Gamification of Education for Young Children

27.09.2018 v1.0

Vaishali Girdhar

UID: 3035207425
+852 6749 5736

Supervisor: Dr. T. W. Chim

Table of Contents

Introduction

The rapid increase in population has caused a sharp incline in academic competition leading to students feeling pressured and worn-out. This can be harmful if experienced at an early age. One way to tackle this problem and making learning more enjoyable is gamification. Gamification of education is the concept of using game design elements in educational contexts usually aiming to make learning more desirable for the students[1].

The idea of gamification for educational purposes has been around for many years now. Earlier, parents used to play games such as peek-a-boo or identifying shapes with their children to help them understand simple concepts at an early age[2]. Then came e-learning platforms such as Lifesaver. Lifesaver is a browser and mobile app based e-learning platform that aims at making the user aware about the basic steps of aid when someone goes into cardiac arrest or is choking. The platform makes strong use of character and story to put the user in different contexts as well as other gamification techniques such as competitiveness by making the game time based and only unlocking advanced levels once lower levels have been passed[3].

However, previous implementations and technologies were limited to web-based and app-based and may not provide the user an immersive experience. Now, with the introduction of virtual reality and augmented reality through devices such as Oculus Rift, Gear VR, LEAP Motion etc. the potential of using gamification in an educational context expands even further. This project aims to use virtual reality to make a game for educational purpose for use by young children to help provide the students a fully immersive experience while learning, hence making learning more enjoyable.

Moreover, traditional education and various e-learning platforms are very much targeted to teach textbook related skills or other utility skills and not soft skills such as hand-eye coordination or development of peripheral vision and reaction time. This project targets the development of soft skills that are usually not taught at schools through a fun music based game that will help develop aforementioned skills in the player.

Furthermore, the use of music and music concepts will help develop academic related skills in the user as well. Research suggests that introduction to music at an early age helps children develop phonetic and reading skills[4], mathematical skills[5], and intrapersonal and social skills, making the game more useful for students[6].

The rest of this paper gives a detailed specification of the objectives of the game, followed by an explanation of the implementation of the game where game design, game story, game features, and game development platform and technologies will be discussed. The paper ends with a proposed project plan including the project schedule and deliverables.

Aim

The goal of this project is to make a educational game for toddlers and young children to help them pick up skills such as hand-eye-coordination, usage of peripheral vision and develop their audio-visual skills.

Since the game is a single-player game, the main aim of this project will be to use as many gamification elements as possible in order to ensure that the player does take some learning out of the game. At the same time, the game is not intended to be directly education based like other quiz games as the main aim of the game is to help the player develop certain skills they may not be able to develop in the classroom.

Objectives

TwinkleTAP proposes to meet the specified aim through achieving the following objectives:

1. Ensuring that the player will need to use more than one of their sensory organs e.g. eyes and ears at the same time
2. Ensuring that the player will need to use their hands to play the game
3. Making the game fun and easy to follow
4. Providing the player instant feedback about their performance after each round of the game
5. Keeping the game competitive by using a timer with a penalty for a wrong answer and a bonus for a correct answer
6. Keeping the user motivated to play the game by allowing the user to only unlock higher levels after giving a good performance
7. Using VR with LEAP to make the game more interactive and immersive
8. Using catchy nursery rhymes as songs so that the students can learn the rhymes while playing the game

Methodology

The development of the project is divided in two main sections- preparation and implementation. Preparation includes preparing materials such as animations, graphic elements, sound effects, in-game music, and setting up the VR set with LEAP motion test the accuracy of the LEAP motion action detection. The implementation stage includes implementing the game design elements, putting the game together using a game development engine, developing test cases and playtesting the game prototype.

1. Preparation

Game Play

The game is a music and rhythm genre game aimed for a VR platform and the player will need to wear the VR headset connected to a Leap motion device to play the game. Once the player is ready, they can select what level they want to play at and their desired song. After which the game will display a vertically rendered mini-piano where the notes corresponding to the music will glow red in color on the piano keys. The player must move their hands and fingers to make a “pointing” motion to “play” each key. As the song progresses the “red” keys will change according to the notes of the song. Once each song is finished the user will be given some feedback and a rating based on their performance.

Throughout the game the player will be competing with a timer; if the player takes too wrong to press a key or presses the wrong key, the timer will go faster. Similarly, if the player presses the eight key, some additional time will be added to the timer. The objective of the game is to finish each song without letting the timer end. As the player progresses, advanced levels with longer and faster songs will be used along with a larger piano with more keys to choose from.

In-Game Graphics and Animations

Most of the basic animations such as widgets, background UI, animations to display score etc. will be made by me using Photoshop. Photoshop will be used since it is a platform which allows both 2D and 3D animations, and I have prior experience with using Photoshop for graphics development. Since the game is planned to be in 2D, all the animations and graphics will be in 2D.

The rest of the graphics will be taken from open-sources platforms via Creative Commons which is a resource that allows users to search for and use open-source graphics without any legal liabilities.

There are various file formats available to make 2D graphics, the table below (Table 1.1) shows a short analysis of some of the common file formats[7]:

Table 1.1 **Graphic File Format Comparison**

Feature	.gif	.jpeg	.png	.psd	RAW	.svg
Lossless	✓	✗	✓	✓	✓	✓
Animation Support	✓	✗	✗	✓	✗	✓
Transparency Support	✓	✗	✓	✓	✗	✓
Layers Support	✗	✗	✗	✓	✗	✓

After analyzing various file formats for images and graphics, it has been decided that .svg format will be used since it is lossless, which means that the image will not be compressed thus giving highest quality images. Moreover, .svg supports transparency by making use of the alpha values along with RGB values, this will make it better for use for odd-shaped graphics. Lastly, .svg format is scalable, unlike .psd format, which means that the image can be "grown" to a large extent without loss of quality. This will be beneficial for the game since the game is being developed in VR and the images used will need to be quite big and good quality.

Music and Sound-Effects

The game is very heavily dependent on short music clips and sound effects, these sound effects will also be collected from open-source resources or free-to-use sources. The file format for the music files used for the gameplay will need to be in .midi since .midi files carry note and timing data along with the music data. This note and timing data will be used to guide the player about which key they must tap next. As for the in-game sound effects .mp3 format will be used since it provides good compression with an acceptable sound quality[8]. The in-game sounds are short and only used for added effect, so it is reasonable to compromise on their sound quality in order to get better compression and smaller music files to ensure that the game works well in real-time and takes up as little memory as possible to prevent lag due to data transfer between files.

Hardware Setup

Once all the graphics have been prepared and the music collected, it is important to set up Oculus and Leap to test the accuracy and compatibility of the devices. This will be done by using a Unity demo tutorial for Oculus with Leap to test whether the two

technologies are sufficient to implement the game. Since there is already a lot of help available for Oculus with Leap, it should not be a problem to do this step. After the demo tutorial has been done successfully, a simple prototype of the game UI will be tested on the devices to check if the graphics are appropriate, if not, the graphics will be changed accordingly.

2. Implementation

Game Engine

There are many game developments available in the market that also support VR development. It is important to choose the right game engine for the project to ensure smooth and effective development. Table 1.2 shows a quick analysis of three mainstream VR game engines[9].

Table 1.2 **Game Engines Comparison**

Feature	Unity 5	Unreal Engine 4	CryEngine
Scripting Language	C#, JavaScript	C++	C++, C#, Lua
VR Support	Yes	Yes	Yes
LEAP Support	Yes	Yes	Yes
2D/3D	2D/3D	2D/3D	2D/3D
Price	Personal: Free Plus: US$35/Month Pro: US$125/Month	Free until certain revenue is made	US$9.90/Month

Unity will be used as the engine of choice for this project since although all the game engines specified meet the requirements for the project, I am most familiar with the Unity framework. Moreover, Unity provides above and beyond support for both Oculus Rift and Leap Motion, this will be beneficial while solving problems during development. Furthermore, there's already a lot of open source Oculus with Leap development done using Unity, which will be beneficial as a references while developing the game.

Along with Unity's built-in APIs other APIs such as LEAP API and TouchZone API[10] (which is an API to detect "touch" using Leap Motion) and Oculus Utilities for Unity[11] will be used to support development and implement some features unique to Leap Motion and Oculus.

Game Prototype Playtesting

Playtesting for the prototype developed will be done in two stages. In the first stage, some of my developer and gamer friends will be asked to playtest the game in order to identify any bugs and get some general feedback about the game quality and how it can be improved. After the first around, I will review the feedback and make any required changes accordingly. In the second stage, the improvised game will be playtested by some young children between the age of 3-6 and their feedback on the game will be taken in order to ensure that they are indeed able to play the game and enjoy it as well as learning from it.

However, since the game is targeted to develop soft skills, it is quite hard to test whether the players have indeed picked up these skills. To check if the game is actually helping the player develop some skills, the player's progress and performance based on their accuracy and speed between similar levels of the game will be recorded and analysed. Also, the players will be asked to repeat some levels after a short break and their new performance compared to their old performance.

Final Deliverable

The final deliverable of the project will be a complete and functional game and a project report. The game will be uploaded on the project website available for download as well as on Oculus game store websites such as TheRiftArcade[12] where players can browse and download the game. The game will be defined as “functioning” if all the features specified in the Game Play section are completed to a satisfactory level and the game runs without any major bugs or defects.

Another criteria to define the project as successful will be to consistently observer better performance between similar levels of the game, this would mean that the player has indeed learnt some skills by the game as the skills required for levels of similar difficulty are identical.

Risk, Challenges and Mitigation Strategies

It is quite common to encounter risks in any project that involves usage of new and unfamiliar technologies. Table 1.3 below identifies some of the risks with this project and suggests some mitigation strategies.

Table 1.3 **Risks and Mitigation Strategies**

Risk Identified	Mitigation Strategy
Leap accuracy - Since the “tap” on a “piano key” in the game will be detected by correlating the x-coordinates of the tap and the coordinates of the key, there is a high chance that the Leap device cannot detect the “tap” accurately when the user’s hands move very fast	A proposed solution would be to either use two Leap devices or to use a Kinect[13] device to detect hand motions. Using two Leap devices will increase the available area to detect movements thus increasing the accuracy of tap detection Using Kinect may help with the situation since Kinect detects the whole body and not just the hands thus increasing accuracy. However if Kinect is used, the motion of the hands may need to be quite large to differentiate one motion from another
Players may not enjoy the game as much as the game speed might be too fast or too slow	To minimize this issue, player feedback will be taken after stage one of play testing and the game speed will be adjusted accordingly
Not enough open-source .midi files are available to make the game	If this occurs, then I will record or generate the music myself using a piano and save the files as midi files. This should not be a major concern as the proposed songs are quite easy to play and I have experience with playing the piano

Proposed Project Schedule

Stage 1 of the project i.e. the preparation stage is planned to be completed by the end of December 2017 (just before the interim presentation and report submission). Stage 2 of the project i.e. the implementation stage is proposed to be completed by the end of March 2018, followed by playtesting in the first half of April 2018. Table 1.4 below shows the detailed project schedule.

Table 1.4 **Proposed Project Schedule**

Date	Work Completed
1 October 2017	First Deliverables - Detailed project plan - Project website
15 October 2017	Basic UI elements such as background and widgets

	Main game piano UI
31 October 2017	In game music and sound effects
15 November 2017	Set up Oculus with Leap and complete unity demo tutorial Integrate game UI elements in Unity and test on Oculus
30 November 2017	Test Leap accuracy while “playing” piano in main game screen
15 December 2017	Generate game landing screen Generate game level selection screen Start interim report
8-12 January 2018	1st Presentation on current progress
21 January 2018	Interim report submission
15 February 2018	Generate game main game screen
1 March 2018	Add game logic such as point calculations, asset initialization, in-game timer, game end logic Read .midi files to get note data
15 March 2018	Translate note data to indicate a “piano key” for the user to “tap” Start final report
31 March 2018	Add widgets to game, e.g. feedback on player performance in form of stars Add sound effects Stage one of playtesting Fix any issues identified in stage one of playtesting
15 April 2018	Stage two of playtesting Final report submission
16-20 April 2018	Final presentation
2 May 2018	Project exhibition

References

[1] “Gamification of Education,” TED-Ed. [Online]. Available: https://ed.ted.com/on/uk36wtol. [Accessed: 30-Sep-2017].

[2] T. C. A. Németh, “Gamification in Education,” Ludus.hu. [Online]. Available: http://ludus.hu/en/gamification/education/. [Accessed: 30-Sep-2017].

[3] “Lifesaver Mobile - Android Apps on Google Play,” Google. [Online]. Available: https://play.google.com/store/apps/details?id=air.com.unit9.lifesaver.android.mobile&hl=en. [Accessed: 30-Sep-2017].

[4] S. H. Anvari, L. J. Trainor, J. Woodside, and B. A. Levy, “Relations among musical skills, phonological processing, and early reading ability in preschool children,” Journal of Experimental Child Psychology, vol. 83, no. 2, pp. 111–130, 2002.

[5] K. Geist and E. A. Geist, “The Patterns of Music: Young Children Learning Mathematics through Beat, Rhythm, and Melody,” Young Children, vol. 67, no. 1, pp. 74–79, Jan. 2012.

[6] S. Hallam, “The Impact of Actively Making Music on The Intellectual, Social and Personal Development of Children and Young People: A Summary,” Voices: A World Forum for Music Therapy, vol. 16, no. 2, 2016.

[7] “Common image file formats,” Comparison tables - SocialCompare. [Online]. Available: http://socialcompare.com/en/comparison/image-file-formats. [Accessed: 30-Sep-2017].

[8] M. Patkar, C. Cawley, and J. Lee, “10 Common Audio Formats Compared: Which One Should You Use?,” MakeUseOf, 16-May-2016. [Online]. Available: http://www.makeuseof.com/tag/audio-file-format-right-needs/. [Accessed: 27-Sep-2017].

[9] “9 best game engines for Virtual Reality development as of 2017,” Slant. [Online]. Available: https://www.slant.co/topics/2202/~game-engines-for-virtual-reality-development. [Accessed: 29-Sep-2017].

[10] “Recognizing Taps with the Leap Motion API,” Leap Motion Blog, 22-May-2014. [Online]. Available: http://blog.leapmotion.com/recognizing-taps-with-the-leap-motion-api/. [Accessed: 30-Sep-2017].

[11] “Oculus Utilities for Unity,” Oculus Utilities for Unity | Developer Center | Oculus. [Online]. Available: https://developer.oculus.com/downloads/package/oculus-utilities-for-unity-5/. [Accessed: 30-Sep-2017].

[12] “About The Rift Arcade,” The Rift Arcade. [Online]. Available: http://www.theriftarcade.com/about/. [Accessed: 30-Sep-2017].

[13] “Kinect,” Wikipedia, 26-Sep-2017. [Online]. Available: https://en.wikipedia.org/wiki/Kinect. [Accessed: 30-Sep-2017].