Using Mobile Serious Games for Learning Programming

Jiayi Zhang, Joan Lu

School of Computing and Engineering
University of Huddersfield,
Huddersfield, West Yorkshire, UK
e-mail: {zjybyron@gmail.com, J.lu@hud.ac.uk}

Abstract— **Serious games, which are entertaining games for learning, have attracted obvious attention in the field of education, and the mobile educational game is an important issue in recent years. The purpose of this article is to propose and evaluate iPlayCode which is a mobile serious game for learning programming. The Felder-Silverman Learning Style Model (FSLSM) and Kolb's experiential learning theory have been used in this game to improve learning efficiency. In addition, simple interface design improves user experience, and the use of game elements eliminates the tension in learning programming. To sum up, iPlayCode incorporates rigorous teaching methods, scientific interface design and practical game elements.**

Keywords-serious game; mobile; learning programming

I. INTRODUCTION

In recent years, serious games, which are entertaining games aimed at learning academic knowledge, have received obvious attention; Mitamura et al. explained that serious games have found great success in improving the learning environment and learning efficiency [1]. In addition, mobile devices are similar to earlier computers, and even now, ordinary computers in their hardware performance. They provide favourable conditions, meaning that people can develop serious games for mobile devices based on mobile operating systems, such as iOS, Android and Windows. Xin emphasised that numerous entertaining games have been developed for mobile learning, such as SimCity, E-Friction, Civilization, and so on [2].

Kapralos argued that because of the rapid development of mobile serious games, it should be determined that they are reasonably designed to achieve the expected objectives [3]. Prensky found that serious games must combine two powerful factors for success in the field of education: one is the key purpose of the game, in which a learner can be pushed into a learning environment unconsciously in the game; the other is the integration of learning methods which enhance the teaching efficiency and reduce boredom from learning at school [4].

iPlayCode, which we developed, aims at simplifying the learning of programming by including entertainment. Mitamura et al. highlighted that learning programming languages is abstract and complex at university-level because students do not feel that the learning environment is motivating and are not interested in computer programming [1]. Therefore, teaching methods play an important role in the educational process, and mobile serious games provide

an opportunity to resolve this issue. The purpose of this article is to propose and evaluate mobile serious games for programming learning based on the learning model.

The rest of the paper is organized as follows. Section II provides the background and problems of serious game. The positive combination of game with learning theories is discussed in detail in Section III. The interface design and implementations are demonstrated in Section IV and Section V. The assessment results are then shown in Section VI, followed by the conclusion in Section VII.

II. RELATED WORKS

Programming languages are used in every corner in the field of computing. But Khenissi et al. argued that teaching and learning programming languages are definitely not a simple task for teachers and students [5]. Learning a programming language requires meticulous logical thinking and students should understand abstract concepts from real problems. Moreover, computer science students come from various academic backgrounds, and, especially, there are ones without basic computer knowledge. Therefore, the development of serious games to improve learning efficiency and motivation is a common topic for many researchers and developers.

A. Serious Game

According to Lu et al. [6], serious games have received significant concern in the field of education, and a mobile educational gaming community has been formed in recent years. Apple announced that the value of the application for iPad and iPhone is more than $10 billion in the App Store in 2013. Furthermore, App Store has more than 65,000 educational applications, primarily intended for iPad, which cover a wide range of fields suitable for different levels and learning styles [7]. Serious games help students gain experience in practice, which resulted in their rising popularity. Furthermore, a serious game is a combination of education and entertainment. Students can discover knowledge in entertainment and refresh their learning attitudes.

B. Fun and Learning

Franzwa et al. state that the biggest challenge for serious games development is the combination of game ideas and teaching concepts [8]. Many educational games focused on educational methods but ignore entertainment sense; on the other hand, numerous entertaining games emphasize means

of entertainment yet reduce educational elements. The authors believe that the evaluation of a serious game should depend on the efficiency of learning. Serious games would be more easily accepted than traditional teaching methods and students could learn better through serious games than by themselves.

C. Application Platform

According to devices used, serious games are divided into three categories, which include computer games, phone games and tablet games. While most serious games with high system requirements are used on the computer, mobile devices run various serious games with low system requirements. In addition, websites are also a very popular serious game media for cross-platform development [6]. Their advantages and disadvantages are summarised in TABLE I.

TABLE I.   A COMPARISON OF SERIOUS GAMES PLATFORM

	Computer	**Phone**	**Tablet**
Storage	Enough	Medium	Medium
Processing Capacity	Strong	Medium	Medium
Operation	Complexity	Simplicity	Simplicity
Interface Size	Large	Small	Medium
Game Effect	Good	Low quality	Medium
Mobility	Bad	Very strong	Strong

This table shows that computer-based games are designed to provide perfect game conditions, but they are limited with certain situations. Moreover, mobile devices, phones and tablets are comprehensive and their characteristics are well-suit to the requirements of mobile serious games.

D. iPlayCode

This application was built for learning programming and the target audience are learners with no programming skills. Through the syntactic judgments and help system, learners can raise their learn how to write correct syntax based on C++ programming language. At the same time, iPlayCode aims to become a serious game that incorporates elements of the game into the learning environment, such as incentives and time constraints.

III. GAMES BASED ON LEARNING

The study by Zapusek et al. [9] found that educational games can increase motivation for learning. However, the quality of learning not only needs motivated learners but also requires attractive learning materials and methods. This section will introduce several learning theories which have been used in iPlayCode as a mobile serious game.

A. Student's Learning Style

Each individual's learning style will affect his / her learning process and their learning outcomes. When students learn to use the method that matches their styles of accessing and processing information, their learning will be more efficient [10]. The Felder-Silverman Learning Style Model (FSLSM) describes four dimensions to distinguish students: Active/Reflective, Sensory/Intuitive, Visual/Verbal, and Sequential/Global [11]:

- Active/Reflective: iPlayCode facilitates both reflective and active learning. While reflective students can learn the correct syntax through their experiments in the game, active ones can also look for the underlying reasons using the help button which provides tips and explanations for the correct answers.
- Sensory/Intuitive: While intuitive learners are capable of understanding programming languages without having many difficulties, sensory ones may find it hard to understand rules and abstract concepts. iPlayCode can help them solve this problem by providing direct examples of each concept and rule.
- Visual/Verbal: Visual learners focus on the images and diagrams, and the verbal learner is more likely to learn through textual representation. A graphical interface avoids the monotony of boring programming languages.
- Sequential/Global: Sequential learners follow a gradual linear learning progress, which is supported by the flow of levels in iPlayCode. However, users are not required to follow such sequence, which is helpful for global learners as they learn from diverse knowledge acquisition.

B. Student's Learning Cycles

Kolb's experiential learning theory describes four major steps of learning processing and knowledge learning process needed to understand and apply a theory (see Figure 1).
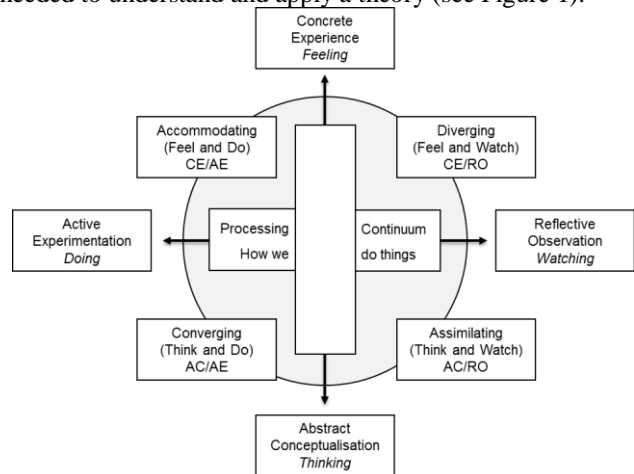

Figure 1. Four Stage Cycle of Kolb's Experiential Theory [12]

It is noteworthy that this theory supports the idea that learning is a cyclical process based on experience, and effective learning depends on positive repetition, as stated by Park et al. [12].

Through continuous repetition, learners will discover new points from different syntax. In addition, Clow analysed the notion that multiple practices will help learners to understand abstract concepts, and that learners can discover underlying theory through observation [13]. If the practice is inconsistent in its outcomes and expectations, learners will think about this problem and generate curiosity.

IV. SERIOUS GAME DESIGN

A. User Interface

Aesthetics & graphics is one of six important elements in supporting serious games systems, according to Mitgutsch and Alvarado [14]. User interface, as a core component, affects the coherence and cohesion of the entire game system. Audiences determine the direction of visual effects design.

1) Look and Feel Styles

According to the questionnaires, university students look forward to more lively and brilliant colours in the game interface. Therefore, designers use vivid colours such as blue, pink, purple and green to attract students on login screen (see Figure 2).

Figure 2. User Interface Design of Home screen

Bright colours are often used to attract attention but they may not be beneficial if the purpose of the designer is to draw students' focus on screen for long period. Instead, such complementary colours as red-green, and white-black will be effective in highlighting learning contents.

2) Structure

The storage capacity of a mobile device is limited, so developers should exploit all possible ways to reduce the file size of the program and installation package. Tabs usually require complex programming and several background images. Therefore, we found that superimposed pictures will facilitate interface design and reduce the file size (see Figure 3).

Figure 3. Tab switches design

Developers only need to change the underlying background to achieve interface effects. At the same time, the hot zone is also designed to solve the limitations of the text menus' focus. The menu event will be triggered as soon as the user touches the hot zone.

3) Usability

Serious games should minimize the user's operation and provide specific necessary tips. The main game interface design is coherent with the style of Select screen (see Figure 4). A timer, which directly connects with time incentives, is placed on the left side of the screen. The more time remains, the higher bonus will be added to the total score.

Pressed Wrong Button

Figure 4. Main Screen

In addition, the entire screen only uses three buttons and developers use multiple gestures as a trigger condition to avoid complex interfaces. In order to reduce duplication of pictures loaded, different buttons use the same set of background images with different words on the pictures.

B. Game Mechanics

Game mechanics influence the game's entertainment and education. iPlayCode has two mechanisms to improve learning efficiency, including incentive mechanisms and

competition mechanisms, which will be discuss in detail in this section.

1) Incentive Mechanisms

Any game must reward excellence players and punish failed players. As a result, this serious game has a punishment mechanism. In details, players will be deducted 5 points when they choose the wrong answer. Furthermore, according to the speed of answering, each correct answer gains up 10 points as an award. However, if the student does not answer the questions within the prescribed time, bonus points will be cancelled.

The reward mode is divided into two sub-modes including medal mode and coins mode. Firstly, visual incentives will trigger visual learners' motivation, which is why medal images as well as texts are used to represent the result (see TABLE II).

TABLE II. MEDAL MODE

0%-39%	40%-69%	70%-100%

Secondly, coin mode is set up to highlight the distinction within one medal as the users with the same medal can have different scores. Moreover, it is also useful to reward users' continuous effort in the way that coins are accumulated throughout the game until it gets to a certain quantity. At this point, users are awarded a certificate to mark their milestones.

2) Competition Mechanisms

Competition also plays an important role in motivating students. In this game, students will need to compete with themselves rather than other people as they can repeat the level until they get a satisfactory result. By such repeated practice, students may find it easier to remember the knowledge and find more details within the problem.

V. IMPLEMENTATION

The main purposes of iPlayCode are to create good user experience and improve the expandability of the game as well as to enhance learning environment. To achieve these goals, we will introduce the construction of the following parts of the game.

A. Interface Implementation

1) Tab Navigation

According to previous design, tab navigation will be divided into two parts, namely, foreground and background. Text menu as foreground controls page contents using CCMenuItemFont class and it can also change the background image layer to show the selected tab. setTag() method defines the tag of label and getTag() method can obtain the tag of label to determine the selected tab.

The effective selection area of text menu is usually limited to the text size or length. setContentSize() method can set a specific area as a text hotspot. Users only need to click on a fixed area to trigger the selected menu.

```
void MenuSelect::menuCallback(CCObject* pSender){
    CCMenuItemImage* item = pSender;
    if (item->getTag() == 1) {// show level 1
        pTabBackground1->setVisible(true);
    } else if (item->getTag() == 2) { // show level 2
        pTabBackground2->setVisible(true);
    }
}
```

2) Simplified Buttons

Buttons are an indispensable part of serious games. However, a large number of button increases the complexity of the interface and the use of them can be confusing to users. Consequently, we reduce the number of buttons using CCMenuItemImage and CCLabelTTF class. In details, CCMenuItemImage class as background can be triggered as an image menu item and its upper layer is a text layer, CCLabelTTF, which can be changed by setString() method. In addition, this serious game is a multi-language application, and changing a text in different languages will be easier than changing an image button. So, text is used in the simplified button. Implementation of this function is as follows:

```
void GamePlay::menuCallback(CCObject* pSender){
  if (menu is answering){
      button0->setString("%d",Right_string);//change text
      button1->setString("%d",Wrong_string);//change text
  } else if (menu is answered) {
      button0->setString("%d",Help_string);//change text
      button1->setString("%d",Next_string);//change text
  } ... // change actions of buttons
}
```

B. Game Elements Implementation

1) Timing

Animation, combined with sound effects, will enhance the game's entertaining effect and a timer will accelerate the speed of the game. On the other hand, the speed of users to make correct decisions is also a crucial criterion to measure the efficiency of their learning. Therefore, CCSpriteFrame and SimpleAudioEngine class will be used to practice the timer in order to improve teaching effectiveness. Schedule Class as system timer will trigger the animation and play sound effects using the following method:

```
void GamePlay:: schedule_step(){
    CCArray*    animFrames    =    CCArray::
    createWithCapacity (30);
    for(int i = 1; i <31; i++){
      CCSpriteFrame*    frame    =    cache->
    spriteFrameByName("clock-%d.png", i);
      animFrames->addObject(frame);
    }
    CCAnimation* animation = CCAnimation::
    createWithSpriteFrames(animFrames, 0.5f);
    clockSprite->runAction(animation);
```

```
        sharedEngine()->playBackgroundMusic("timer.caf");
}
```

2) Self-competition

According to Kolb's experiential learning theory, learning requires repeated practice and thinking. To motivate students to learn and master the basic knowledge, self-competition mechanism plays an important role. The results of users are stored in the database and are shown in the results interface. Results for each subject are also presented in the menu page.

Moreover, all students' results will be stored in myResult[], which is an array of integers, and uploaded to the server so that teacher can check students' learning progress and outcome.

VI. TECHNICAL ASSESSMENT

Alpha testing has been carried out, in this project and this section will provide detailed test results, including functional testing, performance testing and usability testing.

A. Functional Testing

TABLE III describes the performance of different functions in a network environment and a non-network environment. Information reading and writing are running properly in the corresponding module.

TABLE III.   FUNCTIONAL TESTING RESULTS

Functions	Stand-alone	Online
Obtain Information	-	√
Upload Information	-	√
Save Local File	√	-
Virtual Keyboard	√	√
Level Switching	√	√
Screen Transition	√	√
Answer Judgment	√	√
Incentive Mechanism	√	√
System Prompts	√	√
Display Results	√	√
User Logout	-	√

Moreover, main functions of the system obtained positive feedback in both modules.

B. Performance Testing

System performance determines the user experience as well as the fluency of the game. Performance test chose the equipment with different operating systems. Test environments are iOS 7.0.2 (11A501), iOS7.0.3 (11B511) and iOS 7.0.4 (11B554a). The test results are presistent and positive.
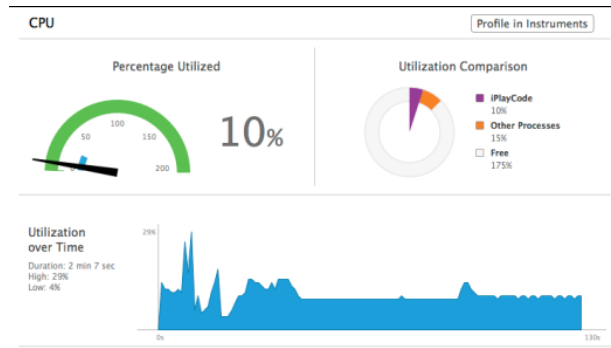

Figure 5. Performance testing of CPU

Figure 5 summarizes the performance of iPlayCode on these systems.

A. Usability Testing

This test includes five components: learnability, efficiency, memorability, errors and satisfaction. 36 students were invited to participate in the usability testing. The test results are as follow:

Learnability: Every user can manage this application when they first encounter this design.

Efficiency: Experienced users can quickly complete a concept within three minutes.

Memorability: When users are asked to play it again, most users can pay direct attention to the questions without thinking about operations steps.

Errors: Users' errors can be corrected by the system prompts. In addition, the back button can easily recover the system from a user's misuse.

Satisfaction: 61% of users prefer to use this game, because the interface is easy to use and manage. Furthermore, they enjoy iPlayCode with incentives and competition mechanisms.

VII. CONCLUSION AND FUTURE WORK

In this paper, based on the learning theory and model, the approach of using the mobile serious game for programming learning has been proposed and evaluated. Learning from this paper are the following:

1. Serious games should be developed as a learning cycle mode to improve the efficiency of learning and differentiated learning should also be designed in the early stage of software development.
2. Serious games should try to use vibrant colours (blue, pink, purple, green) to attract attention and complementary colours to get effective learning. System testing found that simple operation and appropriate prompt improve the user's experience.
3. A property list file implements multi-resolution adaptation and multilingual environment. Automatic data acquisition enhances the data analysis capabilities.

In the future, the development team will expand to include (1) more formative feedback to students, such as grade based on the score, ranking in game centre and the

prompt of learning tasks, and (2) statistical analysis functions for teacher to gain more accurate information about students' engagement, progress, and performance.

REFERENCES

[1] T. Mitamura, Y. Suzuki, and T. Oohori, "Serious games for learning programming languages," Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on. IEEE, 2012, pp. 1812-1817.

[2] C. Xin, "Multiplayer Game in Mobile Phone Serious Game," Artificial Intelligence, 2009. JCAI'09. International Joint Conference on. IEEE, 2009, pp. 56-58.

[3] B. Kapralos, "A course on the design and development of serious games and virtual simulations," Games Innovation Conference (IGIC), 2012 IEEE International. IEEE, 2012, pp. 1-4.

[4] M. Prensky, "Digital game-based learning," Paragon House Publishers, 2007.

[5] M. A. Khenissi, F. Essalmi, and M. Jemni, "Presentation of a Learning Game for Programming Languages Education," Advanced Learning Technologies (ICALT), 2013 IEEE 13th International Conference on. IEEE, 2013, pp. 324-326.

[6] J. Lu, Z. Meng, and J. B. Stav, "A new approach in improving operational efficiency of wireless response system," Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on. IEEE, 2010, pp. 2676-2683.

[7] Apple. Eduction: ipad apps books and more. [Online]. Available from: http://www.apple.com/education/ipad/apps-books-and-more/   [accessed: 2014-01-29]

[8] C. Franzwa, Y. Tang, and A. Johnson, "Serious Game Design: Motivating Students through a Balance of Fun and Learning," Games and Virtual Worlds for Serious Applications (VS-GAMES), 2013 5th International Conference on. IEEE, 2013, pp. 1-7.

[9] M. Zapusek, S. Cerar, and J. Rugelj, "Serious computer games as instructional technology," MIPRO, 2011 Proceedings of the 34th International Convention. IEEE, 2011, pp. 1056-1058.

[10] S. A. Gamalel-Din, "An intelligent etutor-student adaptive interaction framework," Proceedings of the 13th International Conference on Interacción Persona-Ordenador. Elche, Spain, New York, NY, USA: ACM, 2012, pp. 12:1-12:8.

[11] A. K. Hazra, P. Patnaik, and D. Suar, "Influence of learning styles and academic achievement on performance and preference in different learning modes: An adaptive hypermedia approach," Intelligent Human Computer Interaction (IHCI), 2012 4th International Conference on. IEEE, 2012, pp. 1-7.

[12] J. H. Park, B. Abirached, and Y. Zhang, "A framework for designing assistive technologies for teaching children with ASDs emotions," CHI '12 Extended Abstracts on Human Factors in Computing Systems. Austin, Texas, USA, New York, NY, USA: ACM, 2012, pp. 2423-2428.

[13] D. Clow, "The learning analytics cycle: closing the loop effectively," Proceedings of the 2nd International Conference on Learning Analytics and Knowledge. Vancouver, British Columbia, Canada, New York, NY, USA: ACM, 2012, pp. 134-138.

[14] K. Mitgutsch and N. Alvarado, "Purposeful by design?: a serious game design assessment framework," Proceedings of the International Conference on the Foundations of Digital Games. ACM, 2012, pp. 121-128.

[15] F. Manga and J. Lu, "An Investigation in the Impact of Mobile Learning on today's Educational Environment," Proceedings of the 2013 International Conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government. EEE: WorldComp, 2013, ISBN: 1-60132-235-6.