

A Project Report on

Dead Alive

Submitted in partial fulfillment of the requirements for the award
of the degree of

Bachelor of Engineering

in

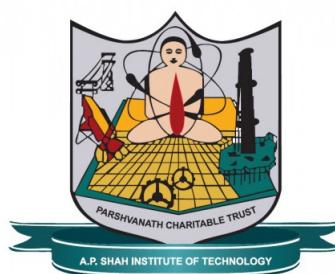
Computer Engineering

by

**Charandeep Singh(16102063)
Meet Maisheri(16102061)
Uttam Bogati(16102059)**

Under the Guidance of

Prof. Jaya Gupta



Department of Branch Name

A.P. Shah Institute of Technology

G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic Year 2019-2020

Approval Sheet

This Project Report entitled "***Dead Alive***" Submitted by "***Charandeep Singh***"(***16102063***), "***Meet Maisheri***"(***16102061***), "***Uttam Bogati***"(***16102059***),is approved for the partial fulfillment of the requirement for the award of the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Prof. Jaya Gupta
Guide

Prof. Sachin Malve
Head Department of Computer Engineering

Place:A.P.Shah Institute of Technology, Thane

Date:

CERTIFICATE

This is to certify that the project entitled "*Title of project*" submitted by "*Charan-deep Singh*" (16102063), "*Meet Maisheri*" (16102061), "*Uttam Bogati*" (16102059), for the partial fulfillment of the requirement for award of a degree *Bachelor of Engineering* in *Computer Engineering*, to the University of Mumbai, is a bonafide work carried out during academic year 2017-2018.

Prof. Jaya Gupta
Guide

Prof. Sachin Malve
Head Department of Computer Engineering

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane

Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Charandeep Singh 16102063)

(Meet Maisheri 16102061)

(Uttam Bogati 16102059)

Date:

Abstract

The goal of this project is to prove the development of a videogame using Unreal Engine, based on an agile methodology that is viable in an economic, quick and sustainable way. This methodology has four stages that are: preproduction, production, testing and postproduction that were advantageous to finish the project on time. To achieve this, we have developed a thrilling multiplayer cooperative third-person shooter featuring massive swarms of artificially intelligent zombies that recklessly rush their living prey. Focused on fast-paced gameplay. In conclusion, we achieve to prove the applicability of the four stages methodology since we made a high quality game in a short period of time, using limited resources.

Contents

1	Introduction	1
1.1	Objectives	1
1.2	Problem Definition	1
1.3	Scope	2
1.4	Technology Stack	2
1.5	Benefits for environment	2
1.6	Applications	3
2	Literature Review	4
3	Planning and Implementation	6
3.1	Planning	6
3.2	Pre production	6
3.3	Production	6
3.4	Post production	7
4	Project Design	8
4.1	Proposed System	8
4.2	Design or Flow	8
4.3	Activity Diagram	8
4.4	Use Case Diagram	10
4.4.1	Description Of Use Case	11
5	Result	12
5.1	In-game Snaps	13
5.2	Blueprints (Note: Not all blueprints are shown here)	20
6	Conclusions and Future Scope	26
Bibliography		27
7	Annexure A	28
7.1	Gantt Chart	28
8	Acknowledgement	29

List of Figures

4.1	Activity Diagram	9
4.2	Use Case Diagram	10

Chapter 1

Introduction

Benefits of using a game engine like Unreal Engine are: reusable code with libraries, the concept of object oriented programming and process of computer generated graphics. The engine has specialized libraries for game development. In addition, the use of object oriented programming, makes it possible to create class the traditional way, but in a very intuitive way such is the case with the Character class, this class contains all the necessary code to create a character on the game. In a multiplayer session, game state information is communicated between multiple machines over an internet connection rather than residing solely on a single computer. This makes multiplayer programming inherently more complex than programming for a single-player game, as the process of sharing information between players is delicate and adds several extra steps. Unreal Engine features a robust networking framework that powers some of the world's most popular online games, helping to streamline this process. This page provides an overview of the concepts that drive multiplayer programming, and the tools for building network gameplay that are at your disposal.

1.1 Objectives

The objective of this project is to build a thrilling multiplayer cooperative third-person shooter featuring massive swarms of artificially intelligent zombies that recklessly rush their living prey. To prove the applicability of the four stages methodology, which is preproduction, production, testing and postproduction.

1.2 Problem Definition

To prove the development of a videogame using Unreal Engine, based on an agile methodology that is viable in an economic, quick and sustainable way. This methodology has four stages that are: preproduction, production, testing and postproduction that were advantageous to finish the project on time.

1.3 Scope

The Scope of our game is limited to survive from Zombie Apocalypse on an Island. We will be having a limited materials for our survival. We can tackle the Zombie using weapons which we found on Island, there is no option to buy weapons or other stuffs. The motive of the game is to survive as well as save the Lighthouse from Zombies capturing it.

1.4 Technology Stack

- Unreal Engine 4 - The Unreal Engine is a game engine developed by Epic Games, first showcased in the 1998 first-person shooter game Unreal. Although initially developed for first-person shooters, it has been successfully used in a variety of other genres, including platformers, fighting games, MMORPGs, and other RPGs.
- Adobe Fuse and Adobe Mixamo- It is a 3D computer graphics software developed by Mixamo that enables users to create 3D characters. Its main novelty is the ability to import and integrate user generated content into the character creator.
- Blender -Blender is a free and open-source 3D computer graphics software toolset used for creating animated films, visual effects, art, 3D printed models, motion graphics, interactive 3D applications, and computer games.
- Adobe 3DS Max- Autodesk 3ds Max, formerly 3D Studio and 3D Studio Max, is a professional 3D computer graphics program for making 3D animations, models, games and images. It is developed and produced by Autodesk Media and Entertainment.
- Photoshop CC -Adobe Photoshop is a raster graphics editor developed and published by Adobe Inc. for Windows and macOS. It was originally created in 1988 by Thomas and John Knoll. Since then, this software has become the industry standard not only in raster graphics editing, but in digital art as a whole.

1.5 Benefits for environment

- Agents of change: youth must be directly involved in message delivery rather than simply act as receivers because empowering young people to be agents of change is likely to lead to longer-lasting behavior change.
- Fostering a relationship with nature: although video games can serve as a channel through which important environmental and social messages can be communicated, it is important not to forget about the importance of physically being in nature. “We need to get people on screens into the green and the green onto screens,” said panelist

Inger Andersen.

- Influencing behavior: measuring outcomes and impact is a priority. There should be a way to measure whether, for example, “green nudges” in games actually impact real-world behaviour.

1.6 Applications

- Gaming technology has been used for over 35 years in the military. For instance, the army has virtual gun ranges set up with M16s that have been converted into what are essentially laser guns, which register on a screen displaying a virtual world of “bad guys” and “good guys” that trainees must learn to distinguish between. If you were to walk in on this type of training, you could easily see the association with gaming technology.
- Gaming technology has been used to augment teaching finger placement and fast typing to grade school children. For instance, it might be Mario that runs along the computer screen as you successfully type the phrase or letters that appear below him. Too slow and Mario won’t make it over the lava river to rescue Princess Peach! This use of gaming technology might seem a bit obvious, but is similar to most of the learning tablets for kids that are loaded with fun games centered around reading, grammar, math, science, and even more advanced subjects like robotics.
- Video games especially action games, have proven to be able to capture the player’s attention for the entire period of the game. This is brought about by the player’s need to achieve certain objectives within the game, and be able to progress to the next level.
- An action game, for example, may require you to be very observant. It requires you to be able to move your joystick or keys while looking at the various features on your screen such as energy levels, oncoming adversaries, ammunition left, available time among other factors, all which are vital to winning. This ensures that the player can observe and react accordingly to all requirements of that particular game.

Chapter 2

Literature Review

- Title- Developing a Videogame using Unreal Engine based on a Four Stages Methodology
- Author - Carlos Mauricio Torres-Ferreyros, Matthew Alexander Festini-Wendorff, Pedro Nelson Shiguihara-Juárez
- About the paper - The goal of this project is to prove the development of a videogame using Unreal Engine, based on an agile methodology that is viable in an economic, quick and sustainable way. This methodology has four stages that are: preproduction, production, testing and postproduction that were advantageous to finish the project on time. To achieve this, we have designed and developed an action platform game following the previously mentioned stages. In conclusion, we achieve to prove the applicability of the four stages methodology since we made a high quality game in a short period of time, using limited resources.

- The main benefits of using a game engine like Unreal Engine are: reusable code with libraries, the concept of object oriented programming and process of computer generated graphics. The engine has specialized libraries for game development, it's not about thinking how to develop a game, it's about using the offered code to focus on the idea and not the technical details. In addition, the use of object oriented programming, makes it possible to create class the traditional way, but in a very intuitive way such is the case with the Character class, this class contains all the necessary code to create a character on the game. Finally, we use complex algorithms to calculate the graphics the game needs, Unreal Engines uses the Disney's Physical Based Rendering [11] to calculate the lights and shadows on real time, so it isn't necessary to program on a low-level programming language, you can focus on decorating the game.
- The article is organized in five sections. In section two, articles that used the Unreal Engine technology and its alternatives will be analyzed, to emphasize the positive and negative aspects of each project, likewise, a comparative analysis with the Vermillion Project will be made to rescue the negative and positive aspects of the Project, as well as the similarities with the rest of the designs. On the third section, Project Vermillion is proposed, an action platform game designed and developed using Unreal Engine game engine. In the fourth section, Vermillion will be validated if the methodology used in game development can reach a high quality visual standard, gameplay and user experience. Finally, the last section will present the conclusions.

Chapter 3

Planning and Implementation

3.1 Planning

- Requirements clarification: We talk about our ideas and requirements to get a technical vision.
- Budget and length estimation: We calculate the time and the cost of our project.
- We will decide the flow of the production according to the agile process methodology.

3.2 Pre production

- Concept Development: The formation of the game. The first graphic materials are being created.
- Game Design: In this step, We use Adobe Photoshop to create textures for our objects. We edit 3D models using Adobe Fuse and Blender. For animating the Characters edited in Blender, we use Adobe Mixamo and Blender again for smooth animations. Then we downloaded Open source models from the Unreal Engine Store and use them as the assets in our project.

3.3 Production

- Prototyping: Before proceeding to the development, we make a working prototype of our idea. First we use basic Unreal Engine models for our project and apply our mechanisms to see how the game will look like and what changes we need to make.

- Implementation: After we get the feedback from our prototype, We use all-modern and up-to-date software and technologies to turn our idea into a working product. We use our created assets and import it into our Project which we are making in Unreal Engine.
- In-game content creation: We create the landscape and save it as a map in our Project. We use Sculpting tool in Unreal Engine to create landscapes. Then we created ocean using the open source ocean plugin. We added foliage to our scene to give life to our landscape. We created mountains and rocks to increase our landscape, then we added textures to the landscape. After the landscape is created we added a lighthouse as it is a part of our story. Then we imported our character models with textures, then we started adding mechanics to the character using the unreal engine blueprints. Firstly we added movement to our characters and the synced animations which were created in adobe mixamo and edited in blender. Then we added shooting mechanisms to the character attached with the character. Then we added our enemies which are zombies and we added the animations and the mechanism for the zombie attack and zombie movement. Then we added Zombie AI so that it can follow and attack our player. Also we added some UI elements such as player health, cross-air, gun, gun ammo. Lastly we added transformations to our character, where our human character transforms into a character with super-natural powers, so we have 3 main human characters and each character has a transformation which has different special abilities. Lastly to make our project operable and user-friendly, we made main menu, we added multiplayer support using unreal engine multiplayer method, we added lobby joining and hosting sessions.
- Testing: We play through the game and report all the bugs (if there are such). Also, we make sure all the initially-planned mechanics are implemented.
- Launch: The game is playable as multiplayer using Steam application.

3.4 Post production

- Support: We are regularly checking on our game and fixing all the sudden problems immediately.
- Updates: We provide the project with regular updates due to unavoidable obsolescence.

Chapter 4

Project Design

4.1 Proposed System

Dead Alive is a sub-genre of action video game, which often test the player's spatial awareness, reflexes, and speed in both isolated single player or networked multiplayer environments.

Here we use weapon as a firearm or some other long-range weapon, and can be used in combination with other tools such as grenades for indirect offense, armor for additional defense, or accessories such as telescopic sights to modify the behavior of the weapons. Most commonly, the purpose of our game is to shoot zombies and proceed through missions without the player character being killed or dying as a result of the player's actions.

Our shooting game is a genre of video game where the focus is almost entirely on the defeat of the character's enemies using the weapons given to the player and survive the Zombie apocalypse.

A common resource found in our game is ammunition, armor or health, or upgrades which augment the player character's weapons.

4.2 Design or Flow

- Planning
- Pre-production
- Production
- Post-production

4.3 Activity Diagram

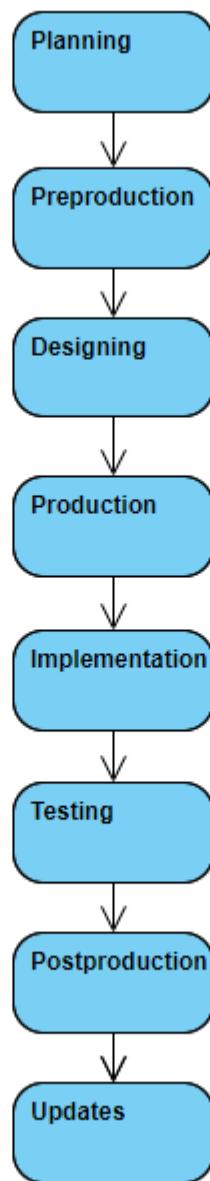


Figure 4.1: Activity Diagram

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system.

4.4 Use Case Diagram



Figure 4.2: Use Case Diagram

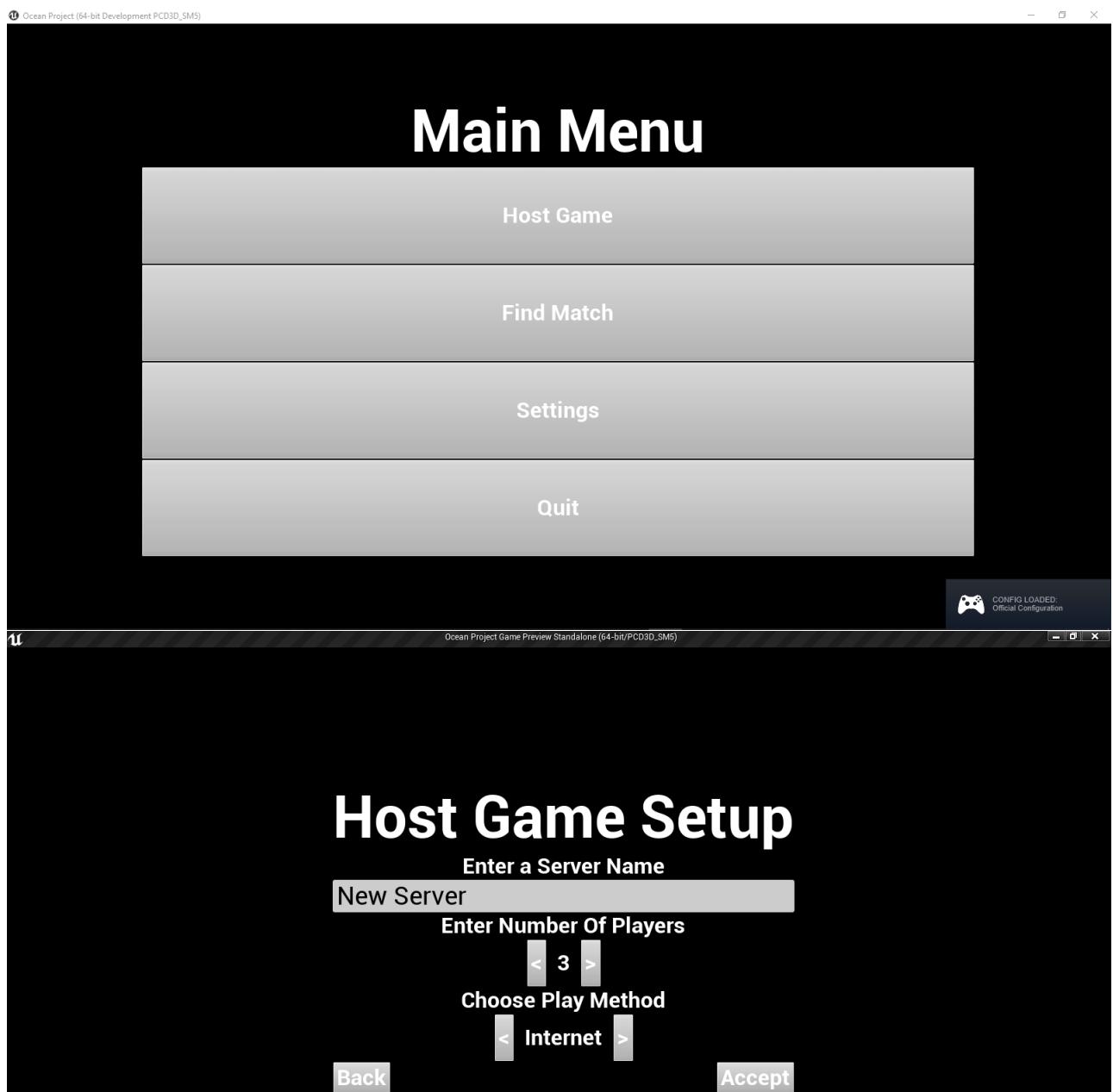
4.4.1 Description Of Use Case

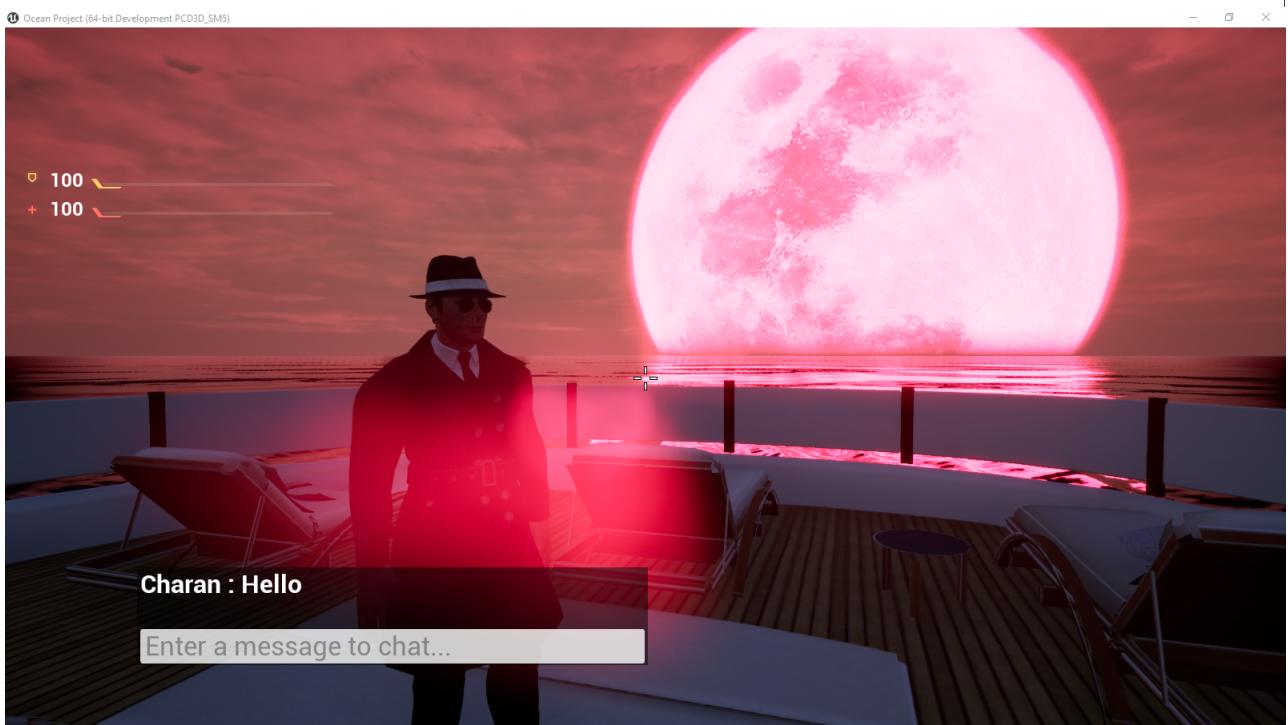
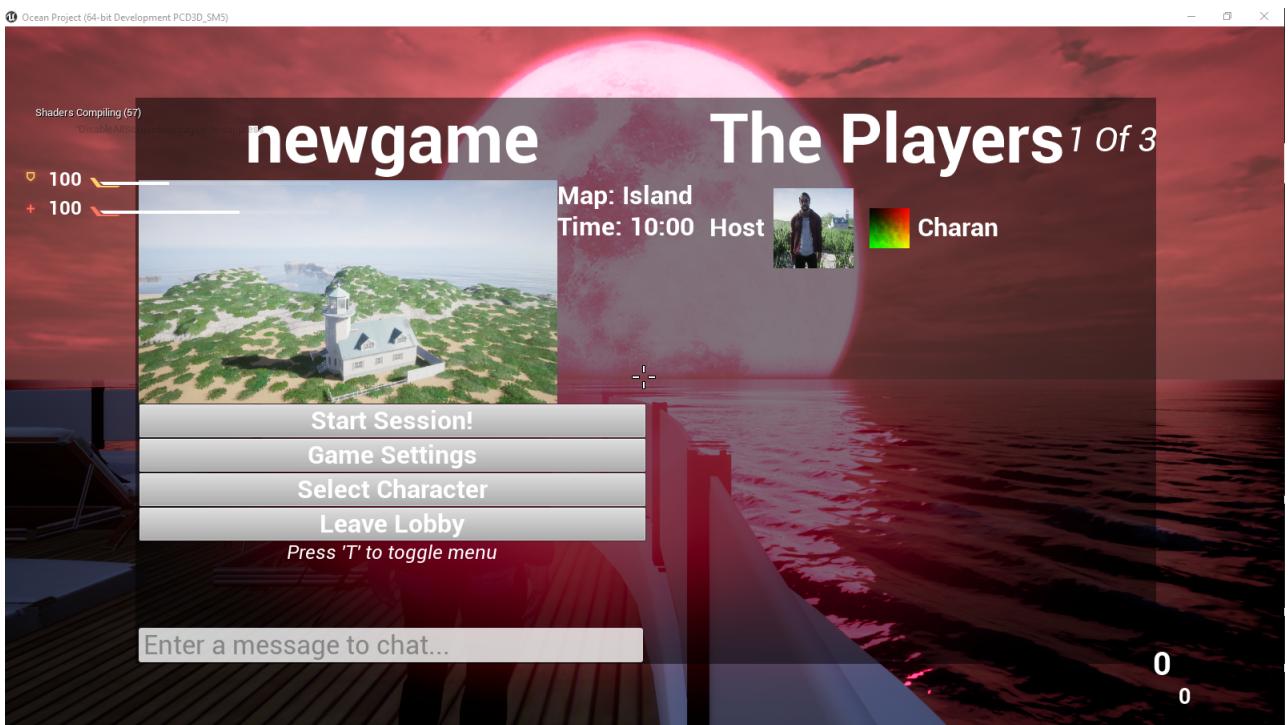
Use cases used in this project are Entering game using main menu. There we get options to create the player account which will be locally saved in the players computer, The player can Host the game making sure that the steam application is turned on. Player who has hosted can edit the map type and duration of the level. The other players can join the session and select the character and click ready. After all the players are ready, Host can decide to start the game. We have also added the chatting options in-game so that players can chat with each other. After the session is ended the host can exit the game, host can also kick the players in the game. Players can view their score at the end of the level according to the duration the player has survived.

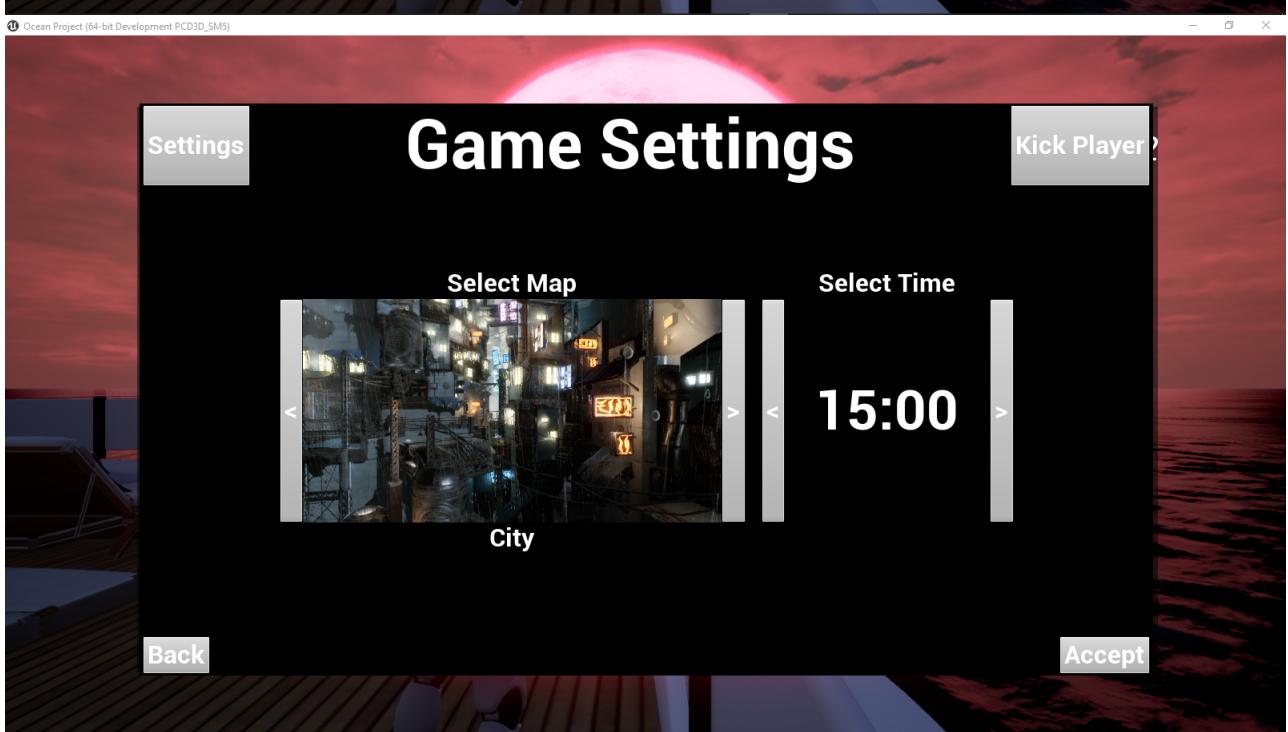
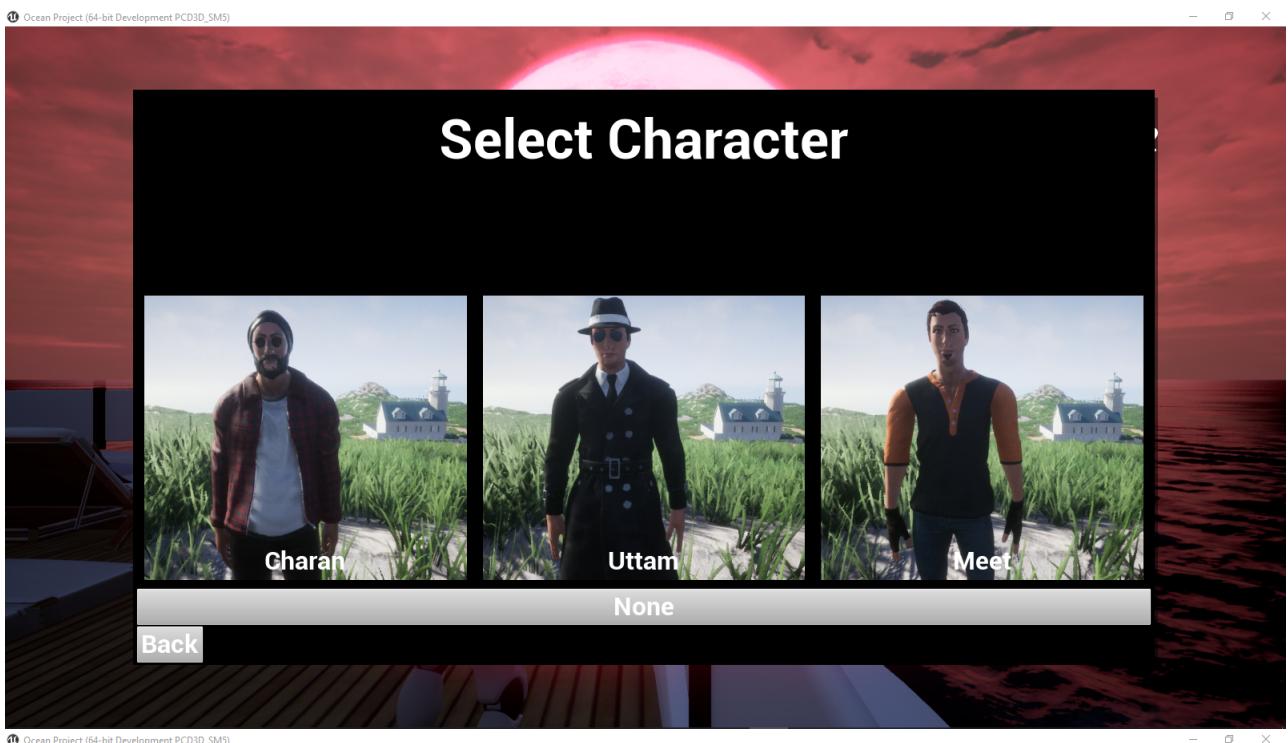
Chapter 5

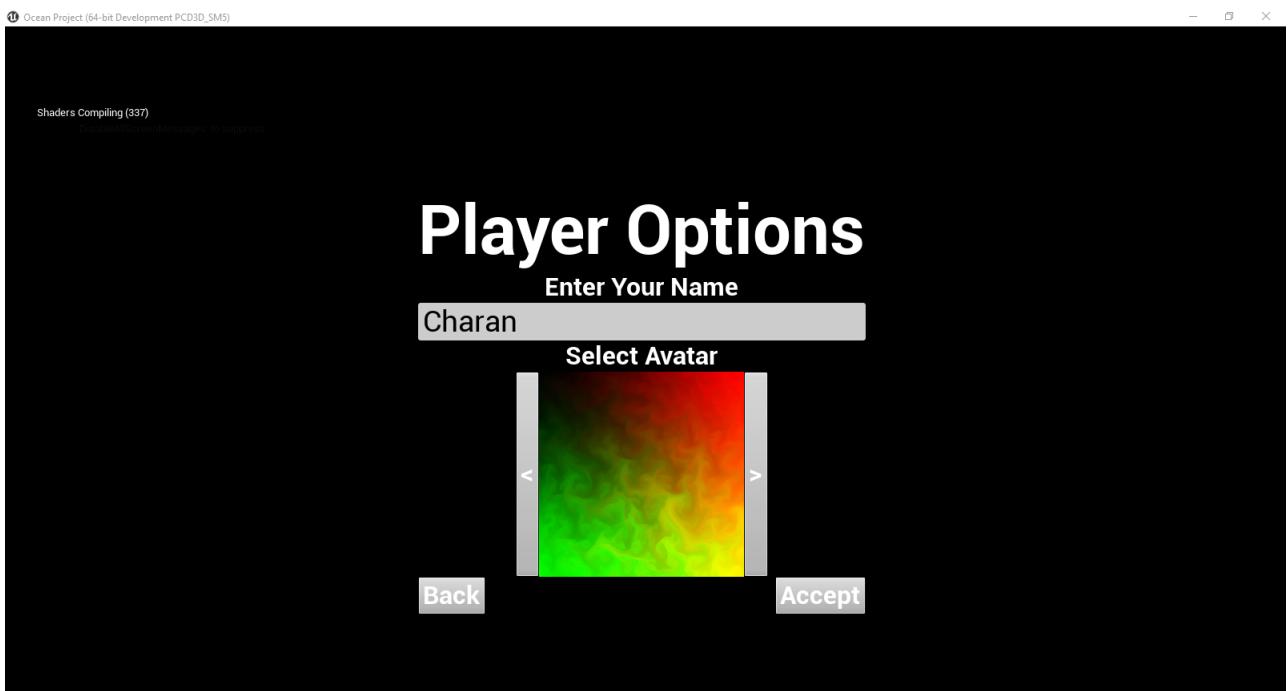
Result

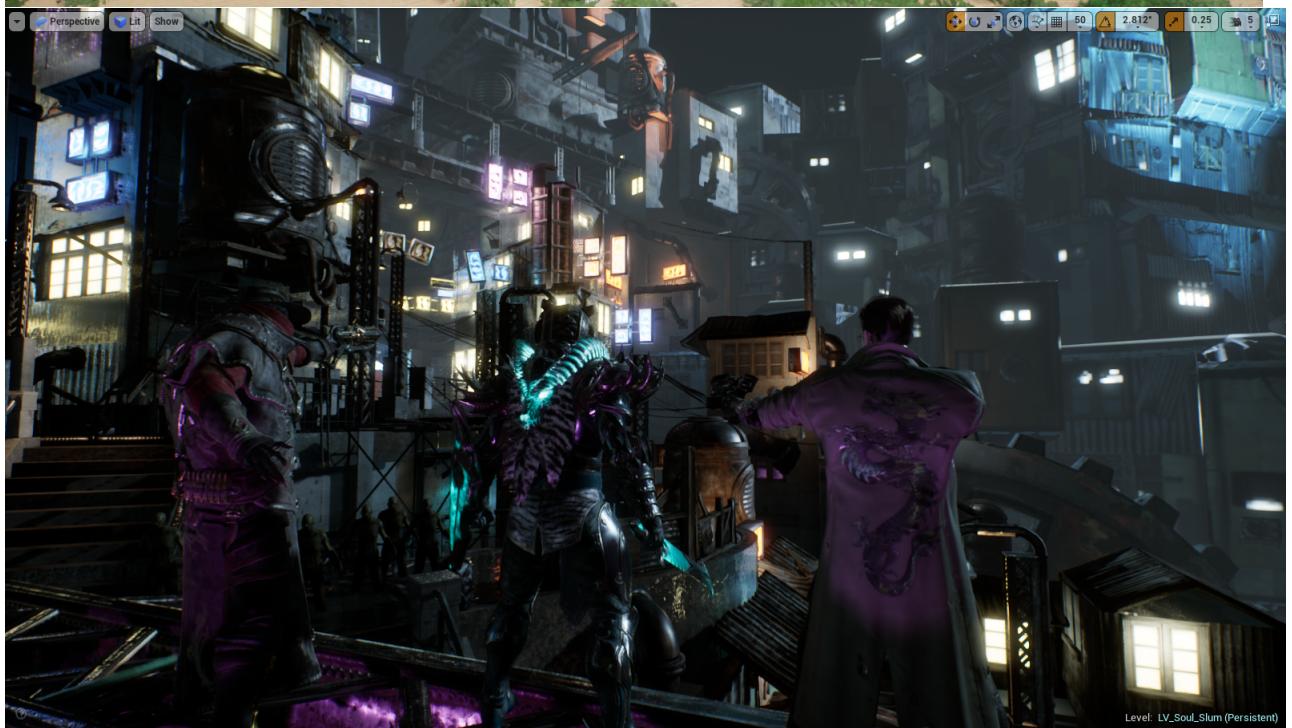
5.1 In-game Snaps







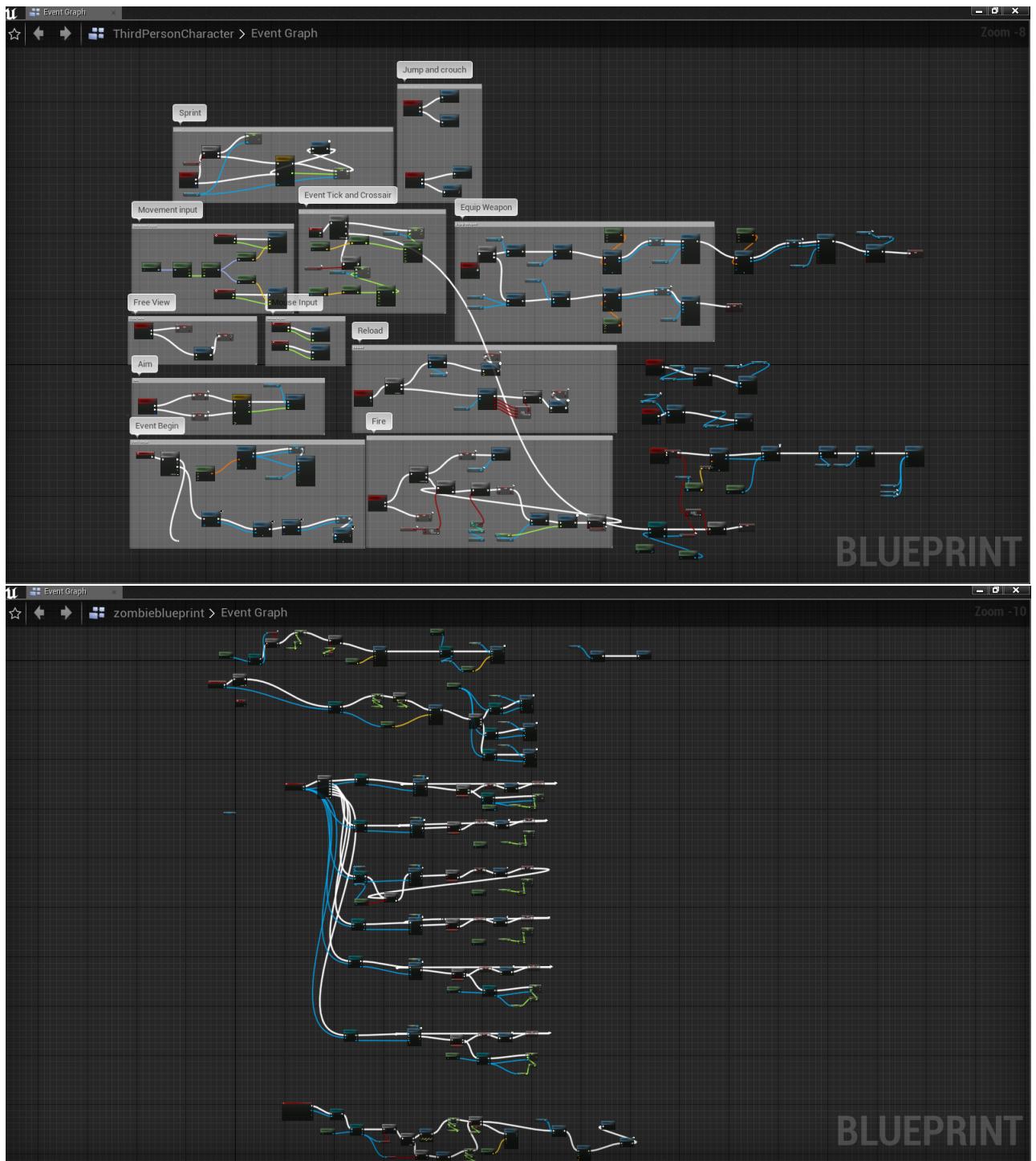


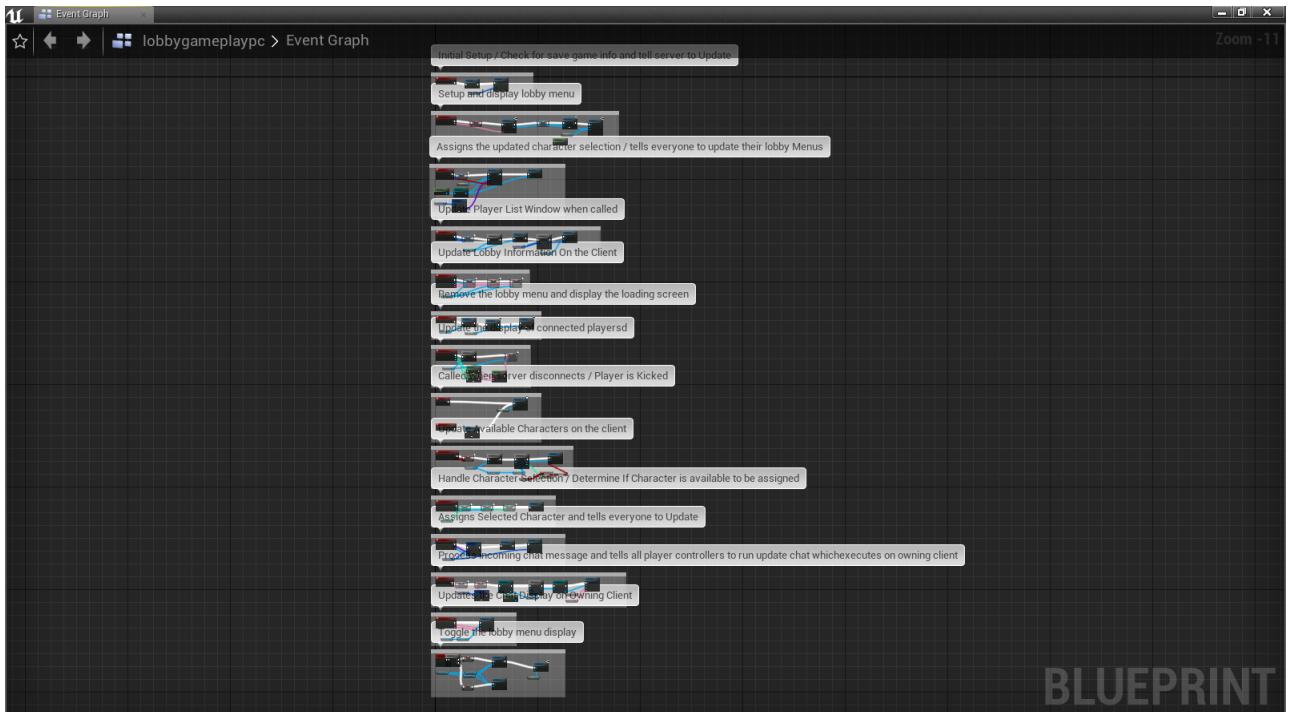




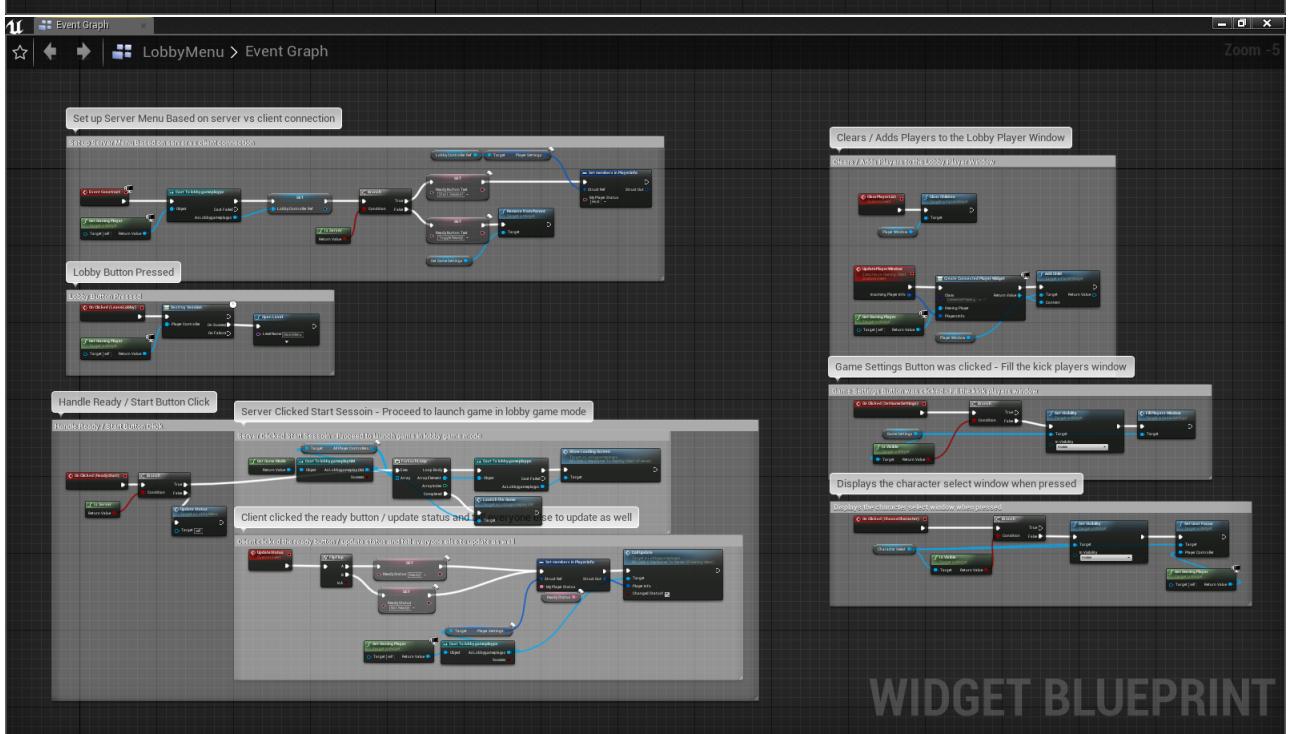


5.2 Blueprints (Note: Not all blueprints are shown here)

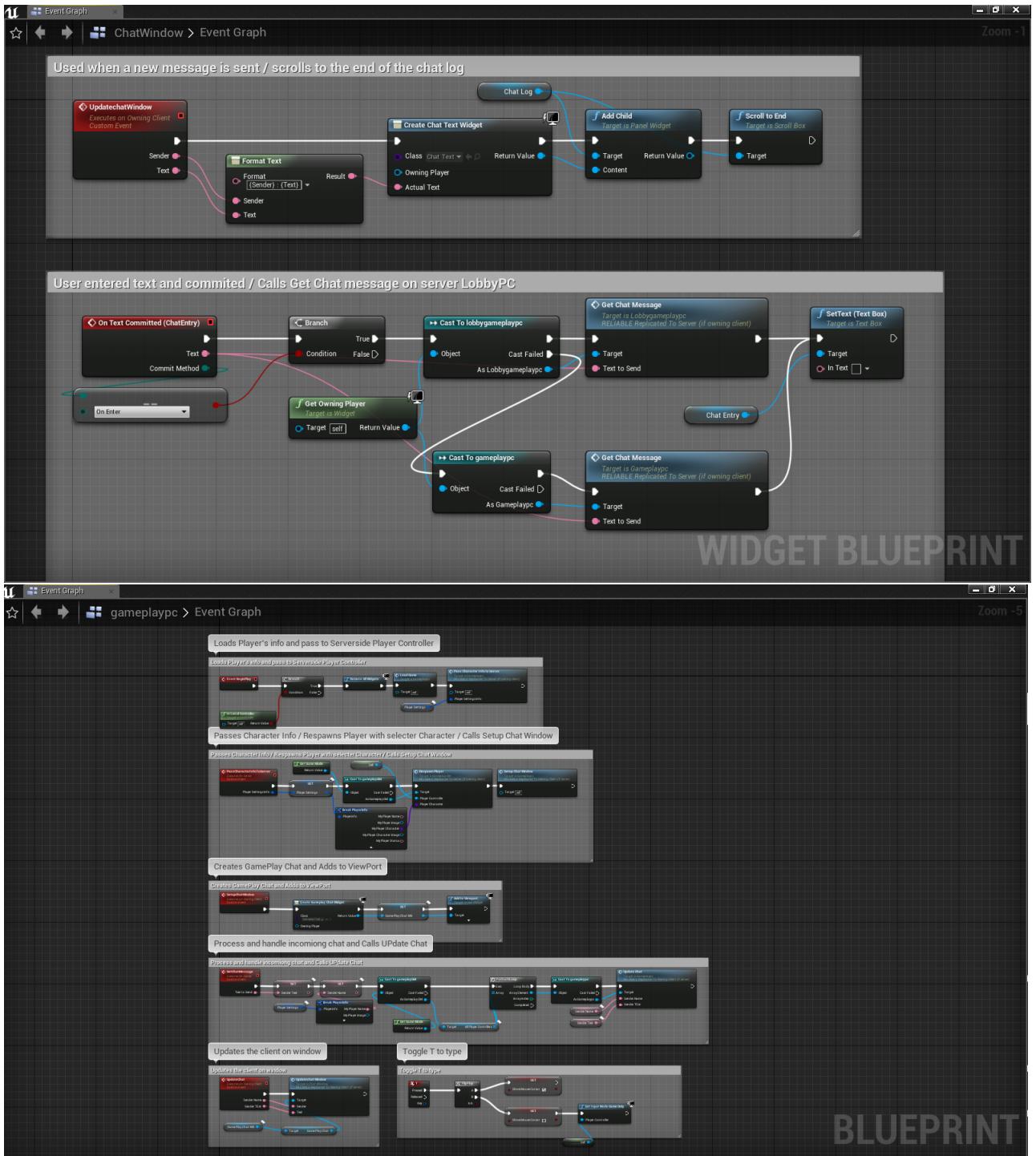


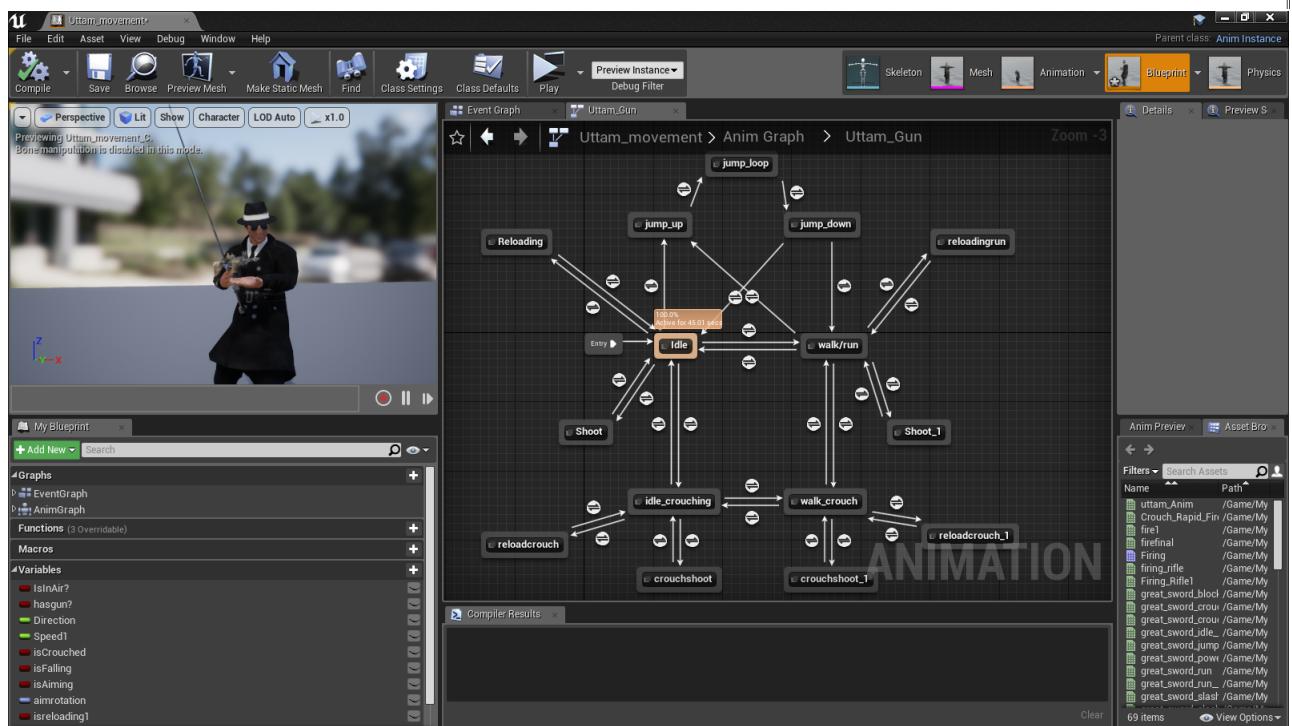
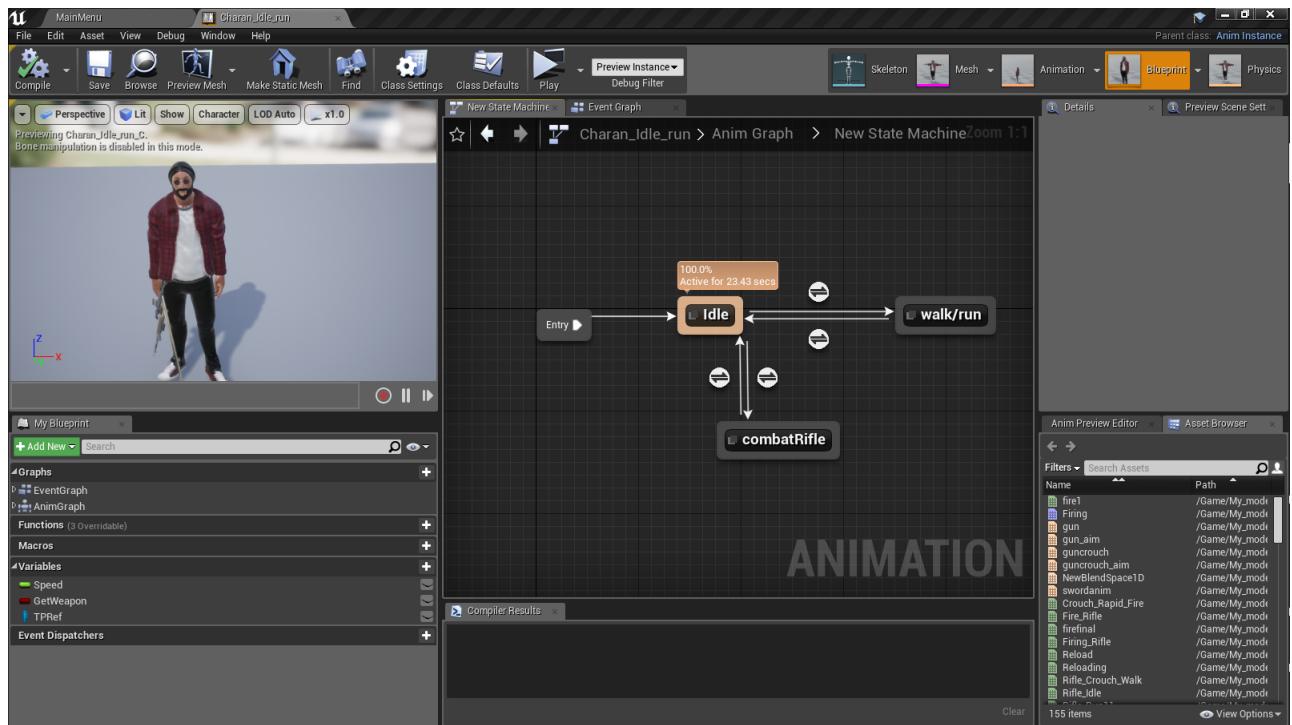


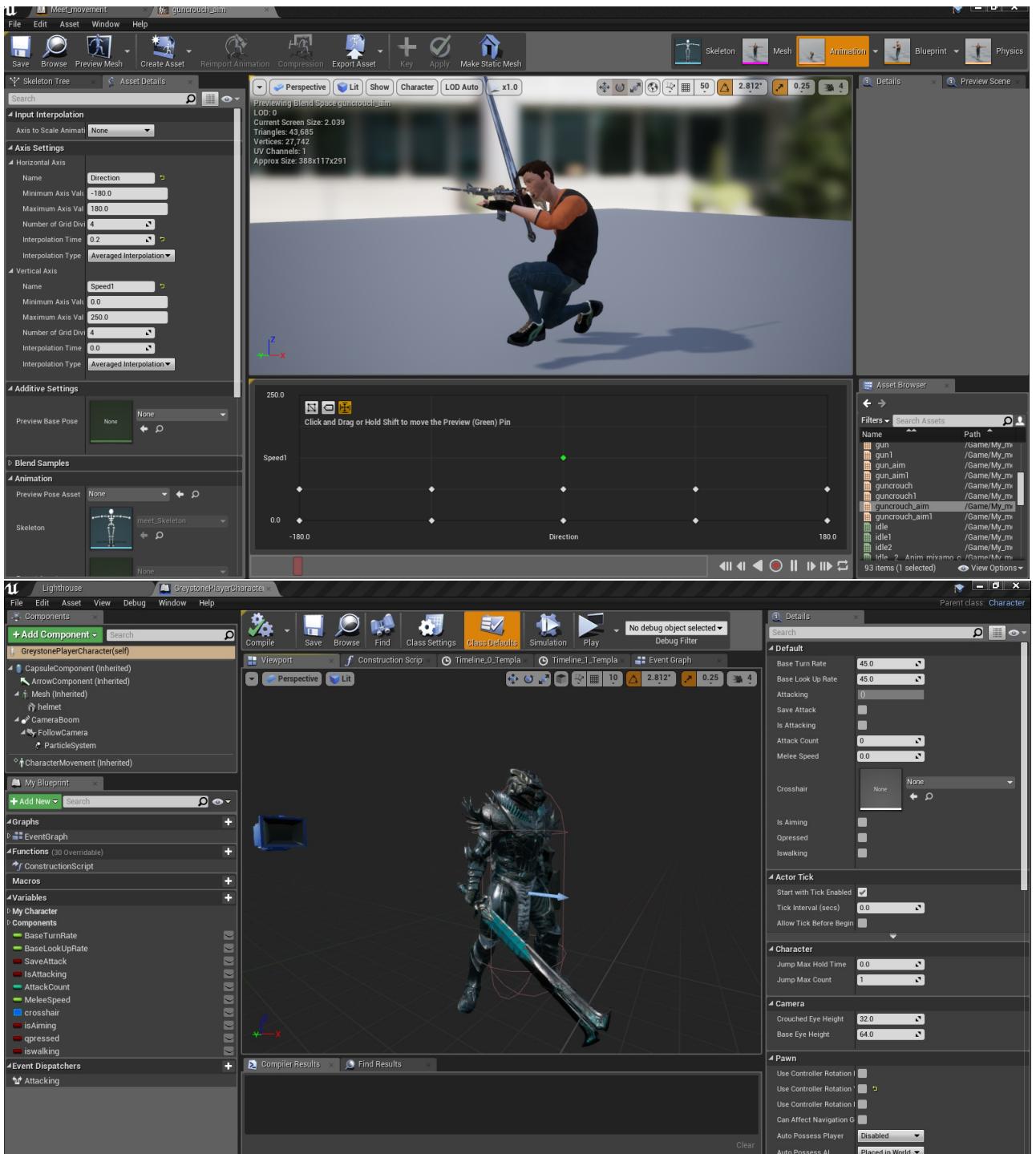
BLUEPRINT

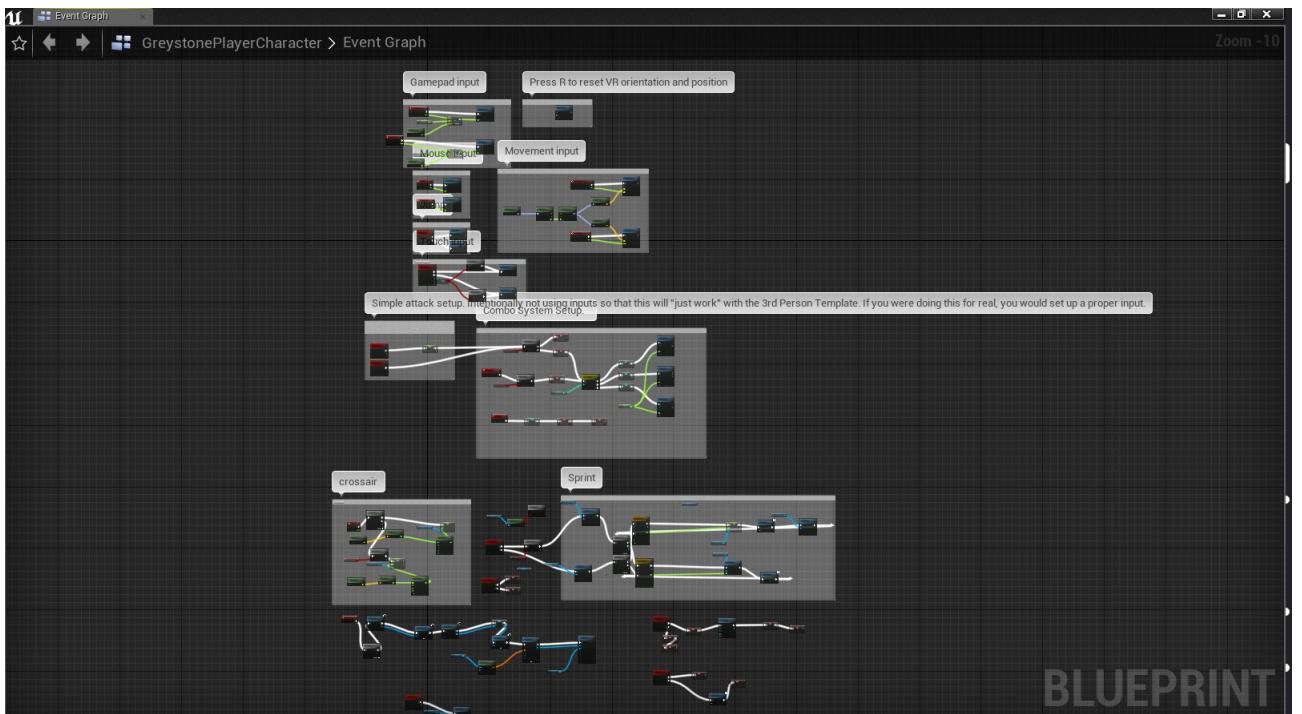


WIDGET BLUEPRINT









Chapter 6

Conclusions and Future Scope

Hence we have completed this journey of making a multiplayer zombie survival game. In future scope we decide to add support for the game to be playable at android. And to minimalise the user experience. With the multiplayer games popularity among a diverse crowd, companies are constantly working to bring new titles to the table and innovating the multiplayer games which are a sheer joy to play. In addition, Multiplayer games let people socialize and give them an opportunity to meet new people and make new friends, it's just like virtual reality, simply euphoric. Here we successfully implemented the agile methodology in our project.

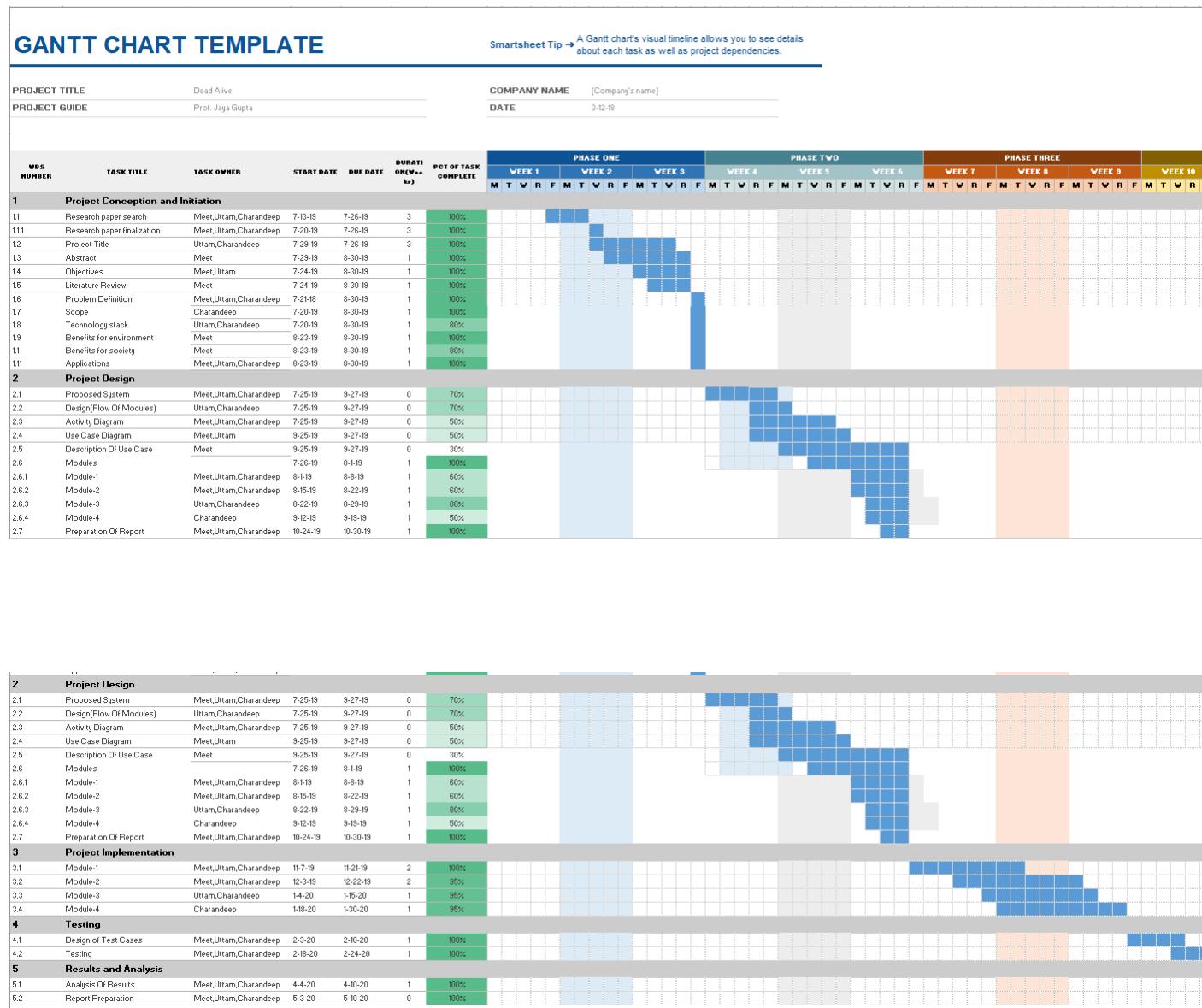
Bibliography

- [1] F. Zhu (2016, 1) Industry Outlook [Online]. Available: <http://fzdschool.com/entertainment design/industry-out look>
- [2] EPIC (2016, 1) Unreal Engine Features [Online] Available: <https://www.unrealengine.com/unreal-engine-4>
- [3] R. Kuskuntla, E. Imsand y J.A. "Drew" Hamilton, "Enhanced Expert Field Medical Training Simulations and their effect on the Modern Combat Life Saver Training Procedures", Auburn University, AL – 36849, 2006, pp 4-5.
- [4] C. Rubino "Level design optimization guidelines for game artists using the Epic Games: Unreal Engine 2", RMIT University, 2006, pp 27-30
- [5] G. Hodkinson "Taking animation project learning into the virtual environment" Massey University, 2015, pp 1-2
- [6] J. Wilson (2015, 1) Tutorial: Physical Based Rendering, And You Can Too! [Online]. Available: <http://www.marmoset.co/toolbag/learn/pbrpractice> .

Chapter 7

Annexure A

7.1 Gantt Chart



Chapter 8

Acknowledgement

We have great pleasure in presenting the report on **Dead Alive** We take this opportunity to express our sincere thanks towards our guide **Prof. Jaya Gupta** Department of Computer Engineering, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin Malve** Head of Department,Computer Engineering, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Amol Kalugade** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Student Name1: Charandeep Singh

Student ID1: 16102063

Student Name2: Meet Maisheri

Student ID2: 16102061

Student Name3: Uttam Bogati

Student ID3: 16102059