

# Plymouth University

## School of Computing, Electronics and Mathematics

PRCO304  
Final Stage Computing Project  
2017/2018

BSc (Hons) Computing and Games  
Development

*Ludwik Bacmaga*

10595124

UX in AR Games

Supervisor: Mr Marius Varga  
Second Marker: Dr Nigel Barlow

## Acknowledgements

I'd like to give my thanks to all those who have encouraged and influenced this project, through participating in testing or just providing ideas for improvements.

Special thanks to Marius Varga, as my project supervisor, supporting me any way possible throughout this project, making it as good as possible, pushing me to try my best.

I would also like to thank my girlfriend, who offered emotional support throughout the project and for listening to me even when she didn't know what I was talking about.

## Abstract

Augmented Reality, AR, is a technology that haven't been explored much as of yet when it comes to game development, or application development, as it is still, debateable, a new technology. There are still areas which can be improved on, but for application success what is very important is to have very good user experience, UX. This research explores this area and tries to implement an AR game, using ARKit, with a well-designed UX that would be enjoyable for the user.

This report discusses in detail the processes used throughout the project, starting at the background and research of AR and existing titles, to specifying the requirement and the process that it took to complete this project. Every stage of implementation is explained, with the design choices that were made in the implementation stage. Explaining how the project has developed from initial implementation, which included bad implementation of UX, and improved towards the final product.

The development technologies used for this project, and mentioned in this report, were ARKit, using Unity game engine and C# programming language for implementation.

Report explains each stage of the implementation, where the report is finished with the project evaluation of the work completed. This is done through project evaluation, future improvements, objective evaluation, lessons learnt in the project, project port-mortem, and conclusions gathered at the end. The purpose of those evaluations is to evaluate project performance, as well as to see whether the objectives been met. Also, to evaluate what went well and what could be improved in this project.

Appendices contain details of project management used for the project, and other assisting documentations supporting this project, including testing results.

## Table of Contents

<b>ACKNOWLEDGEMENTS.....</b>	<b>2</b>
<b>ABSTRACT .....</b>	<b>2</b>
<b>TABLE OF CONTENTS .....</b>	<b>3</b>
<b>1. INTRODUCTION.....</b>	<b>5</b>
<b>2. BACKGROUND, OBJECTIVES &amp; DELIVERABLES.....</b>	<b>5</b>
2.1. BACKGROUND.....	5
2.2. AIM AND OBJECTIVES.....	6
2.3. DELIVERABLES.....	6
<b>3. RESEARCH .....</b>	<b>7</b>
3.1. VISUALISING GRAPHS IN THREE DIMENSIONS .....	7
3.2. STANDARDS FOR AUGMENTED REALITY: A USER EXPERIENCE PERSPECTIVE .....	9
3.3. EXISTING GAMES .....	11
3.4. SUMMARY.....	14
<b>4. METHOD OF APPROACH.....</b>	<b>14</b>
4.1. APPROACH .....	14
4.2. TECHNOLOGIES .....	15
4.3. PROJECT MANAGEMENT.....	15
<b>5. LEGAL, SOCIAL, ETHICAL AND PROFESSIONAL ISSUES .....</b>	<b>17</b>
<b>6. IMPLEMENTATION .....</b>	<b>18</b>
6.1. PROCESS .....	18
6.2. REQUIREMENTS.....	19
6.3. DESIGN .....	20
6.4. PROTOTYPE .....	21
6.4.1. <i>Minimum Viable Product</i> .....	21
6.4.2. <i>Testing</i> .....	23
6.4.3. <i>Results</i> .....	23
6.5. CONTROL PLAN .....	24
6.5.1. <i>User Interface</i> .....	24
6.5.2. <i>Game Interaction</i> .....	24
6.5.3. <i>Visuals</i> .....	25
<b>7. STAGES .....</b>	<b>26</b>
7.1. STAGE 1: GAME DESIGN & TECHNOLOGY ANALYSIS .....	26
7.2. STAGE 2: GAME IMPLEMENTATION .....	27
7.3. STAGE 3: GAME MECHANICS.....	27
7.4. STAGE 4: COMPLETE GAME PROTOTYPE.....	28
7.5. STAGE 5: TESTING & FEEDBACK GATHERING .....	28
7.6. STAGE 6: IMPROVEMENTS IMPLEMENTATION .....	28
<b>8. EXTERNAL LIBRARIES.....</b>	<b>30</b>
<b>9. PROJECT EVALUATIONS.....</b>	<b>30</b>
9.1. END PROJECT REPORT .....	30
9.2. PROJECT PORT MORTEM .....	30
9.3. OBJECTIVE EVALUATIONS.....	32
<b>10. CONCLUSIONS.....</b>	<b>33</b>

10.1. LESSONS LEARNT .....	33
10.2. FURTHER DEVELOPMENT .....	33
10.3. CONCLUSION .....	34
<b>11. REFERENCES.....</b>	<b>35</b>
<b>APPENDICES: .....</b>	<b>37</b>
APPENDIX A: PID .....	37
APPENDIX B: HIGHLIGHTS .....	45
APPENDIX C: STAGE OBJECTIVES, PLANS AND REVIEWS.....	52
APPENDIX D: RESEARCH DOCUMENTS.....	58
APPENDIX E: ETHICS APPROVED APPLICATION.....	63
APPENDIX F: ETHICS SIGNED PARTICIPANTS .....	76
APPENDIX G: DEVELOPER DIARY .....	83
APPENDIX H: TEST RESULTS .....	107

**Word Count:** 10,539

**Project URL:** [https://liveplymouthac-my.sharepoint.com/:f/g/personal/ludwik\\_bacmaga\\_students\\_plymouth\\_ac\\_uk/EhtCoqwaJOVBueHmZ6YjU74Ba2N9LV5-FUgvT2vzHRHdbA?e=uk9L84](https://liveplymouthac-my.sharepoint.com/:f/g/personal/ludwik_bacmaga_students_plymouth_ac_uk/EhtCoqwaJOVBueHmZ6YjU74Ba2N9LV5-FUgvT2vzHRHdbA?e=uk9L84)

**GitHub (alternative):** <https://github.com/Luddzik/OiOi/invitations>

## 1. Introduction

Project focuses on creating developing an Augmented Reality, AR, game with well implemented User Experience, UX. This is achieved by using ARKit technology, created by Apple, and using ARKit Unity Plugin to implement it using Unity 3D game engine. Purpose for this project was to research and develop good UX for AR game, since it is still a new technology.

UX consists of different features, namely: User Interface, Usability, Application Interaction, and any other features that influence user satisfaction with the game, application. By changing implementation of a feature, it will affect the UX; this project aims in finding UX that is enjoyable for the user in AR game.

Creating good UX will be created by creating prototypes, followed by testing and gathering feedback on how to improve on the prototype. The core mechanic of the game would be kept the same, just the implementation would change based on the feedback gathered. By the end, from feedback gathered and also from experience by playing the game, it should be presentable if UX does impact enjoyment for the player.

## 2. Background, Objectives & Deliverables

### 2.1. Background

User Experience, UX, is very crucial for a game, or application, success. Especially within the new technology which is still evolving, like Augmented Reality, AR. Augmented Reality, AR, combines virtual with the real world using devices such as mobile phones.

Before ARCore and ARKit come out, all other kits that offer development in Augmented Reality required a marker to operate, however, ARKit and ARCore focus on markerless implementation. With this new implementation, there are different ways to implement AR and it is important to utilise UX to take advantage of this new technology.

UX takes into account different things, but it focuses on end user experience. It is accomplished by the way that information is presented to the user by the application:

- User Interface, UI
- Usability
- Application Interaction
- User Research and Data
- User Testing

UX uses all of these features to do research and create prototypes, for final product to be user-friendly and intuitive for people to want to come back and use the application again.

## 2.2. Aim and Objectives

Aim for this project is to create a game in Augmented Reality, with a good user experience. It is accomplished by testing out different prototypes and researching existing titles.

Objectives for this project were:

- Gain experience using Unity and ARKit package within Unity
- Experience with Xcode and application development for iOS platform
- Better understanding of UX in AR
- Scope out project milestones for project development
- Do prototypes and testing along the development stage

## 2.3. Deliverables

The deliverables for this project is an iOS game application, implemented in ARKit, which focuses on user experience, UX. Details can be found in section 7.

### 3. Research

For this project, *Standards for Augmented Reality: a User Experience perspective* [2], and *Visualising Graphs in Three Dimensions* [3] research papers were read and studied which affected how the project was implemented. *Visualising Graphs in Three Dimensions* [3] research paper focuses on how information should be presented to be more distinctive for the human eye. The perception of information. *Standards for Augmented Reality: a User Experience perspective* research paper focuses on how user interacts with AR technology directly and how the information is feedback to the user. Those papers were beneficial in developing this project further and to have successful outcome. Moreover, existing game titles were looked at and evaluated to figure out what work well and what doesn't in those titles, adapting positive, relevant, features to this project.

#### 3.1. Visualising Graphs in Three Dimensions

*Visualising Graphs in Three Dimensions*, focuses on graph representation to the user to maximise number of nodes that are distinguishable for the human eye. It compares 2D representation to 3D representation which illustrates the difference and amount of detail that 3D offers compared to 2D representation.

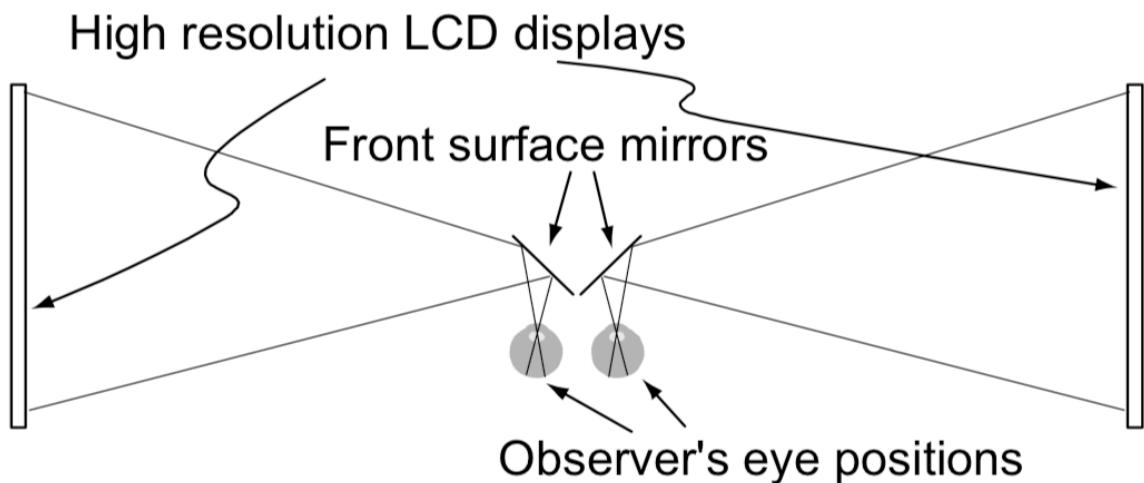
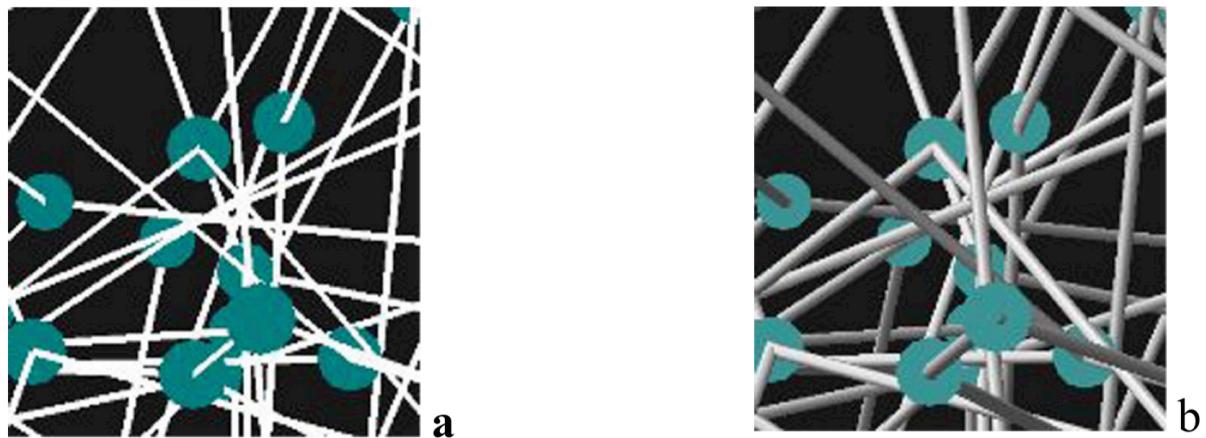
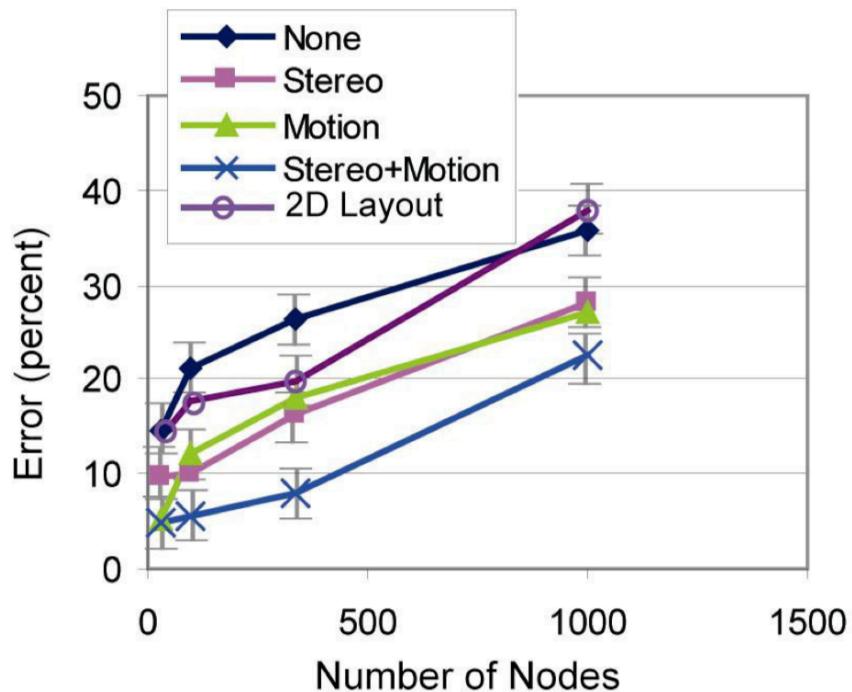


Figure 1. A Wheatstone stereoscope arrangement provides a ghost-free display. [3]



*Figure 2. Snapshot 'a' shows 2D graph representation, snapshot 'b' shows 3D representation. [3]*

*Figure 1* illustrates how the test was constructed, by a person looking at graphs, as seen in *Figure 2*. The response time, and correct answers were recorded, and based on those results the conclusions were made. Each test was done under different conditions, using different techniques to help tester get correct results. The conditions were: stereo, motion, rotation, 2D and 3D image. *Figure 3* shows the error rate as a function of graph size.



*Figure 3. Errors as a function of graph size. Average data based on 14 participants. [3]*

Based on the results from *Figure 3*, having 3D representation s more distinguishable to the human eye to detect. If it is combined with motion and stereo, the error rate is drastically reduced. From that research paper, conclusion was made that factors such as: stereoscopic disparities, kinetic depth, texture, perspective, shadows all affect how visible the distinguishable data is for the viewer.

Based on those findings, this project is using lighting to create shadows, made in 3D, and using motion to assist in perspective of objects for the player. Believe those factors do have impact on the user experience.

### 3.2. Standards for Augmented Reality: a User Experience perspective

*Standards for Augmented Reality: a User Experience perspective* [2] research paper evaluates different ways to interact with AR application, and what outputs are there for application to give feedback to the user. It depends on the application, and what it aims to accomplish, that decides which input and output are used to interact with the user.

In ISO, 9241-201 [1], definition of user experience is: “*a person’s perceptions and responses that result from the use or anticipated use of a product, system or service*”. With that definition, in the paper the user experience, UX, includes: responses that are either physical or psychological, preferences, user emotions, beliefs, perceptions, before/during/after behaviour and accomplishments, as well as the media that are used in the application.

In this research paper, different inputs and outputs were analysed where at the end it provides clear conclusion on what works well for the AR or VR application. However, the inputs and outputs used in the application are affected by the application goals and uses.

Inputs for AR applications that were taken into the account were: tactile, visual, auditory, sensory modalities, kinaesthetic. *Table 1* represents each input with the effects on the UX.

*Table 1: AR Inputs and how it affects UX.*

<i>Input</i>	<i>Input use</i>	<i>Effect on UX</i>
Tactile	<ul style="list-style-type: none"> <li>● Joystick</li> <li>● Keyboard</li> <li>● Touch-screen</li> <li>● Mouse</li> </ul>	<ul style="list-style-type: none"> <li>● Ease of use</li> <li>● Accuracy</li> <li>● Responsiveness</li> <li>● Comfort</li> </ul>
Visual	<ul style="list-style-type: none"> <li>● Camera Tracking <ul style="list-style-type: none"> <li>○ Marker</li> <li>○ Non-Marker</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>● Ease of use</li> <li>● Responsiveness</li> <li>● Accuracy</li> <li>● Privacy/Social comfort</li> </ul>
Auditory	<ul style="list-style-type: none"> <li>● Inferring user context</li> </ul>	<ul style="list-style-type: none"> <li>● Accuracy</li> </ul>

	<ul style="list-style-type: none"> <li>○ Detected frequencies</li> <li>○ Ambient noise level</li> <li>○ Patterns</li> <li>● Voice command</li> </ul>	<ul style="list-style-type: none"> <li>● Social comfort</li> <li>● Responsiveness</li> </ul>
Sensory modalities	<ul style="list-style-type: none"> <li>● Sensors <ul style="list-style-type: none"> <li>○ Detect user/environment</li> <li>○ Hybrid-sensors involving more than 1 sensor</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>● Accuracy</li> <li>● Seamless switching</li> </ul>
Kinaesthetic	<ul style="list-style-type: none"> <li>● Motion tracking</li> <li>● Track body</li> </ul>	<ul style="list-style-type: none"> <li>● Ease of use</li> <li>● Accuracy</li> <li>● Responsiveness</li> </ul>

Based on those findings, there are common features that affect that UX, which are namely the accuracy, responsiveness, and the ease of use. *Table 2* represents different outputs that can be used in AR applications.

*Table 2: Outputs for AR Applications*

Visual	<ul style="list-style-type: none"> <li>● Uses <ul style="list-style-type: none"> <li>○ Contrast</li> <li>○ Field of View</li> <li>○ Brightness</li> <li>○ Depth perception</li> </ul> </li> <li>● Content Quality <ul style="list-style-type: none"> <li>○ Realism</li> <li>○ Abstraction</li> <li>○ Frame rates</li> </ul> </li> <li>● World consistency and stability <ul style="list-style-type: none"> <li>○ Registration</li> <li>○ Stability</li> </ul> </li> <li>● Most important for UX</li> <li>● Can be simple or complex</li> <li>● Implications <ul style="list-style-type: none"> <li>○ Health and Safety <ul style="list-style-type: none"> <li>■ HMD (Headsets) <ul style="list-style-type: none"> <li>● Narrow field of view</li> <li>● Inadequate depth perception</li> </ul> </li> </ul> </li> </ul> </li> </ul>
--------	--

	<ul style="list-style-type: none"> <li>• Low display brightness</li> <li>• Poor ergonomics</li> </ul> <ul style="list-style-type: none"> <li>○ Context Awareness           <ul style="list-style-type: none"> <li>▪ Can lower implication with positional stability that is easier to accomplish</li> </ul> </li> </ul>
<i>Auditory</i>	<ul style="list-style-type: none"> <li>• Directional output (speakers)</li> <li>• Simple to implement</li> <li>• Useful for immersion</li> <li>• Implications           <ul style="list-style-type: none"> <li>○ Disruptive to environment</li> <li>○ Health and Safety</li> <li>○ Privacy</li> </ul> </li> </ul>
<i>Haptic</i>	<ul style="list-style-type: none"> <li>• Limited use</li> <li>• Focuses on touch and feel feedback</li> <li>• Use for Force-feedback systems           <ul style="list-style-type: none"> <li>○ Medical applications</li> <li>○ Game controllers</li> </ul> </li> </ul>

From *Table 2*, it can be concluded that visual output is most important aspect of UX, as it is what users interact with in the AR application. Auditory is secondary impact for UX, as it does help with immersion and to provide a sense of atmosphere, however it is limited to directional output and it may be disruptive to others. Haptic output is most limited as it uses touch and feel feedback, and for this project it will not be used, as it is hard to accomplish on mobile device.

In conclusion, from this research, the features that are focused on this project are visual aspects and auditory, to combine them to influence UX. It is accomplished by creating output that is visually stable, and satisfying field of view. Auditory is being used in helping to achieve immersion with the world that is presented using visual output.

### 3.3. Existing Games

One aspect that had to be considered when implementing and working on this project was how existing titles are implemented and what makes them successful.

*Table 3: Existing titles comparison table.*

Game Name	UI	Indicators	Features
AR Dragon [4]	Menus hidden within few onscreen buttons, once clicked it shows more options.	<ul style="list-style-type: none"> <li>• UI has striking colour buttons for easily</li> </ul>	<ul style="list-style-type: none"> <li>• Dragon needs to be spawn on big enough surface.</li> </ul>

	  <p>Clean design, only buttons showing on lower sides of the screen with expandable menus.</p>	<ul style="list-style-type: none"> <li>distinguishable buttons.</li> <li>World is spawn around you, only way to find out if to look around or if in practise mode (train mode) then the dragon looks in that direction.</li> <li>Distinguishable colours when it comes to pickups, shooting targets and mood of dragon can be seen by the face expression.</li> </ul>	<ul style="list-style-type: none"> <li>Game uses whole world around you, even outside the scanned surface. Can be seen when in train mode, or point spawn.</li> <li>As Dragon grows, it requires bigger space.</li> <li>Can be picked up and placed anywhere (has to be scanned first and a plane) and dragon can only fly within this plane.</li> </ul>
<i>Space Invaders – not released [5]</i>	Non-Existent, very minimalistic.  <p>Only information on screen is how many bullets are left in the gun. It is not released game, just prototype.</p>	<ul style="list-style-type: none"> <li>Pixelated enemies coming towards you from every side, the only way of seeing them is by looking around.</li> <li>Models becoming bigger the closer they are to the player.</li> </ul>	<ul style="list-style-type: none"> <li>Takes advantage of the whole environment around player by spawning enemies coming at the player.</li> <li>Tap the screen to play the game (Shoot)</li> <li>Tracks player location.</li> </ul>
<i>Space Shooter – not released</i>	Non-Existent, prototype build, minimalistic information shown on screen. 	<ul style="list-style-type: none"> <li>No real indicators showing ships – just models floating around player. Have to look around to spot them.</li> </ul>	<ul style="list-style-type: none"> <li>Ships spawning at random location around player</li> <li>Doesn't use planes to spawn objects</li> </ul>

	<ul style="list-style-type: none"> <li>Not interactive models, stand still</li> </ul>	
ARise [6]	<p>Minimalistic UI – Only options and menu buttons at the lower corner.</p>  <p>Simplistic UI, using camera positioning to progress with the game. Fits well with the game mechanic, to have most area to display the game.</p>	<ul style="list-style-type: none"> <li>Symbols that connect together and shine once the camera joins the symbols together.</li> <li>Symbol shine up when connected and the path is created.</li> <li>Use of colours to clearly show progression, player location (by using distinguishable colours compared to whole game), symbols, can see whole level at once.</li> </ul> <ul style="list-style-type: none"> <li>Requires a plane to spawn the world.</li> <li>Uses camera position to play the game.</li> <li>Able to see whole level once loaded, allows player to plan ahead.</li> <li>One spawn location, but by the game mechanic (of progression with camera position), forces people to move around the map.</li> </ul>
Splitter Critters [7]	 <p>Minimalistic UI – Pause and restart button. Clean UI, slide screen to play the game.</p>	<ul style="list-style-type: none"> <li>Colours used to differentiate different game objects.</li> <li>Different texture symbolising cut.</li> </ul> <ul style="list-style-type: none"> <li>Box appears with a cut out which is playable window.</li> <li>Slide to cut and move objects.</li> <li>Able to go in the box to see different layers of game.</li> <li>Bright colours for enjoyable visual experience.</li> </ul>

Titles in *Table 3* were chosen as they look similar to the game that is being implemented in this project. Common features that are implemented in those titles are: bright and distinguishable colours, simplistic and clear UI, clear game objectives and interaction. In the

implementation of this project, those features were taken into the account, affecting how game was implemented. As by looking at existing titles, it illustrates that those are features that work well and provide satisfying UX.

### 3.4. Summary

By completing this research, in summary, it did have an effect on how the project is being approached and is being implemented. From the research, the visual feedback for the user is very important, followed by the sound and the UI, all of which were important features that are focused on in the project. Also, using 3D objects with potential motion for greater recognition for human eye. Using bright colours, to distinguish game world from the real world, combining it with the auditory output for best possible UX. Apart from the output conclusion, the input was also considered, however due to the nature of this AR game and analysis the input options from the research paper, it did make most sense to implement visual and tactile inputs. Combining this input combination, with the outputs, and comparing existing titles, it should provide AR game project with satisfying UX.

## 4. Method of Approach

### 4.1. Approach

The production of the prototypes and tests will follow a Scrum, Agile Methodology [23].

By the use of User Story Mapping, the sprints will be created to set deadlines and have a clear direction what is being implemented at each stage. Each sprint will have specified information with functionality and how it affects the game.

After the sprints are specified, the prototype will be created, followed by quality assurance. Quality assurance will test out the prototype to see if it is suitable for use. Proof of concept will be performed afterwards, which will involve demonstration of the prototype to interested parties and project supervisor to get feedback and make sure prototype is performing as it should.

This method of approach will be repeated, once the technology is successful in testing. The process will be repeated until all agreed features are created. Followed by testing from colleagues and have public testing, where the objective being to get as much feedback as possible.

Reasoning for using Scrum, Agile methodology [23] is the organisation that it provides in the amount of time provided. It offers sprints which have clear goal of what is being achieved, with an option to adjust after each sprint. Having organised sprints, with specified deadlines for each sprint, does help with limited time that this project is under, and it sets clear goals before start of implementation of what should be a focus. Another reason for choosing scrum, agile methodology is familiarity.

Technologies that were used for this project are specified in section 4.2.

#### 4.2. Technologies

Unity 2017.3 [8] – Game Engine

Visual Studio 2017 [9] – Script Editing Software

Adobe Photoshop CC [10] – Graphical Editing Software

macOS High Sierra, Version 10.13.4 [11] – OS of machine

Xcode 9.3 [12] – Install program for iOS

Unity ARKit Plugin, version 1.0.14 [13] – Package used to implement ARKit in Unity

iOS 11, iPhone 7 [14] – Device it was tested and implemented on.

AR game can be implemented using variety different SDK, Kudan [15], Vuforia [16], ARCore [17], and many others, however since goal of this project was to implement good UX experience, focus was on creating a game without a marker. At this moment, ARKit [18] and ARCore offer top quality tracking, and due to having access to only iPhone 7 device, ARKit was chosen as a technology to implement with. Other AR developer kits require to be implemented with a marker, and the game experience is limited due to the marker.

With chosen technology, there were requirements to implement with ARKit, those are reasons for implementing on macOS, having MacBook Pro available to implement on, and iMac in classroom as a back-up. Xcode was required to build on and test project on iPhone device, and using ARKit Unity toolkit to build the game in Unity. Chosen Unity due to familiarity with C# and Unity game engine. Other option was to implement in Xcode, however learning Swift would be required, and with the time limitation there was not enough time to learn it to complete project to high quality.

Adobe Photoshop was used for graphical assets, chosen for familiarity and availability.

#### 4.3. Project Management

Trello is the project management online tool that is used to keep track of the project – that is where the sprints are detailed. Trello is used as it offers to create category cards under tasks fall under. This is very useful for organisation purposes, as it can contain cards like “To do”, “In process”, “Complete”, which assist in organising the project progress. It also contains feature of assigning deadlines to each task, which is another reason why it is management tool of my choice. Tasks could also be colour coded, which is also a good organising tool as it may be used to create task priorities. There are other tools available, such as Kanban, which offers similar services, however, Trello was the management tool of choice due to familiarity as it was used for previous projects previously.

GitHub is the version control tool used for this project. GitHub keeps track of every commit that was made, with time and date, and also what was changed on each commit. This is useful feature to see how project has developed, but also a history track so if something ever breaks, it could always be reverted to previous, working version. GitHub also allows to download the saved project onto other working machine, and shared with other people,

which is useful when working in group and in multiple different workstations. GitHub also has branches, which could be merged with the master branch (main branch), which allows for separate feature development without disturbing the master branch functionality.

Weekly highlights are created to keep track of progress of the first 7 weeks of implementation. Daily developer diary contains daily entries with details of what has occurred on that implementation day.

## 5. Legal, Social, Ethical and Professional Issues

### *Copyright*

*"The Copyright, Designs and Patents Act 1988, is the current UK copyright law. It gives the creators of literary, dramatic, musical and artistic works the right to control the ways in which their material may be used. The rights cover: Broadcast and public performance, copying, adapting, issuing, renting and lending copies to the public. In many cases, the creator will also have the right to be identified as the author and to object to distortions of his work."* [19]

Assets, sounds, and anything else that is used in the project is protected by the copyright. Before using any resources created by others, the copyright was checked, with the author, to make sure that it is legal to be used for this project.

#### Copyrights for items used:

*Planets and projectiles* – Unity Asset Store states that assets that were taken from the Asset Store are able to be used in personal or non-commercial purposes, some even agree to be used for commercial purposes. [20]

*Sound, SFX* – Sound used in this project contain music for healthy and infected planet and projectile shot sound. All of the sounds are agreed to be used for non-commercial purposes. [21]

*Code for mesh deformation* – Able to distribute the code from the site for personal, non-commercial, commercial purposes. [22]

Since the Assets, Sounds, Logos and any other content created by others, are not used for commercial use, no copyright laws were broken in developing this project.

### *Ethical*

In this project, there were testing completed which involved other people. This is covered in University of Plymouth Ethical Form. Any participants that took part in testing, and provided feedback, signed the Ethical form to give consent to use their feedback in this project research.

### *Social and Professional*

With the use of Augmented Reality, AR, and with this game design, it might be necessary to have spacious environment to be able to use and test the game. It can cause issues when playing this game in public, since it might cross with other individuals personal space. Therefore, it is recommended to play this game where plenty of free space is available. In the development of this project, no personal space was invaded, and at all times the project was tested and developed on, there was plenty of empty space available which did not impact other people movement or space.

## 6. Implementation

### 6.1. Process

Initially, organisation was completed on Trello, to keep sprints organised and to have a clear idea what was being accomplished on each sprint. That is also a place where all feedback from Highlights, and any comments to take in to consideration were dropped down, followed by a change in sprint, or future plans, if necessary.

The idea was created by looking at already existing titles, in AR, mobile and computer games. Based on research, the game idea was created, which was split into sprints. The idea evolved with each sprint and feedback, to create more relevant game, and at the same time created a more enjoyable experience for the user.

Following the sprints, it took five sprints to create a playable prototype, which was followed by a testing session with colleagues. At that session, the feedback was gathered, and after analysis the feedback, the UX was not that well implemented, as the game required the user to do extra steps to complete a task, such as to collect ingredient or to activate ability. With that feedback taken into the account, the next sprint was altered to change the game, regarding UI and how the information is presented to the player, with feedback being more positive after the sprint.

Final implementation shows more features of what the current titles in AR are representing, with the more enjoyable user experience by the way that the information is presented and shown to the player. UI is clean, to have as much viewing screen as possible; the player has an option to view instructions how to play the game by clicking help button, this doesn't enforce tutorial on the player if they don't want it; sound is used as atmosphere representation, representing the healthy and infected planet, as well as when infection takes place.

Each sprint was implemented with iteration of different implementations and trying different way of implementing a feature, for example the spread of infection, or the abilities functionality. The implementation that worked the best, and was optimal that work with the rest of the system was kept, and once this was implemented the next feature was being worked on. This project didn't allow for much time to work on graphical implementation of the game, therefore the assets from Unity Assets Store, and online assets, were used to implement the game. All copyrights were checked to make sure it was within copyright to use the assets for this project. Adobe Photoshop was used to create any graphics that I was required to make.

*The following show the outcomes after each sprint:*

First sprint outcome, which involved testing the technology out, was to carry out research, which might affect the development of the project. There were no delays in this sprint and everything was working as intended.

Second outcome involved implementing prototype of flying planets, with an option for a character to jump between them. Sprint was successful, completing the goal that the sprint set out to have, with the feedback to consider depth perception, potentially shaders, and sound to represent the state of the game. After this sprint, there were no changes made to following sprints.

Third sprint outcome was the planet system with infection spreading across planets. It had a solid line between, where the feedback to make it more dynamic, for person to be able to see the progression of infection. Also, this is first time where the game orientation had to be taken into the account, and testing for the orientation was added for the next sprint to identify which orientation the game should be based on – landscape or portrait.

Fourth sprint involved completing the prototype to be ready for the testing to be completed on the next sprint. The outcome involved orientation testing to be completed and test results indicating for the game to be implemented in landscape position. This test was done by giving game to people to try out and monitor in what orientation they play the game. Also, all menus were implemented for the game, so test participants had a play-able game.

Fifth sprint involved to have testing complete, followed by adjusting sprints after to take into the account the feedback obtained at the testing session. Outcome for this sprint was that the game involved too many menus, with players taking extra, unnecessary, steps to do an action. Also, lack of sound impacted on the enjoyment of the game. Based on the feedback, the game was re-created, adjusting following sprints to involve cleaner UI, and researching more on the UX and AR, to implement more enjoyable UX experience for the player, while keeping the game mechanics the same. [APPENDIX H]

Sixth sprint was improvements implementation towards the game. Outcome was more positive, compared to the initial implementation, having more enjoyable UX experience, while at the same having better gameplay. Feedback gathered was taken into the account taken from the testing session and was implemented in this sprint.

## 6.2. Requirements

Minimum software and hardware requirement:

- iOS11
- Xcode 9
- Apple A9 processor (iPhone 6S/SE and newer, iPad 5<sup>th</sup> Generation and newer)
- macOS High Sierra

Project requirements:

For the project, the requirement is to create enjoyable UX experience. That involves creating visually pleasing and easy to understand gameplay. This includes information that is being presented to the player in understandable and logical manner. Goal being for the player to have a feeling of accomplishment, enjoyment, when playing the game and after the game. This is achieved by the way that UI is implemented, the sounds, and the game visuals. For this project, the goal is to implement clean, minimalistic, UI, with good introduction towards game mechanics, combining with well-designed visuals and sounds.

### 6.3. Design

Design of the game evolved with each implementation of the game. The core mechanic of the game, infection spreading across the planets with player goal is to stop it, has stayed the same, however, the way that it is implemented has evolved.

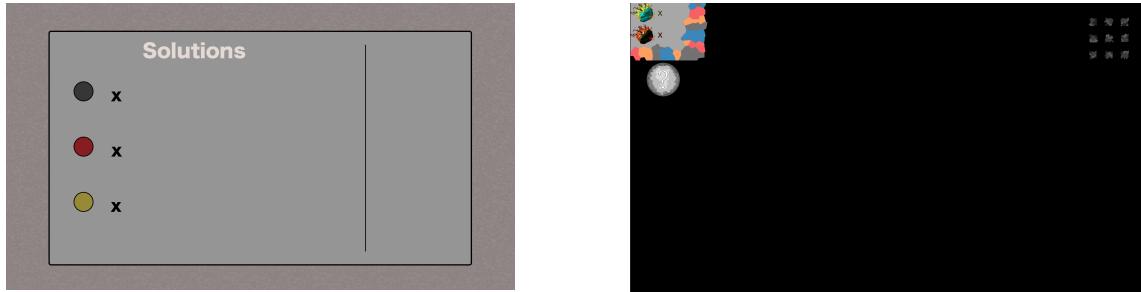


Figure 4: Old game design (on the left), and developed to new design (on right).

Initially the game was implemented very inefficiently, where instructions were only present at the beginning of the game and no way of viewing them again once the game has started. Also, the game was filled with menus. Menus were present to do any action in the game, when the player clicked on the planet, or the pause screen (which was also an inventory screen as seen in *Figure 4*). This did not feel like a pleasant implementation, and also from feedback the players didn't feel like they know what is going on and did not have a good user experience.

This design has evolved to contain minimum number of menus, while still presenting all the information to the player in different manner, little design implementation improvements can be seen in *Figure 4*. The way that the game instructions are presented to the player has also changed. The player starts to play the game normally, with a help button on the side which shows the instructions of how the game operates. This allows the player to reference back to the help whenever they need to, while at the same time the help, tutorial, screen is not forced on the player when they start the game.

The tutorial screen is available to play through, which is optional and it is implemented with timed events which introduce the player to the game mechanics slowly.

The UI contains help button and the pause button, which is clean and maximises the amount of screen available for the gameplay. The game itself shows abilities using particles and prefabs rotating around the planet, which was previously implemented as a text information in a menu. The colours of particles change depending what ability is available to take. Also, the initial platform has more use, as it will hold the available ability that the player can click on to activate.

Planets have different texture which indicate the state that the planet is at, where it is colourful at the healthy state, and red shade when it is infected. This creates a visual representation to the player, which allows player to know what is going on in the game without having any input. Previously most information was presented in the menu system once the planet was clicked on. Moreover, to improve the feedback that game has with the user, once the player has click the planet, the planet now deforms which gives feedback to the player that the planet was interacted with.

Sound was also implemented to indicate the state of the planet, which player can hear depending on the position that they are at. Sound ques are also created to indicate when planet is infected. This improved the playability and user experience, which is also visible on the feedback, compared to the initial implementations where the sound was non-existent.

Assets that are used in the game also improved, as with the time limitation and how much time it takes to implement the game, the assets from the Assets Store were used to improve the graphical aspect of the game. This has improved the visuals greatly.

#### 6.4. Prototype

Prototypes were created to test out the implementation, mechanic, and features in the game. Each sprint had a prototype created, and after sprint five the prototypes have created a playable game experience. Upon that prototype, the game was improved. Each sprint followed this formula: Minimum Viable Product (MVP), Testing, Results from the testing.

##### 6.4.1. Minimum Viable Product

Minimum Viable Product was the minimum implementation of the sprint before completing the sprint. In this project, the MVP was used to create a basic form of the mechanic, the game, but it didn't necessarily contribute in creating an enjoyable gameplay experience and a good user experience. Initial implementation was focused more on implementing the mechanic of the game, however as the project is focused around the user experience, later in the project the focus changed to implementing more enjoyable user experience rather than the mechanic, although initial implementations did allow to test the mechanics of the game.

The MVP for sprint one, sprint specified in section 7.1, was to have a game design that the project will be based on, and at the same time test out the technology, ARKit with Unity. The game design was an idea that was developed further, and changed, throughout the project. Core of game idea stay the same which was to have islands and infection spreads through them, corrupting the islands and the player has to stop it. The initial idea did consist of additional character in the game that the player controls. The technology testing was a tracking test, to see how well ARKit tracks and what could be potentially implemented with this technology. Alongside this, the organisation of time, deadlines, and rough plan of the project was created.

MVP for sprint two, sprint specified in section 7.2, was to create a prototype of islands spawning, and user interaction with the game. The character was implemented in the game, which provided feedback to the user if the character is in correct position (planet), based on where the user has clicked. Basic meshes were used for the implementation and textures to save time. This MVP tested out the islands spread and tracking and player input movement between the planets.

Sprint three MVP, sprint specified in section 7.3, was to implement the infection, the virus, that would spread towards other planets. With an indicator of infected planets. This was implemented using different textures, representing infected planets in red colour, and having a solid line link between the planets, also in red, showing the spread directions. The basic UI, pause screen, was implemented. Start of ingredients spawning on planets, and inventory system was to be started in this sprint too.

Sprint four MVP, sprint specified in section 7.4, focused on completing abilities, with inventory system, to be ready for testing and to gather feedback. This involved ingredient system working, with the solutions created from the ingredients will be placed on the planets to fight the infection. Inventory system was implemented in the pause screen, to save screen space.

Sprint five MVP, sprint specified in section 7.5, was focused on testing the application between colleagues and gather feedback on the game. Analysis of feedback to be completed, and based on feedback alter the following sprints to include the feedback gathered.

Sprint six MVP, sprint specified in section 7.6, focused on altering the game in any way from the feedback gathered that would create a more enjoyable game. This involves improving graphics, UI, and gameplay, even mechanics were to be changed if it required. This sprint focused on creating best possible game with the time provided which focuses on user experience and delivering positive user experience.

#### 6.4.2. Testing

Testing was done after each sprint, and whenever new mechanic or feature was implemented. It was tested on a development device, iPhone 7 (mentioned in section 4.2.), fixing any bugs that were encountered. Major testing was done after sprint 5, which tested the whole prototype with other people gathering feedback while at the same time good moment to monitor any bugs that were encountered to be fixed later.

Apart from bug testing, another reason for testing was to test if the feature implemented works well with the game. This was particularly tested while public testing took place, where feedback gathered reflected the features implemented, and how it could be improved or changed.

Having feedback from both testing sessions [APPENDIX H], suggest that second implementation does offer more enjoyable experience and is clearer in information presentation to the player. There are still bugs that do occur, which was pointed out in both feedbacks, however, apart from that the experience is positive. Distinguishable colours, particles, sounds, and game interaction overall gets a positive feedback, where it can still be improved in how the game presents game objectives once the level begins. Feedback from the first testing sessions were taken into account and implementation was changed to apply those feedback, making UI simpler, where it doesn't require extra unnecessary steps, alongside other feedback. This can be seen in the second feedback form, that experience is more enjoyable and positive, suggesting that the improvements were successful.

#### 6.4.3. Results

Results from testing were evaluated and based on the results the actions took place. Testing took place to get results for the bugs that were in the game but also on the features implemented and how well do they operate in the game. The feedback gathered where the results from the testing and the results showed what features worked well and which should be modified to improve user experience.

After sprint testing, the bugs that were found were fixed, and they varied, which can be seen in developer diary (Appendix G). Public testing indicated results for the features and gameplay, which showed that the UI had to be changed to be more logical and didn't require player to do unnecessary steps to complete an action. The ingredient system wasn't that enjoyable, which was changed towards abilities, and how the information is represented in the game was to be changed to be more attractive to the user. Also, implement the sound to enhance user experience. Based on those feedback, the game was changed.

## 6.5. Control Plan

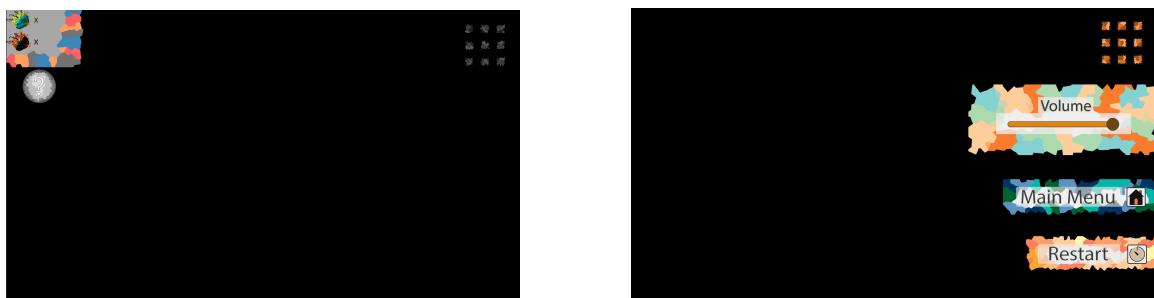
Within the user experience, the features that were controlled were: User Interface, Game Interaction, and Visuals. All of which impact the user experience and those were changed to enhance the experience for the user while keeping the core mechanic of the game the same.

### 6.5.1. User Interface

User Interface, UI, is important in AR, as it impacts how the user interacts with the game, and it can also impact how the information is presented to the player. The UI for this project was influenced by the research of already existing titles of AR and other technologies. Existing titles comparison table is visible in section 3.3.

Initially the UI was implemented to have many menus, with screen UI to be clean, only pause button being displayed on the screen at all times. The pause menu would be combined with the inventory system to minimize how much space UI takes on the screen. The menus for planets would display next to the planet to indicate the information about the planet. However, after testing, this implementation produced poor user experience, with feedback gathered stating that user had to do extra steps to complete basic task, where the steps to complete the task could be completed with one or more less actions.

Upon feedback, the final implementation involved game to be more visual and uses visual assets to show information for the player, with help (tutorial) and pause button living on the screen UI at all times. This implementation gathers more positive feedback delivering better user experience. *Figure 5* illustrating few implementations of UI.



*Figure 5: Gameplay UI (on the left), and pause menu (on the right). Simplistic and simple design that allows quick access for the user.*

### 6.5.2. Game Interaction

Game Interaction focuses on how the game interacts with the user, the player. This involves the player input, how the game presents data for the player, the feedback provided based on the input.

Initially, the game was implemented with introduction text at the beginning of the game, where feedback was not that positive towards that approach [APPENDIX H]. Upon that

feedback, the design of the game was altered, to introduce the mechanics to the player in more interactive manner, which did gather more positive feedback [APPENDIX H], and that is the direction that the game was implemented further.

In this project, game, the game interacts with the player by slowly introducing game features to the player, while still allowing players to view how to play the game by pressing help button. There is also visual feedback to the player, opposed to text or menu, which a player can view to see a state of the game. Also, there is visual feedback in the form of mesh deformation that takes place once the player clicked on the planet. There is also sound implemented, which is a way for a game to indicate to the player when there is a danger while also being used for atmosphere creation.

Player can only interact with the game by the use of a click, and the abilities that are in the game. With all of those game interactions implemented, it creates an enjoyable user experience, based on the feedback.

#### 6.5.3. Visuals

Visuals are final feature that can be controlled to impact the user experience. Initially, basic meshes with basic colours were used in the implementation of the game. As initial plan was to create assets in Blender. However, due to time constrain, assets were gathered from Assets Store. Also, abilities are represented in the game in more appropriate manner, using prefabs and particles which enhance the game appearance. This is also how the game represents some information towards a player, which has more positive feedback unlike the initial representation where it didn't utilize visual aspect that well. The game uses visuals to show the player the state of the game, abilities available to be gathered, and danger incoming to the planet. With visuals, sound is used, to create more engaging and enhanced user experience.

## 7. Stages

The following stages are the milestones, sprints, that were created to complete this project. They were used to keep the project organised, and helped to make sure that project will be completed on time.

### 7.1. Stage 1: Game Design & Technology Analysis

First stage involves project planning and creating a draft of sprints that will be used in development of the project. This was done on Trello, by creating a series of sprints that are milestones for the project. Each sprint took into the account the knowledge of the subject and what had to be completed to make significant progress. The sprints were to be edited when one sprint was completed, to keep the project plans up to date.

That is also the initial stage where the research has started, looking at existing titles that can influence the outcome of this project. Research of literature also started at this time, which carried on throughout the project, taking notes on Trello any relevant literature that is relevant to this project.

First stage was also dedicated on figuring out and creating a game design that will be used for the game to be implemented in this project. This design was creating by looking at existing titles and gather up ideas of what game mechanics would work well with AR.

Game design that was the project based on was on creating a game where the player specifies a spawn location. Once the player decides spawn location, the virtual world (planets) are spawn spread apart of each other. Each planet containing an ingredient of some kind to find evil that will appear in the world. In this game, the evil is the infection that spreads across planets and ingredients are the only way of preventing the spread, and fight the infection. Winning condition being get rid of the infection, and losing condition is where all planets are infected. In this game design, the player is forced to move around, taking advantage of ARKit, while it also allows to test different UX implementation on which this project is focused on.

Upon deciding what game will be implemented, the prototype of ARKit plugin working was implementing, main goal being to test out how the markerless implementation works and how well does it track the virtual world. Previously done projects in Vuforia, where the marker had to be used to play, the ARKit markerless implementation offers better tracking without any additional components. This allows for bigger range gameplay, where the camera doesn't have to view a marker.

With ARKit, the initial prototype was very simple, having specify and spawn location (world centre), followed by the world spawning around the point. To spawn a location, the horizontal plane has to be scanned, where the code for that is already present in the ARKit

Unity Plugin. Upon completion, the tracking was good, where player could view different sides of the world without losing track. Having fundamentals in place, the game prototypes and implementation could begin.

### 7.2. Stage 2: Game Implementation

Second stage that was set for the project was to begin game implementation. It involves implementing the user input, and the virtual world spawn. It was accomplished with taking initial prototype that was used to test out the technology, which then was improved by spawning multiple planets. With the world spawned, that doesn't collide with each other, the user input testing was implemented. It was done by creating a character that jumps to a planet that the player has clicked on.

The character was later in the testing and implementation deleted, as it is unnecessary for this game design. However, the character implementation did assist in making sure that the game interprets the player input correctly, as there was visual character jump feedback to the planet that the player has clicked on.

This also tested out the functions that were used for the implementation, fixing any bugs that may have appeared along the way. Knowing that the functions work as intended, once the character was removed from the implementation, those functions were reused to interpret player input.

### 7.3. Stage 3: Game Mechanics

In the Stage Third of the project, the game mechanics were implemented. The main mechanic, the infection spread, was implemented initially with the planets being connected together by a link, in game it is represented with a prefab that is rescaled in the correct directions. This creates a link between the planets, with a visual red colour implementation that symbolises danger. At this point, the character was still implemented, and once the spread reaches the game character, the player loses the game. This game design was a bit changed later after implementing and testing the whole prototype. As the character was completely removed.

Alongside the spread, the winning condition was also implemented, the ingredients, that could be gathered to fight the infection. Originally the idea was to gather the ingredients from planets and once the player has enough ingredients, the player is able to construct one of three abilities. Initially there was a bomb, permanent health link, and temporary planet shield.

Initially, the ingredients system was implemented in UI that takes a lot of screen space, and adds additional stages for the player to do which were at the end unnecessary. Once the player clicks the planet, the menu system would be displayed next to the planet which

would show what player is allowed to do on the chosen planet. The pause menu was the place where the ingredients information was stored.

#### 7.4. Stage 4: Complete Game Prototype

Stage Four was dedicated on completing the initial prototype to be ready to be used for public testing. The prototype would involve a minimum viable product – UI, working game mechanics (spread, ingredients, crafts system, win/lose scenario). It did also contain introductory instructions, which described everything in the game. However, it was clear that without describing the game, the instructions were unclear.

The graphics for the prototype were very simplistic, as there was no time for complex asset creation, or any animations. Therefore, it didn't look very inviting and was very basic. There were also few bugs that still existed which were not fixed until the next stage. However, for testing and prototyping, it did display the mechanics and how the game would work, with initial UX design with plenty of menus.

#### 7.5. Stage 5: Testing & Feedback Gathering

Stage Five was testing and gathering feedback. Google Forms were used to gather the necessary feedback, with an option to analyse the results at the end. The goal of having minimum of 10 people testing out the game, also signing the ethical approval form.

In testing, the game was presented in front of the testers, where every person played the game, followed by filling out feedback on google forms. The feedback gathered were similar. Feedback being that all the menus were unnecessary where the player had to make additional clicks, moves, which could have been done quicker, with cleverer implementation. Moreover, the feedback also involved the character is not necessary, as well as the instructions at the beginning were not very clear and the visual representation of information could be easier represented.

As it is project about UX, it is important that the information is presented to the player is enjoyable manner, and the gameplay is logical. It involves the introductory segment at the beginning of the game, tutorial, that present that player to the game mechanics and gets familiar with the game. Next stage focused on improving the game prototype, changing game design accordingly, improving the UX in the game.

#### 7.6. Stage 6: Improvements Implementation

Stage Six was focused on improving on prototype that was created. The improvements that were implemented from the initial prototype are as follow.

- UI is simplified, clean, on the screen only showing the pause button, and an indicator on how many planets are healthy and how many are infected. No other UI is

required for this type of game, as from testing and feedback gathered the new implementation uses more of AR technology.

- The character on the planets was removed, as it is unnecessary for this sort of a game. The camera position, the player, is now the only player that plays the game, no other characters to control. This, like with the UI, has good feedback, saying that it uses AR technology more and it is more enjoyable to play.
- There are no more menus next to the planets to indicate the possible actions for the player. The player is able to click on the planets to collect the ability, and use the ability on a specific planet, but apart from that there is no menus. The abilities are indicated by the use of orbiting prefab, with different ability showing different effects. This is more visibly pleasing, also visible to the player without doing any actions. With this implementation, no extra menus are required, and controls are simple and logical.
- There is a mesh deformation applied to the planet when the planet is clicked, for visual feedback, and to enhance user experience.
- Music is added for better engagement with the player and to make the game more atmospheric. Music is used also as an indicator to indicate the distance between objects, as the sounds become quieter based on the distance.
- The planets, and projectiles, are now prefabs taken from the Assets Store, which improve the visuals of the whole game. Overall, particles, improved graphics and animations are used to enhance visual aspect of the game.
- No more introductory instructions at the beginning of the game, it is replaced with timed sequences which present the game to the player. Like a tutorial. It presents different mechanics of the game to the player in the more interactive manner.

All of those improvements did equal to improved feedback, and better user experience in general. That is what this project focuses on.

## 8. External Libraries

Following external library was used in this project: Unity ARKit Plugin version 1.0.14 [13]. This library was used to use ARKit in Unity and build the project, followed by compiling and installing it on iOS developer device.

## 9. Project Evaluations

### 9.1. End Project Report

After the project had reached its conclusion, AR game has been created which satisfies most of the project objectives that were laid out in the Project Initiation Document (PID). AR game is created which illustrates the system possibilities, with satisfying UX. The implementation includes clean, simple UI system, with the optional introduction which introduces a player to the game mechanics. Actions in the game are logical, and simple, with visually pleasing and well-defined depth perspective, for the viewer to distinguish different game objects.

The project was kept to the original objectives, without any major alterations. Within the project, how the game is to be implemented was adjusted to fit the project objective, but the core of the project stayed the same throughout the implementation. The expectations, or possibilities of AR technology might have been tested further, as this project focuses more towards UX in AR rather than the possibilities with AR system. However, possibilities are still presented, by how the game is implemented, with marker-less implementation and game interactions.

At the initial stages of the project, more research could have been done, when the implementation was being organised into sprints. This would allow more time to implement a more polished out game, as the game implementation was adjusted, changed, after week five to fit the objective more. This has occurred due to lack of research completed beforehand, and miss-understanding of what would work well in this system. Even with those issues, the project still manages to fit the objective criteria and completed a product that meets objectives.

The project was implemented, like intended, on macOS, for iPhone, iOS devices. Unity and C# was used to implement the game, by using ARKit Unity Plugin to convert the game to ARKit game and install it using Xcode on the iPhone device. This was planned initially in the PID document and did not have to change throughout the project.

### 9.2. Project Post Mortem

Overall, the project was completed successfully, meeting project objectives, without any major step-backs. The use of agile methodology was used in advantage for this project, to

keep the progress organised and also manage the workload throughout. With the organisation of sprints, there was a clear goal for the sprint that was going to be achieved, which helped with the overall progress of the project. The software and technology usage was also successful, even though time was spent at the beginning of the project dedicated mainly on learning how to operate with the ARKit technology.

Even though the project met the objectives, there were some issues along a way that could have been resolved if organisation was better at the start of the project. The project implementation was changed in the middle of the implementation, which was due to lack of research, and lack of game planning. That created a situation where the prototype was implemented with incorrect features in it, causing the whole project not to meet the objective. This forced a situation where there was very little time to re-create and re-build whole prototype, with the improved features, with very small amount of time. How it could have been avoided is by having done proper research and plan the application before the implementation, creating prototypes and gather feedback. This would delay the start implementation a little bit, but overall the time spend on the implementation would have been longer, as in this project some implementation time was wasted due to change of implementation in the mid-way of the project, once the research was completed. This was the main lesson learnt, if this project was to be done again.

There were no issues with the software, apart from learning ARKit at the beginning of the project, however the organisation took that into the account, by having simpler mechanics and features to be implemented initially. This allowed time to learn ARKit toolkit in Unity. It allowed the project to be implemented in the Unity game engine. The chosen technologies were correct choice, as it did deliver, and due to familiarities with Unity, learning and implementation process was without any major issues. However, the testing of new development took a while as it required to be built from Unity, followed by project build on Xcode onto iPhone device which took long time to test a feature out. Especially if there was a bug that had to be fixed, and project had to be re-tested.

Organisation was well implemented, apart from the initial research, as sprints were planned out for the whole project, keeping track on the management tool. The sprint deadlines were always met, with few exceptions that can be justified. After the sprint was completed, the following sprints were evaluated to see if any chances would have to be done to it, this was primary done after public testing was complete. Also, developer diary was kept up to date, which had entries whenever any implementation took place, which was useful in referring back on the implementation stages and decision choices that were made. The chosen strategy of approach this project was well chosen, using agile methodology, as it created organised workflow.

### 9.3. Objective Evaluations

Objectives in the projects were met, and evaluated by the feedback received when testing the game. Main objective for the project was in creating AR game application, with well implemented UX. This involved having game that interacted with the player, creating enjoyable experience for the player. This was achieved by the way that UI is implemented, and how player interacts with the game that is in logical manner. Visuals and sounds are also used to assist good UX implementation.

For this project, clean UI, with tap screen was implemented for interaction with the game. Visual particles, and assets were used to show the game state to the player, using distinguishable colours and particles to represent the game. Sound was also used in assisting to show the state of the game to the player. Game mechanics are presented to the player one by one, in visual form, which introduces the game to the player gradually, one aspect that is important in good UX. However, the gameplay is not always smooth, as there are occasions where the game doesn't work as intended and it requires to restart a level to work properly, which does affect the UX experience.

Combining all of those features, the game implemented is believed to contain suitable UX for the application created. This could be seen from the two test cases that were completed, one prototype showing badly implemented UX, and final result which contain well implemented UX system. Having positive feedback gathered for the AR game, suggests that the project has successfully completed the project objectives.

## 10. Conclusions

### 10.1. Lessons Learnt

This project was a learning experience, improving knowledge with programming in C#, and using Unity game engine, having minimal experience in using C# and Unity before.

Application and software development skills were also improved, as this is project where application, in this project a game, is created from design stage towards a functional game.

Apart from improving current skills, new skills and knowledge was learned, working with ARKit toolkit to implement game in AR, having no prior experience or knowledge with the technology beforehand. Moreover, focusing project on UX implementation, which required research and improving UX knowledge in creating a successful game, which required to learn more about UX.

With the strict deadlines, this project also required good organisation skills, and required to be have good time management. Both of which were improved when working on this project, keeping to the deadlines that were set throughout the project for the project deadlines, but also the sprint deadlines.

Overall, this project did offer many opportunities to learn new skills, which was highly enjoyable. Apart from learning new things, improving on the current skillset is also important and it was also accomplished in this project, and it was required for the project to achieve the set objectives.

### 10.2. Further Development

If this project was being continued further, there would be some changes which were sacrificed due to time limitations. Primary further development that would take place would be to increase level complexity, as at the moment there is only one level that is available to play. Following the level implementation, the progression system would be implemented, which involves unlocking abilities with levels completed. With the progression, the rewards system would also be implemented, in the form of a star system as many other mobile games use.

Apart from adding features and mechanics, some things that would be changed are the assets used in the game, which suit a theme, with shader interaction when the player clicks on the planet instead of the mesh deformation. Also, menu system would be implemented with full features, and graphically pleasing. Removing all bugs would also be a priority in further development of this project.

Another feature that could be implemented in the future, is to implement sound effects whenever ability was used, potentially with visual feedback, for the sense of emersion and improved user experience.

### 10.3. Conclusion

In conclusion, this project was successful, with well-designed UX in AR game. This was successful with well organised project milestones, using scrum, agile methodology, which organised project with milestones to meet. Testing and gathering feedback was also important, as it did affect the implementation of the game. To avoid issues in this project, research would be completed earlier, to avoid miss-understanding on the project objectives. This project did provide insights into UX and AR game development.

## 11. References

1. S. L. Ames, J. S. Wolffgohn, and N. A. McBrien. The development of a symptom questionnaire for assessing Virtual Reality viewing using a head-mounted display. *Op-tometry and Vision Science*, 82(3):168–176, March 2005.
2. Panagiotis D. Ritsos, Dennis P. Ritsos, Antonis S. Gougiulis (2011). Standards for Augmented Reality: a User Experience perspective. [online] Available at: <http://pdritsos.com/files/Ritsos-et-al-4th-AR-Standards.pdf>
3. Colin Ware, Peter Mitchell. Visualizing Graphs in Three Dimensions. [online] *Data Visualization Research Lab, University of New Hampshire*. Available at: [http://ccom.unh.edu/sites/default/files/publications/Ware\\_08\\_TAP\\_Visualizing\\_graphs\\_in\\_3D.pdf](http://ccom.unh.edu/sites/default/files/publications/Ware_08_TAP_Visualizing_graphs_in_3D.pdf)
4. App Store. (2018). *AR Dragon on the App Store*. [online] Available at: <https://itunes.apple.com/us/app/ar-dragon/id1270046606?mt=8>
5. Dan1\*[one] (2018). *WordInvaders ARKit Demo*. [online] Available at: [https://www.youtube.com/watch?v=sJR\\_f7XCuvA](https://www.youtube.com/watch?v=sJR_f7XCuvA)
6. App Store (2018). *ARise on the App Store*. [online] Available at: <https://itunes.apple.com/us/app/arise/id1281550152?mt=8>
7. App Store (2018). *Splitter Critters on the App Store*. [online] Available at: <https://itunes.apple.com/us/app/splitter-critters/id1107320261?mt=8>
8. Unity3D (2018). *Unity 3D Game Engine*. [online] Available at: <https://unity3d.com>
9. Visual Studio (2018). *Visual Studio*. [online] Available at: <https://www.visualstudio.com>
10. Adobe CC (2018). *Adobe Photoshop CC*. [online] Available at: <https://www.adobe.com/uk/products/photoshop.html>
11. Apple (2018). *macOS High Sierra for Mac*. [online] Available at: <https://www.apple.com/uk/macos/high-sierra/>
12. Xcode (2018). *Xcode 9*. [online] Available at: <https://developer.apple.com/xcode/>
13. Unity Asset Store (2018). *Unity ARKit Plugin*. [online] Available at: <https://assetstore.unity.com/packages/essentials/tutorial-projects/unity-arkit-plugin-92515>
14. Apple (2018). *iOS 11 for iPhone*. [online] Available at: <https://www.apple.com/uk/ios/ios-11/>
15. Kudan (2018). *Kudan Computer Vision*. [online] Available at: <https://www.kudan.eu>
16. Vuforia (2018). *Vuforia*. [online] Available at: <https://www.vuforia.com>
17. ARCore (2018). *ARCore Overview*. [online] Available at: <https://developers.google.com/ar/discover/>
18. Apple (2018). *ARKit*. [online] Available at: <https://developer.apple.com/arkit/>
19. Service U. (2018). *P-01: UK Copyright Law fact sheet*. [online] Copyrightservice.co.uk. Available at: [https://www.copyrightservice.co.uk/copyright/p01\\_uk\\_copyright\\_law](https://www.copyrightservice.co.uk/copyright/p01_uk_copyright_law)

20. Unity3D (2018). *Asset Store Terms of Service and EULA*. [online] Available at: [https://unity3d.com/legal/as\\_terms](https://unity3d.com/legal/as_terms)
21. SoundBible (2018). *About SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/about.php>
22. Catlikecoding (2018). *Catlikecoding Tutorials Unity*. [online] Catlikecoding.com. Available at: <https://catlikecoding.com/unity/tutorials/>
23. Itinfo (2018). *Software Development Methodologies*. [online] Available at: <http://www.itinfo.am/eng/software-development-methodologies/>

Appendices:

Appendix A: PID

PID – Project Initiation Document

# Project Initiation Document

*“Investigating ARKit player-friendly experience”*

Ludwik Bacmaga  
10595124

## **Introduction**

Augmented Reality, AR, is becoming more popular in the recent months, with Apple releasing ARKit [3], and Google releasing ARCore [4], attracting developers to create and release new applications using AR. This technology is still being tested out with applications being developed in different genres, to find what works on AR and what doesn't, and which features take advantage of mix world reality.

There are other AR toolkits available, such as: Vuforia [5], Kudan [6], Wikitude [7], and many more, each one having their own advantages and benefits over the other, however for the purpose of this project the ARKit toolkit will be used.

The aim of this project is to improve my skills in the Augmented Reality field and to explore user experience, UX, within the AR. As ARKit is relatively new, there are features and concepts that haven't been fully explored yet; therefore, what this project aims to accomplish is to investigate the platform further by creating a series of tests which represent different user experiences. By the end, the plan being to create a playable demo of a game, which illustrates capabilities and good user experience on the platform.

## **Background**

Augmented Reality has been out for many years, however most of the time it required a marker to work properly and smoothly. One of the most popular games which does implement AR is "*Pokémon Go*" [8], where it used GPS to track the device location and alter the game based on where the device is.

This being a successful game [1], having over 6 million downloads on Android market alone, shows that there is a market that is interested in AR games on mobile. With the succession of this game, developers saw potential in AR market, and ever since the AR has grown in popularity [2].

ARKit is a new technology introduced by Apple in September 2017, that developers are able to use and develop with for series of Apple products that have A9 chipsets or newer and installed iOS11 [3]. It allows virtual reality to be mixed with real world, which in principle allows for different style applications and experiences for the user. It is different to other Augmented Reality technologies, as it relies on plane detection instead of markets to display the virtual world.

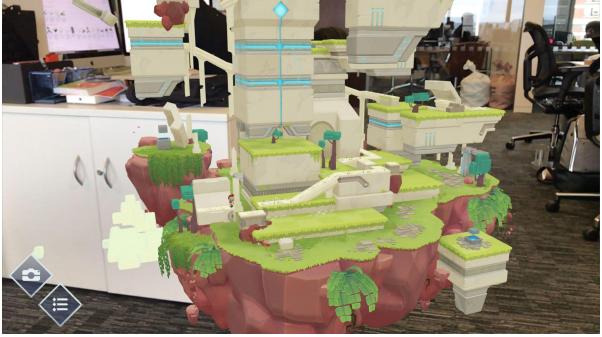
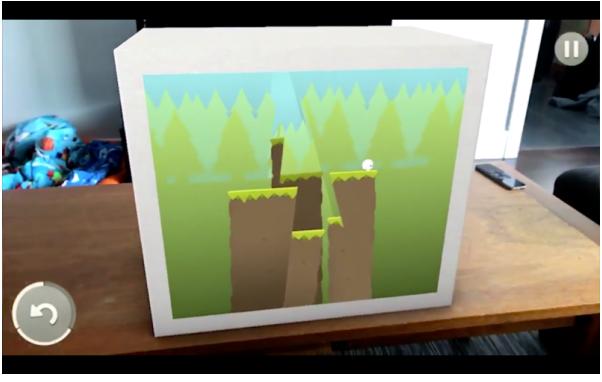
ARKit was released with some exclusive titles such as "*AR Dragon*", which is a fun game of taking care of a dragon. It uses real time mechanic, which impacts the dragon and makes it grow. This encourages you to come back and play, otherwise dragon will fly away and player has to start again. This application just shows what the platform is capable of delivering, and it demonstrates how it is different to the competitors.

There are additional exclusive games for ARKit: "*Stack AR*" [9], "*The Machines*" [10], "*ARise*" [11], "*Splitter Critters*" [12], and more. Those games are exclusive on iOS, implemented in AR, they don't require a marker, and they use ARKit tracking to work. All of those games have similarities, even if they are in different genres, such as that they have bare minimum

UI (except for *The Machines*), to show as much as possible on the screen. *Table 1* shows comparisons between the games and the features that they have in common.

**Table 1**

Game Title	UI	Interactables	Visuals
<b>AR Dragon</b>	<p>Minimalistic – indicative icons on top corners, and interactable buttons on bottom of the screen.</p>  <p>ARDragon gameplay image from:  <a href="http://ostatus.org/wp-content/uploads/2017/09/AR-Dragon-Hack-Cheats-Tips-Guide-2.jpg">http://ostatus.org/wp-content/uploads/2017/09/AR-Dragon-Hack-Cheats-Tips-Guide-2.jpg</a></p>	<ul style="list-style-type: none"> <li>&gt; Tap the screen, or drag to interact with dragon</li> <li>&gt; Different objects, food, games available between dragon and player accessible through buttons on UI</li> <li>&gt; On screen buttons for taking picture, menu button</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Able to be positioned anywhere</li> <li>&gt; Dragon will look at the position of the camera, giving impression of looking at you</li> <li>&gt; Will change appearance with age - get bigger</li> <li>&gt; Shows different reactions depending on mood</li> <li>&gt; Collectibles and objects spawn around the environment to collect, clean up, play with</li> </ul>
<b>Stack AR</b>	<p>Minimalistic – only showing indicative numbers to symbolise progress.</p>  <p>Stack AR gameplay image from:  <a href="http://images.indianexpress.com/2017/09/apple_ios11_stack_ar.jpg">http://images.indianexpress.com/2017/09/apple_ios11_stack_ar.jpg</a></p>	<ul style="list-style-type: none"> <li>&gt; Tap screen to play game</li> <li>&gt; On screen menu buttons which are visible after losing the game</li> <li>&gt; On screen directives, in the form of UI, to display game progress</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Square spawn on each new level of the stack</li> <li>&gt; If player doesn't align the squares perfectly, the remainder gets cut off and falls down to the ground</li> <li>&gt; Square colour changes depending on the stack level</li> </ul>

<b>The Machine s</b>	<p>Medium – There exists a whole UI in front of the game screen, which is interactable by the player, and the main way of playing the game. As it is strategy game, it is required for the genre to play efficiently, however it feels packed when compared to other titles.</p>  <p>The Machine gameplay image from:  <a href="https://i.ytimg.com/vi/1aNpmZ078Ac/maxresdefault.jpg">https://i.ytimg.com/vi/1aNpmZ078Ac/maxresdefault.jpg</a></p>	<ul style="list-style-type: none"> <li>&gt; Tap on screen, menus, UI, to interact with the game</li> <li>&gt; Move the camera to adjust the location where you want to fire, spawn, move characters</li> <li>&gt; Move camera to see the map from different perspective, player is able to plan out strategy how to play the game</li> <li>&gt; On screen directives to show game progress</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Map spawn on the plane</li> <li>&gt; Map changes as the game progresses, more characters, particles</li> <li>&gt; Different position of player gives different viewpoint for the game</li> <li>&gt; Marker on the map shows the location where camera is pointing at, the location of the player choice of action will take place</li> </ul>
<b>ARise</b>	<p>Minimalistic – Only two menu buttons which are positioned at the bottom left of the screen.</p>  <p>ARise gameplay image from:  <a href="https://cdn3.macworld.co.uk/cmsdata/slideshow/3664321/arise.jpg">https://cdn3.macworld.co.uk/cmsdata/slideshow/3664321/arise.jpg</a></p>	<ul style="list-style-type: none"> <li>&gt; Camera movements and position affect the game - player movement required</li> <li>&gt; Position the camera to connect the symbols, which create a path for character to move through</li> <li>&gt; Tap screen to access menus, in UI, positions on bottom corner of the screen</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Whole map gets generated - able to see following obstacles, progress</li> <li>&gt; Moving around to enable new paths for character to wall through, progress with game</li> <li>&gt; Symbolic icons on the path to connect to create path - shine up when connected and working</li> </ul>
<b>Splitter Critters</b>	<p>Minimalistic – Only two menu buttons available on the screen. One on top right and one on bottom left.</p>  <p>Splitter Critters gameplay image from:  <a href="https://venturebeat.com/wp-content/uploads/2017/09/screen-shot-2017-09-19-at-4-14-49-pm.png?w=800&amp;resize=800%2C500&amp;strip=all">https://venturebeat.com/wp-content/uploads/2017/09/screen-shot-2017-09-19-at-4-14-49-pm.png?w=800&amp;resize=800%2C500&amp;strip=all</a></p>	<ul style="list-style-type: none"> <li>&gt; Player interacts with the game by swiping or dragging a cutting line in the game to create walk path for character to reach the goal position</li> <li>&gt; Player is able to walk into the game, enabling player to see layers inside the game</li> <li>&gt; Tap on screen menus, in UI, to access game menus and options</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Game presented inside the box - on a first look it looks like an image</li> <li>&gt; Map changes with every cut player makes</li> <li>&gt; Able to walk inside the box to see the inside of the game - giving illusion of being in the game</li> </ul>

Relevant features from the examples in *Table 1*, and other relevant AR games, can be used to be implemented in this project AR game, to provide good user experience, given those game success. Feedback will be gathered throughout implementation to have it improved based on the game being implemented.

## Motivation

Motivation for this project is to improve my skillset in the field of AR and in user experience, UX. As ARKit and AR is interesting topic and it is area which hasn't been fully explored yet. ARKit being a new toolkit, testing its features and finding limitations sounds promising and interesting.

Previously being part of a team creating AR game, in AINT354, there are a lot of user experience issues with the game which could be improved. UX is big part of why that project didn't do as well as it should, and in this project my goal is to create AR game that will have improved user experience, compared to my previous project.

Having iOS and MacOS available for development, it only seemed suitable to choose this platform and kit to explore in.

## Project Objectives

**Table 2**

UX Feature Name	Prototypes
UI	Prototypes for different UI for the game.
Interaction	Prototypes for different player inputs, game interacting with player - sounds, interactables (buttons, world...)
Visuals	Prototypes to different representation of the world, directives, information provided in game

In this project, the objectives are:

1. Create series of prototypes to test different features for UX, from *Table 2*.
2. Gain feedback on the prototypes created, with ideas for improvement.
3. Using gathered feedback, create a game demo using ARKit.

## Initial Scope

1. Create prototypes to test out following features: player input (swipe, point and click, on screen movement controls), visuals (floating on screen, UI, toggle on/off), player feedback (colour change, sounds), progression (visuals, UI), UI.
2. Gather feedback on the prototypes created, and gather potential improvements.
3. Improve the prototypes based on feedback.
4. Implement a demo for a game, involving features from feedback and implement it in ARKit. Aim being to create visually pleasing, positive user experience demo.

## Resources and Dependencies

This project is dependent on hardware, as it needs to be created on MacOS and have iOS 11 (and A9 chip) to be able to test and implement this project.

Project can be implemented on Mac computers in Babbage 210 lab, as well as personal MacBook. Tested on personal device iPhone 7.

### **Method of Approach**

The production of the prototypes and tests will follow an Agile Methodology.

Initially prioritise the features that prototypes will be created for, this will be priorities by: ease of implementation, existing understanding/knowledge on the feature.

By the use of User Story Mapping, the sprints will be created to set deadlines and have a clear direction what is being implemented at each stage. Each sprint will have specified information with functionality and how it affects the game.

After the sprints are specified, the prototype will be created, followed by quality assurance. Quality assurance will test out the prototype to see if it is suitable for use. Proof of concept will be performed afterwards, which will involve demonstration of the prototype to interested parties and project supervisor to get feedback and make sure prototype is performing as it should.

This method of approach will be repeated, once the technology is successful in testing. The process will be repeated until all agreed features are created. Followed by testing from colleagues and have public testing, where the objective being to get as much feedback as possible.

Possible technologies are Unity3D, C#/Swift, Xcode, although a full evaluation will take place during the project.

### **Project Plan**

Project Plan			
Stage	Expected Start Date	Expected Completion Date	Products/ Deliverables/ Outcomes
1. Initiation		2 <sup>nd</sup> February	PID
2. Sprints	3 <sup>rd</sup> February	7 <sup>th</sup> February	Have a plan and order of implementation which project will follow.
3. Prototypes / tests	8 <sup>th</sup> February	2 <sup>nd</sup> March	Weekly brief update, prototypes and tests completed
4. Testing	5 <sup>th</sup> March	7 <sup>th</sup> March	Present prototype to other people and gather feedback
5. Improvements	7 <sup>th</sup> March	11 <sup>th</sup> March	Improve the prototypes based on feedback gathered
6. Playable game	12 <sup>th</sup> March	3 <sup>rd</sup> May	Weekly brief update, playable game combining previously created and tested features.
7. Draft Report		04 <sup>th</sup> May	PRCO304 Draft Report
8. Final Report		17 <sup>th</sup> May	PRCO304 Report

### **Control Plan**

Majority of the project management will be done through Trello, an online project management software. Supervisor will be invited to the project board to keep a closer view on the project progress.

Weekly brief reports will be produced which will be reviewed with project supervisor and have weekly meeting with supervisor to make sure that the project is on track.

### **Communication Plan**

Weekly meetings will be set with supervisor to go over project progress and any improvements that can be implemented. Also, any concerns and plans moving forward will be discussed at the meetings as well. Those will be documented in weekly brief documentations that are submitted weekly.

### **Initial Risk List**

<b>Initial Risk List</b>	
Delays in deliverables	Constant meetings with tutor will be used to negotiate pragmatic solutions. There will be two weeks implementation time available at the end of the project, to make up for any delays during the project.
Difficulty learning/ using the development technologies	Prototypes and testing is achieved early in project which is used to upscale necessary skills for further on in the project.
Technology failure	At situation where testing (or implementation) device breaks down, the alternative project implementations will be discussed with supervisor.

### **Quality Plan**

<b>Initial Quality Plan</b>	
Requirements	Requirements will be validated to ensure that they are correct, relevant (i.e., traceable to the business objectives), complete, achievable and demonstrable. Prototyping will be employed.
Design validation	The design will be approved between myself and project supervisor. It will be made certain that what is being made is feasible given the time left.
Testing	In-house testing will be performed for each increment.
Public testing	Exposure of the project to the students in university. Feedback will be collected by Google Forms.

### **Legal, Social, Ethical and/or Professional Issues**

Copyrighted content – In the project everything will be either created by myself or clearly stated the source where it is coming from, to avoid breaking copyright laws.

Social and Ethical issues – space that the prototype may cover when it is in use.

Plymouth University Ethics Policy – place where the project will be tested on.

No other issues can be found at this stage of project.

### **Reference:**

- [1] [https://play.google.com/store/apps/details?id=com.nianticlabs.pokemongo&hl=en\\_GB](https://play.google.com/store/apps/details?id=com.nianticlabs.pokemongo&hl=en_GB)
- [2] <http://uk.businessinsider.com/ar-vr-2017-8>
- [3] <https://developer.apple.com/arkit/>
- [4] <https://developers.google.com/ar/>
- [5] <https://www.vuforia.com>
- [6] <https://www.kudan.eu>
- [7] <https://www.wikitude.com>
- [8] <https://www.pokemongo.com/en-uk/>
- [9] <https://itunes.apple.com/gb/app/stack-ar/id1269638287?mt=8>
- [10] <https://itunes.apple.com/gb/app/the-machines/id1280682965?mt=8>
- [11] <https://itunes.apple.com/us/app/arise/id1281550152?mt=8>
- [12] <https://itunes.apple.com/us/app/splitter-critters/id1107320261?mt=8>

## Appendix B: Highlights

### Highlight 1:

PRCO304: Highlight Report	
<b>Name:</b> Ludwik Bacmaga	
<b>Date:</b> 08/02/2018	
<b>Review of work undertaken</b> <i>During this week, there was a focus on organisation, and have a series of sprints mapped out, ready for development. There are currently five sprints in plan, each sprint containing development plan, splitting development into sections, to track progress easier. Every sprint has a deadline set to it, to have a rough idea of what is the object during specific time for this project. Deadlines end two weeks before the submission date, in-case of any delays or problems.</i> <i>Trello was set-up as a management tool and progress tracker, and GitHub repo was created for version-control of the project.</i>	
<b>Plan of work for the next week</b> <i>Start implementation of Sprint One. It will have goals for UX, with game mechanics improving with each sprint. Developer Diary will be implemented starting this week to track development progress and for future reference.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 30/01/2018	
<b>Brief notes from supervisory meeting(s) since last Highlight</b> <i>Feedback provided and recommended improvements for PID.</i>	

### Notes

- Please do not delete rows from the table
- Suggested length: Maximum half page; writing a Highlight should only take about 10 minutes
- Please upload Highlight Reports to the SPMS the day before your supervisory meeting
- The Review should include: review of work undertaken (and comparison with work planned) since the last Highlight, including details of any products/deliverables. Identification of any issues/risks of concern that have arisen since the last Highlight, and any previous issues/risks that are still a concern. Please give consideration to whether – or not – you have spent the necessary 30 hours on the project in the last week.

Highlight 2:

PRCO304: Highlight Report	
Name: Ludwik Bacmaga	
Date: 14/02/2018	
<b>Review of work undertaken</b> <i>During this week, the focus was on creating prototype and start working on planned sprints. Started by doing prototype to test tracking, followed by creating multiple different islands/planets and then creating a character that is able to move around the planets. Everything was tested and fixed any issues that were not working correctly, and found a case under where the character doesn't want to move towards a planet.</i>	
<i>After everything was implemented together, the prototype was tested by a colleague, who give relevant feedback regarding gameplay, UI, visuals and what features could be implemented that would improve overall game.</i>	
<i>Also, every day that implementation was done, an entry for Developer Diary was created to keep close track of progress and any thoughts/difficulties that were encountered during the implementation session. What was also created was a Research document – or start of it at least – where games which are relevant to this project, either in gameplay or features that it uses, are compared to each-other and then evaluate what works and what doesn't.</i>	
<b>Plan of work for the next week</b> <i>Start implementing Sprint Two. Work on the visual representation of the virus and how it will be represented to the player. Continue to have Developer Diary entries, and complete Research document, evaluating findings and look at some titles from VR as AR market is not that big at the moment.</i>	
<i>Testing will be done to test out the features and if everything works well together.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 07/02/2018	
<b>Brief notes from supervisory meeting(s) since last Highlight</b> <i>Create Developer Diary, do Research document comparing game titles, and create a prototype to test out the ARKit tracking.</i>	

### Highlight 3:

PRCO304: Highlight Report	
<b>Name:</b> Ludwik Bacmaga	
<b>Date:</b> 21/02/2018	
<b>Review of work undertaken</b> <i>This week, the virus for the game was implemented. The visual alert shows up, that is basic at the moment, followed by material change of planet which symbolises a virus. Followed by a connection link creation, a tube that connects two planets together as the virus spreads through the game world. Finally, the interaction of virus meeting player was implemented, which in this case symbolises game over screen, and pause menu was created. Everything is basic for the time being to test functionality and note down improvements that can be easily implemented in the current state.</i>	
<i>There weren't many problems along the way, the only trouble with implementation being implementing the link connection between the two planets with the tube. Everything was noted down in the developer diary, which is being up to date, every when the implementation took place with feedback on the progress.</i>	
<i>Throughout the implementation, testing was done to make sure the following implementation works, gather feedback on how to improve on the current state of the game. Following last week's feedback, the planets were made smaller and more spread apart, with now more of them showing up in the game world.</i>	
<b>Plan of work for the next week</b> <i>Start implementing Sprint Three. The sprint three involves the start of creating the ingredients and inventory system that will be used in the game. Ingredients will display what items are available to be picked up from the planets that the character is on, while the inventory system will display the items already collected. Plan to implement indicators of some kind to display availability to pick up the ingredients which will be easy for the player to identify and know what the indicator is symbolising.</i>	
<i>Testing will be done to test out the features and if everything works well together. Also, through the implementation the developer diary will be kept up to date with the implementation progress.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 14/02/2018	
<b>Brief notes from supervisory meeting(s) since last Highlight</b> <i>Happy with the progress. Help with for the following steps in implementation, the visual representation of virus in the game, sound, how it is presented for the player. Got recommendations for paper and book which will help in presenting the information.</i>	

Highlight 4:

PRCO304: Highlight Report	
<b>Name:</b> Ludwik Bacmaga	
<b>Date:</b> 27/02/2018	
<b>Review of work undertaken</b> <i>Aim for this week was to create menu system for the planets, so when player clicks on the planet, the action menu shows up showing what is possible to do as an action. Followed by implementation of the inventory system, and ingredients with the win condition for the player.</i>  <i>Due to other circumstances (informed the supervisor), all of the milestones were unable to be completed on time, having only implemented the menu display once the planet was clicked, with move action available for the player. There were unforeseen trouble in implementation along the way which also slowed down the progress for the implementation. The menu shows up on the left side of the planet.</i>	
<b>Plan of work for the next week</b> <i>Aim to complete Sprint Three, being the implementation of the menus, which show up logically next to the planet with possible actions for the player. Implement the inventory system and ingredients with the win condition for the player. Alongside this goal, bigger scale testing will take place, to test out the whole prototype and gain feedback on how to improve, change the current state of the game.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 21/02/2018 <b>Brief notes from supervisory meeting(s) since last Highlight</b> <i>Do testing for the game orientation, consider using assets pack for the game instead of creating them from scratch. Do visual change to alert system when the virus starts. Think about the sound effects to be implemented in the game.</i>	

Highlight 5:

PRCO304: Highlight Report	
Name: Ludwik Bacmaga	
Date: 07/03/2018	
<b>Review of work undertaken</b> <i>This week the menu system was finished, with the correct interaction between the game and the player. Implemented the inventory system and the ingredients system, which is used for winning the game. Different solutions for winning the game were implemented giving the player an option how they would like to proceed with the game.</i>  <i>The process of implementation had a few issues along the way, with random bugs, mainly in the way that menu is shown for the player and the interaction has some weird bugs where it would not show correctly or at all. This was fixed without any more troubles, and now there is also implemented a selection process for the player if they would like to get the ingredient or place the solution.</i>  <i>The inventory system was implemented on the pause screen, now looking like it takes advantage of the full screen now, instead of just being a menu. The menu is still implemented, just being on the side of the screen. The inventory system so far is pretty basic, and more work will have to be done to enhance it, but for initial testing and how it operates, it is fine.</i>  <i>Inside the inventory system, it shows the collected ingredients and the solutions numbers, with an option of creating a solution if the player has enough ingredients collected. Once the solution been made, the player has an option to place it on the chosen planet – once the player moved to that location.</i>	
<b>Plan of work for the next week</b> <i>Aim for next week would be to implement different solutions, at the moment the game is programmed to have two different solutions, the link and the big bomb solution, which cause different winning conditions for the player. The plan for the week is to implement the effects of each solutions, visual for the player.</i>  <i>Also, plan to do testing of the whole project so far, to gather feedback on the current state of the game. Looking to take feedback from minimum of 10 people.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 27/02/2018 <b>Brief notes from supervisory meeting(s) since last Highlight</b> <i>Complete the sprint goals, do testing along the way. Do testing of the whole game, with the testing documentations and signatures of testers.</i>	

Highlight 6:

PRCO304: Highlight Report	
Name: Ludwik Bacmaga	
Date: 14/03/2018	
<b>Review of work undertaken</b> <i>This week the solutions were implemented, where solution one is 'Big Bomb' and solution two is 'Health Link'. The bare bones are implemented, meaning that the effects are instant without any visual effects. Testing was also completed with people gathering feedback on Google Form. By the testing period, not all bugs were fixed which did affect the result, but the bugs are known and looking for a fix for them. Developer Diary is being kept up to date, with any issues encountered during implementation.</i>	
<i>The Big Bomb implementation was implemented with minor obstacles, which were overcome at the end, however it is instant without any visual feedback for the player when the explosion takes effect. For testing mechanism, it works fine, but for final product it will have to be changes, as it does impact user experience which this project is focused on.</i>	
<i>The Health Link solution was implemented with minor obstacles along a way too, having the code already created in the previously. The code reused was from the linking of infection in this case, limiting the spread to 3 times.</i>	
<i>The known bug/s are the freezing that occurs when infection has no more planets to spread on to, but also at random occasions when infection is being interacted with by the solutions. Both of the bugs are being looked into.</i>	
<i>Testing was also completed from 6 colleagues, with more testing and feedback to be gathered within next few days, delays from original stated number was due to availability. All the feedback was recorded on Google Forms with clear indications what should be changed, what should be implemented to improve user experience for this project.</i>	
<b>Plan of work for the next week</b> <i>Plans for the next week will be to implement the changes from feedback gathered from testing. Evaluate the feedback gathered to plan out and figure out what features have to be changes and what has to be added to enhance user experience. This will include changing some visual effects, adding features and fixing some major bugs which impact gameplay experience.</i>	
<i>From feedback gathered and meetings, the representation of the game will be improved, and apart from improving the game, the sound will be added to enhance experience.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 27/02/2018	
<b>Brief notes from supervisory meeting(s) since last Highlight</b>	
<i>Have a working demo created by this week and re-asses the schedule, if needs to be.</i>	

Highlight 7:

PRCO304: Highlight Report	
Name: Ludwik Bacmaga	
Date: 21/03/2018	
<b>Review of work undertaken</b> <i>This week was dedicated on improving the game, from the feedback received. The feedback wasn't very positive, recommending to change multiple things in the project. At the end what ended up happening is building the project from beginning, re-using anything that is useful from the original prototype. What was implemented this week: the improve graphics for the planets, added sound effects to planets and projectile, have introductory sequence to introduce the player towards the game, created a path now which the spread will go to, and a projectile that can be stopped. Also, the three abilities were implemented, where two out of three work as planned. The ability orb orbits around the planet to inform the player what planet consist what ability.</i>	
<i>There weren't many issues while implementing the new project, code for spawning, choosing which planet to spread on, the bomb ability, for those things the code from previous prototype the code was useful and re-usable. For everything else the code had to be created again. The introductory sequence took the longest time to implement, however, the pacing is still not correct and it will be changed, adjusted.</i>	
<i>The abilities that will be implemented in the game are: time slow, bomb, shield. Time slow slows down time for a limited time, bomb explodes based on the player position – although the visuals are still to be implemented for this ability, and shield protects the chosen planet from being infected. Bomb and time slow ability work as intended, whereas shield ability requires a bit of work to make it work completely.</i>	
<i>Sound effects were also implemented in the game without any major issues. Although, the sounds have to be balanced.</i>	
<i>All of the assets used for the improvement of the game, the prefabs for planets, projectiles, spawning platform were taken from Assets Store.</i>	
<b>Plan of work for the next week</b> <i>Plans for the next weeks are to work on the current state of project and improve on them, make them work in some cases. Do testing and gather feedback along a way.</i>	
<b>Date(s) of supervisory meeting(s) since last Highlight:</b> 14/03/2018	
<b>Brief notes from supervisory meeting(s) since last Highlight</b> <i>Original prototype has bad way of representing information, change the layout of the game, how it is presented towards the player. Add introductory sequence where player is introduced to the game, add sound effects for emersion, show the information to the player is more engaging and better presentable manner (menus in this game don't work that well as they were implemented).</i>	

## Appendix C: Stage Objectives, Plans and Reviews

### Stage One: Organising and Planning

The image shows a digital project management interface, likely Trello or a similar kanban board, overlaid on a photograph of a traditional Chinese pavilion nestled in a snowy forest. The board is organized into several columns:

- To Do...**: Contains "Add a card...".
- In Progress**: Contains "Add a card...".
- Complete**: Lists tasks with completion dates and counts:
  - GitHub Repo: Feb 15, 9/9
  - User Story Mapping: Feb 14, 5/5
  - Tracking Prototype: Feb 14, 1/1
  - Visuals - Sprint One: Feb 15, 2/2
  - Gameplay - Sprint One: Feb 15, 3/3
  - UI - Sprint One: Feb 15, 2/2
  - Interactables - Sprint One: Feb 15, 2/2
  - Sprint One Test: Feb 15, 1/1
  - Visuals - Sprint Two: Feb 22, 4/4
  - UI - Sprint Two: Feb 22, 3/3
  - Intractable - Sprint Two: Feb 22, 2/2
  - Gameplay - Sprint Two: Feb 22, 3/3
  - Sprint Two Test: Feb 22, 1/1
- OnGoing**: Lists tasks with current status:
  - Developer Diary
  - Research: 1/1
- Feedback**: Shows a vertical list of highlighted items.

At the bottom right of the board, there is a placeholder "Add a card...".

Stage Two: Sprint One Started

## Sprint One

in list [User Story Mapping](#)

Due Date



Feb 15 at 8:00 PM

### Description [Edit](#)

The starting point, the MVP of the project.

#### Goals - Gameplay

*Create a floating islands that player can travel to with the character - scientist.*

#### Goals - UX

*Interactables*

- Tap screen,
- Able to use UI,

*Visuals*

- Basic meshes,
- Simple colours for textures,

*UI*

- Buttons to navigate to main menu, pause game,

Stage Three: Sprint Two

## Sprint Two

in list [User Story Mapping](#)

Due Date



Feb 22 at 8:00 PM

### Description [Edit](#)

Improve on Sprint One

#### Goals - Gameplay

*Create a floating islands that player can travel to with the character - scientist. In addition the virus will show up on one of the islands and starts to spread across it.*

#### Goals - UX

##### *Interactables*

- Tap screen,
- Able to use UI,

##### *Visuals*

- Basic meshes,
- Simple colours for textures,
- Different colours for islands,
- Virus indicators,

##### *UI*

- Buttons to navigate to main menu, pause game,
- Menu,

Stage Four: Sprint Three

## Sprint Three

in list [User Story Mapping](#) 

Due Date



[Mar 7 at 8:00 PM](#)

### Description [Edit](#)

Improve on Sprint Two

#### Goals - Gameplay

*Create a floating islands that player can travel to with the character - scientist. In addition the virus will show up on one of the islands and starts to spread across it. The scientist have to collect ingredients and combine them together to fight against the virus.*

#### Goals - UX

##### *Interactables*

- Tap screen,
- Able to use UI,
- Swipe,
- Ingredients,

##### *Visuals*

- Basic meshes,
- Simple colours for textures,
- Different colours for islands,
- Virus indicators,
- Ingredients indicators,

##### *UI*

- Buttons to navigate to main menu, pause game,
- Menu,
- Inventory,
- Action buttons,
- Help (book) - combination list,

Stage Five: Sprint Four

## Sprint Four

in list [User Story Mapping](#)

Due Date



Mar 21 at 8:00 PM

### Description [Edit](#)

#### Goals - Gameplay

*Create a floating islands that player can travel to with the character - scientist. In addition the virus will show up on one of the islands and starts to spread across it. The scientist have to collect ingredients and combine them together to fight against the virus.* - to change based on feedback

----- UPDATE -----

Changing whole game (same formula, different implementation). Now will have a short timed sequence to introduce the player to the game (like a tutorial in a game form). Have abilities floating around planet to fight infection.

#### Goals - UX

##### *Interactables*

- Planets
- Projectiles (infection spread)
- Infection

##### *Visuals*

- Planets
- UI
- Infection Spread
- Sound Effects

##### *UI*

- Pause
- Button 1 (Pause)
- Button 2 (Help)

Stage Six: Sprint Five

## Sprint Five

in list [User Story Mapping](#)

Due Date



May 21 at 10:00 AM

### Description [Edit](#)

Improve from Sprint Four

#### Goals - Gameplay

*Planets are the game world, where one of them gets infected and starts to spread across other planets. Player has to prevent the spread and use abilities to get rid of infection. Sound used for que when projectiles are being fired and sound plays for good and bad planets to symbolise the atmosphere.*

#### Goals - UX

##### *Interactables*

- Tap screen,
- Able to use UI,
- Abilities,

##### *Visuals*

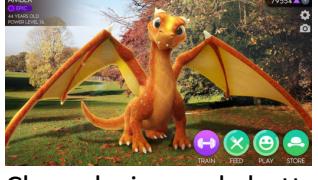
- Prefabs from AssetStore,
- Simple colours for textures,
- Different colours for islands,
- Virus indicators,
- Abilities indicators,
- Sound,
- UI,

##### *UI*

- Buttons to navigate to main menu, pause game, volume levels,
- Menu,
- Help (book)

## Appendix D: Research Documents

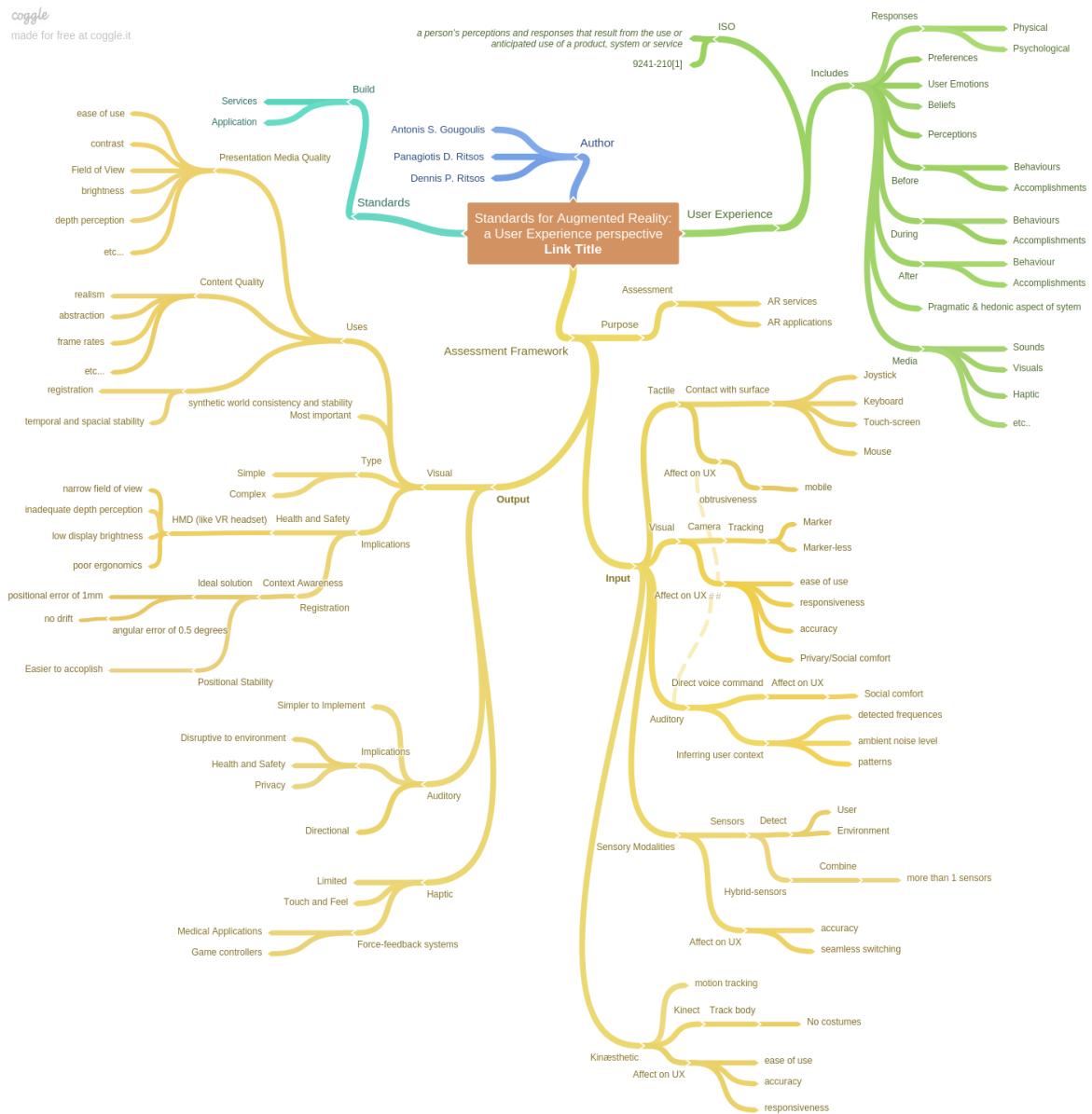
### Game UX Research

Game Name	UI	Indicators	Features
<i>AR Dragon</i>	<p>Menus hidden within few onscreen buttons, once clicked it shows more options.</p>   <p>Clean design, only buttons showing on lower sides of the screen with expandable menus.</p>	<ul style="list-style-type: none"> <li>&gt; UI has striking colour buttons for easily distinguishable buttons.</li> <li>&gt; World is spawn around you, only way to find out if to look around or if in practise mode (train mode) then the dragon looks in that direction.</li> <li>&gt; Distinguishable colours when it comes to pickups, shooting targets and mood of dragon can be seen by the face expression.</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Dragon needs to be spawn on big enough surface.</li> <li>&gt; Game uses whole world around you, even outside the scanned surface. Can be seen when in train mode, or point spawn.</li> <li>&gt; As Dragon grows, it requires bigger space.</li> <li>&gt; Can be picked up and placed anywhere (has to be scanned first and a plane) and dragon can only fly within this plane.</li> </ul>
<i>Space Invaders – not released</i>	<p>Non-Existent, very minimalistic.</p>  <p>Only information on screen is how many bullets are left in the gun. It is not released game, just prototype.</p>	<ul style="list-style-type: none"> <li>&gt; Pixelated enemies coming towards you from every side, the only way of seeing them is by looking around.</li> <li>&gt; Models becoming bigger the closer they are to the player.</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Takes advantage of the whole environment around player by spawning enemies coming at the player.</li> <li>&gt; Tap the screen to play the game (Shoot)</li> </ul>

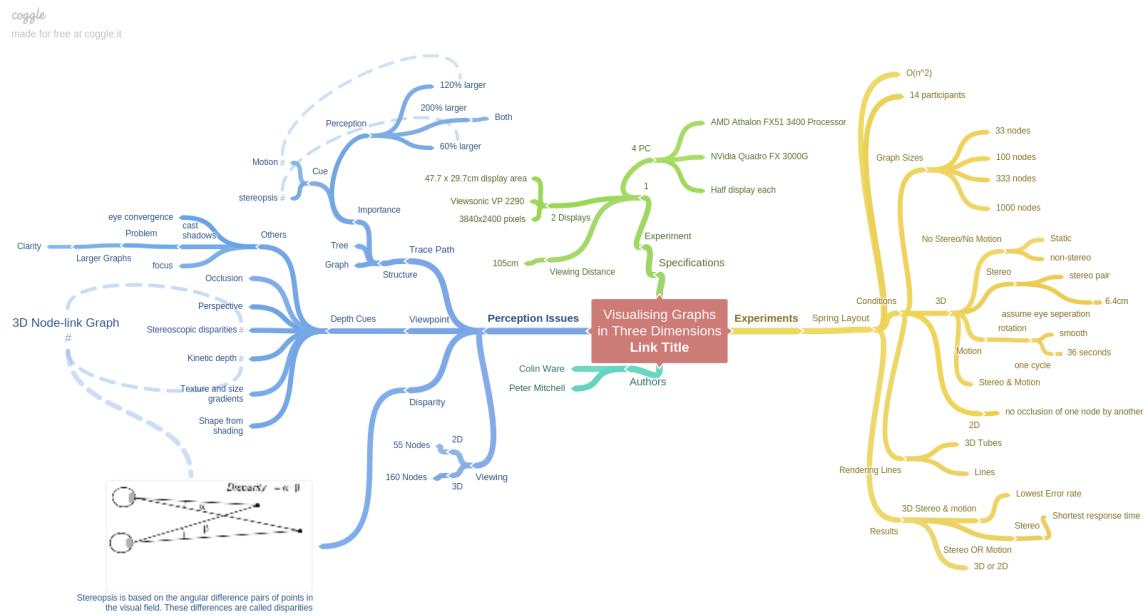
			> Tracks player location.
<i>Space Shooter – not released</i>	Non-Existent, prototype build, minimalistic information shown on screen. 	> No real indicators showing ships – just models floating around player. Have to look around to spot them.  > Not interactive models, stand still	> Ships spawning at random location around player  > Doesn't use planes to spawn objects
<i>ARise</i>	Minimalistic UI – Only options and menu buttons at the lower corner.   Simplistic UI, using camera positioning to progress with the game. Fits well with the game mechanic, to have most area to display the game.	> Symbols that connect together and shine once the camera joins the symbols together.  > Symbol shine up when connected and the path is created.  > Use of colours to clearly show progression, player location (by using distinguishable colours compared to whole game), symbols, can see whole level at once.	> Requires a plane to spawn the world.  > Uses camera position to play the game.  > Able to see whole level once loaded, allows player to plan ahead.  > One spawn location, but by the game mechanic (of progression with camera position), forces people to move around the map.

<p><i>Splitter Critters</i></p>	 <p>Minimalistic UI – Pause and restart button. Clean UI, slide screen to play the game.</p>	<ul style="list-style-type: none"> <li>&gt; Colours used to differentiate different game objects.</li> <li>&gt; Different texture symbolising cut.</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Box appears with a cut out which is playable window.</li> <li>&gt; Slide to cut and move objects.</li> <li>&gt; Able to go in the box to see different layers of game.</li> <li>&gt; Bright colours for enjoyable visual experience.</li> </ul>
---------------------------------	---	---	---

## Standards for Augmented Reality: a User Experience perspective



## Visualising Graphs in Three Dimensions:



## Appendix E: Ethics Approved Application

Faculty of Science and Engineering Ethical Application Form PS 2015/16 Final

### **PRCO304: Approved Ethics Application**

The ethics application below has been approved subject to the following condition:

- Audio/video recording will only be carried out with the written permission of the Participant – and this will be made explicitly clear on the Research Information Sheet

**PLYMOUTH UNIVERSITY FACULTY OF SCIENCE AND ENGINEERING**

**Research Ethics Committee**

**APPLICATION FOR ETHICAL APPROVAL OF RESEARCH INVOLVING  
HUMAN PARTICIPANTS**

---

**1. TYPE OF PROJECT**

**1.1 What is the type of project?**

**STAFF**

- Specific project  
 Thematic programme of research  
 Practical / Laboratory Class

**POSTGRADUATE STUDENTS**

- Taught Masters Project  
 M.Phil / PhD by research

**UNDERGRADUATE STUDENTS**

- Student research project  
 Practical / Laboratory class where you are acting as the experimenter

**2. APPLICATION**

<b>2.1 TITLE of Research project</b> Client and user interaction for Final Stage Computing Project module (PRCO304)
<b>2.2 General summary of the proposed research for which ethical clearance is sought, briefly outlining the aims and objectives and providing details of interventions/procedures involving participants (no jargon)</b> For PRCO304, many students develop (design, build and test) a piece of computer software to solve a real-world problem – possibly for a real-world Client organisation. A key component of this development is interaction with users and members of the Client organisation, which aims to elicit information on the required features of the intended system and (once a system prototype is available) gain feedback on its features. The work is individual (not group) - each student chooses their own topic. For all projects covered by this ethical application: <ul style="list-style-type: none"><li>• All participation will be entirely voluntary, and participants can withdraw during the session.</li><li>• No financial incentive will be given.</li><li>• Participation (or not) will have no bearing on the participants' work (such as performance reviews, promotion, or academic assessment).</li></ul> The client/user interactions covered by this ethical application does <b>not</b> involve: <ul style="list-style-type: none"><li>- ethically sensitive issues</li><li>- children (under 18) and vulnerable adults</li><li>- deception</li><li>- external institutions other than the Client organisation</li><li>- the general public (including family, friends)</li><li>- physical or psychological risk to participants</li></ul> The interactions covered by this application do include (the above-mentioned) interactions relating to the

development of web-applications; they do not include (other forms of) research over the Internet.
If a student wishes to undertake a topic which does not comply with the above, they will need to submit a separate ethics application.
<b>2.3 Physical site(s) where research will be carried out</b> University Campus or Client organisation premises
<b>2.4 External Institutions involved in the research (e.g. other university, hospital, prison etc.)</b> The Client organisation
<b>2.5 Name, telephone number, e-mail address and position of lead person for this project (plus full details of Project Supervisor if applicable)</b> Dr Chris Johnson PRCO304 Module Leader 01752 586244 <a href="mailto:c.johnson@plymouth.ac.uk">c.johnson@plymouth.ac.uk</a>
All students have an individual PRCO304 project with their own individual supervisor
<b>2.6 Start and end date for research for which ethical clearance is sought</b> Start date: 1 February 2017 End date: 1 February 2020
<b>2.7 Has this project received ethical approval from another Ethics Committee?</b> No
<b>2.8 If yes, do you want Chairman's action?</b> No
<b>If yes, please include other application and approval letter and STOP HERE. If no, please continue</b>

### 3. PROCEDURE

<b>3.1 Describe procedures that participants will engage in</b> There are 3 groups of people involved in this work: - academic staff responsible for delivering the module (specifically in the role of project supervisor), referred to as Staff - University of Plymouth students who act as investigators and are enrolled on the PRCO304 module, referred to as Investigators - University of Plymouth students (and in rare cases staff) and members of staff from the Client organisation who act as participants, referred to as Participants  The following protocol will be used:  0. Each investigator must arrange for their specific tasks, interview plan and/or questionnaire to be checked by a member of staff (usually their project supervisor), prior to contacting participants  1. Briefing 1.1 Participants will be provided with an information sheet (sample included as Appendix A), and a consent form (sample included as Appendix B). 1.2 If an anonymous questionnaire is used, then completion of the questionnaire will be taken as informed consent, and the questionnaire will indicate that withdrawal is not possible after submission.  A unique arbitrary number will be written on the information sheet and on all data relating to the current participant (to facilitate anonymous withdrawal at a later date). The information sheet is left with the participant, and the consent form is retained by the investigator.  2. Activity 2.1 System overview – The investigator may outline the aims of the intended system and/or demonstrate features of the system prototype 2.2 Prototype usage tasks – The participants may be asked to undertake artificial (but realistic) tasks using the system prototype - such as finding a page on a web-site - with the researcher observing. A Thinking Aloud Protocol may be used (where participants are encouraged to describe their thoughts during the activity). Video and/or audio recording may be used (with permission from participants) for transcribing purposes, to more accurately record aspects of the participants' experiences of using the software. These recordings may be transcribed or notes made from them, and recurrent themes identified (such as software benefits, errors, and difficulties). These will be used to assess the software's effectiveness, but will not assess individual participants. Counts of the number of participants reporting a theme may be used to
---

*prioritise themes.*

*2.3 Semi-structured Interviews which may be recorded: these will provide information about the participant's views on the required features of the intended system, or provide feedback on the participants usage of the system prototype*

*2.4 Questionnaire handed out on paper ... for the same purposes as 2.3*

*Audio/video recording is not essential and this fact will be conveyed to Participants. Audio/video recording will only be carried out with the written permission of Participants.*

*If permission is given for audio/video recording then it will be setup so that only those who have consented are picked up. For video recording a machine at the extremity of the room will be selected and the video camera fixed so that it can only pick up the user's hands and the screen of that machine. There is no requirement to capture images of participants' faces.*

*When University of Plymouth students act as participants, the research will take place on University of Plymouth premises.*

*When Client staff act as participants, the research will take place either on University of Plymouth or Client premises.*

**3.2 How long will the procedures take? Give details**

10-60 minutes per participant

**3.3 Does your research involve deception?**

No

**3.4 If yes, please explain why the following conditions apply to your research:**

**a) Deception is completely unavoidable if the purpose of the research is to be met**

**b) The research objective has strong scientific merit**

**c) Any potential harm arising from the proposed deception can be effectively neutralised or reversed by the proposed debriefing procedures (see section below)**

**3.5 Describe how you will debrief your participants**

**3.6 Are there any ethical issues (e.g. sensitive material)?**

No

**3.7 If yes, please explain.**

#### 4. BREAKDOWN OF PARTICIPANTS

##### 4.1 Summary of participants

Type of participant	Number of participants between 1 and 20
Non-vulnerable Adults	0
Minors (< 16 years)	0
Minors (16-18 years)	0
Vulnerable Participants (other than by virtue of being a minor)	0
Other (please specify)	0

<b>TOTAL</b>	<i>between 1 and 20</i>
--------------	-------------------------

<b>4.2 How were the sample sizes determined?</b>
Convenience, relating to what is achievable by students at this level within the time available. There is no requirement for statistical significance.
<b>4.3 How will subjects be recruited?</b>
Face-to-face – investigators may ask other students (and in rare cases members of staff) who they already know and regularly come into contact with. Email membership groups – investigators may approach a group that they are a member of and ask to send emails to other members (e.g. ask their programme manager to email other students on their programme, ask the chair of a club/society to email members of that club). Email unknown groups – investigators may ask their project supervisor to send an email invitation to more specialist groups (e.g. specific degree programmes, or clubs and societies). Staff from the client organisation may be nominated by the client organisation.
<b>4.4 Will subjects be financially rewarded? If yes, please give details.</b>
No

## 5. NON-VULNERABLE ADULTS

<b>5.1 Are some or all of the participants non-vulnerable adults?</b>
Yes
<b>5.2 Inclusion / exclusion criteria</b>
Participants will be students (and in rare cases staff) of Plymouth University (excluding partner colleges) or staff from the Client organisation
<b>5.3 How will participants give informed consent?</b>
Via consent form.
<b>5.4 Consent form(s) attached</b>
Yes
<b>If no, why not?</b>
n/a
<b>5.5 Information sheet(s) attached</b>
Yes
<b>If no, why not?</b>
n/a
<b>5.6 How will participants be made aware of their right to withdraw at any time?</b>
Information sheet, Consent form, and orally during briefing.
<b>5.7 How will confidentiality be maintained, including archiving / destruction of primary data where appropriate, and how will the security of the data be maintained?</b>
All audio/video recordings, field notes, and transcriptions will be stored on a password protected computer with access only to the investigator. Transcriptions will use a coding system (such as I – investigator, P – participant, etc.), all personally identifiable data will be substituted (e.g. 'J Smith walked in' would be changed to '[P2] walked in'). Published material (project report) will not identify individual participants (this is checked during assessment). Currently, this module has c. 166 students. It seems likely that around 50% of them will act as investigators.

## 6. MINORS <16 YEARS

<b>6.1 Are some or all of the participants under the age of 16?</b>
No
<b>6.2 Age range(s) of minors</b>
<b>6.3 Inclusion / exclusion criteria</b>
<b>6.4 How will minors give informed consent? Please tick appropriate box and explain (See guidelines)</b>
<b>6.5 Consent form(s) for minor attached</b>
<b>If no, why not?</b>

<b>6.6 Information sheet(s) for minor attached</b>
If no, why not?
<b>6.7 Consent form(s) for parent / legal guardian attached</b>
If no, why not?
<b>6.8 Information sheet(s) for parent / legal guardian attached</b>
If no, why not?
<b>6.9 How will minors be made aware of their right to withdraw at any time?</b>
<b>6.10 How will confidentiality be maintained, including archiving / destruction of primary data where appropriate, and how will the security of the data be maintained?</b>

## 7. MINORS 16-18 YEARS OLD

<b>7.1 Are some or all of the participants between the ages of 16 and 18?</b>
No
If yes, please consult special guidelines for working with minors. If no, please continue.
<b>7.2 Inclusion / exclusion criteria</b>
<b>7.3 How will minors give informed consent? (See guidelines)</b>
<b>7.4 Consent form(s) for minor attached</b>
If no, why not?
<b>7.5 Information sheet(s) for minor attached</b>
If no, why not?
<b>7.6 Consent form(s) for parent / legal guardian attached</b>
If no, why not?
<b>7.7 Information sheet(s) for parent / legal guardian attached</b>
If no, why not?
<b>7.8 How will minors be made aware of their right to withdraw at any time?</b>
<b>7.9 How will confidentiality be maintained, including archiving / destruction of primary data where appropriate, and how will the security of the data be maintained?</b>

## 8. VULNERABLE GROUPS

<b>8.1 Are some or all of the participants vulnerable? (See guidelines)</b>
No
<b>8.2 Describe vulnerability (apart from possibly being a minor)</b>
<b>8.3 Inclusion / exclusion criteria</b>
<b>8.4 How will participants give informed consent?</b>
<b>8.5 Consent form(s) for vulnerable person attached</b>

<i>If no, why not?</i>
<b>8.6 Information sheet(s) for vulnerable person attached</b>
<i>If no, why not?</i>
<b>8.7 Consent form(s) for parent / legal guardian attached</b>
<i>If no, why not?</i>
<b>8.8 Information sheet(s) for parent / legal guardian attached</b>
<i>If no, why not?</i>
<b>8.9 How will participants be made aware of their right to withdraw at any time?</b>
<b>8.10 How will confidentiality be maintained, including archiving / destruction of primary data where appropriate, and how will the security of the data be maintained?</b>

## 9. EXTERNAL CLEARANCES

<b>9.1 Do ALL experimenters in contact with children and vulnerable adults have <u>current DBS clearance</u>? Please include photocopies.</b>
N/A
<i>If no, explain</i>
<b>9.2 If your research involves external institutions (school, social service, prison, hospital etc) please provide cover letter(s) from institutional heads permitting you to carry out research on their clients, and where applicable, on their site(s). Are these included?</b>
N/A
<i>If not, why not?</i>

## 10. PHYSICAL RISK ASSESSMENT

<b>10.1 Will participants be at risk of physical harm (e.g. from electrodes, other equipment)?</b>
No
<b>10.2 If yes, please describe</b>
<b>10.3 What measures have been taken to minimise risk?</b>
<b>10.4 How will you handle participants who appear to have been harmed?</b>

## 11. PSYCHOLOGICAL RISK ASSESSMENT

<b>11.1 Will participants be at risk of psychological harm (e.g. viewing explicit or emotionally sensitive material, being stressed, recounting traumatic events)?</b>
No
<b>11.2 If yes, please describe</b>
<b>11.3 What measures have been taken to minimise risk?</b>
<b>11.4 How will you handle participants who appear to have been harmed?</b>

## 12. RESEARCH OVER THE INTERNET

<b>12.1 Will research be carried out over the internet?</b>
No
<b>12.2 If yes, please explain protocol in detail, explaining how informed consent will be given, right to withdraw maintained, and confidentiality maintained.</b>

### 13. CONFLICTS OF INTEREST & THIRD PARTY INTERESTS

<b>13.1 Do any of the experimenters have a conflict of interest? (See guidelines)</b>
No
<b>13.2 If yes, please describe</b>
<b>13.3 Are there any third parties involved? (See guidelines)</b>
No
<b>13.4 If yes, please describe</b>
<b>13.5 Do any of the third parties have a conflict of interest?</b>
No
<b>13.6 If yes, please describe</b>

### 14. ADDITIONAL INFORMATION

<b>14.1 Give details of any professional bodies whose ethical policies apply to this research</b>
<b>14.2 Please give any additional information that you wish to be considered in this application</b>

### 15. ETHICAL PROTOCOL & DECLARATION

To the best of our knowledge and belief, this research conforms to the ethical principles laid down by the University of Plymouth and by any professional body specified in section 14 above.

This research conforms to the University's Ethical Principles for Research Involving Human Participants with regard to openness and honesty, protection from harm, right to withdraw, debriefing, confidentiality, and informed consent

**Sign below where appropriate:**

#### STAFF / RESEARCH POSTGRADUATES

Principal Investigator: Chris Johnson (19 December 2016)

All members of Computing staff may act as a project supervisor of a project relating to this ethical application

**Staff and Research Postgraduates should email the completed and signed copy of this form to Paula Simson.**

#### UG Students

	Print Name	Signature	Date
Student:	LUDWIK BACMAGA		16/05/2018
Supervisor / Advisor:	MARILIS VARGA		16/05/2018

**Undergraduate students should pass on the completed and signed copy of this form to their School Representative on the Science and Engineering Human Ethics Committee.**

**Signature** \_\_\_\_\_ **Date** \_\_\_\_\_

## **Faculty of Science and Engineering Research Ethics Committee List of School Representatives**

School of Geography, Earth and Environmental Sciences	Dr Sanzidur Rahman Dr Kim Ward
School of Biological Sciences	Dr Victor Kuri
School of Biomedical and Healthcare Sciences	Dr David J Price
School of Marine Science & Engineering	Dr Gillian Glegg (Chair) Dr Liz Hodgkinson
School of Computing, Electronics & Mathematics	Dr Mark Dixon Dr Yinghui Wei
External Representative	Prof Linda La Velle
Lay Member	Rev. David Evans
<b>Committee Secretary:</b> Mrs Paula Simpson	

Committee Secretary: Mrs Paula Simson  
email: paula.simson@plymouth.ac.uk  
tel: 01752 584503

**APPENDIX A: SAMPLE INFORMATION SHEET**

ID: \_\_\_\_\_

**PLYMOUTH UNIVERSITY  
FACULTY OF SCIENCE AND ENVIRONMENT**

**RESEARCH INFORMATION SHEET**

---

**Name of Principal Investigator**

\*\*\*[student name]

---

**Title of Research**

Client and user interaction for Final Stage Computing Project module (PRCO304)

For the above module, I am developing (design, build and test) a piece of computer software to solve a real-world problem. I would like you to help me identify and evaluate the features of software. This work is being done in accordance with the University of Plymouth's 'Ethical principles for research involving human participants':

1. **Informed Consent:** The work may involve one or more of the following:
  - **Observation:** of you undertaking tasks using a prototype of the software, such as finding a page on a web-site (10-30 minutes). You are encouraged to describe your thoughts during the activity. This seeks to evaluate the software (not you) – to see how easy it is to use.
  - **Interviews:** You may be asked to describe your opinions on the desirable features of the software or on the usefulness of a prototype of the software (5-30 minutes). This may be audio recorded to provide an accurate transcription.
  - **Questionnaires:** You may be asked to complete a short questionnaire, asking for your opinions on the desirable features of the software or on the usefulness of a prototype of the software (5-20 minutes). Completion of the questionnaire will be taken as informed consent.
2. **Openness and Honesty:** There is no requirement for deception in this study.
3. **Right to Withdraw:** Signing this form does not commit you to take part in any part of the study. It is merely a record that you have participated willingly. You can withdraw at any time, during any activity, without giving a reason, by contacting the principal investigator and giving your **ID number** (at the top of this document). The data collected will not be in a format that could be used to measure the performance of individuals.
4. **Protection from Harm:** There is no obvious potential for this to occur, but the participant or the researcher can stop the study at any point.
5. **Debriefing:** The results of this study will be included in my project report, which will be available to all participants.
6. **Confidentiality:** Individual Participants will not be identified in any field notes, transcripts, reports, presentations, or publications (internal or external).
7. **Audio/video recording:** Audio/video recording will only take place with the written permission of the Participant

If you are dissatisfied with the way the research is conducted, please contact the principal investigator in the first instance, or the module leader: [c.johnson@plymouth.ac.uk](mailto:c.johnson@plymouth.ac.uk). If you feel the problem has not been resolved please contact the secretary to the Faculty of Science and Environment Human Ethics Committee: Mrs Paula Simson 01752 584503

**The participant should keep this document**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

---

Name of Principal Investigator

\*\*\*[student name]

---

Title of Research

Client and user interaction for Final Stage Computing Project module (PRCO304)

---

Brief statement of purpose of work

To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

---

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: .....

Signature: .....

Date: .....

**The participant should give this document to the researcher**

***APPENDIX C – Sample software prototype usage test***

**Scenario**

Phone is being handed over to the participant, asked to play the game. No instructions were provided to the participant; however, questions were answered if any were asked. Participant had options how to proceed and play the game, what orientation, and how to win the game.

Few ways of achieving the objective, through the use of different abilities that were available in the game. Goal of this testing was to recognise the efficiency of the application information representation towards the player, and how would participant react to the game states.

In this test, few things were observed. First was what orientation the participant helps the phone, which demonstrated what orientation the game was meant to be implemented in, landscape or portrait. As well as the comments that were made throughout, primary on how clear, or un-clear, the game is to play.

**APPENDIX D – Sample Questionnaire for software prototype evaluation**

**Software Prototype Evaluation: Questionnaire**

*This Questionnaire to be completed on GoogleForms.*

*Your participation in this survey will be greatly valued. Completion and submission of this questionnaire will be taken as informed consent. However, you are not required to participate. You can stop at any time. This questionnaire does not assess you in any way. Once submitted, it will not be possible to identify (and therefore withdraw) you.*

Please be honest, and as specific as possible.

Was it easy to understand what had to be accomplish / how to win the game?

1	2	3	4	5	
Completely Unclear	<input type="radio"/> Cristal Clear				

Comments - what did you like / dislike / how to improve? Any other comments?

Your answer

Was interaction with the game logical? (The tap screen and the orientation that the game was implemented in)

1	2	3	4	5	
Bad	<input type="radio"/> Good				

Comments - what did you like / dislike / how to improve? Any other comments?

Your answer

Did it make sense how things are represented in the game?

1	2	3	4	5	
Unclear Representati ons	<input type="radio"/> Clear Representati ons				

Comments - what did you like / dislike / how to improve? Any other comments?

Your answer

Were you encouraged to move around while playing game?

- Yes - Enjoyed it
- Yes - Prefer stationary game, but makes sense for this game
- Yes - Prefer stationary game
- No - More directions required to move around
- No - Not required to move
- No - I like it.

Comments - what did you like / dislike / how to improve? Any other comments?

Your answer

Is the UI (User Interface - the buttons, and menus, text, those things..) clear and logical? (Ignoring the art)

1	2	3	4	5	
Bad UI	<input type="radio"/> Perfect UI				

Comments - what did you like / dislike / how to improve? Any other comments?

Your answer

## Appendix F: Ethics Signed Participants

Faculty of Science and Engineering Ethical Application Form PS 2015/16 Final

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**  
**FACULTY OF SCIENCE AND ENVIRONMENT**  
**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: .....Shui Man Li.....

Signature: .....*Shui Man Li*..... Date: .....17/08/18..

**The participant should give this document to the researcher**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: ..... Charlie Hawkins

Signature: ..... Charlie Hawkins Date: 12.5.18

**The participant should give this document to the researcher**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: ..Aleksandra.Knyshyna.Bacmaga

Signature: ..A.Bacmaga.....

Date: ..12<sup>th</sup> March, 2013

**The participant should give this document to the researcher**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

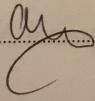
I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: IAIN HOEY

Signature: 

Date: 12/3/18

**The participant should give this document to the researcher**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: ..... *Chen Lee*

Signature: ..... *[Signature]*

Date: 14<sup>th</sup> March, 2018

**The participant should give this document to the researcher**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: STEVEN WATSON

Signature: Steven Watson Date: 12/3/18

**The participant should give this document to the researcher**

**APPENDIX B: SAMPLE CONSENT FORM**

**PLYMOUTH UNIVERSITY**

**FACULTY OF SCIENCE AND ENVIRONMENT**

**Human Ethics Committee Sample Consent Form**

**CONSENT TO PARTICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY**

Name of Principal Investigator  
\*\*\*[student name]

Title of Research  
Client and user interaction for Final Stage Computing Project module (PRCO304)

Brief statement of purpose of work  
To identify/evaluate the (intended) features of the \*\*\*[project topic] Software.

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: ..... *KIRBY CHEN*

Signature: ..... *[Signature]* Date: 12/3/18 .....

**The participant should give this document to the researcher**

## Appendix G: Developer Diary

### Developer Diary

#### **9<sup>th</sup> February 2018**

17:00

Started development with having a look through scripts that are inside ARKit unity plugin. No testing or changing code was done as of yet, just looking to through the scripts to get familiar with them and how they operate.

The goal for today was to start research and comparing more games which were implemented in AR, that use environment around player to work – how is UX designed in those games. VR games are to be considered too.

20:55

AR game titles have been looked at, where features and gameplay can be relevant to my project idea. Found out that at the moment there are not many titles that have relevant game mechanic, design that my project is focused on (world spawning around the player). Games looked at have minimalistic UI, have to figure out a clever way to minimize UI in the game.

Would be helpful to look at some VR games, to find out how they indicate information around the player.

#### **10<sup>th</sup> February 2018**

15:15

Had some issues uploading files to GitHub, where progress from today was deleted and had to start from beginning. I have done what was recommended to me, creating a prototype which tests tracking of world even when the spawning location is not in camera view.

This was implemented using detecting horizontal plane, as it will work with any user input, however I thought of implementing some kind of board game into this game and having horizontal plane to have as a starting point sounds like a good idea.

Game involves having a cube in the spawn location which is initiated by the player, and then based on that location spawning two spheres, which have different locations. Player can easily view them and look at them without looking at the spawning location and it tracks very well. No issues.

There was a bit of learning required to get used to the code and how it would operate. After I got familiar with the code and how it operates, I didn't run into any other issues.

19:30

There was an error with Developer Diary file, it was corrupted, had to re-download file that was backed up on GitHub.

Islands, which look like planets with very distinguishable colours for the time being, is implemented. First thoughts are that the planets are too close together, are a bit too big

(size of 0.5 on all axes on Unity). For the initial implementation it is fine, will have to test it out and get feedback on the situation. Need to gather also feedback on the spawn location, if the plane is required or not – will depend on feedback and how player wants to play the game.

Created player character – a cylinder for initial implementation – where the top positions of each sphere are the locations that the player will be able to travel between. Not able to travel yet, but does spawn at the correct position.

There was a small challenge of getting the top of each sphere to be recognised as a spawn location, but it was overcome. The locations are stored in an array which can be easily accessed. At the moment, everything is being implemented in one document, which will change soon. It is done this way at the moment for testing purposes.

### ***11<sup>th</sup> February 2018***

**17:15**

Today focus on development was to get movement (tap on island to appear on that planet). This was challenging. Initially I tried to implement it using HitResult that is in ARKit plugin, however it didn't work as after searching on the documentations, it appears that it only detects plains, and real-world objects and interactions, not what I was looking for.

My second approach was to use raycasts, which worked at the end, but there were problems to overcome along the way. Initial problem was getting 2D input to be transformed into the ray – it was quickly overcome by just double-checking code. One problem was detecting which planet it hit. As dependant on the planet hit, character will appear on that planet. Once I detected which planet was hit, the player appeared inside the planets – which isn't visible – instead of the spawning locations.

It did work when I hard-coded the location as it had the same offset throughout every island, but I wanted it to work with the location points that I initialised every planet with. After further testing and changing things, I realised that it was the code that I saved the location points at that was a problem and the way that I accessed them. Now they do get written and read properly. There is an issue that the very planet (last in array) that is in the origin doesn't get read properly, so have to investigate it.

**20:50**

The tap movement system works, for as long as the island/planet is not at the origin, it has offset of at least 2-axis from the spawn location. One solution that may overcome this issue is adding 0.1f to one axis (as it doesn't work if position has the same 2-axis as the spawn location). Minor issue that can be worked around, and one case where it won't work.

As the implementation is concerned at the moment, it is slow, as whenever I make any changes, I have to wait first for build to finish in unity, before building it on iPhone using Xcode which is a slow process, so testing is not as quick as I would have liked which affects the speed at which it is implemented.

The goal for today was achieved, where tapping does work to move the character. It has a bug that can be worked around as it is very minor condition which it works under.

### **17<sup>th</sup> February 2018**

14:15

Started implementing Sprint Two today. Started with implementing testing buttons to have a prototype position where they might be. Full UI implementation will be done at later date. Today I started implementing Virus and first thing which I wanted to implement is virus showing up in game. This was done without many problems using Coroutines. When it was implemented there were some issues, but they were issues setting up (assigning materials again since I renamed the variable), which were easily fixed. Once they were fixed, the game plays like expected and I set it up to initially wait 90 seconds before virus shows up but it can be easily changed.

At the moment, the whole project is being implemented on one single script. When everything is working and being implemented – thinking at the end of Sprint Three – then I will split it up into separate scripts to be neat and easier to manage the code.

### **18<sup>th</sup> February 2018**

13:55

Today I am focusing on visualisation of virus and the spread of the virus. I increased the spawning planet size to 10 – for better testing – and made the planets smaller (to 0.2f), will get feedback in testing to see if it is better to have the planets bigger or smaller.

I managed to implement spreading to the nearby planets without any troubles. I used mathematical Pythagoras equation to find the closest planet for which it will be infected. Once infected, at the moment it just changes colours and the planet tag changes from “Planet” to “Infected”, this information is used to find out if the planet is to be infected or if it’s already infected.

By testing it out, the spread of planets at the moment doesn’t look great and some planets are too close to each other, so once everything is implemented together, the plan of where each planet will be located – with the agreed size – will be implemented.

When was looking through the code, I found little bug that I fixed, where initially when I was calculating distance, I was only taking account x, y axis and y axis was taken into the equation twice, fixed it by changing one y axis to z axis.

15:40

For testing purposes, I reduced the time which takes for infection to start from 90.0f seconds to 20.0f, and also the time which it takes to spread is quicker than it will be implemented in the final product – 5.0f is set to at the moment.

The alert system is implemented – in portrait mode, have to get feedback which orientation the game should be implemented – when the infection begins. It starts 5.0f seconds before the actual infection appears on the planets with outline and text appearing on the screen. It looks ugly. But for testing purposes and gaining feedback, I think it will work. It will be changed later to look more pleasing.



Today was focused on creating a connection between the planets. It had a lot of problems along the way, in a form of first not displaying the prefab of cylinder at all. I tried to implement each connection inside an array, but for current implementation it wasn't implemented. Goal was to have a connection going from one planet to another, however at the moment all connections go from one – the beginning planet.

The prefab was initially along the y-axis, however later during testing it didn't display in correct direction, so the mesh was rotated 90 degrees in x-axis. This displayed the mesh in correct direction in testing, but not long enough which has to be adjusted in the code.

It was implemented using localScale, with prefab of cylinder being created of scale 0.02f in all axes. Using transform, I calculated the distance between the two points (the two planets), and calculated the midpoint from which the prefab was displayed. Tried to calculate how to stretch it along the correct path, initially trying to work in y-axis, but after many tries and testing, it wasn't working properly, followed by x-axis with the similar result. After testing out z-axis at the end, it worked, however the value has to be multiplied by 50.0f otherwise it is not the correct size. I am not sure to that value – with the distance being calculated before – but it has to be implemented and works at the moment.

All points go off one planet at the moment, which has to be changed to the links coming off correct planets.

### **20<sup>th</sup> February 2018**

12:15

Fixed situation where the link points only go off one planet. Now the links go off the closest planets – just as it should be implemented in the first place. Also, saved the connections – links – in an array for easy access at later time if needed, compared to just creating them when needed and not being able to access them again.

The information of start and end planet (which planets are connected to each-other) is stored in dictionary, key being the start planet and list of ints is created as the values for all the connections coming off that planet. The link is stored in an array, where at the moment I can't think of a solution to link the array and the dictionary together. In the meantime, whenever I need to access the link, I will look at start and end planet and based on the location of the planets and the link position I will figure out which link it is relating to.

Fixed a small bug, where the player starting position wasn't saved until the first move was made – the initial position of player wasn't saved. It is solved now and planet location is saved initially when game is loaded.

13:20

Dictionary was removed due to not working as intended. It was breaking the game, so for now dictionary is deleted, but the initial array for the links is still in place. By the use of planet locations, I will identify which link players are relating to.

After getting some feedback on the pause menu location – the feedback was to keep it on the bottom right (in portrait mode) for easy access and doesn't get in the way. Also, at the moment the character is able to go around every planet – what I want to implement is the limitation that player will only be able to travel around closest planets – not able to travel to any planet that they want.

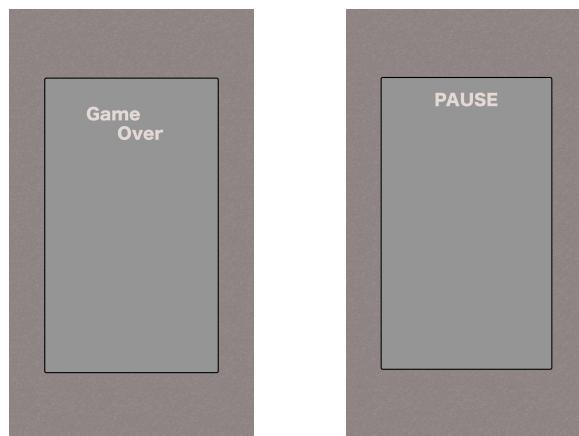
16:35

Implemented Pause menu where it has "Main Menu", "Restart", "Back" buttons, where main menu is yet to be implemented. It stops the time when the pause is in place. Also, have added command that never dims or puts the phone to sleep when the application is open.

Also added GameOver screen, which occurs once the player is on the infected planet. Surprisingly there were troubles to make it work, not sure what the problems were, but they seem to be simple like not referencing or calling the function properly and had to change locations when the GameOver function is called. I had it to scan every frame to see if the planet that the player is on infected or not and based on that affect the result, but it didn't always show the correct results, when the infection spread towards the planet that character was on already, the GameOver screen wasn't triggered. Therefore, I added if statement in the spread function which checks that if the player position is the same as the spread position then GameOver screen is triggered.

Also, made Game Manager game object which holds the scripts which control the UI and states of the game.

Everything implemented is very basic with simplistic graphics and functionality, the idea being to improve it once everything is implemented and working as intended.



**26<sup>th</sup> February 2018**

16:00

Did testing of 8 people testing the orientation of the game, the result being landscape orientation. The orientation of the game was changed to landscape. I used 3D TextMesh available in Unity to display text on menus which will be displayed next to the planet. The planet has 4 locations (front, left, right, back) where the menu can be displayed in. At the moment, the menu only shows up on the left side of the planet every time. The menu has one big intractable button – that is not active at the moment but will have point of action once programmed that will do an action. Actions available are move, pickup ingredients, place mix solution on the planets to cure the infection.

No issues at implementation at the moment, however it takes a while to think of how to implement it logically. Instantiating the prefab for menu to be displayed at the right location. Looking to have the right rotation to be in the direction of the camera. Meshes and colours are basic for the time being to minimize time spend on implementation and easier to make changes based on the feedback.

**17:40**

In progress on implementing which menu shows up on the menu. There are complications in the form of menu disappearing without displaying anything. Not sure the reason for that. Created separate variables to track the input. Have different functions to be called at times to track inputs and when the action button is pressed.

**20:20**

Issues occurred when trying to implement pressing the button, where the game would not load (the game world that is), unless the game is restarted through the pause button. Another issue was that the menu was not showing, or it wasn't counting the press correctly, either blinking or just not appearing, or appearing without being active. Also, another issue being the presses were not registered correctly, as moving the finger around would initiate the different functions. Currently, the game is implemented with touch.phrase == TouchPhrase. Began to track the press, and it does work as intended when it comes to presses, however it does not turn off the menu once the planet is pressed again. It only works initially when the planet is pressed.

**3<sup>rd</sup> March 2018**

**15:20**

Had an issue when it comes to turning off the planet menu as it did not work when was clicking planet again. Due to this, I decided to implement a red exit button which turns off the menu, and once it was implemented it worked as intended without any problems. Using raycasts to detect when the exit button was hit as it is in the tag on its own called "Exit".

Thinking of 3 win conditions (done by doing different combinations of ingredients gathered). Bomb that explodes and heals the planets / Healthy link spread / Health solution place on planet.

**4<sup>th</sup> March 2018**

**14:35**

Implemented all states: Move, Solution, Collect, Collected buttons, however it does not show in correct manner, there are bugs as the points/how many ingredients there is

available on the planet is not deducted. Have to fix those bugs and have working version before continuing implementation.

15:55

Implemented the text showing the number of ingredients available on the planet. Once it is collected, the menu disappears. Now everything is showing as it should, once the ingredients hit number 10, the player will be able to place the solution on the planet, and at the moment it send the player automatically into the winning screen. This will be changed to match the winning conditions, where player plays the game according to the mix combination.

There were a little trouble with displaying the ingredients on the menu plate. First I tried to search for the object by `.GetComponent<TextMesh>().text`, however it did not display any results, therefore I changed my strategy to first `.Find("Text")` (that's how the object is called in the prefab), followed by `.GetComponent<TextMesh>().text`. That seems to work. Then I was working on displaying text, where first I used `string.Format(args)`, however it did not show any results, it ended with not showing anything. Then I tried just entering "string" + arg way, however it did not work either. Reasoning for nnot working is unknown to me, however, when I tried to create a separate string variable and use `string.Concat(args)` it worked perfectly.

Once this was implemented, when the player clicked the collect button, the whole game would freeze, which was unknown error. However, I think it was as in collect function, I called recursive function for `MenuAction` which displayed correct text and menu options available for the planet. I passed the argument of player planet position, which I assume cause some sort of recursive call just freezing the game. I fixed it by not calling the `MenuAction` function, and once the ingredients were collected I changed so the menu just disappears.

That is the reason for menu disappearing once the ingredients were collected, not intended solution as I wanted the text to just change, however it does work as it is. The player just has to click on planet again to view possible options. It is fine as for testing the functionality, this solution may be changed at later date.

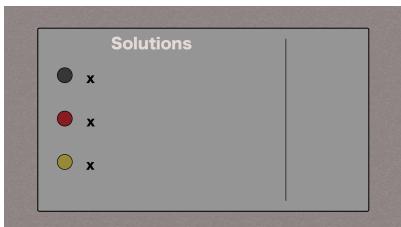
16:40

Implemented a split choice when the player has enough ingredients for solution, and if there are more ingredients available on the planet. Changed the ingredients required to place the solution to 5, previously being 10, for easier testing. Whole game will need to be balanced while everything is implemented and working. Now player can choose what to do. Next thing to implement is the inventory system, which initially will be implemented with the pause screen.

**5<sup>th</sup> March 2018**

12:45

Implemented the basics for inventory system, where it splits the screen with the pause menu, only showing the basics of the inventory for now. It shows how many ingredients the



player collected, with an option in inventory to create new solutions to win the game. The menu is moved to the side, which I think takes better advantage of the whole screen, while at the same time it has less interact-able buttons in the game screen, showing more of the screen for the gameplay.

The numbers get updated as the player pickups the ingredients, it is reflected in the numbers on the pause/inventory screen.

Plan to implement next is the solution choice in the planet menu, and the creation of solutions when the ingredient number is of certain number and the solutions having different effect on the game.

#### **8<sup>th</sup> March 2018**

11:40

Was implementing the two solutions for the game, including the UI in the menu, the code to count the different solutions, namely "Health Link" and "Big Bomb". The coding was implemented and there were no issues, however when I was implementing the UI, for some reason the UI did not want to appear, the solution button stopped working. It happened after I have changed the tag name for the solution button. This problem was fixed once I reloaded the project in Unity. Now everything is working as intended.

#### **9<sup>th</sup> March 2018**

21:00

Created the prefabs and everything ready to have implemented for two different winning solutions for the game. The link solution was copied and applied in similar way that the spread of the infection was implemented, with different green texture. The big bomb is yet to be implemented but will have a semi-transparent texture on a bigger sphere and the planets touched by the sphere will be 'healthy'.

The linking is not working properly, as when the link goes over the infected planet, it does not stop the infection, i.e. it's not working properly and it gets stuck. No other progress is made. Don't know why that happens, more testing will be done later to figure out the cause of that.

#### **10<sup>th</sup> March 2018**

10:55

Today goal is to finish implementing the two solutions for the game. Fixing some bugs with linking system, namely where the infection could spawn on the healthy planet. Now I have added additional check to see if the planet that is being infected is healthy or not with the tag check, as the planets have tags to see if they are healthy/normal/infected. Having those extra checks fixes this issue. Another change that was implemented is that there is a wait now when the link is put down so the first link doesn't occur immediately. Additionally, I added a constrain that the health link can only occur 3 times so the game won't end when the link is put down and it makes player to keep playing.

12:00

Implemented the bomb mechanic, where it has a radius and anything within this radius becomes healthy. I have implemented it by taking a collider of a bomb prefab and checking the bounds.interection(planet collider.bounds). If it is within those bounds then it is within the bomb range. Once it was implemented, it works as intended. Increased the blast radius, originally from 0.5, then 1.0, and now to 2.0 for it to be more impactful. Then reduced it back to 1.5, as 2.0 was too big and was the size of almost whole game world.

One thing I will have to change, or add, is the tag for bomb explosion, as it can be infected by the virus again. Also, looking to implement the instruction screen at the beginning when the game is launched and also the ingredients recovery, so after certain time the planets have extra ingredients.

Will implement the winning condition (when infection reaches = 0), and also a possibility of collecting ingredients once the infection started.

15:20

Implemented the different tag for planet, which represents healthy planet, but it can still be infected. Implemented the ingredients recovery – at the moment every 10 seconds the planets ingredients is increased by 1. Also, the menu system is now displayed the closest position to the player, has to be refreshed, the planet has to be clicked again. It will rotate to the camera too. I had to change the prefab to be face in the opposite direction to be visible in the game world correctly. But it does work as intended so far.

Next thing to implement will be to have instructions at the beginning of the game, and also increase the time in game when everything happens before doing public testing.

**12<sup>th</sup> March 2018**

11:20

Implemented the instruction for testing session, saying what to achieve in the game and what do spheres mean in the inventory/pause screen. Also increased the number of ingredients required for solution two (the link solution) to 10, from 5 initially, as it is a permanent effect and I find it to be fairer. Will see how it will impact the feedback. Another thing that was changed for the game test is that now the infection alert will popup 10 seconds in the game now, instead of 15 which was before, and the spread now occurs every 8 seconds instead of 5.

16:05

There were few bugs in the implementation, mainly the player couldn't click on the planet when it was green (healthy) and there was no win condition – the winning screen didn't appear. Both of which were addressed and the bugs were fixed. Planet click was fixed by adding the healthy planet tags to the click recognition code where menu shows up, and the winning screen was fixed by moving the condition from Coroutines function to update function.

The number of planets is also being implemented up to 20 (from originally 10), to increase the game time. No issues were encountered when fixing bugs or increasing number of planets. Except of making crazy-stupid mistakes (typos) and forgetting to change the number of objects (to take into the account the new planets too), also pasting the part of code twice, overwriting and breaking the game. This caused the game not to behave as intended, not changing materials and such.

The game freezes at some occasions, think it might be due to not able to find new planets or something to do. Looking into solution now.

**17:00**

Changed the prefab for the bomb back to 2.0f to have bigger impact on the game as I rearranged the planets. Now I check in the game if there is a planet to spread on to, if not then the virus Coroutines is stopped.

**16<sup>th</sup> March 2018**

**10:25**

Started implementation of fixing/changing the game to be more enjoyable. Starting from scratch and will use the code whenever I can from my unsuccessful prototype. I will change the menus, by deleting them, as they are not required. Add music for sound effects and better experience. Have models from assets store to use for better visuals. Have introductory sequence at the beginning of the game, instead of written instructions to introduce player to the game. Infected planet will have a projectile shooting at other planets which can be interfered.

Found the planet assets that I will use for the game. Have 7 planets (6 useable for my game) which can be presented and used for the game. Have created and modified the scale (range between 0.025 to 0.05 to have bigger range of different planets, and have the colours/shaders attached to them working for the game.

**12:55**

Implementing code for starting sequence, where planet moves from bottom of the spawning point up to its position, however, it does not display anything. After investigating the project, I figured out that I have forgot to put the prefabs and the array list objects in (the materials and the planets prefab). Testing it if it will display correctly or not, if it doesn't then I will test without the transition, to see if the planet will display at all. I implement the game currently in different scripts to make it easier to change at later date and easier to read. Looks nicer too.

**13:10**

After testing the spawning of planets, it does not display anything (apart of the spawning location). Investigating cause now.

13:30

Only displaying one planet, I put all spawning in one script, but it only spawns one initial planet before stop working (had to not use the shaders provided with the planets as it didn't work when implemented in the game).

16:20

Tried to find out what was the fault with the code written in the Spawn script. I have copied what was previously in the WorldSpawn script and the WorldSpawn script was working fine, and I compared it next to each other and both being identical (apart of script name). It was just phrased differently to be clearer and tidy. However, it did not work and was only ever displaying 1 planet. Tried restarting – didn't work

After trying to figure it out, I have copied the WorldSpawn script again, and deleted all the unnecessary data from the script (pretty much everything apart from the PlanetsSpawn function) and tested if it will work fine. Surprisingly, it did display the data and information correctly with planets displaying correctly. So, based on that code, I will modify it and work on it to complete my task given, still uncertain why the initial Spawn script wasn't working.

16:55

Implemented the random planet to spawn at the locations, instead of the same planet as it was done at the beginning. The planets after testing do seem like they are too close together, especially since those prefab planets are bigger than the ones previously used, and the distance apart will be increased once everything else will be implemented.

I have disabled the particles. I have also disabled the blue square showing up as I believe it affects the experience. The player has to scan the area a bit by moving the camera around and then the player will be able to spawn the world by clicking somewhere.

22:00

Manage to implement the transition moves from the spawning position up to the desired planet position, with scaling as it goes up (with Time.deltaTime). There were difficulties, as I was testing and doing more implementation and trying out new ways to implement (having a variable of Spawn to access the transform location of planet, or different functions, or have GetComponent<Transform>() which none work). Had to have only position to go across in the public function in the spawned script, that's the only way that it works when I implement the transitions.

Ended up having a Vector3 as a position to go across to script which I store variable for the planet original position and then I move it from the spawned location up to the planet location with increments of Time.deltaTime. Scale was easier to implement once the transition was done, I have taken the one float from the localScale of the planet (knowing that x, y, and z have the same float variable) and I stored it in a variable which then I had another variable to make it smaller, and just like the transition, I increment the scale with Time.deltaTime (without any multipliers, as it seems to give the smoothest transitions).

I have created a Get and Set functions (for Scale and Position (transition)) in the Spawn script, as the variables are private and that's one way of accessing them from the other scripts. I access them, and modify, in the GameManager script. Looking to make a popup with description next, followed by an infection introduction, followed by the infection spread, then ingredients, then world spawn. The description text might be left towards the end. The sound will be implemented at the end too.

22:35

Based on the already created transition, I implemented another planet showing up with the planet changing material to red (to symbolize infection). Thing that has to be still implemented is the instructions, and the shooting spread of infection, ingredients in the available planets, and the sound.

**17<sup>th</sup> March 2018**

12:00

Found a space ship and platform which will be used in the game for the spawning location and the infection spread missile. Implementing the infection spread now. The spawning location is at 0.2f scale on all axis, while the projectile is at 0.04f scale on all axis.

13:30

Manage to make the projectile go from planet to another planet, although originally, I wanted to have the transition code to be in the projectile prefab, but it wouldn't work for some reason (nothing shows up), and some variables had to be assigned were unassigned if the object wasn't in the scene.

I have overcome the problem of instantiating the prefab in the Spawn script when the planets are being spawned, and setting its active state to false and calling all functions and calculations in the GameManager script. The projectile script will have a script Destroy, which will take affect when player clicks on the prefab. Also, once the projectile was working, it was facing incorrect way, so I rotated the prefab to -90 degrees in the y-axis which solved the problem.

Also, while doing testing, I noticed that the prefab is still a bit too big in proportion to other objects in the game, therefore the object scale will be reduced to 0.02f.

14:05

Implemented the path, which makes up of little spheres, which indicate the trajectory of the missile. It is implemented using GameObject array. No issues encountered implementing this.

Made the planets move quicker – to 5.0f – but it is too quick, so it will go back to traditional 1.0f.

16:30

Made the path to appear one by one, and now the planet changes colour to warning colour when planet is being infected.

**17:30**

Was implementing the hit collider, when realised it didn't work on the projectile, and then I figured out that it is because the prefab doesn't have a collider, the problem was solved once I added the collider to the prefab.

**18:15**

Implemented a reset point if the projectile hits the planet at tutorial session. Limited the click range from 10.0f to 5.0f to encourage player to move around. When the projectile reaches about  $\frac{1}{4}$  distance left to the planet to infect, the Time.timeScale is reduced to 0.1f for information popup and time for player to adjust.

**18:35**

Hit range reduced to 1.0f, as it feels like you can touch close to you objects but not far away objects. The distance at which the planet becomes infected is now reduced, so it gets infected quicker (once the projectiles disappear in the planet).

---- 19:15 -----

Thinking how to implement the ingredients for the planets. Decided that there will be three solutions/options to choose from being: Slow Time, Bomb, Shield. Each one of those will be available on different planets, floating around the planet that will be available to be picked up when clicked on the planet. Thinking of those solutions to float in orbit around the chosen planet.

The UI being the pause (top right), planet status (healthy/infected in top left), abilities located in bottom right (with availability ring around each ability).

Things that still need to be done:

> Graphical assets (introductory information, abilities, ability indicators, planet statuses, pause)

Graphics taken from Flaticon.com (free to use, have to state the author). Have ability icons from this website.

> Sound (background music, sound effects when projectile gets shot, sound effect when planet gets infected, when planet gets healed)

Chipmunk sound from soundbible.com (talking sounds, searched: inaudible talking sound effect) Also sounds for infection bg music (water drops) and shot fired (shotgun). Attribution 3.0 licence.

BG music has text where it's from

> Win/Lose screens/conditions

> Ability coding and effects

**18<sup>th</sup> March 2018**

12:40

Implementing the sound, and as I have never implemented the sound in the game firstly I was trying to figure out everything and how it would work. After having just one audio listener (on camera), I added audio source to the camera, as a background music. Firstly, it didn't work, and after further testing I realised that the device has to be unmuted for it to work, and after that it worked fine. But in testing I come to conclusion that it works better as a planet music instead of background music.

The music for infected planet is also implemented, having issues by script on planet trying to activate it, therefore it is activated by taking AudioSource component when the planet becomes infected and it plays. However, I find the radius is too far and has to be reduced (currently maximum is 0.4).

12:50

Realised when looking at variables, that the settings were set for 2D sound, and had to be changed to 3D for fading to take effect. Now it works as planned. After testing, the range got increased (dependant from planet, as planets are different sizes). Before all planets had 0.4 for maximum distance, but due to size I changed it to be between 0.2 to 0.4. Now the distance will increase to get better experience and hear the sound. Increased now between 0.65 to 0.8.

AudioMixers are used to mainly control the volume of the sounds.

13:15

Thinking I will increase the sound of infected planet to a larger area. The sound for shooting was also implemented, and thinking the sound for destroy has to be added too. For now it is okay, and have to complete other tasks.

14:50

Now that the sound is in place, I started to work on infection spread. I rearranged the code a bit so it would be easier to skip tutorial if player wants to just start a game. I looked at the code from the original prototype to base the foundations on when it comes to choosing which planet to spread on (random choice between 3 closest planets). I created a list of infected planets which will be used for calculations and knowing which planets are infected.

There were some complications, which were solved by checking the code. One mistake that I made was when I initialised the List, I forgot to add = new List<int>(); which caused planets not to appear at the point where infection planet number was being added to the list. Also, at the end there is a problem where the game breaks when the tutorial is finished and the next spread happens. So, have to investigate where the issue lies.

In the code, the calculation functions are created in the Spawn script, and the GameManager just calls them whenever required and it keeps track of the game state.

18:10

Investigating issue regarding crashing and not spreading (after tutorial) infection. There was an error made by me in Spread function in Spawn script, where I checked distance between the planets and would only ever consider one option (the closest one), instead of 3 and

choosing one at random. Meaning that the results would not be accurate and not work properly. Also, I am changing to code to have 5 projectiles available at a time to shoot from random infected planets.

19:20

Relooked at the code to correct any mistakes and to modify the code to work with 5 projectiles (and paths) at the same time. There is a bug as the infection is moved to another planet (even though it is still present in game). Crushing from time to time. Investigating.

20:00

Solution was fixed (problem was that after the game started, after the tutorial session thing, the infection would go to random planet). Problem was that I was choosing random planet, and didn't relate it to the infectedPlanet list. It is fixed now and working as intended.

The function was created to deactivate the projectile and path, if it was clicked, however, upon further investigation it was deleted, as the one already created had additional function to change the planet material to good planet, which is required anyway (did not realise this when creating the new function).

Also, the wait time between the movement of planets or projectiles was decreased from 0.1f to 0.05f which produces smoother movement.

There is a mistake in the code when the game starts, where the projectile keeps moving even when clicked. It is fault with the coding. Due to fix now. (as there isn't a variable to breakthrough the checklist).

20:15

There are some minor things being changed to make it work, some mistakes were created (like mis-typing the start location). Fixed now, and after looking why planets don't get infected, turned out that I do require a separate function for deactivating projectile and path if infected as the other function also turns the planet good, which is the opposite what is being tried to achieve. The planets do only infect 1 by 1 which was not original intention. Looking for a way to fix that.

21:05

Was trying to implement different Coroutines for different projectiles and call them when they are needed, but it just froze the game and did not work, so returned it to one projectile at a time, and will try to find a solution for simultaneous shooting when all other necessary things are implemented.

Make the infection material smoother (shiny) to be more distinguishable as it does look black in some cases.

Thinking of simultaneously shooting projectiles (random number), and in while loop have condition which checks all projectile status before continuing and waiting for shooting next bunch. Will test it once abilities are implemented.

Thinking of how many abilities can user store at a time. Should player have access to all abilities and just the counter increases as they pick them up, or have limited number of abilities at disposal (e.g. 3) where they have to use them before they can pick new ones up.

Bomb will take place at the camera position (the player), instead of the planet. Radius TBC. Thinking of have coloured balls to fall around the planet indicating different abilities. Blue for time slow, orange for bomb, teal colour for shield.

### **19<sup>th</sup> March 2018**

10:20

Updated the music, where the sounds in the game only come from planets and the projectile. The good planet has a good, relaxing sound (previously background music), and infected planet has a sound like before. I am changing the sounds as I change the material, first stopping sound then exchanging files followed by playing sound again.

Decided to do it as the player will move around and any sound in the game will be either good or bad, no other distracting sounds. Also, I couldn't find a background sound that I would say that fit the game.

20:45

Was implementing the abilities, and the time ability works as intended without any problems. The shield ability, I found tricky to implement, as I have to register when the ability is wanted to be casted and when then player clicks which planet they want to shield. It is not working as of yet, with only simple square UI buttons on bottom right symbolizing the abilities activation button. The ability counter is on top right.

While testing, there was an issue that show up, and that is UI buttons registering with the game world objects when being clicked. Once the abilities are working, that is a thing that has to be looked into and prevented as it can have consequences on the gameplay.

The shield ability at the moment is implemented in this manner: The player clicks the shield button, which then has bool variables to see if the next click has been made yet or not and where the click has been done. Upon completion, that information is being fed back to the function to decide whether to shield the planet or not. However, since the number of shields doesn't go down, there is something wrong with the code.

21:15

After changing some variables (the initial check when the shield ability is activated was set to the opposite (had to check for when its false, it was activating at true) it works 40%. It doesn't always work, as it has only specific time window when it works and there are some issues and bugs with it at the moment, especially with the counter and when it gets refreshed, how it works. Also, in testing there is a time when it just stops working altogether

for unknown reason as of yet. Looking for a fix. Firstly, want to implement bomb to see if it will work as intended.

21:30

Using the code (and modifying it for the new implementation of the code), from the first prototype for the bomb, it is working as intended. Not much has to be changed and modified to make it work, apart from changing some variable names and re-directing where the relevant functions are located. Still have to make the counter work for that, but it is not hard, and have to make the visual representation of the explosion. First want to make the functionality working which is done. The only ability that has issues is the shield ability, and it is to be looked at tomorrow.

At the moment known issues:

- Pacing of the game (the speed of planets/projectiles/spread)
- Distance between planets
- Shield ability
- Counter
- UI buttons interacting with the game world

**20<sup>th</sup> March 2018**

20:00

Today I was only available to work on the project in evening time, as since got to present it tomorrow, the shield ability will not work 100% at the moment. Instead I implemented the planet healthy and infected counter on top right corner. Also improved counter and ability usage for bomb and time abilities, and added graphics for the abilities. If (at the moment only bomb has this feature and time ability) ability is inactive, then the button turns grey and doesn't do anything.

21:00

I made the ability balls that orbit around the planet as an indication what ability is available on which planet. It works fine on first implementation without any issues. Implemented it using RotateAround function with transform.parent to get the planet origin to rotate around.

Material of the balls also change once the ability been changed, with no ball showing if no abilities are available on specific planet. Will be changed visually at later date; this basic visual implementation is done for testing purposes.

21:30

Done some instructions at the beginning of the game (in the tutorial section), to help player to get to know the game. To be changed later. The abilities also now display at correct time, and not at the very beginning of the game. Also, the game now behaves with abilities correctly (to set ability to planet and to change accordingly).

22:00

How the game is presented, the ability balls are presented after everything else is presented to the player, so the initial state of abilities is false, but there seems to be a problem

regarding activating them again through code (since it is in the prefab). To overcome this problem, I first GetComponent<Transform> of whole prefab, then Find("Ability") which is how the prefab is called for ability, which then I access the gameObject followed by setting it true (visible) or false (deactivated).

For better gameplay, the planets should be more spread apart, but it will be done after feedback that will be received tomorrow.

**17<sup>th</sup> April 2018**

11:25

Having an emergency situation, forced me to not make as big progress as was originally planned.

From feedback gathered, the plan of implementation is as follow:

✓ get rid of ability UI, have UI presented to the spawn location and player has to click it to activate it. Bomb will activate when clicked again on screen, slow time has immediate effect, and shield planet the player has to click planet to activate.

> have ability prefab to be more distinguishable, better texture. (To be done when back in university) – thinking of different way of representing the ability available on the planet. Such as: particle system around the planet.

✓ have spread projectiles working independently, game can spawn 5 projectiles at a time.

> create new scenes to introduce player to the game. At the moment scenes planned are as follow: first screen introducing planet, and infection mechanic to the player with the ability to shoot the projectile. Second screen will contain abilities explanation. Third scene will have a game.

✓ create shader ripple effect when player clicks the planet (from catlikecoding). Have particles around the abilities to make it more pretty.

> create a hologram spawning effect at the spawn location (To be done when back in university)

Starting off with the first point, the UI for abilities was removed and now working on moving ability to the spawn location (with y-axis increased by 0.1f initially), where the ability will float.

16:15

Abilities around the planets are still displaying, but now when clicked, it will check and the ability will show on the spawn location ready to be activated. Once the ability or the spawn location is clicked, the ability turns off, ready to do its action (not yet implemented). It took a while as had to get familiar with the code again and remove some unnecessary things which will not be used. Namely, the ability UI and the code and functions which are involved

with the previous ability usage. The functions of how the ability works is still in the code, just how to activate it has changed.

**16:50**

Made the ability on the spawn location move up and down, smooth motion, and rotate around for better visual effect.

**18<sup>th</sup> April 2018**

**14:00**

Implemented the new functionalities with the new implementation of the abilities. It is not tested, just coded in. Now the time is not affected, just the projectile speed is affected when time slow ability is activated. Bool variables keep track of when the ability is active and pulsing effect (with scaling) takes place for better visuals of the ability prefab. How code is activated has changed, however how the code operates stays the same, with only difference being that the bomb now affects projectiles too.

**16:40**

Implemented the projectiles to be independent. This allows for multiple projectiles to be shot at once, maximum of 5. It is implemented by adding a new script to each planet which calls Coroutines which work as before just now independently. It is implemented; however it is not tested as of yet. To be tested. How it works is the same as it was before, as it was working properly (shooting a projectile to the planet).

**19<sup>th</sup> April 2018**

**13:05**

Tested the code on my device to see if it works fine. In testing, the abilities did not display, and work, as intended so the code had to be changed a bit to adjust and make the game work. Things that were changed, were mostly references which ability is active, as well as when and had to make sure that the ability is referenced correctly to the code. Also, changed that only initial touch gets registered, whereas before it was initial touch and drag, where the drag feature is disabled now due to not working well with the ability system. When drag is activated, the abilities are immediately activated.

Testing code for shooting and spread infection now as in testing it was visible that it is not working properly by shooting multiple projectiles at ones to the wrong planet with the path pointing to completely different planet. And planets being randomly infected.

**14:50**

Looking through the code, I couldn't figure out why the projectile shoot was not working properly. So, the way that the information is stored, e.g. the planet position in array and the Vector3 position, is more accurate by creating the function which is called by the spread Coroutines. It seems that the references were not working properly before, as now it is working as planned.

Next thing to work on would be to have ripple effect when the player clicks the planet.

**20:30**

Following Catlikecoding tutorial on mesh deformation, the working deformation was implemented where on click the planet would deform. It was tested on mouse clicks, but it wasn't tested on AR just yet. It is to follow, but with the instructions from the tutorial, there are no problems.

**20<sup>th</sup> April 2018**

10:50

Implemented the mesh deformation to my AR game. No issues implementing it inside the game, where every click represents the input. However, when testing I think it is not bouncy enough, therefore I increased the force to 20.0f, springForce to 100.0f and damping to 2.0f. With those settings the spring/deformation looks more satisfying.

Next will be improvements to graphics (ability) and creating different scenes. Also, when projectile is destroyed I think of creating a particle system to show it dies. (also want to show planet death if it is infected for too long.) After create a menu system for the game.

12:20

Updated the graphics for abilities, however, due to limitations of my macbook and not having separate mouse, I am unable to create graphics as I want, therefore I will leave this job until I arrive back in university. I will focus on creating the scenarios to present the game better.

#### **Things to complete before submission:**

>Particles for ability with different colours (DONE)

>Tutorial system

>Help menu

>Menu system

>Visual representation of bomb and shield ability when used

**10<sup>th</sup> May 2018**

12:30

Started working on finishing my project. I have started on creating graphical assets (UI) before completing coding side of the game. I have looked around the web for inspirations for icons to be used in the game for pause and help button, which ended up to be icons that I believe should work fine in the game environment. However, it is still not implemented in game so it has to be tested and confirmed if anything should be changed or not. Crystallised feature on photoshop was used to get good effect on the graphics which fit well with the game atmosphere.

#### Done Graphics:

>Help (active/inactive)

>Menu/Pause (active/inactive)

>Tutorial Text BG

>Click symbol

#### Still to be complete:

>Menu system  
>Help graphics (ingame tutorial)  
>Visual representation for hologram style on the base

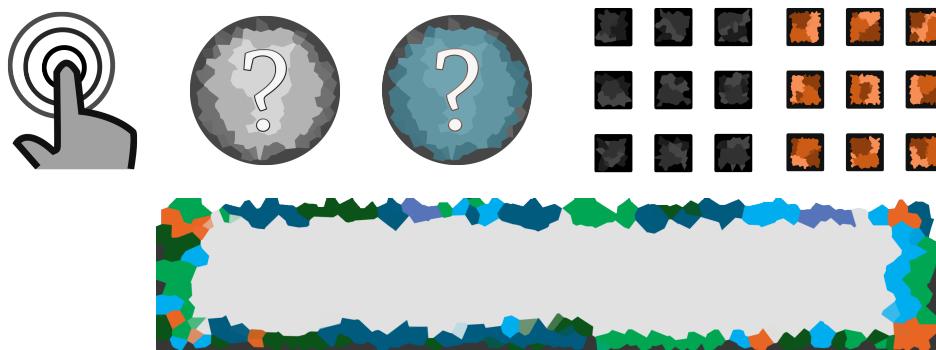
**11<sup>th</sup> May 2018**

15:25

Working on changing abilities visuals on how they present in the game. Through testing with particles, I decided to create the visuals for the planets through the particles instead of the prefab ability. Colours indicate different ability available, with shield ability also using particles. All the particles been implemented, just require to be tested. Testing will be complete once the tutorial system is implemented. Testing will follow on that.

When implementing particles, there weren't any issues implementing, just the case of playing with different values to make sure that it has effect that I am looking for, looking and comparing different effects to eachother.

UI to be soon implemented, hopefully tomorrow, to test it and implement it.



**12<sup>th</sup> May 2018**

14:30

Implemented the code for the choice of game start or tutorial, where tutorial sequence starts if tutorial button is pressed. The sequence still has to be edited to fit the tutorial. UI and everything else to be created after the coding is done. Looking to test the implementation tomorrow once, when the whole prototype implementation is complete. Testing will be done at beginning of next week, with finishing report.

Now there is a variable of bool type that tracks if menu is turned on or off, if its on then the game is not interactable.

19:30

Implementing time sequence for tutorials, set everything up on the UI script, and edited spawn script, implementing gamemanager script, where found out that lambda expression had to be used to make WaitUntil() work. Simple to implement and is working in the code, to be tested tomorrow. Made sequence notes to reference what has to be implemented in the code.

Still the code has to be edited to implement and use the new abilities representation. To be done after the sequence tutorial is complete.

**20:00**

Timed sequence is implemented, meaning tutorial section is complete. Have to assign variables in the editor, and test it, but in theory the coding is complete. Now edit code to take advantage of the new visual abilities.

**20:40**

Implemented new ability visuals in the code. It is also assigned in the unity editor. Still to be tested. There were no issues, or any problems implementing the code for either the tutorial sequence or the visuals. Had to update some functions, and add some to make it work properly between different scripts. Also, fixed issue where the abilities did not refresh. Now abilities refresh every 5 seconds. There are no compiler errors – good sign.

**13<sup>th</sup> May 2018**

**18:00**

Upon testing, there are multiple different errors and bugs that pop up:

- > Abilities not always showing
- > Not always able to click on the planet to deform
- > Shielded ability not showing particles (at all)
- > Spread occurs to random planets, projectiles fly at random locations, paths don't go to correct places
- > Infected planet counter is incorrect
- > Bomb not working as intended

The only thing that seem to be working as intended is Time slow ability.

**22:00**

On a way of fixing those bugs, before finishing implementing the tutorial system. Because tutorial introduces the game just in slower pace, making the game work is my priority.

Changed the way that the materials of planets change material. Before it was done in two different scripts, transferring material across. Now it is all done in Planet script. This should tell me which planets are infected more reliably as I check for tag in every frame. It does seem to fix the infected planet counter. Good news. Next part to fix is the spread indicator, to be accurate. Once that is complete, fix the abilities and then tutorial before implementing UI and menu system.

**14<sup>th</sup> May 2018**

**20:40**

Looking to continue fixing bugs, and after looking and debugging the code carefully, I noticed that the spread was inconsistent regarding path showing and projectiles shot. Turns out that one planet was not labelled correctly, and there was an issue with my math calculations that a path was created, where I multiplied by a number which was not consistent so the distance is not calculated properly between the two planets. Notice occasional bug of path not disappearing, but it is working well enough to progress – can't

find out reason for this happening, and due to time limitation other bugs have to be fixed too.

Next step in the implementation is fixing bugs with the abilities.

22:35

Bug encountered where all abilities don't work properly. This bug is in the form that once any ability is activated, no planet can be clicked afterwards, only projectiles can be clicked and after projectile is clicked, the game goes back to normal.

Thinking of changing projectile prefab to asteroid as I think it fits to the game in general more.

22:45

Found bug which caused not ability to click on the planets after ability was activated. Reasoning was that the code was written in wrong order, where variable was changed to incorrect state which caused confusion. Everything is in the correct order now. Time ability is tested and is working as intended.

Now looking to examine bomb ability and shield ability.

**15<sup>th</sup> May 2018**

20:20

Working on bomb ability, and the starting position implementation of where the bomb start is changed, so instead of Vector3 position, the planet number location in the array is passed through and the bomb gets position from that by transform.position. Also, noticed that the ability couldn't be chosen (twice if it was the same). This was fixed by resetting active ability to 0, as when ability is picked up it checks if it is active ability so it couldn't be picked up.

Results being that it seems like the bomb is fixed, and is working, most of the time. It is still getting stuck at random moments and can't figure out the reason why, but for the time being it will have to do.

The bomb prefab turns on and it won't turn off (most of the time), and if the planet is inside the bomb (not touching the vertices), then the planet will still be infected. This will be adjusted with the blast radius of the bomb, but still something that has to be kept in mind. One solution that I am thinking of doing at the moment is just making the texture transparent, since it is still playable, just the prefab won't turn off.

21:00

Was considering checking the collider to see if they are correct, since it seems like that is the place where it seems to be freezing up, however nothing could be found. Therefore, for the game prototype implementation, the texture was turned down to transparent so it doesn't affect gameplay, since the game can be still played when using this ability.

Considering whether to move towards implementing UI, and tutorial session, or should I work on shield ability.

22:35

Tried to find a solution to bomb not working properly, and after trying out different implementations, tried Coroutines, different counter, reference to the collider, using separate script on the bomb with “OnCollisionEnter” and every one of those tries either give the same, worse, or no results at all. Therefore, the implementation went to the original implementation, and just going to accept that sometimes it does not work. It is a bug that due to time, I don't have time to fix.

23:35

Was looking at shield code, and just tried to see how it operates and doesn't work, and somehow, I changed the reference on the number that the shield affects and it works. Although, find that 5.0f goes very quickly, so it is increased to 15.0f to see if it works as intended. UI implementation will follow, followed by tutorial finished implementation (barebones for tutorial are setup). Added a like so the screen never goes black, doesn't go sleep.

### ***16<sup>th</sup> May 2018***

16:15

Done graphics for “Setting”, “Main Menu”, “Restart”, and implemented basic UI for the game. New scenes were created too, now having Tutorial, Game, Menu screen, where menu starts the game. Everything works as intended without any issues. Settings page will only contain Volume slide, as no other settings are required that I can think of. Therefore, the settings page in the pause screen will be replaced by the volume slider. Rest of graphics will be completed tomorrow and implemented. All transitions work, and settings page will be implemented tomorrow too.

### ***17<sup>th</sup> May 2018***

18:50

Implemented all graphics, implemented and completed tutorial, made volume slider with coding that works in between different scenes. Testing completed, with final result of sometimes working and sometimes bugs out and doesn't work at all. When everything works as intended, then it works flawlessly and works well.

## Appendix H: Test Results

### First Test Results (First Prototype):

UX Game Feedback

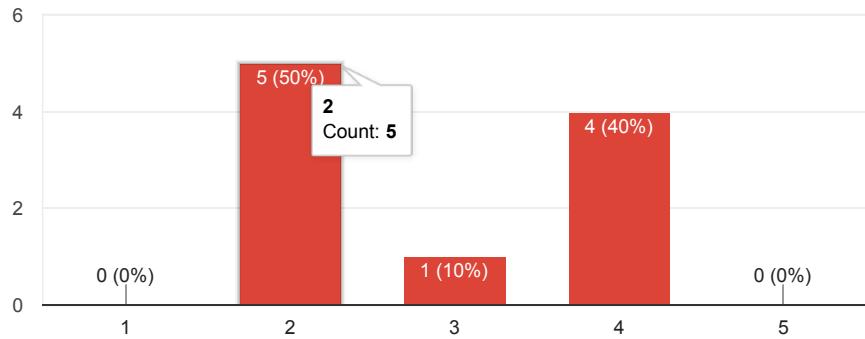
18/05/2018, 00:51

## UX Game Feedback

10 responses

Was it easy to understand what had to be accomplish / how to win the game?

10 responses



Comments - what did you like / dislike / how to improve? Any other comments?

9 responses

It was confusing (2)

Nice concept, I don't really play games and I've only played an AR game once so I had to get my head around the mechanics but it was easy enough to understand once I'd played through it once.

Once you have played it once or twice it is intuitive to understand

Game was fast paced which feels good for an AR game

It was easy to understand when also explained in person but might have been hard from just the menus,

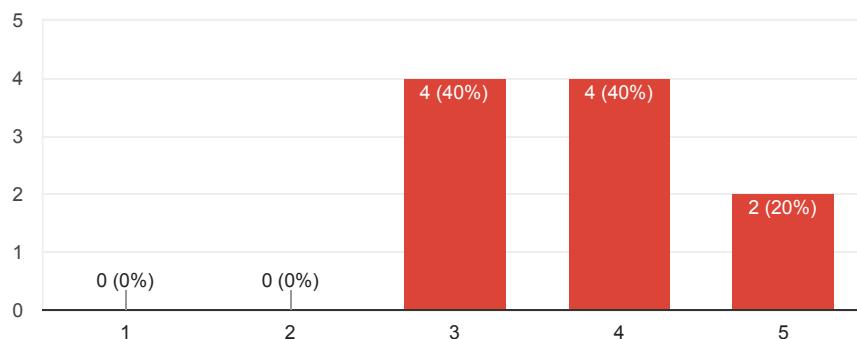
The ingredient is not shown in the game screen. You have to look for it in the menu.

Unclear objectives in the game.

Unclear. No warning, thing happen all of a sudden.

### Was interaction with the game logical? (The tap screen and the orientation that the game was implemented in)

10 responses



## Comments - what did you like / dislike / how to improve? Any other comments?

9 responses

instructions need to be shorter and clearer (2)

I understood what to do, I just wasn't always sure what to click on (the world or the square).

Not fond of the spheres appearing directly inside of where you are standing - it's a little confusing

Color of buttons could be hard to see depending on lighting

Yes, good orientation and intuitive to tap

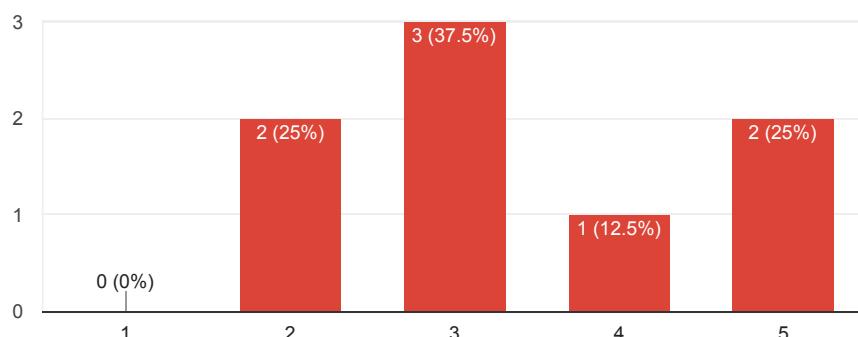
I hope the interaction menu on the planet is more distinctive and always face to the user.

Logical interaction, but it did involve unnecessary actions

Interaction made sense.

## Did it make sense how things are represented in the game?

8 responses



## Comments - what did you like / dislike / how to improve? Any other comments?

9 responses

Yea, I liked the idea of the planets (2)

Colour co-ordination made it easy to understand e.g. red = bad, green = good.

It's very clearly represented - only improvement would be more UI on the main screen to show you how many resources and weapons you have collected

Green and red are very distinct colours, easy to differentiate

It was a bit hard to tell exactly what was happening with the infection

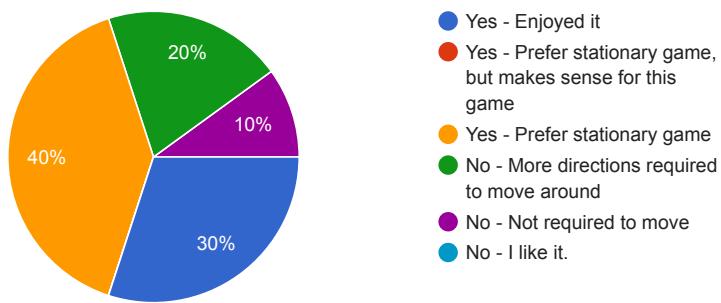
Same comment as above

There were additional clicks that had to be done that didn't have to be.

No warning, things are shown in menu only when planet is selected, no way to see what planet has available beforehand.

## Were you encouraged to move around while playing game?

10 responses



## Comments - what did you like / dislike / how to improve? Any other comments?

6 responses

I think it would have been better if I'd played it in a bigger room.

Would be better to have planets be a minimum distance before being intractable

I liked moving around and the spatial element a lot

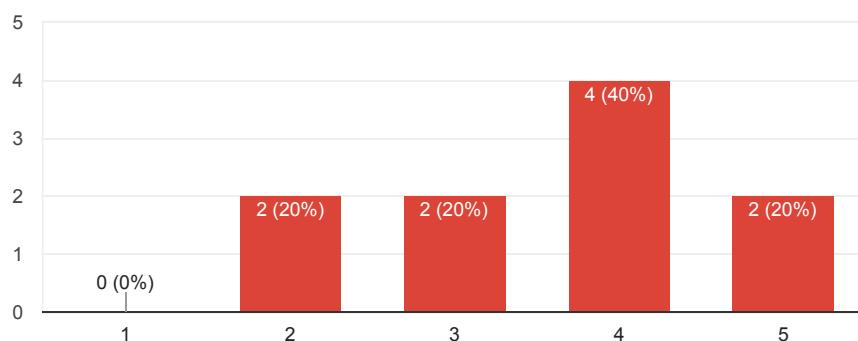
Maybe the planets occupy too much space

Could stand in the middle and touch any planet.

Could hit any planet from any position.

## Is the UI (User Interface - the buttons, and menus, text, those things..) clear and logical? (Ignoring the art)

10 responses



## Comments - what did you like / dislike / how to improve? Any other comments?

8 responses

I liked it yes (2)

It wasn't complicated, I think I would be more used to it if I played games.

Pause and inventory menu combined is good for this game

Yes, the UI was very good

Don't have many UI now

UI was small around the planets, not comfortable, uncertain what had to be done.

Clear UI but not required for this game I think.

## Any other comments?

4 responses

no (2)

Good job :)

Having an indicator of which planet will be infected next will be good to avoid unexpected loss

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Additional Terms

Google Forms

## Second Test Feedback (Improved Game Implementation):

OiOi UX Game Feedback

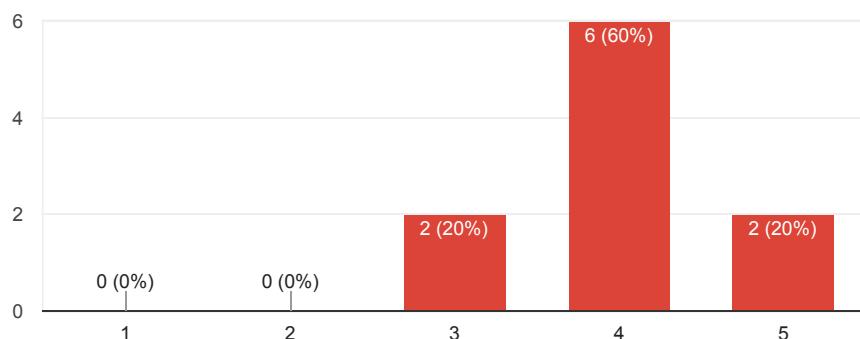
18/05/2018, 01:07

# OiOi UX Game Feedback

10 responses

Was it easy to understand what had to be accomplish / how to win the game?

10 responses



Comments - what did you like / dislike / how to improve? Any other comments?

7 responses

The tutorial are clear to mention it.

I like the tutorial showing animations of the game mechanics. It would be even better if there is more UI guiding the player in the game.

Game was clear, after reading instructions.

Yes.

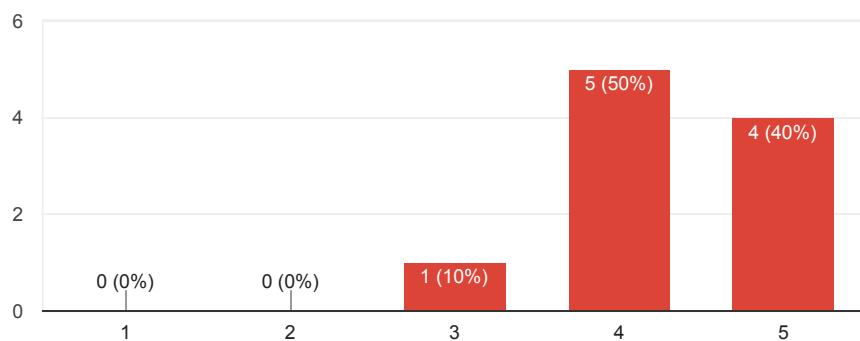
It would be good to have goal of game show up before level starts.

Lack of instructions apart from tutorial or help screen.

Simple game to play.

### Was interaction with the game logical? (The tap screen and the orientation that the game was implemented in)

10 responses



### Comments - what did you like / dislike / how to improve? Any other comments?

7 responses

The control is very intuitive, and the tapping on the planet did show interesting responses.

Clear interaction, enjoy planet deformation.

Yes

Logical game interaction.

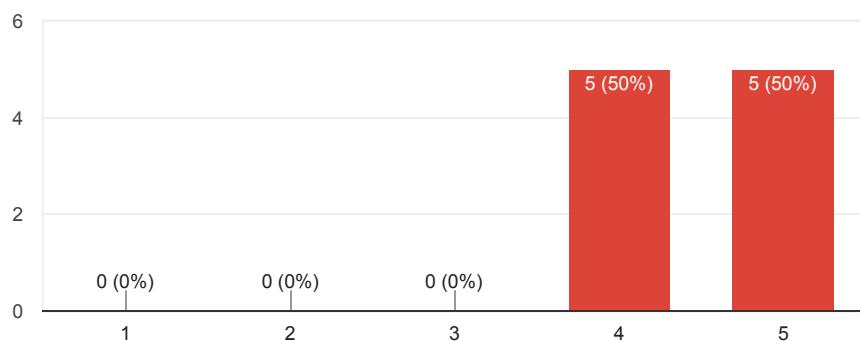
yes.

Could use more directives, where is the projectile coming from?

Fitting interactions.

## Did it make sense how things are represented in the game?

10 responses



## Comments - what did you like / dislike / how to improve? Any other comments?

6 responses

I like the particle that comes out from the planet and the cube that shows different skills, but I was

sometimes confused by the particles that show resources and the particles that shows the activated skills.

Colourful game, distinguishable colours.

Yes.

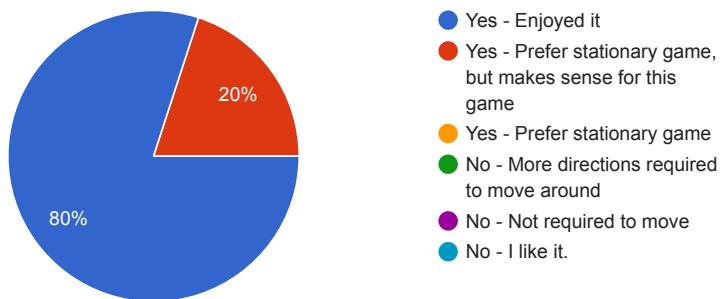
Good colour representations and particle usage.

Made sense once instructions were read.

Distinguishable colours make it clear between good and bad objects.

## Were you encouraged to move around while playing game?

10 responses



## Comments - what did you like / dislike / how to improve? Any other comments?

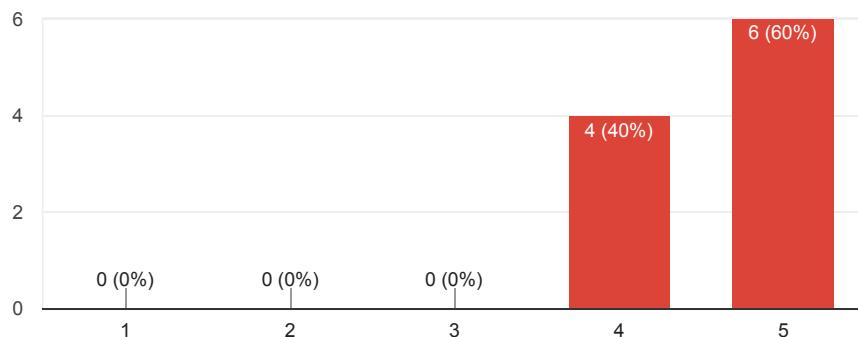
2 responses

I really enjoy moving around and just look at the planets. There is nothing to complain about.

Options of how far away planets are from each other.

**Is the UI (User Interface - the buttons, and menus, text, those things..) clear and logical?**

10 responses



**Comments - what did you like / dislike / how to improve? Any other comments?**

5 responses

The UI is colourful and lively. The only thing that I hope to improve is the grey text in one of the tutorial screen which is hard to see sometimes

Simple UI.

Good sound effects to symbolise danger.

UI fits this game.

Fitting ui.

## Any other comments?

6 responses

Good job!

Bug fixing, sometimes the game froze and had to restart.

Sometimes game doesn't respond.

good.

Apart from occasional game breaking moments, it is enjoyable game.

enjoyable experience.

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Additional Terms

Google Forms