2012 Fourth IEEE International Conference On Digital Game And Intelligent Toy Enhanced Learning

Integrating domain experts in educational game authoring

A case study

Eugenio J. Marchiori, Ángel Serrano, Ángel del Blanco, Iván Martínez-Ortíz, Baltasar Fernández-Manjón
Department of Artificial Intelligence and Software Engineering
Complutense University
Madrid, Spain
emarchiori@fdi.ucm.es, aserrano@e-ucm.es, {angel.dba, imartinez, balta}@fdi.ucm.es

***Abstract*—Authoring educational games introduces difficult problems because it is the product of multidisciplinary work, integrating very different experts with different backgrounds that use different terminology. In this paper we discuses how a team composed of computer science experts, an education expert and two medical experts successfully tacked the problem of designing and implementing an educational video game. An approach consisting of different tools and strategies was used to ensure educational value, correctness and completeness of the knowledge represented in the game. The game's goal is to teach basic medical first response procedures to young students (12-15 year old) by using photorealistic representations of the situations and videos with correct realization of the procedures. The game was successfully completed and is currently available online and being tested with real students.**

Educational video game; multidisciplinary development; flow-diagrams; rapid-prototyping; case study

I. INTRODUCTION

Educational video games have been established as a useful activity in education, allowing educators to take advantage of their learning-by-doing [1], problem solving [2] and narrative features [3] and their ability to generate and keep flow states [4]. Nonetheless, the ability of games to keep students engaged and motivated throughout a lesson is a key aspect in their educational success [5]. The design of effective educational video games implies taking into account a balance between entertainment and educational value.

For this reason, involving domain experts in game creation (from inception to design and from initial development to testing) becomes crucial [6]. Achieving this effective integration can be very complicated as different problems arise, such as the use of different vocabularies and differing short and long-term goals for the game. For example, programmers typically are more interested in the game dynamics, at the same time that they want to reduce complexity and increase maintainability of the code, while domain experts usually want to capture every detail and every nuance of their field.

In this paper we show how the use of existing sample games (i.e. games representative of the technologies available for development), flow diagrams (complemented with some light scripting), rapid prototyping and full involvement of all experts in each decision helped achieve optimal results (Figure 1) within a limited time frame. Each of those elements plays an essential role in order to achieve a synergy of work of all implied experts. Sample games created previously for the same game engine provide a framework for domain experts to understand the narrative potential and limitation of games, creating a shared knowledge among all parties involved. Flow-diagrams provide a high-level overview of the game, allowing for the identification and definition of key elements and concepts. Early addition of scripts (i.e. textual description of complex steps in the interaction) to flow-diagrams helps define better conversations, interaction with game characters or elements and provides a representation of some parts of games which are easier too understand by some domain experts.

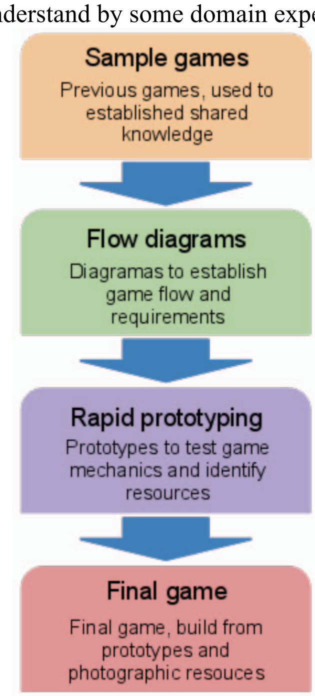

Figure 1. Steps to achieve a game through collaboration.

The use of rapid prototyping is key to achieving good results within a reasonable time (and cost). Prototypes provide a fast way to detect conceptual leaps (e.g. if an important concept was no introduced in the game) or to identify excess or lack of complexity in different parts. Prototypes are very helpful in providing domain experts with a functionally complete version of the game to test and improve upon. Experts have to agree that all the knowledge

978-0-7695-4663-6/12 $26.00 © 2012 IEEE
DOI 10.1109/DIGITEL.2012.20

is properly represented in the game in an iterative process of revision and validation before continuing development. Once they reach a consensus, the prototype is used to identify the required art assets reducing the time consuming and high cost task of producing wrong or inappropriate resources.

We describe how this approach and these tools were applied in the creation of a real-world game, designed to teach high-school students basic concepts about first-response assistance. In the development of this game two domain experts were involved (experienced first-response emergency doctors), together with an educational expert and a team of computer science engineers. All members collaborated closely from the very first stage and were actively involved in the creation of diagrams, prototypes and all other material that needed to be developed in order to obtain the final game.

This paper is divided as follows: In section II we detail our approach and its tools. In section III we describe the first-response game and how we approach its design. Section IV describes the actual rapid-prototyping and implementation process. Section V describes the final game and how it fits the goals of the different experts. Finally, finishing remarks are provided in section VI.

II. Approach and Tools

We used the <e-Adventure> game-authoring tool [7], which allows for the creation of educational video games with reduced cost. Games developed with this platform are usually of the adventure or *point-and-click* genres, with varying degrees of realism. With the use of photographs and videos a very realistic and true-to-life representation of procedures and situations can be achieved.

When designing a game with experts in different fields, the first problem we identified was the differing expectations. Although most people are only familiar with commercial games (including the domain experts involved in the design of our game), for which millions of dollars and several years of development are required [8], typical educational games are developed within tight budgets and need to be deployed months and sometimes weeks after the initial conception to be cost-effective. To tackle this problem in expectations we presented existing <e-Adventure> games, created using similar resources and tools as the ones available for the new game, to all experts in an initial briefing in an effort to ensure that we all shared a common knowledge.

As the extraction and gathering of the domain expert knowledge is another problem that arises is the description of the game, we created flow-diagrams from the first meeting onwards, which were refined and completed with each iteration. The tools used to create these diagrams go from a basic whiteboard to <e-Adventure> WEEV, a tool with a visual language created specifically for the description of educational video games [9]. This tool is oriented towards educational games, with specific features that allow for the direct definition of evaluation mechanisms. However, to increase collaborative work, we used some light scripting (in plain text) for the creation of conversations and the description of some specific and complex situations because this does not require any previous knowledge from domain and educational experts and helps to create a shared initial understanding.

The required prototypes were created directly with <e-Adventure>, the same tool used to create the final game. This allows for the same interaction mechanisms and metaphors to be used in the prototypes as in the final game, but with simple sketch drawing resources. The ability to do this is another advantage of the <e-Adventure> platform, which allows for games to be created from draft drawings of the different situations and elements that where to be included in the game.

This approach contrasts with others, especially those that use storyboards, given that the <e-Adventure> platform is especially suited for the creation of functional prototypes that can be gradually perfected to achieve the final game. Some storyboarding could be used, especially as a complement to the flow-diagrams, where a representation of the state of the game in a point of the diagram needs to be made more clear. However, we used storyboarding just as a complement and not an integral part of our design process.

III. Designing the Basic First-Response Game

With two medical experts, one educational expert and a team computer science engineers we set out to design and implement a First-Response game in the course of one months. This game was intended to teach basic concepts and to be targeted at high-school students in Spain (the game was financed by CATEDU[1], the regional centre for educational technology for the Aragón region). After introducing the domain experts to examples of previous games developed with the <e-Adventure> platform, such as the HTC game [10] (freely available online[2]), we proceeded to try to place the basic procedures in first-response assistance in the same frame of reference within a flow diagram.

The first diagram focused on two things, identifying the main situations (i.e. when first-response assistance should be provided, such as when a person is unconscious on the floor) and the basic procedures to deal with these situations (e.g. applying CPR to unconscious patients who are not breathing) (Figure 2.a). Some concerns arouse, such as the heavy reliance of the <e-Adventure> platform on textual explanations and multiple-choice questions, which required for a new mechanism to provide "graphical" choices that needed to be introduced by the engineers. At the same time, educational concerns arose from the way to evaluate the game, as different situations would mean that either the game presented different cases in a succession or students would only become familiar with one problem. The medical experts found concerns as well, given that different procedures must be applied depending on factors other than the initial situation (e.g. an unconscious person might be breathing or not, requiring different actions).

The first diagram was digitalized and perfected over the following days, allowing for new problems to be identified

[1] http://www.catedu.es
[2] http://catedu.es/eadventure/primerosauxilios/PrimerosAuxiliosGame.jar

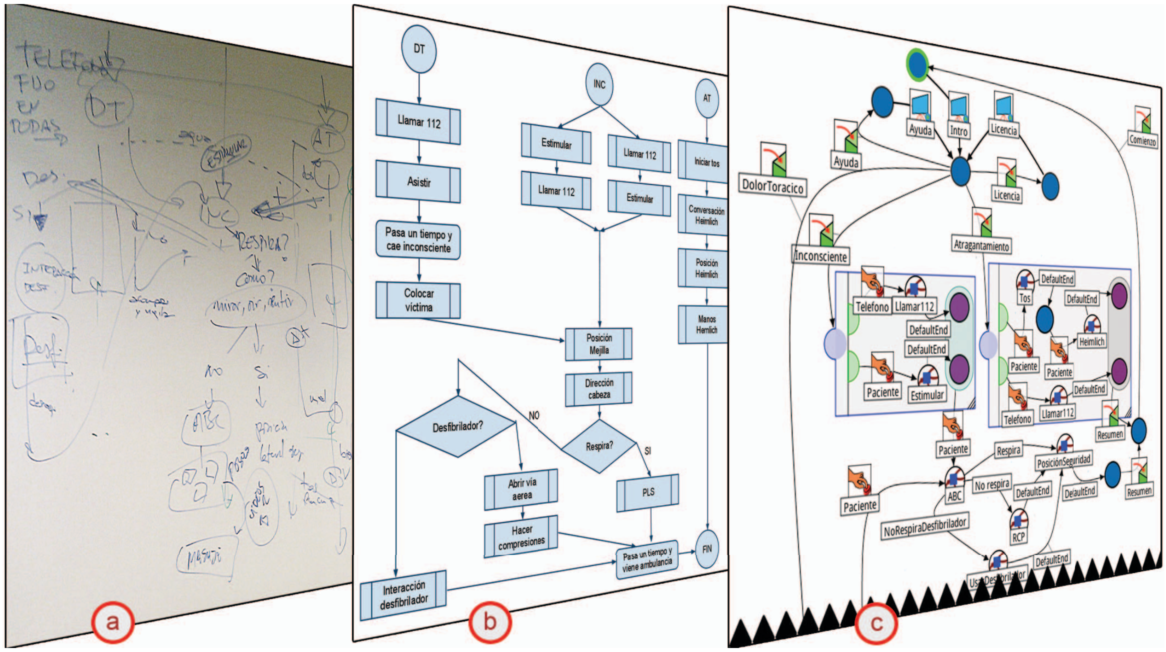

Figure 2. Flow diagrams, from simple whiteboard diagrams to complex and detail representations in <e-Adventure> WEEV.

(Figure 2.b). For instance, although the first "overview" diagram assumed generic situations, detailed analysis of the different steps helped us to identify that many of those situations actually required several smaller steps to be correctly applied. Moreover, other educational concerns appeared as to how to treat bad responses, which could have immediate consequences (e.g. the patient dies) or could have deferred consequences (e.g. the player looses points, but the game continues). The second option was chosen, with particular support from the educational expert, as no to encourage high-school students to fall into “comical” situations by repeatedly "killing" patients on purpose.

The final diagram was created with the <e-Adventure> WEEV specific language for the definition of educational video games [9] (Figure 2.c). This language was only used by the computer science experts as it currently only exists in beta stage, but it allows for the mapping of the broad concepts described in the previous diagrams into actual game elements and metaphors. Different technical concerns were identified at this stage (e.g. how and when to show the phone to users, which they had to use to call an ambulance), which allowed them to be addressed before the first prototype was developed. WEEV allows work to be done at a higher level that the concrete implementation in <e-Adventure>, while still focusing the diagram around real game constructs.

IV. From Rapid-Prototyping to the Implemented Game

Once the game was thoroughly described and all the main concepts, questions and mechanisms were identified, simple sketches of the different scenes, positions, options and game elements were created and photographed. These elements were composed into an actual game using the <e-Adventure> platform (Figure 3.a). This resulted in a fully functional prototype, although the different paths were added one by one. After each path was added, the prototype was distributed between all the team members, how had to confirm that it met all requirements. Changes were made *on-the-fly* to the prototypes, which were then redistributed.

The prototype process allowed for all the interactions, all screens and all elements in the game to be fully described before the actual graphic resources were captured. This is a key element in the creation of a photorealistic game, as the capture process is much simplified and achieves better results if done in a single session.

After the resources capture session, we processed these resources to fit the game needs (trimming images, changing resolutions, editing videos, etc.). These processed resources were used to gradually replace the sketches in the prototype game, in a way that allowed us to gradually test these resources within a fully functional version of the whole game (Figure 3.b). All the experts were involved in this process, testing the actual representation of elements in the final game to make sure that it met all requirements (e.g. that postures and hand placements were seen correctly).

The final prototype, now with the actual resources, went though a last phase where the final elements where added to the game, such as an introduction, a help screen, the licensing terms, etc. With this, the same game that was once a simple prototype was turned into the final game, ready to be evaluated by high-school students to measure its educational potential.

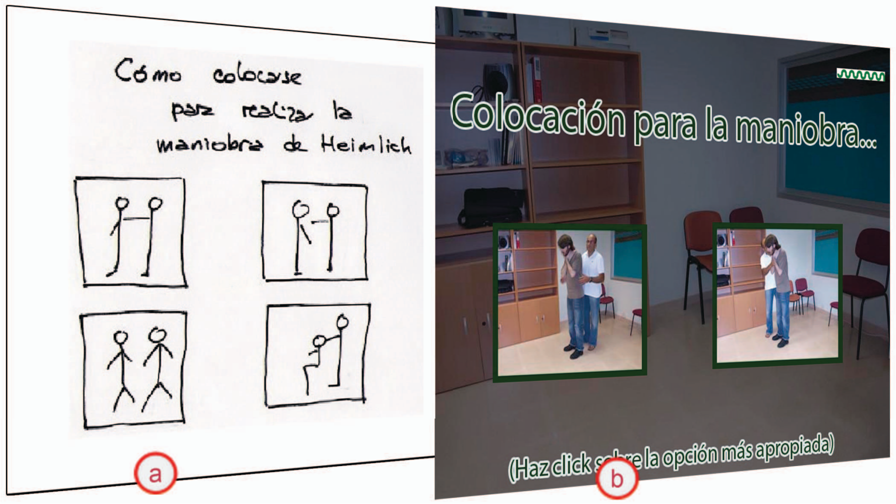

Figure 3. prototypes with sketch drawings to the final implementation with photographs.

V. THE BASIC FIRST-RESPONSE GAME

The first-response game allows users to deal with three different initial situations: choking, unconsciousness and thoracic pain. The third situation joins the second one after a short while (i.e. the person with thoracic pain falls unconscious), simplifying the game but still providing three distinct flows. Besides, each situation changes each time it is played, as some factors are left to random chance (e.g. is there an automatic defibrillator available or not?). Complex procedures are described in videos, where the user sees a medical expert performing it correctly (Figure 4.a).

Each situation in the game receives separate scores when completed, ranking from 1 to 10. Once the user finishes, she sees the score next to the completed option (Figure 4.c). Users can replay from any situation, in order to achieve a perfect score for each one. Besides, once one of the situations is completed, the player can see a summary of all the required procedures, making it easier to go over the key elements in that situation before replaying it (Figure 4.b).

It is important to note that the final game only penalises the students for mistakes by reducing the score. The player is forced to choose a different option when mistaken, only moving forward when the right one is selected. This decision was based on the educator suggestion; because it was considered better that allowing mistakes to have actual immediate consequences, which usually means that the player is taken back to the initial state of the situation when choosing the wrong option. The alternative approach was used in other games, but resulted to cumbersome and boring for some players.

Figure 4. The final first-aid game uses videos (a), summaries of the procedures (b) and individual scores for each situation (c).

The medical concepts covered in the game include the Hemlich manoeuvre to help choking patients, the use of an automatic defibrillator, the ABC (Airways, Breathing, Chest) method to see if a patient is breathing normally, and basic CPR (Cardio-Pulmonary Resuscitation), among others. The game also reflects the very real fact that one of the most important things to do is to call an ambulance as soon as possible, strongly penalising players who do not do that, as suggested by the medical experts.

In this actual experience we found that even following our approach and using our tools, some problems still arise. The biggest problem after the game was created was that expert assumptions about common knowledge meant that some important concepts did not make it into the game. Some of these were so important in the context of the game that it needed modifications after being finished. For example, the fact that survival chances fall about 10% for each minute that nothing is done to help a patient was considered popular knowledge by the medical experts and did not come up until after the first version of the game was ready to be deployed.

Other problems arise by the growing familiarity of all people involved in the game, resulting in some errors being carried over and overlooked. One way to solve this is to beta test the game with other users, especially with experts. For instance, in this game, it helped us to find that we had failed to reflect the fact that applying the automatic defibrillator in a wet body (without drying it first) could have fatal consequences. The game was later modified to reflect this fact.

VI. Final Remarks

Creating an educational game is a challenging task, and involving domain experts as part of the process in more than just a consulting role, can make it even more challenging. In this paper we presented our approach and the tools that helped us to successfully integrate all different experts in the creation of an actual educational game.

We believe that mostly using flow diagrams and using <e-Adventure> to develop prototypes early in the development and for the final game can be applied in many other situations. For example, we used the same procedure in the creation of a game to teach students English vocabulary and typical phrases used in traveling to an English speaking country (targeted to Spanish students).

The fact that, as detailed in the account of our experience, experts fail to detail every important concept can be one of the most challenging things to solve. Although in our case some of these facts came up very late in the development, involving more people in a purely consulting and beta-testing basis (especially other experts) can help mitigate this problem.

Current work in <e-Adventure> is focused on finalizing the WEEV tool to better support this approach to game development, especially in perfecting the system that allows game descriptions to be automatically converted to working prototypes.

Acknowledgment

The Ministry of Education (grant TIN2010-21735-C02-02) and the Ministry of Industry (grants TSI-020110-2009-170, TSI-020312-2009-27) have partially supported this work, as well as the Complutense University of Madrid and the Regional Government of Madrid (research group GR35/10-A and project e-Madrid S2009/TIC-1650), and the PROACTIVE EU project (505469-2009-LLP-ES-KA3-KA3MP) and the GALA EU Network of Excellence in serious games. We would like to thank Gaspar Ferrer, our educational expert, and José Fermín Suberviola and Antonio Giménez Valverde, our medical experts, for their collaboration and hard work in the creation of the Basic First-Response game. Moreover, we appreciate the trust and effort of CATEDU (www.catedu.es) in commissioning us for the creation of this game and their support in the process.

References

[1] C. Aldrich, *Learning by Doing: A Comprehensive Guide to Simulations, Computer Games, and Pedagogy in e-Learning and Other Educational Experiences*, San Francisco, CA: Pfeiffer, 2005.

[2] L.P. Rieber, “Seriously considering play: Designing Interactive Learning Environments based on the Blending of Microworlds, Simulations and Games,” *Educational Technology Research and Development*, vol. 44, 1996, pp. 43-58.

[3] M.D. Dickey, “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*, vol. 54, 2006, pp. 245-263.

[4] J. Chen, “Flow in games (and everything else),” *Communications of the ACM*, vol. 50, 2007, pp. 31-34.

[5] R. Garris, R. Ahlers, and J.E. Driskell, “Games, Motivation and Learning: A Research and Practice Model,” *Simulation & Gaming*, vol. 33, 2002, pp. 441-467.

[6] D.R. Michael and S.L. Chen, *Serious Games: Games That Educate, Train, and Inform*, Muska & Lipman/Premier-Trade, 2005.

[7] J. Torrente, Á. Del Blanco, E.J. Marchiori, P. Moreno-Ger, and B. Fernández-Manjón, “<e-Adventure>: Introducing Educational Games in the Learning Process,” *IEEE Education Engineering (EDUCON) 2010 Conference*, Madrid, Spain: IEEE, 2010, pp. 1121-1126.

[8] S. Wexler, K. Corti, A. Derryberry, C. Quinn, and A. van Barneveld, *The eLearning Guild: 360 Report on Immersive Learning Simulations*, 2008.

[9] E.J. Marchiori, Á. del Blanco, J. Torrente, I. Martinez-Ortiz, and B. Fernández-Manjón, “A Visual Language for the Creation of Narrative Educational Games,” *Journal of Visual Languages & Computing*, vol. 22, Dec. 2011, pp. 443-452.

[10] P. Moreno-Ger, J. Torrente, J. Bustamante, C. Fernández-Galaz, B. Fernández-Manjón, and M.D. Comas-Rengifo, “Application of a low-cost web-based simulation to improve students’ practical skills in medical education,” *International Journal of Medical Informatics*, vol. 79, Jun. 2010, pp. 459-67.