Applied Information Technology And Computer Science Vol. 5 No. 1 (2024) 355-372

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

Development of Money Mobile Game-Based Learning Application

Yim Pei Chen[1], **Ezak Fadzrin Ahmad Shaubari**[1]*

[1] Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2024.05.01.020
Received 31 July 2023; Accepted 25 May 2024; Available online 30 August 2024

Abstract: Money is a medium of transaction in the form of banknotes and coins. The traditional way to teach about money in mathematics class at a primary school is through textbooks or other written material, which is less interactive. Therefore, a mobile application, namely Simple Money is developed on the Android platform to help kids learn money. The target users for this project are Year 1 primary school students. This application was implemented by using 2D game-based learning approach and developed by using Multimedia Mobile Content Development (MMCD) methodology. This application is developed to assist target user to recognize the pattern and value of money, convert money using different combinations of coins or banknotes, and add and subtract the value of money. The result from user testing shows a higher acceptance level with an average percentage of 77.42%. More games and levels and reward systems can be improved in the future.

Keywords: Game-Based Learning, Mobile Application, Money, Android

1. Introduction

The main function of money is as a medium of exchange, which is usually provided or accepted for the buying or selling of goods, services, or other items [1]. It is common for people to use money to purchase the goods and services that they need in daily life. In Malaysia, the primary school curriculum includes a lesson on money in mathematics class. This strategy is the effort of the Malaysian government to help kids understand money literacy. The objectives of the government are to provide kids with fundamental financial education, develop healthy financial habits, and establish the foundation for financially responsible Malaysians [2]. However, the traditional teaching method of using a textbook or printed material is less interactive and interesting. Students lose their attention to the learning courses because traditional classroom learning depends on boring learning techniques and a lack of engagement [3]. Therefore, an alternative learning method is needed to help kids learn money. Applying a game-based learning approach can be a helpful tool to assist the kids in learning money.

Recently, game-based learning has gained popularity and become a global phenomenon. According to the results, it has become an essential trend to include game design and development in mobile learning [4]. This is because there is a lot of potential to improve education and students' learning outcomes by integrating mobile technology and providing a suitable learning approach [4]. In addition,

*Corresponding author: ezak@uthm.edu.my

students' cognitive and emotional abilities are improved when educational games are integrated into mathematics learning [5].

Furthermore, there is a lack of applications that are focusing on introducing and learning the currency of the Malaysian Ringgit in the market. Most of the existing applications are focused on well-known and foreign currencies such as the US dollar (USD) and European (EUR). There isn't enough interest in the development of a game-based learning application for the Malaysian Ringgit. Therefore, a mobile game-based learning application, namely "Simple Money" is developed.

The objectives of this project are to design a "Simple Money" mobile application using 2D game-based learning approach, to develop a money learning application on an Android-based device, and to perform alpha and beta testing for the developed application. This application is developed for Year 1 primary school students in Malaysia to learn the topic of money. The subject matter expert (SME) of this project is Ms. Tan Chai Xin, who is a primary school mathematics teacher. This application can be an alternative learning approach for the target user to learn about money.

The developed application, Simple Money consists of three modules which are matching, equivalent, and counting. Each of the modules consists of three different levels of difficulties and there is a time limit to complete the tasks for each module. For the matching module, the cards face down in random order and users must turn over two cards at a time and match the same coins or banknotes. For the equivalent module, the coins or banknotes were displayed and users must drag coins or banknotes worth the same amount of money to the coins or banknotes displayed. For the counting module, this application will display the coins and banknotes as the answer options and users have to click the coins or banknotes to purchase the items. Lastly, the navigation buttons functioned properly without any errors.

2. Related Work

In this section, the domain background of this project which is money topic in Year 1 primary schools in Malaysia is discussed. Moreover, the approach and technology used in this project and the comparative analysis are explained.

2.1 Money topic in Year 1 primary schools in Malaysia

It is critical to begin teaching children about money at a young age. This is because children can learn about money, which can help them develop the good habits, skills, and attitudes they will need to manage their finances in the future, and their young age is a critical time [6]. Therefore, individuals have to learn financial knowledge and skills through methods such as reading and formal education at school to ensure they can manage their finances after growing up [6].

Money is covered in Chapter 4 of the Year 1 mathematics textbook as part of the Year 1 primary school curriculum in Malaysia. The money topic in the mathematics textbook of Year 1 is about learning to recognize money, value money, convert money, get money, record money, and solve problems related to money [7]. The coins and banknotes covered in the syllabus of the Year 1 mathematics textbook include 5 cents, 10 cents, 20 cents, 50 cents, RM1, RM5, and RM10 [7]. At the end of the topic, students should be able to describe the similarities and differences between coins and banknotes. Moreover, students can recognize the value of money and convert money by using different combinations of coins and banknotes. Lastly, students should be able to add and subtract the amount of money.

2.2 Game-based learning and Unity

In this project, game-based learning is chosen as the approach to develop the application for kids to learn about money. Game-based learning is a type of gameplay or game environment that can achieve learning outcomes after the users explore their cognitive skills to solve problems and challenges [8].

Commonly, game-based learning consists of different difficulty levels that are highly interconnected. There are several types of game-based learning approaches, such as flashcard-type games, simulation games, interactives, puzzle games, quiz games, strategy games, and reality-testing games [9]. Game-based learning applications can allow the users to experience the game model of the learning experience and are able to provide immediate feedback, defined tasks, and challenges [8]. Furthermore, most of the students may not follow the instructions of the teachers or dislike problem-solving, so the use of educational games can provide the students with an environment that is entertaining and motivating for them [10].

Unity [11] is a cross-platform game engine that is supported on the desktop, mobile, console, and web, including Windows, Mac, iOS, Android, and WebGL. This game engine provides the developer with the tools to build 2D and 3D games, augmented reality, virtual reality, and games with simulations. It has a powerful visual interface for making games and the coding language used was C#, which is a simple, modern, general-purpose, object-oriented programming language. C# has several features, including a statically typed and easy-to-read language and a library with high-level functionality. This can help developers reduce development time.

2.3 Comparative Analysis

This section discusses on existing applications related to money learning. The existing applications that have been chosen for study and discussion include Kids Learning Money Lite [12], Kids Coins Count Money Lite [13], and Calculating with Money [14]. The main menu interface of the existing applications was shown in Figure 1.

Figure 1(a): Kids Learning Money Lite [12]

Figure 1(b): Kids Coins Count Money Lite [13]

Figure 1(c): Calculating with Money [14]

Table 1 shows the comparison between existing application and developed application. For the developed application, it focused on the target user of Year 1 primary school students in Malaysia and is focused on Malaysian Ringgit currency only. The strength of the developed application was it is free to play and does not have any advertisements which can influence the attention of the users. The developed application consists of exit buttons and the confirmation panel was pop out to ensure the action of users. Furthermore, the total levels were displayed to give the users can choose the difficulty of the game and clear instruction was provided to improve the user experience. However, the developed application only covers three levels for each module.

Table 1: Comparison between existing application and developed application

Element	Kids Learning Money Lite	Kids Coins Count Money Lite	Calculating with Money	Simple Money
Target User	Kids and adults	Children aged 6-10 years	Children	Year 1 primary school students in Malaysia
Platform	Android	Android	Android and iOS	Android
Language	English	English, Spanish, Catalan, German, French, Italian and Greek	English, France and Dutch	English
Currency	US Dollar, Euro, and British Pounds	US Dollar, Euro, Pounds and Mexico Peso	Euro	Malaysian Ringgit
Purchases	In-app purchases	Free to use	In-app purchases	Free to use
Exit button	Does not exist	Does not exist	Exist but does not pop out confirmation message	Exist and pop out confirmation message
Strengths	-Provide tutorial on how to play the game -Does not have advertisements -The total levels were displayed	-Free to play -Multiplayer modes -Can review the progress for each user	-The difficulty of game was rated using stars	-Free to play -Does not have advertisements -The total levels were displayed -Provide clear instruction -Provide sound effects and background music
Limitation	-Not all the level is free to play -Does not provide clear instruction -Only gameplay of drag-and-drop coins or bills to reach the target amount	-The advertisement was displayed at the top of the screen and pop out frequently -It does not have the level selection -Does not provide clear instruction	-Not all games are free to play -The advertisement was displayed at the bottom of the screen -The image and item's name are confusing -Does not provide clear instruction	-Only cover 3 levels for each module

3. Methodology

The methodology that will be used for this project is discussed in this section. The methodology of Multimedia Mobile Content Development (MMCD) [15] was used for the development of the application, Simple Money. This methodology was chosen for the project since the application is a game-based learning application. Additionally, MMCD methodology can improves in shortening the development of applications and maximizing the data and processing usage on mobile devices. Figure 2 shows the model of Multimedia Mobile Content Development that consists of five main stages which are the application idea creation stage, structure analysis stage, design process stage, development of main functions stage, and testing stage. Each of the stages will be further discussed in the sub-sections below.

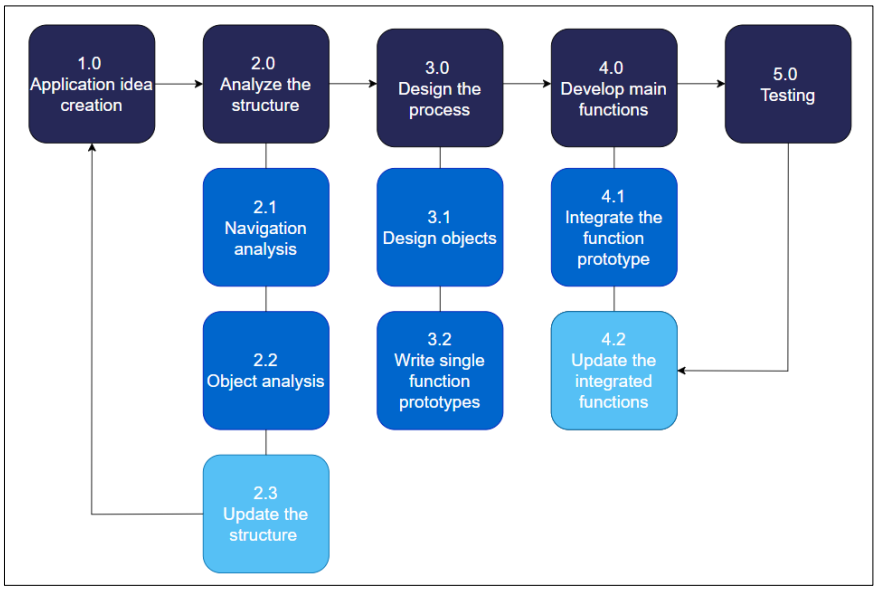

Figure 2: Multimedia Mobile Content Development (MMCD) [15]

3.1 Application Idea Creation

The first stage for Multimedia Mobile Content Development (MMCD) is the application idea creation stage. In this stage, the idea of the application was proposed and discussed with the supervisor for his suggestions and opinion. There are two requirements to be determined for the project which are user requirements and application requirements. In this stage, the application idea creation check list was shown in Table 2 while the user analysis was tabulated in Table 3.

Table 2: Application idea creation check list

Item	Description
Type of application	Game-based learning
Target user	Year 1 primary school students
Target device	Android-based device
Unity	• Version 2020.3.32f1 • Free aspect
Adobe Photoshop CC	• Design graphic assets • Design background
Adobe Illustrator CC	• Design buttons • Design 2D images
User interface	Background (main menu page, module selection page, level selection page)
Images	The coins and banknotes of Malaysian Ringgit, images of item in the counting module
Audio	• Background music • Correct answer sound effect • Wrong answer sound effect
Application synopsis	Simple Money is a game-based learning application for Year 1 primary school students in Malaysia to learn about money. It could be used as an additional medium to enhance the understanding of the students for the money topic in the mathematics subject. This application is designed to cover the syllabus of recognize money, conversion of money and operation of money.

Table 3: User analysis

Stakeholder category	Role in product	Design implications	Actions needed
Subject Matter Expert (SME)	Content consultant expert in the related field	Simple user interface design	• All the buttons shall be in consistent sizes and shapes. • The buttons shall be icon-based instead of text. • The fonts size and font family shall be readable to the target user. • Use more image instead of text.
		Easy to navigate	• The navigation structure is simple and not too complex. • Consists of setting, play, replay, back, next, home, and exit buttons.
		Easy to learn	• The game style is simple. • The instruction of the game was clear and short. • Use simple English word.
		Reliable content, and simple word choice	• The suitable content as taught in the topic money for Year 1 mathematics. • Words are simple to understand.

3.2 Analyze the Structure

Structure analysis is the second stage of the MMCD methodology. In this stage, two components were analyzed, which are navigation analysis and object analysis used in the application. The navigation analysis and object analysis were necessary because they identify the important features of a well-structured MMCD. The improper analysis of these analyses will cause a negative effect on the following design stage and affects the development process. Figure 3 shows the navigation structure while Figure 4 shows the main flowchart of the proposed application. The content structure was attached in Appendix A.

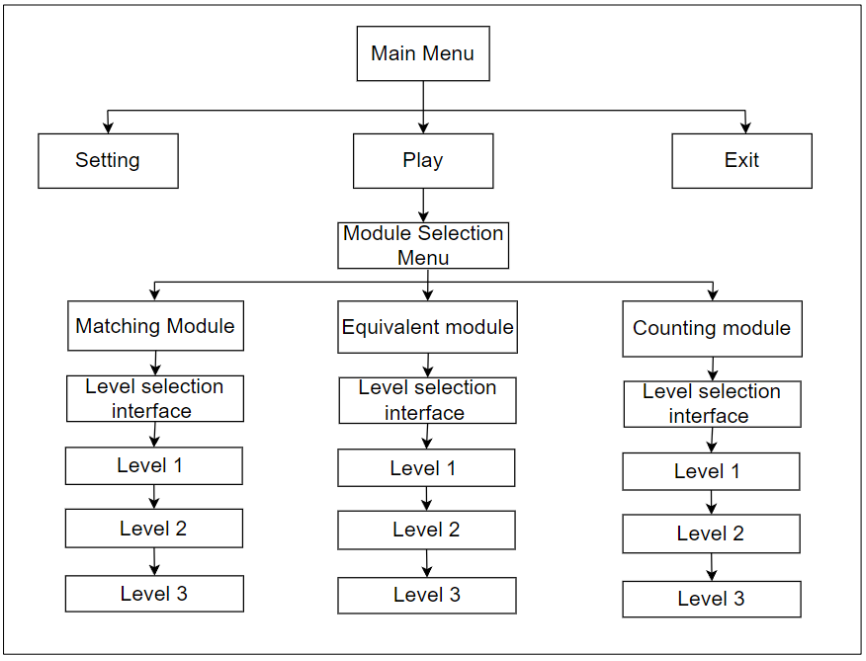

Figure 3: Navigation Structure

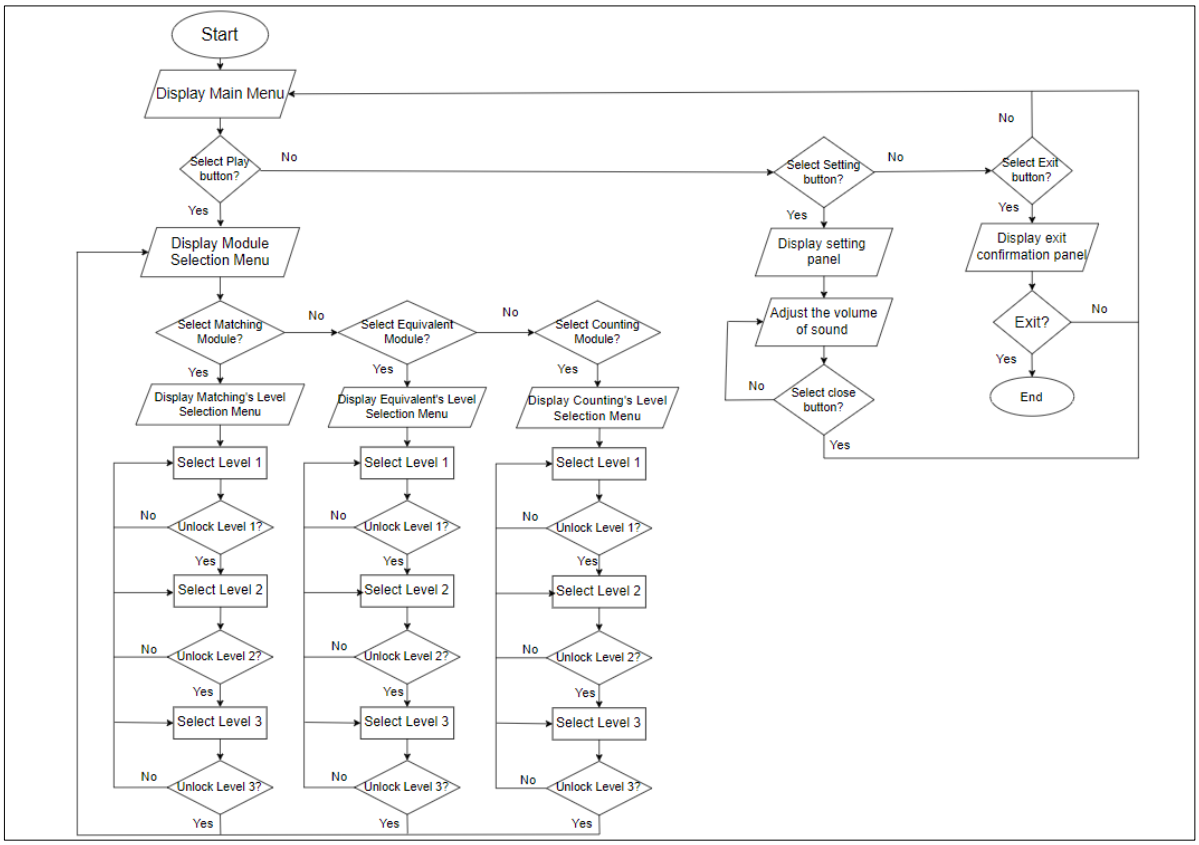

Figure 4: Main flowchart of Simple Money

3.3 Design the Process

The third step of MMCD is the process design stage. It includes two sub-components, which are designing the objects and writing the script for the single-function prototype. The design objects which included buttons, graphics, and backgrounds were designed by using Adobe Photoshop [16] and Adobe Illustrator [17]. Unity [11] was used to integrate the required assets as well as the scripting. Table 4 shows the button design and their function description. The interface design was attached in Appendix B.

Table 4: Button design

Button	Function Description	Button	Function Description
	This is a play button.		This is a check answer button.
	This is a home button.		This is a no or close button.
	This is a setting button.		This is a Matching module button.
	This is an exit button.		This is an Equivalent module button.
	This is a next button.		This is a Counting module button.
	This is a back button.		This is a level 1 button.
	This is a replay button.		This is a level 2 button.
	This is a yes button.		This is a level 3 button.

This is a clear answer button.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class ChangeScene : MonoBehaviour
{

    public void ChangeSceneByButton(string sceneName)
    {
        SceneManager.LoadScene(sceneName);
    }
}
```

Figure 5: Single function prototype of Simple Money

Based on Figure 5, a library of `UnityEngine.SceneManagement` was added to write the single-function prototype of managing scene-loading behavior. This library was used to handle the class of `SceneManager`, which it involves the behavior of `LoadScene` with a parameter named `sceneName`.

3.4 Develop Main Functions

In this stage, the main functions of the application were developed by scripting which is using the C# programming language and then integrated into Unity. For the matching module, the scripting is focusing on the functionality of two cards flipping at each time. Next, the scripting is mainly focusing on the functionality of dragging the money and drop it in the answer box for the equivalent module. The scripting for the counting module is focused on clicking the money and storing the current value for each time the money is clicked.

Table 5: Integration of scripting in Unity

Functions	Scripts	Descriptions

Flip Cards	`void AddListeners()`
	`{`
	`  foreach(Button btn in btns)`
	`  {`
	`  btn.onClick.AddListener(() => PickAPuzzle());`
	`  }`
	`}`
	`public void PickAPuzzle()`
	`{`
	`  string name = UnityEngine.EventSystems.EventSystem.current.currentSelectedGameObject.name;`
	`  if (!firstGuess)`
	`  {`
	`  popSound.Play();`
	`  firstGuess = true;`
	`  firstGuessIndex = int.Parse(UnityEngine.EventSystems.EventSystem.current.currentSelectedGameObject.name);`
	`  firstGuessPuzzle = gamePuzzles[firstGuessIndex].name;`
	`  btns[firstGuessIndex].image.sprite = gamePuzzles[firstGuessIndex];`
	`  }`
	`  else if (!secondGuess)`
	`  {`
	`  popSound.Play();`
	`  secondGuess = true;`
	`  secondGuessIndex = int.Parse(UnityEngine.EventSystems.EventSystem.current.currentSelectedGameObject.name);`
	`  secondGuessPuzzle = gamePuzzles[secondGuessIndex].name;`
	`  btns[secondGuessIndex].image.sprite = gamePuzzles[secondGuessIndex];`
	`  countGuess++;`
	`  StartCoroutine(CheckIfThePuzzleMatch());`
	`  }`
	`}`
		In `GameController.cs`, `PickAPuzzle` is a function for the user to flip the card when they click the card button. After the users click the two cards, it will set the `firstGuess` and `secondGuess` as `true` to enable the function of flipping card.

Drag and Drop	`public void OnDrag(PointerEventData eventData)`
	`{`
	`  transform.position = GetMousePos() + offset;`
	`}`
	
	`public void OnPointerDown(PointerEventData eventData)`
	`{`
	`  popSound.Play();`
	`  offset = transform.position - GetMousePos();`
	`  canvasGroup.alpha = 0.5f;`
	`  canvasGroup.blocksRaycasts = false;`
	`}`
	
	`public void OnPointerUp(PointerEventData eventData)`
	`{`
	`  RaycastResult raycastResult = eventData.pointerCurrentRaycast;`
	`  if (raycastResult.gameObject?.tag == destinationTag)`
	`  {`
	`  transform.position = raycastResult.gameObject.transform.position;`
	`  }`
	`  canvasGroup.alpha = 1;`
	`  canvasGroup.blocksRaycasts = true;`
	`}`
		`OnPointerDown()` function triggers when it is in pressed state, `OnDrag()` function is called during dragging and `onPointerUp()` function is called when the touch is released. The combination of these functions allows the drag and drop functionality of the `GameObject`. When the touch is released, the `onPointerUp` will detect whether the position of

Functions	Scripts	Descriptions
		releasing touch is same with the `destinationTag` or not.
Counting Money	`public void ClickButton(int buttonValue) { currentValue += buttonValue; int ringgit = Mathf.FloorToInt(currentValue / 100); int sen = Mathf.FloorToInt(currentValue % 100); currentValueText.text = "RM "+ringgit.ToString() + "." + sen.ToString("00"); popSound.Play(); }`	`CountQuizManager.cs` consists a `ClickButton()` function which is calculate the `currentValue` for each time the money button was clicked. Different money buttons have its own value and the values were set in the Inspector of Unity.
Count Down Timer	`// Update is called once per frame public void Update() { if(currentTime > 10) { countDownText.text = currentTime.ToString("00") + "s"; } if (currentTime <= 10) { countDownText.text = currentTime.ToString("0") + "s"; countDownText.color = Color.red; } if (currentTime <= 0) { countDownText.color = Color.red; countDownText.text = "0s"; GameOver(); } if (!CompletePanel.activeSelf) { currentTime -= 1 * Time.deltaTime; } }`	The function of `Update()` is to reduce the `currentTime` by 1 second for each frame when the `CompletePanel` is not active. There is an if statement where the `currentTime` is less than or equal to 0, the `GameOver()` function was called to fail the level.

3.5 Testing

The final stage of MMCD is the testing stage and it will be discussed in this section. Alpha testing was conducted to test the functionality and effectiveness of the application while beta testing was conducted to get feedback from the target user after experiencing the application. The functionality of the buttons was tested during the alpha testing. This testing was to ensure that the buttons is well functional and the same as the expected result. The improvement or correction action was taken for any errors found in the testing. The results of alpha testing were shown in Table 6 while the results of beta testing were discussed in section 4.

Table 6: Results of alpha testing for Simple Money

Test	Expected Result	Actual Result	Correction Action
Play button	Navigate to Module Selection scene.	Works well as expected.	Not needed.
Home button	Navigate to Home Menu scene.	Works well as expected.	Not needed.
Setting button	Display the Setting panel.	Works well as expected.	Not needed.
Exit button	Display the Exit Confirmation panel.	Works well as expected.	Not needed.
Next button	Navigate to the next level of the current level.	Works well as expected.	Not needed.

Test	Expected Result	Actual Result	Correction Action
Back button	Navigate to previous scene.	Works well as expected.	Not needed.
Replay button	Navigate to the current level scene.	Works well as expected.	Not needed.
Yes button	Close the application	Works well as expected.	Not needed.
Clear answer button	Clear the input answer from the user.	Works well as expected.	Not needed.
Check answer button	Check the answer for the question.	For the equivalent module, the answer is correct if equal to the target value without fill up all the answer box.	In the if statement, check the bool variable for the answer box is true to detect whether it was filled.
No button	Close the Setting panel and Exit Confirmation panel.	Works well as expected.	Not needed.
Matching module button	Navigate to the Matching Level Selection scene.	Works well as expected.	Not needed.
Equivalent module button	Navigate to the Equivalent Level Selection scene.	Works well as expected.	Not needed.
Counting module button	Navigate to the Counting Level Selection scene.	Works well as expected.	Not needed.
Level 1 button	Navigate to the Level 1 of the selected module.	Works well as expected.	Not needed.
Level 2 button	Navigate to the Level 2 of the selected module.	Works well as expected.	Not needed.
Level 3 button	Navigate to the Level 3 of the selected module.	Works well as expected.	Not needed.

4. Results and Discussion

The beta testing was conducted by using System Usability Scale (SUS) to evaluate the user acceptance of the application. System Usability Scale (SUS) is a reliable tool for measuring the usability of an application and it is simple and popular used with the 10 survey questions [18]. A set of questionnaires was prepared on Google Form and distributed to the target users to collect the result for the user acceptance test. In this beta testing, there are 30 respondents involved to obtain their feedback regarding the application, Simple Money. After that, the data was illustrated by using pie charts and bar charts to present visually. Figure 6 and Figure 7 below illustrates the gender analysis of respondent and analysis of respondent's experience in using learning application respectively.

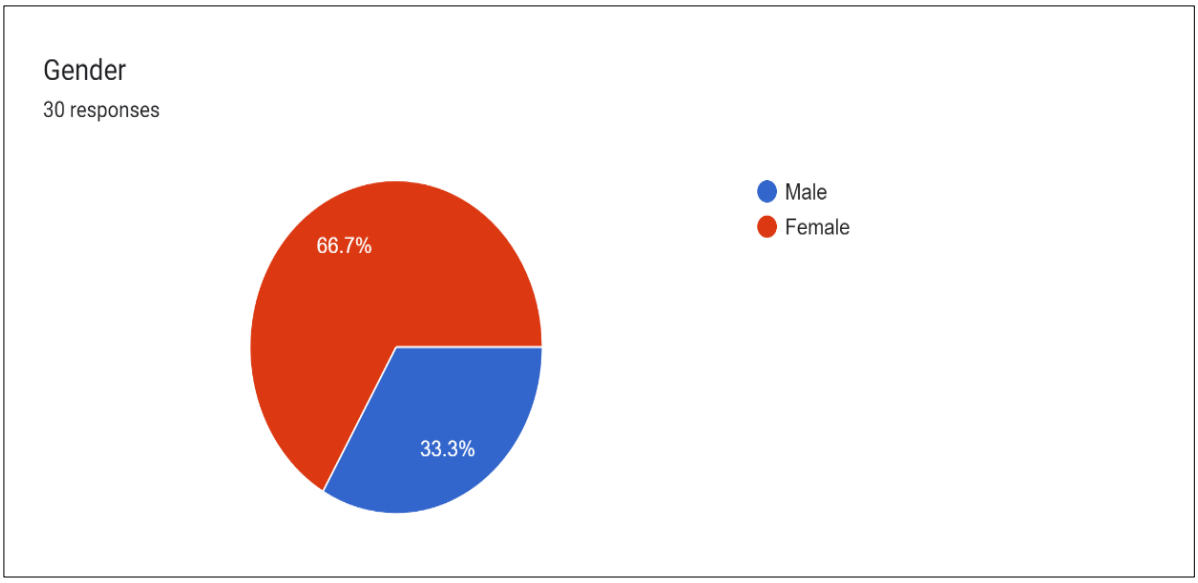

Figure 6: Gender analysis of respondent

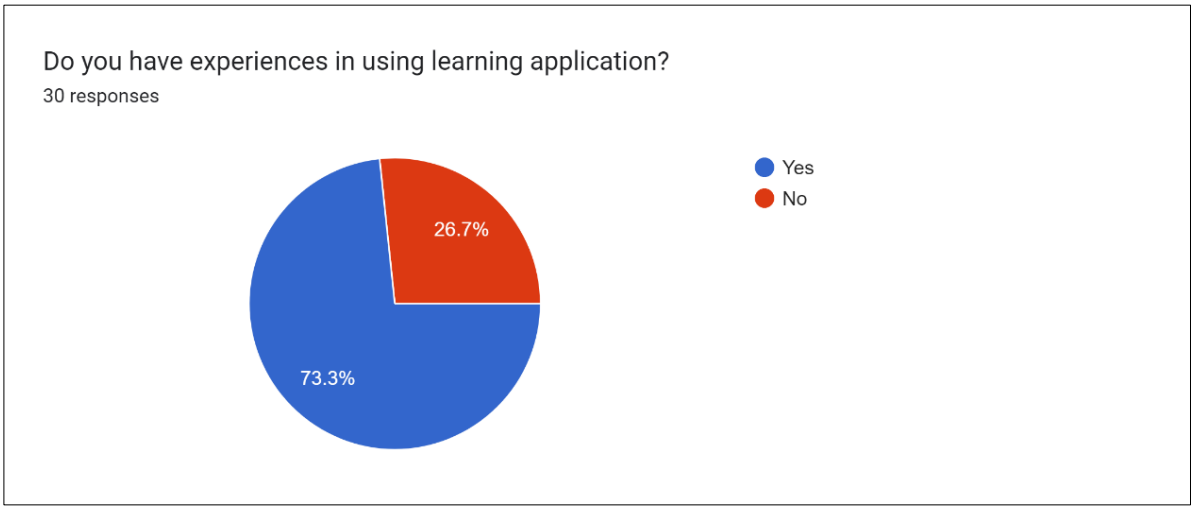

Figure 7: The analysis of respondent's experience in using learning application

Figure 6 shows that there are 20 respondents (66.7%) are female and 10 respondents (33.3%) are male in this user acceptance test. The respondents were provided with the Android Package Kit (APK) file in the Google Form to download on their mobile phone and play it. Based on Figure 7, most of the respondents (73.3%) have experience in using learning applications and the rest respondents (26.7%) have no experience in using the learning applications.

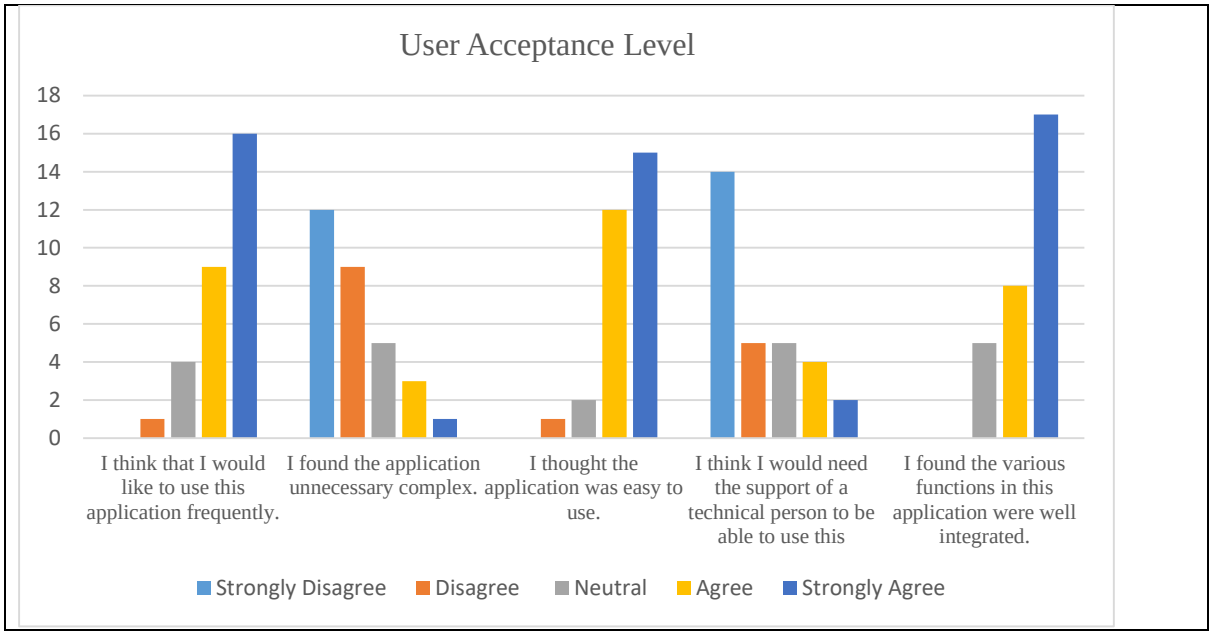

Figure 8: The user acceptance level analysis of respondent (Question 1 - 5)

This section will discuss the user acceptance level for the application. There are 10 questions were prepared for the target user to analyze the user acceptance level. Based on Figure 8, 16 respondents (53.3%) strongly agreed that they would like to use this application frequently, 9 respondents (30%) agreed with this statement, 4 respondents (13.3%) answered "Neutral", while only 1 respondent claimed disagreed with this statement. The statement of the application was unnecessary complex, there are only 3.3% of the respondent was strongly agree, 10% of respondents answered "Agree", and followed by 16.7% of respondents answered "Neutral". However, there are 30% and 40% of the respondents disagreed and strongly disagreed with this statement.

The next question is asking the user whether the application was easy to use or not. There was a total of 15 out of 30 respondents (50%) who strongly agreed that the application was easy to use, while 12 respondents (40%) agreed the application was easy to use also. On the other hand, there were only 2 respondents (6.7%) answered "Neutral" and 1 respondent (3.3%) disagreed that the application was easy to use. Moreover, 2 users (6.7%) strongly agreed that they need the support of a technical person to be able to use this application and 4 users (13.3%) agreed with this statement. However, there are 5 respondents (16.7%) who claimed moderate and disagreed respectively and 14 respondents (46.7%) strongly disagreed that they need support to use this application. For the statement of the various functions in this application were well integrated, most of the respondents (56.7%) strongly agreed and 26.7% of respondents agreed. Plus, 16.7% of respondents remained neutral about the statement.

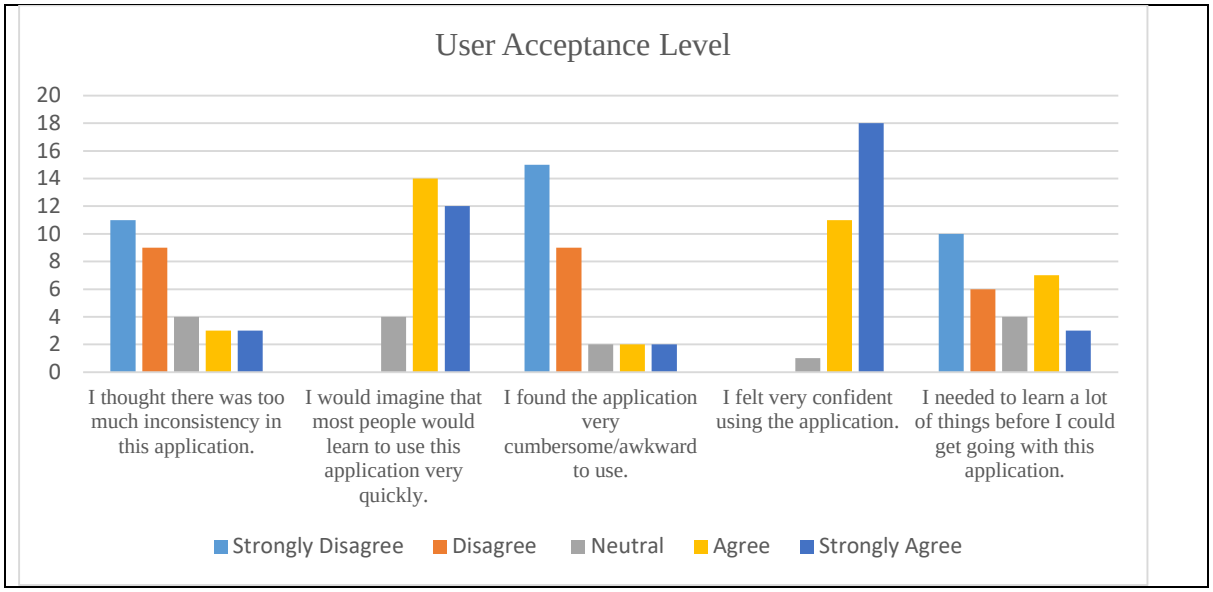

Figure 9: The user acceptance level analysis of respondent (Question 6 - 10)

Based on Figure 9, there are 3 out of 30 respondents (10%) claimed strongly agreed and agreed with the statement of there was too much inconsistent in this application, followed by 4 respondents (13.3%) choosing the neutral option. On the other hand, there are 9 respondents (30%) disagreed that the application has inconsistency, while 11 respondents (36.7%) strongly disagreed with this statement. Besides, 12 out of 30 respondents (40%) strongly agreed that they could imagine that most people would learn to use this application very quickly, 14 respondents (46.7%) agreed with the statement, while only 4 respondents (13.3%) remained neutral. Furthermore, the following question is whether the application was very awkward to use or not. Most of the respondents (50%) strongly disagreed that the application was very cumbersome to use, followed by 9 respondents (30%) who disagreed with the statement. However, there are 2 respondents (6.7%) respectively answered "Neutral", "Agree", and "Strongly Agree" for this statement.

In addition, most of the respondents (60%) strongly agreed that they felt confident using this application, while 11 respondents (36.7%) agreed with the statement. There is only 1 respondent (3.3%) choosing the neutral option. Lastly, there are 10% of respondents, 3 out of 30 respondents, strongly agreed that they need to learn a lot of things before I could get going with this application, while 7 respondents (23.3%) chose the agree option to the statement. Moreover, 4 respondents (13.3%) claimed moderate that they need to learn a lot of things before playing the game. However, there are 6 respondents (20%) and 10 respondents (33.3%) respectively claimed disagreed and strongly disagreed with the statement.

The beta testing for the user acceptance test was conducted by using the method of System Usability Scale (SUS). Therefore, the SUS score was calculated by using the SUS formula to determine whether the application was acceptable or not.

Table 7: Respondent's Score

Respondent	Item Score										Odd Score	Even Score	Total Score
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10			
R01	4	2	5	3	4	1	4	1	5	3	17	15	80
R02	5	1	5	2	4	4	5	1	4	3	18	14	80
R03	5	1	5	2	5	2	5	2	4	2	19	16	87.5
R04	4	1	3	3	5	2	4	1	5	4	16	14	75
R05	5	5	5	5	5	5	5	5	5	5	20	0	50

Respondent	Item Score										Odd Score	Even Score	Total Score
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10			
R06	5	4	5	4	5	4	5	4	5	4	20	5	62.5
R07	4	3	4	4	5	3	4	3	5	4	17	8	62.5
R08	5	1	4	1	3	3	3	2	3	5	13	13	65
R09	5	1	4	1	4	1	4	1	4	1	16	20	90
R10	3	2	5	1	5	2	5	1	5	4	18	15	82.5
R11	5	2	5	1	5	1	5	1	4	4	19	16	87.5
R12	3	2	5	2	5	1	4	2	4	4	16	14	75
R13	4	2	4	1	3	1	5	2	5	1	16	18	85
R14	5	1	2	4	5	1	5	1	5	3	17	15	80
R15	5	3	5	1	5	1	5	1	5	1	20	18	95
R16	5	1	4	3	5	2	4	1	4	1	17	17	85
R17	5	1	4	2	4	2	4	2	5	5	17	13	75
R18	5	4	5	3	4	5	4	3	5	3	18	7	62.5
R19	2	3	4	5	5	4	3	4	5	2	14	7	52.5
R20	4	3	5	4	3	5	3	5	5	2	15	6	52.5
R21	3	4	5	1	4	3	5	2	5	1	17	14	77.5
R22	4	1	5	1	5	1	4	1	5	1	18	20	95
R23	3	2	5	1	3	1	4	1	5	2	15	18	82.5
R24	5	1	5	1	5	1	5	1	5	1	20	20	100
R25	5	2	3	2	5	3	4	2	5	2	17	14	77.5
R26	4	3	4	1	5	1	4	2	4	1	16	17	82.5
R27	5	2	4	3	4	2	3	2	4	1	15	15	75
R28	4	1	4	1	3	2	4	1	4	2	14	18	80
R29	4	1	4	1	5	2	4	1	4	4	16	16	80
R30	5	2	4	1	4	2	5	1	4	1	17	18	87.5
											Average Score		**77.417**

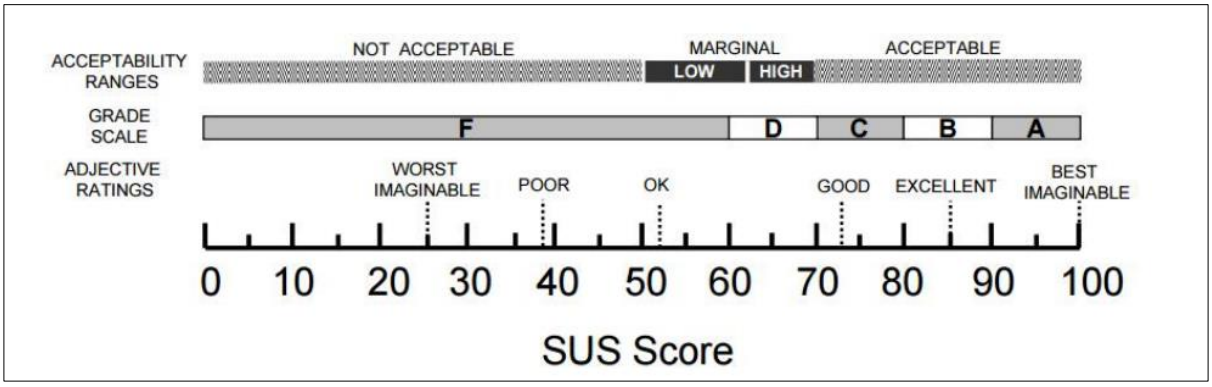

Figure 10: The Scale of the SUS Score [19]

In short, the average percentage of the respondents by using the System Usability Scale (SUS) was 77.42% which falls in acceptable range based on Figure 10. Therefore, this game-based learning application is acceptable.

5. Conclusion

In conclusion, Simple Money application is an alternative approach to help Year 1 primary school students learn money and enhance their understanding of the concept of money that was learned in formal education. Simple Money mobile application using 2D game-based learning approach was successfully developed by following a well-planned Multimedia Mobile Content Development (MMCD) methodology.

Based on the result of alpha testing, most of the buttons in the application worked well as expected with a correction action being taken to fix the bug of the check button. For the beta testing, the user acceptance test was conducted with the result of an average percentage of 77.42% which is in an acceptable range. This result shows that Simple Money is suitable for the target user and they would like to use this application frequently. Although the application was successfully developed, however, there are some limitations to the application. For instance, the game levels for each of the module was limited where there are only 3 levels for each module. Moreover, this application was available for the Android platform only. Therefore, the future work recommended was to develop more games and different levels of difficulty for the target user to learn while playing to enhance their understanding of the topic. The application also can be developed for another platform such as iOS. Besides, the application can improve with a reward system to increase the interest of the user.

In a nutshell, all the three objectives for this application have been achieved. Moreover, the application can be improved with the suggested recommendation for future work.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Appendix A

This section shows the content structure of the application.

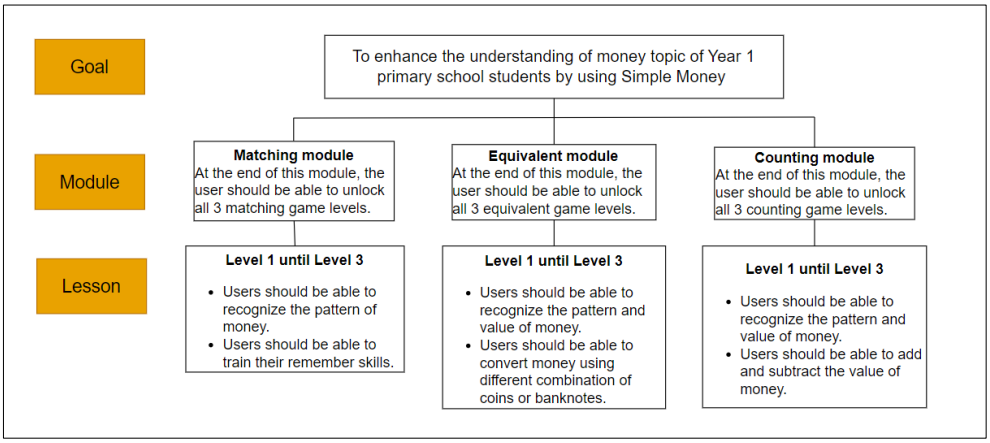

Figure 11: Content Structure of Simple Money

Appendix B

This section shows the interface design of the application.

Figure 12: Main Menu Interface

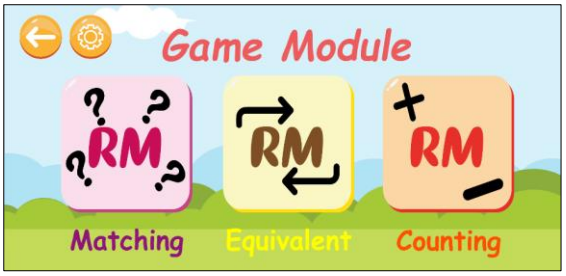

Figure 13: Game Module Selection Interface

Figure 14: Matching Levels Selection Interface

Figure 15: Counting Levels Selection Interface

Figure 16: Equivalent Levels Selection Interface

Figure 17: Matching Level 1 Interface

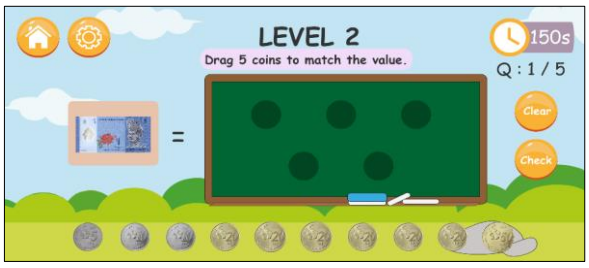

Figure 18: Equivalent Level 2 Interface

Figure 19: Counting Level 3 Interface

References

[1] J. K. Galbraith, Money: Whence It Came, Where It Went, 1st ed. Reading, Princeton University Press, 2017. [E-book] Available: Google Play.

[2] S. H. Fu, D. Tahil, W. C. Chow, L. K. Leong and L.L. Tiong, "Introducing Ringgit Malaysia Via Mobile Application: A Case Study," in 5th International Conference on Teacher Learning and Development (ICTLD) 2021, 2021, pp. 95 – 103.

[3] M. Hafeez, "Effects of Game Based Learning in Comparison of Traditional Learning to Provide Effective Learning Environment – A Comparative Review," International Journal of Social Sciences & Educational Studies, vol. 8, no. 4, pp. 100-115, 2021, doi: 10.23918/ijsses.v8i4p100.

[4] C. Y. Chang and G. J. Hwang, "Trends in digital game-based learning in the mobile era: a systematic review of journal publications from 2007 to 2016," International Journal of Mobile Learning and Organisation, vol. 13, no. 1, pp. 68-90, 2019, doi: 10.1504/IJMLO.2019.096468

[5] L. D. Pramata and W. Setyaningrum, "Game-Based Learning: The effects on student cognitive and affective aspects," Journal of Physics: Conference Series, vol. 1097, no.1, pp. 012123, 2018, doi: 10.1088/1742-6596/1097/1/012123

[6] R. Wahab, M. F. Sabri and N. N. Ramli, "Money Literacy, Shopping and Purchasing Patterns among Primary School Students in Malaysia," Journal of Education and Human Development, vol. 5, no. 1, pp. 146-162, 2016, doi: 10.15640/jehd.v5n1a16

[7] Y. L. Chan et al., "UNIT 4: MONEY," MATHEMATICS YEAR 1 TEXTBOOK (2016 Edition), 2016.

[8] M. S. Qolbi, Z. Z. A. Thaariq, S. F. Az-Zahroh, M. M. Anwar and N. Faiza, "Design and Development of Game Based Learning Applications for Mathematics Learning Based on Multiple Language to Develop Verbal Capabilities," Jurnal Pendidikan dan Pembelajaran, vol. 26, no. 2, pp. 51-56, 2019.

[9] K. Dimitra, K. Konstantinos and Z. Christina, "Types of Game-Based Learning in Education: A Brief State of the Art and the Implementation in Greece," The European Educational Reseacher, vol. 3, no.2, pp. 87-100, 2020, doi: 10.31757/euer.324

[10] M. Akour, H. Alsghaier and S. Aldiabat, "Game-based learning approach to improve self-learning motivated students," International Journal of Technology Enhanced Learning, vol. 12, no. 2, pp. 146-159, 2020, doi: 10.1504/IJTEL.2020.10027116

[11] "Unity Real-Time Development Platform | 3D, 2D VR & AR…" [Online]. Available: https://unity.com/. [Accessed Nov 25, 2022].

[12] (2014). Kids Learning Money Lite (Version 1.0.4) [Mobile App]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=com.mangustapp.learnmoney.lite&hl=en&gl=US.

[13] (2016). Kids Coins Count Money Lite (Version 2.1) [Mobile app]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=boriol.learn.moneylite&hl=en&gl=US

[14] (2016). Calculating with money (Version 2.0.2). [Mobile app]. Retrieved from Google Play Store.https://play.google.com/store/apps/details?id=com.companyname.geldrekenen&hl=en&gl=US

[15] W. S. N. Saifudin, S. Salam, and M. H. L. Abdullah, "Multimedia Mobile Content Development (MMCD) Framework and Methodology for Developing M-Learning Application," Journal of Technical Education and Training, vol.4, no.1, pp. 15-21, 2012.

[16] "Official Adobe Photoshop | Photo and Design Software." [Online]. Available: https://www.adobe.com/my_en/products/photoshop.html. [Accessed Dec 20, 2022].

[17] "Industry-leading Vector Graphic Software | Adobe Illustrator." [Online]. Available: https://www.adobe.com/my_en/products/illustrator.html. [Accessed Dec 20, 2022].

[18] S. McLellan, A. Muddimer, and S. C. Peres, "The Effect of Experience on System Usability Scale Ratings," Journal of Usability Studies, vol. 7, no. 2, pp. 56 – 67, 2012.

[19] A. Bangor, P. Kortum, and J. Miller, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale," Journal of Usability Studies, vol. 4, no. 3, pp. 114 – 123, 2009.