

Education Apps. One Step Beyond: It's Time for Something More in the Education Apps World

Francisco J. Cardenal and Vivian F. López

Abstract In this publication the key parts of the process undertaken to develop an educational fun math app-centric world are collected. The key points in the design, idea, analysis, development and testing of an App that combines learning with fun.

Keywords App · Application · Mobile · Apps · Education app · Blended education · Math

1 Introduction

Education Applications are one of the categories that more increase in the Apple App Store or App Store. It's a growing market that allows us to gain knowledge of many subjects, including mathematics. To prove it, you just have to find the Education category in the App Store. The market of education Apps is a continuously growing industry and becomes one of the few sectors whose numbers improved every year. According to Forrester Research forecasts [1], the mobile device market will triple in 2017. Focusing on the education section, there are several schools that use mobile devices to base their learning. The first time we heard of it was a Scottish school, but now only the public schools of San Diego uses 20,000 mobile devices. According to Apple itself, in late 2013 the company of the bitten apple sold a whopping 8 million mobile devices exclusively for educational institutions [2]. The latest data indicate that there are more than 80,000 education apps on the market. The education Apps accounted for more than 9 % of the total [3]. Over 80 % of the applications in the Education category of the iTunes Store were aimed at children. In 2009, 47 % of the

F.J. Cardenal (✉) · V.F. López
Departamento Informática Y Automática, University of Salamanca,
Plaza de La Merced S/N, 37008 Salamanca, Spain
e-mail: pacocardenal@pacocardenal.com

V.F. López
e-mail: vivian@usal.es

best selling Apps were intended for children. Now, that percentage is 72 % [4]. The percentage of Apps for children continues to increase, while the adult market declines. The Apps for Preschoolers occupy 58 % of all categories and have increased by 23 % [5].

The main objective of this work is the development of a specific application that covers a significant part in mathematical terms of the current formal education: working with decimals. The whole process to reach that goal will be discussed in this article in the following sections.

In Sect. 2 we present education apps, Sect. 3 provides a short overview of the actual market of education apps. In Sect. 4 we present our education app. Section 5 explains the conclusions of the paper.

2 Education Apps. The Study

The target of the App is a child of about 9 or 10 years, beginning with the use of fractions and decimals. Choosing the iPhone as target platform has been made by the boom that mobile devices are having on our society, using them, among other fields, for education. This is also the cause that sparked our interest in the software development for these devices.

After deciding the development of this application, the first step was to communicate with people who could guide us both in education (and more specifically children education), and developing education Apps. Specifically, we communicated with the creators of education applications Teachley [9] (Apple Design Award winners at WWDC 2014), who gave us very useful notions and concepts to develop our App. They have confirmed what we have indicated earlier: learning is more effective if the student, in this case the child, is involved. This is what stimulates their attention and the reason that teaching a particular topic or subject becomes part of their leisure and approaches it more like a fun as an obligation. After several meetings with them, the following conclusions were drawn:

- The application should be entertaining. This is the main point of the whole project and we wanted to develop an application that does not reach the level of game, but it will be very close to it.
- The user interface should be simple. The third pillar on which the application would be based was the simplicity and clarity of the user interface. The display should show only the essentials and make users agree to it clearly and quickly as shown in Fig. 1.

On the development of education applications, the following lines were drawn to follow in the application:

- Simplify. The less the better.
- Increasing the degree of difficulty of the App from a below average user level to a higher level.

Fig. 1 The game inside the App



- Perform any readily observable modification when a change in a difficulty level is performed.

The first tip led us to simplify the interface as simple as possible, the second to manage the complexity of the questions asked in the application as we will discuss below, and the third to create a separate file for modifications of each difficulty level present in the project. Thus, children are offered a new way of teaching and a new way of learning. The concept of learning through play forms the basis of the entire project and how to improve education, children education in this case.

3 The Actual Market of Education Apps

All parents are concerned about the poor quality or lack of interest in their children's early school stages. Ludic education applications can be a way of improving education today. To achieve this, these applications must be a real mix of play and learning. Many of the applications currently on the market are little more than colored numbers that help the education of the children of the house calling his attention.

Most educational Apps that can be found on websites like Eduapps [6] or eduTecher [7] are very simple Apps without a defined goal that can be applied to formal education. Likewise occurs with the iCuardenos App [8] or the awarded Teachley Apps [9]. Neither offers to cover several courses comparable to current child formal education levels in Spain.

With all the above facts, it is not unreasonable to say that the market of educational fun apps for kids is an emerging market, offering a good opportunity for the future and values of stability and more than enough benefits. The education market needs a change. Parents need a change. Children need a change. And education Apps could be this expected change in educational quality and quantity.

4.1 Application Features

Unified Process

The Unified Process is a generic framework for software development. Its main advantages are: standardization, flexibility and adaptation. As fundamental concepts emphasize discipline, which is used to organize key management and development activities, and artifacts from the results. The various stages of that process are listed below: initiation, development, construction and transition. In the requirements phase different use case diagrams have been created to reflect the system's functionality. Figure 3 shows the use case diagram *Home Menu Selection*.

Also, these phases are combined to form an engineering stage and another stage of production. Returning to the term of discipline, used to organize the management and development activities, the various disciplines are: requirements, analysis, design, implementation and testing.

Sprite Kit

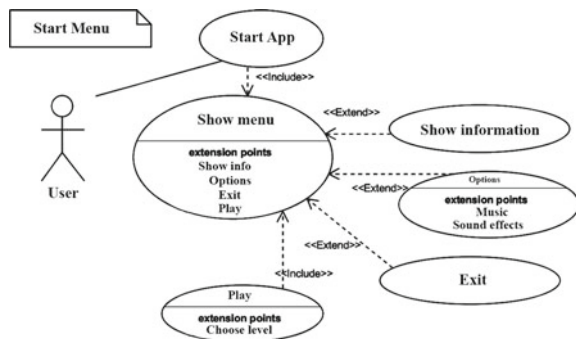
Sprite Kit is a framework [10] that offers both graphical capabilities like animation and sound to deal with all matters relating to the graphic elements of a game developed for mobile devices that have the Apple iOS 7 operating system or higher. This way we can create a fairly complex game or application in a simple, rapid and standardized way.

Sprite Kit is much easier and faster to use than engines like Cocos2D and Unity, and becomes the perfect place to develop applications that require high interaction from the user, games especially, by new developers and programmers. Sprite Kit is available for both iOS 7 and OS X 10.9 and above and supports different types of items.

Game design

In terms of design, the first step was to create the class diagram. Most classes descend from one of the most widely used classes in Sprite Kit: the Scene class [11], which is the basis of many of the screens appearing in the App. The class that focuses so much time spent with the application as the time spent on its

Fig. 3 Use case diagram "Home Menu Selection"



development is the Game class, which will be carrying most of workload game included in the app. To help complete the design we also developed the activity diagram where the activities performed by the application are observed.

The design has been one of the project star phases, with the longest run. The reason has been the quest for the best graphical interface and the most suitable items and pictures optimized for the device screen. The design goal was to make an application with a simple look but both attractive and intuitive for use again. In terms of design, we have two distinct parts: interface design and game design.

Interface application design

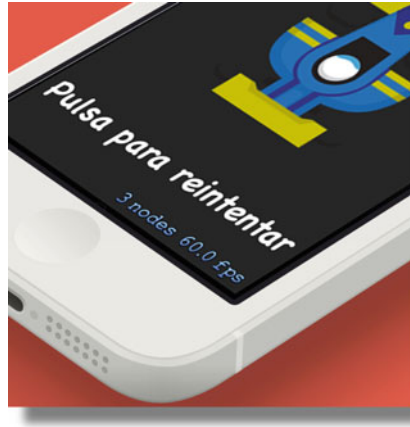
The application screens should show all-important information without data that overload the interface. Due to the size of the device, everything shown on the screen should be large enough to display it. We wanted to design an interface that was attractive but not too overdone because, although there is a game running underneath, the area where the user should focus is the panel where the questions appear. An important aspect of the interface was interactivity. Although the user should not press more than true or false button, he simply must feel that his choice was used to control his character, in this case his vehicle. It is that sense of connection between a typical touch application, where only a series of buttons are pressed, and a game controlled by movements of the device, which has satisfied us with the result achieved.

As for the different parts of the interface, we have two very distinct parts: the game running underneath, and the application of true false that sits atop. To leave the maximum free space available on the screen to ask questions and the characters roam the greatest possible route, we have placed all the controls in the lower parts (buttons True-False) and higher (indicating question number, time and remaining lives). Furthermore, to give the game more movement, we decided, in addition to vehicles, encourage the bottom of each level to give a greater sense of mobility, so that the bottom of each level occupies 50 % more height screen device to which it is intended. Through a photographic image processing used with small changes and tweaks, an acceptable enough interface to enjoy the lighter side of the application was achieved.

Interface game design

The lighter side of the application is the biggest part of the design stage and therefore the part in which more time has been used. At each stage of this design have been tested using children in that age range for errors, problems or potential upgrades. Other point within the project has been the digital image processing, a key part in the development of games and it has led us to acquire new knowledge in this field. Report all images used in the application by modifying original images were created with free license, but not all of them allow distribution of the application for benefits. To get the feel of movement, we need a sequence of still images that will generate an animated image [12]. In turn was due to take extra care with background, colors and contrasts as seen in Fig. 4, due to the small screen of the device, which could lead to a bad user experience. Indicate that, in addition to graphics, the game includes both music and sound effects that acclimate the activity and provide feedback to the user.

Fig. 4 Color and contrast on screen



Architecture

The application was developed using the Model View Controller pattern MVC, which isolates presentation and data. With Sprite Kit and iOS 7 is very easy to follow this pattern. The MVC [13] consists, as its name indicates, of three pillars:

- **Model:** responsible for the business logic, the part of algorithms and data. The model can be a single object or be a multiple structure. Represents the information or knowledge model.
- **View:** responsible for presenting the view or interface. The view is the visual representation of the model. The view is related to the model through the controller, and it is thanks to this through which they can communicate.
- **Controller:** driver intermediary between Model and View and responsible for all notifications. It is the real link between the user and the system, and is responsible for managing the inputs and outputs of the application.

Implementation

Due to the novelty of the Sprite Kit framework, the available documentation is hard to find and the proposed examples are not as frequent as with other advanced graphics engines. For this reason, we have relied on examples found in the bibliography appended to implement a project consisting of automatic generation questions for each difficult level, changing parameters of each level, sprite animation, position on screen and tap detection. Furthermore, it has different levels adapted to play a major difficulty. To provide the application addictiveness feature, it was tried to adjust the difficulty as possible in relation to the speed of the opponent vehicle and the category of mathematical operations. This adjustment was made based on the evidence we commented afterwards.

Evolution

The application has been developed following an iterative model, so that changes and increases in the application have been many and varied. Project started focusing on the lighter side of the application. After get it, we started to develop a graphic

interface that allows its proper use. After that, the different application screens were incorporated, linked and provided with new graphics. Finally both logic and presentation were refined and the most interesting questions were selected and adapted for each difficulty level.

We want to reiterate that the application was initially designed to develop using Cocos2D [14], but the App evolved to Sprite Kit, for being, in our opinion, a more suitable environment, and for being Apple's own environment. Once it was decided to use this tool, the first thing to be achieved was to unite a specific learning and a fun experience. To achieve it, we searched both educational and recreational applications and different material of studies on this subject, which led us to the foundation of the application: to face an opponent and reach the finish line before him using our knowledge.

After considering several options such as multiple choice answers, direct confrontations with other types of games or eliminating wrong answers, we chose to make an application based on true-false, but with wide penalty of random correction, thus avoiding fast and ignorant response. This true-false application was very simple in the beginning, showing a series of questions consecutively, and default iOS buttons. After this first model, we proceeded to customize buttons, expanding the number of questions and generate them in random order. Finally, we implemented the load of questions from an external file.

The next phase would determine the game associated with this other game of true false. We had to develop a layer that run below the principal and maintained addiction and learning. After evaluating different options as horse racing games or board games, we chose a drag car race, but customizing them with formula 1 cars, for being a known sport, adapting their colors to red and blue representing the Ferrari teams (with Fernando Alonso as a representative of our country) and several of its rivals teams. In this game our character was only showed at first. A small portion of the screen represented a road on which our car was circulating to reach goal. The implementation of the victory was not done by position, but right questions because they always represented the same distance that the car run on the screen.

Then we proceeded to parameterize the distance traveled depending on how quickly the user selected the correct answer. It thus had a variation in the distance that moved the character and didn't allowed us to determine their position by the number of correct answers, so this algorithm was modified to depend on the position on the screen. To complete this application layer, adding an opponent character to play against increased the thrill of the game. The first time it was added, the opponent is moved when the user answer a question, moving the two vehicles at a time. However, it was understood that it would be a more challenging situation to see how the enemy was moving across the screen while the user was trying to solve the proposed question. This was the technique chosen for the final version of the car race layer.

Upon completion of both layers, we had to relate them. To do this, we linked the appropriate action to each of the buttons with the movement and distance traveled by the vehicles due to information of response time remaining and the speed of the

opponent that resides in the LevelData.plist file. Furthermore both this distance as that traveled by our vehicle was optimized so that the maximum number of possible questions were displayed on the screen.

4.2 Experiments and Results

Tests

The tests were conducted at an early stage in the simulator itself provided by the Apple programming environment, and then we installed the application on a real device (iPhone 4) to be tested by different people. Before going into the test section, we would like to stress that in order to test the application on a physical device and not just into a simulator, you need to have an Apple's iOS developer license, which was obtained by integrating us into the iOS development program of the University of Salamanca and the friendliness and helpfulness of José Rafael García-Bermejo Giner (Coti).

As we said before, we had a number of children to test the application and the support of their parents to learn from a more adult perspective. In these tests several bugs and small errors in the game were discovered. In tests we had specifically 4 children with the next ages: 8, 9, 10 and 12 years old. Recall that the target user of the application is a child between 9 and 10 years old. The choice of these ages between 8 and 12 years was conducted to have a clear idea of whether both interface as difficulty was adjusting correctly.

Difficulty tests

In the first tests only a mathematical question appeared on screen and you should answer true or false. They performed a total of ten questions and we determined both the number of correct answers and the time taken to resolve them. The questions were chosen at random from a set of 30 different levels of low, medium difficulty, where addition and subtraction of whole numbers and decimal numbers was included. These tests were basic to adapt the difficulty of the different levels in the application.

Interface tests

The second objective of this test group was helping us to made a sufficiently attractive and intuitive App. Everybody should access to any of the screens and options naturally and play the game without explanation. During these tests there was no help screen, so the children had no way of knowing how it worked except for practice.

The first product delivered to them, had only the game screen with a similar interface to the final one, and where they should always answer the same 10 questions that appeared in random order. The result was more than satisfactory because, thanks to addiction caused by the game, they wanted to continue using the application until it was not necessary, and they evolved largely in their response times.

As to what concerns us, the interface, initially it had a true or false buttons similar to accept or cancel options present in the iOS 7 mobile operating system.

However, these buttons were not sufficiently intuitive and we had to explain several times what that meant and how they should respond in each case. In addition, the game screen in which our vehicle was moving (then we had not an opponent vehicle), was between True-False buttons and the text of the question, which appeared at the top of the screen. Being such a short distance, the look of the game was not as great as in the final version.

The third set of tests was performed with the modified interface very close to the final version. It was found that the user response was good in both management and addictiveness, so it was decided to keep this interface to the final product.

Results

This results was taken from a group of ten children between 10 and 12 years old who answered 30 of 100 random math adding and subtracting operations. In first place in a notebook and, in second place, within the App.

The children answered correctly to 65 % of the questions in the traditional way (only with the notebook) and 83 % with the App. Then, we used the App first, and the results were 90 % of correct answers using the App and 76 % with the hand-writing way.

We repeated this tests ten times, and the results with the App were an average of 14 % better than the same kind of operations answered on a notebook, without any game pushing the children to improve their results. Also, the response time was 30 % better with the use of the App.

5 Conclusions

In this paper, we have presented an App that combines learning with fun. The children's education Apps that are now on the market are all aimed at improving small educational or psychological aspects of the children to whom they are addressed.

We have described the current state of the work but there are many open lines of work. We have finished the touch and feel of the App, defined the structure of the cases and the game models, and we have been working on alternatives of storage and variations of the main racing game.

Current lines of work are focused on new cognitive science research. We want to capture meaningful data for teachers about what to teach next. Our intention is to help students in their learning process and to help teachers in their teaching process as well.

We have seen improvement in children's mental calculus and we think that this kind of Apps will be an important part of formal education in a future. We want it and we will work hard to achieve it.

Acknowledgments This work has been supported by the Spanish Government through the project iHAS (grant TIN2012-36586-C01/C02/C03) and FEDER funds.

This App was awarded with the TCUE-5 prize [15] at University of Salamanca.

References

1. Gownder, J.P., et al.: Global business and consumer tablet forecast update, 2013 to 2017. Forrester Research forecast. <http://bit.ly/1gujliI>. Accessed 16 Mar 2015
2. Apple Inc. Apple education web page. <https://www.apple.com/es/education/>. Accessed 16 Mar 2015
3. Universia España. Ya existen más de 80.000 aplicaciones educativas. <http://bit.ly/1L5ES2T>. Accessed 16 Mar 2015
4. Universidad de La Rioja. Aplicaciones educativas para dispositivos móviles. <http://bit.ly/1GMfDCY>. Accessed 16 Mar 2015
5. Common Sense Media Inc. Best Preschool Apps. <https://www.commonsensemedia.org/lists/best-preschool-apps>. Accessed 16 Mar 2015
6. Raúl Santiago y Alicia Díez. Eduapps, aplicaciones educativas. <http://www.eduapps.es>. Accessed 16 Mar 2015
7. Bellow, A.: Educational technology web tools. <http://www.edutecher.net/>. Accessed 16 Mar 2015
8. Rubio. iCuadernos by Rubio. <http://www.icuadernos.com/>. Accessed 16 Mar 2015
9. Teachley, LLC. Research, learning, fun. <http://www.teachley.com>. Accessed 16 Mar 2015
10. iUridium. Sprite Kit tutorials. <http://www.sprite-kit.com/>. Accessed 16 Mar 2015
11. Volevodz D.: iOS 7 Game Development, ISBN:978-1-78355-157-6 (2014)
12. Penn, J., et al.: Build iOS Games with Sprite Kit, ISBN: 978-1-94122-210-2 (2014)
13. Conway, J., et al.: iPhone Programming, ISBN: 978-0-32194-205-0 (2014)
14. Cocos2d-x.org. Open Source Game Development. <http://www.cocos2d-x.org/>. Accessed 16 Mar 2015
15. Fundación Universidades y Enseñanzas Superiores CYL. Transferencia de conocimiento Universidad empresa. <http://www.redtcue.es/>. Accessed 16 Mar 2015