UNIVERSITI TEKNOLOGI MARA

SCAFFOLDED MOBILE EDUCATIONAL GAME FOR LEARNING CPU SCHEDULING

NAJAH RAIHAN MUSTAFAR @ MOKHTAR

BACHELOR OF COMPUTER SCIENCES (Hons.)
MULTIMEDIA COMPUTING

December 2016

Universiti Teknologi MARA

Scaffolded Mobile Educational Game For Learning CPU Scheduling

Najah Raihan Mustafar @ Mokhtar

Thesis submitted in fulfilment of the requirements
for Bachelor of Computer Sciences (Hons.)
Multimedia Computing
Faculty of Computer and Mathematical Sciences

December 2016

SUPERVISOR APPROVAL

SCAFFOLDED MOBILE EDUCATIONAL GAME FOR LEARNING CPU SCHEDULING

By

NAJAH RAIHAN MUSTAFAR @ MOKHTAR

2014564619

Thesis was prepared under the supervision of the project supervisor, Assoc. Prof. Dr Syed Ahmad Sheikh Aljunid. It was submitted to the Faculty of Computer and Mathematical Sciences and was accepted in partial fulfilment of the requirements for the degree of Bachelor of Computer Sciences (Hons.) Multimedia Computing.

Approved by

……………………..

Assoc. Prof. Dr. Syed Ahmad Sheikh Aljunid

Project Supervisor

JANUARY 31, 2017

STUDENT DECLARATION

I certify that this thesis and the project to which it refers is the product of my own work and that any idea or quotation from the work of other people, published or otherwise are fully acknowledge in accordance with the standard referring practices of discipline.

……………………………………………

NAJAH RAIHAN MUSTAFAR @ MOKHTAR

2014564619

JANUARY 31, 2017

ACKNOWLEDGEMENT

Alhamdulillah, praises and thanks to Allah because of His Almighty and His utmost blessings, I was able to finish this thesis within the time duration given. Firstly, my special thanks goes to my supervisor, Assoc. Prof. Dr Syed Ahmad Sheikh Aljunid for guiding me in finishing this project. Secondly, I would like to thank my project coordinator, Dr Marina Ismail for assisting me in completing this project.

Special appreciation also goes to my beloved parents for giving me moral support and financial support to complete this project.

Last but not least, I would like to give my gratitude to my dearest friends who were always helping me in discussion and suggestion of project ideas.

ABSTRACT

One of the main topics in Operating System course is CPU scheduling which is often considered difficult by Computer Science and Information Technology students to learn and visualize. This is because student found that learning traditionally is boring and non-engaging. Hence, an alternative to design a scaffolded mobile educational game is carried out to solve these problems. In addition, a few scaffolding characteristics are identified to be adapted in this mobile educational game to enhance student engagement. This project focuses on ADDIE model which acquire five phases; Analysis, Design, Development, Implementation and Evaluation. This method approach is systematic and generic for developing learning tools. A total of 20 Computer Science students evaluated the usability and effectiveness of this game prototype. A set of survey questionnaires was given to the students to collect feedback. The results from the survey show that 95% of the 20 students strongly agreed that scaffolding components applied in this mobile educational game prototype which are instruction and hint assist them in understanding CPU scheduling concept while 75% of the respondents positively agreed that this educational game prototype provides easier learning. Thus, students will be more engages and improve their learning in CPU scheduling from this scaffolded mobile educational game.

TABLE OF CONTENTS

CONTENT **PAGE**

CHAPTER THREE: METHODOLOGY

CHAPTER FOUR: ANALYSIS AND DISCUSSIONS

CHAPTER FIVE: CONCLUSION

APPENDICES

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

OS	Operating System
FCFS	First Come First Serve
SJF	Shortest Job First
SRTF	Shortest Remaining Time First
CS	Computer Science
IT	Information Technology

CHAPTER 1

1.1 Project Background

Operating System is among the harder core courses for computing students. This is because this course focuses more on abstract concepts in the underlying operating systems, which are purposely hidden and encapsulated from normal users to simplify their usage but yet required knowledge for computing students. Their knowledge system is complicated and virtual (Liu, J.J, 2010). Learning operating systems involves a lot of memorization and understanding of numerous related concepts.

CPU scheduling is one of the most important topics in operating systems course. However, simply learning from textbooks is boring and non-engaging. Due to these difficulties faced by students, this project is carried out to design a mobile educational game for learning CPU scheduling using the elements of fun and engagement. Scaffolding characteristics are adapted in educational game to improve the effectiveness of learning and understanding of process management in CPU.

Scaffolding is a process of teaching which refers to the way temporary support is provided to guide learners based on communication between expert and learner (Denton, 2014). Some characteristics of educational scaffolding have been identified to help learners. For game educational scaffolding, Aljunid and Jantan (2012) have identified three characteristics. These are (a) game support is given based on students' responses to help them understand and relate new knowledge to the prior one while (b) transfer of responsibility defines that support is gradually removed as students' skills increase and the learning process is continued using their skills independently and (c) instructions are given at appropriate level and continuity to show students what to do for next.

This scaffolded mobile educational game can help students learn CPU scheduling algorithms in an alternative approach filled with fun and engagement factors.

1.2 Problem Statement

Operating System is a core and very hard course for computer science and information technology students to understand and visualize. Its knowledge system is complicated and virtual (Liu, J.J, 2010). CPU scheduling is one of the most important topics in this course. However, learning Operating System traditionally is boring and non-engaging. Textbooks usually provide static representations. Suranauwarat (2015) state that learning CPU scheduling from textbooks is not reliable because students are not able to understand the illustration of CPU scheduling algorithms without concrete examples given. As a result, students are not able to gain insight into exactly how the algorithms work in real-world operating systems. This will contribute to the loss of student’s engagement and attention.

Therefore, a new approach to create a mobile game based learning of CPU scheduling which is engaging and fun based on educational game scaffolding characteristics is mooted. This mobile game application is easy to access and download, and can be played anywhere and anytime to increase the engagement of the Operating System concepts materials pertaining to CPU scheduling.

1.3 Objectives

The objectives for this project are

1. To determine the scaffolding characteristics and game elements for mobile educational game design.
2. To develop a scaffolded mobile educational game for learning CPU scheduling.
3. To test the usability of the mobile educational game prototype.

1.4 Scope

The scope for this project is

1. Focuses on Computer Science and Information Technology students in university.
2. Highlighting the learning of CPU scheduling algorithms in a fun and engagement way in order to increase the understanding, application and analysis of the relevant Operating System concepts.
3. Use Android platform to develop the game.

1.5 Significance

For these students, this project will enable the fun and engaged interactivity of learning CPU Scheduling. Besides that, this project will attract students to learn CPU scheduling from an interesting game design. This project will also help the students to improve their learning in a better way.

1.6 Conclusion

To conclude, this chapter described about the background of this project discussing on how game-based learning can help students to be engaged in learning the concept of Operating System (OS). Some issues about students' problems in learning CPU Scheduling have been identified in the problem statement. In order to solve these issues, the objectives must be achieved to ensure that this project is successful following the scopes provided. Hence, this project will be a good impact to the students for learning CPU Scheduling.

CHAPTER 2

LITERATURE REVIEW

2.0 Introduction

This chapter contains seven subtopics to be discussed. The subtopics are computing education, operating system, CPU scheduling, scaffolding, educational games, mobile educational games, and summary.

2.1 Computing Education

In this education field, Operating System (OS) is the main course for computer science and Information Technology students. Research by Regehr (2010) indicated that students found that learning OS materials hard and challenging. Furthermore, computing education will encourage students to influence the world since the whole world today depends on technology and Internet which emphasize the understanding of operating system (Karnena, 2007).

Also, taking OS course gives students more knowledge and understanding to become a better programmer, a disciplined problem solver, and high confidence level (Saha, 2014). Moreover, students are exposed to the ideas of OS material about data abstraction and complexity (Regehr, 2010). In addition, according to Regehr (2010), computing education provides the students relevant skills and inner thinking about computer systems thus improve their creativity, development skills and able to operate computer hardware through OS interaction.

2.2 Operating System

An Operating System (OS) is the most important program that is initially loaded into a computer. It manages the computer's **memory, processes,** software and hardware. According to Rouse (2014), this program (OS) runs on a computer to manage other application programs by making requests for services through application program interface (API). It also controls the access of the computer system’s resources manager (Bhowmick, 2014). OS allows communication between hardware components of a computer and software components of the computer system to access the computer services (Bhowmick, 2014).

In addition, Microsoft Windows, Apple Mac OS X and Linux are the top three commonly used Operating System (“Understanding Operating System”, n.d). These modern OS use different designed GUI which enables the users to use them easily. Microsoft Windows is the most well-known OS because it comes preloaded on most new PCs in the world today. As for Apple Mac OS X, it comes preloaded on all new Macintosh computers. However Apple computers may be more expensive but many people prefer to use it. Besides that, Linux is a free open-source OS. Modification of this OS can be done freely by anyone. It also provides many different versions to choose such as Ubuntu, Fedora and Mint.

2.3 CPU Scheduling

CPU Scheduling is a process of multiprogramming in Operating System which allows one process to use the CPU while another process waits for I/O to be executed (Silberschatz et al., 2013). Multiprogramming allows multiple processes running at the same time. Although it is possible to run multiple processes simultaneously, CPU can execute only one instruction at a time. Hence CPU scheduling decides which process should occupy the resources in order to maximize the CPU utilization (Warren, 2007). Ayoma (2010) explained that it is required to share the CPU among available processes to able each process get a chance to be executed. Furthermore, the scheduling affects the system's performance which means to make the overall system efficient, fast and fair (Silberschatz et al., 2013).

2.3.1 Criteria

Scheduling system comprised of different criteria to achieve when considering the best scheduling algorithm. Each criteria gives different result based on different scheduling algorithm ("Types Of Scheduling", 2010). There are six criteria identified to compare the scheduling algorithms performance.

a) CPU Utilization

To maximize the CPU utilization, the CPU needs to be kept as busy as possible. CPU will be working most of the time and not waste any CPU cycle. On a real system CPU usage should range from 40% (lightly loaded) to 90% (heavily loaded).

b) Throughput

To increase throughput which means to increase the total number of processes completed per unit time.

c) Turnaround Time

Turnaround time refer to the amount of time taken to execute a particular process and need to be minimized.

d) Waiting Time

Minimize waiting time define that the amount of time a process wait in the ready queue to acquire the CPU need to be decreased.

e) Load Average

To minimize load average, the CPU needs to decrease the number of processes residing in the ready queue waiting for their turn.

f) Response Time

Minimize the time taken from when a request submitted until the first response is produced.

2.3.2 Scheduling Algorithms

Scheduling algorithms are used to make the system efficient and fair allowing processes to fully utilize the CPU (Silberschatz et al., 2013). It can be divided into two categories which are Preemptive Scheduling and Non-Preemptive Scheduling.

According to Ayoma (2010), for Preemptive Scheduling, processes are prioritized based on different factors and CPU manages the process with highest priority to be given the chance of being executed while for Non-Preemptive Scheduling, the CPU is able to control single process until the process is complete executed before allocates another process. There have been several scheduling algorithm applied in multiprogramming system.

2.3.3 Types of Scheduling Algorithms

Among the Non-Preemptive Scheduling algorithms are First Come First Serve, Shortest Job First and Priority while the Preemptive Scheduling algorithms consists of Shortest Remaining Time First, Round Robin, Multilevel Queue, Multilevel Feedback Queue and Priority.

First Come First Serve (FCFS) is a non-preemptive scheduling algorithm which means the process that arrives first is executed while the CPU is not allocated to another process until the current process is complete. It is simple and easy to implement but poor in performance because it contribute high average waiting time.

Shortest Job First (SJF) is also a non-preemptive scheduling algorithm which emphasizes the shortest process to be executed first. This algorithm chooses the next smallest fastest process to do next. It is considered as the fastest scheduling because it minimizes waiting time. However it is impossible to implement because there is no way to know the exact length of next CPU burst.

Priority scheduling can be either preemptive or non-preemptive. In non-preemptive priority scheduling, CPU executes process that arrives with the highest priority after completing the present running process while in preemptive priority scheduling, the running process is preempted by a new arrival of higher priority process. Both provide good response for the highest priority processes but it may starve lowest priority processes which the low priority processes need to wait forever because there are always other processes around that have higher priority.

Shortest Remaining Time First (SRTF) is a preemptive scheduling which is similar to the concept of SJF which allows process with shortest burst time to execute first. However for SRTF the arrival of shorter process cause the existing running process preempted. This scheduling algorithm provides high throughput but longer process may starve for execution.

Round Robin scheduling assigns time quantum for each process. If process finishes its burst before the time quantum timer expires, then it is swapped out of the CPU and continues allocate other process. If the timer goes off first, then the process is swapped out of the CPU and moved to the back end of the ready queue. This preemptive scheduling gives all processors to share the CPU equally and provides good response time for short processes.

Multilevel Queue scheduling is established in multiple queues. One queue is scheduled to get time relative to the other queues. It is preemptive scheduling because process cannot switch from queue to queue.

Multilevel Feedback-Queue allows process to move from one queue to another for some circumstances. For example, if the process changes between CPU-intensive and I/O-intensive, it will switch queue. This scheduling is the most flexible for multiprogramming because it can be tuned for any situation. However the implementation is complex due to all the adjustable parameters.

2.4 Scaffolding

Scaffolding describes a temporary support provided by one person to another (Denton, 2014). On the other hand, Sandhana (2009) mentioned that the support in scaffolding techniques is provided by an expert to a learner. In education, scaffolding refers to teaching techniques which provides ongoing support for learners to master a new concept or to complete a task ("Scaffolding", 2011).

According to Pinantoan (2013), a research done by psychologist Lev Vygotsky found that scaffolding can be used to measure the two level of learners called "Zone of Proximal Development" which are 1) the actual development that has been established by completed development cycles and 2) the potential development which is the expected outcome of learner's improvement under collaboration with more capable peers.

2.4.1 Advantages of Scaffolding

Instructional scaffolding is often used in classroom and it provides supportive learning environment for students (Meyer & Turner, 2002). For example, teachers may give explanations and examples or prepare handouts and hints to improve the student's ability towards understanding courses and achieving goals independently (Alibali, 2006). The most importance is students are allowed to do as much of the task as possible while the teachers only attempt to help them with tasks that are beyond their current capability (Lipscomb, Swanson & West, 2015).

Scaffolding also can reduce negative perceptions that students may experience when they get discouraged or having difficult task without any help ("Scaffolding", 2015). Having students demonstrate the task independently will determine whether the students are learning or not (IRIS, n.d).

2.4.2 Characteristics of Scaffolding

There are three characteristics of scaffolding described by Denton (2014) which are a) contingency b) fading and c) transfer of responsibility.

a) **Contingency** describes how the experts aware of the learner's current level of ability. This can be done through observation or assessment to acquire learner's understanding.

b) **Fading** is the gradual withdrawal of contingencies, or supports. Fading occurs as the expert observes that the learner is becoming competent and allow learner to work independently.

c) **Transfer** of responsibility refers to the learner’s new knowledge or skill so that there is a change in cognitions or affect. Transfer of responsibility is also directly related to the learner’s control over subject matter and perceptions of self-efficacy.

2.4.3 Guidelines Of Scaffolding

Larkin (2001) states that scaffolding is one of effective instruction that enable teacher to convey information to fit individual student needs. There are eight guidelines suggested by Larkin (2001) which are most of teachers follow in developing scaffolded lessons.

a) Develop suitable task that match the curriculum goal.
b) Define a shared goal for all students to achieve through engagement in specific tasks.
c) Identify student interest or needs and observe improvement based on those abilities.
d) Provides instruction that student understand.
e) Provide encouragement and praise as well as ask questions and have students explain their progress to help them stay focused on the goal.
f) Give feedback to allow the student monitor their own progress.
g) Create an encourage environment where students comfortable taking risks without fear
h) Promote responsibility for independent learning.

2.5 Educational Game

Game has been widely used in recent educational field for teaching. According to Wang et al. (2009), educational game is commonly used in university and school as an alternative to traditional teaching method. It is proved by some research works that learners become more motivated and interested in learning by playing games (Tlili, Essalmi, Jemni & Kinshuk, 2015).

According to Wen and Ching (2008) game is interactive and provides graphical user interface (GUI) that attract learners to more positive and spontaneous learning. Most game is effective and practical for learners to experience (Wen & Ching, 2008).

Besides that, there are various types of game genre which can be used as educational game. For example, simulation game is modeled to match real world situation (Grace, 2005). One of the most common situations for simulation is management simulation. In addition, "SimSE" is an example of computer-based game environment for learning software engineering (Navarro, Baker & Hoek, 2004). This game is a single player game which allows the player to be a manager of a company and manages all the activities of his employees (aspect of software engineering process).

2.5.1 Advantages of Educational Game

Playing educational game helps learners with self-esteem, concentration and memory as they can see how far their achievement in each completed game stage (Schuna, 2015). According to Persky (2001) educational game trains learners to work on problem solving towards achieving goals. By playing game, learners can work in passionate and enjoyable environment.

2.6 Mobile Educational Game

Scholars and researchers from the educational gaming field have recognize that mobile gaming industry is considered as valuable tools for learning (Santamarina et al., 2010). Furthermore, mobile educational game is perceived as a learning tool which emphasize on integration of educational game into mobile learning to produce an informal educational to be engaging and effective (Molnar, n.d).

In addition Klopfer (2011) mentioned that the strength of mobile platform is acceptable in developing educational game. The mobility of this device allows distribution of education to be out of school and accessed by learners anytime anywhere (Santamarina et al., 2010)

Besides that, there are various types of mobile educational game development which offer an engaging environment for learners. For instance, augmented reality mobile game is created to expose learners the interaction between real world environment and virtual information to enhance the learning of complex contents (Klopfer, 2011).

2.6.1 Characteristics in Scaffolding Mobile Educational Game

For game educational scaffolding, Jantan & Aljunid (2012) have identified three characteristics which are a) games scaffold student's effort, b) gradual transfer of responsibility and c) instruction at appropriate level and continuity.

a) Game scaffold student's effort

In scaffolding, support is given to help learners relate new knowledge to the prior one. As mentioned by Lipscomb et al. (2004) the support is given to allow learners complete task as much as possible. The supports are given based on the learner’s response towards achieving goals.

b) **Gradual transfer of responsibility**

This characteristic is implemented in scaffolded educational game where the support is gradually removed as students’ skills increase and the learning process is continued using their skills independently.

c) **Instruction at appropriate level and continuity**

Instructions are given to show learners what to do next. This is parallel to game design where Denton (2014) state that providing hints, feedbacks and clues advances the learner’s performance.

2.7 Summary

To conclude, this chapter provides a brief description of the introduction of literature review, operating system education, scaffolding, educational game and mobile educational game. This chapter shows the ideas and reviews of the researcher related to this project. Besides that, this chapter guides the expected outcome of this project.

CHAPTER 3

PROJECT METHODOLOGY

3.0 Introduction

This chapter focuses on methodology in development of mobile game for learning CPU scheduling on iOS platform. Methodology defines a systematic and theoretical analysis of methods applied to carry out this project. This chapter explains five stages of the project which are analysis, design, development, implementation and evaluation.

3.1 Project Methodology

This project apply ADDIE model as a guideline to design scaffolded mobile game for learning CPU scheduling. This model is often used by professional instructional designers for technology-based teaching. It focuses on clear learning objectives, good quality design and well-structured contents. Hence ADDIE is the most suitable method approach for building effective learning tool.

3.2 ADDIE Model

ADDIE model comprise five important phases which are Analysis, Design, Development, Implementation and Evaluation. Table 3.2 shows the step-by-step framework of these five stages.

Table 3.2 Project Framework Diagram

Stage	Task	Activities	Deliverables
Analysis	• Identify problem statement, objectives, target audience, scope and significance	• Discuss the flow of project with supervisor • Review articles and journals	• Problem statement and target audience identified. • Objectives identified • Scope identified • Significance identified
	• Understanding concept of OS & CPU scheduling, purpose of mobile educational game and scaffolding in educational game	• Identify related game genre for this project • Review related articles on scaffolded educational game	• Reviewed Literature on OS & CPU scheduling algorithm • Reviewed Literature on concept of scaffolding • Reviewed Literature on educational and mobile game
Design	• Design a scaffolded educational game for learning CPU scheduling	• Designing game interface	• Game Storyboard • Game interface • Scaffolding characteristics implemented. • Instruction given to

			explain how the game works • Hints given to support player • Feedback pop to encourage player
Development	Game development	• Develop game prototype Tool • GameSalad Creator	• Game prototype developed
Implementation	• Pre-testing the prototype • Analyze prototype’s problem	• Test prototype on Android • List prototype’s problems	• Prototype limitations
Evaluation	• Testing game prototype • Determine effectiveness and improvement of prototype	• Collect feedback and survey • Review training effectiveness • Report performance results	• User feedback • Project evaluation report

3.3 Project Activities

Project activities provide further explanation about the activities of each phase in the project framework as shown in Table 3.2.

3.3.1 Analysis

Analysis is the first phase which requires the study of problems that related to this project. This phase is a process in gaining knowledge and information through reviewing articles and journals. Besides that, since scaffolding will be the main element adapted in this project, analysis is done to gain more understanding of scaffolding in mobile educational game development.

By the end of this phase, problem statement, objectives, scope and significance are identified and listed as a guide for the next phase.

3.3.2 Design

Second phase in this project is design. This phase includes the activities of selecting a suitable game genre for CPU scheduling and designing game interface. The flow of the game was also briefly designed. Storyboard and game interface were the deliverables of this phase. Scaffolding characteristics and game elements were also identified which were implemented to this project.

Three scaffolding characteristics identified are

1. Scaffold learner's effort.
2. Instruction at appropriate level.
3. Gradual transfer of responsibility.

Five game elements identified are

1. Feedback is the hallmark of games provided for players to encourage them playing.
2. Challenge where game challenges players to achieve goals and outcomes which are not simple and straightforward.
3. Rules define the game which is the structure that allows the artificial construct to occur.
4. Level indicates game progress which serves as a marker for players to know where they stand in a gaming experience over time.
5. Linearity which refers to the degree to which a game restricts player freedom or control.

Hence, at the end of this phase, objective 1 which is to determine the scaffolding characteristics and game elements of mobile educational game design was achieved.

3.3.3 Development

Development is the third stage in this project. Development of game prototype was carried out in this phase. The tool needed to develop the game is GameSalad Creator. For this phase, the deliverable is the mobile game prototype. Thus, objective 2 which is to develop mobile educational game was achieved.

3.3.4 Implementation

The fourth phase is implementation where the game prototype was pre-tested on Android platform. The prototype was also analyzed in order to detect any problems that may arise during the testing. The problem that may occur was listed such as graphic size and user interface.

3.3.5 Evaluation

Evaluation was the last phase which survey and testing had been done to collect feedback from user. The delivery of the prototype was observed in order to improve the effectiveness of the game prototype. Final evaluation determines how the prototype was developed. Hence, at the end of this phase the usability of the prototype was evaluated and objective 3 was achieved.

3.4 Documentation

Documentation is the last stage and most important for this project. This stage includes report writing and slide preparation for final presentation. A complete final year report and project slide is delivered by the end of this project evolution.

3.5 Summary

To conclude, this chapter explained the project methodology pertaining ADDIE model for project method which includes stage of Analysis, Design, Development, Implementation and Evaluation. Thus, an effective game prototype was successful created by using this method and correct tool.

CHAPTER 4

DISCUSSION AND RESULTS

4.0 Introduction

This chapter discusses the discussion and results of this project. This chapter explains in details the initial process of project development and results achieved from this project. This chapter covers on scaffolding component for mobile educational game, develop a scaffolded mobile educational game for learning CPU scheduling as well as evaluation of the game prototype. The discussion and results are based on the objectives as mentioned in Chapter 1.

4.1 Objective 1: To determine the scaffolding component for mobile educational game design

The first objective of this project was achieved by applying scaffolding elements in the design of mobile educational game for learning CPU scheduling. The game was designed with a few scaffolding elements added which make this mobile educational game different from other mobile educational game design. The scaffolding characteristics identified from the elements applied are (S1) Scaffolds student's efforts, (S2) Instruction at appropriate level and (S3) Gradual transfer of responsibility.

4.1.1 Scaffolding elements in game design

There are several game elements that meet the characteristics of scaffolding implied in the mobile educational game design such as levels, rules, challenge, feedback and linearity. Table 4.1 shows the mapping of scaffolding characteristics with the game elements.

Table 4.1

Scaffolding Characteristics	**Game Elements**
S1. Scaffold learner's effort	1. Feedback 2. Challenge
S2. Instruction at appropriate level	3. Rules
S3. Gradual transfer of responsibility	4. Level 5. Linearity

Each of the scaffolding elements were presented in the game storyboard. Storyboard will explain each of the scaffolding elements and their characteristics. Figure 4.1 until Figure 4.9 explain the storyboard.

Figure 4.1 show the **Level** scene presents the third scaffolding characteristic which is **(S3) Gradual transfer of responsibility**. Students need to play the game level by level. They are responsible to complete each level in order to proceed to next level.

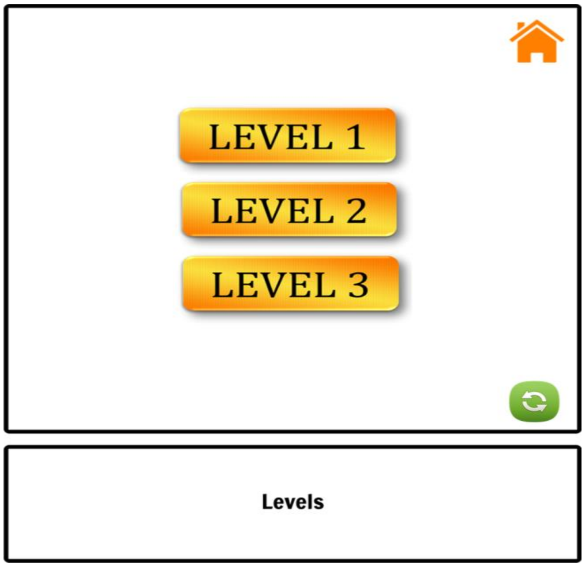
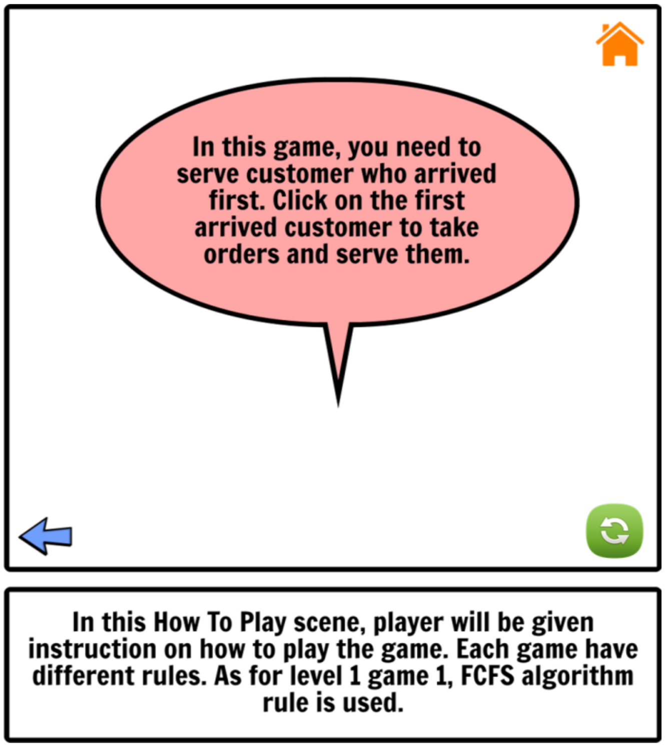


Figure 4.1 (S3)

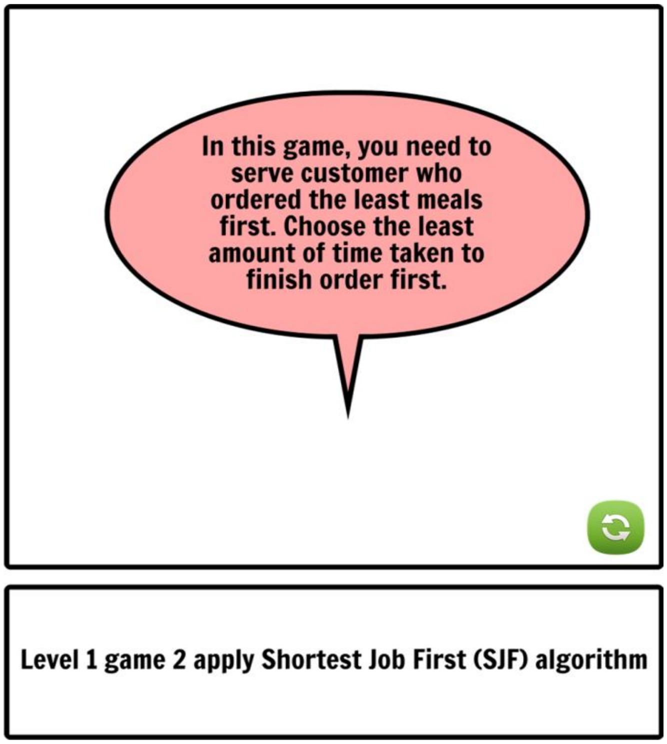
Figures below show the ‘How To Play’ scene define **rules** where **(S2) Instruction at appropriate level** is given. The game rules will be shown if button ‘How To Play’ is clicked. The rule was given to explain the player on how to play and how the game works. Each level consists of different game rules. Figure 4.2, 4.3 and 4.4 explain different rules for each different game.



In this game, you need to serve customer who arrived first. Click on the first arrived customer to take orders and serve them.

In this How To Play scene, player will be given instruction on how to play the game. Each game have different rules. As for level 1 game 1, FCFS algorithm rule is used.

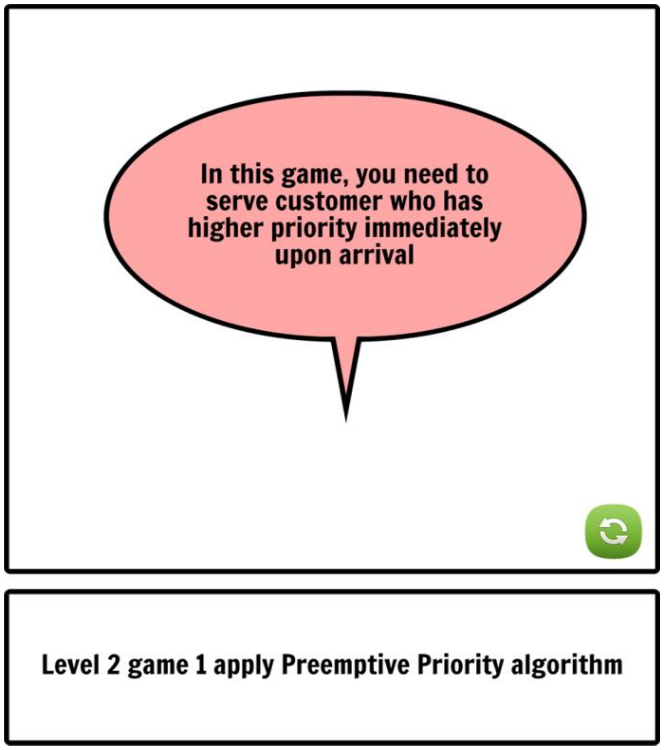
Figure 4.2 (S2)



In this game, you need to serve customer who ordered the least meals first. Choose the least amount of time taken to finish order first.

Level 1 game 2 apply Shortest Job First (SJF) algorithm

Figure 4.3 (S2)

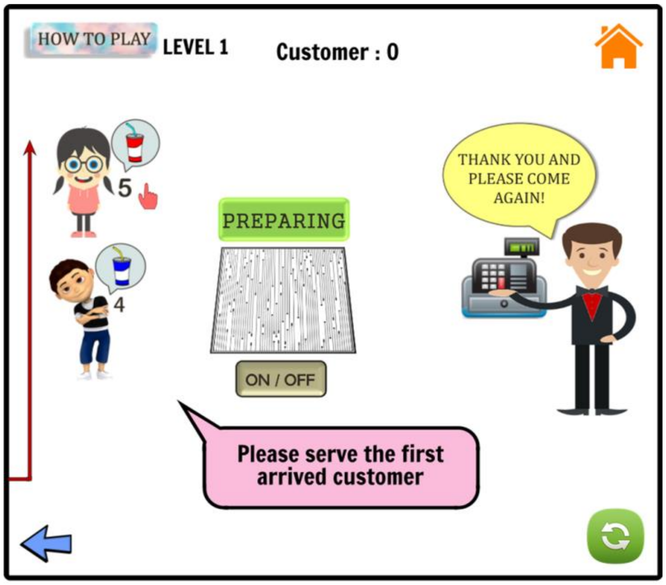


In this game, you need to serve customer who has higher priority immediately upon arrival

Level 2 game 1 apply Preemptive Priority algorithm

Figure 4.4 (S2)

Each game will provide hints or clues to player. **Hints** will be given based on the player's performance. If the player seems continuously do mistakes or take too much time in playing, hints will be popped out. The hint also satisfied the scaffolding characteristic which is **(S2) Instruction at appropriate level**. If the player is able to play well, hint is not given. Figure 4.5 and Figure 4.6 show the game hint.

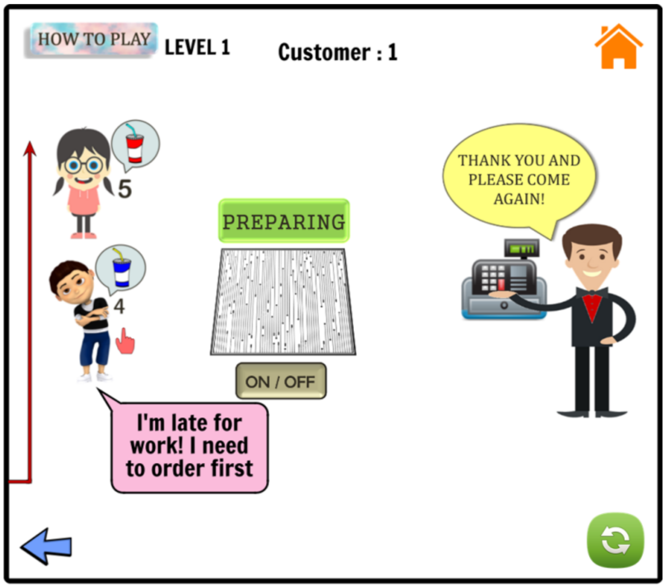


THANK YOU AND PLEASE COME AGAIN!

Please serve the first arrived customer

If player choose wrong customer, a pop up box appears to instruct player to choose the correct customer based on the rule given.

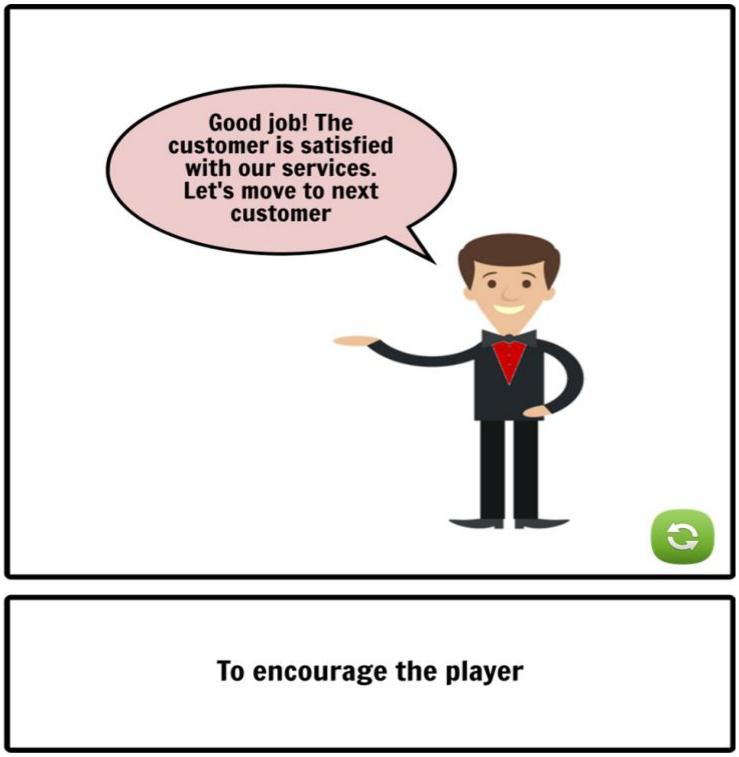
Figure 4.5 (S2)


HOW TO PLAY
LEVEL 1
Customer : 1
5
4
PREPARING
ON / OFF
THANK YOU AND PLEASE COME AGAIN!
I'm late for work! I need to order first

If player seems to have problem in choosing the right customer, hint is given (pop up dialogue box)

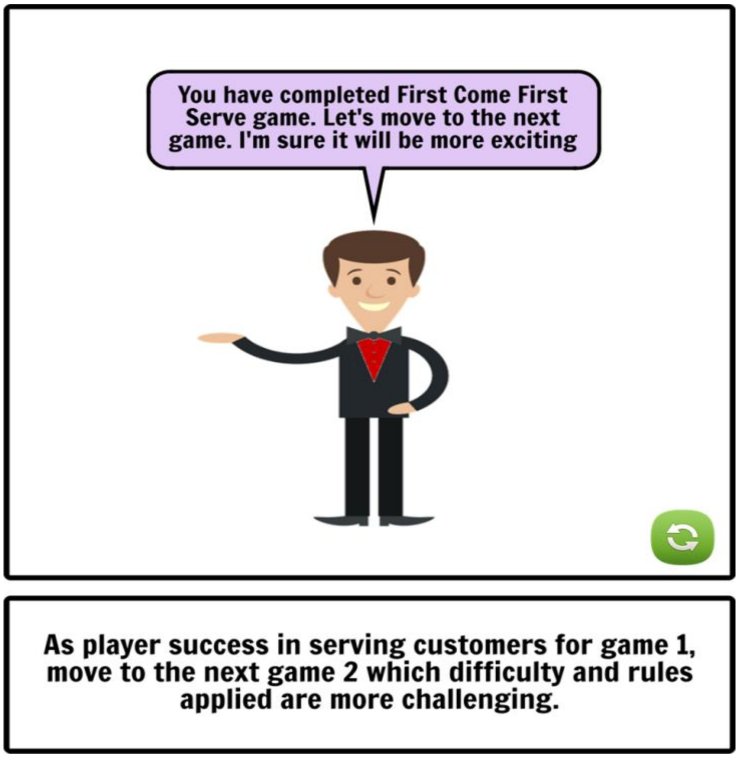
Figure 4.6 (S2)

Feedback is the most important game element in game design. It is also considered as scaffolding component because it **(S1) Scaffolds student's effort**. Feedback shows the most crucial scaffolding component in educational game design because it encourages students to keep trying and complete a certain task. Figure 4.7 and Figure 4.8 show example of feedback in this game design.


Good job! The customer is satisfied with our services. Let's move to next customer

To encourage the player

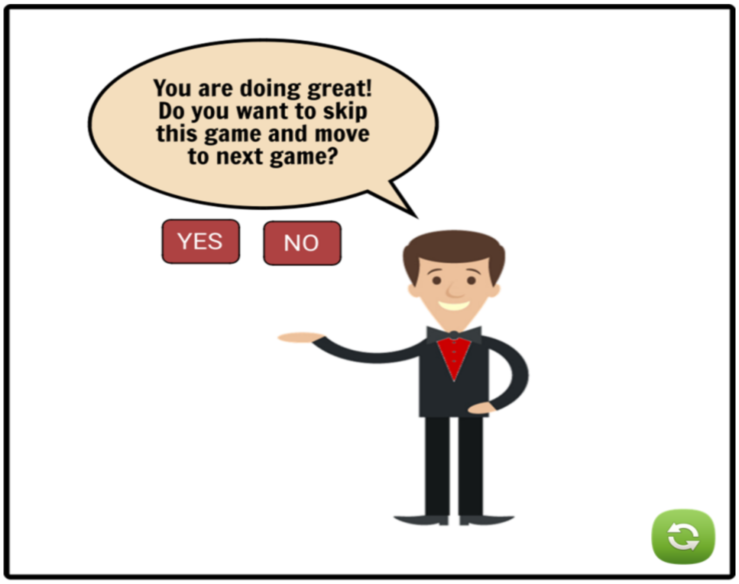
Figure 4.7 (S1)


You have completed First Come First Serve game. Let's move to the next game. I'm sure it will be more exciting

As player success in serving customers for game 1, move to the next game 2 which difficulty and rules applied are more challenging.

Figure 4.8 (S1)

If the player seems to play well and understand the game flow, a **chance** to skip current level is linearly given. This chance satisfied scaffolding characteristic **(S3) Gradual transfer of responsibility** because support given is gradually decreases as player manages to understand the concept well. Figure 4.9 shows example of chance given to the player.


You are doing great! Do you want to skip this game and move to next game?
YES
NO

If player seems to understand playing the game correctly without much hint given, an option is provided whether to continue finish current game or skip and move to next game.

Figure 4.9 (S3)

4.2 Objective 2: To develop a scaffolded mobile educational game for learning CPU scheduling.

The second objective of this project was achieved by designing and developing a scaffolded mobile educational game for learning CPU scheduling. The tool used in this game development is GameSalad Creator.

4.2.1 Game design interface

Figures 4.10 until Figure 4.30 show the game interface. The interface consists of home screen, level screen, tutorial screen, recap screen and play screen. The flow of the game is shown in user manual

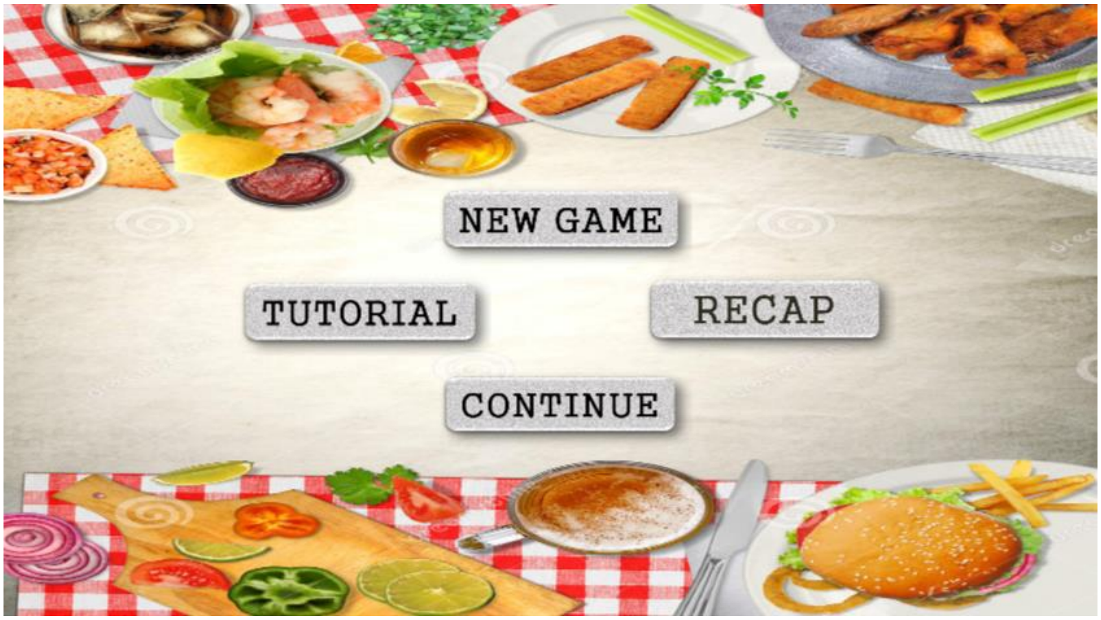


Figure 4.10

Figure 4.10 shows home screen which consists of four buttons which are new game, tutorial, recap and continue. New game button refer to play screen as shown in description below.

New game screen

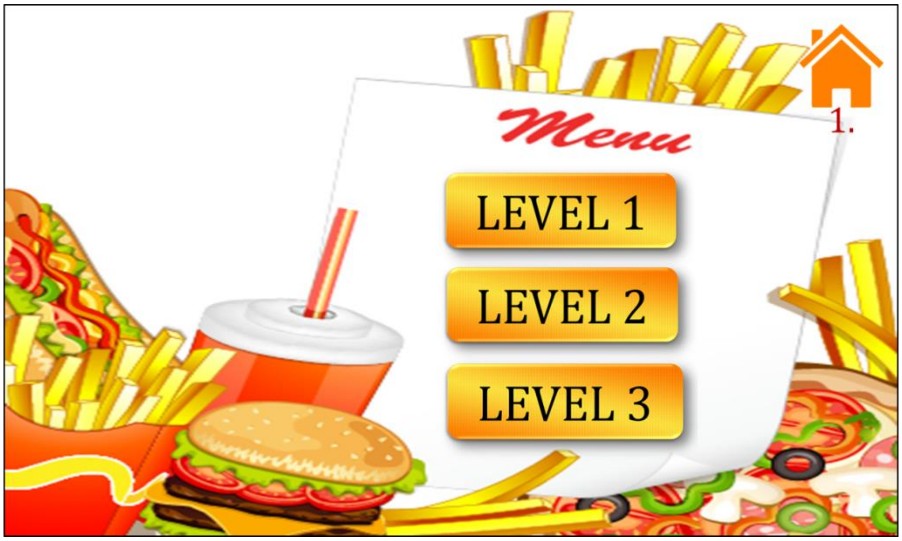


Figure 4.11 (S3)

Description:

This new game screen consists of level screen and play screen. Figure 4.11 shows the level screen. Each level consists of different game. Each game has different rules applied. The details described below based on labelled number.

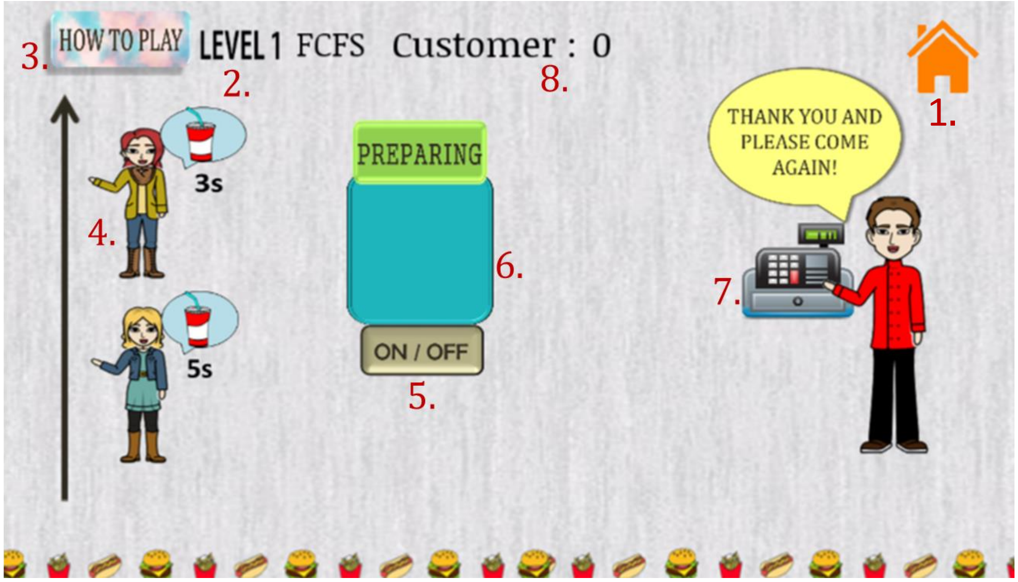
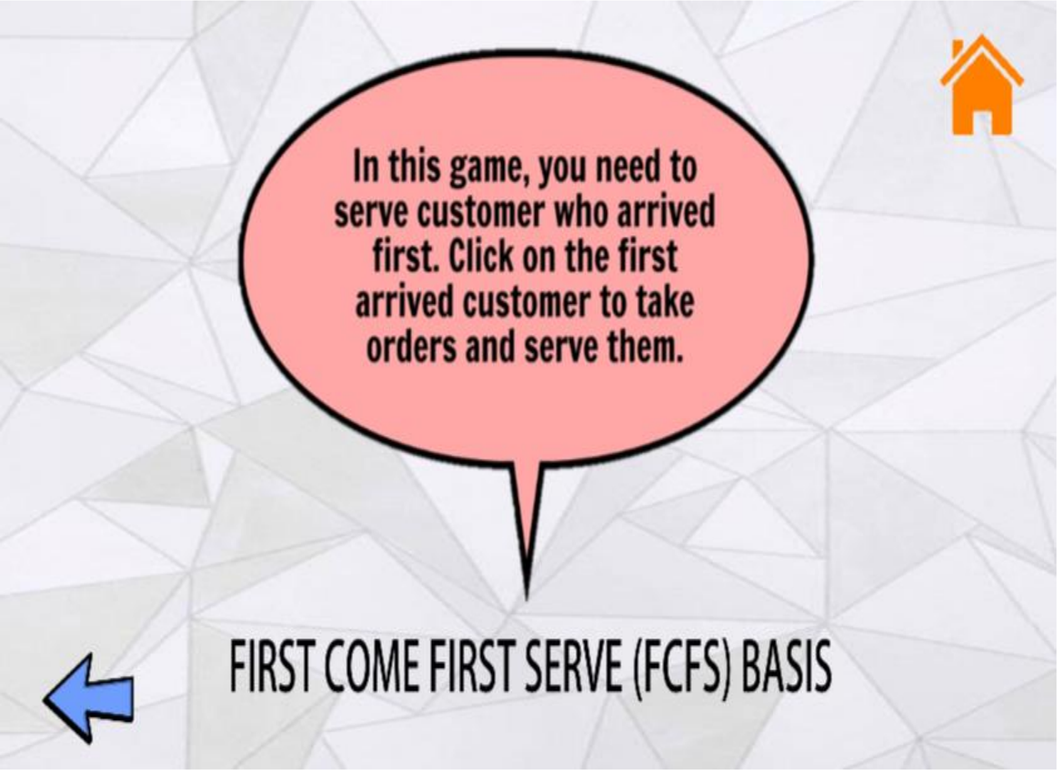


Figure 4.12

Description:

Figure 4.12 shows the play screen for level 1. The details in this play screen described below based on labelled number.

1. This icon represent home button which redirects the player to home screen.
2. Present the current level of the game played.
3. This button act as a guide button which shows player the game rules in this level.
4. This is the customer who arrives with order along with serving time for her order.
5. On/Off button to start and stop preparing the customer's order.
6. Selected customer is placed in this blue area (machine) before player start serving the order.
7. Cash machine where customer pays the bill once her order finished.
8. Presents the number of served customer. The number will increase as more customers successfully served.



In this game, you need to serve customer who arrived first. Click on the first arrived customer to take orders and serve them.

FIRST COME FIRST SERVE (FCFS) BASIS

Figure 4.13 shows game rules for level 1 and (S2)

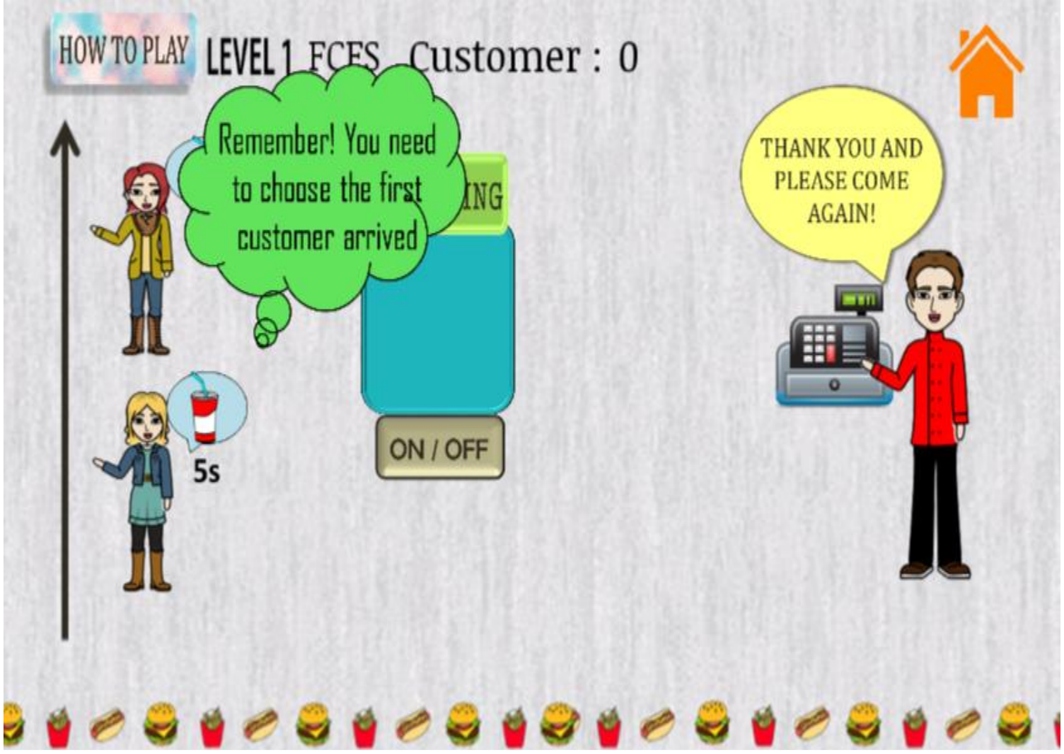

HOW TO PLAY
LEVEL 1 FCFS Customer : 0
Remember! You need to choose the first customer arrived
ING
5s
ON / OFF
THANK YOU AND PLEASE COME AGAIN!

Figure 4.14 shows game hint in level 1 and (S2)

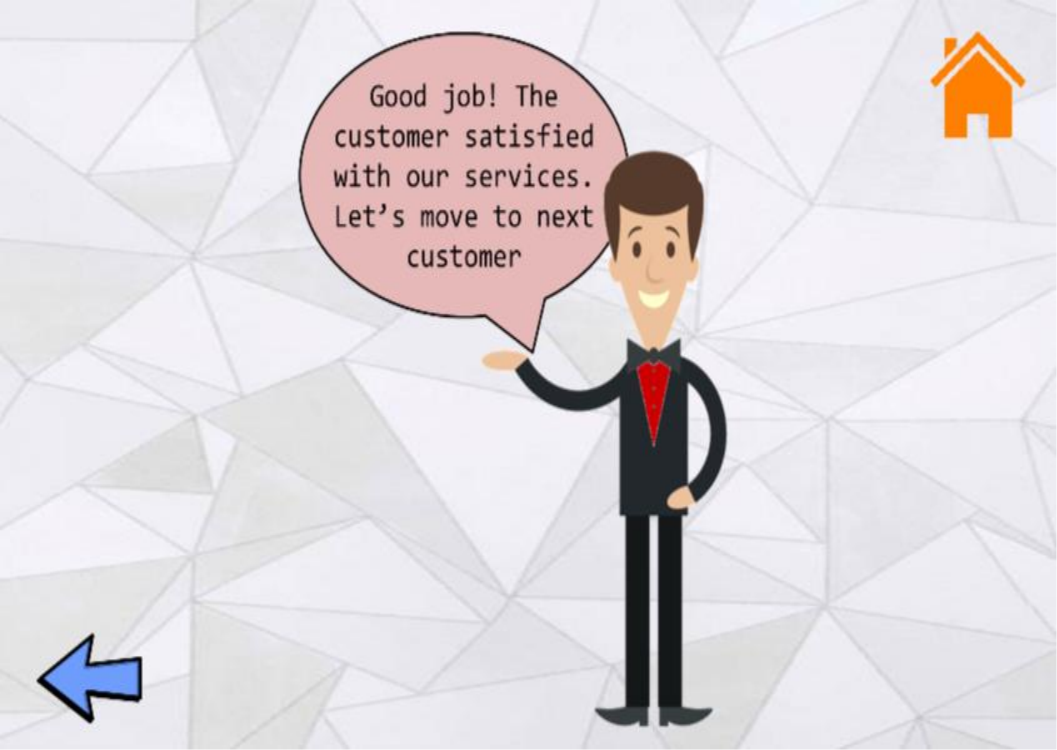

Good job! The customer satisfied with our services. Let's move to next customer

Figure 4.15 shows game feedback in level 1 and (S1)

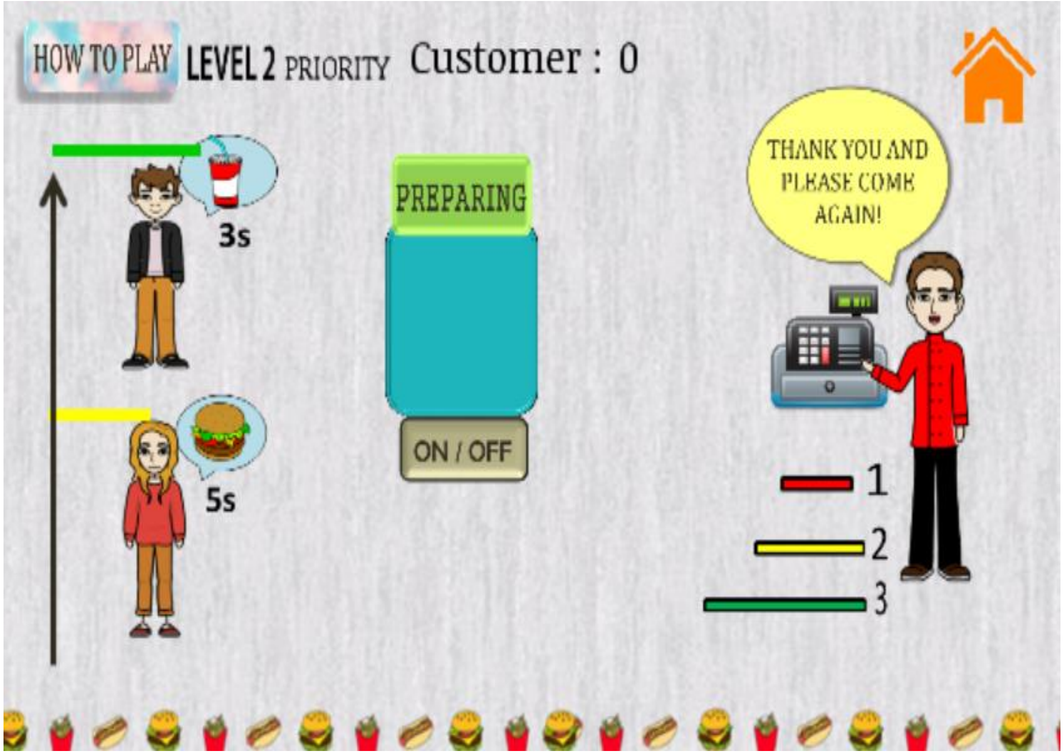
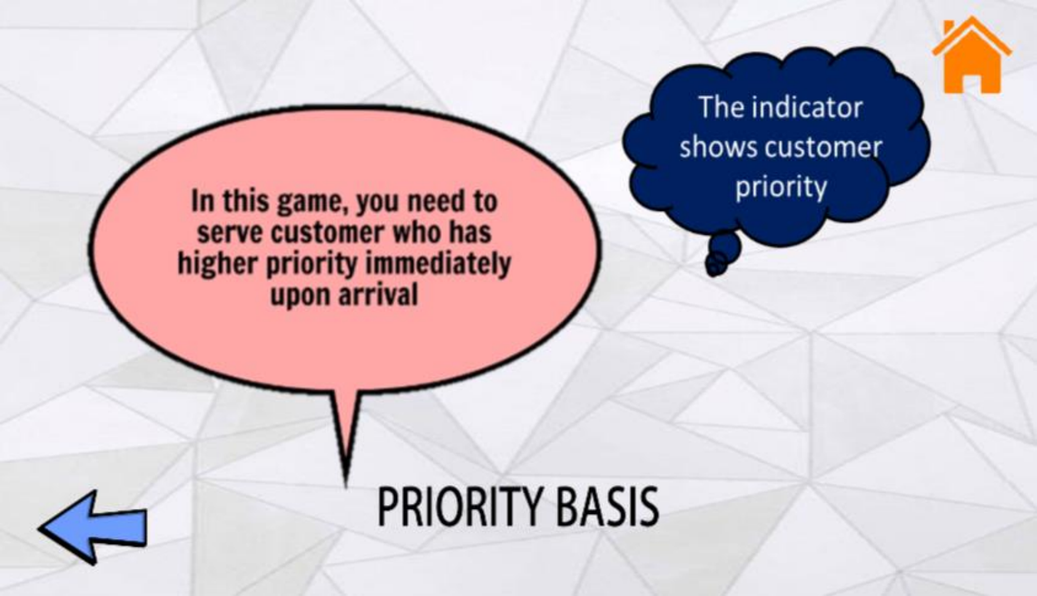


Figure 4.16 shows the play screen for level 2.



In this game, you need to serve customer who has higher priority immediately upon arrival

The indicator shows customer priority

PRIORITY BASIS

Figure 4.17 shows game rules in level 2 and (S2)

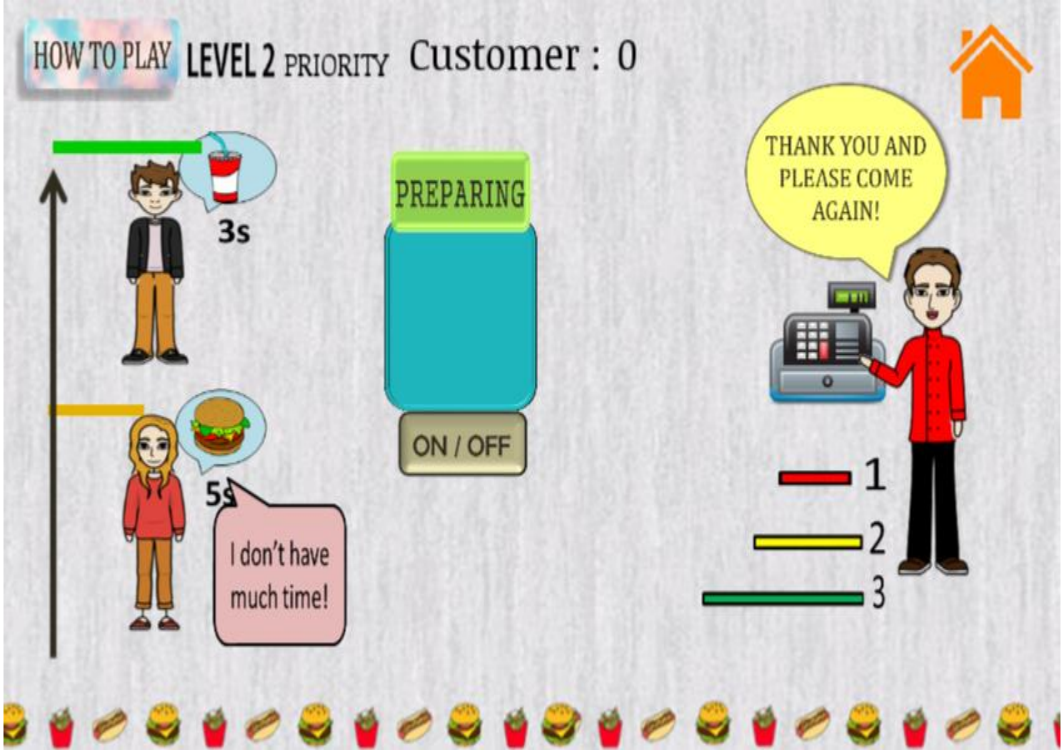


Figure 4.18 shows game hint in level 2 and (S2)

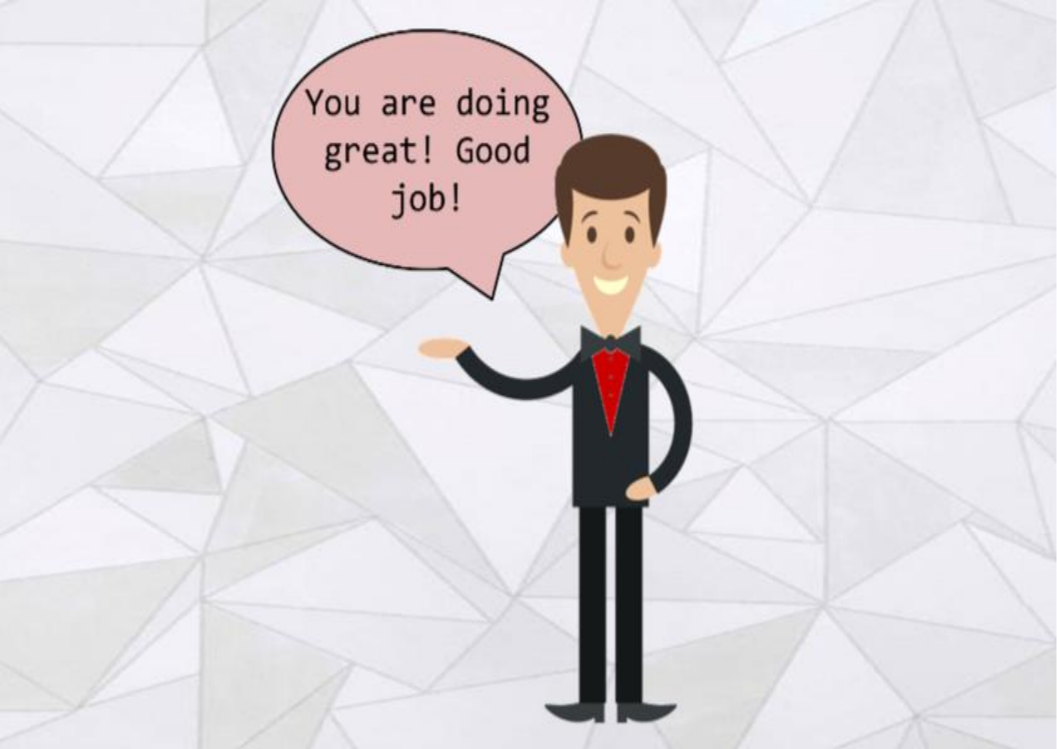


Figure 4.19 shows game feedback in level 2 and (S1)

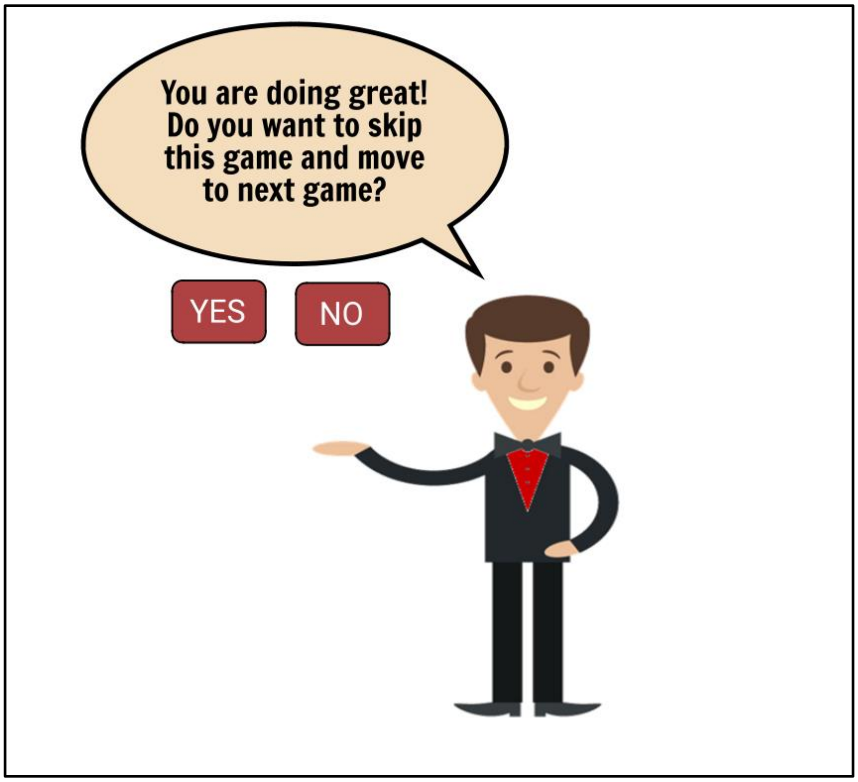

You are doing great! Do you want to skip this game and move to next game?
YES
NO

Figure 4.20 shows chance provided based on linear player’s performance and (S3)

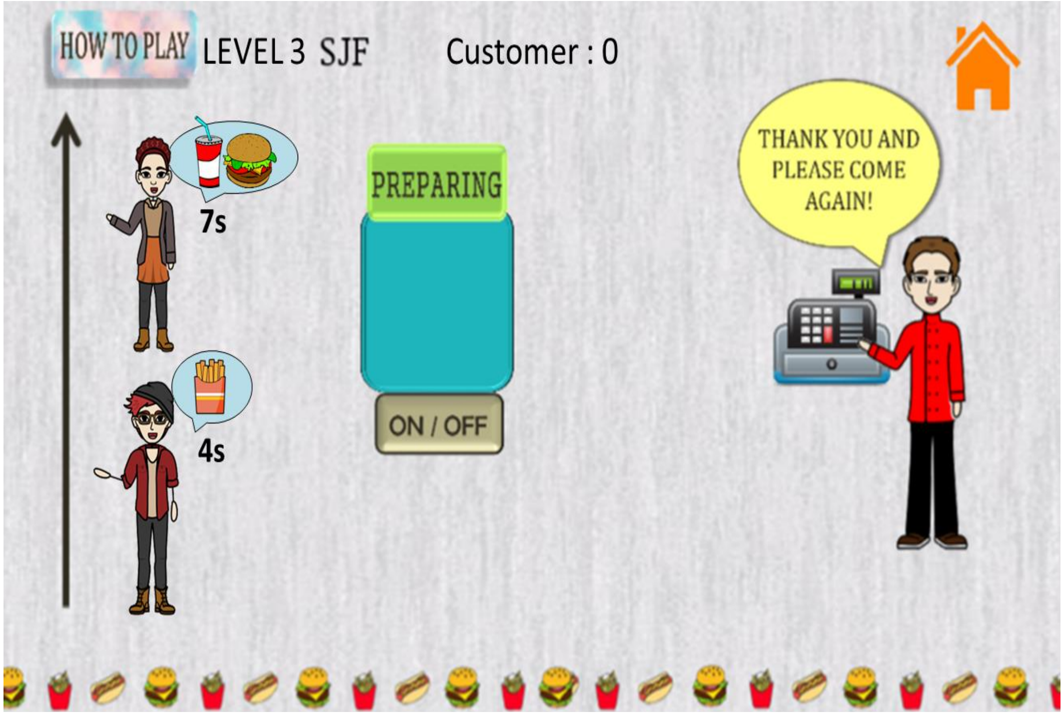
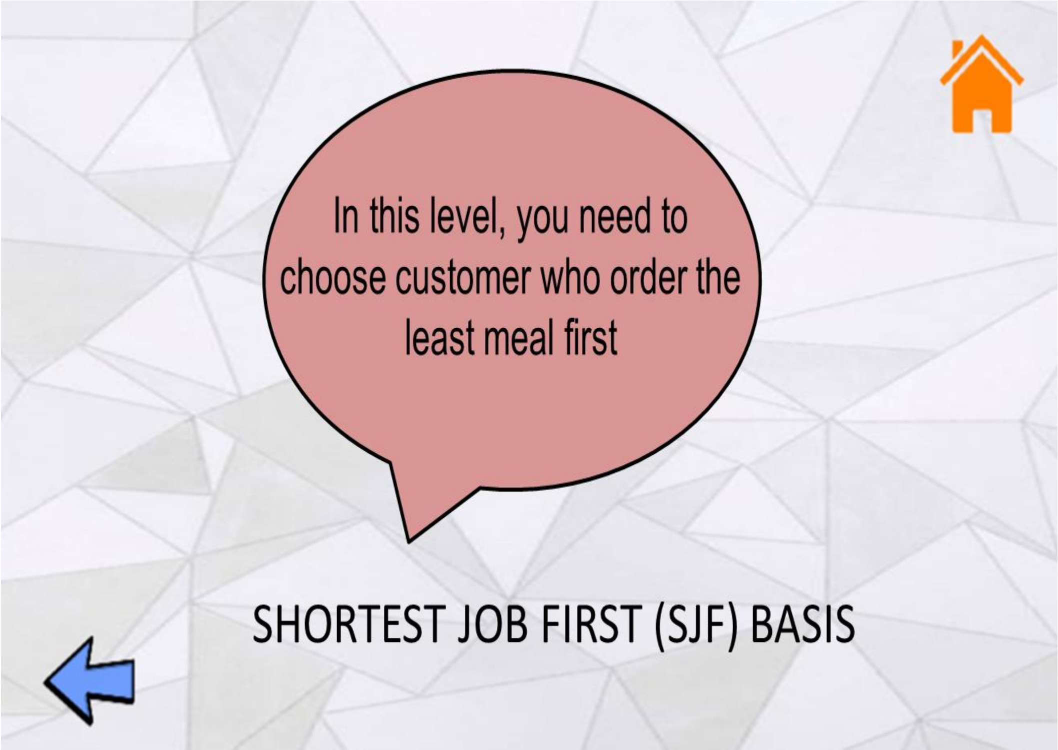


Figure 4.21 shows the play screen for level 3



In this level, you need to choose customer who order the least meal first

SHORTEST JOB FIRST (SJF) BASIS

Figure 4.22 shows game rule in level 3 and (S2)

Tutorial screen for level 1

Description:

Figure 4.22 until 4.24 show tutorial screen for level 1. This tutorial explains the steps on how to play the game. The instruction is given as shown in the figures above

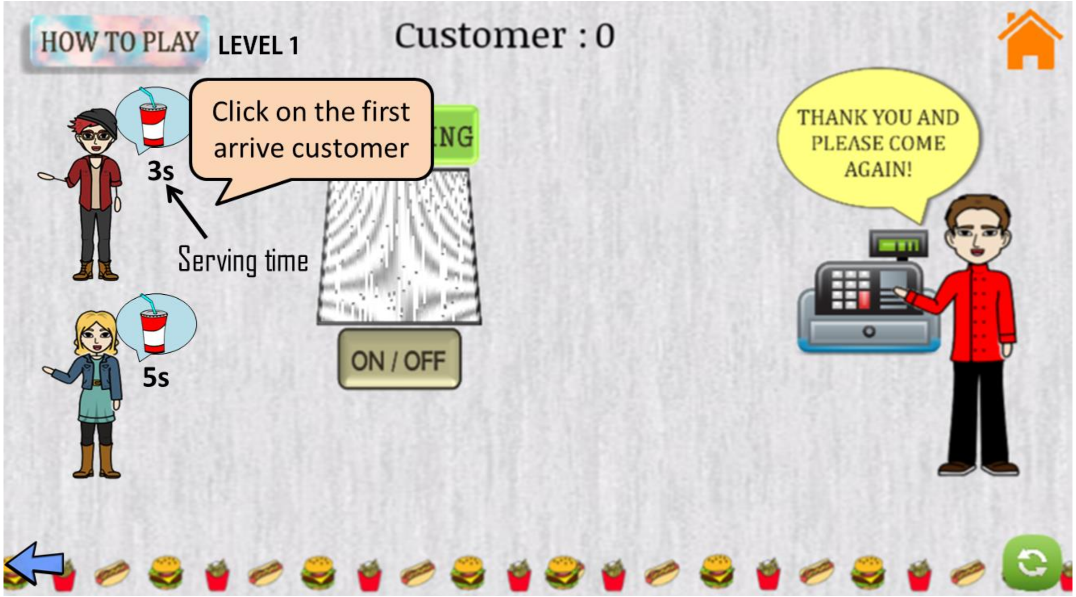

HOW TO PLAY
LEVEL 1
Customer : 0
Click on the first arrive customer
3s
Serving time
5s
ING
ON / OFF
THANK YOU AND PLEASE COME AGAIN!

Figure 4.23

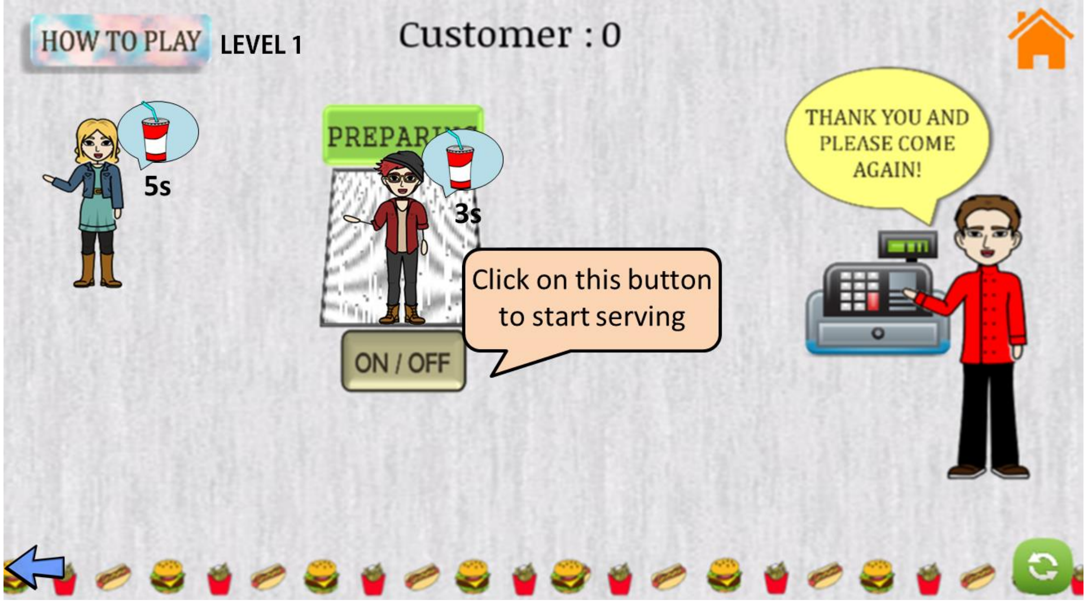

HOW TO PLAY
LEVEL 1
Customer : 0
5s
PREPARI
3s
Click on this button to start serving
ON / OFF
THANK YOU AND PLEASE COME AGAIN!

Figure 4.24

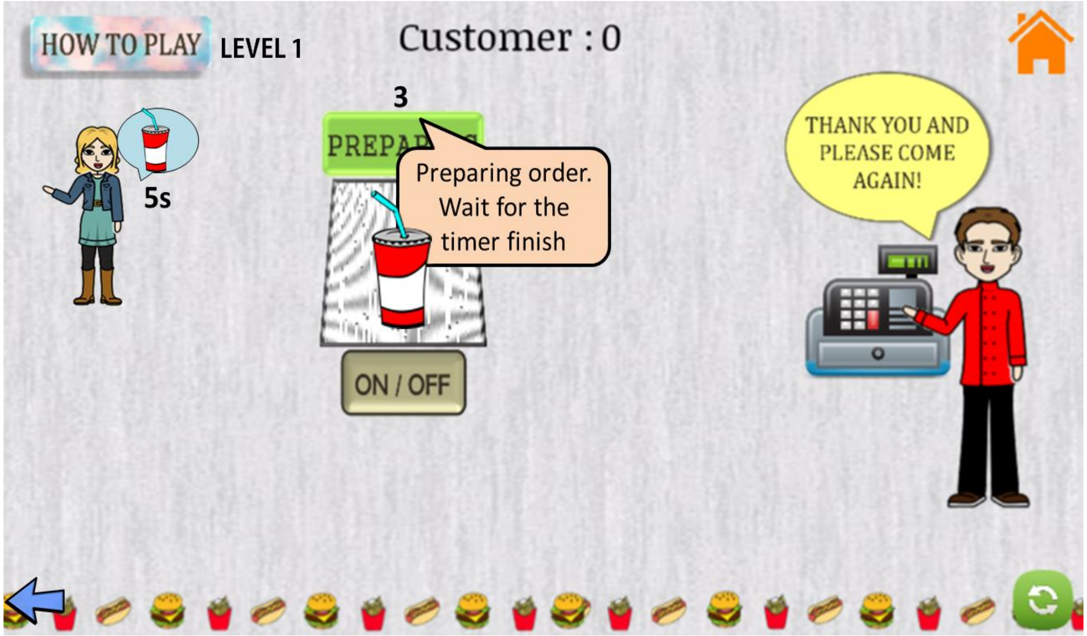

HOW TO PLAY
LEVEL 1
Customer : 0
3
PREPAR
5s
Preparing order. Wait for the timer finish
THANK YOU AND PLEASE COME AGAIN!
ON / OFF

Figure 4.25

Recap

Description:

Recap screen will show the analogy of the game played in the process of CPU scheduling. Figure 4.25 until Figure 4.29 below shows recap for Level 1 which describes the FCFS algorithm.

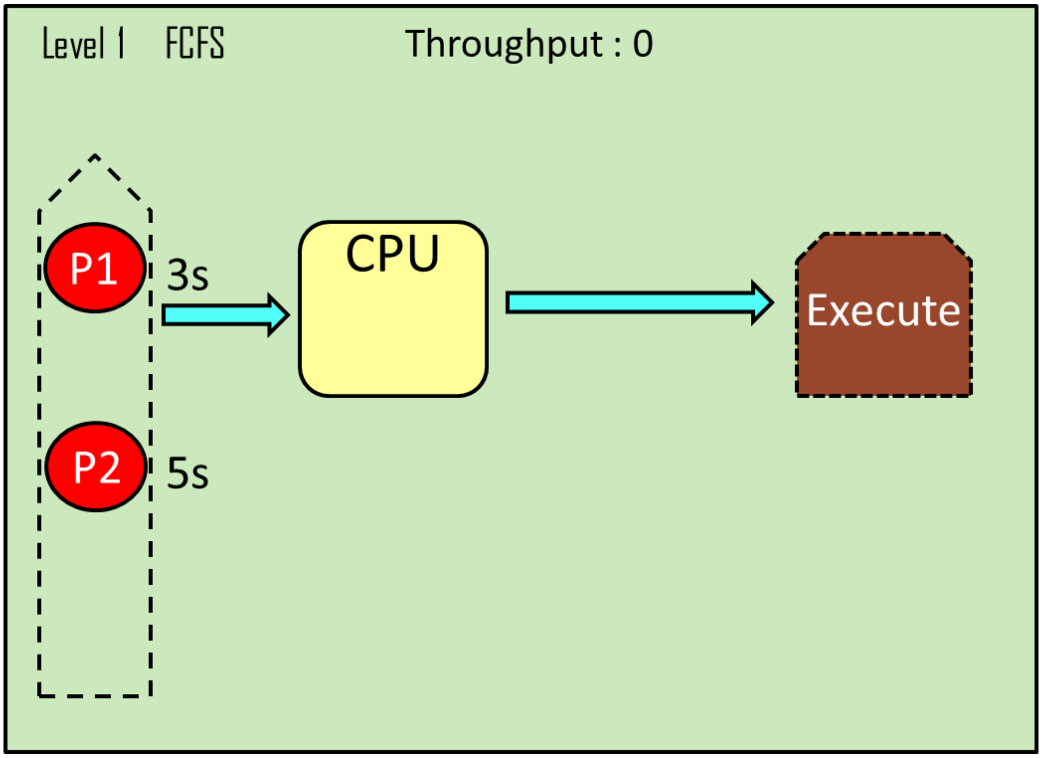


Figure 4.26

Process 1 (P1) and process 2 (P2) arrive in different arriving time and different burst time.

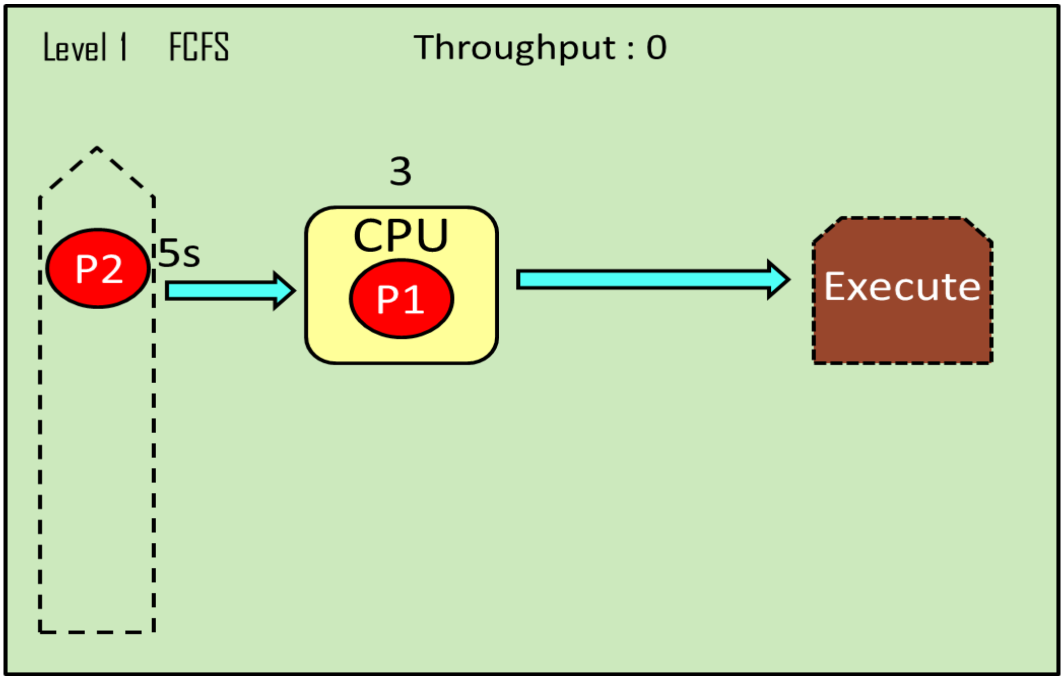


Figure 4.27

P1 is allocated to CPU first because it arrives before P2. CPU runs P1 based on the burst time

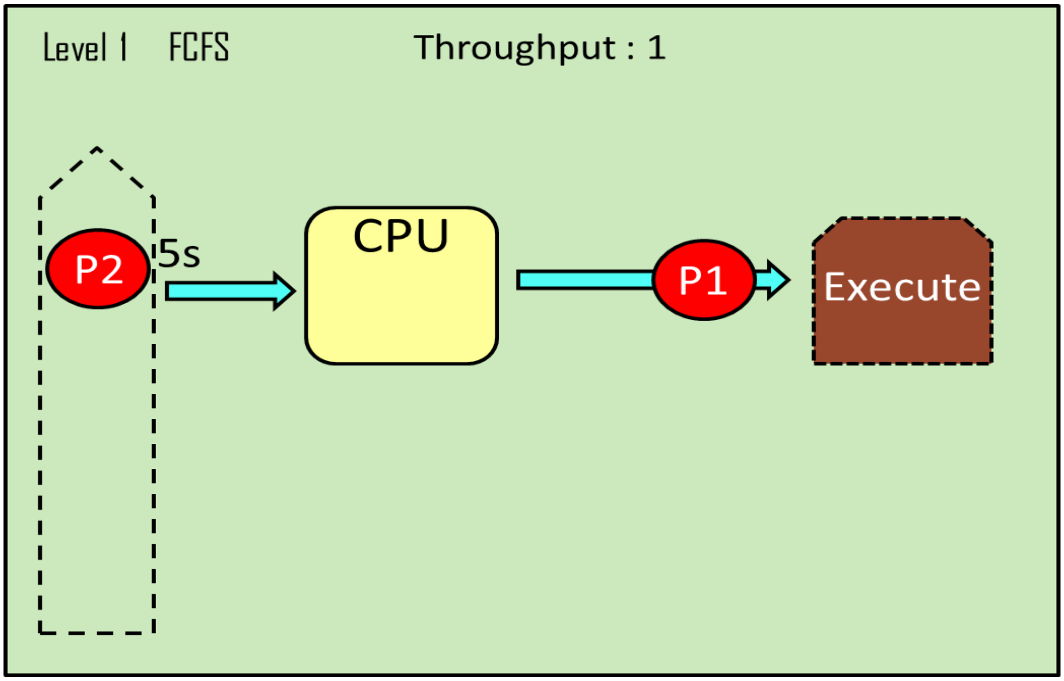


Figure 4.28

Once the running time finish, P1 is executed and throughput increases.

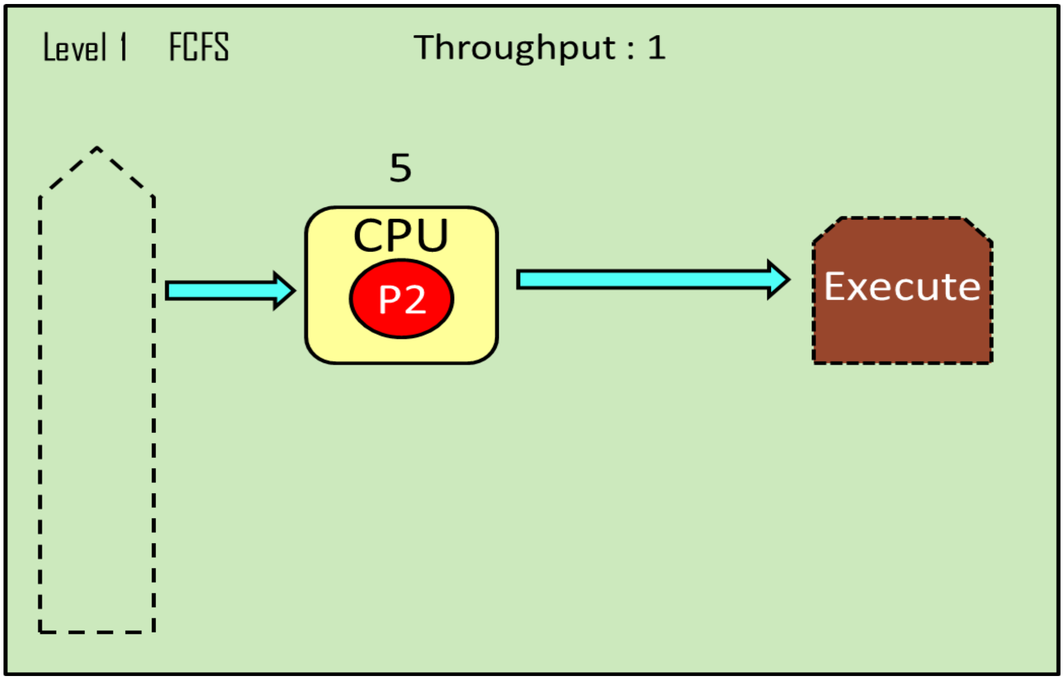


Figure 4.29

The next process (P2) is allocated to CPU to make sure CPU is fully utilized.

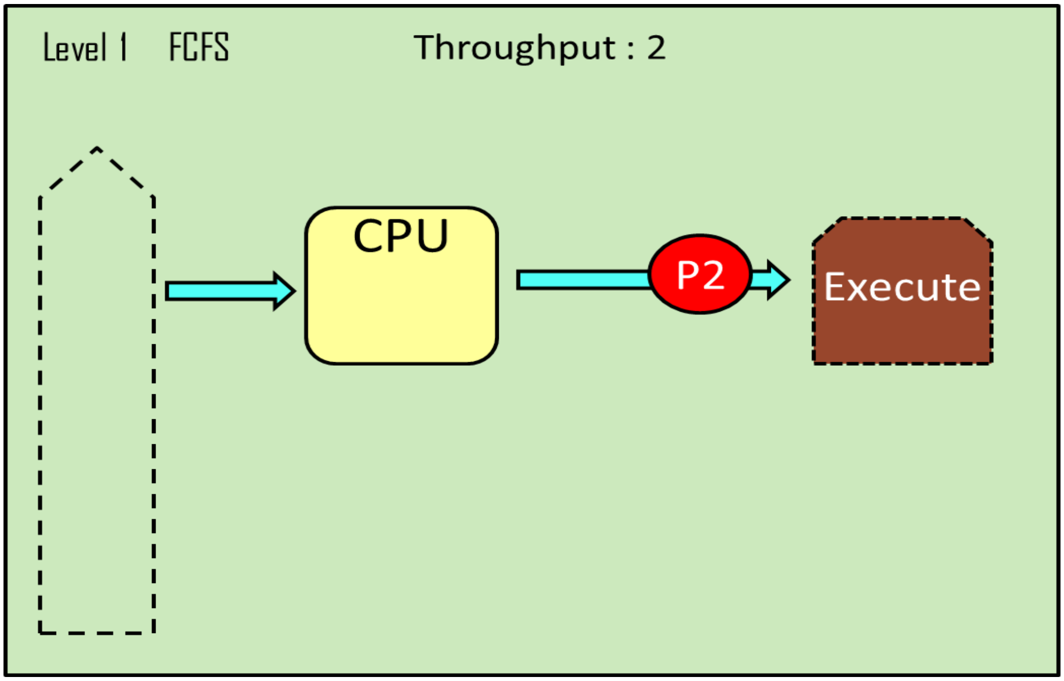


Figure 4.30

P2 continues running and execute when the running time is over. The throughput then increases.

4.3 Objective 3: To test the usability of the mobile educational game prototype.

The third objective of this project was achieved by testing the usability of the game prototype. The testing was evaluated by twenty CS students. The students tested the usability of the game and complete the evaluation of the prototype by doing survey questions. The survey will be shown at Appendix B (Survey of Scaffolded Mobile Educational Game for Learning CPU Scheduling).

The results from the survey evaluation are recorded as shown in Table 4.2 until Table 4.11

Table 4.2: Does this game helps you in learning CPU Scheduling?

Status	Total	Percentage
Strongly Agree (5)	16	80%
Agree (4)	2	10%
Neutral (3)	2	10%
Disagree (2)	0	0%
Strongly Disagree (1)	0	0%

Table 4.2 show the result of the first question in the survey carried out. From the table above, maximum percentage is 80% which is strongly agree with the question stated. Meanwhile, minimum percentage is 10% which is neutral with the question stated.

Table 4.3: Does the difficulty of this game increases too fast level by level?

Status	Total	Percentage
Strongly Agree (5)	0	0%
Agree (4)	0	0%
Neutral (3)	3	15%
Disagree (2)	4	20%
Strongly Disagree (1)	13	65%

Table 4.3 show the result of the second question in the survey carried out. From the table above, maximum percentage is 65% which is strongly disagree with the question stated. Meanwhile, minimum percentage is 15% which is neutral with the question stated.

Table 4.4: Do the hints provided in this game help you in completing the given task?

Status	Total	Percentage
Strongly Agree (5)	19	95%
Agree (4)	1	5%
Neutral (3)	0	0%
Disagree (2)	0	0%
Strongly Disagree (1)	0	0%

Table 4.4 show the result of the third question in the survey carried out. From the table above, maximum percentage is 95% which is strongly agree with the question stated. Meanwhile, minimum percentage is 5% which is agree with the question stated.

Table 4.5: Do the feedback in this game discourage you to continue playing?

Status	Total	Percentage
Strongly Agree (5)	0	0%
Agree (4)	0	0%
Neutral (3)	1	5%
Disagree (2)	2	10%
Strongly Disagree (1)	17	85%

Table 4.5 show the result of the fourth question in the survey carried out. From the table above, maximum percentage is 85% which is strongly disagree with the question stated. Meanwhile, minimum percentage is 5% which is neutral with the question stated.

Table 4.6: Is this game not challenging?

Status	Total	Percentage
Strongly Agree (5)	0	0%
Agree (4)	4	20%
Neutral (3)	3	15%
Disagree (2)	13	65%
Strongly Disagree (1)	0	0%

Table 4.6 show the result of the fifth question in the survey carried out. From the table above, maximum percentage is 65% which disagree with the question stated. Meanwhile, minimum percentage is 15% which is neutral with the question stated.

Table 4.7: In your opinion, is the game too restrictive in terms of the possible choices that you would like to choose?

Status	Total	Percentage
Strongly Agree (5)	0	0%
Agree (4)	10	50%
Neutral (3)	2	10%
Disagree (2)	8	40%
Strongly Disagree (1)	0	0%

Table 4.7 show the result of the sixth question in the survey carried out. From the table above, maximum percentage is 50% which agree with the question stated. Meanwhile, minimum percentage is 10% which is neutral with the question stated.

Table 4.8: Does the instructions provided in each level assists you in understanding the game?

Status	Total	Percentage
Strongly Agree (5)	17	85%
Agree (4)	2	10%
Neutral (3)	1	5%
Disagree (2)	0	0%
Strongly Disagree (1)	0	0%

Table 4.8 show the result of the seventh question in the survey carried out. From the table above, maximum percentage is 85% which is strongly agree with the question stated. Meanwhile, minimum percentage is 5% which is neutral with the question stated.

Table 4.9: Does this game makes learning CPU scheduling harder?

Status	Total	Percentage
Strongly Agree (5)	0	0%
Agree (4)	0	0%
Neutral (3)	0	0%
Disagree (2)	4	20%
Strongly Disagree (1)	16	80%

Table 4.9 show the result of the eighth question in the survey carried out. From the table above, maximum percentage is 80% which is strongly disagree with the question stated. Meanwhile, minimum percentage is 20% which is disagree with the question stated.

Table 4.10: Do you prefer to learn CPU scheduling using this game than textbook?

Status	Total	Percentage
Strongly Agree (5)	18	90%
Agree (4)	2	10%
Neutral (3)	0	0%
Disagree (2)	0	0%
Strongly Disagree (1)	0	0%

Table 4.10 show the result of the ninth question in the survey carried out. From the table above, maximum percentage is 90% which is strongly agree with the question stated. Meanwhile, minimum percentage is 10% which is agree with the question stated.

Table 4.11: Do you think this game helps you get a better understanding and application of CPU scheduling?

Status	Total	Percentage
Strongly Agree (5)	18	90%
Agree (4)	2	10%
Neutral (3)	0	0%
Disagree (2)	0	0%
Strongly Disagree (1)	0	0%

Table 4.11 show the result of the tenth question in the survey carried out. From the table above, maximum percentage is 90% which is strongly agree with the question stated. Meanwhile, minimum percentage is 10% which is agree with the question stated.

4.4 Summary

This chapter explains and accomplishes the targeted objectives of this project which are

1. To determine scaffolding characteristics and game elements for mobile educational game design.

   Suitable game elements and scaffolding characteristics were identified and storyboard was used in the prototype design to make the process of designing smooth. The prototype's interface were organized and visualized smoothly.

2. To develop a scaffolded mobile educational game for learning CPU scheduling.

   A scaffolded mobile educational game for learning CPU scheduling was developed using GameSalad Creator. The game was created based on an analogy of customer service in a café. This analogy was used to show learner an easy concept of learning CPU scheduling algorithm.

3. To evaluate the usability of the scaffolded mobile educational game.
   This game was tested on Android platform and a total of 20 Computer Sciences students evaluate the game prototype. The results from the survey shows that most of the students found this scaffolded mobile educational game for learning CPU scheduling is interesting and engaging.

CHAPTER 5

CONCLUSION

5.0 Introduction

This chapter summarized what have been done to achieve the objectives of this project. This project had gone through several activities which are theoretical study, scaffolding identification, prototype design, prototype development, testing and documentation.

5.1 Conclusion

Theoretical study had been carried out based on difficulties faced by CS and IT students in learning operating system course. Those theoretical studies found that student having a problem in learning using static representations such as textbook and lecture notes. Analysis is also done to gain more understanding of scaffolding in mobile educational game development.

First objective of this project was achieved by identifying scaffolding characteristics and game elements for mobile educational game design. Several game elements were identified and implemented in this project such as level, feedback, challenge, rules and linearity. These five game elements fulfill the characteristics of scaffolding discussed which are (S1) Scaffold learner’s effort, (S2) Instruction at appropriate level and (S3) Gradual transfer of responsibility.

The most important activities carried out in this project are design and development of the prototype. In designing the prototype, storyboard is used to visualize and organize the content of this project. The development of the prototype is implemented using selected tool which is GameSalad. These two activities were able to achieve the second objective which is to develop a scaffolded mobile educational game for learning CPU scheduling.

The third objective of this project is to evaluate the usability of the scaffolded mobile educational game. To accomplish the evaluation of this project, prototype testing is carried out from twenty CS and IT students to test the usability and effectiveness of the game prototype. The game prototype was tested on Android platform. Objective three is attained as the testing is put through as well as the results from the survey evaluation.

REFERENCES

Bhowmick, S. (2014, September 7). The Importance Of An Operating System. Retrieved from https://pwntoken.wordpress.com/2014/09/07/2-importance-of-the-operating-system/

Choong, M.W. & Lo, H.H. (2012). Mobile Educational Game. *2012 4th International Conference on Itelligent and Advanced Systems*, 811-813

Coffey (n.d). Scaffolding. Retrieved from http://www.learnnc.org/lp/pages/5074

David W Denton (2014, January 27). Characteristics Of Scaffolding and Activities for Using It in Classroom. Retrieved from http://www.davidwdenton.org/characteristics-of-scaffolding-and-activities-for-using-it-in-classrooms/

EDPY 301 (2017) Scaffolding Quizlet. Retrieved from https://quizlet.com/15695424/edpy-301-scaffolding-flash-cards/

Education Reform. (2015). *Scaffolding*. (2015). Retrieved June 4, 2015, from http://edglossary.org/scaffolding/

Galvin, P. B., Gagne, G., & Silberschatz, A. (2013). *Operating system concepts*. John Wiley & Sons, Inc..Jantan, S. R., & Aljunid, S. A. (2012, October). An experimental evaluation of scaffolded educational games design for programming. In *Open Systems (ICOS), 2012 IEEE Conference on* (pp. 1-6). IEEE.

GCF LearnFree.org. (n.d). *Understanding Operating System.* Retrieved from http://www.gcflearnfree.org/computerbasics/understanding-operating-systems/1/

Grace, L. (2005). Game type and game genre. *Retrieved February*, *22*, 2009.

Klopfer, E. (2008). *Augmented learning: Research and design of mobile educational games*. Mit Press.

Larkin, M. (2002). Using Scaffolded Instruction To Optimize Learning. ERIC Digest.

Lavín-Mera, P., Torrente, J., Moreno-Ger, P., Pinto, J. A. V., & Fernández-Manjón, B. (2009). Mobile Game Development for Multiple Devices in Education. *iJET*, *4*(2), 19-26.

Meyer, D. K., & Turner, J. C. (2002). Using instructional discourse analysis to study the scaffolding of student self-regulation. *Educational psychologist*,*37*(1), 17-25.

Navarro, E. O., Baker, A., & Van Der Hoek, A. (2004, January). Teaching software engineering using simulation games. In *ICSIE'04: Proceedings of the 2004 International Conference on Simulation in Education*.

Pinantoan, A. (2013, March 20). Definition of Scaffolding. Retrieved from http://www.opencolleges.edu.au/informed/teacher-resources/scaffolding-in-education-a-definitive-guide/

Puntambekar, S., & Hubscher, R. (2005). Tools for scaffolding students in a complex learning environment: What have we gained and what have we missed?. *Educational psychologist*, *40*(1), 1-12.

Regehr, J. (2010, June 8). Why take an operating system course. Retrieved from http://blog.regehr.org/archives/164

Suranauwarat, S. (2013). A Visual and Interactive Learning Tool for CPU Scheduling Algorithms. *Int J Comput Sci*, *10*, 509-518.

Tlili, A., Essalmi, F., & Jemni, M. (2015, December). An educational game for teaching computer architecture: Evaluation using learning analytics. In *2015 5th International Conference on Information & Communication Technology and Accessibility (ICTA)* (pp. 1-6). IEEE.

Tlili, A., Essalmi, F., & Jemni, M. (2015, July). A mobile educational game for teaching computer architecture. In *2015 IEEE 15th International Conference on Advanced Learning Technologies* (pp. 161-163). IEEE.

Torrente, J., Del Blanco, Á., Marchiori, E. J., Moreno-Ger, P., & Fernández-Manjón, B. (2010, April). < e-Adventure>: Introducing educational games in the learning process. In *IEEE EDUCON 2010 Conference* (pp. 1121-1126). IEEE.

Valk, J. H., Rashid, A. T., & Elder, L. (2010). Using mobile phones to improve educational outcomes: An analysis of evidence from Asia. *The International Review of Research in Open and Distributed Learning*, *11*(1), 117-140.

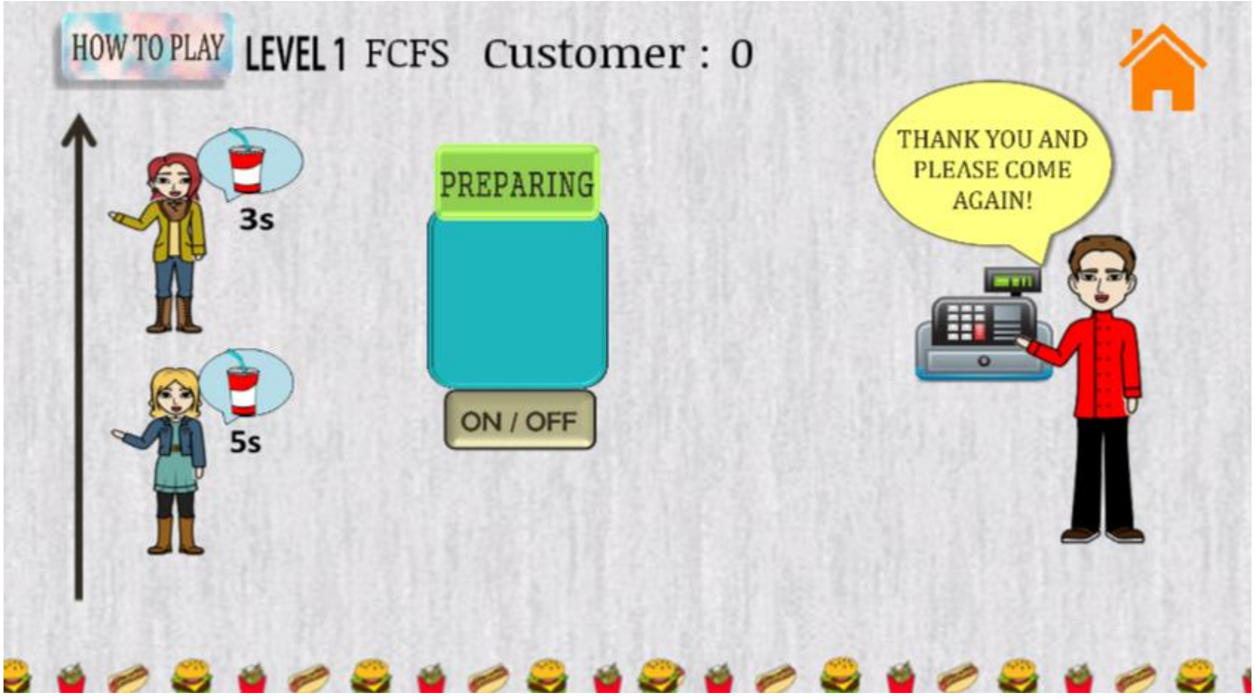
Van de Pol, J., Volman, M., & Beishuizen, J. (2010). Scaffolding in teacher–student interaction: A decade of research. *Educational Psychology Review*,*22*(3), 271-296.

Walqui, A. (2006). Scaffolding instruction for English language learners: A conceptual framework. *International Journal of Bilingual Education and Bilingualism*, *9*(2), 159-180.

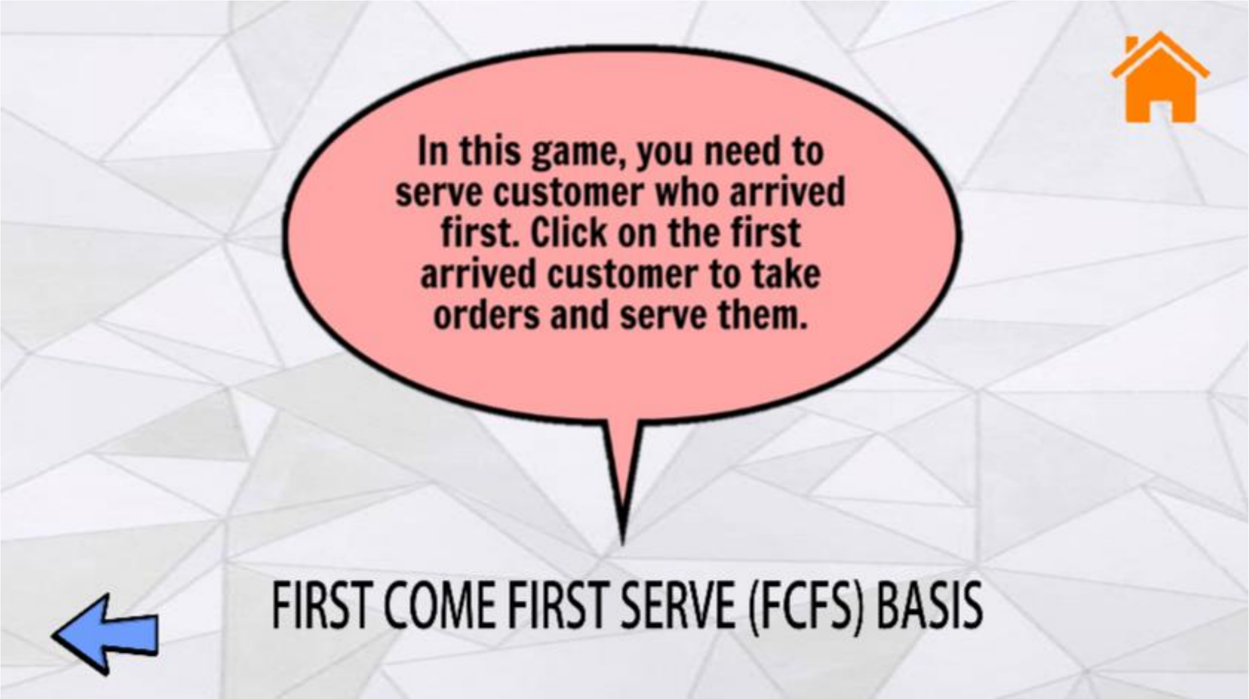
Wijethunga, A. G. (2010, December 6). CPU Scheduling. Retrieved from http://www.ayomaonline.com/academic/cpu-scheduling/

APPENDIX A
(USER MANUAL)

Play Level 1

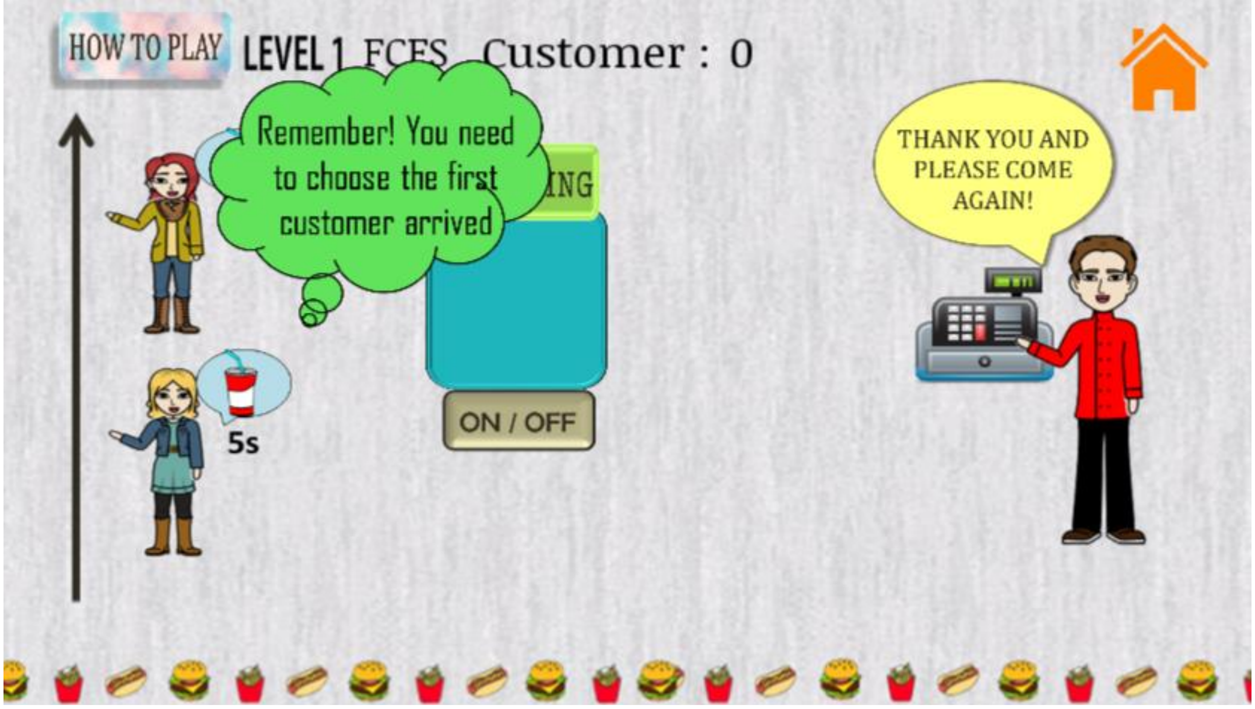


a. Customers arrive and queue.

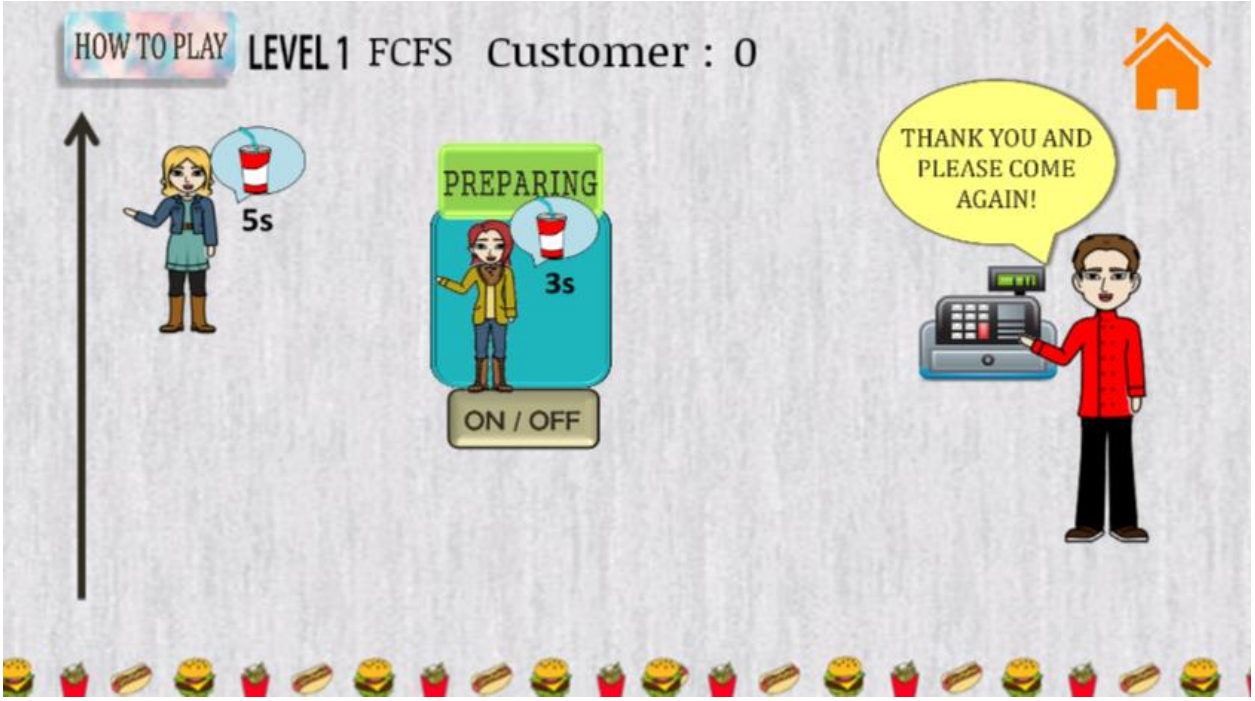

In this game, you need to serve customer who arrived first. Click on the first arrived customer to take orders and serve them.

FIRST COME FIRST SERVE (FCFS) BASIS

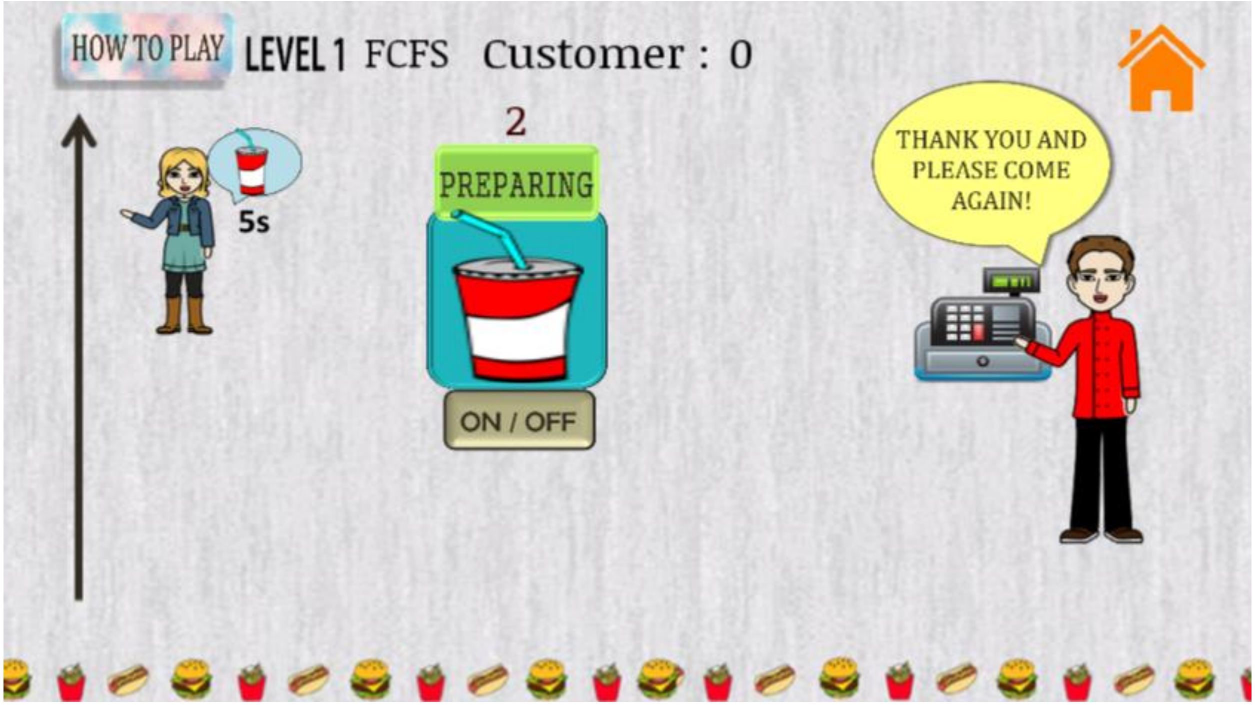
b. Click on How To Play button and the game rules (instruction) is shown.


HOW TO PLAY
LEVEL 1 FCFS Customer : 0
Remember! You need to choose the first customer arrived
ING
5s
ON / OFF
THANK YOU AND PLEASE COME AGAIN!

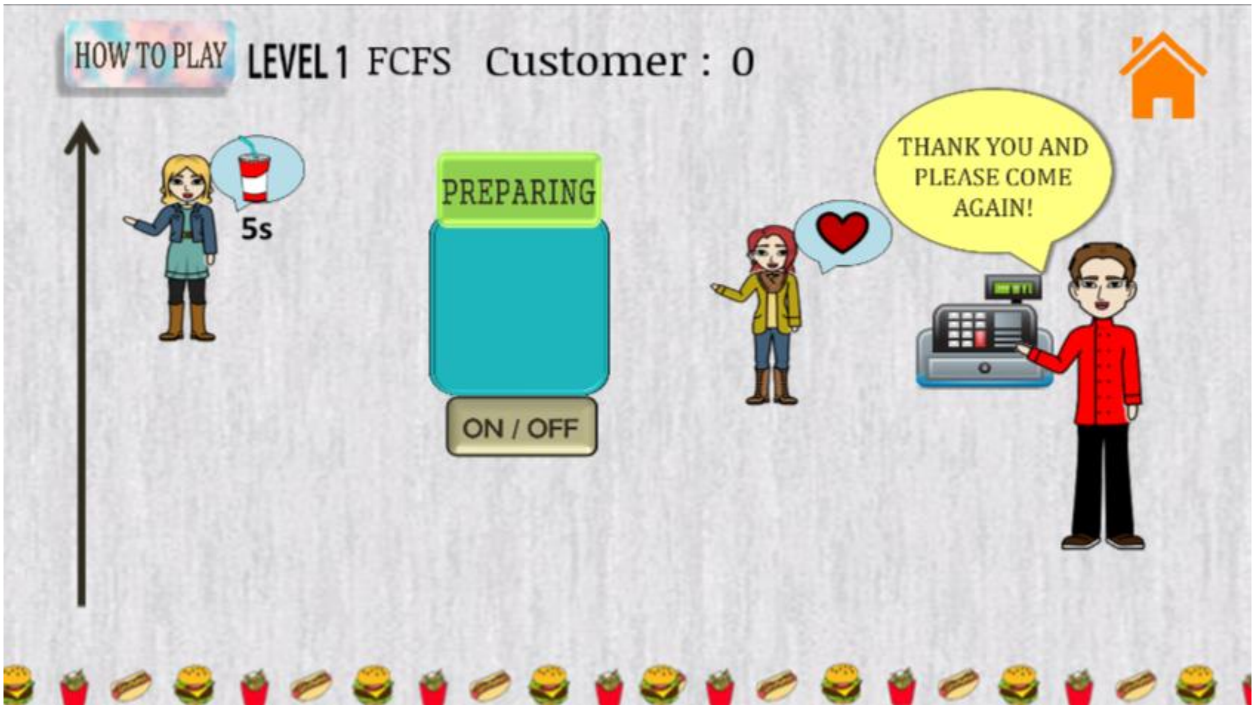
c. Hint is popped up if player choose wrong customer



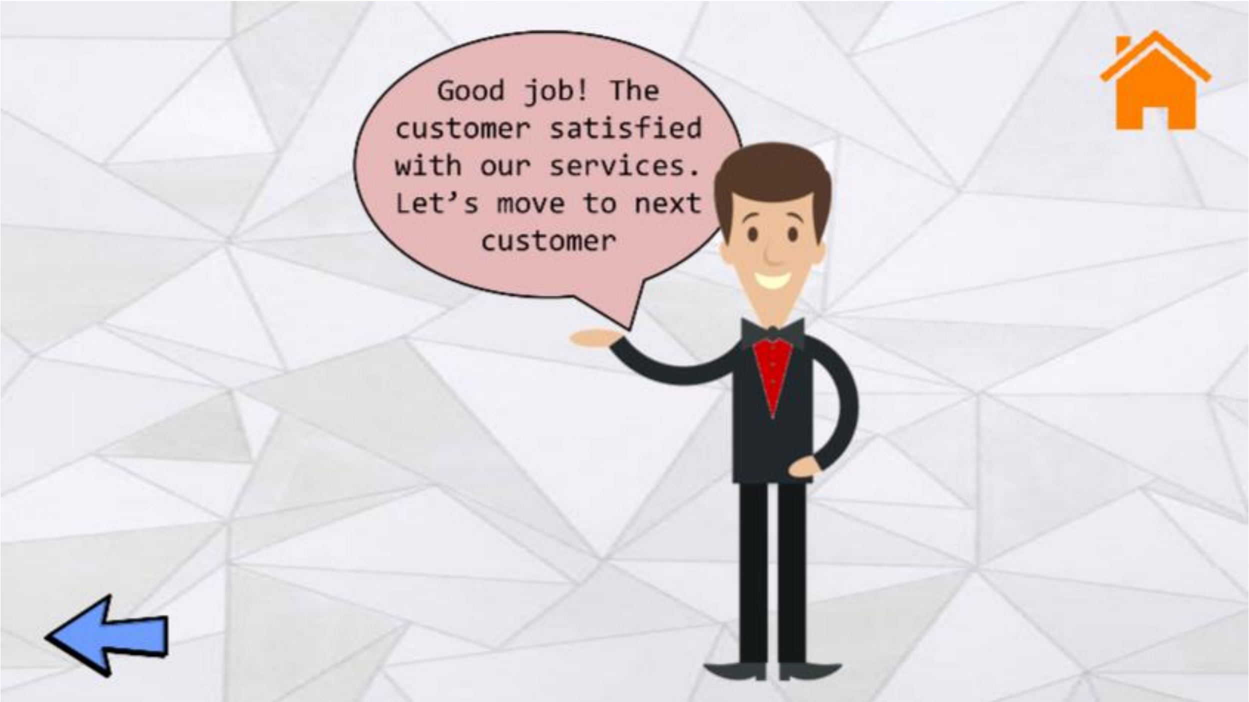
d. Player needs to select the right customer based on rules given



e. Once customer is selected, player needs to start preparing the order while the serving timer will start counting.

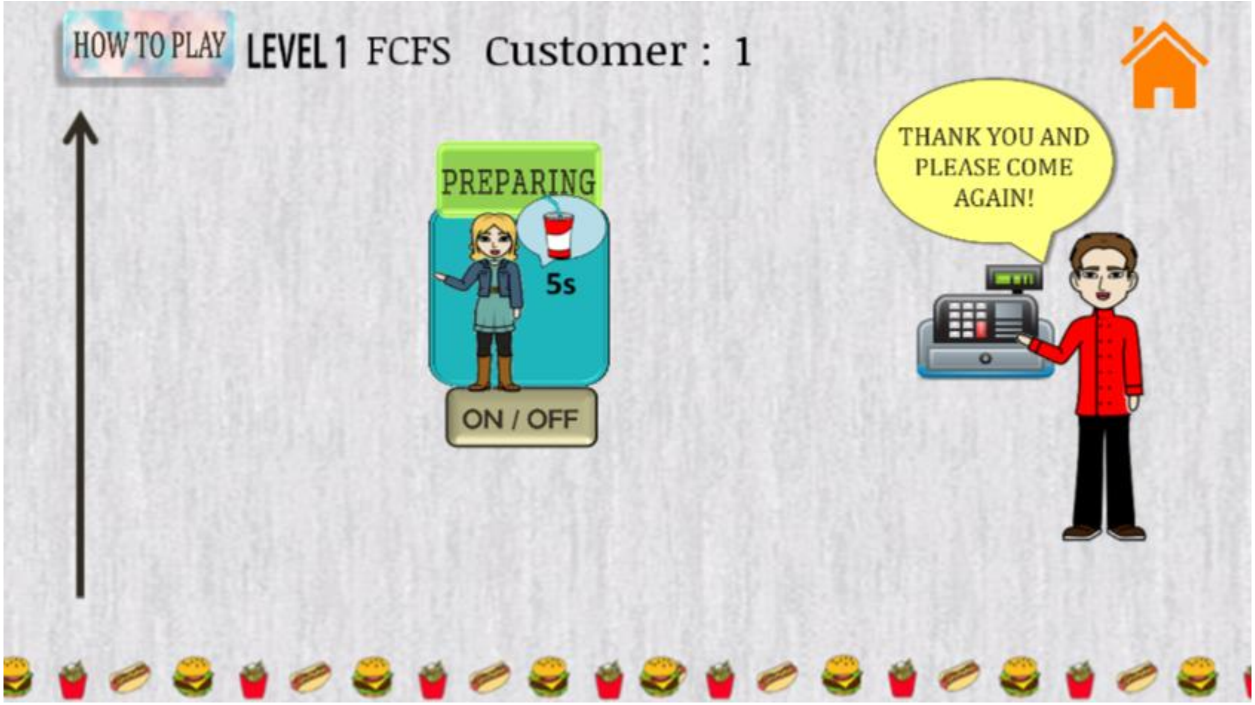


f. When serving timer finish, customer will automatically move to pay at the cashier.

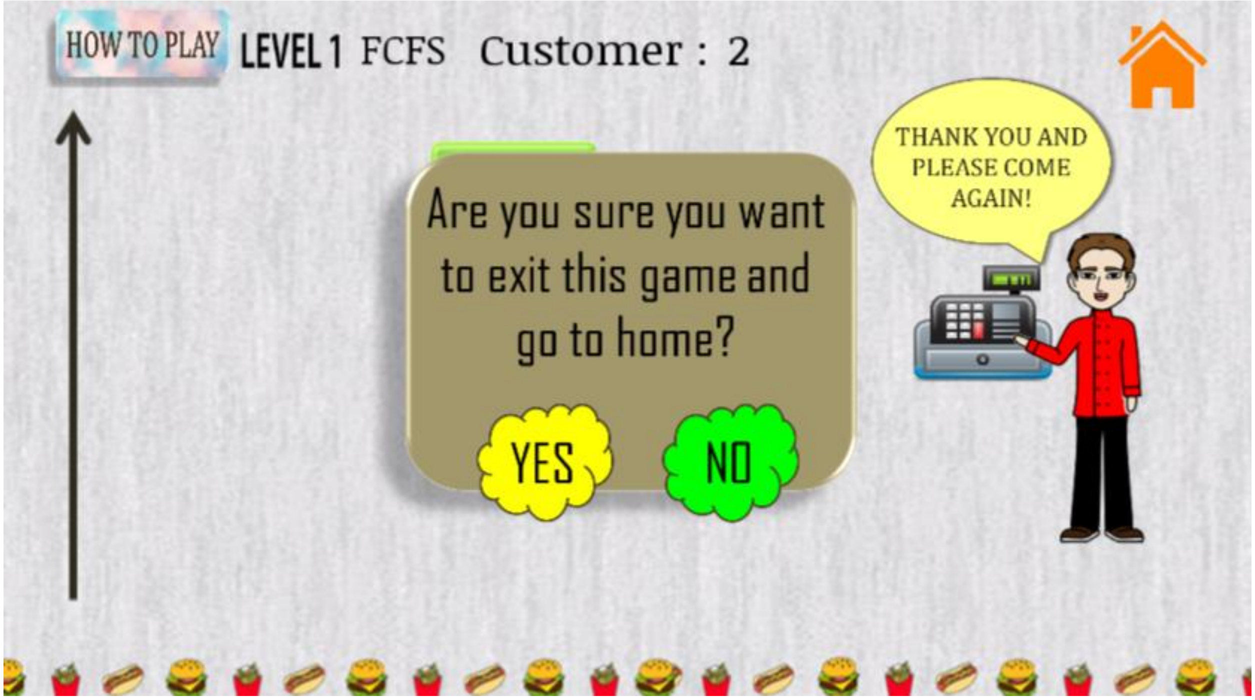


Good job! The customer satisfied with our services. Let’s move to next customer

g. Once cashier get paid, number of customer (act as score) increases by one. Feedback is given to encourage player continue serving the next customer available.



h. Continue serving with the next customer



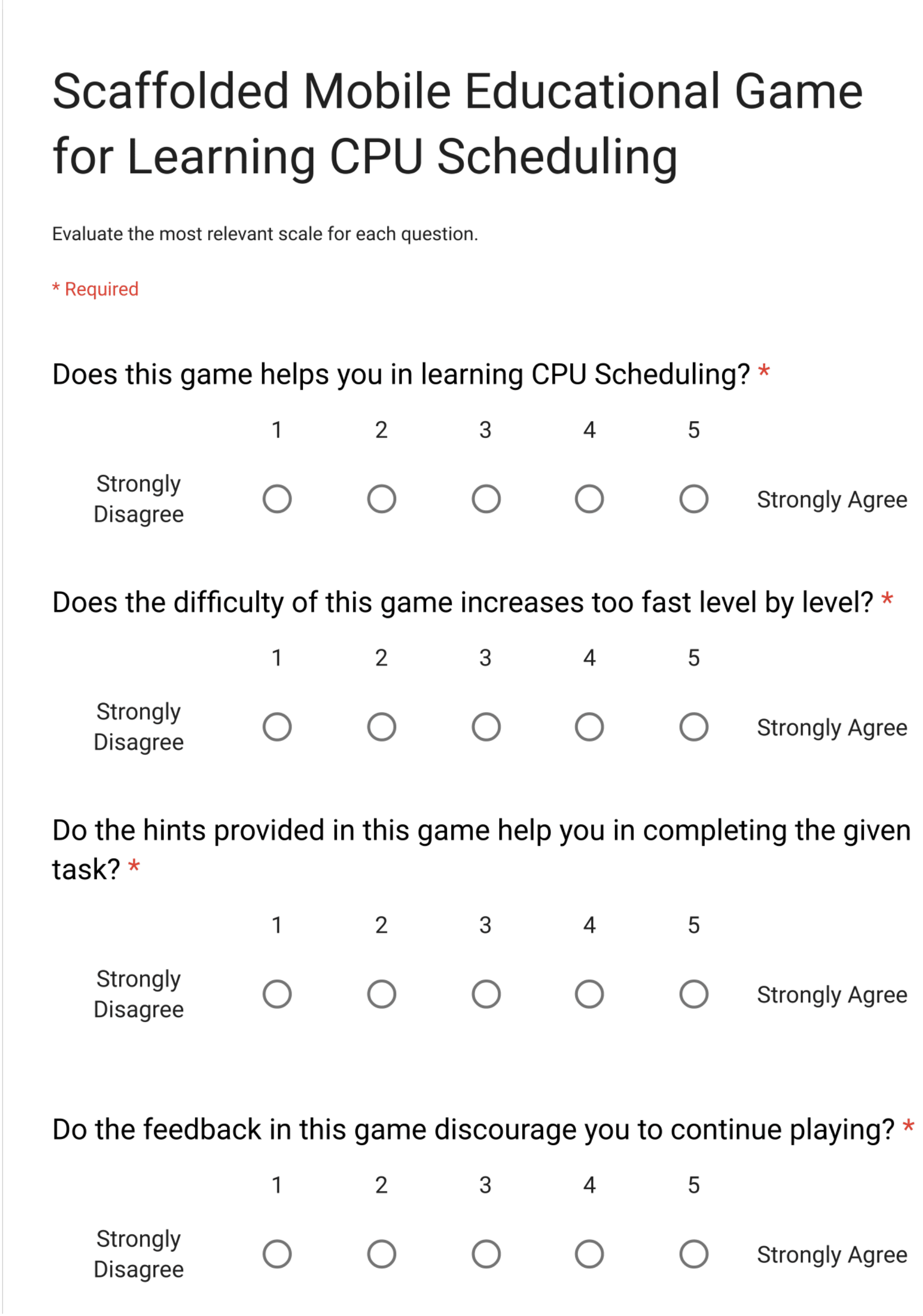
Are you sure you want to exit this game and go to home?

i. Click home button to exit game.

APPENDIX B

(SCAFFOLDED MOBILE EDUCATIONAL GAME FOR LEARNING CPU SCHEDULING SURVEY)

SURVEY QUESTIONS



Scaffolded Mobile Educational Game for Learning CPU Scheduling

Evaluate the most relevant scale for each question.

* Required

Does this game helps you in learning CPU Scheduling? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Does the difficulty of this game increases too fast level by level? *

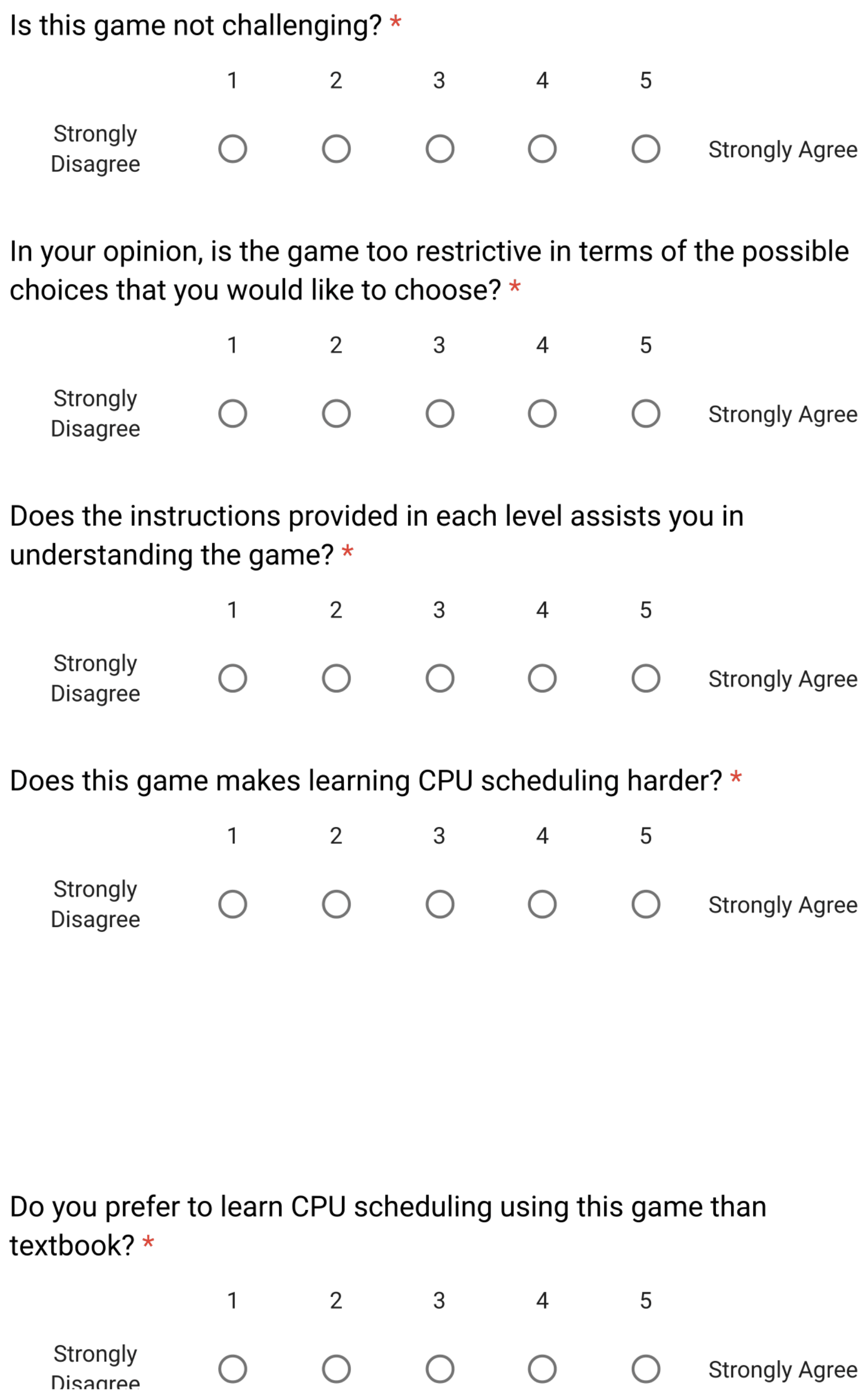
	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Do the hints provided in this game help you in completing the given task? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Do the feedback in this game discourage you to continue playing? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree



Is this game not challenging? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

In your opinion, is the game too restrictive in terms of the possible choices that you would like to choose? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Does the instructions provided in each level assists you in understanding the game? *

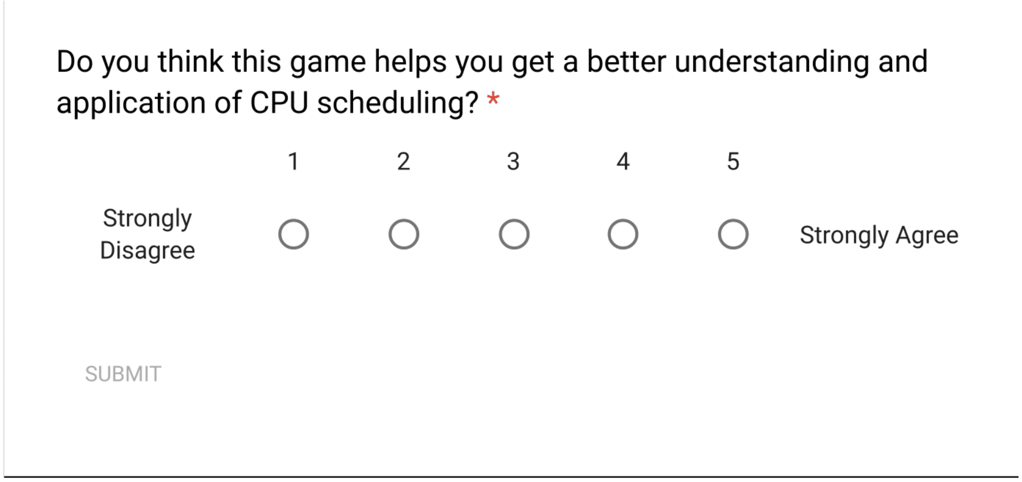
	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Does this game makes learning CPU scheduling harder? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Do you prefer to learn CPU scheduling using this game than textbook? *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

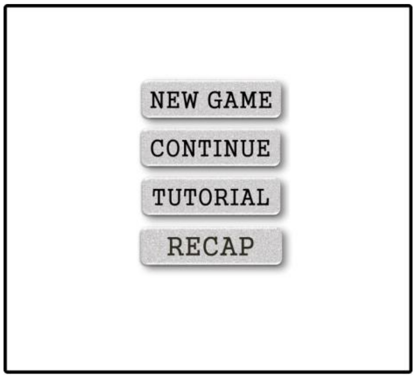


Do you think this game helps you get a better understanding and application of CPU scheduling? *

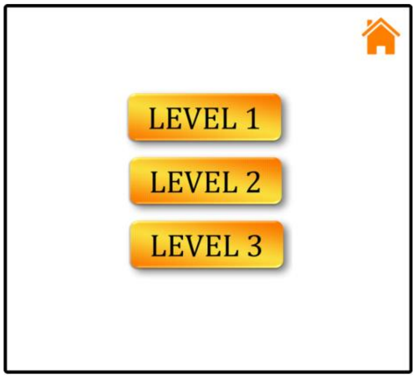
	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

APPENDIX C

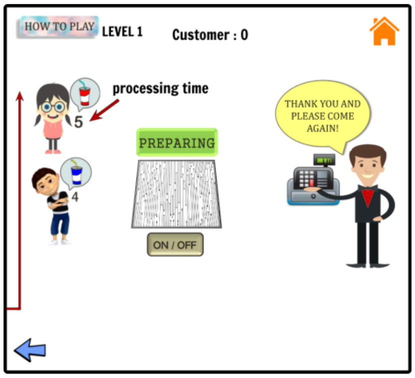
(STORYBOARD)



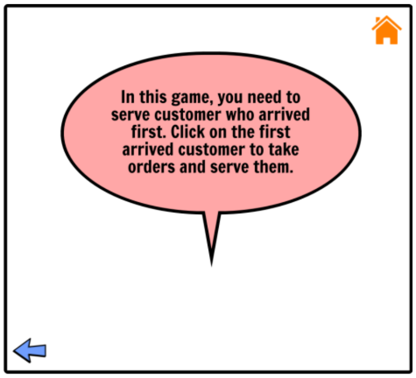
Home



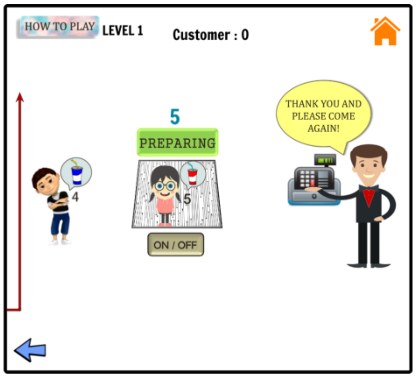
Levels



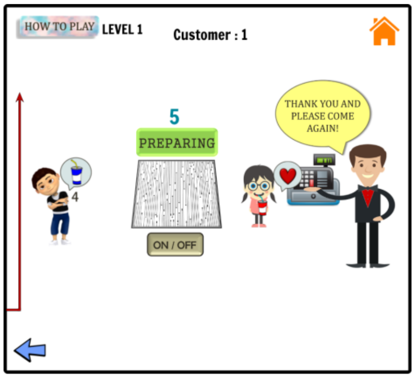
Level 1 scene. Customers arrive and the player need to choose which customer need to be served first following the game rules. The game rules will be shown if button 'How To Play' is clicked.


In this game, you need to serve customer who arrived first. Click on the first arrived customer to take orders and serve them.

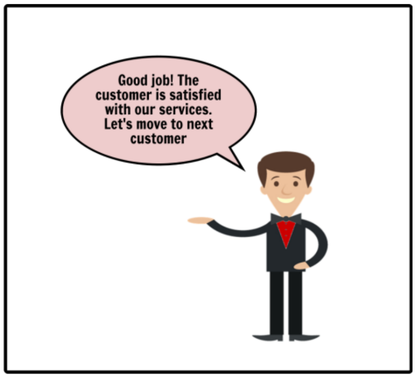
In this How To Play scene, player will be given instruction on how to play the game. Each game have different rules. As for level 1 game 1, FCFS algorithm rule is used.



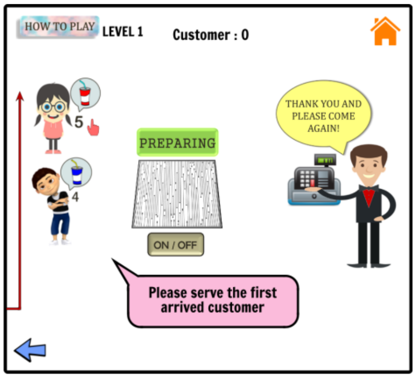
When chosen customer is clicked, she will be served and machine(CPU) will prepare her order. Timer for processing time is displayed.



Once the order is finished, the customer will pay at the counter. The number of customer served of the day increases as the order is paid.


Good job! The customer is satisfied with our services. Let's move to next customer

To encourage the player

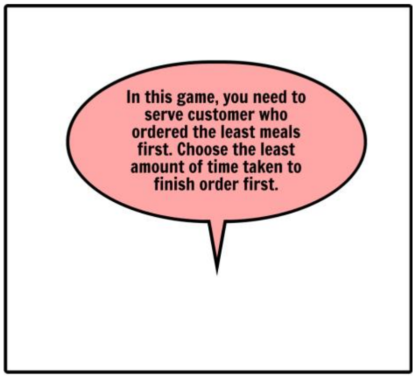


Please serve the first arrived customer

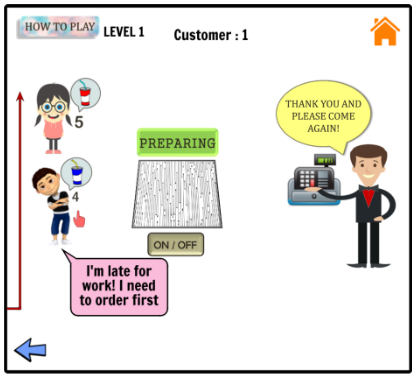
If player choose wrong customer, a pop up box appears to instruct player to choose the correct customer based on the rule given.

You have completed First Come First Serve game. Let's move to the next game. I'm sure it will be more exciting

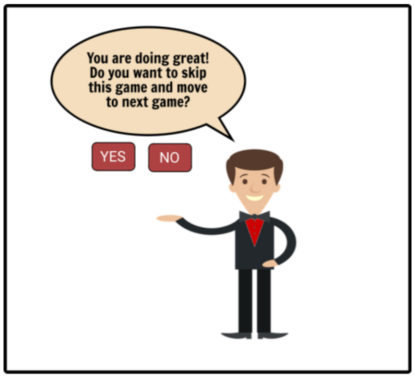
As player success in serving customers for game 1, move to the next game 2 which difficulty and rules applied are more challenging.


In this game, you need to serve customer who ordered the least meals first. Choose the least amount of time taken to finish order first.

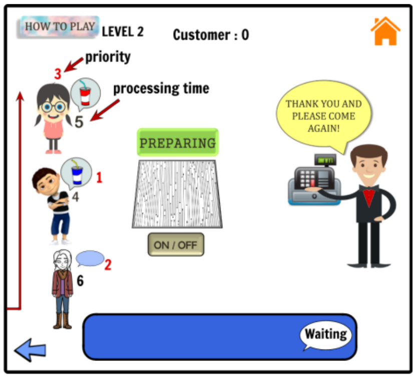
Level 1 game 2 apply Shortest Job First (SJF) algorithm


HOW TO PLAY
LEVEL 1
Customer : 1
5
4
PREPARING
ON / OFF
THANK YOU AND PLEASE COME AGAIN!
I'm late for work! I need to order first

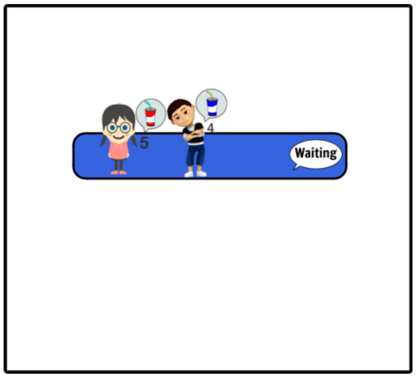
If player seems to have problem in choosing the right customer, hint is given (pop up dialogue box)


You are doing great! Do you want to skip this game and move to next game?
YES
NO

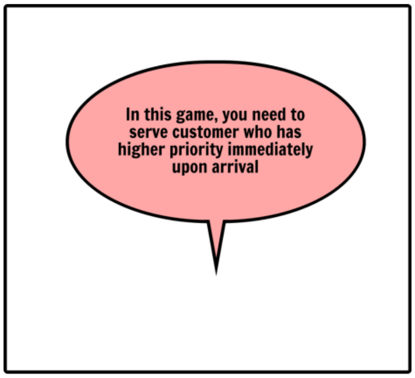
If player seems to understand playing the game correctly without much hint given, an option is provided whether to continue finish current game or skip and move to next game.



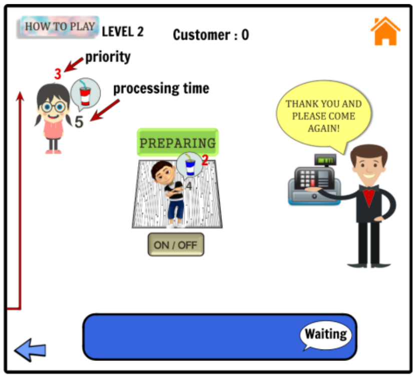
Level 2 scene game 1. Customers arrived with priority. Waiting queue block is added in this game.



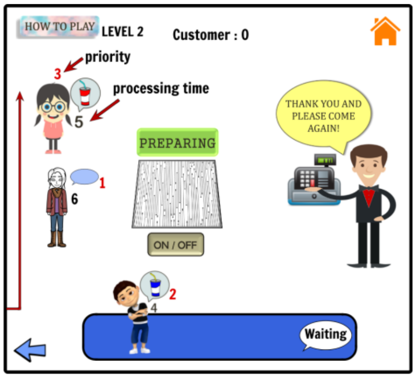
Waiting queue block is where customer's order is halted by the other order placed. Order in this queue will continue preparing after certain order is finished.


In this game, you need to serve customer who has higher priority immediately upon arrival

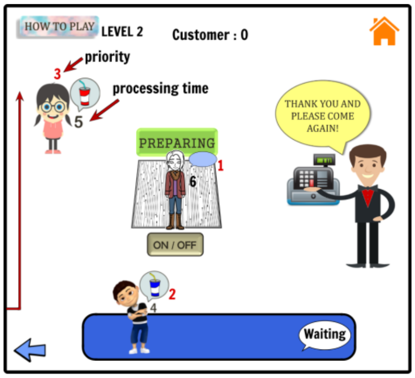
Level 2 game 1 apply Preemptive Priority algorithm



Customers arrive with priorities and player choose the highest priority to prepare order first



If new customer arrives with higher priority than existing customers, existing customer's order will be halted and placed in waiting queue block



New priority customer's order will take place. Once her order finished, previous customer will continue to place his order and leave the waiting queue

APPENDIX D

(PROJECT IN-PROGRESS FORM)

UNIVERSITI TEKNOLOGI MARA

CSP600 and CSP650

Faculty of Computer & Mathematical Sciences

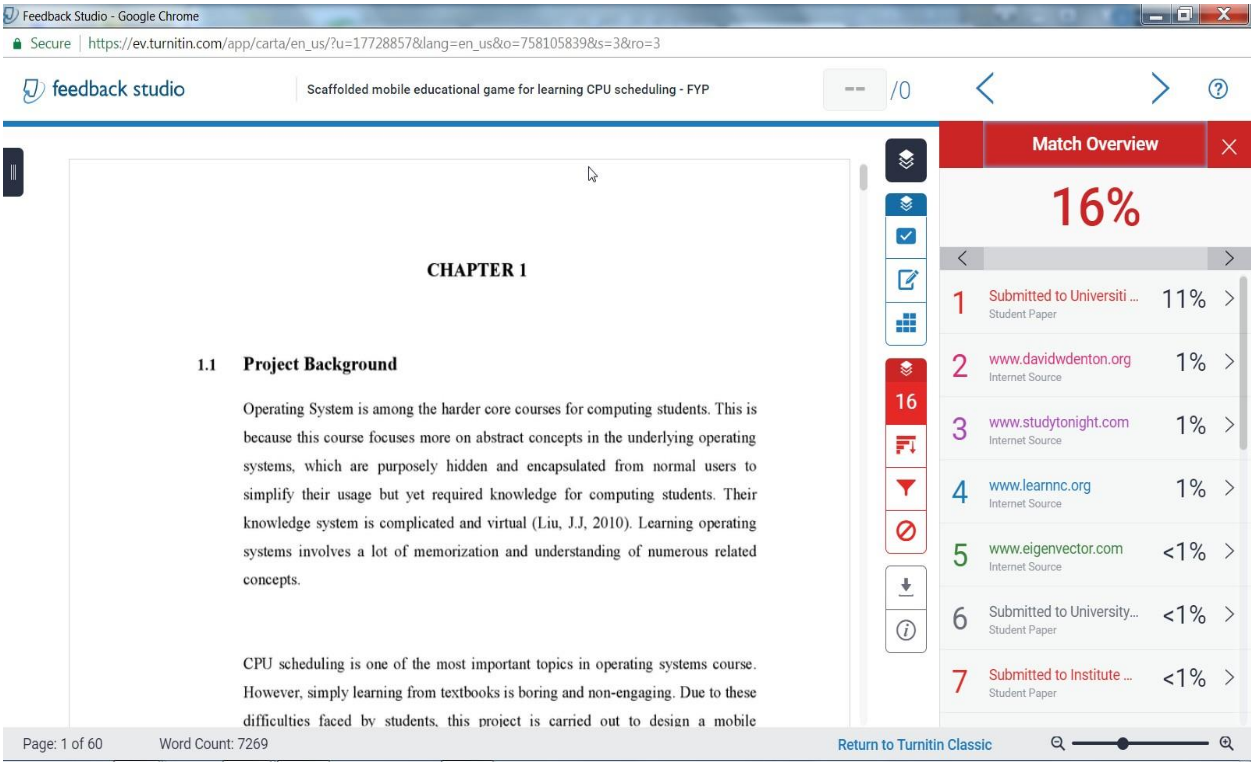
F3 – PROPOSAL / PROJECT IN-PROGRESS FORM

STUDENT NAME	**NAJAH RAIHAN MUSTAFAR @ MOKHTAR**	**STUDENT ID**	**2014564619**
PROGRAM	**BACHELOR OF COMPUTER SCIENCES (HONS.) MULTIMEDIA COMPUTING**		
SUPERVISOR	**ASSOC. PROF. DR. SYED AHMAD BIN SHEIKH ALJUNID**		
TITLE	**SCAFFOLDED MOBILE EDUCATIONAL GAME FOR LEARNING CPU SCHEDULING**		

MEETING DATE	COMPLETED ACTIVITIES	SUPERVISOR/CO-SUPERVISOR	
		NEXT ACTIVITIES/COMMENTS	SIGNATURE
17/3/2016	FYP Title Discussion		
22/3/2016	Identification of Game elements and Scaffolding		
23/3/2016	FYP Topic Discussion- CPU Scheduling		
28/3/2016	Discuss preliminary objectives, problem statement and review scaffolding game design		
1/4/2016	Finalize problem statement, objective, scope, background and significance		
5/4/2016	Chapter 1 Draft and Review OS Concept on CPU Scheduling		
29/4/2016	Identification of Mobile Development tools		

MEETING DATE	COMPLETED ACTIVITIES	SUPERVISOR/CO-SUPERVISOR	
		NEXT ACTIVITIES/COMMENTS	SIGNATURE
13/5/2016	Mind-mapping the brief details of CPU Scheduling Concept		
20/5/2016	Brief understanding about OS		
15/6/2016	Discussion for proposal presentation		

APPENDIX E
(TURNITIN RESULT)



Feedback Studio - Google Chrome
Secure
https://ev.turnitin.com/app/carta/en_us/?u=17728857&lang=en_us&o=758105839&s=3&ro=3
feedback studio
Scaffolded mobile educational game for learning CPU scheduling - FYP
-- /0
Match Overview
16%
16
Page: 1 of 60
Word Count: 7269
Return to Turnitin Classic

CHAPTER 1

1.1 Project Background

Operating System is among the harder core courses for computing students. This is because this course focuses more on abstract concepts in the underlying operating systems, which are purposely hidden and encapsulated from normal users to simplify their usage but yet required knowledge for computing students. Their knowledge system is complicated and virtual (Liu, J.J, 2010). Learning operating systems involves a lot of memorization and understanding of numerous related concepts.

CPU scheduling is one of the most important topics in operating systems course. However, simply learning from textbooks is boring and non-engaging. Due to these difficulties faced by students, this project is carried out to design a mobile

1	Submitted to Universiti ... Student Paper	11%
2	www.davidwdenton.org Internet Source	1%
3	www.studytonight.com Internet Source	1%
4	www.learnnc.org Internet Source	1%
5	www.eigenvector.com Internet Source	<1%
6	Submitted to University... Student Paper	<1%
7	Submitted to Institute ... Student Paper	<1%

Figure 1 Turnitin Result