Applied Information Technology And Computer Science Vol. 3 No. 1 (2022) 186-203

AITCS

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

Educational 2D Learning Simulation On Civil Engineering for Civil Engineering Students

Teoh Jing Hong[1], Mohd Farhan Md. Fudzee[1]*

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

*Corresponding Author Designation

DOI: https://doi.org/10.30880/aitcs.2022.03.01.012
Received 30 July 2021; Accepted 04 September 2022; Available online 31 May 2022

Abstract: While there were many constructions simulation in the market, only a few addresses structural integrity in the simulation and structural integrity testing requires specialized equipment and thus, costly. Therefore, the project covers an Android, 2D educational, construction, budget management, puzzle learning simulation that will address the mentioned issues. The title of the simulation is "Build for All". This project will be developed based on gamification method which will be published on Android and evaluated with alpha/beta testing. GDLC was used during the development of the simulation. The requirements of end user or customer which is civil engineering students are researched and identified. Alpha and Beta Testing has been conducted with the latter having total of 18 testers in which 53.8% are target users. Based on their feedback, reviews are mostly positive with 83.4% are at least satisfied with the simulation in which coincides with the 38.9% or 7 people are interested at the gameplay. Therefore, the proposed simulation shall provide a rewarding, informative themed content that provide challenges and serve as a good reference or practice which structural engineering principles are an important aspect in the simulation.

Keywords: Android, 2D, Construction, Civil Engineering, Structural Integrity, GDLC

1. Introduction

While there are existing simulation applications that features constructing buildings, structural integrity engineering Android simulations are rare on the Android Play Store. Structural integrity can be defined as an engineering field that aims to ensures a structure is able to withstand the expected load as well as the further study of structural failures. However, structural integrity testing is costly due to requiring specialized tools, and time consuming. Non-destructive testing (NDT) as a testing and analysis method is used to identify the structural damage or assess the material or structure properties. Besides, there are less such topic on other mediums beside paperbacks or digital documents. Therefore,

*Corresponding author: farhan@uthm.edu.my

gamification of structural integrity engineering would provide an alternative, cost-saving, easy-to-use platform for testing player-built structural design.

The objective for Build for All is to develop the simulation based on gamification approach. It would be a 2D side scrolling simulation with 2D objects which will be published on Android. The simulation will be evaluated by using alpha/beta testing.

The scope of the simulation spans several aspects such as the featured language will be English, levels with the inclusion of 2D such as buttons and sprites. Two phases: "Planning" and "Playing" phase in the simulation allows the player to act as the designer by building or modifying infrastructure and test out their structural integrity. Furthermore, each level will provide different scenarios such as gravitational force to test the player's knowledge. Gravity and structural physics are also proposed as well for more realism.

In this report, the contents are categorized as follow with the first chapter being the introduction, the second is the related work with subchapters including structural engineering, gamification approach, and comparison of existing applications. Meanwhile, chapter 3 discusses about the methodology or framework of the project, detailing the phases of GDLC methodology. On chapter 4, the analysis and design are discussed, with the subchapters explaining the analysis, system design, script integration and testing. The final chapter concludes the project, and followed by acknowledgement and Appendix A, B and C, which are the storyboard, user testing and supported platform, respectively.

2. Related Work

2.1 Structural Engineering

According to Jerome & Susan [1], "Structural engineering is the discipline which is concerned with identifying the loads that a structure may experience over its expected life, determining a suitable arrangement of structural members, selecting the material and dimensions of the members, defining the assembly process, and lastly monitoring the structure as it is being assembled and possibly also over its life." Therefore, a structure is a collection of building components that are connected and organized in a way that it can hold out against the load that are applied on it.

Several highlighted topics are shown by Arthur et. al. [2] that highlights the concerns of structural engineering, which the following aspects including, gravitational forces, wind, snow, water, ice, earthquake, thermal and earth pressure must be addressed beforehand to ensure the structure will not collapse under certain stress or load. Otherwise, deformation, fracture, instability, general yielding, and excessive deflection which results in failure of components.

Therefore, the stated aspects are the challenges that will be implemented in the simulation to test the player-built structures. They exert force that acts as the load for the tested structures. If the structure is poorly built, the structures will collapse due to the excessive stress placed upon them.

2.2 Gamification Approach

Gamification refers to the implementation or design of features that are found in games on services or applications according to Sebastian et al. [3]. In recent years, gamification has gained some traction as a means of garnering more engagement and attention from users. Thus, gamification could lead to higher user retention and interactions. As such, several key game mechanics attributes are to be taken into consideration when gamifying an application stated by Cristina [4]. In Build for All, the game mechanics is feedback based and its benefits is engagement for achiever type players.

- Game Mechanics Type: Progression, Feedback, Behavioral.
- Benefits: engagement, loyalty, time spent, influence, fun, SEO, UGC, Virality.
- Personality types: explorers, achievers, socializers, and killers.

2.3 Comparison of Existing Applications

All the applications including "Build a Bridge!" [5], "Construction World – Build City" [6], "Construction City 2" [7] and the proposed simulation "Build for All" were compared and analyzed along with the proposed simulation with its features, strengths, and weaknesses as shown in Table 1. The advantages of "Build for All" over the other mentioned applications are user friendly, suitable color scheme, and having unique gameplay. However, it has only one featured language which is English.

Table 1: Software and hardware requirement for user

Features	Build a Bridge! [5]	Construction World - Build City [6]	Construction City 2 [7]	Build for All
Category	-Puzzle -Casual -2D/3D	-Puzzle -Game -Casual -2D	-Puzzle -2D	-Puzzle -Game -2D
Platform	Android 4.1 or higher	Android 5.0 or higher	Android 4.4 or higher	Android 4.1 or higher
Language	-English -Simplified Chinese -Traditional Chinese -French -German -Hindi (India) -Bahasa Indonesia -Italian -Japanese -Korean -Polish -Portuguese (Brazil) -Russian -Spanish -Turkish	-English -Portuguese -Spanish -German -Bahasa Indonesia -French, -Korean -Japan	-English -Polish -German -Spanish -Russian -Thai -Italian -Turkish -Portuguese -French	-English
Visuals	3D low polygon graphics	2D, cartoon graphics with bright color	2D, cartoon graphics	2D, cartoon graphics with bright color
Audio	Rich sound effects and relaxing music	Rich sound effects and country music	Rich sound effects and pop music	Relaxing music
User Interface	-Simple -Intuitive -Colorful -Responsive buttons	-Simple -Intuitive -Bland colored -Unresponsive	-Simple -Intuitive -Responsive buttons	-Simple -Intuitive -Colorful -Responsive buttons

Table 1: (cont)

Features	Build a Bridge! [5]	Construction World - Build City [6]	Construction City 2 [7]	Build for All
Gameplay	Two phases gameplay with the first phase involving the planning and building of bridge and the second is the vehicle test driving through the bridge.	Take control of one or more machines or vehicles to solve puzzles involving construction and disaster management objectives	Take control of one or more machines or vehicles to solve puzzles involving construction objectives	Two phases gameplay with the first phase involving the planning and building of disabled-accessible infrastructure and the second involving the navigation of disabled character from the start to the end point.
Theme	-Bridge -Construction -Transportation -Modern -Environmental	-Transportation -Construction -Modern -Environmental	-Transportation -Construction -Modern -Environmental	-Transportation -Construction -Modern -Environmental
Level Design	Hand-crafted	Hand-crafted	Hand-crafted	Hand-crafted
Strength	-Free -Intuitive UI -Easy-to-learn gameplay -Responsive buttons -Many available languages	-Free -Intuitive UI -Easy-to-learn gameplay -A lot of levels -Many available languages	-Free -Intuitive UI -Responsive buttons -Easy-to-learn gameplay -Responsive buttons -A lot of levels -Many available languages	-Free -User friendly -Intuitive UI -Easy-to-learn gameplay -Responsive buttons -Suitable color scheme - Unique Gameplay
Weaknesses	-Takes a while to load game due to high graphic, do not have Malaysian contents -the accuracy of the game response with the thumb is sometimes inaccurate -frequent advertisement	-Do not have Malaysian contents -Frequent and obstructive advertisement -Unresponsive buttons -Clunky movement of vehicles -Unsuitable fonts	-Do not have Malaysian contents -Frequent and obstructive advertisement -Clunky movement of vehicles -Unsuitable fonts	-Only one language is available

3. Methodology

According to Rido & Yani [8], Game Development Life Cycle (GDLC) methodology is used in the project as it provides an overview of a game production cycle that involves the assigning of game production tasks to their respective phase. There are a total of 6 phases which distinctively describes

the task required for the game development. In addition, the system requirements analysis, design as well as script integration and testing are also discussed.

3.1 Initiation Phase

During the initiation phase, the initial concepts are defined along with a simulation description that explains the rough idea on the project. A title proposal and its initial description is proposed.

3.2 Pre-production Phase

Pre-production begins which started the enhancement of the initial concepts as well as the prototype development. The mechanics, gameplay, genre, fun factors, and other technical elements are documented during this phase. Based on the lists, minimalistic, drag and drop, structural building puzzle system is proposed as the game design document (GDD).

The first iteration of the production cycle is started to develop a prototype and subsequent cycle will enhance the existing project.

3.3 Production phase

During the first iteration, placeholder assets and initial scripts are used to speed up the development of the prototype. On subsequent cycles, the required assets including sprites, audio files and implementation scripts are developed based on the GDD.

During this phase, incomplete modules are developed, gameplay balancing, and bug fixing are done. For instance, the drag and drop module are developed, balanced, and enhanced. Missing assets such as UI sprites are also created during this phase.

3.4 Testing phase

Functionality testing is done through alpha testing. The implemented modules are tested by playtesting based on test cases based on modules such as budget system. If bugs arise, corrective measures are done. Balancing is also done to ensure the difficulty of the simulation are not extreme.

3.5 Beta phase

After proceeding to beta testing, further refinement testing as well as difficulty balancing is done to ensure the fun factors are followed based on the testers' feedbacks. Bugs, clunky gameplay, annoying music, and other issues are addressed if the needs arise in this phase.

3.6 Release

After the end of the last production cycle, the final version of the project is built and released. An application bundle based on Android platform is built and exported on Unity to be installed on any Android devices.

3.7 System Analysis

User analysis is conducted to study the requirements of the end-user; thus, several design implications are detailed. At the same time, an architect, Lim Wei Liang, has also been interviewed as the subject matter expert for advice on the purpose of the project content. The recorded implications are tabulated and detailed which are displayed in Table 2.

Table 2: User analysis

Stakeholder Category	Role in Product	Design Implication	Action Needed
Subject Matter Expertise	Content expert in civil engineering	• Distinctive materials with different properties	• Different materials that have different load strength properties shall be provided. • Materials shall be clearly distinguishable and explained in the first level for accessibility for new players.
		• Budgeting system as limiting factor	• Budgeting module shall be provided to provide additional challenges beside environmental factor. • Different cost shall be assigned to each material to challenge the player in planning out the structure.
Customer	End user of the application	• Intuitive and simple interface • Simple and readable text • Easy to learn game mechanism • Responsive feedback • Rewarding, fun gameplay • Suitable and relevant content	• Use familiar logos and graphics that are recognizable by most people. • Use familiar, recognizable interface layout styles. • Use colors to create contrast and highlights.
			• Use different, contrasting font size for readability. • Use simple wording and phrases for readability. • Use different colors of text.
			• Provide tutorials in the first level. • Obvious, intuitive interface for easier controls.
			• Use sound effects when button is pressed and dragging and dropping of objects. • Animation, graphics, and sound effects are displayed and played during gameplay and triggering of win or lose conditions.
			• Use points system to reward players in the end of the level.
			• Use real-world props for familiarity and relevance.

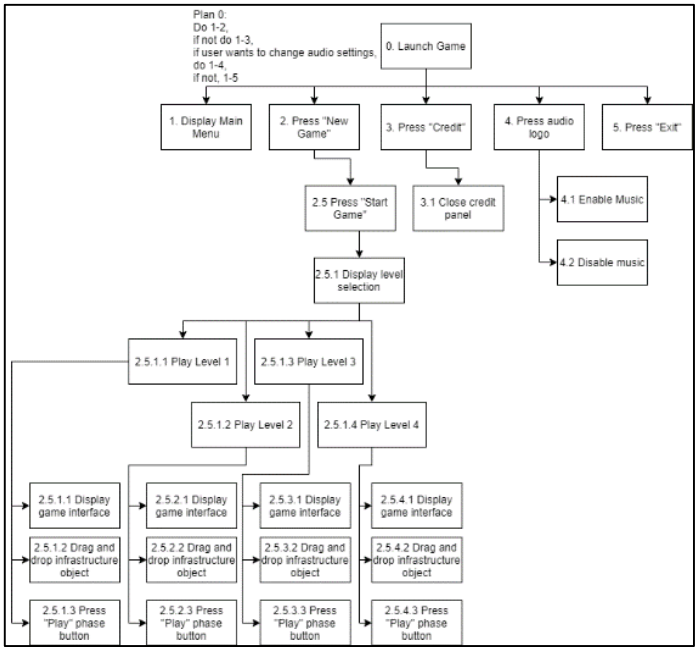

Figure 1: Hierarchical Task Analysis

Furthermore, Hierarchical task analysis (HTA) is also included in this section. It details the tasks that the user will follow while using the simulation. John [8] stated that tasks are broken down into subtasks for more levels of detail. Each task or operation must achieve certain goals under certain input conditions, listing the actions to acquire the said goals with feedback. The HTA of "Build for All" are illustrated as shown in Figure 1.

Requirement analysis is also conducted to meet with the expectations and needs of the end users as the requirements are required to be stated clearly as a guideline for the development roadmap. Hence, the functional requirements and non-functional requirements of the simulation for the user are stated in Table 3 and 4.

Table 3: Functional requirement

Functional requirement	Description
Autonomous System Activity	• If the user is a new player, the system should provide short tutorial instruction during the first level. • The system should display the requirements panel after the level is loaded. • The system should check for lose condition if the structure does not meet the requirements. • The system should check for win condition if the structure meets the requirements. • The system should provide a screen showing the losing and winning description and the main menu, level selection and restart buttons if the losing or winning conditions has met. • The system should automatically save the simulation status such as unlocked level and scoring of each level after the user has press the "Start Simulation" button or triggered the win condition. • The system should unlock the next level after the user has win the current level. • The system should display a panel showing the main menu, level selection and replay level buttons on triggering the losing condition
User Interaction	• The system shall provide the user with the ability to touch screen input interactivity. • The system shall provide the user with the ability to perform certain functionalities using buttons. • The simulation shall provide the user with the ability to tap on the button with a highlighted display. • The simulation shall provide the user with the ability to choose different difficulty level. • The simulation shall provide the user with the ability to choose any level on the level selection screen. • The system shall provide the user with the ability to extend the structure components by dragging and dropping into the simulation world. • The system shall provide the user with the ability to undo placement of structure component. • The system shall provide the user with the ability to reset the "Planning" phase. • The system shall provide the user with the ability to access the menu panel in the level selection and playing scenes. • The system shall provide the user with the ability to access the "Testing" phase from the "Planning" phase by tapping on the green "play" button. • The system shall provide the user with the ability to access the "Planning" phase from the "Testing" phase by tapping on the "restart" button. • The system shall provide the user with the ability to navigate to the level selection menu after triggering the losing and winning condition.

Table 3: (cont)

Functional requirement	Description
	• The system shall provide the user with the ability to navigate to the level selection menu after triggering the losing condition. • The user shall provide the user with the ability to replay the level after triggering the losing condition and winning condition. The user shall provide the user with the ability to navigate to the main menu after triggering the losing condition and winning condition.

Table 4: Non-functional requirement

Non-functional requirement	Description
Usability	• The first-time user shall learn the basics controls of the simulation in the first level. • The simulation should be able to be launched offline.
Performance	• The simulation should be able to be launched under 20 seconds. • The creation of a new save files of the simulation should be under 15 seconds. • The loading of a save file of the simulation should be under 15 seconds. • The loading of a level of the simulation should be under 10 seconds. • The transition from "Planning" to "Playing" phase and vice versa should be under 2 seconds. • The reset of level should take under 2 seconds. • The undo action should take under 1 seconds.
Portability	• The simulation should only be played on Android platform.
Cultural	• The simulation shall use English as the simulation language. • The simulation shall provide Malaysian themed levels.
Capacity	• The simulation size should be under 1.5GB.
Reliability	• The simulation should have 99% reliability while running without any other applications running on the background.

3.8 Software and Hardware Requirement

In this section, the hardware and software requirements are detailed in Table 5. The table listed the minimum requirement of the simulation to be launched in the user Android mobile device.

Table 5: Software and hardware requirement for user

Requirement	Criteria	Description
Software	Operating System	• Android Nougat 7.0 and higher
	Graphics API	• OpenGL ES 2.0+, OpenGL ES 3.0+, Vulkan
Hardware	Display Resolution	• 1280 x 720 px (16:9 ratio)
	CPU	• ARM64
	RAM	• 2GB RAM or higher
	Disk Storage	• 1GB of free disk space or higher
	GPU	• Adreno 405 or equivalent (550Mhz or above) or higher

3.9 System Design

In this section, the structure of the simulation is laid out which will be introducing the Navigational Structure, Flow Chart, and Interface Design.

The navigational structure shown in Figure 2 provides an overview of the navigational flow of the simulation.

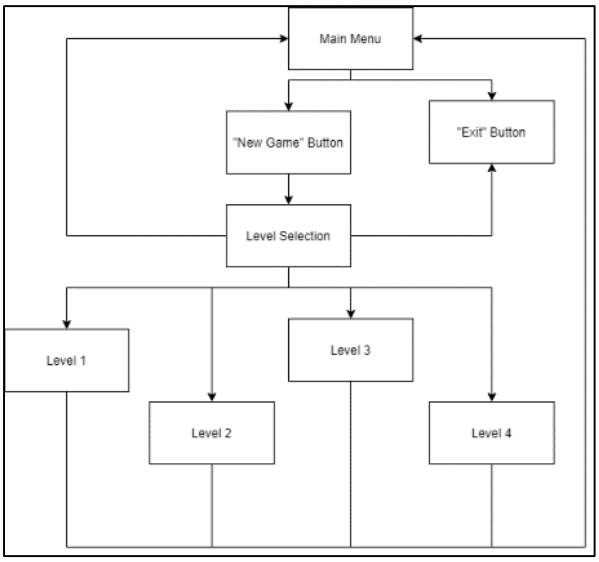

Figure 2: Navigational structure

On the other hand, the flow charts of “Build for All" are illustrated in Figure 3 and 4 to show the user’s action flow as well as the subroutines.

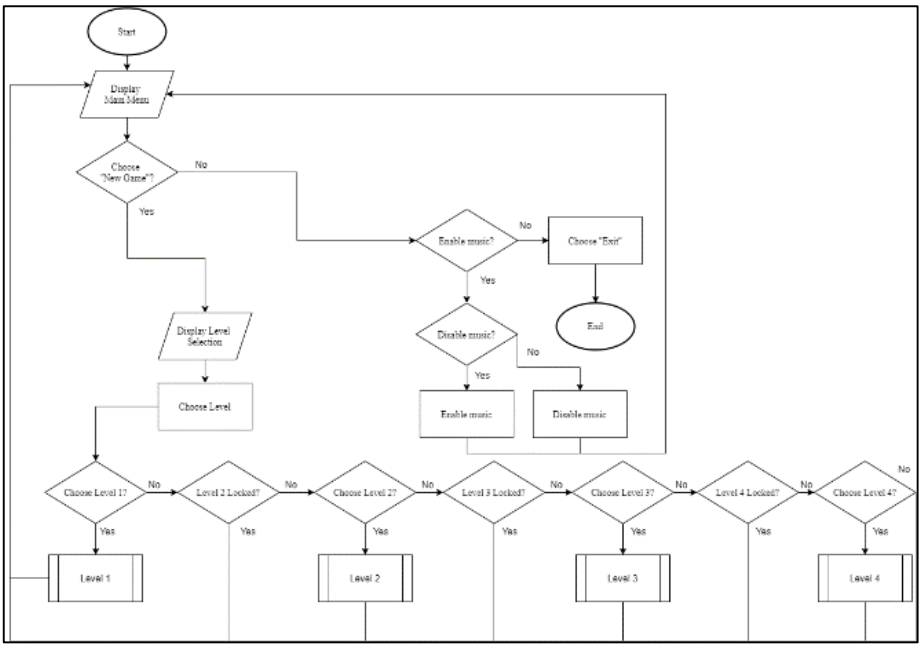

Figure 3: Main Flow chart of “Build for All”

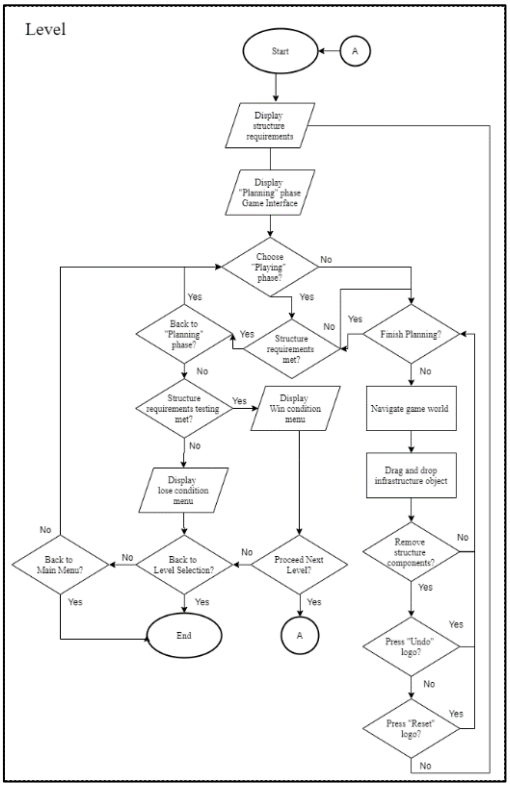

Figure 4: Level Subroutine Flow chart of "Build for All"

Based on Figure 5 and 6, the design theme for the simulation is minimalistic, calming, and pale colour to induce a feeling of calmness. The interface layout contains very few buttons and graphics to follow the minimalistic theme since the mobile screen is small and easily cluttered with interface elements. Since the simulation features universal design, building elements such as budget and infrastructure are featured in the gameplay.

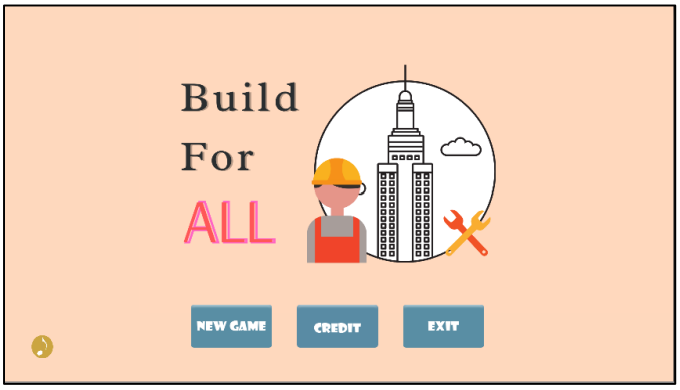

Figure 5: Main Menu of "Build for All"

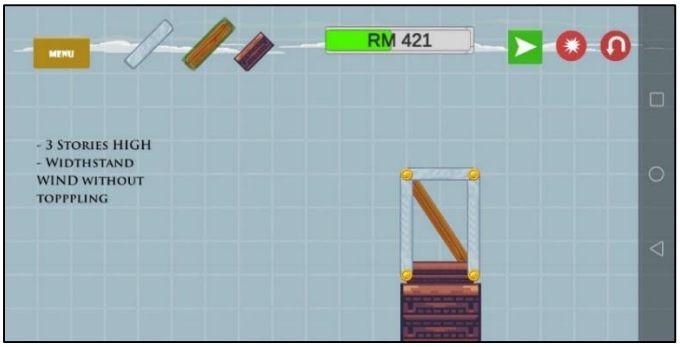

Figure 6: Simulation level of "Build for All"

3.10 Script Integration and Testing

After finishing developing and downloading the assets, the modules including drag and drop building, budget, force exertion, building requirement analysis, wind moving, undo and audio modules are programmed into scripts and integrated with the assets. Figure 7 shows the main gameplay mechanics which is the drag and drop building system in C# and the output, respectively. It starts by checking for finger tapping inputs and starts the bar creation. The position and rotation bar are then updated on each frame based on the finger's position until the release of finger to finalize the position and rotation.

```
public void OnPointerDown(PointerEventData eventData)
{
    if (Time.timeScale != 0)
    {
        return;
    }
    else
    {
        if (Input.touchCount == 1 && BarCreationStarted == false)
        {
                BarCreationStarted = true;
                StartBarCreation(Vector2Int.RoundToInt(Camera.main.ScreenToWorldPoint(eventData.position)));
        }
```

Figure 7: Initializing structure creation on finger tap

4. Result and Discussion

After script and asset integration, alpha and beta testing will begin. Firstly, the functionalities of modules are first tested before releasing to early testers for beta testing. Table 6 depicts the alpha testing results of Build For All.

Table 6: Alpha testing results

Module Test	Expected Result	Actual Result	Corrective Action
Audio Button	Enabling and disabling of music.	Work as intended without errors.	Not needed.
New Game Button	Navigate to the level selection scene.	Work as intended without errors.	Not needed.
Exit Game Button	Exit and closes the simulation application.	Work as intended without errors.	Not needed.
Menu Button	Access the menu panel and pauses the simulation.	Work as intended without errors.	Not needed.
Resume Button	Closes the menu button and resumes the simulation.	Work as intended without errors.	Not needed.
Level Selection Button	Navigate to the level selection scene.	Work as intended without errors.	Not needed.

Table 6: (cont)

Main Menu Button	Navigate to the main menu scene.	Work as intended without errors.	Not needed.
Material Options Buttons	The chosen material will be highlighted, and the subsequent built material is based on the chosen option.	Work as intended without errors.	Not needed.
Budget Slider	The slider value and colour will update accordingly on building and undo.	Work as intended without errors.	Not needed.
Play Button	Resume the time and allow time to pass.	Work as intended without errors.	Not needed.
Reset Button	Restart the level.	Work as intended without errors.	Not needed.
Undo Button	Remove the previous material bar and points objects, as well as updating the budget slider.	The number of previous point objects are removed incorrectly.	Create a C# list to store points history to accurately remove the number of previous points.
Wind Object	Move to the right in a constant speed and applying force to structures in an area.	Work as intended without errors.	Not needed.
Bar Creation Module	Create bar by dragging and dropping to set the start and end points of a bar.	Some coordinates are inaccessible to place point objects.	Round every grid coordinate to integer to standardize the grid.
Win and Lose Condition Module	Check requirements for winning and losing	Work as intended without errors.	Not needed.
Force Exertion Display	Display intensity of red colour based on force exerted onto the material.	Work as intended without errors.	Not needed.

Based on Acyan, Reha and Cigdem [9], a modified System Usability Scale (SUS) is used during the beta testing. Figure 8 shows the 5-point scale ranging from “not fun” to “very fun”, 10 (55.6%), 5 (27.8%) and 3 (16.7%) chooses 5, 4 and 3 respectively, indicating that most of the players are satisfied with the game.

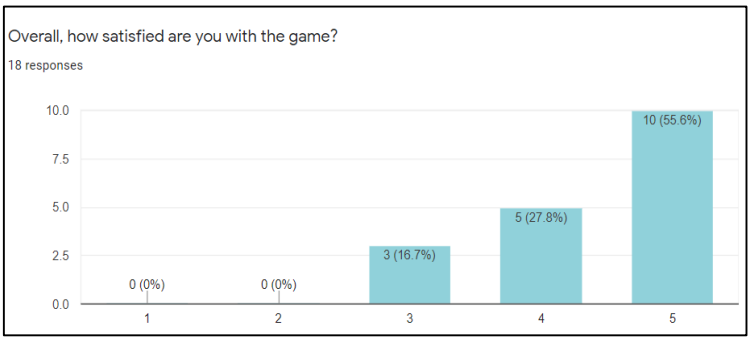

Figure 8: Satisfaction feed backed by testers

Meanwhile, Figure 9 shows there are 8 or 38.9% testers are in favor of the gameplay. The other major favorite aspects are smooth and user-friendly UI and game mechanics, favored by 5 or 27.8% of

the testers. The minor favorite aspects are the easy-to-understand controls and beautiful graphic theme, which both having 2 or 11.1% testers. Bug-free and captivating music aspects only favored by one each (5.6%).

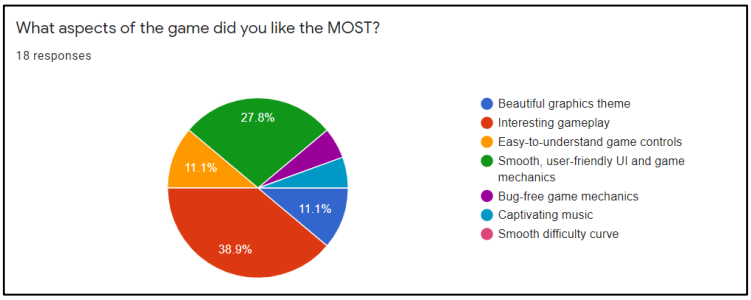

Figure 9: Favorite aspects feed backed by testers

At the same time, drawbacks and recommendation of improvement are also voiced out in Figure 10, such as the simulation being unbalanced or having no difficulty curve as well as clunky UI and game mechanics were majorly voiced out by the 9 or 50% of the testers as shown in Figure 9. 3 or 16.7% of people feels that the gameplay is boring and hard-to-understand controls. Only 2 or 11.1% of testers opined that the music is unsuitable, and 1 (5.6%) tester feels that there are not enough levels.

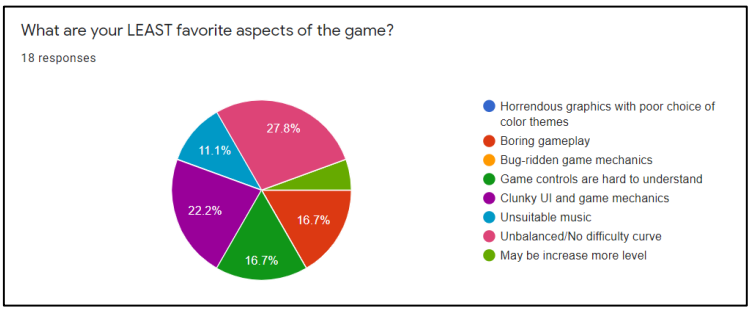

Figure 10: Least favorite aspects feed backed by testers

5. Conclusion

In short, “Build for All” is an Android, 2D educational, construction, budget management, puzzle simulation which will be developed based on GDLC methodology. The simulation has met its objectives which adheres to the requirements of the civil engineering. It can provide a platform for civil engineering practitioners to practice their crafts in a 2D space presented as a construction puzzle. The simulation has also tested through alpha and beta testing which has ensured its quality along with constructive feedbacks.

Feedbacks had assisted a lot in a way to further improve the simulation. Based on positive feedbacks, the simulation has interesting gameplay, beautiful graphic theme, captivating music and smooth, user friendly UI and game mechanics. On the other hand, unbalanced and hard-to-understand game controls are the drawbacks voiced out by the testers.

Therefore, it can be concluded that the application is able to bring forth a valuable platform for many users to learn and build useful skillset and structures for the targeted users which is civil engineering students.

Acknowledgement

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

Appendix A - Storyboard

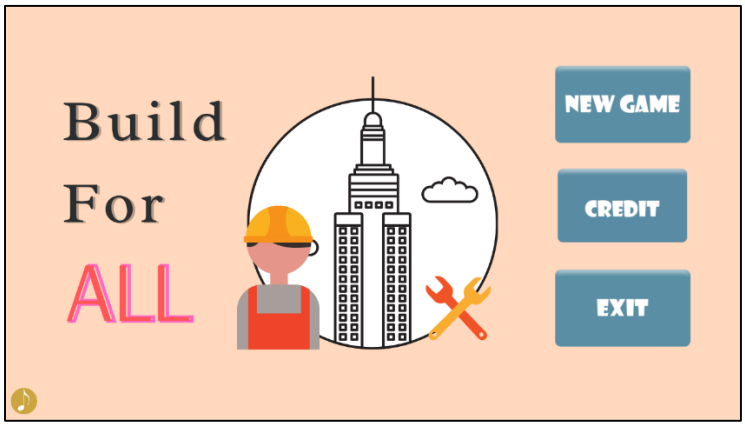

Figure 11: Main Menu Panel

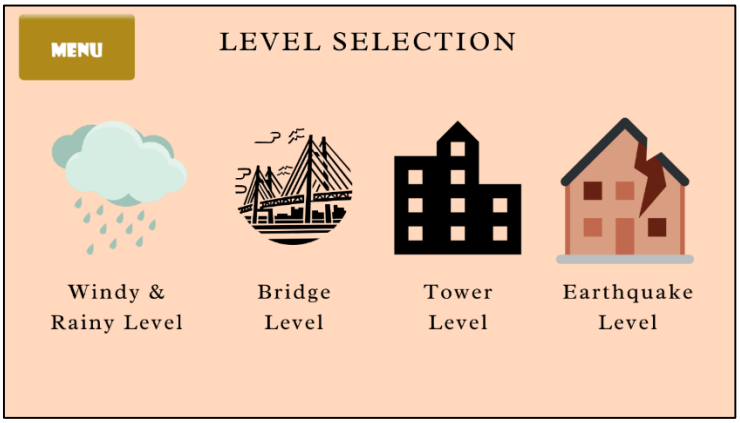

Figure 12: Level Selection Menu Panel

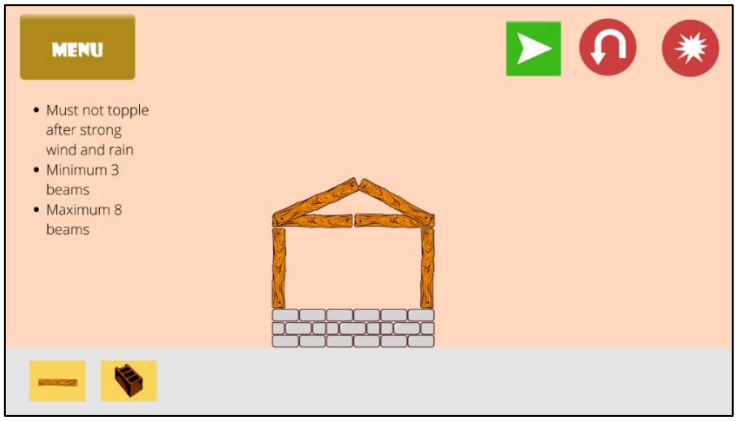

Figure 13: "Planning" phase of simulation level which implements drag and drop building system

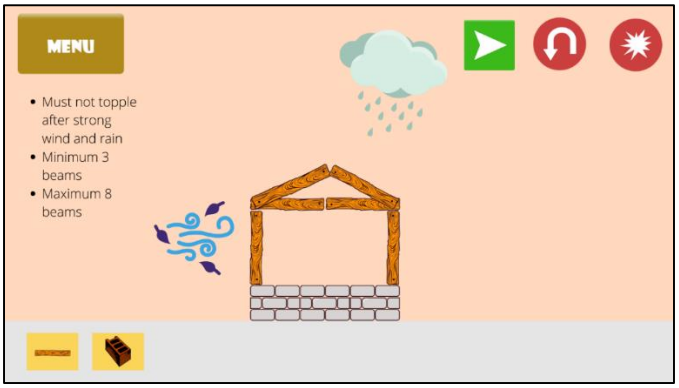

Figure 14: "Testing" phase tests the structural integrity of the building

Figure 15: Win panel on triggering win condition

Figure 16: Lose panel on triggering lose condition

Appendix B - User Testing

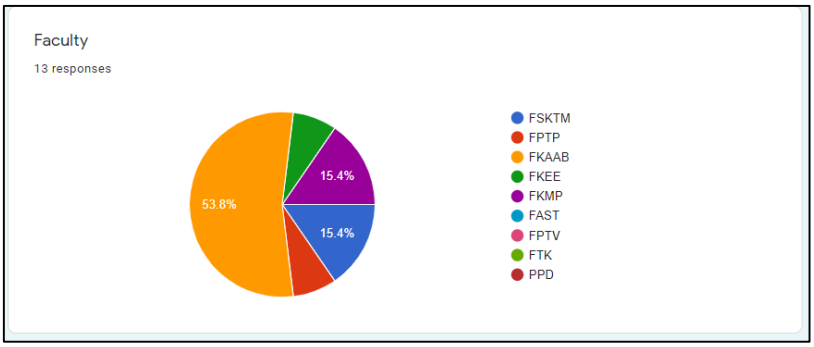

Figure 17: Faculty of UTHM testers

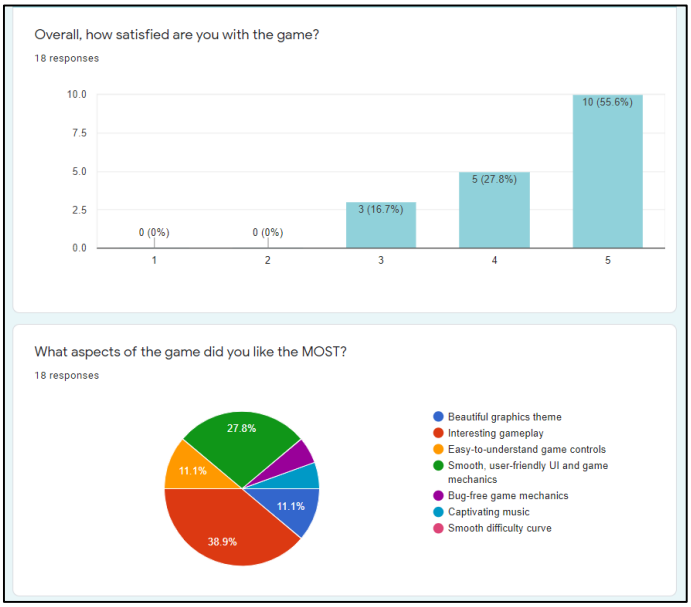

Figure 18: Feedback form tester's positive feedback

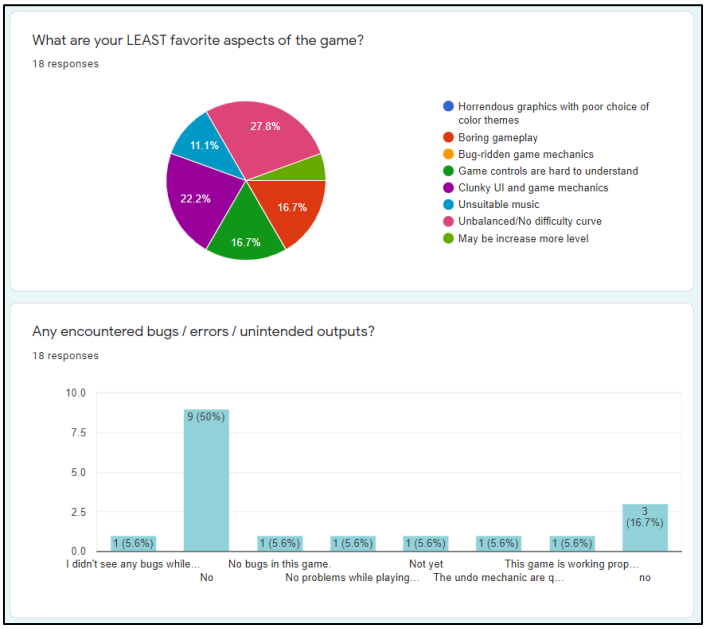

Figure 19: Feedback form tester's negative feedback

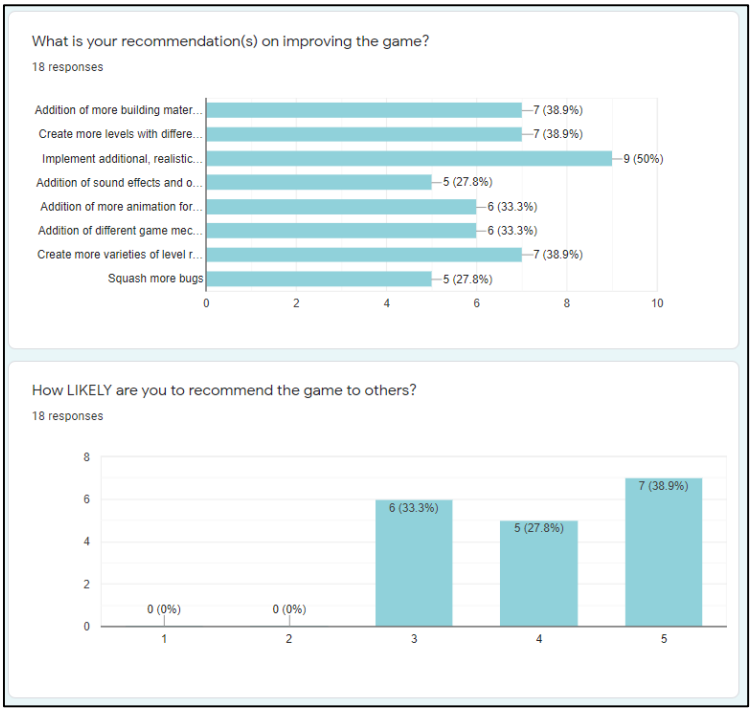

Figure 20: Feedback form tester's recommendation feedback

Appendix C – Supported Platform

Figure 20: Build for All supported on Android device

References

[1] J. J. Connor and S. Faraji, *Fundamentals of structural engineering: Second edition*. 2016.

[2] P. Arthur, R. J. Schmidt, J. Hayton, and K. Hepburn, *Second Moment (Moment of Inertia) of a Plane Area*. 2003.

[3] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, *From Game Design Elements to Gamefulness: Defining "Gamification."* 2011.

[4] C. C. I. Muntean, “Raising engagement in e-learning through gamification,” 2011.

[5] “Build a Bridge! - Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=com.boombitgames.BridgeLowpoly&hl=en&gl=US. [Accessed: 07-Jul-2021].

[6] “Construction World - Build City - Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=com.magicwandstudios.crisiscity&hl=en&gl=US. [Accessed: 07-Jul-2021].

[7] “Construction City 2 – Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=com.heavyfall.constructioncity2&hl=en_IN&gl=US. [Accessed: 07-Jul-2021].

[8] R. Ramadan and Y. Widyani, “Game development life cycle guidelines,” *2013 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2013*, pp. 95–100, 2013, doi: 10.1109/ICACSIS.2013.6761558.

[9] Aycan Kaya, Reha Ozturk, and Cigdem Altin Gumussoy, “Usability Measurement of Mobile Applications with System Usability Scale (SUS),” pp. 389–400, Jan. 2019, doi: 10.1007/978-3-030-03317-0_3.