Available online at www.sciencedirect.com

ScienceDirect

Procedia - Social and Behavioral Sciences 228 (2016) 443 – 448

Procedia Social and Behavioral Sciences

2nd International Conference on Higher Education Advances, HEAd´16, 21-23 June 2016, València, Spain

iLearnTest – framework for educational games

Ana C.R. Paiva[ab], Nuno H. Flores[ab]*, André G. Barbosa[a], Tânia P.B. Ribeiro[a]

[a]*INESC TEC, Portugal*
[b]*Department of Informatics – Faculty of Engineering of the University of Porto, Portugal*

Abstract

Games are a part of human life since ancient times, present not only at childhood but throughout most of our adult life. A growing area of research focuses on the development of games for teaching and learning in various areas of expertise. These are called serious games. They intend to capture the attention, to motivate and to encourage user engagement through the use of recreational and entertainment elements, thus facilitating the learning process. This article describes iLearnTest, a framework for developing serious online games which are capable of supporting education of several subjects, allowing for a suitable training of participants, thus preparing them to meet the job market needs. This paper describes the architecture of the framework, the structure of the game and presents some results from performed experiments as to validate the overall approach.

Peer-review under responsibility of the organizing committee of HEAd´16

Keywords: serious games; iLearnTest.

1. Introduction

For a long time now, games are a part of human life. They are present not only in our childhood, but throughout adulthood. Over the past few years, studies have shown that the use of games, as a complement to traditional learning, is much more efficient than just using the traditional teaching method (Thirty, Zoucas, & Silva, 2011) (Schneiderman, 2004). Games can be effective educational tools, since they are fun, they motivate the user, facilitate

* Corresponding author.
E-mail address: nuno.flores@fe.up.pt

Peer-review under responsibility of the organizing committee of HEAd´16
doi:10.1016/j.sbspro.2016.07.068

learning and increase the storage capacity of what was taught, exercising the mind and intellectual thoughts. When games are used in this context, they are defined as serious games (Kapp, 2012).

Besides engaging and fun, serious games give the user the opportunity to learn theoretical concepts at their own pace, receive a score for their performance and conduct practical exercises as a game, improving and supplementing their learning process in a specific area. Serious games are the subject of increasing interest (Kapp, 2012) (Mitchell & Savill-Smith, 2005) (Mayer, 2014), and consequently, new technologies have emerged to assist in the development of such games.

There is already a good amount of solutions out there for game development, but they are rather generic when it comes to developing specific serious (e.g. educational) games, not coping with structure reusability when only its (learning) contents change. Examples of platforms that allow the creation of 2D games for diverse platforms (web, mobile devices, computers) are **Stencyl** (Stencyl, 2014), **Construct 2** (Construct2, 2007) and **GameSalad** (GameSalad, 2011). These systems provide a "drag and drop" interface to facilitate game development using a set of building blocks, assisting users with no programming skills. Even so, there is the option of coding new features and complex behaviors for more advanced users. **Unity** allows building 2D and 3D games for web, desktop, consoles and mobile devices platforms (Unity 3D, 2014). From the moment you open the Unity editor, the user can instantly import any type of resource and create content for complex worlds with building blocks.

The main benefit of these frameworks is to allow the development of games without writing code. However, they come with a steep learning curve, and without "templating" for reuse, that is, the game developer has to rebuild everything from scratch when changing the teaching subject.

2. iLearnTest

The iLearnTest is a framework to build games more easily, not intended to replace traditional teaching, but instead to provide an additional option to help the educational process, thus increasing the interest of the students to the subject being learned.

The iLearnTest contains several features that makes it attractive to teachers and students:

- Provides a set of game templates that facilitate the game construction;
- Separates the game content from the game implementation;
- Incorporates challenges to promote the engagement of students;
- Allows students to learn at their own pace;
- Gives feedback about the score achieved and the correct and wrong answers in each game so that students can aspire to get the highest score on the next attempt.

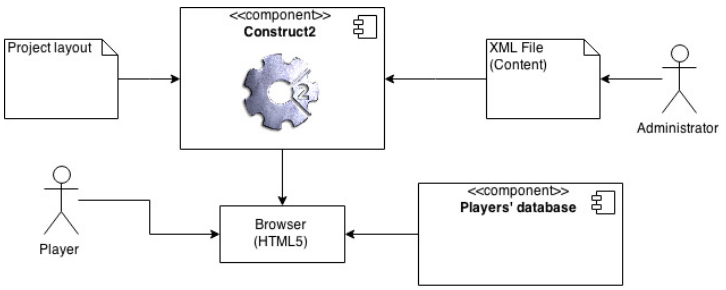

Fig. 1. Architecture of the iLearnTest.

The iLearnTest (Figure 1) is developed on top of Construct2 (Ribeiro & Paiva, 2015), as this offers certain features which protrude relative to the other technologies studied. Construct2 provides a XML plugin that can parse and read data from XML documents and uses XPath[†] to access the XML document.

The goal of the iLearnTest framework is to speed up the process of building a new game by providing game templates that can be reused to learn different subjects. The contents of the entire game is provided in a XML file (read via AJAX) and, after that, the game generation process is automatic. The framework generates the game in HMTL5 that can be accessed simultaneously by several players through the browser.

In order to embed the resulting game in a website it is necessary to include an *iframe* in the webpage's HTML code. An *iframe* is essentially a portal to another webpage so it allows opening up and displaying multiple pages within one webpage. More technical details can be found in (Barbosa, 2015).

The iLearnTest framework is available online with authentication access and has a database (Figure 1) with the information of the players and their scores. The scores of each player is shown in the web page of the game which promotes competition among them, thus contributing to enhance learning.

2.1. Content structure of the game

The iLearnTest content is structured as a tree with three layers: the top level with six chapters (Figure 2), the middle level with 36 subsections (six for each chapter) and the bottom layer with 216 subsubsections (six for each subsection of the middle layer). The reason why this limitation happens is that the contents must be read into existing game elements in order to access their properties through their unique IDs. However, it is perfectly possible to define a structure with less chapters and sections than the maximum values presented.

Each chapter (and subsections) may have several pages, i.e., it may have pages for theoretical content and challenges that students can play in order to practice the theoretical concepts. The XML file with the content of the overall game is structured accordingly (see, for instance, Figure 3b). It contains the chapters and subsections, theoretical content (i.e., learning objectives and theory) and challenges.

Figure 2. Main menu screen (top layer).

The iLearnTest provides a set of game templates (here called challenges) that can be reused in different projects to teach different subjects. For instance, the challenge shown in Figure 3a can be used to split lower and upper letters or to split odd and even numbers.

The set of challenge templates provided by iLearnTest is:

Guess Concepts: This challenge displays empty slots for the letters of a word that is the answer to a question. The player has to discover such answer/word. If he does not know the answer he can ask for help. In this situation, the game provides some hints showing a letter of the word and decreasing the possible final score.

[†] A query language for XML, much like SQL used for relational databases.

Split into two sets: The student has to drag-and-drop elements on the screen to two different boxes that split the elements in two different concepts (see example in Figure 3a).

Select items: In this challenge, the student has to pick the elements that are "falling" on the screen and that correspond to a correct concept within a specific context and should ignore the ones that are incorrect.

2.2. Playing the game

The first window of the game (Figure 2) has six colorful platforms (corresponding to the chapters of the top layer) that give access to the subsections. A similar screen is shown for the other subsections (middle and bottom layers).

The navigation through the platforms is controlled by the arrow keys. To move to the layer below, the user needs to put the "avatar" on top of the platform and press "down" on the keyboard. Consequently, the "avatar" submerges and falls on another page that provides access to the contents of the selected chapter or section.

While the "avatar" is on top of a platform, it displays a balloon with the appropriate chapter heading (Figure 2). Each platform also presents the score obtained by the user so far (numerator) versus the total points a player can get playing all the challenges of each subject (denominator). The white platform ("Go Back" in Figure 2) is present on every page to allow the player to return to the previous page, or if in the home menu to finish the game.

Within each chapter or section there is a theoretical introduction followed by one or more pages with the challenges that the students should play in order to practice the acquired knowledge.

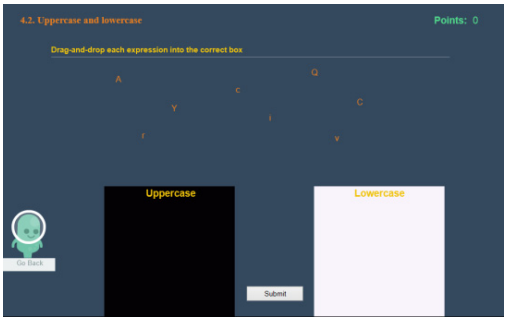

Fig. 3. a) Split lower and upper letters

```
<subchapter title="4.2. Uppercase and lowercase" points="16">
    <learningobjectives>
        <learningobjective number="LO-4.2.1">Objective1</learningobjective>
        <learningobjective number="LO-4.2.2">Objective2</learningobjective>
        <learningobjective number="LO-4.2.3">Objective3</learningobjective>
    </learningobjectives>
    <texts>
        <paragraph>paragraph1</paragraph>
        <paragraph>paragraph2</paragraph>
    </texts>
    <games>
        <dragdropboxes black="Uppercase" white="Lowercase">
            <word box="white" >r</word>
            <word box="white" >o</word>
            <word box="black" >A</word>
            <word box="black" >Y</word>
            <word box="black" >Q</word>
            <word box="black" >C</word>
            <word box="white" >v</word>
            <word box="white" >i</word>
        </dragdropboxes>
    </games>
</subchapter>
```

Fig. 3. b) XML file with the content required

3. Validation

To evaluate the effectiveness of the iLearnTest as a framework to develop new games and as a game for learning, we designed two different experiments.

3.1. First experiment

Sixteen master students, acting as "teachers", had to construct a challenge to split lower and capital letters (Figure 3). These students were finalists of the Master in Informatics and Computing Engineering (MIEIC) at Faculty of Engineering of the University of Porto (FEUP). The goal was to evaluate the time taken to develop such game using only Construct2 and compare it with the time taken to build the same game with iLearnTest. As such, the students were divided into two groups. The objectives were explained to all students and the time taken to achieve the goal by each team was registered. Apart from the overhead time needed to install and learn how to work with both tools, the students using iLearnTest spend approximately 5 times less implementing the game than the students using only Construct2.

3.2. Second experiment

Besides evaluating the framework from the "teacher" point of view, an experiment was also performed from the point of view of the "student" in order to assess the usefulness of iLearnTest in the learning process.

In this experiment, the iLearnTest contained materials to teach/learn Software Testing concepts and techniques. This content was based on the Foundation Syllabus (ISTQB, 2011) provided by the ISTQB (International Software Testing Qualification Board). iLearnTest was configured with 6 chapters similar to the structure of such syllabus (ISTQB, 2011). In each chapter the students had to read some theory and then they had a challenge to play in order to practice the learning concepts. For instance, in the case of learning the difference between *black-box* and *white-box* test design techniques, the students should play the game template presented in Figure 3a, but this time with test case design techniques on the screen instead of letters and black-box and white-box as labels in the two different boxes.

Having iLearnTest loaded with the proper content about software testing, the experiment took eighteen students of the fifth year of the MIEIC at FEUP, with no knowledge in the Software Testing area, as subjects. These students were randomly divided into two groups (Group A and Group B) of nine elements each. The study materials available for the two groups was different: Group A had only access to the Syllabus of the ISTQB and the web; Group B was allowed to use the iLearnTest, Syllabus and web. At the end, both groups underwent an examination.

After explaining the purpose and procedure of the experiment, a small demo (2 minutes) on how to navigate through the iLearnTest menu was presented to students of Group B. All the students were given a list of learning objectives to be evaluated in a final examination.

All students had an hour and forty-five minutes to study where, upon completion, all study materials were collected and the final exam (in the form of a survey) was distributed among the students. The survey consisted of multiple choice questions, similar to the ISTQB certification exams, however covering the learning objectives that have been studied and that were a subset of the syllabus. For thirty minutes the participants had to answer twenty-one questions. In the end, the exams were collected and the results were analyzed.

During study time, observation of the students in their learning process was conducted. Group A was highly focused, taking notes of topics considered most important, reading all subjects indicated in the Syllabus provided and taking ISTQB certification exam examples found on the web. On the other hand, group B was more excited with the game challenges of the iLearnTest, becoming distracted enough to miss the first few minutes of understanding how the game worked, while having fun with the "avatar", and then playing all the challenges of the different chapters regarding different subjects.

Table 1. Results of the experiment.

Group	Final exam mark	Average of the marks in MIEIC
A	16.3	15.7
B	16.5	13.3

The obtained results are shown in Table 1 (in a 0 to 20 mark scale). It compares the two groups's final exam mark, with the average of the group's students marks, obtained during the courses attended at MIEIC. Although the students of group B had an average mark at MIEIC (13.3) 2.4 lower that group A (15.7), the final mark at the experiment's exam was better. Group B achieved a 16.5 mark, while group A achieved 16.3. (0.2 worse). The final marks are not very different but their average in MIEIC is lower (13.3) so it would be expected to have a lower mark in the final exam. Thus, possible explanations may be the use of iLearnTest and the fact that group B showed more enthusiasm and motivation throughout the experiment, favouring a more enjoyable learning process. Therefore, it rendered a higher mark in the examination despite the average of the marks in MIEIC being lower.

4. Conclusions

This article presents a framework, iLearnTest, for developing games for leaning. This framework provides reusable templates, allowing the same gaming challenges to be used, for teaching different subjects in an easy way. This article presents the details of the framework and two experiments performed to validate the overall approach. Results showed that iLearnTest may be useful in both contexts: to build new games easier and to help the learning process.

In short, the validation of the iLearnTest was positive and showed that the resulting games are effective in transmitting software testing expertise. The use of this new study method proved to be even more efficient than the mere use of just the Syllabus.

Throughout the experiment it was possible to notice that the students using iLearnTest were more motivated and got the best results in the exam performed at the end of the experiment. In addition, the students mentioned that iLearnTest was really useful because it provided immediate feedback allowing to better understand the concepts.

As future work, further experiments are to be performed, with more students and other subjects besides software testing.

Acknowledgments

This work is financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme within project «POCI-01-0145-FEDER-006961», and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) as part of project «UID/EEA/50014/2013».

References

Barbosa, A. G. (2015). *Platform for Educational Games Generation (MSc thesis).* Faculty of Engineering of the University of Porto.

Construct2. (2007). Retrieved from Scirra: www.scirra.com

Foudation Level Syllabus. (2011). ISTQB.

GameSalad. (2011). Retrieved from www.gamesalad.com

ISTQB. (2011). *Certified Tester - Foundation Level Syllabus.*

Kapp, K. M. (2012). *The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education.* Pfeiffer.

Mayer, R. E. (2014). *Computer Games for Learning: An Evidence-Based Approach.* MIT Press, Cambridge.

Mitchell, A., & Savill-Smith, C. (2005). *The use of computer and video for learning - A review of the literature.* London: Learning and Skills Development Agency.