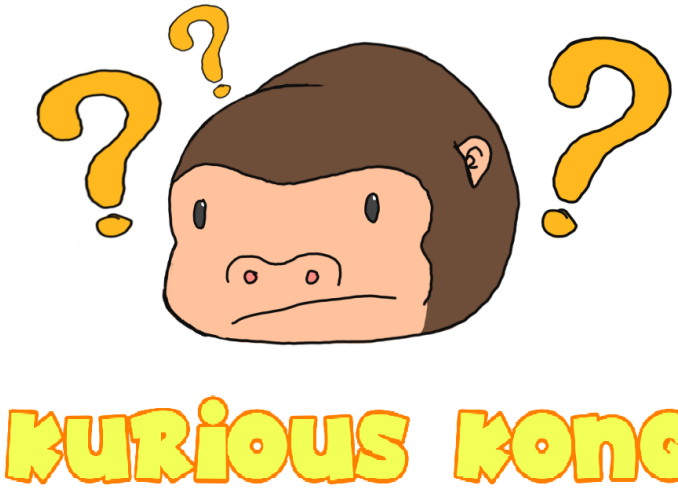
# ASSIGNMENT 2

## COSC2625 BUILDING IT SYSTEMS 2018
## 'KING KONG AND FRIENDS 2.0'




Kyongsub Kong | s3634359
Ming Jie Guan | s3723009
Ty Ty Chau | s3668469
Matthew McCarthy | s3718180
Huanghao Li | s3669467

# Table of Contents

# MEET OUR TEAM

"King Kong and Friends is made up of RMIT students with diverse culture, characters and personalities, which can maximise our outputs with creativity and various perspectives. We have teamed up together to translate dreams into reality."

**Kyongsub Kong**
Leader
/ Programmer
s3634359

**Ming Jie Guan**
UI Designer
s3723009

**Ty Ty Chau**
Programmer
/ Graphic Designer
s3668469

**Matthew McCarthy**
Graphic Designer / Sound Producer
s3718180

**Huanghao Li**
Programmer
s3669467

# PROJECT DESCRIPTION

The Project 'Kurious Kong' is an android application based on 2D platformer. Our project 'goal is to develop an education app for young children. The product's goal is to make the learning process more interesting for them. We intend on developing a game that contains a few mini-games, each has its own topic and it uses education as a means of progression. Once the player has cultivated enough points through completing challenges and quizzes within the mini-games, they can spend those points for cosmetics items. As stated, the main goal is to help kids to understand the topics better and that they could apply the knowledge in their class and hope to become the best within their class, maybe even school.

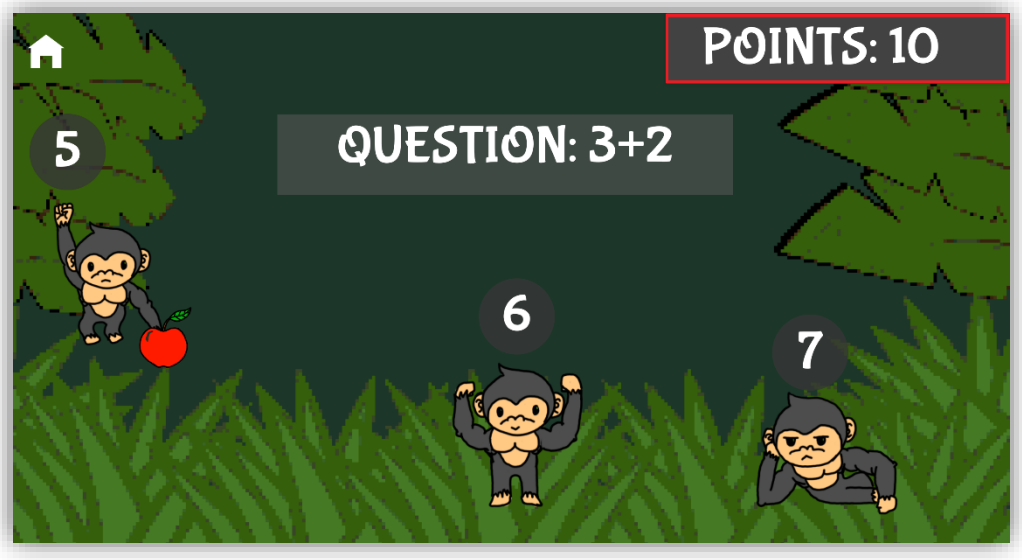
We aim to successfully develop a product that bring children knowledge as well as entertainment. The application will have three and maybe four categories (yet to be completely decided) upon launching the app. Each category is labelled with the topic of its own, there are two maths related and one to two literacy related. For maths, it will have "Addition" and "Multiplication" and for literacy, "Spelling" and perhaps Picture to Speech (yet to plan this one out).

The game itself will have difficulties, these can be unlocked once the player reaches a certain level. Levels can be increase by completing questions. For an example, if you complete 10 questions without any mistakes, you gain some points.

If everything goes well, the team is planning to extend the functionality of the app. What we meant by extending is that to add more function to it, the ones we have come up so far are login system, whereas the user can register and login, they can save their progress this way and no need to start all over again on a new device. Some type of in game points that the player obtained upon completing weekly challenges and if we have online mode, where players can challenge each other, the winner will gain some of the points as well. The points then can be used to purchase cosmetic items. To ensure we have enough funding to keep the game going, there will be some kinds of premium currency, those can be obtained through microtransaction, it can then be used to buy exclusive cosmetics items. Another is that maybe some kind of energy bar system, to prevent kids gets too addicted to the game, not that they would but as a safety measure, for an example there will be 5 energies, each time the player starts a challenge and fail, then it'll consume an energy, if the player successfully completes the challenges then the energy won't be consumed.

# CORE FEATURES

1. **Pointing System**

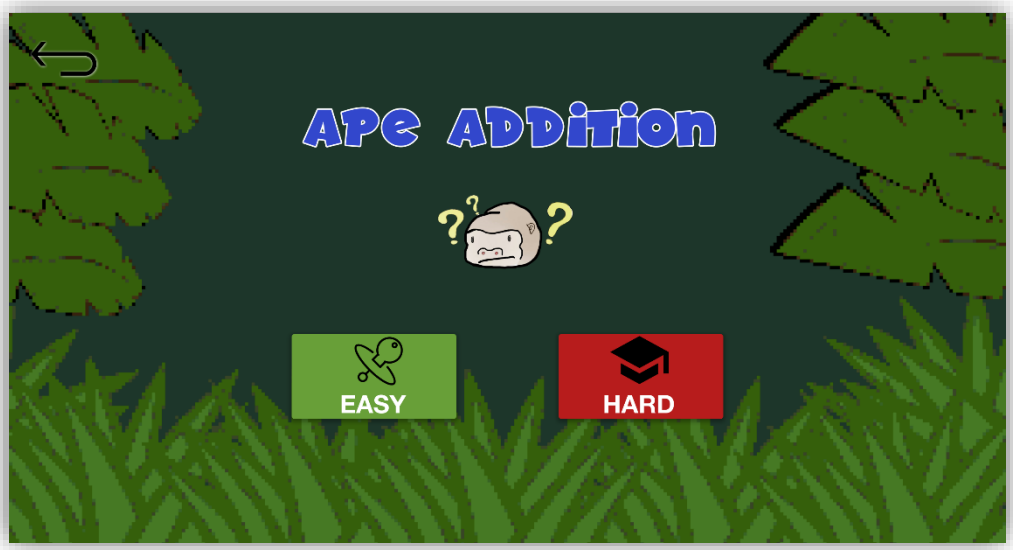




Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

## 2. Difficulty Option



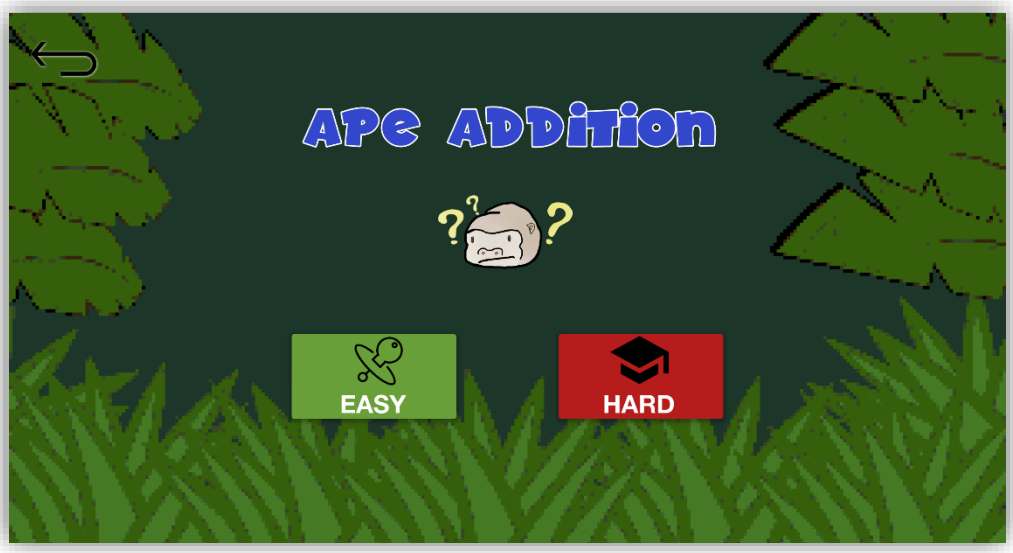


Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

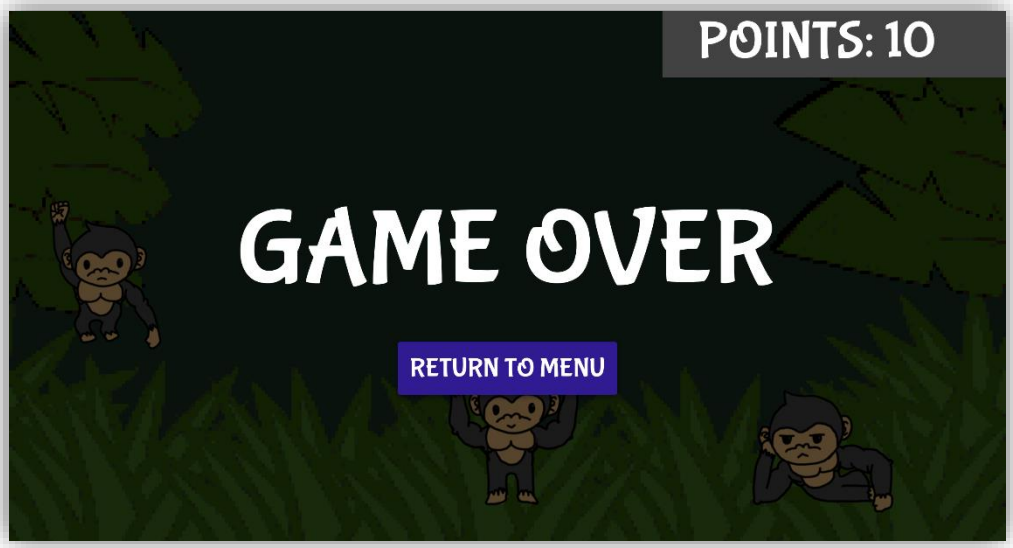
Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

### 3. Addition Game

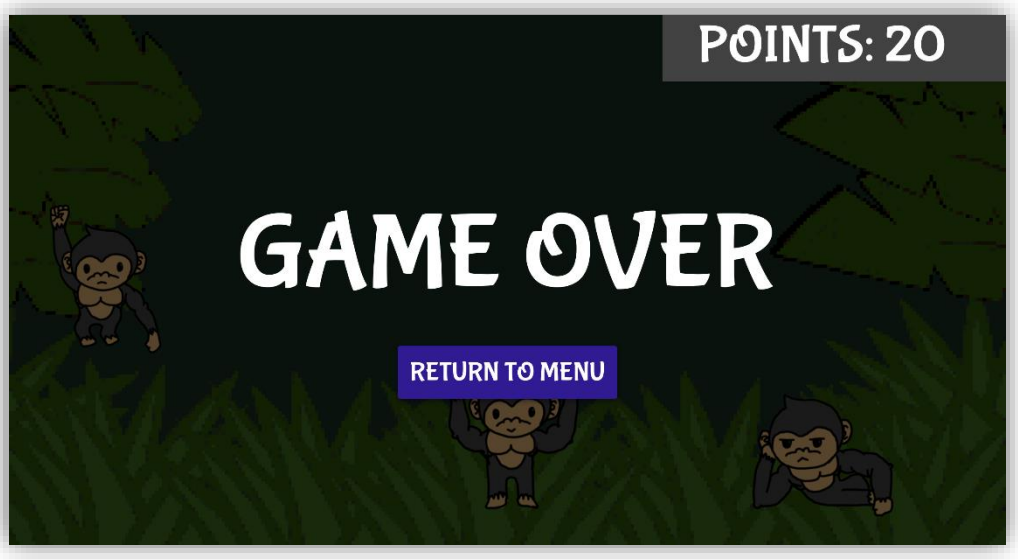





POINTS: 10
GAME OVER
RETURN TO MENU


POINTS: 0
50
QUESTION: 33+27
65
60


POINTS: 20
50
QUESTION: 33+27
65
60



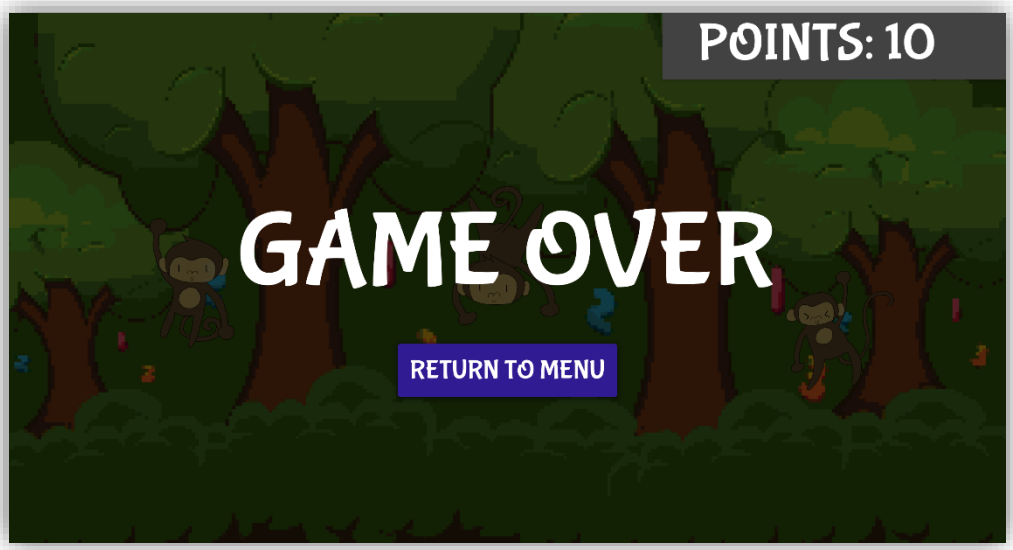

Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

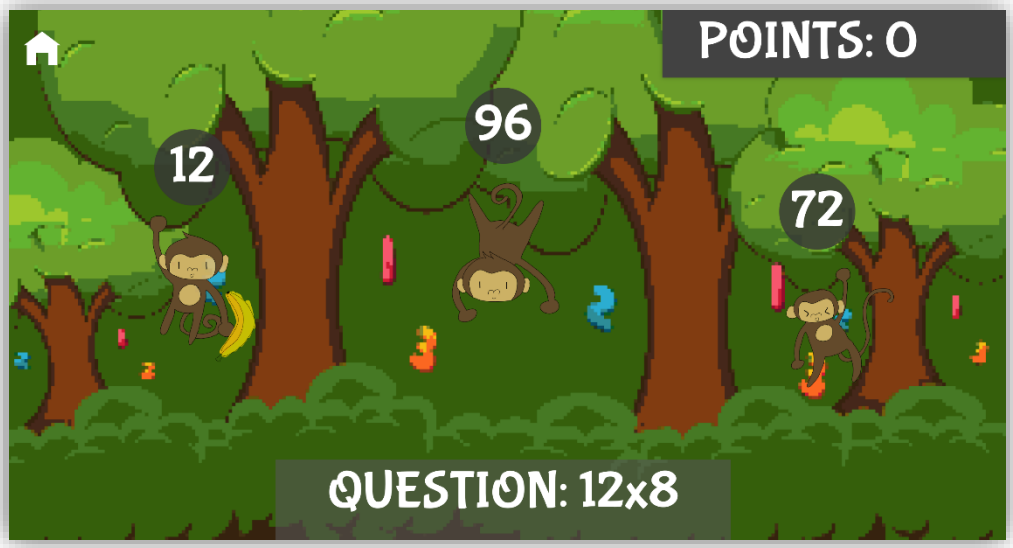
Validation Test

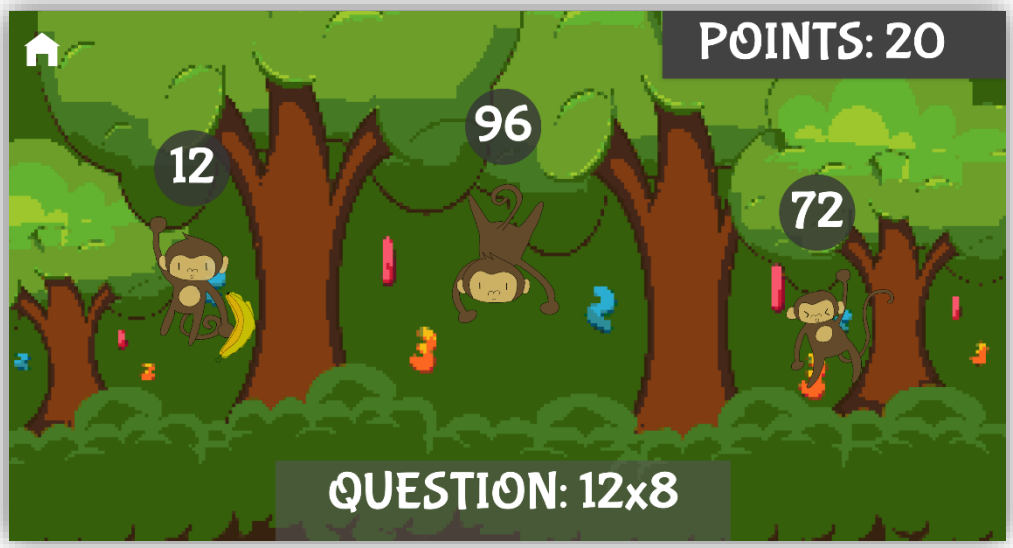
The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

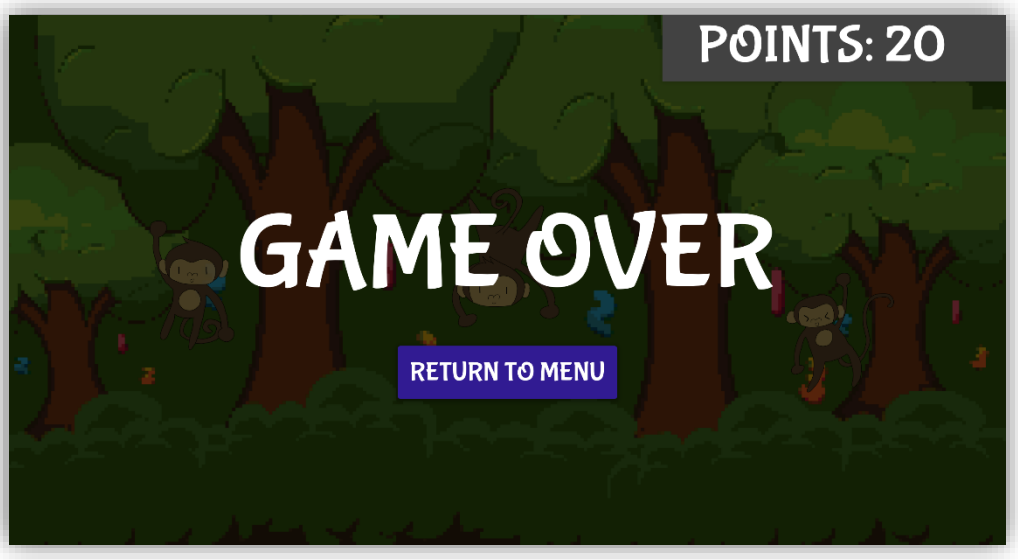
4. **Multiplication Game**




POINTS: 10
GAME OVER
RETURN TO MENU


POINTS: 0
12
96
72
QUESTION: 12x8


POINTS: 20
12
96
72
QUESTION: 12x8

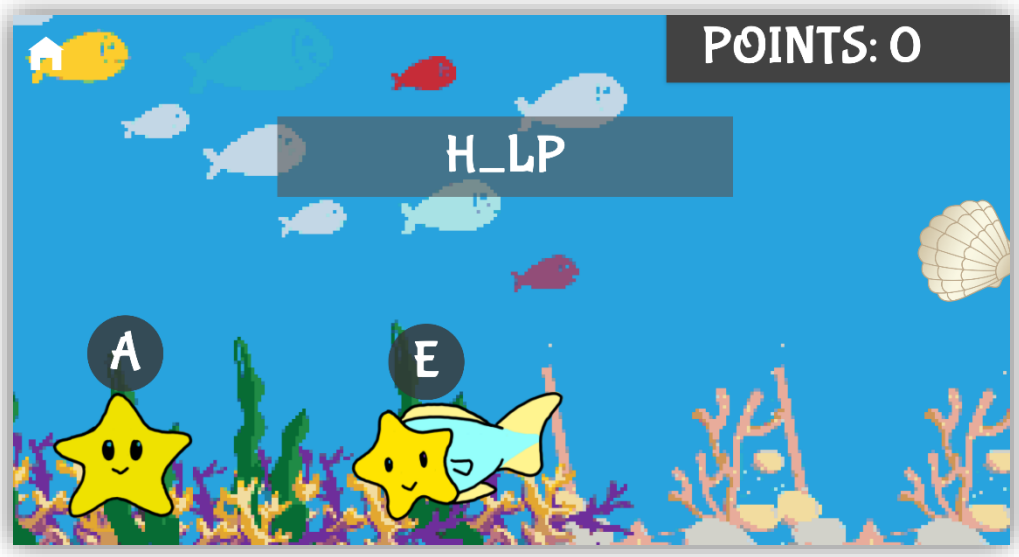


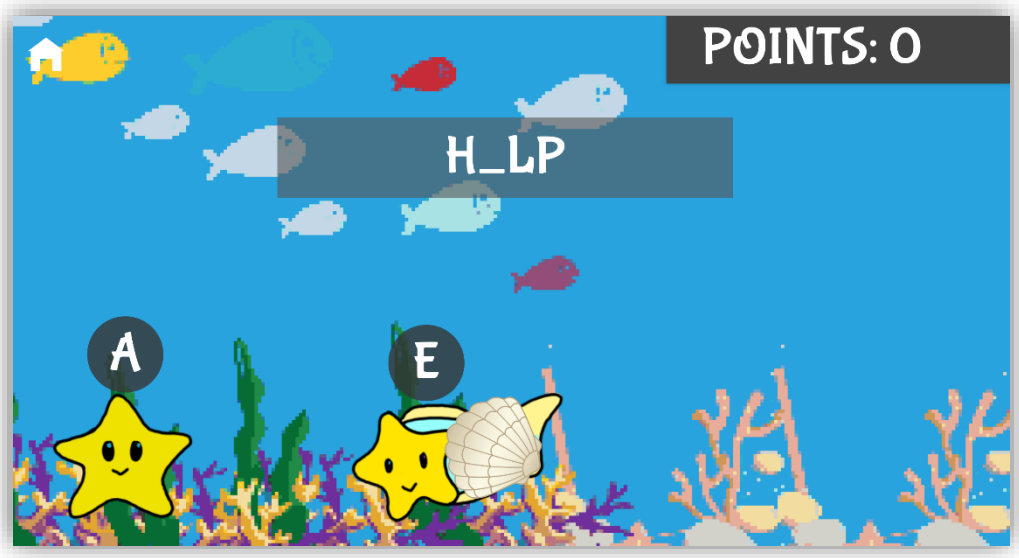

Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

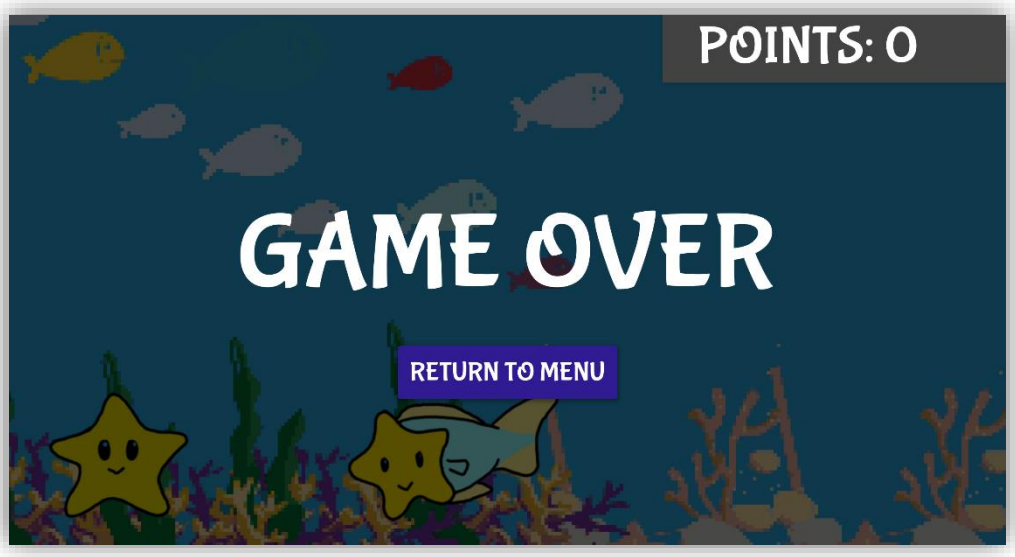
Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

5. **Spelling Game**

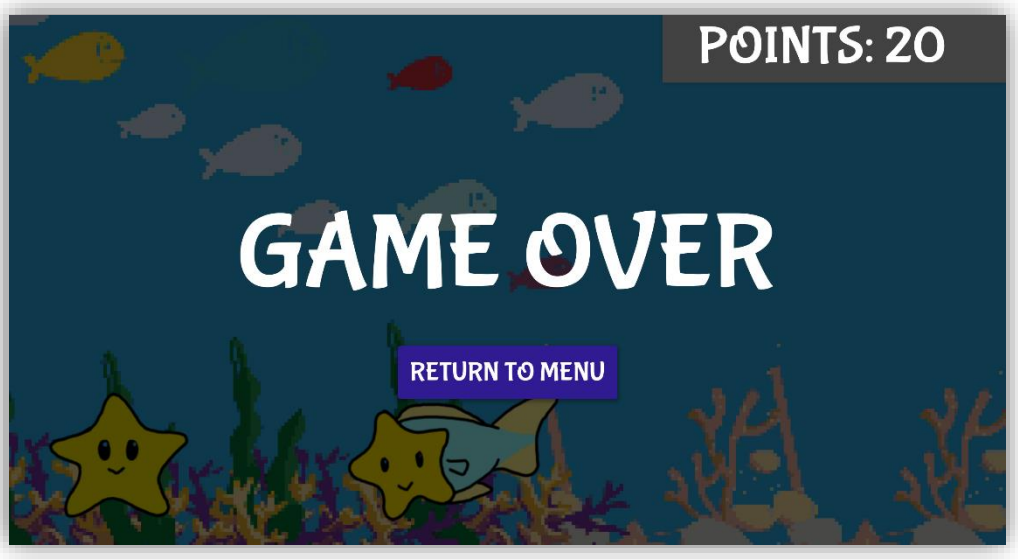




POINTS: 0
GAME OVER
RETURN TO MENU


POINTS: 0
AVO_ADO
C
K


POINTS: 20
AVO_ADO
C
K




Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

# TIME ESTIMATION

| Kurious Kong | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | Renee | Kyong | Ming | Ty | Matthew | Huanghao | Av. |
| MVF 1 | Pointing System | | 20 | 10 | 12 | 15 | | 14 |
| MVF 2 | Difficulty Option | | 15 | 20 | 20 | 15 | | 18 |
| MVF 3 | Addition Game | | 55 | 30 | 30 | 60 | | 44 |
| MVF 4 | Multiplication Game | | 30 | 10 | 20 | 25 | | 21 |
| MVF 5 | Spelling Game | | 40 | 30 | 20 | 35 | | 31 |
| EVF 1 | Avatar | | 25 | 6 | 8 | 35 | | 17 |
| EVF 2 | Account Creation | | 25 | 40 | 20 | 35 | | 30 |
| EVF 3 | Login Function | | 20 | 30 | 15 | 20 | | 21 |
| | | | | | | | Total : | 230 |

## JUSTIFICATION

- **MVF 1 – POINTING SYSTEM**

| | |
|---|---|
| STEPS | Designing + Coding + Testing |
| | Designing in-app pictures seems quite simple compared to other designing work. Also, royalty free font will be used for numbers and texts. So, design process may take 5 hours in total. However, coding is expected to take longer time, because none of us is familiar with C# language which has to be used to create our project game in Unity. Validating test has to be done to test whether it functions properly or not and errors from this test should be corrected. |
| TIME | 5(Designing) + 10(Coding) + 5(Testing)= 20 |

- **MVF 2 – DIFFICULTY OPTION**

| | |
|---|---|
| STEPS | Designing + Coding + Testing |
| | Design for this feature is simply to make two boxes with options. Coding will be implemented once two different game modes for each game are created. This feature asks users to choose one option out of two modes. Coding and Validation test will take less than 10 hours. |
| TIME | 5(Designing) + 5(Coding) + 5(Testing)= 15 |

- **MVF 3 – ADDITION GAME**

| | |
|---|---|
| STEPS | Researching + Learning + Designing + Coding + Testing |
| | Research for each game needs at least 5 hours in order to draw the overall picture of the feature. Learning Unity takes at least 5 hours to understand its basic function and how it works and another 5 hours are necessary to know how C# language works. Designing background, characters, animations and sound will take at least 20 hours. Coding is expected to take 15 hours, including 3 random numbers generation, a random equation generation and a basic correction with effect sound and a graphic. |

| | |
|---|---|
| | Validation Test : 5 hours |
| TIME | 5(Researching) + 10(Learning) + 20(Designing) + 15(Coding) + 5(Testing)= 55 |

- **MVF 4 – MULTIPLICATION GAME**

| | |
|---|---|
| STEPS | Designing + Coding + Testing |
| | This game basic function will be similar to the first game, so we can save time for researching and learning. However, designing might take the same time as it, since the multiplication game will has totally different background, characters and animations. Code will be similar to the first game, which saves us time on coding.<br>Validation Test : 5 hours |
| TIME | 20(Designing) + 5(Coding) + 5(Testing)= 30 |

- **MVF 5 – SPELLING GAME**

| | |
|---|---|
| STEPS | Researching + Designing + Coding + Testing |
| | We need to research suitable words which will be used in the game to the age of the users. This game basic function will be different to the other two games, because this needs to generate some random words with a blank and example letters that users can select. designing will take the same time as the others. Since code will be different to the others, which enables us to spend at least 15 hours on coding.<br>Validation Test : 5 hours |
| TIME | 3(Researching) + 20(Designing) + 15(Coding) + 5(Testing)= 43 |

- **EVF 1 – AVATAR**

| | |
|---|---|
| STEPS | Designing + Coding + Testing |
| | Time estimation for designing avatars varies considerably in quality and quantity. However, we have to limit the time for designing avatars, because our project game is for education with 2D pixelated images.<br>Validation Test : 5 hours |
| TIME | 15(Designing) + 5(Coding) + 5(Testing)= 43 |

- **EVF 2 – ACCOUNT CREATION**

| | |
|---|---|
| STEPS | Designing + Coding + Testing |
| | The steps for EVF 2 are Designing + Coding + Testing.<br>We need to design the main menu for the game to create this feature. Coding for this feature requires some SQL codes, php and a web hosting server to store users' accounts and passwords. Since these are our first attempts, we cannot estimate the exact time for this. However, we believe all the coding process will take roughly 20 hours and we need another 5 hours for validating. |
| TIME | 5(Designing) + 20(Coding) + 5(Testing)= 30 |

- **EVF 3 – LOGIN FUNCTION**

| | |
|---|---|
| STEPS | Designing + Coding + Testing |
| | This feature also needs some SQL codes, php and a web hosting server to retrieve users' accounts and passwords. Coding should be implemented with account creation.<br>Validation Test : 5 hours |
| TIME | 5(Designing) + 10(Coding) + 5(Testing)= 20 |

# TECHNOLOGIES

The technologies the team has been using throughout this project are divided under four categories, collaborative workspace, software, tools and resources. In this second report, it will have a few changes compare to the previous report:

- **Collaborative workspaces**

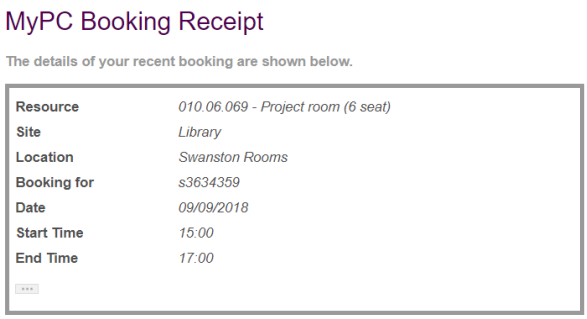
- Trello:
  'Trello' is a project management tool online, which we are recommended to use for the project. This helps us to set our goal, keep each individual member on track as well as to set each person a goal, or a project to do. By using this, we can get to see how much each member is contributing to the project and make sure that everyone did their work equally.

- GitHub:
  At the moment, we only use 'Github' for sharing draft reports, however as we officially begin to start coding, we will be using GitHub to push and pull our codes each other to test and fix any bugs that we might encounter.

- Google Drive:
  Everyone has a role to take part when it comes to writing a report. 'Google Drive' is the most ideal tool we could use to write our own part and then share it with each other, combine it and finalise it in real-time together.

- Messenger:
  All members are more active on 'Facebook', so we decided to use 'Facebook messenger' to talk and discuss with each other most of the time. Messenger is more likely to allow us to have a conversation instantly due to its portability.

- Face-to-face meet up:
  The team tends to gather together in the library and sometimes in a pre-booked room when most of us have free time. We discuss about the project more and work on it together and help one another out if anything needs to be done. A face-to-face meeting helps us to easily understand others' thoughts or ideas.


MyPC Booking Receipt

The details of your recent booking are shown below.

| | |
|---|---|
| Resource | 010.06.069 - Project room (6 seat) |
| Site | Library |
| Location | Swanston Rooms |
| Booking for | s3634359 |
| Date | 09/09/2018 |
| Start Time | 15:00 |
| End Time | 17:00 |


MyPC Booking Receipt

The details of your recent booking are shown below.

| | |
|---|---|
| Resource | 010.06.069 - Project room (6 seat) |
| Site | Library |
| Location | Swanston Rooms |
| Booking for | s3634359 |
| Date | 09/09/2018 |
| Start Time | 13:00 |
| End Time | 15:00 |

- **Software**

- Unity:
  We use 'Unity' as our main program for this 2D game project as it is a very viable program. 'Unity' can also be used to develop games for iOS, PC, PS4 and more, not just limited to android apps only.

- Krita:
  We use 'Krita', a free sketching app, to design and sketch sprites for our project. Our designer draws characters, logo and the like with this app.

- Photoshop cc:
  'Photoshop' is mainly used for designing the background and animations for the project.

- Visual Studio:
  'Visual Studio' is a code editor program that installed with Unity. We use this to code and develop our application by using C# language throughout this project.

- Proto.io / Invision:
  'KingKong and Friends' uses 'Proto.io' to develop our high-fidelity prototypes and to see what the product should look like. It's like a guideline for us to use and trace our steps. 'Invision' is used to improve the prototypes by creating hotspots on an action button/link.

- Audacity:
  'Audacity' is utilised to edit and create sound effects, background music for our project.

- **Tools**

- Laptop - Touch screen + pen:
  We use a touchscreen laptop that comes with a pen to design and finalise our images and sprites with precisions.

- Android Mobile Phones:
  If our project game is completed in the allowed time, we plan to test the game with some android devices.

- **Resources**

- Google and YouTube:
  We search and learn how to code and utilise the apps that are required to use for the project by using 'Google' as well as 'YouTube' for demonstrations. This is a big help in this project as majority of the members have no experience in using the programs that are listed above.

- Unity Tutorial:
  The tutorials are provided on the 'Unity' website itself. This helps young developers to design and develop their first game from the beginning. We use some videos on the website to aid us with developing the project.

- Royalty Website:
  We obtained most of the sample sounds on this website and some were edited by one of our members, who is in charge of the soundboard.

- Pixel-stitch:
  Since 'Kurious Kong' is based on pixelated images for mini games, 'Pixel-stitch' is used for image pixelization.

# APPENDIX

## EXTENDED FEATURES

### 1. Avatar




Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

## 2. Account Creation

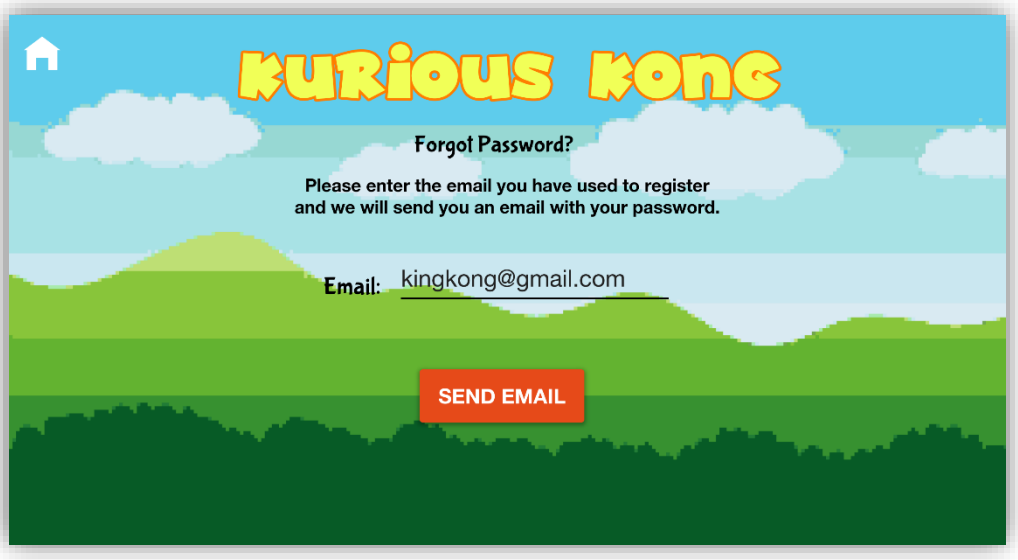

Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.

### 3. Login Function






Implementation of a point system where users are awarded points for correctly answering a question. Users' can use this feature to compete with others adding a competitive feature to the game.

Validation Test

The game shows visual feedback of the points being earned by correctly which all adds up every time a question is answered correctly and the total points will be shown when its game over.