, the use of analytics helped our students to reflect on and refine their designs.

Our authoring environment and tracking tools are open-source and freely available for developing, prototyping, or testing games with analytics (at github.com/e-ucm). They can be freely used in other similar courses and even beyond university teaching. Both our data tracker and our analytics framework can be integrated with other authoring tools and games, reducing the costs of providing GLA support to other authoring tools.

ACKNOWLEDGMENT

This work has been partially funded by Regional Government of Madrid (eMadrid S2018/TCS4307, co-funded by the European Structural Funds FSE and FEDER), by the Ministry of Education (TIN2017-89238-R, PID2020-119620RB-I00), by MIT-La Caixa (MISTI program, LCF/PR/MIT19/5184001) and by Telefonica-Complutense Chair on Digital Education and Serious Games.

REFERENCES

- [1] T. Susi, M. Johansson, and P. Backlund, "Serious Games – An Overview," *Elearning*, vol. 73, no. 10, p. 28, 2007, doi: 10.1.1.105.7828.
- [2] M. Freire, Á. Serrano-Laguna, and B. Iglesias, *Game learning analytics: Learning analytics for serious games*. 2016. doi: 10.1007/978-3-319-17727-4_21-1
- [3] Y. Chaudy, T. Connolly, and T. Hainey, "Learning Analytics in Serious Games: a systematic review of literature," in *European Conference in the Applications of Enabling Technologies*, Jun. 2014, no. November, [Online]. Available: https://www.researchgate.net/publication/268146930_Learning_Analytics_in_Serious_Games_a_Review_of_the_Literature.
- [4] Y. Chaudy, "An Assessment and Learning Analytics Engine for Games-based Learning An Assessment and Learning Analytics Engine for Games-based Learning Yaëlle Chaudy Borghini Thesis submitted in partial fulfilment of the requirements of the University of the West of Sc.," no. December 2015, 2016, doi: 10.13140/RG.2.1.4932.5040.
- [5] S. Powell and S. MacNeill, "Institutional Readiness for Analytics," vol. 1, no. 8, pp. 1–11, 2012, [Online]. Available: <http://publications.cetis.ac.uk/wp-content/uploads/2012/12/Institutional-Readiness-for-Analytics-Vol1-No8.pdf>.
- [6] C. Alonso-Fernández, A. Calvo-Morata, M. Freire, I. Martínez-Ortiz, and B. Fernández-Manjón, "Applications of data science to game learning analytics data: A systematic literature review," *Comput. Educ.*, vol. 141, no. June, p. 103612, 2019, doi: 10.1016/j.compedu.2019.103612.
- [7] A. Calvo-Morata, C. Alonso-Fernández, M. Freire, I. Martínez-Ortiz, and B. Fernández-Manjón, "Creating awareness on bullying and cyberbullying among young people: Validating the effectiveness and design of the serious game Conectado," *Telemat. Informatics*, vol. 60, 2021, doi: 10.1016/j.tele.2021.101568.
- [8] C. Thomas and C. Yaëlle, "Specification and evaluation of an assessment engine for educational games: Integrating learning analytics and providing an assessment authoring tool," *Entertain. Comput.*, vol. 30, no. July, pp. 1–16, 2019, doi: 10.1016/j.entcom.2019.100294.
- [9] C. Alonso-Fernández, I. Martínez-Ortiz, R. Caballero, M. Freire, and B. Fernández-Manjón, "Predicting students' knowledge after playing a serious game based on learning analytics data: A case study," *J. Comput. Assist. Learn.*, vol. 36, no. 3, pp. 350–358, 2020, doi: 10.1111/jcal.12405.
- [10] T. Scheneider and R. Lemos, "Use of Learning Analytics Interactive Dashboards in Serious Games," *Int. J. Innov. Educ. Res.*, vol. 8, no. 3, pp. 150–174, 2020, doi: 10.31686/ijier.vol8.iss3.2220.
- [11] E. J. Marchiori, J. Torrente, Á. Del Blanco, P. Moreno-Ger, P. Sancho, and B. Fernández-Manjón, "A narrative metaphor to facilitate educational game authoring," *Comput. Educ.*, vol. 58, no. 1, pp. 590–599, 2012, doi: 10.1016/j.compedu.2011.09.017.
- [12] E. A. Boyle *et al.*, "An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games," *Comput. Educ.*, vol. 94, pp. 178–192, 2016, doi: 10.1016/j.compedu.2015.11.003.
- [13] T. M. Connolly, E. A. Boyle, E. MacArthur, T. Hainey, and J. M. Boyle, "A systematic literature review of empirical evidence on computer games and serious games," *Comput. Educ.*, vol. 59, no. 2, pp. 661–686, 2012, doi: 10.1016/j.compedu.2012.03.004.
- [14] J. B. Hauge *et al.*, "Implications of learning analytics for serious game design," *Proc. - IEEE 14th Int. Conf. Adv. Learn. Technol. ICALT 2014*, pp. 230–232, 2014, doi: 10.1109/ICALT.2014.73.
- [15] R. Halverson and V. E. Owen, "Game-based assessment: An integrated model for capturing evidence of learning in play," *Int. J. Learn. Technol.*, vol. 9, no. 2, pp. 111–138, 2014, doi: 10.1504/IJLT.2014.064489.
- [16] Á. Serrano-Laguna, I. Martínez-Ortiz, J. Haag, D. Regan, A. Johnson, and B. Fernández-Manjón, "Applying standards to systematize learning analytics in serious games," *Comput. Stand. Interfaces*, vol. 50, pp. 116–123, Feb. 2017, doi: 10.1016/j.csi.2016.09.014.
- [17] V. E. Owen and R. S. Baker, "Fueling Prediction of Player Decisions: Foundations of Feature Engineering for Optimized Behavior Modeling in Serious Games," *Technol. Knowl. Learn.*, vol. 25, no. 2, pp. 225–250, 2020, doi: 10.1007/s10758-018-9393-9.
- [18] L. Lindert and B. Su, "The evolution of SCORM to Tin Can API: Implications for instructional design," *Educ. Technol.*, vol. 56, no. 4, pp. 44–46, 2016.
- [19] A. R. Cano, B. Fernandez-Manjon, and A. J. Garcia-Tejedor, "Downtown, a subway adventure: Using Learning analytics to improve the development of a learning game for people with intellectual disabilities," *Proc. - IEEE 16th Int. Conf. Adv. Learn. Technol. ICALT 2016*, no. 2017, pp. 125–129, 2016, doi: 10.1109/ICALT.2016.46.
- [20] A. Bakhoui, R. Dehbi, M. T. Lti, and O. Hajoui, "Evolution of standardization and interoperability on E-learning systems: An overview," *2017 16th Int. Conf. Inf. Technol. Based High. Educ. Training, ITHET 2017*, 2017, doi: 10.1109/ITHET.2017.8067789.

- [21] W. Dann, D. Cosgrove, D. Slater, D. Culyba, and S. Cooper, "Mediated transfer: Alice 3 to Java," *SIGCSE'12 - Proc. 43rd ACM Tech. Symp. Comput. Sci. Educ.*, pp. 141–146, 2012, doi: 10.1145/2157136.2157180.
- [22] L. F. Maia *et al.*, "LAGARTO: A Location based Games Authoring Tool enhanced with augmented reality features," *Entertain. Comput.*, vol. 22, no. May, pp. 3–13, 2017, doi: 10.1016/j.entcom.2017.05.001.
- [23] S. Ternier, R. Klemke, M. Kalz, P. Van Ulzen, M. Specht, "AR Learn : Augmented reality meets augmented virtuality," *Journal of Universal Computer Science - Technology for learning across physical and virtual spaces*, 18(15), 2143-2164
- [24] J. Torrente, Á. Del Blanco, E. J. Marchiori, P. Moreno-Ger, and B. Fernández-Manjón, "<-Adventure>: Introducing Educational Games in the Learning Process," in *IEEE Education Engineering (EDUCON) 2010 Conference*, 2010, pp. 1121–1126, doi: 10.1109/EDUCON.2010.5493056.
- [25] Y. Barlas and V. G. Diker, "A Dynamic Simulation Game (UNIGAME) for Strategic University Management," *Simul. Gaming*, vol. 31, no. 3, pp. 331–358, Sep. 2000, doi: 10.1177/104687810003100302.
- [26] J. Higdon and S. Miller, "Educational Gaming for the Rest of Us: Thinking Worlds and WYSIWYG Game Development," 2009, [Online]. Available: <https://www.learnlib.org/p/32482/>.
- [27] Y. Chaudy and T. Connolly, "Specification and evaluation of an assessment engine for educational games: Empowering educators with an assessment editor and a learning analytics dashboard," *Entertain. Comput.*, vol. 27, no. September 2017, pp. 209–224, 2018, doi: 10.1016/j.entcom.2018.07.003.
- [28] J. M. Mota, I. Ruiz-Rube, J. M. Doderó, and I. Arnedillo-Sánchez, "Augmented reality mobile app development for all," *Comput. Electr. Eng.*, vol. 65, pp. 250–260, 2018, doi: 10.1016/j.compeleceng.2017.08.025.
- [29] E. Lovos, I. Basciano, E. Gil, and C. Sanz, "La Producción de Juegos Serios Móviles . Posibilidades y Desafíos para el Docente de Nivel Superior," in *Congreso Argentino de Ciencias de la Computación - CACIC 2020*, 2020, pp. 144–153, [Online]. Available: <http://sedici.unlp.edu.ar/handle/10915/114077>.
- [30] E. C. E. Vidal, J. F. Ty, N. R. Caluya, and M. M. T. Rodrigo, "MAGIS: mobile augmented-reality games for instructional support," *Interact. Learn. Environ.*, vol. 27, no. 7, pp. 895–907, Oct. 2019, doi: 10.1080/10494820.2018.1504305.
- [31] V. M. Perez-Colado, I. J. Pérez-Colado, M. Freire-Morán, I. Martínez-Ortiz, and B. Fernández-Manjón, "UAdventure: Simplifying narrative serious games development," in *Proc. - IEEE 19th Int. Conf. Adv. Learn. Technol. ICALT 2019*, 2019, doi: 10.1109/ICALT.2019.00030.
- [32] V. M. Perez-Colado, D. C. Rotaru, M. Freire, I. Martínez-Ortiz, and B. Fernández-Manjón, "Learning analytics for location-based serious games," in *2018 IEEE Global Engineering Education Conference (EDUCON)*, Apr. 2018, vol. 2018-April, pp. 1192–1200, doi: 10.1109/EDUCON.2018.8363365.
- [33] I. J. Perez-Colado, A. Calvo-Morata, C. Alonso-Fernández, M. Freire, I. Martínez-Ortiz, and B. Fernández-Manjón, "Simva: Simplifying the scientific validation of serious games," *Proc. - IEEE 19th Int. Conf. Adv. Learn. Technol. ICALT 2019*, pp. 113–115, 2019, doi: 10.1109/ICALT.2019.00033.
- [34] I. J. Perez Colado, V. M. Perez Colado, I. Martínez Ortiz, M. Freire Moran, and B. Fernandez Manjon, "Simplifying serious games authoring and validation with uadventure and SIMVA," *Proc. - IEEE 20th Int. Conf. Adv. Learn. Technol. ICALT 2020*, pp. 106–108, 2020, doi: 10.1109/ICALT49669.2020.00039.
- [35] M. D. Dickey, "Game Design Narrative for Learning: Appropriating Adventure Game Design Narrative Devices of Interactive Learning Environment," *Educ. Technol. Res. Dev.*, vol. 54, no. 3, pp. 245–263, 2006, doi: 10.1007/s11423-006-8806-y.
- [36] V. M. Pérez-Colado, I. J. Pérez-Colado, M. Freire-Morán, I. Martínez-Ortiz, and B. Fernández-Manjón, "Simplifying the creation of adventure serious games with educational-oriented features," *Educ. Technol. Soc.*, vol. 22, no. 3, pp. 32–46, 2019.

6.5. Extending Narrative Serious Games Using Ad-Hoc Mini-games

6.5.1. Cita completa

Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021b). **Extending Narrative Serious Games Using Ad-Hoc Mini-games**. In W. Zhou & M. Yi (Eds.), *Advances in Web-Based Learning – ICWL 2021: 20th International Conference, ICWL 2021*, Macau, China, November 13–14, 2021, Proceedings (pp. 63–74). https://doi.org/10.1007/978-3-030-90785-3_6

6.5.2. Resumen original de la publicación

Narrative games have proven their effectiveness as serious games in different domains and for different purposes, such as promoting learning or increasing user awareness. However, there are many situations where the narrative model falls short and can benefit from being extended with puzzles or mini-games to afford more flexibility or explore non-narrative mechanics more adequate to the task at hand. In our uAdventure game authoring environment, a narrative serious game provides support for the driving narrative, managing stories, conversations, and other narrative elements, together with an integrated game learning analytics support. We present how we have extended uAdventure to support the inclusion of mini-games within a host narrative game, while allowing hosted mini-games to access uAdventure services through a streamlined interface. As a case study, we describe how this extension has been used by students learning serious games development to create narrative games with ad-hoc puzzles that use alternate mechanics to achieve game-specific goals.



Extending Narrative Serious Games Using Ad-Hoc Mini-games

Víctor M. Pérez-Colado^(✉) , Iván J. Pérez-Colado , Iván Martínez-Ortiz ,
Manuel Freire-Morán , and Baltasar Fernández-Manjón 

Department of Software Engineering and Artificial Intelligence, Complutense University of Madrid, C/ Profesor José García Santesmases, 9, 28040 Madrid, Spain

victormp@ucm.es

Abstract. Narrative games have proven their effectiveness as serious games in different domains and for different purposes, such as promoting learning or increasing user awareness. However, there are many situations where the narrative model falls short and can benefit from being extended with puzzles or mini-games to afford more flexibility or explore non-narrative mechanics more adequate to the task at hand. In our uAdventure game authoring environment, a narrative serious game provides support for the driving narrative, managing stories, conversations, and other narrative elements, together with an integrated game learning analytics support. We present how we have extended uAdventure to support the inclusion of mini-games within a host narrative game, while allowing hosted mini-games to access uAdventure services through a streamlined interface. As a case study, we describe how this extension has been used by students learning serious games development to create narrative games with ad-hoc puzzles that use alternate mechanics to achieve game-specific goals.

Keywords: Serious games · Narrative games · Mini-games · Game authoring · Game analytics

1 Introduction

The use of video games in teaching has attracted great interest from teachers and researchers for their ability to improve the interest and retention of players by keeping them motivated [1, 2]. As their primary goal is not that of entertainment, such games are often called serious games; for example, players may face a series of challenges to improve knowledge or cognitive skills [3]. Genres such as action games allow players to improve their dexterity and reflexes, while simulation games use realistic environments where players can test and apply their knowledge without being exposed to the risks of real environments [4].

Due to the different characteristics of serious games, for example to their genre, the requirements for their design and development also vary. For example, simulation games, which are very effective for specific learning, are also very costly to develop due to the high level of detail required to emulate the target domain. Therefore, when

© Springer Nature Switzerland AG 2021
W. Zhou and Y. Mu (Eds.): ICWL 2021, LNCS 13103, pp. 63–74, 2021.
https://doi.org/10.1007/978-3-030-90785-3_6

developing a serious game, the genre chosen is key for both its effectiveness and educational applicability as well as for its development feasibility in terms of requirements and costs. While development costs can sometimes be reduced by using pre-built assets (such as those available at Unity's Asset Store), such assets will require significant effort and expertise to customize for any sufficiently-specific domain.

Among the different types of games, narrative games offer a good balance between simplicity and flexibility, as they allow players to play roles and perform meaningful tasks in an environment not excessively expensive to create, but rich enough to achieve immersion. In particular, the "point and click" subgenre of narrative games [5] often relies on interactive conversations to deliver content, using characters who play different roles in the story; together with different objects and interactions to solve logical puzzles where players can apply their knowledge and learn from their mistakes [6]. For this reason, narrative games have been applied in very different domains and with very different purposes (e.g., learning, awareness).

However, narrative mechanics also have limitations. For example, they are not suitable for the development of other types of cognitive skills, such as pattern recognition, reflexes, or physical skills in general, which can be taught through mechanics present in other genres (e.g., action games to develop reflexes) [7]. Despite the possibility of simulating the mechanics from other genres, the adaptation needed to implement them is complex and costly. For example, developers that intend to use a narrative engine to develop a visual puzzle would need to manually analyze the different states of the puzzle and configure a series of narrative elements (and their corresponding graphical representations) to represent the state of the puzzle. This is impractical, complex and, above all, unnatural for a game developer; and is the case of commercial narrative content authoring tools such as Adobe Captivate¹, Articulate 360² and ITyStudio³.

We describe an extensible model that allows enriching narrative games with other mechanics, such as puzzles; and an implementation of this model in a tool to demonstrate the feasibility of this approach. For the integration of mini-games in narrative games we have taken as a basis uAdventure [8], an open-source authoring tool for serious narrative games built on the Unity game platform. The challenge is to incorporate other mechanics into uAdventure's narrative game model, which increases its versatility and allows new capabilities to be addressed, all without significantly increasing the difficulty of game creation. However, the inclusion of these new mechanics through the mini-games requires game creators to know how to program the mini-games. This approach has been tested in two case studies with students within the serious games course of the Videogame Development degree at Complutense University of Madrid.

The following sections of this article describe the mini-game integration model, the use cases of this integration model, lessons we have learned from our experiences, and finally conclusions and future lines of work.

¹ <https://www.adobe.com/es/products/captivate.html>.

² <https://articulate.com/360>.

³ <https://itystudio.com/>.

2 Extending the uAdventure Narrative Model Through Ad-Hoc Mini-games

uAdventure is a tool that simplifies the prototyping and development of games for non-experts, such as most teachers, as no programming is required. The authoring metaphor with uAdventure is based on creating a series of scenes, where the author will use the narrative for the player to develop a role through the game. Players learn new content or develop additional awareness by interacting with other players, participating in in-game conversations, or solving logical puzzles. For situations where it is too complex to adapt the content to the narrative model, uAdventure also provides other types of scenes where it is possible to include videos and interactive animations that can be interspersed with the main narrative content [8].

Furthermore, uAdventure includes other educational-specific features, such built-in assessment of learners and the capability to deploy and integrate games with other educational platforms by means of widely used educational standards [9]. In addition, player actions can be recorded and used internally in the game (through a mechanism of variables and conditions to change game behavior), as well as externally through the built-in learning analytics mechanism. Use of built-in learning analytics allows creators to not only access the final results at the end of the game (e.g. score, completeness, progress), but also to analyze the full sequence of actions that each player took to achieve that score, with minimal work required from game authors [10, 11].

Despite the benefits of uAdventure for most developers, the point-and-click narrative model it implements may fall short of the requirements of some games. For example, the narrative model ill-suited for puzzles with a very large number of states. For example, in a Rubik's Cube representation, where pattern recognition and spatial recognition skills can be learned, a direct three-dimensional representation would be much simpler than a representation through narrative elements. Our proposal provides support for including self-contained, alternate game mechanics that are more appropriate for the type of skills to be worked on in those parts of the serious narrative game that require it. The challenge is to achieve the integration of those alternate mechanics within the narrative game model of uAdventure, therefore increasing its versatility and allowing new skills to be addressed while avoiding a large increase in the difficulty of creating such games. While the integration of these new mechanics would certainly require a programming effort, once created, our goal is for the mechanics to be freely reusable by other authors without needing to program them themselves.

The process of adding new mechanics is done by programming mini-games integrated into uAdventure [12, 13]. The mini-games are launched from uAdventure, which delegates full control of the game execution to them (Fig. 1). With this model it is possible, for example, to develop an action mini-game that can read the state of the game to adapt its difficulty to the narrative moment in which it is found and use conversations during the mini-game itself to explain it or as part of its operation. The mini-games are developed as new components integrated with uAdventure, so mini-game authors must know how to program for the Unity platform. In this regard, mini-games can exploit all the capabilities of Unity to create the new mechanics, but they must also be integrated with the services offered by uAdventure: the core of the game; the narrative engine, and the learning analytics engine.

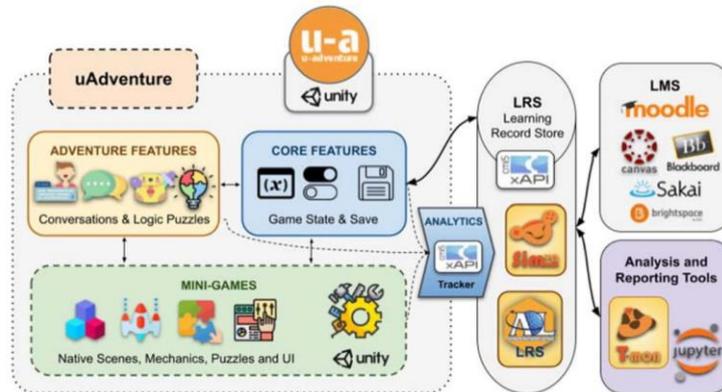


Fig. 1. Mini-game integration into the uAdventure game authoring tool. Mini-games allow additional game mechanics to be added to uAdventure, and can make use of uAdventure-provided features such as conversations, persistence, or xAPI-SG analytics.

The core game system is the system in charge of maintaining and managing the global state of the game. Using it, the mini-game can inspect the current state of the main game as a result of the previous actions that the player has carried out up to this moment. This allows the mini-game to adapt to its context of use within the current session; for example, by adjusting its own difficulty, or the style or visibility of certain features. In addition, mini-games can also save significant actions performed by players to the game state. In summary, through the game state, it is possible to communicate bidirectionally with uAdventure and thus read the different parameters that will define the configuration of the game; and write in uAdventure those parameters that are relevant once the main narrative game resumes.

uAdventure's narrative engine provides support for narrative content and interaction as well as corresponding visual effects. Through the narrative engine it is possible to launch conversation sequences and customizable options without leaving the mini-game. The mini-game developer has access to this uAdventure functionality, for example, to show tutorials or to pause the mini-game when it reaches a milestone, displaying a conversation at that moment. In addition, it allows scenes to be changed, and different events to be triggered once the mini-game ends, continuing with the flow of the game while taking into account the results of the mini-game. Finally, the learning analytics engine provides mini-game developers with a means to generate logs of the player's progress through the mini-game, which can be stored locally or sent to a remote storage location for further analysis. The main component of the analytics engine is the xAPI tracker [14] which provides a high-level interface to hide the details of the xAPI-SG specification and allows developers who are not experienced in analytics to generate traces to analyze the player's progress through the mini-games. Using this feature, a mini-game developer can, for example, generate traces that describe the interactions of the player with the interactive elements present in the mini-game.

From a more technical point of view, the uAdventure mini-game feature is based on Unity's scene mechanism. The developer of a mini-game must create a Unity scene into which to place the mini-game. In it, by means of the various features available in Unity (such as the physics engine, lighting, or its user interface system) and its scripting interfaces, the scene and its behavior will be shaped, thus defining its concrete mechanics and the layout of the mini-game. For example, in a serious game involving perception evaluation, the user could use the physics engine to create a realistic environment in which different particles move more naturally. Once the scene is created and an identifier is assigned, the developer will be able to reference the scene from the main game as any other native uAdventure scene, thus allowing the application of the uAdventure's authoring metaphor that does not require programming.

This trade-off allows advanced users (programmers) using uAdventure to use native Unity scenes without great effort; while offering non-expert users the possibility of reusing previously-developed mini-games within their own uAdventure games. Thanks to this extensibility model, it is possible to create an ecosystem where programmers and teachers can collaborate to improve the educational experience through the resulting game. In addition, thanks to uAdventure's built-in services provided to the mini-game, it is possible to create parameterized mini-games so that the experience is not unique to specific games and can be customized by teachers using uAdventure's visual editor.

Because this integration architecture is quite flexible, it can be adapted to all kinds of situations with different requirements. Although mini-games do not need to comply with any particular structure, Fig. 2 describes our recommended structure for mini-games, focusing on allowing the mini-game to be executed in a subrogated way to the host game, thus allowing the game to be decoupled from the main game and thus promoting a simpler distribution of roles in its development. In order to establish an independent execution of the mini-game, the input and output parameters must be defined beforehand. The input of parameters should be defined before launching the mini-game through the visual programming system of uAdventure, and will be read at the beginning of the execution of the mini-game through the core; while the output of parameters will become available at the end of the mini-game, to be incorporated into the state of the game for later use.

While the mini-game is running, it should not manipulate game variables, thus avoiding collateral effects. In order to return control from the mini-game to the main game, we have introduced a new special command that returns to the previous scene, available using uAdventure's visual programming system. Although mini-games have access to the narrative engine, and can execute conversations and provide other visual feedback, it should rely on this command to switch scenes. Finally, in terms of learning analytics, the mini-game should generate its specific traces by calling the analytics tracker directly. Mini-games developed following these recommendations act as uAdventure components and thus are reusable across games.

3 Pilots of uAdventure and Ad-Hoc Minigames

The mini-game integration model in uAdventure has been tested with students in two case studies within the Serious Games course of the Video Game Development Degree at the Faculty of Computer Science of the Complutense University of Madrid. In other words,

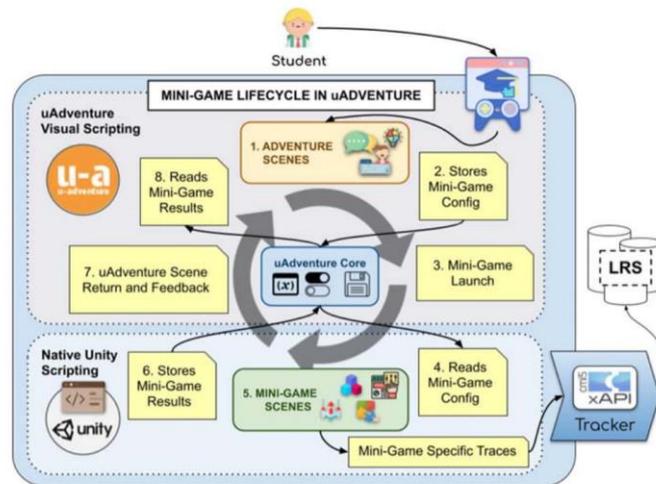


Fig. 2. Mini-game lifecycle within uAdventure. The transition between the uAdventure and mini-game scenes allow exchange of inputs and outputs, as well as the use of the tracker for analytics. The launching process of the mini-game is customizable through uAdventure’s visual editor. The actual mechanics of mini-games must be developed using Unity scripts

it has been tested with students who know how to program games, although they have limited experience in the creation of narrative games. The first case was a pilot test with the objective of addressing a need for a more flexible environment in uAdventure; and analyzed the strengths and weaknesses of the proposed integration model. The second case is more complex in terms of educational goals and narrative depth, using multiple mini-games as part of an escape room-like narrative game whose objective is to teach competencies related to computational thinking.

3.1 Pilot: Serious Game for Airport Protocols Training

In the 2019–20 academic year, students used uAdventure to prototype and create narrative games, but since they knew how to program with Unity, they were also asked to enhance these games with other mechanics and visual effects. To do so, they used the proposed mini-game system, which provided them with the flexibility of the Unity engine without losing the advantages of uAdventure.

One of the games designed by the students aimed to teach different protocols typically found in air travel (Fig. 3). More specifically, it had three educational objectives: to learn to differentiate between the objects that can be carried in the flight cabin and those that must be carried in the cargo hold; to examine the different pieces of documentation required on a flight and the different problems associated with these documents; and to remind flyers of the different safety rules that must be followed during flight.

To address the first objective, the students chose to develop a mini-game in which, through the use of “drag and drop” mechanics, players had to overcome the challenge of

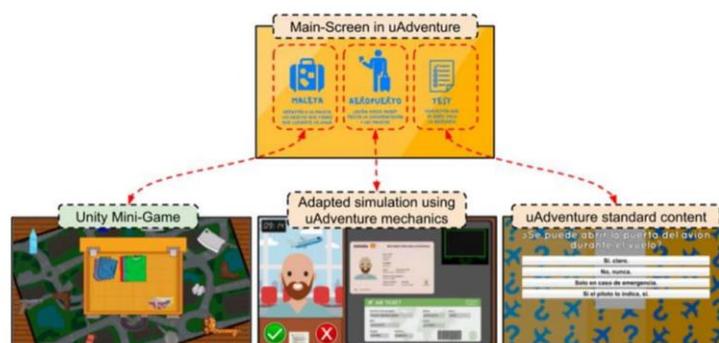


Fig. 3. Integration of different mechanics in uAdventure in the serious game “Vuela” (Fly) to teach flying protocols

being able to make the most of the space in the suitcase, by including only the essential (and valid) objects for cabin carry-ons. In this case, the mini-game not only taught players to differentiate objects by their visual appearance, but also took advantage of the limited space mechanics to infer which objects should actually travel in the checked luggage due to size constraints.

Regarding the educational goal of teaching the rules related to documentation, the developers opted, utilizing the different narrative resources of uAdventure, to create a simulation in which players temporarily take on the role of an officer who must review the different documents of would-be passengers. This mechanic is particularly interesting, as it represents the case in which the mechanics of a narrative engine are used to develop a limited simulation that is at the limit before being necessary a mini-game due to the complexity of its development and the large number of elements involved in its different states. In this case, the uAdventure mini-game used a simulation in which the face and shoulders of passengers are visible to one side and, to the other side, the passenger’s documentation appears. Players must then decide whether or not each passenger can board the plane, identifying possible errors or contradictions in the documentation and thus favoring deductive learning. Finally, the educational objective related to safety rules was addressed through a questionnaire built using uAdventure’s conversation system.

Despite being a first proof of concept of mini-game integration, the students were able to successfully use mini-games to develop their own mechanics extending uAdventure’s narrative model. Additionally, during development, multiple shortcomings were identified, including lack of details regarding the different APIs to take advantage of uAdventure’s mechanics, the communication flow between the mini-game and the main game, and the integration of analytics from the mini-games. As a result, documentation was significantly improved in time for the second round of testing.

3.2 Developing an Escape Room Game for Promoting Computational Thinking

During the 2020/2021 academic year, the enhanced version of uAdventure was used to create narrative games, while placing more emphasis on the importance of learning

analytics. Unlike the previous year, as a lesson learned, students received additional guidance on the use of the different uAdventure systems within mini-games, including two-way communication, the use of the narrative engine, and the use of the tracker.

The uAdventure mini-games model allowed a group of students to develop an escape room-themed narrative game named “The Paranormal Mansion”, teaching competencies related to computational thinking. To this end, the game focuses on the investigation of paranormal events in a mansion and intersperses different mini-games focused on developing specific computational thinking skills, including decomposition, pattern identification, abstraction, and algorithms. These skills would have been either not possible or practical to convey using only standard uAdventure mechanics.

The mini-games (Fig. 4) address different topics and visual styles and use Unity’s three-dimensional capabilities, visual effects, and user interfaces. Also, all mini-games have three levels of difficulty. The implemented mini-games are: shapes and colors (3), which develops abstraction by representing numbers using geometric shapes and debugging by requiring players to use shapes to create a specific sequence of numbers; ethical hacking (4), where players must combine a set of cards with color bars to generate a new card with a specific pattern of color bars, by first identifying how cards combine their patterns and later using trial and error until they achieve the correct pattern; crazy rings (5), where players must rotate several groups of rings until the colored flags of each ring are aligned, by identifying the rotation patterns and decomposing the problem into smaller problems associated with each ring; electrician (6), where players illuminate a series of boxes by abstracting the pattern of changes that occur when interacting with each box, exercising both pattern identification and abstraction; and finally number maze (7), where players must input a path of numbers that connects the blue box to the red box by following a series of rules, while both developing both algorithms that adhere to the rules and abstraction by representing the path with numbers.



Fig. 4. “The Paranormal Mansion”, a narrative game with integration of mini-games for the development of computational thinking skills. Images (1) and (2) show game scenes created with uAdventure in the point and click style with conversations and options. The rest of the images are mini-games: (3) shapes and colors (4) ethical hacking, (5) crazy rings, (6) electrician, (7) number maze. The image (8) shows the score obtained from one to three stars.

The interaction mechanisms used in the puzzles and their high number of states (and steps) needed to solve them, were the motivations for the students to develop them as mini-games. For example, in the crazy rings game (5), due to the large number of buttons and turning possibilities, the titular rings can reach a large number of different configurations, making it extremely complicated to simulate this behavior exclusively through narrative mechanics.

In this case, in addition to using its capabilities as an engine, the students take advantage of the different uAdventure systems so that, at the end of the execution of the mini-games, they can store a score which ranges from 1 to 3 stars. This score is then used in the credits screen and in the different analytics that the game generates. In addition, each time a mini-game ends, students made use of uAdventure’s narrative engine to create appropriate conversations.

During the development of this game and its associated mini-games, students used uAdventure’s learning analytics system to create and send traces that would allow the progress of players through mini-games to be examined. Their goal was to detect possible problems in the difficulty of the games and the progress of levels, and to find possible improvements for future iterations. To this end, the students sent enough traces from the mini-games to be able to recreate the players’ actions during the completion of the mini-games; as well as to measure the progress/completeness of the mini-games themselves.

After the game was played by 8 users, all the traces generated by the mini-games were analyzed in order to compare their complexity based on the number of steps required to complete each of them. In Fig. 5, we can observe how the electrician mini-game was particularly complex, closely followed by the crazy rings mini-game and the one on ethical hacking. These results corroborate feedback from the players, who had already pointed out that these mini-games could be too complex. In addition, an unexpected result that was not pointed out by the players was that the complexity in the levels was

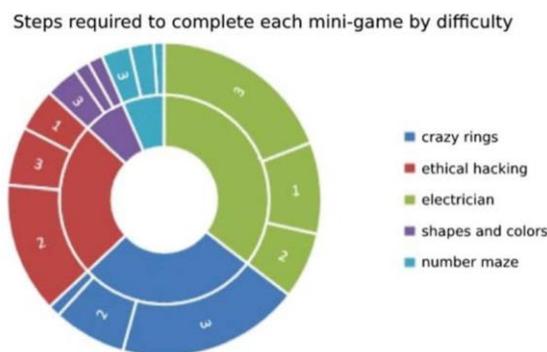


Fig. 5. Comparison of the number of steps required to pass a difficulty level in one of the mini-games. The legend shows the different mini-games names. In the graph, the inner layer represents each mini-game, while the outer layer represents each difficulty level in each mini-game.

not progressive in all cases, for example in the electrician mini-game, level 1 requires on average more steps than level 2.

4 Conclusions and Future Work

The extension model based on mini-games presented in this article increases the flexibility of the narrative model implemented in uAdventure. This versatility allows, on the one hand, to include new game mechanics to address skills that were previously very difficult to address within the narrative model. On the other hand, this approach enables a new development model where game developers can benefit from narrative aspects while, at the same time, making greater use of the full capabilities of the Unity platform.

The mini-games are built using uAdventure's native capabilities, such as the narrative engine or learning analytics. In particular, the use of the learning analytics capability is especially relevant for the evaluation of the capabilities and skills to be trained in the mini-games, in line with uAdventure's philosophy of providing analytics by default and with little effort. As seen in the case studies, easy access to analytics also facilitates the development and validation of the mini-games.

The proposed extension model has been validated in two case studies with game development students, where two serious games have been developed in which mini-games have played major roles. In these two serious games, the mini-games allowed cognitive skills to be addressed such as space management or computational thinking, which would have been difficult to tackle using pure narrative elements. We consider that these two case studies validate our proposal, but do not fully exploit its potential. Some of the mini-games developed in the case studies could, with minimal modifications, be reused in other games as new uAdventure game components.

Currently, uAdventure games can be reused (i.e. it is possible to reload a game into the uAdventure editor), with the aim of customizing the educational experience. For example, by adapting the resources or texts to the group of students who will use it. This has been one of the notable features of uAdventure and we believe it can also be applied to mini-games. As future work, we are working to be able to package and exchange mini-games (Fig. 6) through Pumva, a repository of games and mini-games created with and for uAdventure. For ease of use, a mini-game will be packaged together with a descriptor of the input and output variables configuration, as well as its different analytics traces. Using this descriptor, the uAdventure editor will display a friendly interface through which to configure the imported mini-game. On the other hand, we also want to improve learning analytics, with the goal of offering default analytics also for mini-games; and to offer templates for the xAPI traces generated by the mini-games, thus complying with xAPI's goal of keeping traces/statements self-contained. The traces will also include information about the context in the game where the mini-game has been integrated, further simplifying certain analysis tasks. Thanks to this new extension model, the separation of responsibilities between game and mini-game developers will allow educational experts to create games in uAdventure using mini-games in an even simpler way.

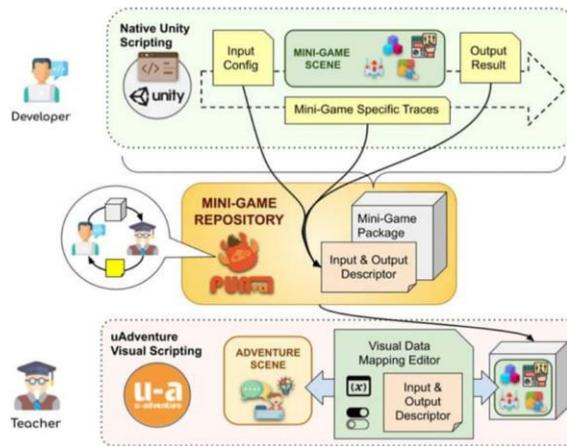


Fig. 6. uAdventure mini-game exchange model through Pumva. The mini-game must be packaged together with a descriptor that identifies its input and output parameters and analytics traces.

Acknowledgements. This work has been partially funded by Regional Government of Madrid (eMadrid S2018/TCS4307, co-funded by the European Structural Funds FSE and FEDER), by the Ministry of Education (TIN2017–89238-R, PID2020-119620RB-I00), by MIT-La Caixa (MISTI program, LCF/PR/MIT19/5184001) and by the Telefonica-Complutense Chair on Digital Education and Serious Games.

References

1. Susi, T., Johannesson, M., Backlund, P.: Serious games – an overview. *Elearning* **73**(10), 28 (2007)
2. Michael, D.R., Chen, S.: *Serious Games: Games that Educate, Train and Inform*. Thomson Course Technology (2006)
3. Mitchell, A., Savill-Smith, C.: *The Use of Computer and Video Games for Learning: a Review of the Literature*, Learning and Skills Development Agency, London (2004)
4. Center for Technology Implementation in Education: *Learning with Computer Games and Simulations*. American Institute Research (2014)
5. Dickey, M.D.: Game design narrative for learning: appropriating adventure game design narrative devices and techniques for the design of interactive learning environments. *Educ. Technol. Res. Dev.* **54**(3), 245–263 (2006)
6. Amory, A.: Building an educational adventure game: theory, design and lessons. *J. Interact. Learn. Res.* **12**(2), 249–263 (2001)
7. Baptista, R., Coelho, A., Vaz de Carvalho, C.: Relation between game genres and competences for in-game certification. *Significance* **13**(6), 28–35 (2016)
8. Pérez-Colado, V.M., Pérez-Colado, I.J., Freire-Morán, M., Martínez-Ortiz, I., Fernández-Manjón, B.: Simplifying the creation of adventure serious games with educational-oriented features. *Educ. Technol. Soc.* **22**(3), 32–46 (2019)

9. Pérez-Colado, I.J., Pérez-Colado, V.M., Freire-Morán, M., Martínez-Ortiz, I., Fernández-Manjón, B.: e-learning standards in game-based learning? In: 21th International Conference on Advanced Learning Technologies (ICALT 2021), pp. 81–82 (2021)
10. Hauge, J.B., et al.: Implications of learning analytics for serious game design. In: IEEE 14th International Conference on Advanced Learning Technologies (ICALT 2014), pp. 230–232 (2014)
11. Serrano-Laguna, Á., Martínez-Ortiz, I., Haag, J., Regan, D., Johnson, A., Fernández-Manjón, B.: Applying standards to systematize learning analytics in serious games. *Comput. Stand. Interfaces* **50**, 116–123 (2017)
12. Bellotti, F., Berta, R., De Gloria, A., Primavera, L.: Adaptive experience engine for serious games. *IEEE Trans. Comput. Intell. AI Games* **1**(4), 264–280 (2009)
13. Barbosa, A.F.S., Pereira, P.N.M., Dias, J.A.F.F., Silva, F.G.M.: A new methodology of design and development of serious games. *Int. J. Comput. Games Technol.* **2014**, 1–8 (2014)
14. Perez Colado, I.J., Perez Colado, V.M., Martinez Ortiz, I., Freire Moran, M., Fernandez Manjon, B.: Simplifying serious games authoring and validation with uadventure and SIMVA. In: 21th International Conference on Advanced Learning Technologies (ICALT 2020), pp. 106–108 (2020)

6.6. Democratizing Game Learning Analytics for Serious Games

6.6.1. Cita completa

Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021a). **Democratizing Game Learning Analytics for Serious Games**. *International Conference on Games and Learning Alliance (GALA)*, 164–173. https://doi.org/10.1007/978-3-030-92182-8_16

6.6.2. Resumen original de la publicación

Interest in the field of serious games (SGs) has grown during the last few years due to its multiple advantages. For example, SGs provide immersive learning environments, where risky or complex scenarios can be tested in safety while keeping players engaged. Moreover, the highly interactive nature of serious games opens new opportunities for applying learning analytics to the interaction data gathered from the gameplays. These interaction data can be used, for example, to measure the impact of serious games on their players. At e-UCM, we have developed open code tools to support serious game learning analytics (GLA), especially an xAPI tracker that collects the player interactions and sends them to a cloud analytic store, SIMVA. Although this tracker uses the xAPI specification as a basis, it includes extensions tailored to our tools. However, not all game developers have the knowledge to operate our analytics infrastructure or are willing to use our tools. We present the design of a GLA system based on existing software modules, focused on collecting and storing analytics generated by SGs in xAPI format. The main elements of this lean architecture are the Learning Record Store (LRS) and the xAPI tracker. With this work, we aim to facilitate and lower the barrier of applying learning analytics in serious games.



Democratizing Game Learning Analytics for Serious Games

Víctor M. Pérez-Colado^(✉) , Iván J. Pérez-Colado , Iván Martínez-Ortiz ,
Manuel Freire-Morán , and Baltasar Fernández-Manjón 

Department of Software Engineering and Artificial Intelligence, Complutense University of Madrid, C/ Profesor José García Santesmases, 9, 28040 Madrid, Spain
victormp@ucm.es

Abstract. Interest in the field of serious games (SGs) has grown during the last few years due to its multiple advantages. For example, SGs provide immersive learning environments, where risky or complex scenarios can be tested in safety while keeping players engaged. Moreover, the highly interactive nature of serious games opens new opportunities for applying learning analytics to the interaction data gathered from the gameplays. These interaction data can be used, for example, to measure the impact of serious games on their players. At e-UCM, we have developed open code tools to support serious game learning analytics (GLA), especially an xAPI tracker that collects the player interactions and sends them to a cloud analytic store, SIMVA. Although this tracker uses the xAPI specification as a basis, it includes extensions tailored to our tools. However, not all game developers have the knowledge to operate our analytics infrastructure or are willing to use our tools. We present the design of a GLA system based on existing software modules, focused on collecting and storing analytics generated by SGs in xAPI format. The main elements of this lean architecture are the Learning Record Store (LRS) and the xAPI tracker. With this work, we aim to facilitate and lower the barrier of applying learning analytics in serious games.

Keywords: Game learning analytics · Serious games · xAPI · Tracker

1 Introduction

The growth in interest in serious games (SGs) is due to their multiple advantages, such as greater attention retention capacity, the ability to teach skills through fun mechanics, or to simulate real environments safely, ensuring more authentic learning. Additionally, SGs that include assessment capabilities can produce evidence-based assessments.

However, few SGs include assessment and, even when present, it is rare to have scientifically validated the assessment with enough learners and in real environments. In most cases, SGs are developed as black box systems [1, 2]. That is, the SG only provides a result (e.g., score), and although they may have a complex internal logic that responds to the interactions and educational needs of the learner, this is not observable from the outside. This implies that there is no evidence of the learning process produced with the game, limiting its educational usefulness, and hindering its scientific validation [1, 3].

© Springer Nature Switzerland AG 2021
F. de Rosa et al. (Eds.): GALA 2021, LNCS 13134, pp. 164–173, 2021.
https://doi.org/10.1007/978-3-030-92182-8_16

At best, the game provides scores, progress, or evaluations as output data; but there is a lack of specific data about the player's process to obtain these results.

To improve the usefulness and reliability of SGs, work has been done mainly on formal game validation [4] and a white-box evaluation or reporting model [5]. Through validation it is possible to ensure that the game produces the expected outcome and, therefore, its application in a real setting should be effective. The most popular method used for validation of SGs is through questionnaires applied before and after using the game in which the desired change is reflected (usually this change is also compared with a control group) [4]. On the other hand, through a white box evaluation, the game sends out the meaningful interactions of the player with the game. This exhaustive report adds meaning or extra information to the activity and can be monitored in real time or analyzed a posteriori to assess the player and even to predict the player's outcome. A particular use case of application of this approach is stealth assessment [6]. We consider that, by combining both methods, formal validation and learning analytics (LA) could be a method capable of providing both assessment and scientific validations.

The open issue is that, despite recognizing the usefulness and potential of LA in the SG community, its application is still limited [7]. Usually, in those cases where analytics are implemented, it is done ad-hoc and from scratch. But implementing LA in games requires specific skills (software development or data analytics) that are not very common in small or medium-sized development studios, which are the ones that usually develop SGs. The use of LA requires not only the deployment of a complex software infrastructure often only within reach of experts, but also meeting both the general and specific educational needs of each game and the regulatory needs for the treatment of information (e.g., EU GDPR privacy law). It is, therefore, necessary to develop methodologies and software modules that simplify this complex task.

To systematize and democratize the application of game learning analytics (GLA), the responsibilities of SG developers should be clearly delimited, with a primary focus on selecting relevant in-game events to be analyzed by data analytics experts. To achieve this goal, two aspects need to be considered from the very beginning: the data exchange format (to achieve common semantics on which to perform the analysis), and the Application Program Interface (API) to be used to communicate formatted data, so SGs can be easily deployed in different environments. Both aspects can be simplified with software modules to both assist with formatting and communicating this information outside the game (i.e., a tracker) and receiving and storing it somewhere (usually in the cloud) securely, while providing at least basic query and analysis functionalities.

Our approach is centered on the use of an e-learning standard designed to capture user interaction as part of an educational scenario such as xAPI (eXperience API), that in the specific case of SGs, provides a format to represent user interactions within the game and that will allow us to record the user traces within the game. The xAPI specification (version 2.0 will become an IEEE standard during 2021) is a specification that provides a statement-based data model that is generic and extensible. The structure of a basic statement consists of the following elements: an actor (*who* performs the action), a verb (*what* action is performed) and an object (the action's *target*). The purpose of this basic structure is to enable communities of practice to define precisely, through custom application profiles, specific vocabularies for each of the elements of statements.

For example, there is an application profile for SGs (xAPI-SG), created by e-UCM in collaboration with ADL as part of the European H2020 RAGE project [8].

As part of previous projects, we developed an open-source tool (tracker) that implements xAPI-SG and simplifies the communication of user traces between the game and an external store, without requiring the SG developer to be an xAPI expert. Until now, this tracker communicated SGs with our analytics tools such as SIMVA and T-Mon, which simplify the management of SG experiments and deployments and automate some aspects of data analysis on xAPI traces [9]. Although these tools are free and open-source, technical knowledge is required for their use and deployment. On the other hand, the xAPI specification also defines a standard API that focuses mainly on sending and querying xAPI *statements*. This standardized API allows an educational tool, such as a SG, to communicate with a generic xAPI Learning Record Store (LRS).

The present work describes how we have combined and adapted our technology (primarily the tracker) to interact with any standard LRS. This should democratize GLA by making it more accessible to game developers, allowing SG developers, educational organizations, and teachers to use the LRSs that best suits their specific needs. For example, some will opt to use an LRS that is already integrated with their existing institutional Learning Management System (LMS), while others may chose a commercial or open-source LRS offered as SaaS. With these enhancements, we can maximize the benefits of using our tracker during SG development (by being able to interact with our SG validation tools) while allowing choice regarding the LRS to use once deployed.

The following sections describe the current model for collecting GLA using the e-UCM group tools; how SG LA can be collected using reusable generic software components; the modifications to our tools to further the democratization of analytics for SGs; and finally, present conclusions and future work.

2 A Tailor-Made Scalable Learning Analytics Ecosystem

Systematization of LA in SGs requires software to store analytics, simplify the collection of analytics within the SG, and communicate it with the analytics store. This section describes our experience developing tools for these problems.

The e-UCM group has developed different tools related to the collection and analysis of data generated by SGs. The most recent is the SIMple VALidator (SIMVA) platform [9]. SIMVA's main objective is the creation and orchestration of SG validation experiments using GLA. Usually, SGs are validated by using questionnaires before and after the activity in which participants play the SG. SIMVA includes an integrated questionnaire management tool to simplify this process. In addition, during the game activity, SIMVA takes care of collecting the LA generated by the SG [7, 10].

Using SIMVA it is possible to design different user experiments (called studies) for different games with full control over the flow of the experiment, offering, for example, the possibility of comparing two versions of the same game (A/B testing). As part of the management of the experiments, SIMVA allows us to generate a list of experiment participants identified exclusively by anonymized identifier tokens, which allow the actions of each participant in the different activities of a study to be linked together, while complying with regulations related to privacy and data protection. Likewise, the

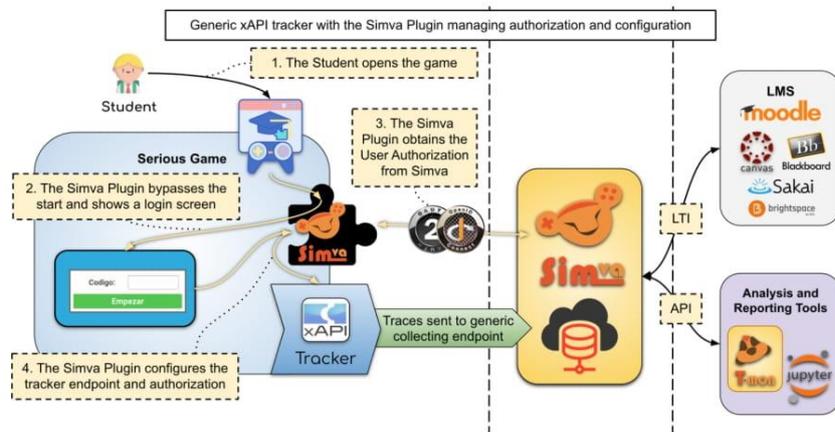


Fig. 1. Interaction flow of a SG with SIMVA, and of SIMVA with other tools such as T-Mon.

researcher or teacher involved in the experiment can monitor the progress of participants in each activity. Finally, SIMVA offers an API that can be used by data analysis experts to download and analyze the data collected during the experiment. This API is compatible with AWS S3, which is one of the APIs commonly supported by big data and data analytics tools such as Jupyter Notebooks.

From a game development perspective, SIMVA provides a plugin compatible with the Unity platform that simplifies authentication and trace submission for study participants/players. The usual process of interaction of a SG with SIMVA is as follows (Fig. 1): 1) players launch the game; 2) the game displays a simplified login interface, where players enter their 5-letter participant code; 3) the game connects to SIMVA by sending the study code and participant code, verifying that the user is assigned to the study and that the activity to be performed is to play the game (for instance, studies may require their participants to complete a survey before playing; in this case, an informative message with a link to the survey would be displayed, and the game would not start); 4) the game obtains the necessary configuration data to use the xAPI tracker and gameplay would start, during which analytics will be sent to SIMVA.

Once the experiment is finished, it is time for data science experts to analyze the data generated during the experiment. To simplify this task, we have created the T-Mon tool built on top of Jupyter Notebooks. T-Mon is a data analysis tool specialized on xAPI data compatible with the xAPI-SG profile, and provides a set of generic analyses and predefined visualizations adapted to this xAPI-SG profile.

Game learning analytics is a complex and error-prone process, requiring skills not always present in a game studio. Collecting game analytics data alone requires linking the game to a data store, identifying the user who is playing the game, and considering many other aspects of security and reliability. The following sections discuss the current capabilities of our analytics tracker, and the design changes needed to extend it so that user traces can be sent to xAPI-standard data stores such as the Learning Record Store (LRS).

3 The e-UCM xAPI Tracker

Adding support for LA to a SG can be a challenge for game developers who do not have experience in LA. This section describes the main features of the tracker software component to identify the functionality required to make it work with a generic LRS.

To simplify the analytics collection process, we have developed a series of reusable software components, called xAPI trackers, compatible with different platforms and programming languages (Unity, .NET, JavaScript, and Java), which were developed and used as part of the EU H2020 projects RAGE and BEACONING. These trackers are middleware components that simplify the sending of traces from games (solving authentication and communication problems), avoiding an extensive knowledge of the xAPI-SG application profile. The rest of this section focuses on the Unity tracker description, which has the most advanced features and the widest application in SGs.

The xAPI tracker provides a high-level interface for interacting with a compatible analytics store (e.g., LRS), specifically targeted for SG development. It currently implements the xAPI-SG application profile and includes another xAPI application profile for creating geopositioned games [11]. In addition, the tracker has other features that differentiate it from other existing xAPI support modules (see Fig. 2):

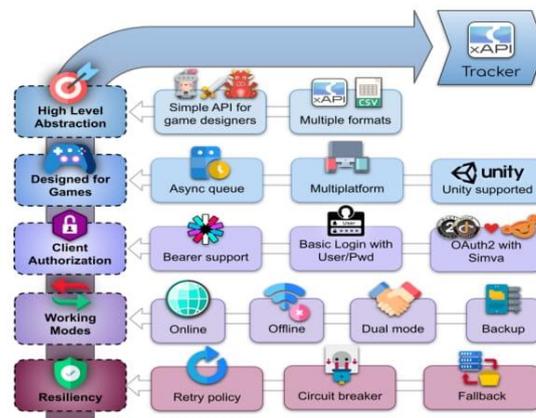


Fig. 2. e-UCM xAPI Tracker functionalities. Its five key aspects are shown on the left.

The first and main feature of the tracker is to provide a high-level and simplified interface so that developers and designers do not have to be experts in the xAPI format to record the game session or send interaction information. In fact, it allows traces to be created in multiple formats. For example, in addition to xAPI format, it is also possible to generate the traces using the more compact CSV format. For example, this is the format used to store a local backup of the traces inside the gaming device (if the backup mode of the tracker is enabled).

The tracker is implemented using Unity's primitives (such as network access), without which it would be more complex to create cross-platform SGs. In addition, the

tracker has an interface to build partial traces to maintain the logical order of events even considering the internal logic of the game and its possible interactions.

The tracker supports authentication and privacy aspects. The current xAPI specification mandates only basic web authentication and OAuth1, neither of which is without problems. Basic authentication only authenticates tools, but not users; and, if game privileges are desired, different credentials for the same game must be created, complicating its use. On the other hand, OAuth1 has been discouraged for some time due to protocol security flaws. Thus, our tracker has been updated to support the new OAuth2 and OpenID connect standards, which are considered secure and allow not only to identify the tool but also the user who is using it. This improves security and minimizes the possibility that a user can access or send data on behalf of another user.

The tracker currently provides three modes of operation: online, offline, and with backups. These modes reflect our experience and the complexities of the actual use and application of SGs in real educational environments, where technical limitations and limited network reliability in schools or institutions where games are deployed games are a frequent occurrence. In online mode the tracker sends the data to an analytics store such as SIMVA, in offline mode the tracker stores the data on the gaming device, and in backup mode both strategies are combined to achieve greater reliability.

The tracker must be robust to be able to operate in non-reliable environments where technical problems may emerge. To this end, the tracker incorporates different strategies, such as exponential back off retry, bulkhead isolation, or circuit-breaker. For example, if the tracker detects that the analytics store is not responding properly for a certain period (which is configurable), it automatically switches to offline mode, and after a certain period it will retry the pending operations.

Despite the advanced features described above, before this work our tracker only implemented the xAPI-SG application profile, and due to issues with authentication and authorization was not designed to interact directly with a standard LRS.

4 DA Standard-Based Scalable Generic Game Learning Analytics Infrastructure

The creation of a platform to support LA for SGs requires different components depending on intended goals and required functionalities. Basic analytics support which includes only the collection of traces from a single game for later analysis is very different from a complete system that supports multiple games and also supports real-time user interaction analysis (e.g., H2020 BEACONING or RAGE); or a complete system to support experimentation with SGs such as SIMVA, where playing a game is only one of multiple activities that can be built into a study.

Game analytics processes are complex and fragile, with many sources of potential errors, such as communication, latency or availability – and it is not always possible to develop a fully integrated approach. The present work describes a strategy for implementing a GLA system based on existing software modules, focusing on the collection and storage of analytics generated by SGs in xAPI format. The main elements of this lean architecture are the Learning Record Store (LRS) and the xAPI tracker.

By adopting the full xAPI specification, which includes both a data format and programming interface for standards-compliant LRSs, we can reduce development costs and allow analytics collected in different deployment environments to share a conforming LRS, or the use of existing xAPI API-compatible services to interact with those LRSs. At the time of this writing, several popular LRSs are available:

- ADL LRS is an open source LRS by ADL, supporting OAuth1. However, this LRS is a reference implementation, designed only to carry out proof-of-concept activities with small numbers of users.
- Learning Locker is an open source LRS with support for basic data analysis and visualization and a business rules layer for easier integration with other systems. It also has an enterprise-oriented SaaS version.
- Rustici LRS is the main reference LRS, as Rustici works closely with ADL; and is offered as part of SCORM Cloud, an LMS for xAPI activities that can act as a mediator between traditional LMSs that support SCORM 1.2 or LTI and CMI-5 packages. This LRS supports similar data processing and business rules to Learning Locker, but offloads reports, analysis and visualizations to SCORM Cloud.
- Yet Analytics LRS is a commercial LRS that is offered in three versions: as a Cloud deployment, as a self-contained LRS compatible with a SQL interface installable on-premises, and as a hardware appliance that can even be deployed in experiments or field activities. To provide analytics and visualizations, Yet Analytics offers an xAPI Sandbox, a free platform based on its Yet Pro v2 LRS that provides on-demand LRSs and supports multiple dashboards.
- Watershed LRS is a commercial LRS that provides reports, performs data conversion to different formats and allows editing trace data or metadata. Watershed's LRS is currently free, although features such as dashboards, analysis, or visualizations require paid licenses.
- Apereo OpenLRW is an open source project that provides an xAPI-compliant LRS, as well as other analytics specifications such as IMS Caliper or IMS OneRoster. To analyze data, it can be connected with Apereo OpenDashboard-API, a framework for creating dashboards and visualizations.

An LRS is only useful if it receives traces that it can store for later analysis. We have identified three open-source trackers that can be used for development of SGs in Unity:

- TinCan.NET by Rustici Software is developed on .NET Framework 3.5, and can therefore be used from within the Unity platform. However, due to the peculiarities of Unity's .NET support, development of cross-platform games presents several issues. In addition, its use of synchronous communication clashes with the Unity game development model. Rustici also provides a JavaScript version: TinCan.JS.
- UnityGBLxAPI (formerly GBLxAPI) by Dig-iT! Games is a wrapper for TinCan.NET, with several improvements: i) better setup and compatibility, since it is Unity-specific; ii) an asynchronous queue approach to send the statements, which avoids game freezes while sending data; and iii) cross-platform compatibility of the generated games by using the specific network primitives offered by Unity. As a limitation, this tracker loses the ability to interpret LRS certain responses, preventing

developers from relating traces to each other via xAPI StatementRefs; also, it may be difficult to update if the underlying TinCan.NET library changes.

- **Unity-xAPIWrapper** by ADL is a lightweight tracker, with less dependencies and easier to use than the above alternatives. It uses Unity primitives for network communication, with similar advantages to those of UnityGBLxAPI. Its main limitations are that it is very simple, and requires more work from developer to achieve a flexible, secure, and resilient tracker; and a low rate of updates.

To use the xAPI API to communicate with an LRS from a game, certain configuration parameters must be known: base URL of the LRS, and usernames and passwords for the LRS. The xAPI specification does not detail how activities obtain these parameters. Fortunately, this problem has already been addressed in the CMI-5 specification [12]. CMI-5 is considered to be a natural evolution of SCORM, but using xAPI to communicate activity interactions with an LRS. This specification addresses several aspects: packaging, launching (including the exchange of credentials and configuration parameters needed to communicate with an LRS), and an xAPI application profile (xAPI-CMI5) that provides insight into the status and progress of an activity. Despite its benefits, CMI-5 adoption is currently very limited, especially by LMSs.

5 Adaptation of the Tracker for Use in the Generic Architecture

The use of a standard LRS can have advantages, especially for the actual exploitation of SGs in real educational settings, for example, by integrating them with a pre-existing e-learning infrastructure. We believe that our tracker has several unique and desirable characteristics, but we are also aware that to lower the entry barrier of applying LA we should be compatible with xAPI-compliant LRSs. In this regard, we have identified the following modifications to achieve this compatibility with our tracker: i) improve adherence to the xAPI specification, ii) provided support for the CMI-5 protocol, and iii) implement different modes of operation that allow running both SIMVA and a generic LRS configured either manually or via CMI-5 release.

Two additional modifications allow the standard to be followed more closely. On the one hand, integration with SIMVA extends the standard with additional services such as user authentication or retrieval of game configuration. We replaced them with traces following the CMI-5 profile. On the other hand, the generated traces were intended to be stored in a specific storage associated with the specific activity created in SIMVA to represent the game session. However, standard LRSs do not partition data like this. Using the *context.contextActivities* property of a statement, it is possible to specify the context in which it has been generated, for example, specifying in which SG the trace was generated, the educational activity that represents the session (or sessions) where the SG is used, or even the course where the educational activity is being included. Additionally, by using the *context.registration* property we can identify statements that belong to each attempt of a player.

Moreover, implementing the CMI-5 launch protocol allows the tracker to receive both the location of the LRS where information should be sent, as well as the authentication of the user who is developing the activity. Should CMI-5 not be available, the tracker can read this configuration from locations chosen at development time.

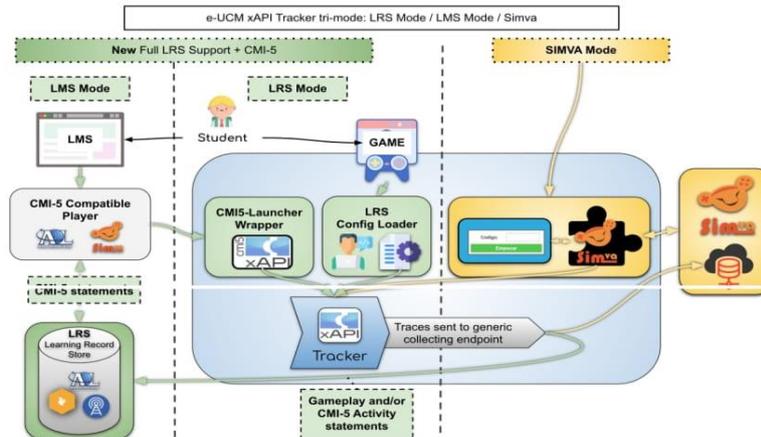


Fig. 3. Tracker working modes, internal components and interaction with external systems

Finally, in addition to the operating modes described in Sect. 3 (online, offline, backup), three working modes have been established: the SIMVA mode (the current mode), the standard LRS mode and the LMS-CMI-5 mode (Fig. 3). The SIMVA mode represents the tracker behavior prior to proposed modifications, and uses specific capabilities provided by SIMVA to simplify validation of SGs. In the standard LRS mode, the tracker will be configured to use the xAPI API exclusively, obtaining the configuration parameters for the LRS from the pre-established locations configured during development. Finally, in LMS-CMI-5 mode, LMS support will be used to launch the SG as a CMI-5 activity, retrieving configuration parameters directly from the LMS.

6 Conclusions and Future Work

Learning analytics is a decisive step forward in adopting serious games as reliable tools in the learning process. Analytics provides evidence beyond simple results, and allows insights into how players achieved those results. However, GLA implementations are still complex and fragile, and require significant technical skills to deploy.

In previous work, we proposed a SG analytics architecture that relies on the use of xAPI as a data standard and on a set of software modules to capture SG information (tracker) and communicate it to a data warehouse in the cloud for secure storage and analysis (SIMVA, T-Mon). However, our approach is difficult to integrate with existing infrastructure. To decrease the cost of entry and increase reliability and adoption, this work proposes a similar architecture using pre-existing software components in the xAPI ecosystem (e.g., trackers and LRS). Thanks to the use of standards, both xAPI and LRS, it is possible to swap some of the software components without causing vendor lock-in. Furthermore, it is possible to reuse components such as LRS, which are increasingly already deployed in existing e-learning infrastructure.

Since the e-UCM tracker was, until now, strongly coupled with e-UCM tools such as SIMVA and did not offer full support for third-party LRSs, we are implementing

two new standards-compatible modes for the tracker: an LRS mode to connect to any standards-based LRS, and an LMS mode to allow our newly CMI-5 compliant tracker to convert SGs into packages that can be easily deployed as activities in LMSs.

Acknowledgements. This work has been partially funded by Regional Government of Madrid (eMadrid S2018/TCS4307, co-funded by the European Structural Funds FSE and FEDER) and by the Ministry of Education (TIN2017-89238-R, PID2020-119620RB-I00).

References

1. Chaudy, Y., Connolly, T.: Specification and evaluation of an assessment engine for educational games: empowering educators with an assessment editor and a learning analytics dashboard. *Entertain. Comput.* **27**(September), 209–224 (2018). <https://doi.org/10.1016/j.entcom.2018.07.003>
2. Freire, M., Serrano-Laguna, Á., Manero-Iglesias, B., Martínez-Ortiz, I.: Game learning analytics: learning analytics for serious games. *Learn. Des. Technol.* <https://doi.org/10.1007/978-3-319-17727-4>
3. Marchiori, E.J., Torrente, J., Del Blanco, Á., Moreno-Ger, P., Sancho, P., Fernández-Manjón, B.: A narrative metaphor to facilitate educational game authoring. *Comput. Educ.* **58**(1), 590–599 (2012). <https://doi.org/10.1016/j.compedu.2011.09.017>
4. Boyle, E.A., et al.: An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games. *Comput. Educ.* **94**, 178–192 (2016). <https://doi.org/10.1016/j.compedu.2015.11.003>
5. Hauge, J.B., et al.: Implications of learning analytics for serious game design. In: *Proceedings of the IEEE 14th International Conference on Advanced Learning Technologies, ICALT 2014*, pp. 230–232 (2014). <https://doi.org/10.1109/ICALT.2014.73>
6. Shute, V.J.: Stealth assessment in computer-based games to support learning. *Comput. Games Instr.* **55**(2), 503–524 (2011)
7. Alonso-Fernández, C., Calvo-Morata, A., Freire, M., Martínez-Ortiz, I., Fernández-Manjón, B.: Applications of data science to game learning analytics data: a systematic literature review. *Comput. Educ.* **141**(June), 103612 (2019). <https://doi.org/10.1016/j.compedu.2019.103612>
8. Serrano-Laguna, Á., Martínez-Ortiz, I., Haag, J., Regan, D., Johnson, A., Fernández-Manjón, B.: Applying standards to systematize learning analytics in serious games. *Comput. Stand. Interfaces* **50**, 116–123 (2017). <https://doi.org/10.1016/j.csi.2016.09.014>
9. Perez-Colado, I.J., Calvo-Morata, A., Alonso-Fernández, C., Freire, M., Martínez-Ortiz, I., Fernández-Manjón, B.: Simva: simplifying the scientific validation of serious games. *Proceedings of the IEEE 14th International Conference on Advanced Learning Technologies, ICALT 2014*, pp. 113–115 (2019). <https://doi.org/10.1109/ICALT.2019.00033>
10. Alonso-Fernández, C., Martínez-Ortiz, I., Caballero, R., Freire, M., Fernández-Manjón, B.: Predicting students' knowledge after playing a serious game based on learning analytics data: a case study. *J. Comput. Assist. Learn.* **36**(3), 350–358 (2020). <https://doi.org/10.1111/jcal.12405>
11. Perez-Colado, V.M., Rotaru, D.C., Freire, M., Martinez-Ortiz, I., Fernandez-Manjon, B.: Learning analytics for location-based serious games. In: *2018 IEEE Global Engineering Education Conference (EDUCON)*, vol. 2018, pp. 1192–1200, April 2018. <https://doi.org/10.1109/EDUCON.2018.8363365>
12. Bakhouyi, A., Dehbi, R., Lti, M.T., Hajoui, O.: Evolution of standardization and interoperability on E-learning systems: an overview. In: *2017 16th International Conference on Information Technology Based Higher Education and Training, ITHET 2017* (2017). <https://doi.org/10.1109/ITHET.2017.8067789>

Referencias

- Abt, C. C. (1987). *Serious Games*. University Press of America. <https://rowman.com/ISBN/9780819161482/Serious-Games>
- Afreen, R. (2014). Bring Your Own Device (BYOD) in higher education: Opportunities and challenges. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, 3(1), 133–164.
- Ahlqvist, O. (2017). Location-Based Games. In *International Encyclopedia of Geography: People, the Earth, Environment and Technology* (pp. 1–4). John Wiley & Sons, Ltd. <https://doi.org/10.1002/9781118786352.wbieg0298>
- Ahmad, A., Law, E. L. C., & Moseley, A. (2020). Integrating Instructional Design Principles in Serious Games Authoring Tools: Insights from Systematic Literature Review. *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*, 1–12. <https://doi.org/10.1145/3419249.3420133>
- Alonso-Fernandez, C., Calvo, A., Freire, M., Martinez-Ortiz, I., & Fernandez-Manjon, B. (2017). Systematizing game learning analytics for serious games. *2017 IEEE Global Engineering Education Conference (EDUCON), April*, 1111–1118. <https://doi.org/10.1109/EDUCON.2017.7942988>
- Alonso-Fernández, C., Calvo-Morata, A., Freire, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). Applications of data science to game learning analytics data: A systematic literature review. *Computers & Education*, 141(June), 103612. <https://doi.org/10.1016/j.compedu.2019.103612>
- Alonso-Fernández, C., Martínez-Ortiz, I., Caballero, R., Freire, M., & Fernández-Manjón, B. (2020). Predicting students' knowledge after playing a serious game based on learning analytics data: A case study. *Journal of Computer Assisted Learning*, 36(3), 350–358. <https://doi.org/10.1111/jcal.12405>
- Althoff, T., White, R. W., & Horvitz, E. (2016). Influence of Pokémon Go on Physical Activity: Study and Implications. *Journal of Medical Internet Research*, 18(12), e315. <https://doi.org/10.2196/jmir.6759>
- Anastasiadis, T., Lampropoulos, G., & Siakas, K. (2018). Digital Game-based Learning and Serious Games in Education. *International Journal of Advances in Scientific Research and Engineering*, 4(12), 139–144. <https://doi.org/10.31695/IJASRE.2018.33016>
- Andone, I., Blaszkiewicz, K., Böhmer, M., & Markowetz, A. (2017). Impact of location-based games on phone usage and movement: A Case Study on Pokémon GO. *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, 1–8. <https://doi.org/10.1145/3098279.3122145>
- Arango-López, J., Gallardo, J., Gutiérrez, F. L., Cerezo, E., Amengual, E., & Valera, R. (2017). Pervasive games: Giving a meaning based on the player experience. *Proceedings of the XVIII International Conference on Human Computer Interaction, Part F1311*, 1–4. <https://doi.org/10.1145/3123818.3123832>

- Arango-López, J., Gutiérrez Vela, F. L., Collazos, C. A., Gallardo, J., & Moreira, F. (2021). GeoPGD: methodology for the design and development of geolocated pervasive games. *Universal Access in the Information Society*, 20(3), 465–477. <https://doi.org/10.1007/s10209-020-00769-w>
- Ardito, C., Costabile, M. F., de Angeli, A., & Lanzilotti, R. (2012). Enriching Archaeological Parks with Contextual Sounds and Mobile Technology. *ACM Transactions on Computer-Human Interaction*, 19(4), 1–30. <https://doi.org/10.1145/2395131.2395136>
- Arkenson, C., Chou, Y.-Y., Huang, C.-Y., & Lee, Y.-C. (2014). Tag and seek: A Location-Based Game in Tainan City. *Proceedings of the First ACM SIGCHI Annual Symposium on Computer-Human Interaction in Play*, 315–318. <https://doi.org/10.1145/2658537.2662986>
- Aslan, S. (2016). *Digital Educational Games: Methodologies for Development and Software Quality* (Doctoral dissertation). https://vtechworks.lib.vt.edu/bitstream/handle/10919/73368/Aslan_S_D_2016.pdf
- Aslan, S., & Balci, O. (2015). GAMED: digital educational game development methodology. *Simulation*, 91(4), 307–319. <https://doi.org/10.1177/0037549715572673>
- Backlund, P., & Hendrix, M. (2013). Educational games - Are they worth the effort? A literature survey of the effectiveness of serious games. *2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES)*, 1–8. <https://doi.org/10.1109/VS-GAMES.2013.6624226>
- Bakhouyi, A., Dehbi, R., Talea, M., & Hajoui, O. (2017). Evolution of standardization and interoperability on E-learning systems: An overview. *2017 16th International Conference on Information Technology Based Higher Education and Training (ITHET)*, 1–8. <https://doi.org/10.1109/ITHET.2017.8067789>
- Barko, T., & Sadler, T. D. (2013). Practicality in Virtuality: Finding Student Meaning in Video Game Education. *Journal of Science Education and Technology*, 22(2), 124–132. <https://doi.org/10.1007/s10956-012-9381-0>
- Benford, S., Rowland, D., Flintham, M., Drozd, A., Hull, R., Reid, J., Morrison, J., & Facer, K. (2005). Life on the edge: supporting collaboration in location-based experiences. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '05*, 721. <https://doi.org/10.1145/1054972.1055072>
- Blumberg, F. C., Altschuler, E. A., Almonte, D. E., & Mileaf, M. I. (2013). The Impact of Recreational Video Game Play on Children's and Adolescents' Cognition. *New Directions for Child and Adolescent Development*, 2013(139), 41–50. <https://doi.org/10.1002/cad.20030>
- Boyle, E. A., Hainey, T., Connolly, T. M., Gray, G., Earp, J., Ott, M., Lim, T., Ninaus, M., Ribeiro, C., & Pereira, J. (2016a). An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games. *Computers & Education*, 94, 178–192. <https://doi.org/10.1016/j.compedu.2015.11.003>
- Boyle, E. A., Hainey, T., Connolly, T. M., Gray, G., Earp, J., Ott, M., Lim, T., Ninaus, M., Ribeiro, C., & Pereira, J. (2016b). An update to the systematic literature review

- of empirical evidence of the impacts and outcomes of computer games and serious games. *Computers and Education*, 94, 178–192. <https://doi.org/10.1016/j.compedu.2015.11.003>
- Cagiltay, N. E., Ozcelik, E., & Ozcelik, N. S. (2015). The effect of competition on learning in games. *Computers & Education*, 87, 35–41. <https://doi.org/10.1016/j.compedu.2015.04.001>
- Callaghan, M., McShane, N., Eguíluz, A. G., & Savin-Baden, M. (2018). Extending the Activity Theory Based Model for Serious Games Design in Engineering to Integrate Analytics. *International Journal of Engineering Pedagogy (IJEP)*, 8(1), 109. <https://doi.org/10.3991/ijep.v8i1.8087>
- Calvo-Morata, A., Alonso-Fernández, C., Freire, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2021). Creating awareness on bullying and cyberbullying among young people: Validating the effectiveness and design of the serious game Conectado. *Telematics and Informatics*, 60, 101568. <https://doi.org/10.1016/j.tele.2021.101568>
- Calvo-Morata, A., Alonso-Fernández, C., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). Improving Teacher Game Learning Analytics Dashboards through ad-hoc Development. *Journal of Universal Computer Science*, 25(12), 1507–1530.
- Cauberghe, V., & de Pelsmacker, P. (2010). Advergaming. *Journal of Advertising*, 39(1), 5–18. <https://doi.org/10.2753/JOA0091-3367390101>
- Ceconello, M., Spagnoli, A., Spallazzo, D., & Tolino, U. (2015). Playing design mobile serious games to valorize design culture in the urban space. *2015 Digital Heritage, September*, 671–674. <https://doi.org/10.1109/DigitalHeritage.2015.7419595>
- Cervi-Wilson, T., & Brick, B. (2018). ImparApp: Italian language learning with MIT's TaleBlazer mobile app. In *Innovative language teaching and learning at university: integrating informal learning into formal language education* (Issue June, pp. 49–58). Research-publishing.net. <https://doi.org/10.14705/rpnet.2018.22.775>
- Chang, C. Y., & Hwang, G. J. (2019). Trends in digital game-based learning in the mobile era: a systematic review of journal publications from 2007 to 2016. *International Journal of Mobile Learning and Organisation*, 13(1), 68. <https://doi.org/10.1504/IJMLO.2019.096468>
- Chaudy, Y. (2016). *An Assessment and Learning Analytics Engine for Games-based Learning* (Issue December 2015). <https://doi.org/10.13140/RG.2.1.4932.5040>
- Chaudy, Y., Connolly, T., & Hainey, T. (2014). Learning Analytics in Serious Games: a systematic review of literature. *European Conference in the Applications of Enabling Technologies*, November. https://www.researchgate.net/publication/268146930_Learning_Analytics_in_Serious_Games_a_Review_of_the_Literature
- Cheng, Y.-M., Lou, S.-J., Kuo, S.-H., & Shih, R.-C. (2013). Investigating elementary school students' technology acceptance by applying digital game-based learning to environmental education. *Australasian Journal of Educational Technology*, 29(1), 96–110. <https://doi.org/10.14742/ajet.65>

- Cheung, P. (2003). *Charles River City: An Educational Augmented Reality Simulation Pocket PC Game by (Master's thesis)*. <https://dspace.mit.edu/handle/1721.1/27096>
- Chittaro, L. (2006). Visualizing Information on Mobile Devices. *Computer*, 39(3), 40–45. <https://doi.org/10.1109/MC.2006.109>
- Çiftci, S. (2018). Trends of Serious Games Research from 2007 to 2017: A Bibliometric Analysis. *Journal of Education and Training Studies*, 6(2), 18. <https://doi.org/10.11114/jets.v6i2.2840>
- Corti, K. (2006). Games-based Learning: a serious business application. In *PixelLearning: Vol. 34(6)* (pp. 1–20). https://www.cs.auckland.ac.nz/courses/compsci777s2c/lectures/Ian/serious_games_business_applications.pdf
- Cotroneo, D., Orlando, S., & Russo, S. (2007). Characterizing Aging Phenomena of the Java Virtual Machine. *2007 26th IEEE International Symposium on Reliable Distributed Systems (SRDS 2007)*, 127–136. <https://doi.org/10.1109/SRDS.2007.22>
- Coulter, B., Klopfer, E., Sheldon, J., & Perry, J. (2016). Discovering Familiar Places: Learning through Mobile Place-Based Games. In C. Steinkuehler, K. Squire, & S. Barab (Eds.), *Games, Learning, and Society* (pp. 327–354). Cambridge University Press. <https://doi.org/10.1017/CBO9781139031127.025>
- Cowan, B., & Kapralos, B. (2017). An Overview of Serious Game Engines and Frameworks. In *Intelligent Systems Reference Library* (Vol. 119, pp. 15–38). https://doi.org/10.1007/978-3-319-49879-9_2
- Davis, J. (2020). *Capture Latitude and Longitude with xAPI (GeoLocation) and save to Learning Record Store [Part 3] | Digital Learning Solutions*. Digitallearningsolutions.Com.Au. <https://digitallearningsolutions.com.au/capture-latitude-and-longitude-with-xapi-geolocation-and-save-to-learning-record-store-part-3/>
- de Freitas, S. (2018). Are games effective learning tools? A review of educational games. *Journal of Educational Technology & Society*, 21(2), 74–84.
- de Heer, J., de Groot, T., Hrynkiwicz, R., Oortwijn, T., & Porskamp, P. (2020). Devilish Dilemma Games: Narrative Game Engine and Accompanying Authoring Tool. In T. Ahram (Ed.), *Advances in Human Factors in Wearable Technologies and Game Design. AHFE 2019. Advances in Intelligent Systems and Computing* (Vol. 973, pp. 169–178). https://doi.org/10.1007/978-3-030-20476-1_18
- del Blanco, Á., Serrano Laguna, Á., Martínez Ortiz, I., Fernández Manjón, B., & Stanescu, I. A. (2013). Integrating serious games into e-learning platforms: Present and future. *The 9th International Scientific Conference ELearning and Software for Education*. <https://doi.org/10.12753/2066-026X-13-145>
- Dickey, M. D. (2005). Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. *Educational Technology Research and Development*, 53(2), 67–83. <https://doi.org/10.1007/BF02504866>
- Dickey, M. D. (2006). Game Design Narrative for Learning: Appropriating Adventure Game Design Narrative Devices and Techniques for the Design of Interactive

- Learning Environments. *Educational Technology Research and Development*, 54(3), 245–263. <https://doi.org/10.1007/s11423-006-8806-y>
- Djaouti, D., Alvarez, J., Jessel, J., & Rampnoux, O. (2011). Origins of Serious Games. In *Serious Games and Edutainment Applications* (pp. 25–43). Springer London. https://doi.org/10.1007/978-1-4471-2161-9_3
- Djaouti, D., Alvarez, J., & Jessel, J.-P. (2011). Classifying Serious Games. In *Serious Games: Mechanisms and Effects* (Issue 2005, pp. 118–136). <https://doi.org/10.4018/978-1-60960-495-0.ch006>
- do Valle, P. O., Reis, E., Menezes, J., & Rebelo, E. (2004). Behavioral Determinants of Household Recycling Participation. *Environment and Behavior*, 36(4), 505–540. <https://doi.org/10.1177/0013916503260892>
- Edmonds, R., & Smith, S. (2017). From playing to designing: Enhancing educational experiences with location-based mobile learning games. *Australasian Journal of Educational Technology*, 33(6), 41–53. <https://doi.org/10.14742/ajet.3583>
- Egenfeldt-Nielsen, S. (2006). Overview of research on the educational use of video games. *Nordic Journal of Digital Literacy*, 1(3), 184–214. <https://doi.org/10.18261/ISSN1891-943X-2006-03-03>
- Faouaz Santillana, D., García Cárdenas, A., & Poyatos Morate, Á. (2021). *Juegos Serios para Promover el Pensamiento Computacional y la Programación (Bachelor's thesis)*. <https://eprints.ucm.es/id/eprint/66940/>
- Fessakis, G., Bekri, A. F., & Konstantopoulou, A. (2016). Designing a mobile game for spatial and map abilities of kindergarten children. *Proceedings of the European Conference on Games-Based Learning*, 183–191. <https://login.bucm.idm.oclc.org/login?url=https://www.proquest.com/conference-papers-proceedings/designing-mobile-game-spatial-map-abilities/docview/1859715160/se-2?accountid=14514>
- Fidalgo-Neto, A. A., Tornaghi, A. J. C., Meirelles, R. M. S., Berçot, F. F., Xavier, L. L., Castro, M. F. A., & Alves, L. A. (2009). The use of computers in Brazilian primary and secondary schools. *Computers & Education*, 53(3), 677–685. <https://doi.org/10.1016/j.compedu.2009.04.005>
- Freire, M., Serrano-Laguna, Á., Manero-Iglesias, B., & Martínez-Ortiz, I. (2016). Game learning analytics: Learning analytics for serious games. In *Learning, design, and Technology*. <https://doi.org/10.1007/978-3-319-17727-4>
- Fu, Q.-K., & Hwang, G.-J. (2018). Trends in mobile technology-supported collaborative learning: A systematic review of journal publications from 2007 to 2016. *Computers & Education*, 119, 129–143. <https://doi.org/10.1016/j.compedu.2018.01.004>
- Garneli, V., Giannakos, M., & Chorianopoulos, K. (2017). Serious games as a malleable learning medium: The effects of narrative, gameplay, and making on students' performance and attitudes. *British Journal of Educational Technology*, 48(3), 842–859. <https://doi.org/10.1111/bjet.12455>
- Gibson, D., & de Freitas, S. (2016). Exploratory Analysis in Learning Analytics. *Technology, Knowledge and Learning*, 21(1), 5–19. <https://doi.org/10.1007/s10758-015-9249-5>

- Gipps, C. (2011). *Beyond Testing (Classic Edition)*. In *Beyond Testing (Classic Edition)*. Routledge. <https://doi.org/10.4324/9780203182437>
- Göbel, S. (2017). *Autorenumgebung für Serious Games - StoryTec: Eine Autorenumgebung und narrative Objekte für personalisierte Serious Games*. <http://tuprints.ulb.tu-darmstadt.de/6941/>
- Göbel, S., Salvatore, L., Konrad, R. A., & Mehm, F. (2008). StoryTec: A Digital Storytelling Platform for the Authoring and Experiencing of Interactive and Non-linear Stories. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (pp. 325–328). https://doi.org/10.1007/978-3-540-89454-4_40
- Grace, L. (2005). *Game Type and Game Genre*. http://aii.lgracegames.com/documents/Game_types_and_genres.pdf
- Green, C. S., & Bavelier, D. (2015). Action video game training for cognitive enhancement. *Current Opinion in Behavioral Sciences*, 4, 103–108. <https://doi.org/10.1016/j.cobeha.2015.04.012>
- Gunter, G. A., Kenny, R. F., & Vick, E. H. (2006). A case for a formal design paradigm for serious games. *The Journal of the International Digital Media and Arts Association*, 3(1), 93–105. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.93.3845&rep=rep1&type=pdf>
- Guy, I., Ronen, I., & Raviv, A. (2011). Personalized activity streams: Sifting through the “river of news.” *RecSys’11 - Proceedings of the 5th ACM Conference on Recommender Systems*, 181–188. <https://doi.org/10.1145/2043932.2043966>
- Halverson, R., & Owen, V. E. (2014). Game-based assessment: an integrated model for capturing evidence of learning in play. *International Journal of Learning Technology*, 9(2), 111. <https://doi.org/10.1504/IJLT.2014.064489>
- Hansman, C. A. (2001). Context-Based Adult Learning. *New Directions for Adult and Continuing Education*, 2001(89), 43. <https://doi.org/10.1002/ace.7>
- Hauge, J. B., Berta, R., Fiucci, G., Manjon, B. F., Padron-Napoles, C., Westra, W., & Nadolski, R. (2014). Implications of Learning Analytics for Serious Game Design. *2014 IEEE 14th International Conference on Advanced Learning Technologies*, 230–232. <https://doi.org/10.1109/ICALT.2014.73>
- Hsu, H.-Y., & Wang, S.-K. (2010). Using Gaming Literacies to Cultivate New Literacies. *Simulation & Gaming*, 41(3), 400–417. <https://doi.org/10.1177/1046878109355361>
- Hsu, Y. C., Ho, H. N. J., Tsai, C. C., Hwang, G. J., Chu, H. C., Wang, C. Y., & Chen, N. S. (2012). Research trends in technology-based learning from 2000 to 2009: A content analysis of publications in selected journals. *Educational Technology and Society*, 15(2), 354–370.
- Huizenga, J., Admiraal, W., Akkerman, S., & ten Dam, G. (2009). Mobile game-based learning in secondary education: engagement, motivation and learning in a mobile city game. *Journal of Computer Assisted Learning*, 25(4), 332–344. <https://doi.org/10.1111/j.1365-2729.2009.00316.x>

- Hwang, G. J., & Wu, P. H. (2014). Applications, impacts and trends of mobile technology-enhanced learning: a review of 2008-2012 publications in selected SSCI journals. *International Journal of Mobile Learning and Organisation*, 8(2), 83. <https://doi.org/10.1504/IJMLO.2014.062346>
- Hwang, G.-J., & Chang, S.-C. (2016). Effects of a peer competition-based mobile learning approach on students' affective domain exhibition in social studies courses. *British Journal of Educational Technology*, 47(6), 1217–1231. <https://doi.org/10.1111/bjet.12303>
- Iqbal, M. (2021). *Pokémon Go Revenue and Usage Statistics (2021) - Business of Apps*. Businessofapps.Com. <https://www.businessofapps.com/data/pokemon-go-statistics/>
- Jean Justice, L., & Ritzhaupt, A. D. (2015). Identifying the Barriers to Games and Simulations in Education. *Journal of Educational Technology Systems*, 44(1), 86–125. <https://doi.org/10.1177/0047239515588161>
- Kantardzic, M. (2019). *Data Mining: Concepts, Models, Methods, and Algorithms* (3rd editio). Wiley-IEEE Press. <https://www.wiley.com/en-us/Data+Mining%3A+Concepts%2C+Models%2C+Methods%2C+and+Algorithms%2C+3rd+Edition-p-9781119516071>
- Karoui, A., Marfisi-Schottman, I., & George, S. (2015). Towards an Efficient Mobile Learning Games Design Model. *European Conference on Game Based Learning EGCBL*, 276–285. <https://doi.org/hal-01383195>
- Karoui, A., Marfisi-Schottman, I., & George, S. (2016). Mobile Learning Game Authoring Tools: Assessment, Synthesis and Proposals. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 10056 LNCS* (Issue October 2017, pp. 281–291). https://doi.org/10.1007/978-3-319-50182-6_25
- Karoui, A., Marfisi-Schottman, I., & George, S. (2017). A Nested Design Approach for Mobile Learning Games. *Proceedings of the 16th World Conference on Mobile and Contextual Learning*, 1–4. <https://doi.org/10.1145/3136907.3136923>
- Karoui, A., Marfisi-Schottman, I., & George, S. (2020). JEM Inventor: a mobile learning game authoring tool based on a nested design approach. *Interactive Learning Environments*, 0(0), 1–28. <https://doi.org/10.1080/10494820.2020.1753214>
- Kasapakis, V., & Gavalas, D. (2015). Pervasive gaming: Status, trends and design principles. *Journal of Network and Computer Applications*, 55, 213–236. <https://doi.org/10.1016/j.jnca.2015.05.009>
- Kato, P. M. (2012). Evaluating Efficacy and Validating Games for Health. *Games for Health Journal*, 1(1), 74–76. <https://doi.org/10.1089/g4h.2012.1017>
- Kazimoglu, C., Kiernan, M., Bacon, L., & Mackinnon, L. (2012). A Serious Game for Developing Computational Thinking and Learning Introductory Computer Programming. *Procedia - Social and Behavioral Sciences*, 47, 1991–1999. <https://doi.org/10.1016/j.sbspro.2012.06.938>
- Kenny, R., & Gunter, G. (2011). Factors affecting adoption of video games in the classroom. *Journal of Interactive Learning Research*, 22(2), 259–276.

- Kent, S. L. (2010). *The Ultimate History of Video Games: from Pong to Pokemon and beyond... the story behind the craze that touched our lives and changed the world*. Prima Communications, Inc.P.O. Box 1260 Rocklin, CA, United States. <https://dl.acm.org/doi/book/10.5555/559522>
- Kirkle, S., Tomblin, S., & Kirkley, J. (2005). Instructional design authoring support for the development of serious games and mixed reality training. *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2005*, 1–11. <http://ntsa.metapress.com/index/fxq6l776cdq10598.pdf>
- Kjeldskov, J., & Paay, J. (2007). Augmenting the City with Fiction: Fictional Requirements for Mobile Guides. In K. Cheverst (Ed.), *Workshop on HCI in Mobile Guides, Mobile HCI 2007* (pp. 1–6). University of Lancaster. <http://people.cs.aau.dk/~jesper/pdf/conferences/Kjeldskov-C45.pdf>
- Klopper, E., & Sheldon, J. (2010). Augmenting your own reality: Student authoring of science-based augmented reality games. *New Directions for Youth Development*, 2010(128), 85–94. <https://doi.org/10.1002/yd.378>
- Klopper, E., Squire, K., & Jenkins, H. (2002). Environmental Detectives: PDAs as a window into a virtual simulated world. *Proceedings. IEEE International Workshop on Wireless and Mobile Technologies in Education, May 2014*, 95–98. <https://doi.org/10.1109/WMTE.2002.1039227>
- Kriglstein, S., Brandmüller, M., Pohl, M., & Bauer, C. (2017). A location-based educational game for understanding the traveling salesman problem. *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, 1–8. <https://doi.org/10.1145/3098279.3122130>
- Laamarti, F., Eid, M., & el Saddik, A. (2014). An Overview of Serious Games. *International Journal of Computer Games Technology*, 2014, 1–15. <https://doi.org/10.1155/2014/358152>
- LeBlanc, A. G., & Chaput, J.-P. (2017). Pokémon Go: A game changer for the physical inactivity crisis? *Preventive Medicine*, 101, 235–237. <https://doi.org/10.1016/j.ypmed.2016.11.012>
- Lehmann, S. E. (2013). *TaleBlazer: Implementing a Multiplayer Server for Location-Based Augmented Reality Games (Master's thesis)*. <https://dspace.mit.edu/handle/1721.1/85441>
- Lemcke, S., Haedge, K., Zender, R., & Lucke, U. (2015). RouteMe: a multilevel pervasive game on mobile ad hoc routing. *Personal and Ubiquitous Computing*, 19(3–4), 537–549. <https://doi.org/10.1007/s00779-015-0843-2>
- Lindert, L., & Su, B. (2016). The evolution of SCORM to Tin Can API: Implications for instructional design. *Educational Technology*, 56(4), 44–46.
- Lloret Irlés, D., Morell Gomis, R., Marzo Campos, J. C., & Tirado González, S. (2018). Validación española de la Escala de Adicción a Videojuegos para Adolescentes (GASA). *Atención Primaria*, 50(6), 350–358. <https://doi.org/10.1016/j.aprim.2017.03.015>
- López-Arcos, J. R., Vela, F. L. G., Padilla-Zea, N., & Rodríguez, P. P. (2016). Interactive Narrative design for geolocated experiences. *Proceedings of the XVII International*

- Conference on Human Computer Interaction*, 1–2.
<https://doi.org/10.1145/2998626.2998666>
- Lucke, U., & Rensing, C. (2014). A survey on pervasive education. *Pervasive and Mobile Computing*, 14, 3–16. <https://doi.org/10.1016/j.pmcj.2013.12.001>
- Lund, K., Coulton, P., & Wilson, A. (2013). Revealing the Participation Inequality in Mobile Location Based Games. *Computers in Entertainment*, 11(3), 1–13. <https://doi.org/10.1145/2582186.2633433>
- Marchiori, E. J., Ferrer, G., Fernández-Manjón, B., Povar-Marco, J., Suberviola, J. F., & Giménez-Valverde, A. (2012). Video-game instruction in basic life support maneuvers. *Emergencias*, 24(6), 433–437.
- Marchiori, E. J., Torrente, J., del Blanco, Á., Moreno-Ger, P., Sancho, P., & Fernández-Manjón, B. (2012a). A narrative metaphor to facilitate educational game authoring. *Computers & Education*, 58(1), 590–599. <https://doi.org/10.1016/j.compedu.2011.09.017>
- Marchiori, E. J., Torrente, J., del Blanco, Á., Moreno-Ger, P., Sancho, P., & Fernández-Manjón, B. (2012b). A narrative metaphor to facilitate educational game authoring. *Computers and Education*, 58(1), 590–599. <https://doi.org/10.1016/j.compedu.2011.09.017>
- Marszał, P. (2016). *Adding Unity3D an Authoring Layer for Non-Programmers (Bachelor's thesis)*. https://pubman.e-ucm.es/drafts/e-UCM_draft_289.pdf
- Martín del Pozo, M. (2014). Herramientas para crear videojuegos para el aprendizaje: creación de un videojuego básico con eAdventure. *III Congreso Ibérico de Innovación En Educación Con Las TIC (IeTIC 2013)*, 814–832. <https://dialnet.unirioja.es/servlet/articulo?codigo=7600955>
- Martín-Maldonado Jiménez, D., & Sandoval Ferrandis, J. (2017). *Desarrollo y edición de juegos serios con Unity y E-Adventure (Bachelor's thesis)*. <https://eprints.ucm.es/id/eprint/45246/>
- McQuiggan, S., Kosturko, L., McQuiggan, J., & Sabourin, J. (2015). Mobile Learning: A Handbook for Developers, Educators, and Learners. In *Technology Enhanced Learning*. John Wiley & Sons, Inc. <https://doi.org/10.1002/9781118938942>
- Medlock-walton, M. P. (2013). *TaleBlazer: a platform for creating multiplayer location based games (Master's thesis)*. <https://dspace.mit.edu/handle/1721.1/77448>
- Mehm, F. (2010). Authoring serious games. *Proceedings of the Fifth International Conference on the Foundations of Digital Games - FDG '10*, 271–273. <https://doi.org/10.1145/1822348.1822390>
- Mehm, F., Göbel, S., & Steinmetz, R. (2012). Authoring of Serious Adventure Games in StoryTec. *Proceedings of the 7th International Conference on Edutainment, and Proceedings of the 3rd International Conference on E-Learning and Games for Training, Education, Health and Sports*, 7516 LNCS, 144–154. https://doi.org/10.1007/978-3-642-33466-5_16
- Melero, J., & Hernandez-Leo, D. (2017). Design and Implementation of Location-Based Learning Games: Four Case Studies with “QuesTInSitu: The Game.” *IEEE*

- Transactions on Emerging Topics in Computing*, 5(1), 84–94.
<https://doi.org/10.1109/TETC.2016.2615861>
- Mettis, K., & Valjataga, T. (2020). Learning through Creating Artefacts for Outdoor Settings with Smartzoos Application: Students Challenges and Experiences. *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*, 110(13), 67–68. <https://doi.org/10.1109/ICALT49669.2020.00027>
- Miller, B., Rutherford, T., Pack, A., Vilches, G., & Ingram, J. (2021). cmi5 Best Practices Guide: From Conception to Conformance. In *Rustici Software*.
- Montola, M., Stenros, J., & Waern, A. (2009). *Pervasive Games: Theory and Design*. CRC Press. <https://doi.org/10.1201/9780080889795>
- Moreno-Ger, P. (2008). *Game Classification: Fire Protocol (2008)*. Gameclassification.Com. <https://gameclassification.com/EN/games/14558-Fire-Protocol/index.html>
- Moreno-Ger, P., Blesius, C., Currier, P., Sierra, J. L., & Fernández-Manjón, B. (2008). Online learning and clinical procedures: Rapid development and effective deployment of game-like interactive simulations. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5080 LNCS, 288–304. https://doi.org/10.1007/978-3-540-69744-2_22
- Mota Macías, J. M., & Ruiz-Rube, I. (2017). VEDILS: a toolkit for developing Android mobile apps supporting mobile analytics. *Seventh European Business Intelligence & Big Data Summer School (EBISS 2017)*.
- Muñoz-Cristóbal, J. A., Rodríguez-Triana, M. J., Gallego-Lema, V., Arribas-Cubero, H. F., Asensio-Pérez, J. I., & Martínez-Monés, A. (2018). Monitoring for Awareness and Reflection in Ubiquitous Learning Environments. *International Journal of Human–Computer Interaction*, 34(2), 146–165. <https://doi.org/10.1080/10447318.2017.1331536>
- Nadolski, R. J., Hummel, H. G. K., van den Brink, H. J., Hoefakker, R. E., Slotmaker, A., Kurvers, H. J., & Storm, J. (2008). EMERGO: A methodology and toolkit for developing serious games in higher education. *Simulation & Gaming*, 39(3), 338–352. <https://doi.org/10.1177/1046878108319278>
- Neustaedter, C., Tang, A., & Judge, T. K. (2013). Creating scalable location-based games: Lessons from Geocaching. *Personal and Ubiquitous Computing*, 17(2), 335–349. <https://doi.org/10.1007/s00779-011-0497-7>
- Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: a model and seven principles of good feedback practice. *Studies in Higher Education*, 31(2), 199–218. <https://doi.org/10.1080/03075070600572090>
- Open Geospatial Consortium. (2010). *gml:AbstractGeometry*. http://www.datypic.com/sc/niem21/e-gml32_AbtractGeometry.html
- Ozdamli, F., & Uzunboylu, H. (2015). M-learning adequacy and perceptions of students and teachers in secondary schools. *British Journal of Educational Technology*, 46(1), 159–172. <https://doi.org/10.1111/bjet.12136>

- Pachler, N., Bachmair, B., & Cook, J. (2010). Mobile Learning: Structures, agency, practices. In *Springer* (Issue November 2019). Springer US. <https://doi.org/10.1007/978-1-4419-0585-7>
- Pánek, J., Gekker, A., Hind, S., Wendler, J., Perkins, C., & Lammes, S. (2018). Encountering Place: Mapping and Location-Based Games in Interdisciplinary Education. *The Cartographic Journal*, 55(3), 285–297. <https://doi.org/10.1080/00087041.2017.1386342>
- Papazoglou Papazoglakis, P. (2013). The Past, Present and Future of SCORM. *Economy Informatics*, 13(1), 16–26. <https://www.proquest.com/docview/1547987470>
- Perez-Colado, I., Alonso-Fernandez, C., Freire, M., Martinez-Ortiz, I., & Fernandez-Manjon, B. (2018). Game learning analytics is not informagic! *2018 IEEE Global Engineering Education Conference (EDUCON), 2018-April*, 1729–1737. <https://doi.org/10.1109/EDUCON.2018.8363443>
- Pérez-Colado, I. J. (2016). *uAdventure: Desarrollo del intérprete y de un emulador de videojuegos de eAdventure sobre Unity3D (Master's thesis)*. <http://eprints.ucm.es/38644/>
- Perez-Colado, I. J., Calvo-Morata, A., Alonso-Fernandez, C., Freire, M., Martinez-Ortiz, I., & Fernandez-Manjon, B. (2019). Simva: Simplifying the Scientific Validation of Serious Games. *2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)*, 113–115. <https://doi.org/10.1109/ICALT.2019.00033>
- Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2019). Simplifying the creation of adventure serious games with educational-oriented features. *Educational Technology and Society*, 22(3), 32–46.
- Pérez-Colado, V. M., Pérez-Colado, I. J., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2021). A Tool Supported Approach for Teaching Serious Game Learning Analytics. *2021 IEEE Frontiers in Education Conference (FIE)*, 1–8. <https://doi.org/10.1109/FIE49875.2021.9637062>
- Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2017). Simplifying location-based serious game authoring. In J. M. Doderó, M. S. Ibarra Sáiz, & I. Ruiz Rube (Eds.), *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM)* (pp. 1–9). ACM. <https://doi.org/10.1145/3144826.3145395>
- Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021a). Democratizing Game Learning Analytics for Serious Games. *International Conference on Games and Learning Alliance (GALA)*, 164–173. https://doi.org/10.1007/978-3-030-92182-8_16
- Pérez-Colado, V. M., Pérez-Colado, I. J., Martínez-Ortiz, I., Freire-Morán, M., & Fernández-Manjón, B. (2021b). Extending Narrative Serious Games Using Ad-Hoc Mini-games. In W. Zhou & M. Yi (Eds.), *Advances in Web-Based Learning – ICWL 2021: 20th International Conference, ICWL 2021, Macau, China, November 13–14, 2021, Proceedings* (pp. 63–74). Springer, Cham. https://doi.org/10.1007/978-3-030-90785-3_6

- Pérez-Colado, V. M., Rotaru, D. C., Freire-Morán, M., Martínez-Ortiz, I., & Fernández-Manjón, B. (2018). Learning analytics for location-based serious games. *2018 IEEE Global Engineering Education Conference (EDUCON), 2018-April*, 1192–1200. <https://doi.org/10.1109/EDUCON.2018.8363365>
- Persico, D., & Pozzi, F. (2015). Informing learning design with learning analytics to improve teacher inquiry. *British Journal of Educational Technology*, *46*(2), 230–248. <https://doi.org/10.1111/bjet.12207>
- Pimmer, C., Mateescu, M., & Gröhbiel, U. (2016). Mobile and ubiquitous learning in higher education settings. A systematic review of empirical studies. *Computers in Human Behavior*, *63*, 490–501. <https://doi.org/10.1016/j.chb.2016.05.057>
- Pishtari, G., Rodríguez-Triana, M. J., Sarmiento-Márquez, E. M., Pérez-Sanagustín, M., Ruiz-Calleja, A., Santos, P., Prieto, L., Serrano-Iglesias, S., & Våljataga, T. (2020). Learning design and learning analytics in mobile and ubiquitous learning: A systematic review. *British Journal of Educational Technology*, *51*(4), 1078–1100. <https://doi.org/10.1111/bjet.12944>
- Premsky, M. (2001a). *Digital game-based learning*. McGraw-Hill.
- Premsky, M. (2001b). Fun, Play and Games: What Makes Games Engaging. In *Digital Game-based Learning* (Issue 1, pp. 1–31). McGraw-Hill.
- Prümper, J. (1993). *Software-evaluation based upon ISO 9241 Part 10* (pp. 255–265). https://doi.org/10.1007/3-540-57312-7_74
- Ramayah, T., Lee, J. W. C., & Lim, S. (2012). Sustaining the environment through recycling: An empirical study. *Journal of Environmental Management*, *102*, 141–147. <https://doi.org/10.1016/j.jenvman.2012.02.025>
- Regal, G., Matheiss, E., Sellitsch, D., & Tscheligi, M. (2018). Mobile location-based games to support orientation & mobility training for visually impaired students. *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*, 1–12. <https://doi.org/10.1145/3229434.3229472>
- Ribeiro, F. R., Silva, A., Silva, A. P., & Metrôlho, J. (2021). Literature Review of Location-Based Mobile Games in Education: Challenges, Impacts and Opportunities. *Informatics*, *8*(3), 43. <https://doi.org/10.3390/informatics8030043>
- Ronimus, M., Kujala, J., Tolvanen, A., & Lyytinen, H. (2014). Children’s engagement during digital game-based learning of reading: The effects of time, rewards, and challenge. *Computers & Education*, *71*, 237–246. <https://doi.org/10.1016/j.compedu.2013.10.008>
- Rosenbaum, E., Klopfer, E., & Perry, J. (2007). On location learning: Authentic applied science with networked augmented realities. *Journal of Science Education and Technology*, *16*(1), 31–45. <https://doi.org/10.1007/S10956-006-9036-0/FIGURES/3>
- Rubino, I., Barberis, C., Xhembulla, J., & Malnati, G. (2015). Integrating a Location-Based Mobile Game in the Museum Visit. *Journal on Computing and Cultural Heritage*, *8*(3), 1–18. <https://doi.org/10.1145/2724723>
- Ruipérez-Valiente, J. A., & Kim, Y. J. (2020). Effects of solo vs. collaborative play in a digital learning game on geometry: Results from a K12 experiment. *Computers &*

- Rupp, A. A., Gushta, M., Mislevy, R. J., & Shaffer, D. W. (2010). Evidence-centered design of epistemic games: Measurement principles for complex learning environments. *Journal of Technology, Learning, and Assessment*, 8(4), 1–48.
- Sancho, P., Torrente, J., & Fernández-Manjón, B. (2012). MareMonstrum: A contribution to empirical research about how the use of MUVES may improve students' motivation. *Journal of Universal Computer Science*, 18(18), 2576–2598. <https://doi.org/10.3217/jucs-018-18-2576>
- Schneider, T., & Lemos, R. (2020). Use of Learning Analytics Interactive Dashboards in Serious Games. *International Journal for Innovation Education and Research*, 8(3), 150–174. <https://doi.org/10.31686/ijer.vol8.iss3.2220>
- Schwabe, G., & Göth, C. (2005). Mobile learning with a mobile game: design and motivational effects. *Journal of Computer Assisted Learning*, 21(3), 204–216. <https://doi.org/10.1111/j.1365-2729.2005.00128.x>
- Serrano-Laguna, Á., Martínez-Ortiz, I., Haag, J., Regan, D., Johnson, A., & Fernández-Manjón, B. (2017). Applying standards to systematize learning analytics in serious games. *Computer Standards & Interfaces*, 50, 116–123. <https://doi.org/10.1016/j.csi.2016.09.014>
- Setar, L., & MacFarland, M. (2012). Top 10 fastest-growing industries. In *IBIS World*. <https://www.phillymag.com/wp-content/uploads/sites/3/2012/04/Fastest-Growing-Industries-2.pdf>
- Setya Murti, H. A., Dicky Hastjarjo, T., & Ferdiana, R. (2019). Platform and Genre Identification for Designing Serious Games. *2019 5th International Conference on Science and Technology (ICST)*, 1–6. <https://doi.org/10.1109/ICST47872.2019.9166177>
- Sharples, M., Taylor, J., & Vavoula, G. (2010). A Theory of Learning for the Mobile Age. In B. Bachmair (Ed.), *Medienbildung in neuen Kulturräumen* (pp. 87–99). VS Verlag für Sozialwissenschaften. https://doi.org/10.1007/978-3-531-92133-4_6
- Sidique, S. F., Lupi, F., & Joshi, S. v. (2010). The effects of behavior and attitudes on drop-off recycling activities. *Resources, Conservation and Recycling*, 54(3), 163–170. <https://doi.org/10.1016/j.resconrec.2009.07.012>
- Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Buckingham Shum, S., Ferguson, R., Duval, E., Verbert, K., & Baker, R. S. (2018). *Open Learning Analytics: an integrated & modularized platform: Proposal to design, implement and evaluate an open platform to integrate heterogeneous learning analytics techniques*.
- Škorić, I., & Milić, M. (2010). Computers in school: A student's perspective. *MIPRO 2010 - 33rd International Convention on Information and Communication Technology, Electronics and Microelectronics, Proceedings*, 1056–1061. <http://www.scopus.com/inward/record.url?eid=2-s2.0-77956391185&partnerID=40&md5=73d7545802ea15a9f9ecff4c84944274>

- Smith, R. (2010). The Long History of Gaming in Military Training. *Simulation & Gaming*, 41(1), 6–19. <https://doi.org/10.1177/1046878109334330>
- Southerton, C. (2014). Zombies, Run! Rethinking Immersion in Light of Nontraditional Gaming Contexts. In *Transmedia Practice: A Collective Approach* (Issue January 2014, pp. 131–141). BRILL. https://doi.org/10.1163/9781848882614_013
- Spallazzo, D., & Mariani, I. (2018). *Location-Based Mobile Games*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-75256-3>
- Squire, K. D. (2008). Video Games and Education: Designing Learning Systems for an Interactive Age. *Educational Technology: The Magazine for Managers of Change in Education*, 48(2), 17–26. https://www.researchgate.net/publication/228731376_Video_Games_and_Education_Designing_learning_systems_for_an_interactive_age
- Stănescu, I. A., Stefan, A., Kravcik, M., Lim, T., & Bidarra, R. (2012). Interoperability strategies for serious games development. *The 8th International Scientific Conference ELearning and Software for Education*, 2(1), 373–378. <https://doi.org/http://dx.doi.org/10.18278/il.2.1.4>
- Taborda Mosquera, J. P., Arango-López, J., Gutiérrez Vela, F. L., Collazos, C., & Moreira, F. (2021). Analyzing effectiveness and fun through metrics applied to pervasive gaming experiences. *Universal Access in the Information Society*, 20(3), 545–554. <https://doi.org/10.1007/s10209-020-00768-x>
- Ternier, S., Klemke, R., Kalz, M., van Ulzen, P., & Specht, M. (2012). *AR Learn: Augmented reality meets augmented virtuality. May 2014*.
- Chaudy, Y., & Connolly, T. (2019). Specification and evaluation of an assessment engine for educational games: Integrating learning analytics and providing an assessment authoring tool. *Entertainment Computing*, 30(July), 100294. <https://doi.org/10.1016/j.entcom.2019.100294>
- Tiew, K., Watanabe, K., Ahmad Basri, N. E., Md. Zain, S., & Basri, H. (2013). Level of Recycling Awareness and Responses to On-campus Recycling Facilities: Case Study-Universiti Kebangsaan Malaysia Students. *Journal of Civil Engineering, Science and Technology*, 4(1), 1–7. <https://doi.org/10.33736/jcest.102.2013>
- Tobias, S., Fletcher, J. D., Yun Dai, D., & Wind, A. P. (2011). Review of Research On Computer Games. In *Computer Games and Instruction* (pp. 127–221). Information Age Publishing. http://www.adlnet.gov/wp-content/uploads/2012/02/Tobias_Fletcher_games_20120215.pdf
- Torrente, J., del Blanco, A., Marchiori, E. J., Moreno-Ger, P., & Fernandez-Manjon, B. (2010). <e-Adventure>: Introducing educational games in the learning process. *IEEE EDUCON 2010 Conference*, 1121–1126. <https://doi.org/10.1109/EDUCON.2010.5493056>
- Valjataga, T., Tammets, P., Tammets, K., Savitski, P., Jaa-Aro, K.-M., & Dias, R. (2017). Designing Learning Experiences in Zoos: A Location-Based Game Development Toolkit. *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*, 84–86. <https://doi.org/10.1109/ICALT.2017.95>

- Vidal, E. C. E., Ty, J. F., Caluya, N. R., & Rodrigo, M. M. T. (2019). MAGIS: mobile augmented-reality games for instructional support. *Interactive Learning Environments*, 27(7), 895–907. <https://doi.org/10.1080/10494820.2018.1504305>
- Volkmar, G., Wenig, N., & Malaka, R. (2018). Memorial Quest - A Location-based Serious Game for Cultural Heritage Preservation. *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, November*, 661–668. <https://doi.org/10.1145/3270316.3271517>
- Wagner-Greene, V. R., Wotring, A. J., Castor, T., Kruger, J., Mortemore, S., & Dake, J. A. (2017). Pokémon GO : Healthy or Harmful? *American Journal of Public Health*, 107(1), 35–36. <https://doi.org/10.2105/AJPH.2016.303548>
- Walther, B. K. (2005). Atomic actions -- molecular experience: theory of pervasive gaming. *Computers in Entertainment*, 3(3), 4–4. <https://doi.org/10.1145/1077246.1077258>
- Wilkinson, P. (2016). A Brief History of Serious Games. In *Entertainment Computing and Serious Games* (Vol. 9970, Issue October, pp. 17–41). https://doi.org/10.1007/978-3-319-46152-6_2
- Wouters, P., van Nimwegen, C., van Oostendorp, H., & van der Spek, E. D. (2013). A meta-analysis of the cognitive and motivational effects of serious games. *Journal of Educational Psychology*, 105(2), 249–265. <https://doi.org/10.1037/a0031311>
- Wu, P. H., Hwang, G. J., & Tsai, W. H. (2013). An expert system-based context-aware ubiquitous learning approach for conducting science learning activities. *Educational Technology and Society*, 16(4), 217–230.
- Yang, C., & Shao, H. (2015). WiFi-based indoor positioning. *IEEE Communications Magazine*, 53(3), 150–157. <https://doi.org/10.1109/MCOM.2015.7060497>
- Zhang, C. (2010). *GameBuilder, an outdoor AR creation software (Master's thesis)*. <https://dspace.mit.edu/handle/1721.1/61244>
- Zyda, M. (2005). From visual simulation to virtual reality to games. *Computer*, 38(9), 25–32. <https://doi.org/10.1109/MC.2005.297>