Superior Under Terror

Final Year Project

Session 2015-2019

A project submitted in partial fulfillment of the degree of

BS in Software Engineering

Department of Software Engineering

Faculty of Computer Science & Information Technology

The Superior College, Lahore

Spring 2019

Type (Nature of project)	[✓] **D**evelopment [ ] **R**esearch [ ] **R&D**
Area of specialization	Gaming
FYP ID	FYP-BSSE-F18-024

Project Group Members

Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	BSEM-F15-163	Faiq Ali	Faiqali306@gmail.com	
(ii)	BSEM-F15-162	M. Zafar Hayat	Zafarhayat412@gmail.com	
(iii)	BSEM-F15-132	M. Sajawal	Sajawalzahid101@gmail.com	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

1 Plagiarism Free Certificate

This is to certify that, I Faiq Ali S/D of Muhammad Rafique, group leader of FYP under registration no FYP-BSSE-F18-024 at Software Engineering Department, The Superior College, Lahore. I declare that my FYP report is checked by my supervisor.

Date: 22-08-2019 Name of Group Leader: Faiq Ali Signature: _____

Name of Supervisor: Mr. Ifraseab Afzal

Designation: Lecturer

Signature: _____

HoD: Dr. Arfan Jaffar

Signature: _____

Project Report

Superior under Terror

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
	1.0		<Original Draft>	
			<Changes Based on Feedback from Supervisor>	
			<Changes Based on Feedback From Faculty>	
			<Added Project Plan>	
			<Changes Based on Feedback from Supervisor>	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

_____

Name:_____

Date:_____ Signature:_____

PROJECT MANAGER

Comments: _____

_____

Date:_____ Signature:_____

HEAD OF THE DEPARTMENT

Comments: _____

_____

Date:_____ Signature:_____

Dedication

This work is dedicated to Allah Almighty our creator, our best companion, our strong pillar, our source of inspiration, wisdom, knowledge and understanding. This work is also dedicated to his most beloved Prophet Hazrat Muhammad (Peace be upon Him). He has been the source of my strength throughout this program and on His wings only have we soared. We also dedicate this work to our parents, teachers and friends who made us believe that we are able to do some brilliant work.

Acknowledgements

Special thanks to Allah Almighty and His most beloved Prophet, Hazrat Muhammad (Peace be upon Him).

We are really thankful to my supervisor, Sir. Ifraseab Afzal, for his guidance and constant supervision as well as for providing necessary information regarding the project and also for their support in completing the project.

We are also thankful to our former supervisor, Miss. Tayyaba Farhat, for her motivational support, devotion and sincerity.

We also want to thank to our parents, friends and teachers, who supported us and taught us that we can do anything if we decide.

Executive Summary

Player passes through the difficult stunt using the high level controls in an effort to kill the terrorists and to collect different collectables from the environment. Each collectable increases the score, health or bullets of player. Once the target is achieved a level gets completed. An arrow helps the user as a guide as to collect the collectable which get disappears when collected. Game can be played by one player at a time.

Our case study will show how well this component-based development approach worked on our project where we choose to develop a 3D graphics first person shooting game, named “Superior under Terror”. Some basic requirements for the mobile game were given from the beginning, such as:

3D Graphics: The game must contain 3D environment models to render the game.

Models: Some major components will be used like containers, prefabs, terrain etc.

Enemy: Enemy characters will be used.

Sounds: The sound effects will be added, like firing sound, power-up sounds, and joy sounds.

Table of Contents

List of Tables

List of Figures

Chapter 6

Appendices

Appendix A

Appendix C

Chapter 1
Introduction

Chapter 1: Introduction

This chapter is about the introduction of “Superior under Terror”. The complete project plan will be discussed in this section like WBS, roles and responsibility matrix and Gantt chart.

1.1 Background

This project is creation and implementation of game using the Android system and our Final year project. The reasons for creating the game on the Android system are many, The android market currently has the largest market share of the mobile devices and is growing, it eliminates the need for expensive hardware as most(if not all) the Android development tools are freely available. The reason for developing a Mobile game is that it is currently growing and vibrant market with more people playing mobile games than tradition games on console meaning a greater market share potential and can be developed without needing a team of people. Lastly why develop a game, A game is unlike anything we have developed before needing a lot of planning and logical thinking but the structure of the game constantly changing states presented a unique challenge and we’ve always has a keen interest games in which we why we wish to understand how they worked. This idea is more accessible and allows for greater enjoyment and this first time happening to implement a game on superior university and this first time happening to implement a game on superior university environment.

1.2 Motivations and Challenges

Motivation behind making a game is our interest in games, especially in the category of shooting games. We came across number of challenges everyday as we were not familiar with this environment. Some of them are:

1.2.1 Learning Unity basics

As we had never got idea how to make a game during our degree, not even got any session related to game development, it was a great challenge to understand how unity works and Too many game Ideas

When we started the project we had a lot of ideas to make a game. It requires a lot of brainstorming to come-up with the game that provide with the challenging yet practical solution of developing a game.

1.2.2 Game Design-Related Challenges:

Designing needs a lot of creativity, designing the progressive levels to the game with the ability to engage the player and superior university environment was one of the tough thing we come across.

1.3 Goals and Objectives

Goals of our project are

- User friendly efficient and lucrative system.
- Minimum maintenance cost (may be graphics definition).
- Availability of expected requirements within the android device configuration.
- Easy to operate.
- The game with measured coding, professional thinking.
- Game should be enjoyable yet challenging.

1.4 Literature Review/Existing Solutions

WHY DO PEOPLE PLAY GAMES?

Research on the motivations for games playing has been carried out by researchers across a number of disciplines. One of the earliest, and most cited, research works is by (Thomos Malone, 1952)(Malone 1981) who identified three main ways in which games were able to motivate players: fantasy, challenge and curiosity. Other research confirms these findings; for example, in research using educational software, Amory et al (1988) identified curiosity ("what happens if I do this") as a common motive in playing a game. Presumably the fact that something does happen encourages players to proceed and the quality of what happens in terms of user engagement is the factor that keeps them playing. The TEEM data suggests that degree of difficulty is important here; for children to enjoy playing, the game must be neither too difficult nor too hard (McFarlane et al 2002). A key concept that frequently emerges in the literature is that of 'flow', first discussed by Csikszentmihalyi (1990). This is summarized by several

researchers as “the state in which we are so involved in something that nothing else matters”, which has clear relevance to research into games and play. Debate on the issue of ‘flow’ centers around how the ‘state’ can be created in an individual, and measuring how it might make a person more receptive to receiving, comprehending and using educational-based content and skills (we will go on to discuss in more detail how ‘flow’ might apply to the design of learning games in Section 4).A 2001 survey (ESA) produced four main reasons for gameplay, namely:

87% of most frequent computer and video game players said the number one reason they play games is because it’s fun

• Games are challenging (72%)

• Games are an interactive social experience that can be shared with friends and family (42%)

• Games provide a lot of entertainment value for the money (36%).

Therefore, no clear consensus emerges on the reasons why people play digital games. This is unsurprising since the games themselves vary enormously and, as some researchers point out, the individuality of the player provides a sometimes complex set of reasons for game play. Poole (2000) notes that: “Videogames are powerful, but they are nothing without humans to play them. So, the inner life of videogames how they work is bound up with the inner life of the player.”

1.4.1 CATEGORISING GAMES

As games have become more complex in terms of graphics, complexity, interaction and narrative, so a variety of genres have increasingly come to dominate the market. There is, however, no standard categorization of such games; different stakeholders in the games industry, e.g. game outlets, developers, academics, web review sites, use a taxonomy appropriate to their own audience. Such categorizations are discussed in or want (2000), who also illustrates the system employed by Herz (1997) which closely resembles that used by many in the contemporary games industry. The Herz system presents these major categories:

• Action games - these can be subcategorized into shooting games, ‘platform’ games (so called because the players’ characters move between on-screen platforms) and other types of games that are reaction-based

• Adventure games - in most adventure games, the player solves a number of logic puzzles (with no time constraints)in order to progress through some described virtual world

• Fighting games - these involve fighting computer-controlled characters, or those controlled by other players

• Puzzle games - such as Tetris

• Role-playing games - where the human players assume the characteristics of some person or creature type

• Simulations - where the player has to succeed within some simplified

• Sports games

• Strategy games - such as commanding armies within recreations of historical battles and wars. Even with this taxonomy, there are exclusions; a small number of games will be released every year that defy categorization. In addition, some games fall into more than one category; for example, football manager games (where you buy, sell, select and position players) arguably fall into the categories of simulation, strategy and sports games. This classification also leaves out the individual or multiplayer contrast, which is making a real difference to how games can be played.

1.5 Gap Analysis

As studies has revealed that a large number of player (72%) like to play games that included challenging aspect to play game also 87% of most frequent computer and video game players said the number one reason they play games is because it's fun. So to add challenge and fun combining in the category of shooting brought us with the idea to make a game of stunts.

Moreover, no game is using Superior (The Superior University, n.d.) Environment right now. So it is also an opportunity for us. So we grab it.

1.6 Proposed Solution

As studies has revealed that a large number of player (72%) like to play games that included challenging aspect to play game also 87% of most frequent computer and video game players

said the number one reason they play games is because it's fun. So to add challenge and fun combining in the category of racing brought us with the idea to make a game of stunts.

1.7 Project Plan

One of the most complicated aspects of game development is planning. Some would argue that small projects don't need to take this step; they simply need to work on the project until it's done. This is the farthest thing from the truth.

1.7.1 Game Design and Concepts

In this project, we were left free to decide what type of game we wanted to develop. The suggestion was that a shooting game would be suitable, since such a game usually depend on advanced assets, e.g. animated models. After some brainstorming, it was decided that a shooting game should be developed. However, there were two different shooting game ideas, but this is the totally different idea about the superior under terror this could be guide them then how to protect and save our people from terrorism and how to face them all in real life experience.

1.7.2 Our Development Process

In this project, component-based development was chosen as a keystone for our

- Requirements specification
- Component analysis
- Requirements modification
- System design with reuse
- Development and integration
- System validation

The decision was mainly based on the size and the duration of the project. We had a limited amount of time, and during this time we wanted a rapid development.

When it came to deciding whether to go with throwaway prototyping or exploratory development, the latter was chosen. This decision was based on the following facts. The greatest problem was not to understand the basic requirements, but how much time it would take

implementing the many possible features. Given the circumstances, the best strategy was first to implement the game foundation, and subsequently implement one feature after another.

1.7.3 Work Breakdown Structure

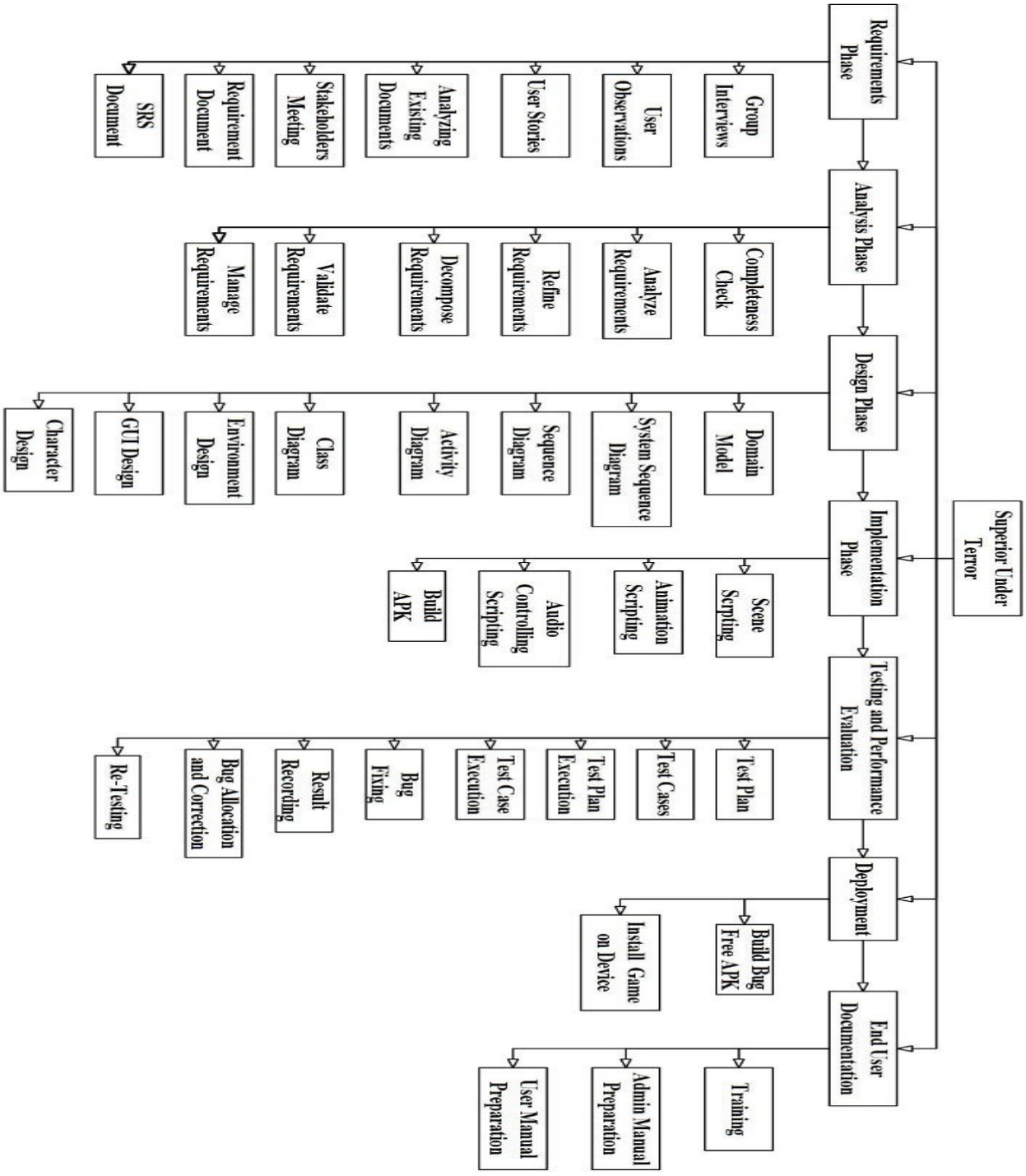


Figure 1. 1: WBS

1.7.4 Roles & Responsibility Matrix

Table 1. 1: Roles and Responsibility matrix

WBS #	WBS Deliverable	Predecessor Activity #	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1. Requirements Phase				
1.1	Group Interviews	None	1	M. Sajawal
1.2	User Observations	None	1	M. Zafar Hayat
1.3	User Stories	None	1	Faiq Ali
1.4	Analyze Existing Documents	1.2	1	M. Zafar Hayat
1.5	Stakeholders Meetings	1.1	1	M. Sajawal
1.6	Requirement Document	1.1, 1.2, 1.3, 1.4, 1.5	2	Faiq Ali
1.7	SRS Document	1.6	3	Faiq Ali
2. Analysis Phase				
2.1	Completeness Check	1	2	M. Sajawal
2.2	Analyze Requirements	2.1	2	M. Zafar Hayat
2.3	Refine Requirements	2.2	2	Faiq Ali
2.4	Decompose Requirements	2.3	2	Faiq Ali
2.5	Validate Requirements	2.4	1	M. Sajawal
2.6	Manage Requirements	2.5	1	Faiq Ali, M. Zafar Hayat, M. Sajawal
3. Design Phase				
3.1	Domain Model	2	2	M. Zafar Hayat

3.2	SSD	2	2	M. Sajawal
3.3	SD	3.2	2	Faiq Ali
3.4	Activity Diagram	3.3	3	Faiq Ali
3.5	Class Diagram	3.1, 3.2	2	M. Zafar Hayat, M. Sajawal
3.6	Environment Design	3.4	3	Faiq Ali
3.7	GUI Design	3.4, 3.5	2	Faiq Ali, M. Sajawal, M. Zafar Hayat
3.8	Character Design	3.7	3	Faiq Ali
4. Implementation Phase				
4.1	Scene Scripting	3	5	M. Zafar Hayat
4.2	Animation Scripting	3	5	Faiq Ali
4.3	Audio Controlling Scripting	3	5	M. Sajawal
4.4	Build APK	4.1, 4.2, 4.3	2	Faiq Ali, M. Sajawal, M. Zafar Hayat
5. Testing and Performance Evaluation				
5.1	Test Plan	4	3	M. Sajawal
5.2	Test Case	4	3	M. Zafar Hayat
5.3	Test Plan Execution	5.1	2	Faiq Ali, M. Sajawal
5.4	Test Case Execution	5.2	2	Faiq Ali, M. Zafar Hayat
5.5	Bug Fixing	5.3, 5..4	3	Faiq Ali
5.6	Result Recording	5.5	2	Faiq Ali

5.7	Bug Allocation and Correction	5.6	3	Faiq Ali, M. Sajawal, M. Zafar Hayat
5.8	Re-Testing	5.7	6	Faiq Ali, M. Sajawal, M. Zafar Hayat
6. Deployment				
6.1	Build Bug Free APK	5	1	M. Sajawal
6.2	Install Game on Android	6.1	1	M. Zafar Hayat
7. End User Documentation				
7.1	Training	6	3	M. Sajawal
7.2	Admin Manual Preparation	6	4	M. Zafar Hayat
7.3	User Manual Preparation	6	5	Faiq Ali

1.7.5 Gantt Chart

Figure 1. 2: Gantt Chart

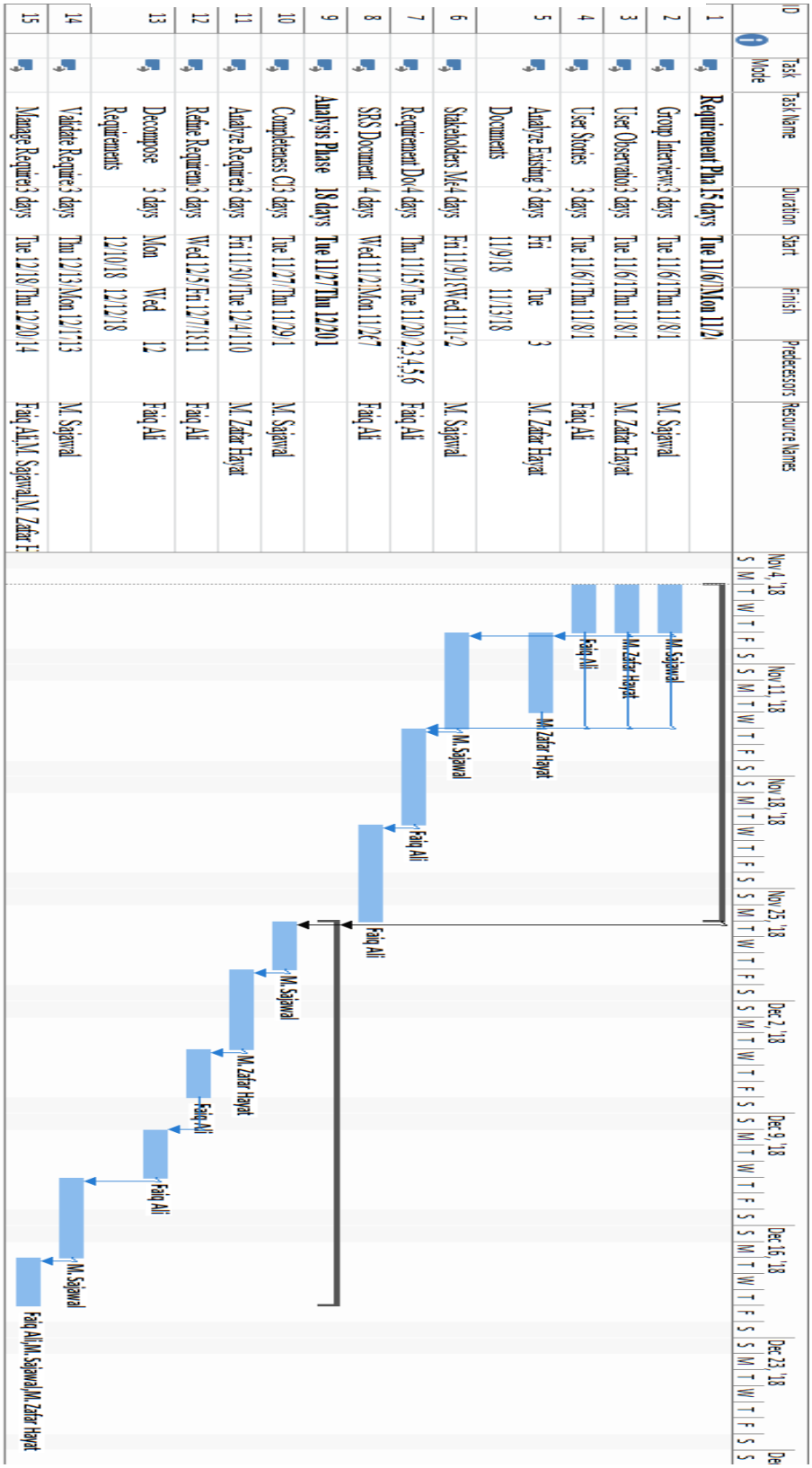


Figure 1. 3: Gantt Chart

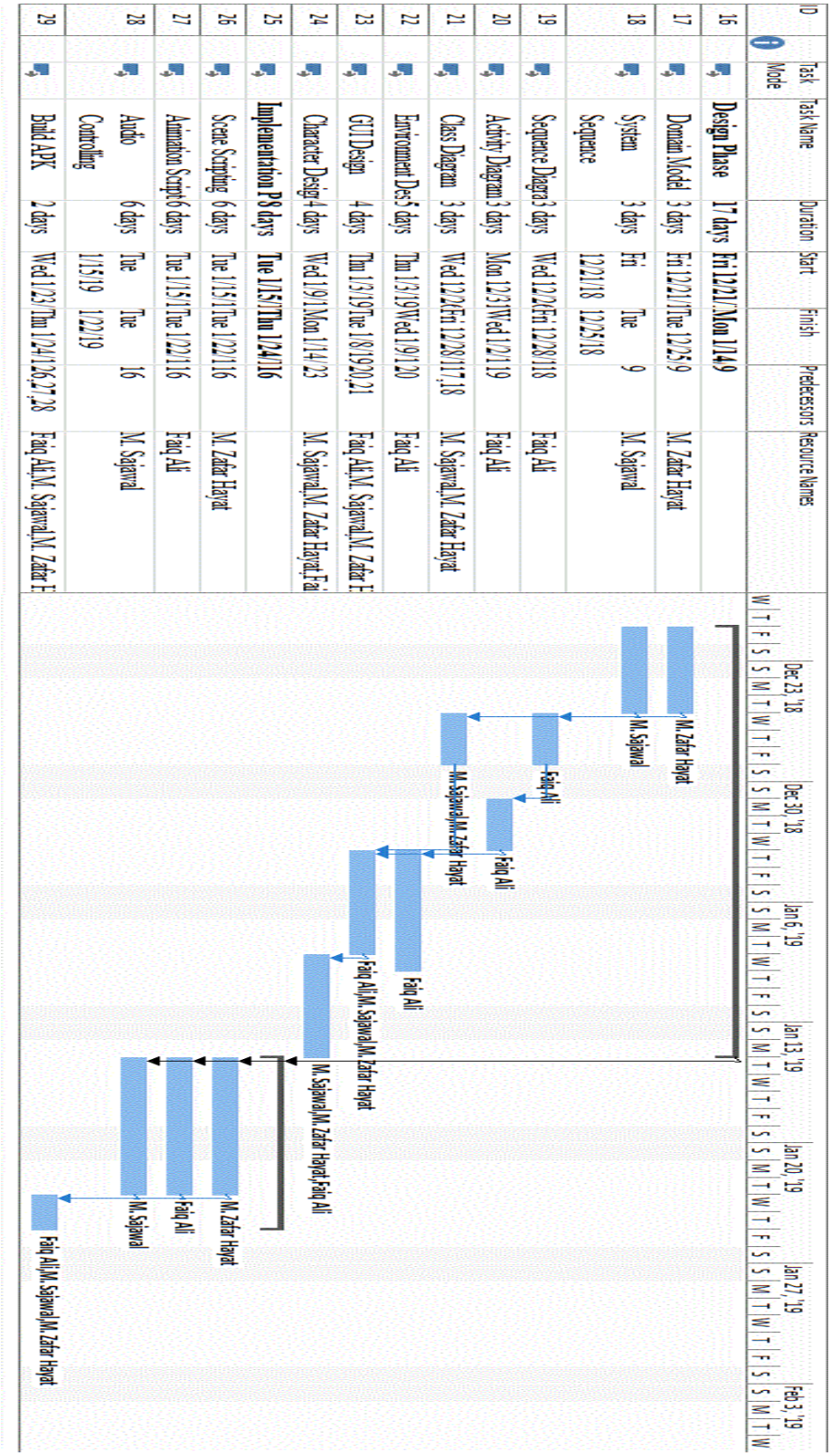
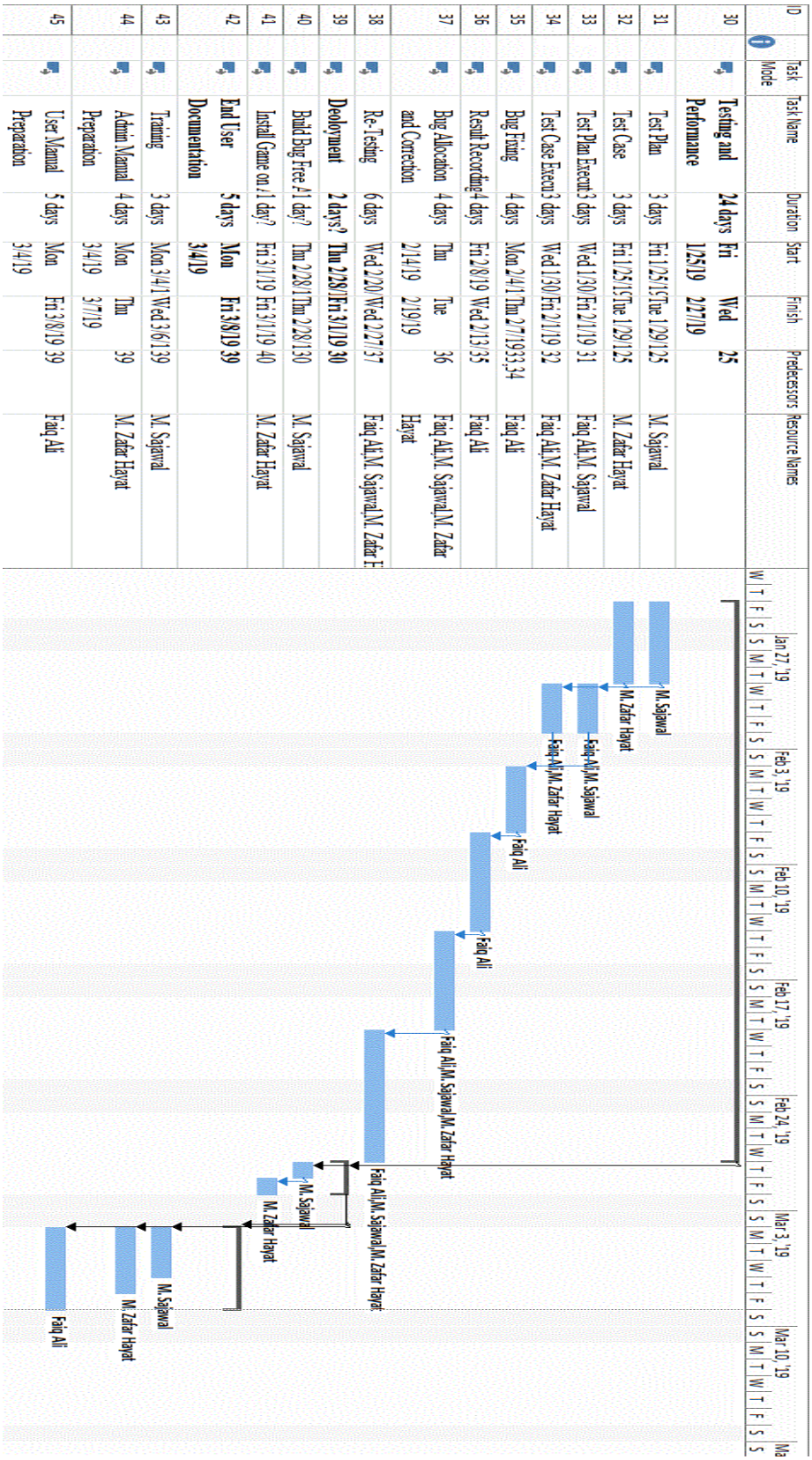


Figure 1. 4: Gantt Chart



1.8 Report Outline

This chapter is all about the:

- Background
- Motivations and Challenges
- Goals and objectives
- Literature Review
- Gap Analysis
- Proposed Solution
- Project Plan

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1 Introduction

2.1.1 Purpose

Games have become an integral part of everyday life for many people. A traditional game often presents a situation where “players engage in an artificial conflict, defined by rules and results in a quantifiable outcome. Such artificial conflicts are often represented as a puzzle or a challenge, and having the puzzle solved or the challenge resolved provides a real-world purpose to the game players. This type of games is sometimes referred to as “serious” games. However, this kind of traditional, or “serious games” has been increasingly replaced by electronic games, especially for the so-called “game generation”. This generation typically consists of “digital natives,” who, in contrast to the “digital immigrants” of the older generation, grew up playing a lot of games and who are trained in skills such as “dealing with large amounts of information quickly even at the early ages, using alternative ways to get information, and finding solutions to their own problems through new communication paths”.

The game Superior under Terror is a Shooting game, and its purpose is to offer player a challenging and enjoyable experience.

The purpose of this document is to set up the requirements for the development of Android 3D First person shooting game. The intended customer is really anyone with an android device. A shooting game is simple and fun to play making it available to anyone.

2.1.2 Document Conventions

HUD – Head up display

3D graphics - Graphic rendering technique featuring three-dimensional objects.

FPS – First person shooting.

GUI – Graphical User Interface.

RLS- reload Systems.

UI – User Interface.

SUE-superior university environment.

2.1.3 Intended Audience and Reading Suggestions

The document is intended for developers, project managers, supervisor, users, testers, IT Experts and documentation writers. Any User easily read this document through Table of content.

2.1.4 Product Scope

Superior under terror is for those who love the shooting that comes with performing serious stages. Player can enjoy the thrill of shooting through spectacular landscape. Control the stages and enjoy the smooth controls with vibrant graphics that will make shooting epic.A fun action packed adventure lies in the game. Upgrading arms to help get through the difficult levels.

Player passes through the difficult stages using the high level arms controls in an effort to get the collectables from the environment. Each collectable increases the arms score of player. Once the target shooting is achieved a level gets completed. An arrow helps the user as a guide as to collect the collectable which get disappears when collected. Game can be played by one player at a time. The main purpose is to give a special shooting experience to those who want to enjoy the thrilling stunts.

Graphic quality of the game is medium as it has multiple benefits to consider:

Devices: High quality graphics games cannot be played on devices which do not support them.

Performance: A high quality graphics slower down the performance of game as the FPS get lower. Game is designed in such a way that increases player interest by increasing the difficulty level. Numbers of levels designed are four in starting. To increase the user curiosity, all the levels are locked except the first one and when player completes the level the next level gets unlocked (allowing user to play next level). Superior under terror business objective is to generate revenue by launching it to Google Play Store.

Constraint: It is to be stated that 3D Modeling is not done by any of the group members as the focus was to learn unity as well as development of game.

2.2 Overall Description

2.2.1 Product Perspective

Superior under terror is thrill based shooting game which is a family-member of the shooting games category on Google Play-Store. The Game is uploaded to Google Play account and the user after making a search on the shooting based game can download the game.

Once the game is downloaded and installed on the android devices, can be played. All the features of the game will be available to the user.

2.2.2 Product Functions

Product functions are as follows:

- Game must have a main menu – GUI interface from where player can select to play or adjust the settings of the game.
- Game should have the shooting selection menu from which the model of the environment should be selected.
- A Game selection mode must be added give user choice regarding the environment to play in.
- A user graphical user interface (GUI) which must let the user to decide which level to play.
- A game-play UI which has all the functionalities of the Arms controls to use, that is essential to play game.
- Pause Panel which must pop-up when pause button is pressed.
- A camera must be added to give the look and feel of the environment using slider.
- Game-play UI should have the shooting control functionalities like Arms, health power, grenade, and movement buttons left and right.
- Game-play must have a navigation which shows the whole plan of the area in kilometers for shooting.

Requirements we managed to meet the major part of all our high priority requirements, such as single player functionality with shooting controls and a fun game play.

2.2.3 User Classes and Characteristics

User of the games will be the lover of the shooting games specially the category of the people who love to make survival action and want free arms with the actual 3D experience. This category is the main target of our business because of the passion and love for shooting and action. More frequent interaction with the game will generate good revenue as well. This category will love the environment and new way of introducing the shooting game with different modes.

Another category of users will be the category with respect to 'age'. This game caters the audience that has an average age of ten to fifteen years. Also the teenagers and users of age 20-28 will love to play the game as well. Their frequency of use may rely on the level of interest and other factors.

One category which is not generally considered is developers and tester who want to play the game for testing purposes as well as to improve the game in terms of features and functionality. New learner of the unity game developer may also take part in playing the games.

2.2.4 Operating Environment

The game is intended to be run on the Android operating system. All the user having the android phone can play the game but with the minimum Android 4.1 Jelly Bean and minimum API level 14 whereas the target API level is 19.

2.2.5 Design and Implementation Constraints

Although there are many engines and tools available for game coding, 3D modeling and 2D designing. One should know the pros and corns of software or application before implementing it so that future constraints could not create hurdles to make any game or impose any conditions.

These software are used to make the game which have their respective major functionalities in the development of the game.

Tools	Outcomes
Unity	Game Environment Scripts
MS Project	Gantt Chart
MS Visio	UML Digrams

MS Word	Documentation

Table 2. 1: Tools

It is to be stated that 3D Modeling is not done by any of the group members as the focus was to learn unity as well as development of game.

2.2.6 User Documentation

User easily familiar to game through tutorials that helps to user improve shooting skills. Also provide guidance with playing step by steps in practice mode.

2.2.7 Assumptions and Dependencies

The final destination of our game's operation will be the Android mobile device. However, Unity will be responsible for both the construction of the game and its integration within the Android framework.

The minimum Android 4.1 Jelly Bean and minimum API level 14 whereas the target API level is 19. The game is not intended to run on PC, MAC, Linux and IOS.

2.3 External Interface Requirements

2.3.1 User Interfaces

Splash screen

The first screen that comes up on the launch of game is ‘Splash screen’ which show the logo of the company/studio.

Main-Menu

After splash screen, main menu helps user to define the actions to be perform before playing it. The player confronts with the main menu screen which has multiple components in it. Component of the main menu are as follows: `

Background – An image showing the screenshot of the environment of the game-play.

Score Board – Here user sees the score achieved.

Setting Option – User can adjust the settings of the play mode and “sound on” and “sound off” functionality.

Rate Game Option – On clicking the “Rate Us” user can rate the game.

Play Option - On pressing the “Play” button user is now directed to select the car on car selection screen.

Setting Panel

By pressing the setting button, main menu disappears and a setting panel appears. By default the game sound remains “on”.

The Setting Panel has the following component in it:

Sound On-Off Option – This function is used to make the sound on or off.

Play Mode

Play mode gives the facility to user to choose the way he wants to play. Basically it focuses on the convenience level of user to play the game.

Arrow Mode – On selection of this option user has the ability to play game in which buttons are provided in the game play for user.

Back Button – Back button disappear the current setting panel and shows the main menu to user.

Arms Selection

“Play” option will make arms selection panel to appear, disappearing the main menu panel. Here user has the ability to rate the game because a rate panel appears on it; user can either rate the game or user can do it later.

User can select the unlocked arms and can buy the arms he wants with the gained scores and can play with that during the game.

Background – background image for the arms selection panel.

Right left buttons – Right and left buttons let the user to see the models of the weapon.

Select Button – User can select the mode to play in the game.

Select Mode

Background – background image with modes.

Back Button – to go back to car selection

Mode Options:

Career Mode Option – this option is for those player who wants to play level based game.

Investigation Mode – This option is a hidden option as a blackhorse.

Survival Mode– This mode is against endless wave of enemies.

Level Selection:

On Level Selection panel user have the ability to select the level of his choice but to unlock to next level, completion of the previous level is required.

Level Selection Options – User can select the level of his choice.

2.3.2 Hardware Interfaces

Hardware Interface for Making Game:

Hardware Interface to run:

"Superior under terror" is a mobile gaming application designed specifically for the Android platform and is functional on both mobile smart phones and tablets. Gaming application data is stored locally on the game engine elements. Now the Android platform is graphically adaptable with a 2 dimensional graphics library and a 3 dimensional graphics library based on OpenGL ES 2.0 specifications as well as hardware orientation, scaling, pixel format conversion and accelerated 3D graphics.

All users having the android phone can play the game but with the minimum Android 4.1 Jelly Bean and minimum API level 14.

2.4 System Features

2.4.1 Main Menu

The player confronts with the main menu screen. This is the top most priority screen in which user needs to interact with to play game. Here user sees the score achieved. User sees the "Rate Us" button and "Play" buttons. User can adjust the settings of the play mode and "sound on" and "sound off" functionality.

Stimulus/Response Sequences

Step 1: if the player presses the play button on the main-menu interface.

Step 1.1: The main menu disappears and another panel for Arms selection appears which contains the Arms selection options.

Step 2: if the player presses the “Rate Us” button.

Step 2.1: User is directed to the online “rate us” platform.

Step 3: if the player presses the “Setting” button.

Step 3.1: User is prompted with a panel where user can adjust the settings of the play mode and “sound on” and “sound off” functionality.

Functional Requirements

REQ-1: Main menu screen should contain background image showing the menu background.

REQ-2: Main menu panel should contain the score view of user which user has achieved. In case, user is playing game at very first time, zero score should show.

REQ-3: Main menu panel should contain the score view of user which user has achieved even after exiting the game.

REQ-4: Main menu panel should contain the buttons: “Play”, “Setting” and “Rate Us”.

REQ-5: if the player presses the “Play” button the main menu should disappear and another panel for car selection should appear which should contain the arms selection options.

REQ-6: if the player presses the “Setting” button the main menu should disappear and another panel for the setting of game should come on the screen where user should be able to on and off the sound of the game along with the ability to adjust the mode of the game.

2.4.2 Setting Panel

Description and Priority

On selection of the setting by pressing the setting button, main menu disappears and a setting panel appears. By default the sound remains “on” if the user needs to off the sound it presses the sound off option then sounds get off. This screen also enable user to choose between the play modes which are, in our case is, “Arrow Mode” and “Tilt Mode”. By default mode is arrow mode.

Stimulus/Response Sequences

Step 1: The player may press the sound off button.

Step 1.1: Game sound gets off. On click again the sounds get on.

Step 2: Player may want to change the mode of the game. Player selects the “Tilt Mode” or “Arrow Mode” by pressing the corresponding buttons.

Step 2.1: Game has the ability to change the mode to the selected one.

Functional Requirements

REQ 1: On click setting button on main menu, main menu should disappear and setting panel should appear.

REQ 2: Setting panel should have sound on and off buttons.

REQ 3: Setting panel should also have the “Tilt Mode” and Arrow Mode buttons

REQ 4: Setting Panel should have a background (a background image).

REQ 5: By default sound should be on.

REQ 6: On press sound off button sound should get off and on click sound on the sound should get on.

REQ 7: By default player mode should be “Arrow Mode”.

REQ 8: On selection of the corresponding buttons user should be able to play game in that mode.

2.4.3 Arms Selection

Description and Priority

After pressing the “Play” button Arms selection panels get appear disappearing the main menu panel. Here user has the ability to rate the game because a rate panel appears on it; user can either rate the game or user can do it later. If the user want to rate the game, it is directed to online rating channel. On the other hand if user selects to do it on some later stage by pressing the “No Thanks” button, “Rate Us” panel disappears and now user can select the Arms it want.

User can select the unlocked Arms and can buy the Arms he wants with the gained scores and can play with that during the game.

User score/cash get deducted after purchasing arms. If user does not have enough cash to purchase a arms then he is unable to play with that arms.

Stimulus/Response Sequences

Step 1: The player presses the “Play” button.

Step 2: Player is prompted with the arms selection panel on which Rate Us panel asks user to rate now or later.

Step 3: If the user wants to rate the game,

Step 3.1: it is directed to online rating channel.

Step 3.2: If user wants do not want to rate the game for now.

Step4: “Rate Us” panel disappears.

Step 5: User selects the arms by using right left arrow buttons.

Step 6: Arms selection done if the purchase price is less than or equal to the score gained by user.

Functional Requirements

REQ 1: After pressing the “Play” button, player should see the Arms selection panel dis-appearing the main menu panel.

REQ 2: Player should be prompted with the arms selection panel on which Rate Us panel should appear asking the user to rate now or later.

REQ 3: User should not see this panel again if user has rated once.

REQ 4: User should be directed to the online rating platform.

REQ 5: On denial of rating the game user “Rate Us” panel should get disappear.

REQ 6: player should see the Arms models on right and left button clicks.

REQ 7: There should be a limited numbers of Arms to select from.

REQ 8: Every arms should have purchase price except one which is players default arms.

REQ 9: There should have the ability to select arms or buy it.

REQ 10: if the user presses the buy button arms selection should be done if the purchase price is less than or equal to the score gained by user otherwise it should not be able to select.

REQ 11: If a Arms model selected / purchased by player, it should be able to play in the game level with that arms model.

REQ 12: Player should be able to see his score on the selection panel.

2.4.4 Select Modes

Description and Priority

After selecting the Arms to play with, player is asked to choose between the modes to play with. Player has three options to select: “Career Mode”, “Free Mode” and “Emergency Mode”.

Here user has the option to go back to main menu by clicking the “Back” button.

Also Player has the ability to select the given mode. By pressing the Career Mode user is shown with the “Level Selection” panel in which user can select the unlocked level of his choice.

On the other hand, if player selects the “Free Mode” then “Select Mode” panel will appear with two modes: “Day Mode” and a “Night Mode” to choose from. If user selects the day mode game plays in the light mode. On the other hand if the player chooses “Night Mode” user plays the game in the night scene.

Stimulus/Response Sequences

Step 1: The player selects or buy arms on the arms selection panel.

Step 2: Player is prompted with the “Select Mode” panel on which two modes (Career and Free Modes) are available.

Step 3: If the user wants to play in a Career Mode it presses the corresponding button.

Step 3.1: Mode user is shown with the “Level Selection” panel in which user can select the unlocked level of his choice.

Step 4: If user wants play in a free mode, user chooses “Free Mode” button

Step4.1: User is given options to select from “Day Mode” and “Night Mode”.

Step 4.2: User selects the Day Mode.

Step 4.2.1: Game plays in day light scenario.

Step 4.3: User selects the Night Mode.

Step 4.3.1: Game plays in the night scene.

2.4.5 Level Selection

Description and Priority

The level selection screen is the primary way for the player to choose between different levels. The game is separated into multiple levels. On Level Selection panel user have the ability to select the level of his choice but this choice is restricted to unlock to advance level by playing more and more game and completing the previous level. By default first level is unlocked to let the player to play game.

Stimulus/Response Sequences

Step 1: Player selects the Career Mode.

Step 2: Game appears with the level selection panel which have multiple levels to show.

Step 3: Player chooses the unlocked level.

Step 4: Player is presented with the game play

Functional Requirements

REQ-1: On pressing Career Mode button user should be shown a Level Selection panel.

REQ-2: User should see multiple levels on Level Selection panel in which only first level should be unlocked if the player is playing game for first time and other levels should be locked.

REQ-3: If a player selects a level, game must start that level.

REQ-4: If user has completed a level, next level should be unlocked on Level Selection panel.

2.4.6 Game Play

Description and Priority

The Game-Play screen is the screen the player will see every time upon level start of the game. Through this interface, the player can use the car controls by using the game play user interface, can see the updated score. Player can, pause game anytime it wants from here.

Stimulus/Response Sequences

Step 1: The player launches the game from their portable device.

Step 2: The start screen loads and appears, prompting the player with buttons: “Play

Game”, “Rate Us”, “Setting”.

Step 3: The player presses one of the buttons, triggering its respective function.

Step 4: Player select the Arms from Arms Selection screen and select the mode from Select Mode to play in.

Step 5: User select the level to play in Career Mode and Day or Night Mode in Free Mode.

Step 6: On selection of the level or mode user plays the game in the game play corresponding to the desired action.

Functional Requirements

REQ-1: The game play screen must load and appear every time the game level selected or game mode selected.

REQ-2: If the player presses the pause button, game pause pop-up will appear.

REQ-3: If the player completes the game, level complete pop-up will appear describing the achieved score of player.

2.4.7 Camera Rotation

Description and Priority

360 degree rotating camera is added in game environment placed on the game UI as a slider. This camera helps user to see the complete environment using the slider.

Stimulus/Response Sequences

Step 1: The player notices the camera and checks the target area of camera.

Step 2: Player moves the button on the slider.

Step 3: Camera rotates and shows the other objects in the environment.

Functional Requirements

REQ 1: Camera must be added to the game play user-interface.

REQ 2: There should be a slider that when moved should show the complete environment.

REQ 3: Camera must consistently update its current position.

2.4.8 Pause Menu

Description and Priority

The player should be able to pause anytime during game-play, and this screen fulfills that requirement. The pause menu also allows the player to navigate between game-play and the level selection and title screens. The portable nature of the console renders player convenience paramount, so this feature must be included.

Stimulus/Response Sequences

Step 1: The player presses the pause button on the game-play interface.

Step 2: The level pauses, drawing up the pause menu which prompts the player with three options: “Resume,” “Restart” and “Home”

Step 3: The player presses one of the buttons, triggering its respective function.

Functional Requirements

REQ-1: The “Restart” option must restart the level from its beginning.

REQ-2: The “Resume Game” option must continue the game without any change to the character’s vector or the state of the level from the moment of the pause action.

REQ-3: The “Home” option must start the game from main menu.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

- Game Loading Time -Performance of game must be good in term of loading time of the game. Loading time of the game should not be more than 5 seconds.
- Game Lag- Game should not create much lagging while playing it: The game should not have less than 30 FPS on all devices.
- Mobile processors and graphics can easily render the game frames without buffering or any other kind of frames dropping.

2.5.2 Safety Requirements

Use of Sounds

Agley sound should be avoided so that user health must be taken care of.

Sound quality should be high or medium.

Use of Lights

Light intensity should not exceed normal light quality so as to protect the eyes of the user.

Use of Textures and Graphics

Textures have psychological impact; the game must take stages of the combination of the colors used in images and the textures for objects. Agley textures must be avoided to take care of the psychological effects.

Environment of the Game

The game must be in such an environment which should create a peaceful yet enjoyable impact on the player's health.

2.5.3 Security Requirements

Data Confidentiality

The game will not ask for access of any personal data.

2.5.4 Software Quality Attributes

Adaptability

- Game should be developed in such a way that if it needs to be launched on other platform other than Android, it should not require major changes.

Availability

- When launching a game on android devices as prescribed in SRS, it should get launch not more than 5 seconds.
- Game should not be crashed in any condition of fault, bug or in case of errors.

Maintainability

- To make a new update in future, to improve the features and functionalities, the design and development of the game should be capable enough that it could be changeable, stable and testable.

Reliability

- Session handling should be done in such a way that the score of the user should maintain after even the game has exited.
- Shooting controls should not behave in the manner that produces the desired results i.e. reload of arms should reload every time the player uses the function.
- The flow of the game should remain the same as designed.

Reusability

- For updating game purpose, for making another game with the advanced features and to remove the un-intended bugs, the game developing team should make a development in such a way that it can be used in the future purposes.

Usability

- Game design should be simple in a manner that it should be user friendly and easy to learn.
- Game GUI should have meaningful symbols to guide users about the functionality of it.
- Graphic quality should be good enough to be rendering on all mobile devices with good speed.

Chapter 3

Use Case Analysis

Chapter 3: System Analysis

Use case analysis is a technique used to identify the requirements of a system (normally associated with software/process design) and the information used to both define processes used and classes (which are a collection of actors and processes) which will be used both in the use case diagram and the overall use case.

3.1 Use Case Model

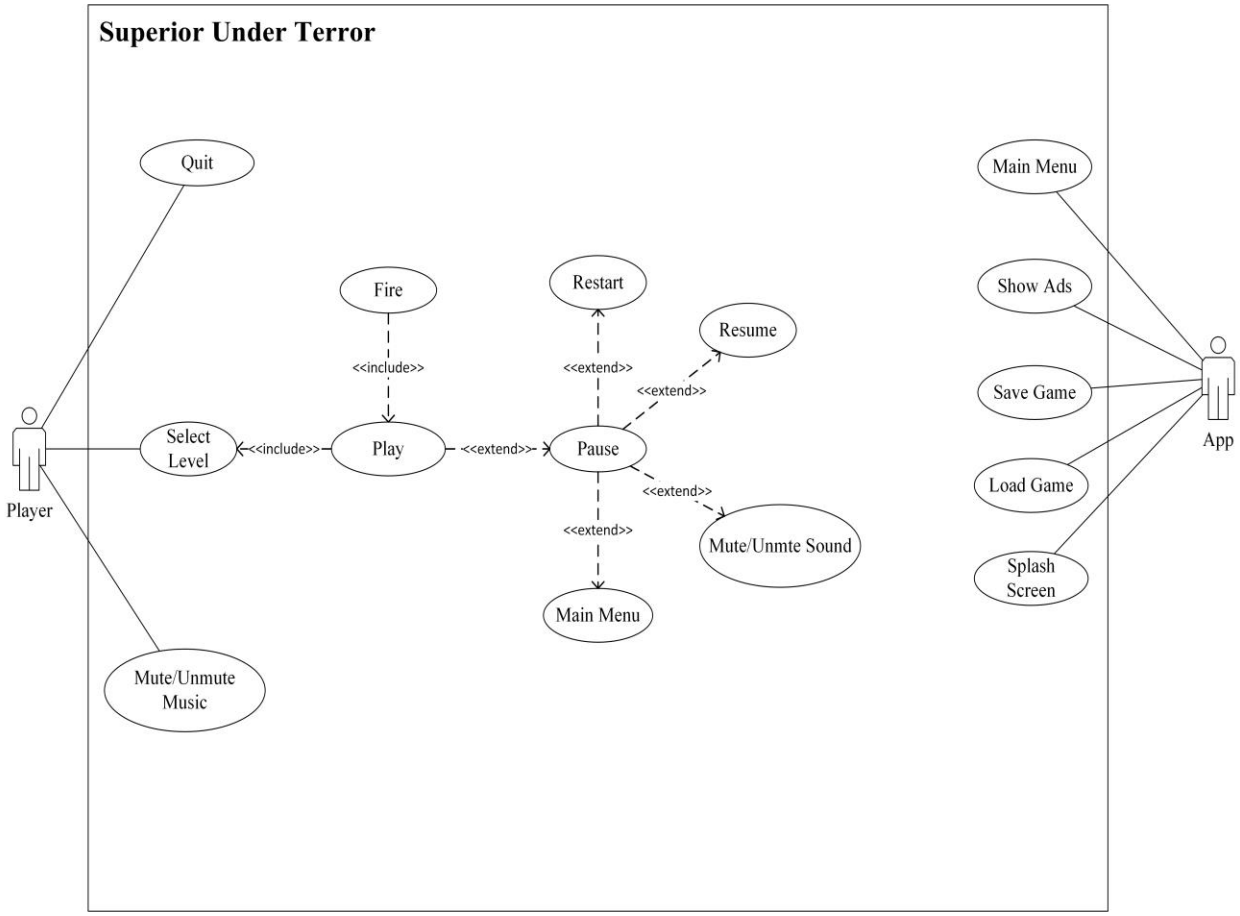


Figure 3. 1: Use Case Diagram

3.2 Fully Dressed Use Cases

3.2.1 Quit

Pre-Conditions

1. Game is must on.
2. User are on main menu.

Post Conditions

Game will be quitted successfully.

Main Success scenario

1. Open main menu.
2. Press quit option.

Alternatives

Google ads may show at that time

3.2.2 Select level

Pre-Conditions

1. Game is must on.
2. User must select the career mode option.

Post Conditions

Level of career mode shown to gamer.

Main Success scenario

1. Open the game.
2. Press select mode option from main menu.
3. Press career Mode option.
4. Select the level of game.

Alternatives

1. Mode menu may not appear and google ads open.
2. Level of game menu not shown.
3. Some other level load.

3.2.3 Play game

Pre-Conditions

1. Game is must on.
2. User must select the game mode.

Post Conditions

Game played with selected environment.

Main Success scenario

1. Open the game.

2. Press select mode option from main menu.
3. Select mode and then press play game.

Alternatives

1. Google ads open.
2. Game not start due to some error.

3.2.4 Fire

Pre-Conditions

1. Game is on.
2. User must press the fire button.

Post Conditions

Bullet fired from gun.

Main Success scenario

1. Press fire button in game.
2. Bullet fired from gun.

Alternatives

1. Bullet fired but sound of fire not played.
2. Fire animation not played.

3.2.5 Pause

Pre-Conditions

1. Gamer is playing game.
2. User must select the pause button.

Post Conditions

Pause menu shown to gamer.

Main Success scenario

1. Press the pause button.
2. Game pause menu appears.

Alternatives

Google ads open.

3.2.6 Return to main menu

Pre-Conditions

1. Game is on pause.

Post Conditions

Main menu open on screen.

Main Success scenario

1. Press pause game from game.
2. Press return to main menu.

Alternatives

Main menu may not appear and google ads open.

3.2.7 Restart

Pre-Conditions

Game is on pause.

Post Conditions

Game restarted again.

Main Success scenario

1. Press pause game from game.
2. Press restart button.

Alternatives

1. Pause menu may not appear.
2. Google ads open.

3.2.8 Resume

Pre-Conditions

1. Game is on pause

Post Conditions

Game started where the gamer left.

Main Success scenario

1. Press pause game from game.
2. Press resume button.
3. Move to game again.

Alternatives

1. Google ads open.

2. Game started from start.

Chapter 4

System Design

Chapter 4: System Design

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

4.1 Architecture Diagram

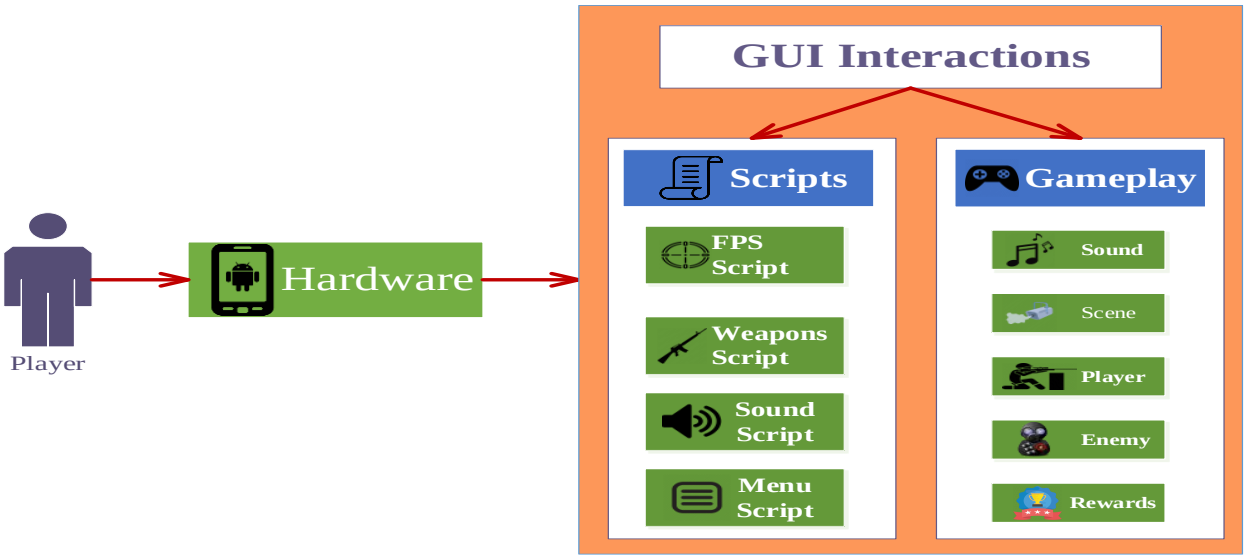


Figure4. 1: System Architecture

4.2 Domain Model

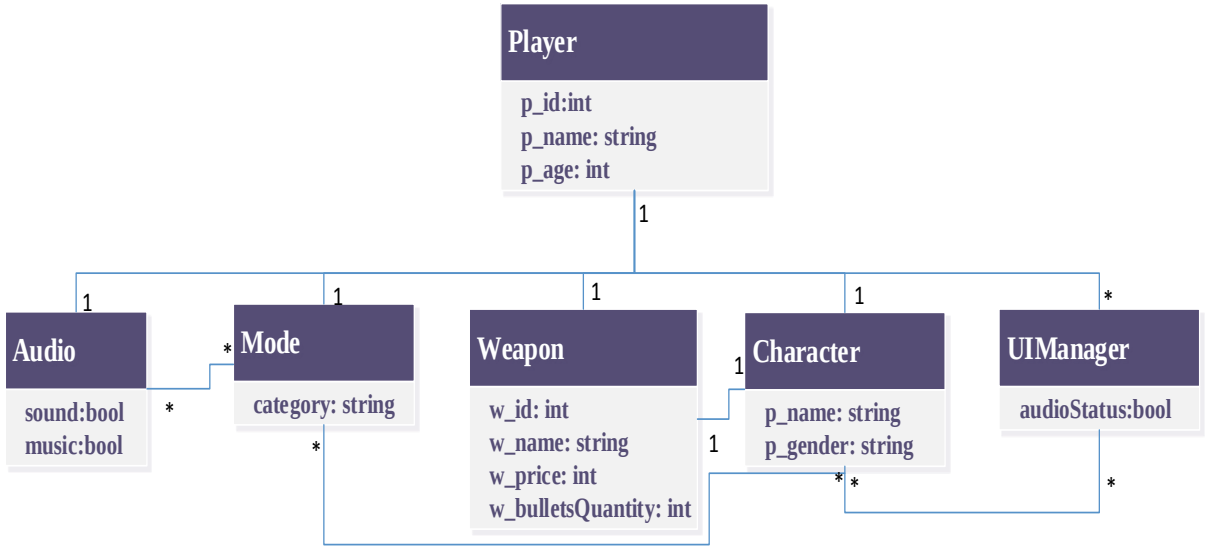


Figure4. 2: Domain Model

4.3 Class Diagram

Figure4. 3: Class Diagram

4.4 Sequence / Collaboration Diagram

4.4.1 Play Game

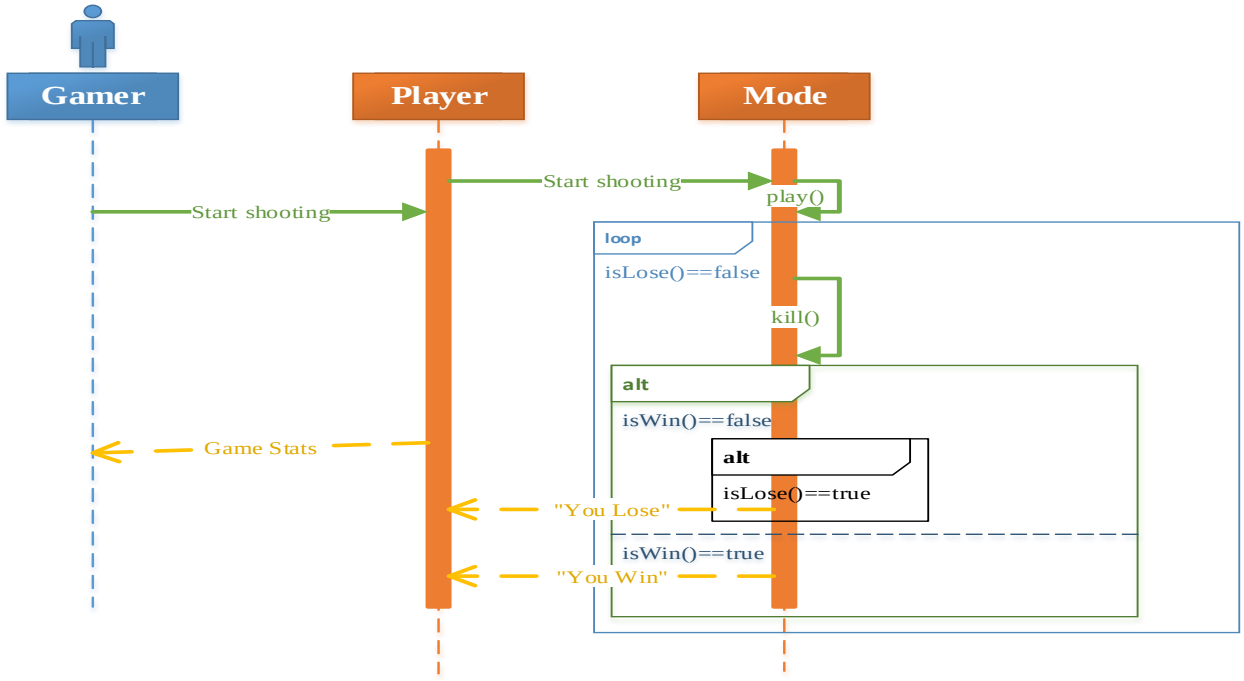


Figure4. 4: Play Game SD

4.4.2 Shoot Bullet

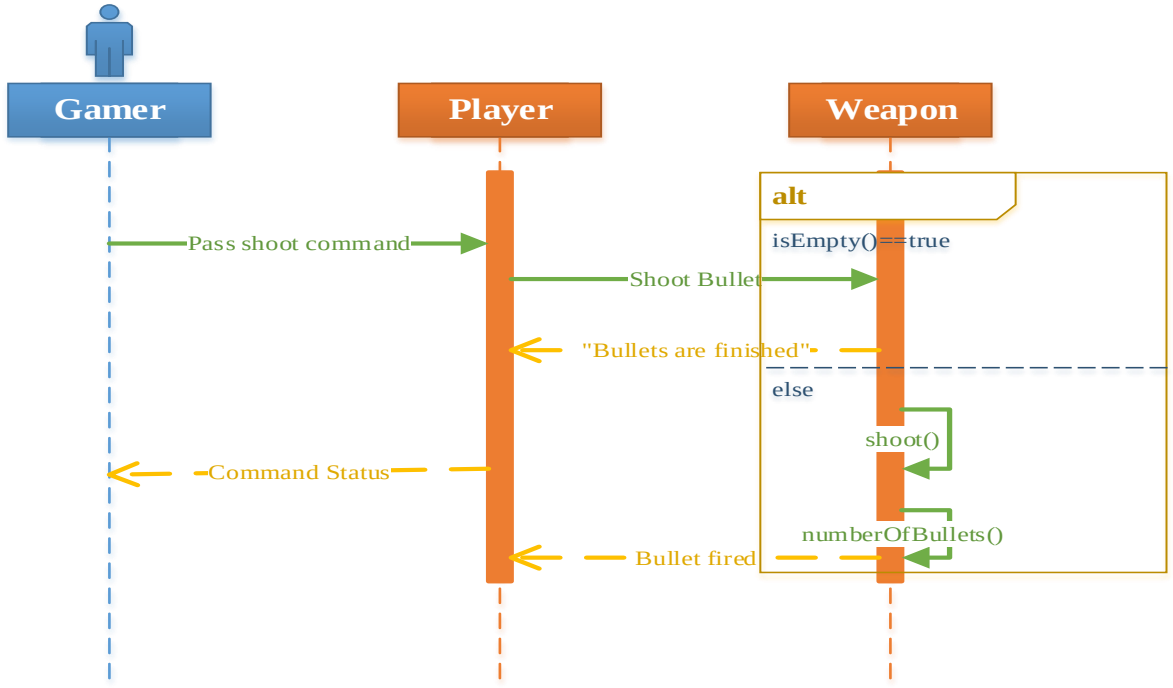


Figure4. 5: Shoot Bullet SD

4.4.3 Start a new game

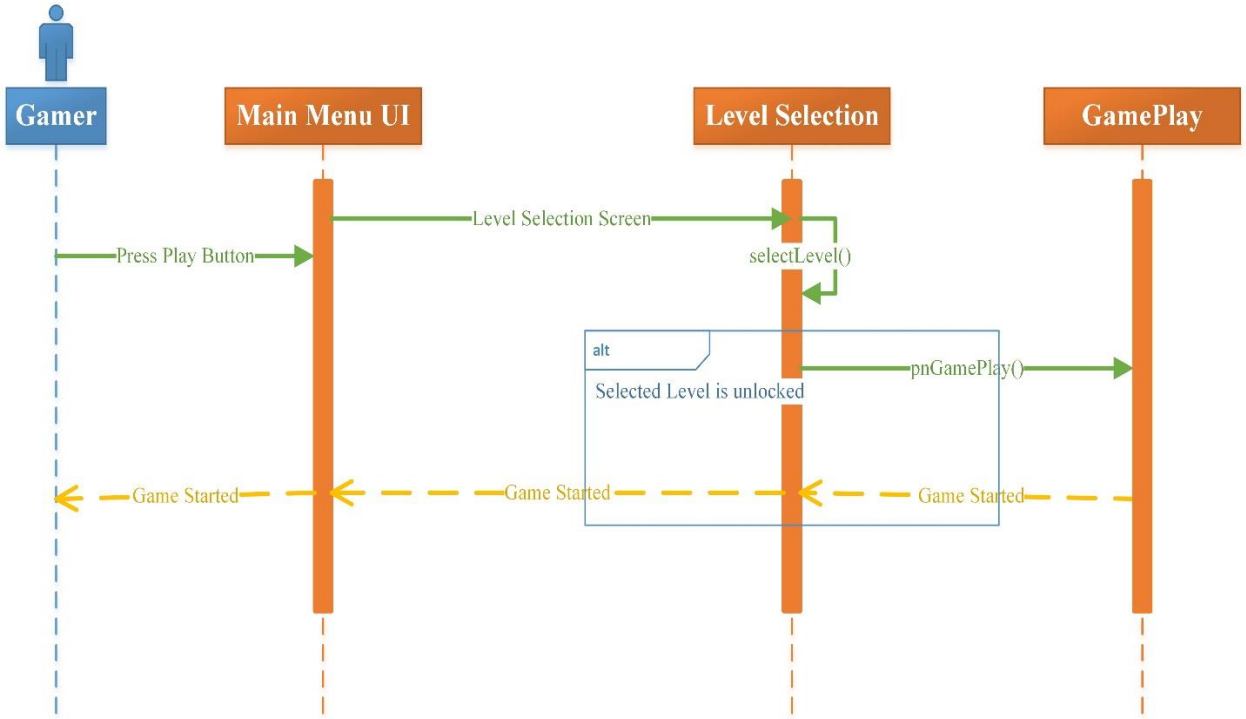


Figure4. 6: Start a new Game

4.5 Operation contracts

Table 4. 1: Operation Contracts

Operation	Kill()
Cross reference	Play
Pre-Condition	The game must be in running state
Post Condition	The health power will be low Player will be able to select the option for get high health
Operation	isLose()
Cross reference	Play
Pre-Condition	Game must be in running state
Post Condition	Health power will be reduce Message will be displayed You Lose The shooter will be become slow due to low power In case of any other selection, the game will be shift to main menu
Operation	IsWin()

Cross reference	Play
Pre-Condition	Game must be in running state
Post Condition	Score will be displayed The message will be displayed You win The game will shift the next level In case of any other selection, the game will be shift to main menu
Operation	isEmpty()
Cross reference	Play
Pre-Condition	Game must be in running state Shoot command must be passed
Post Condition	If bullets are empty the enemy will not be shot
Operation	Shoot()
Cross reference	Play
Pre-Condition	Game must be in running state Shoot command must be passed
Post Condition	The player will be shoot Enemy will be die Enemy health power will be reduce
Operation	NoOfBullets()
Cross reference	Play
Pre-Condition	Game must be in running state The character must have weapon
Post Condition	If the bullets become less the alert sound will be generated. The noOfBullets will be displayed continuously
Operation	selectLevel()
Cross reference	Select Level
Pre-Condition	Level option must be displayed User must be able to select the level
Post Condition	The level menu will be displayed The player will be able to select the level before to start the game.

4.6 Activity Diagram

4.6.1 Quit Game

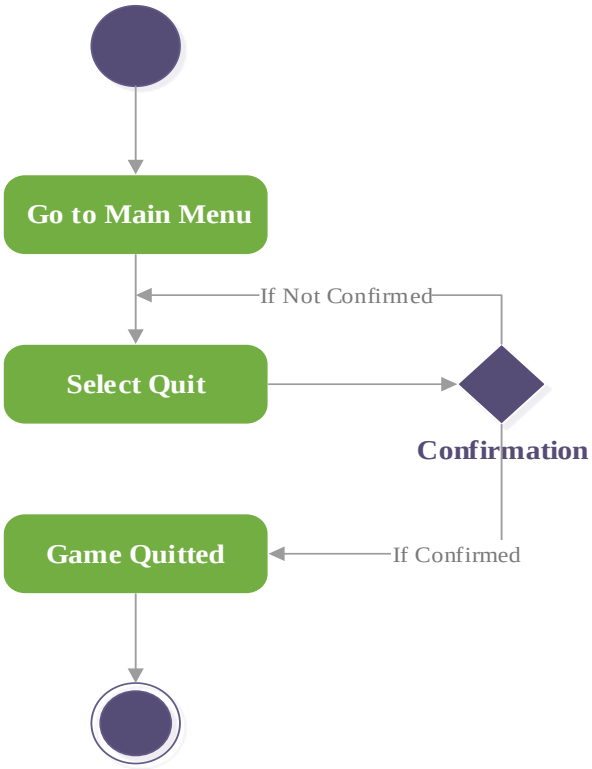


Figure4. 7: Quit Game Activity

4.6.2 Leave Running Game

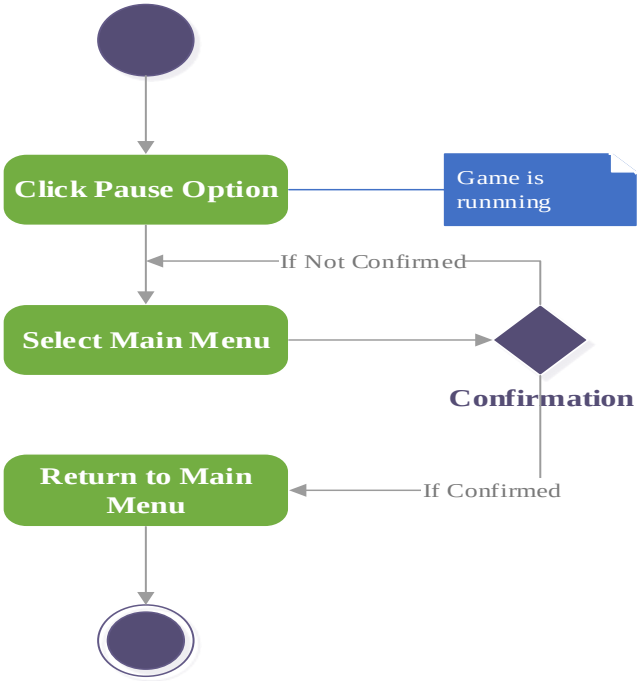


Figure4. 8: Leave Running Game Activity

4.6.3 Start Game

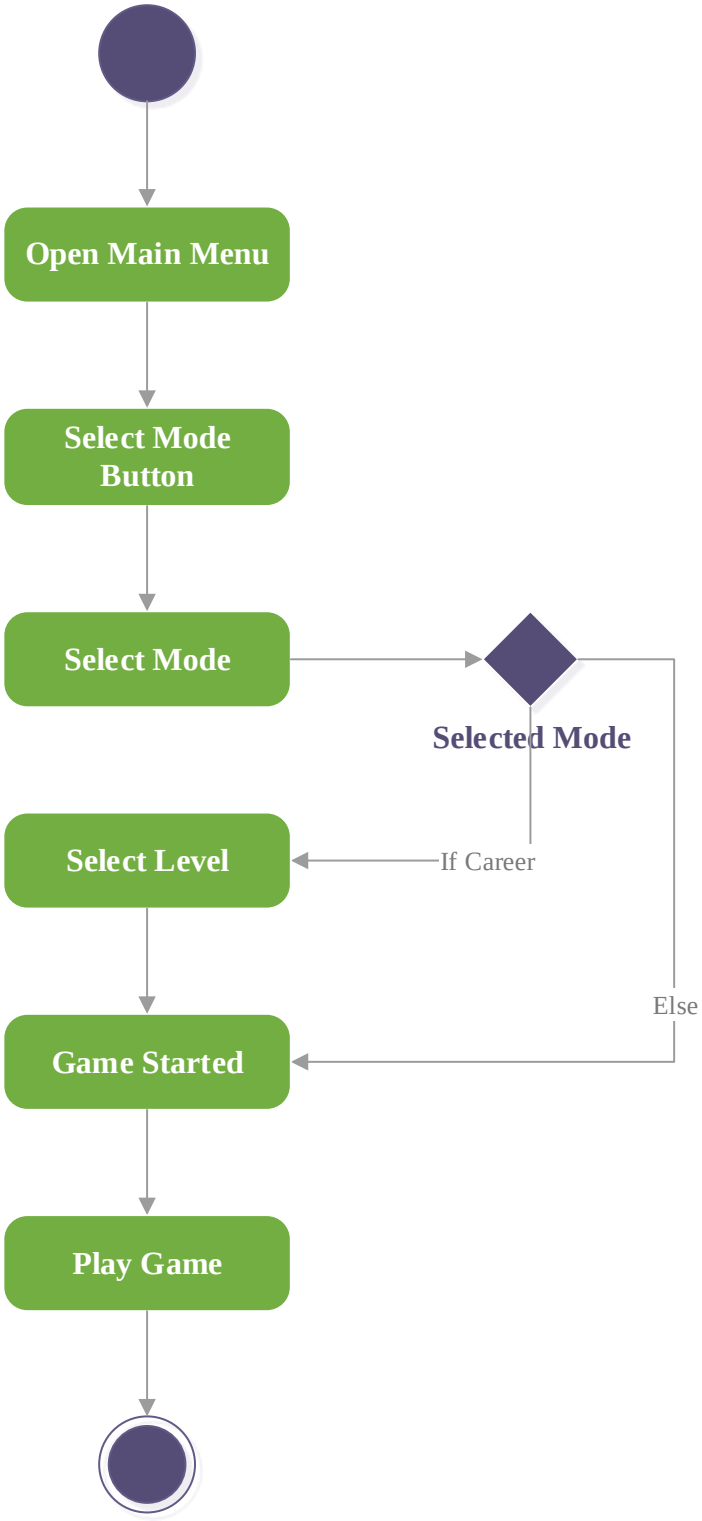


Figure4. 9: Start Game Activity

4.6.4 Restart Game

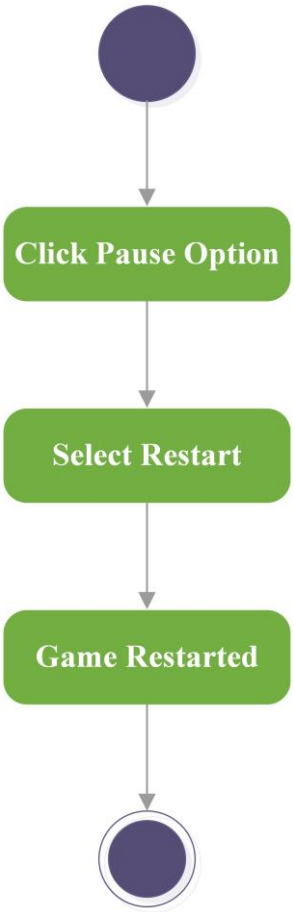


Figure4. 10: Restart Game Activity

4.7 State Diagram

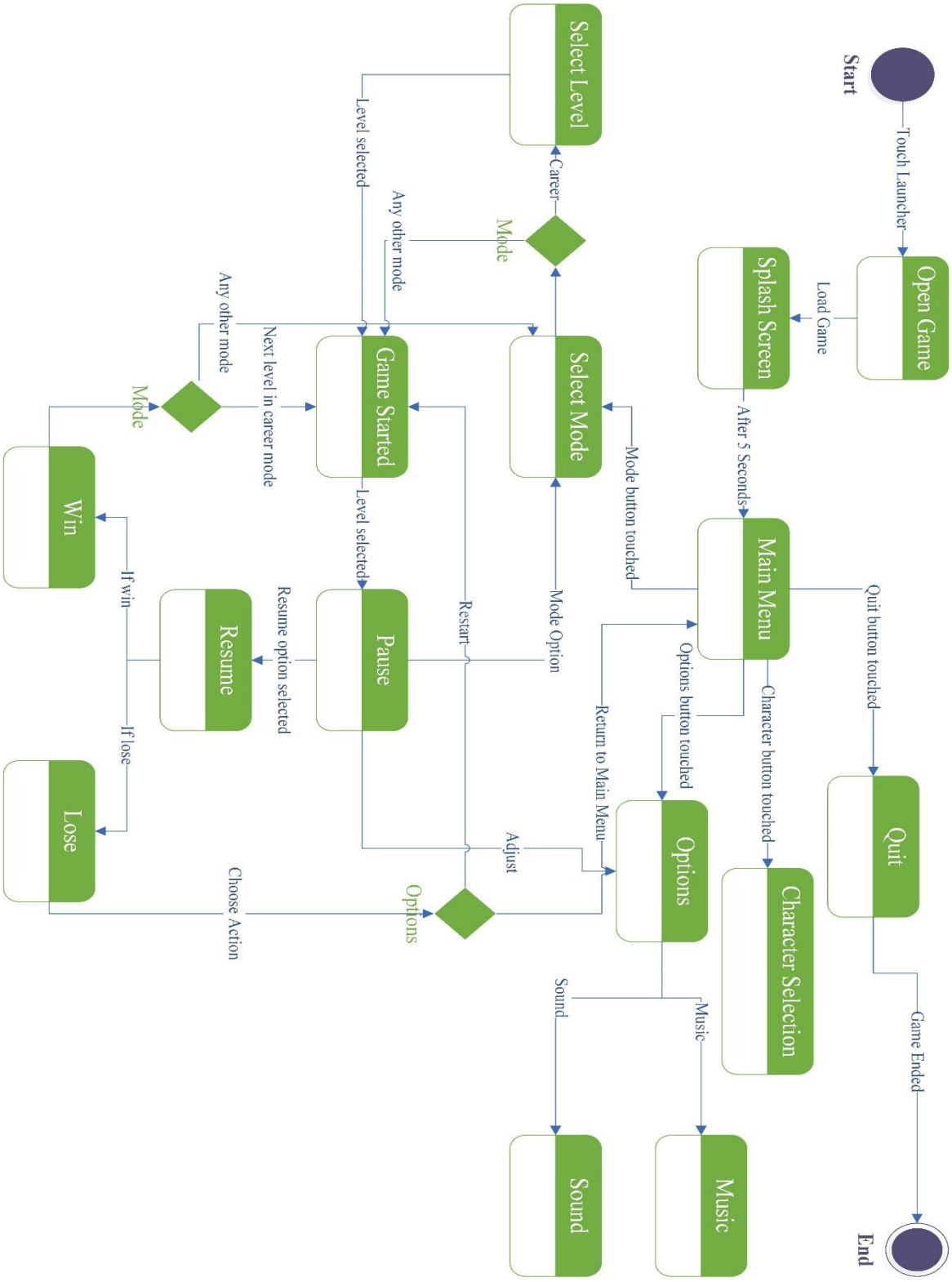


Figure4. 11: State Diagram

4.8 Component Diagram

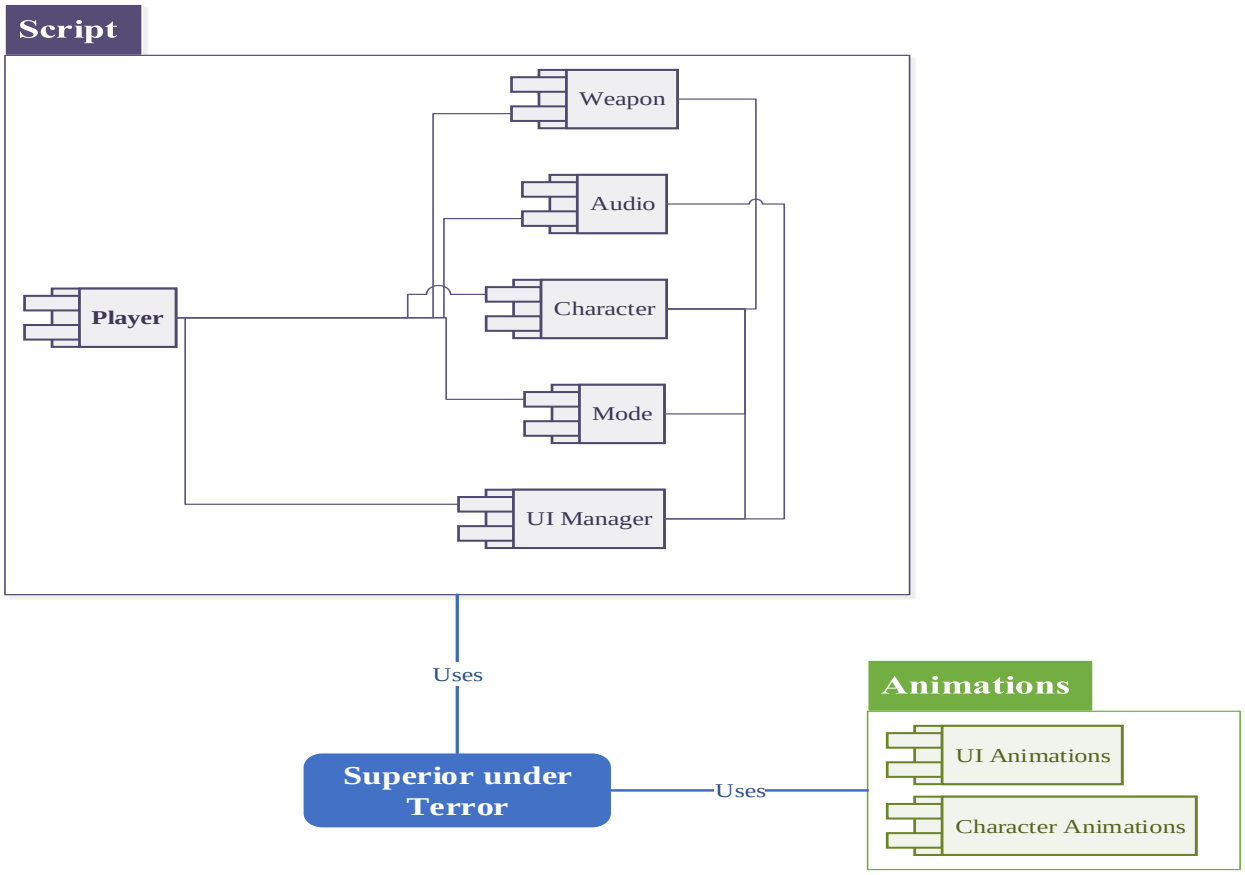


Figure4. 12: Component Diagram

4.9 Deployment Diagram

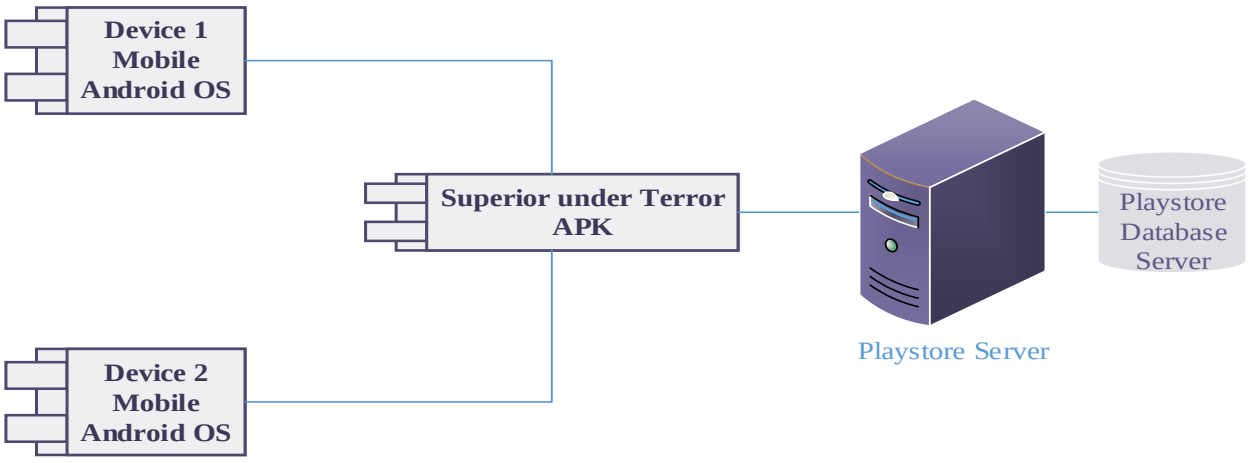


Figure4. 13: Deployment Diagram

4.10 Data Flow diagram

4.10.1 Level 0

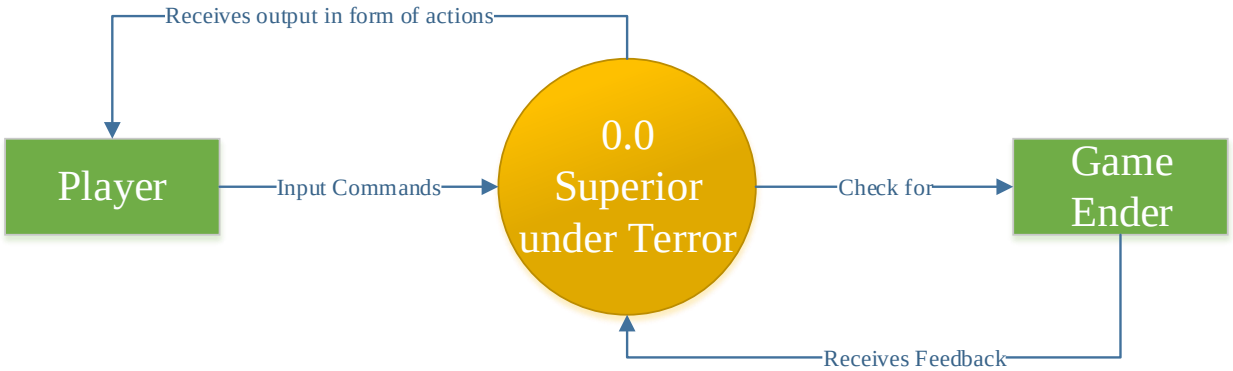


Figure4. 14: DFD Level 0

4.10.2 Level 1

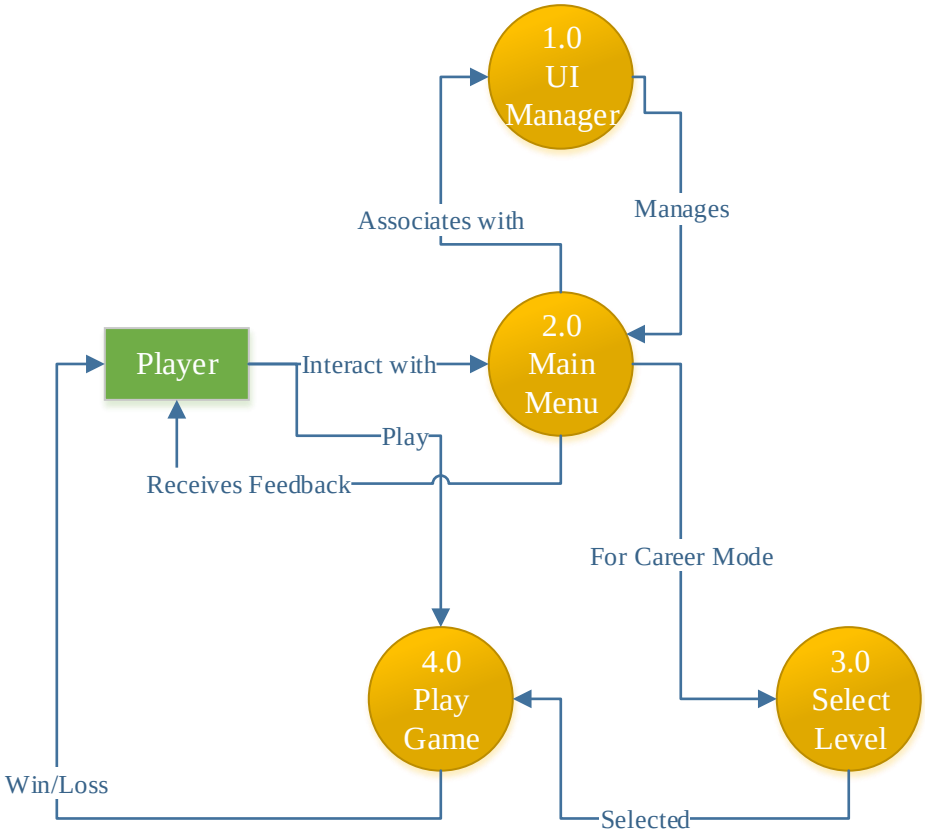


Figure4. 15: DFD Level 1

Chapter 5

Implementation

Chapter 5: Implementation

5.1 Important Flow Control

5.1.1 Start Game

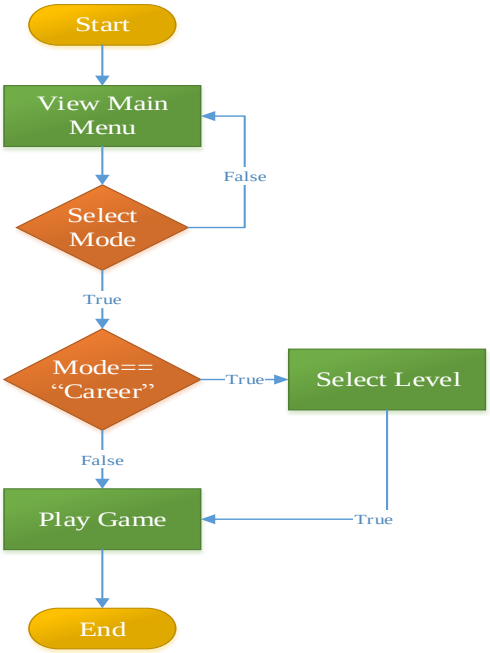


Figure 5. 1: Select Mode

5.1.2 Play Game

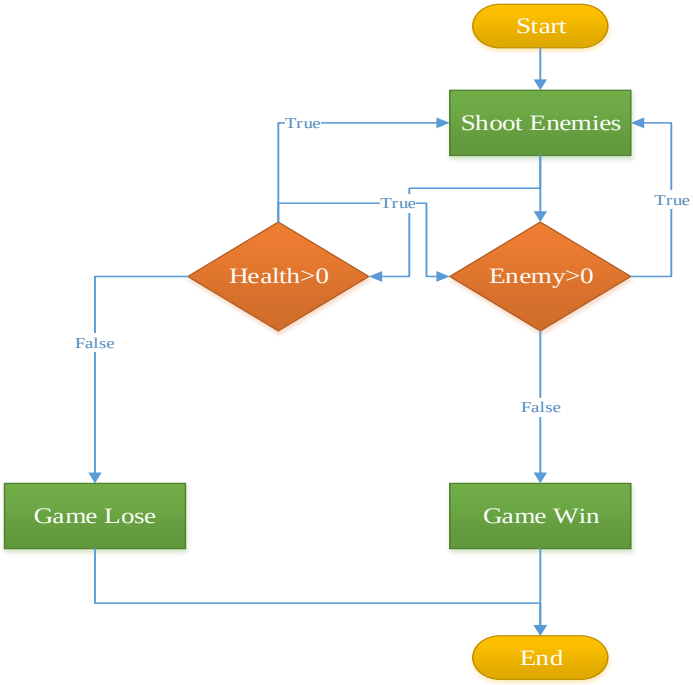


Figure 5. 2: Play Game

5.2 Components, Libraries, Web Services and stubs

The unity components are as follows:

- Characters
- Animations
- Game Objects
- Terrain
- Pre-Feb
- 3D Objects

5.3 Deployment Environment

The deployment environment is (Android, n.d.). I will briefly talk about the Android platform. Android is an operating system based on the Linux kernel and designed predominantly for touchscreen mobile devices such as smartphones and tablets. It was first developed by Android Inc., which Google backed financially and later bought in 2005. Android was officially revealed in 2007 along with the founding of the Open Handset Alliance - a group of hardware, software, and telecommunication companies keen to progress open standards for mobile devices (Android, n.d.). Android provides a rich application framework that allows anyone to build innovative apps and games for mobile devices. Android development is the process of creating applications for the Android operating system. They are usually developed in the JAVA programming language using the Android SDK, but there are many other development tools available to use. As of July 2013, more than 1 million applications have been developed for Android (Warren, 2013). Android devices can be interacted with using touch inputs, such as swiping, tapping and pinching. There is also internal hardware such as accelerometers, gyroscopes and proximity sensors that also allow a unique form of interaction.

The Superior under Terror is using Superior University environment and its asset will look like this and game.

5.3.1 Main Building

Figure 5. 3: Main Building (Front View)

Figure 5. 4: Main Building (Right View)

Figure 5. 5: Main Building (Left View)

5.3.2 University Stage

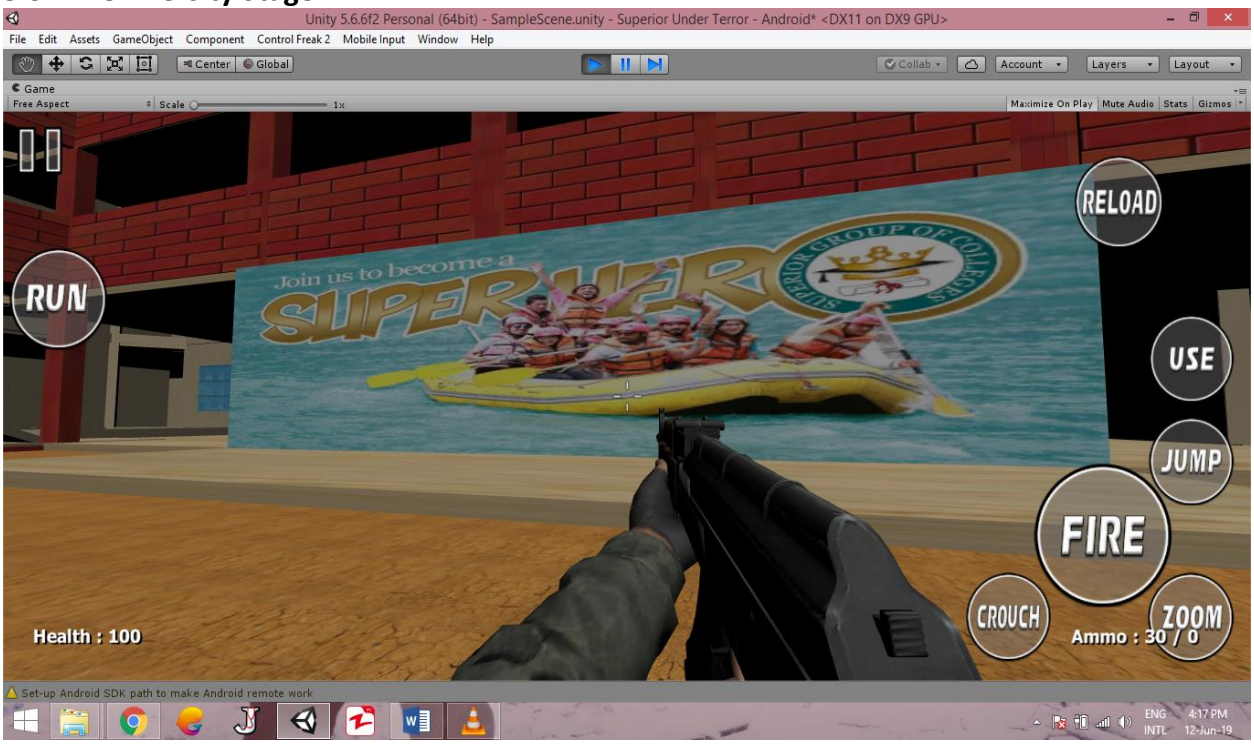


Figure 5. 6: University Stage

5.3.3 University Classroom

Figure 5. 7: Superior University Environment (Classroom)

5.3.4 University Auditorium (Front View)

Figure 5. 8: Superior University Environment (Auditorium Front View)

5.3.5 University Auditorium (Back View)

Figure 5. 9: Superior University Environment (Auditorium Back View)

5.3.6 University Faculty (Entrance)

Figure 5. 10: Faculty Room Entrance

5.3.7 University Faculty (Toilet)

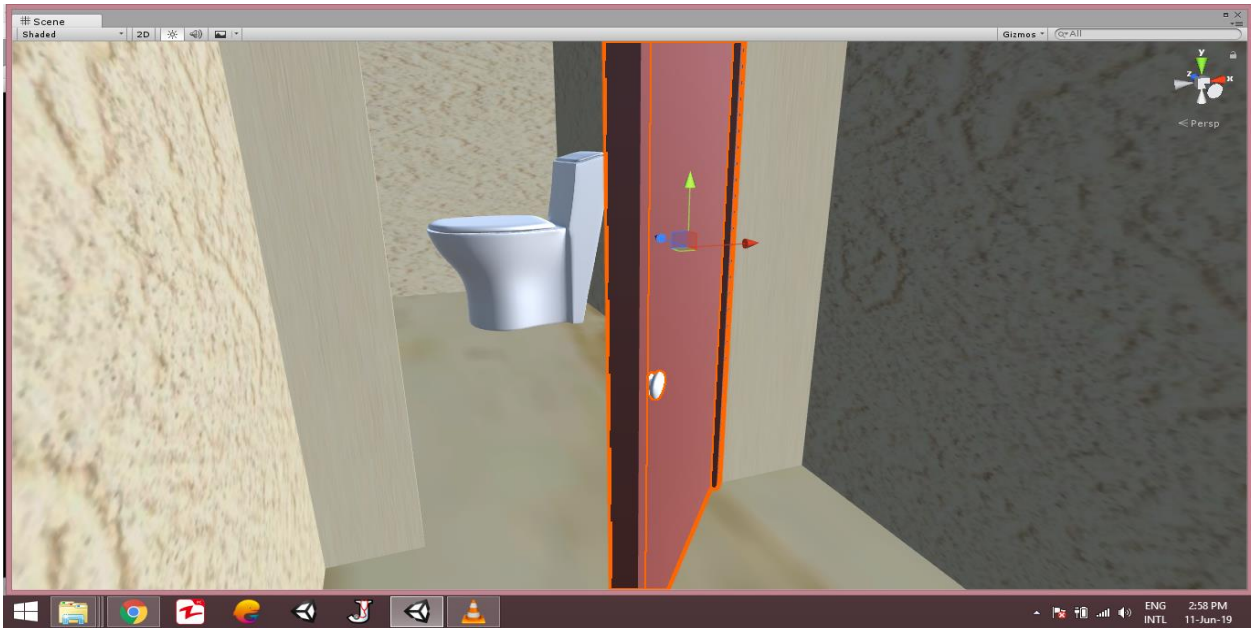


Figure 5. 11: Faculty Room Toilet

5.3.8 University Faculty (Office)

Figure 5. 12: Faculty Office

5.3.9 Headshot

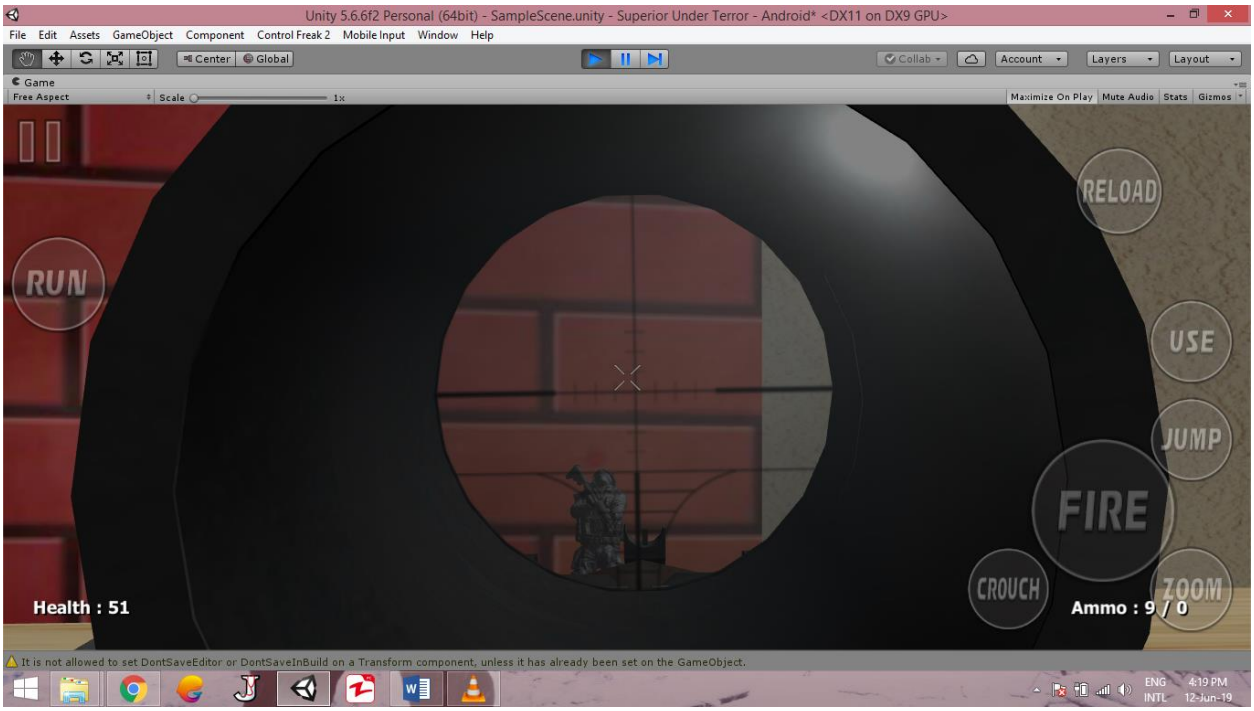


Figure 5. 13: Headshot

5.3.10 Health Reduction

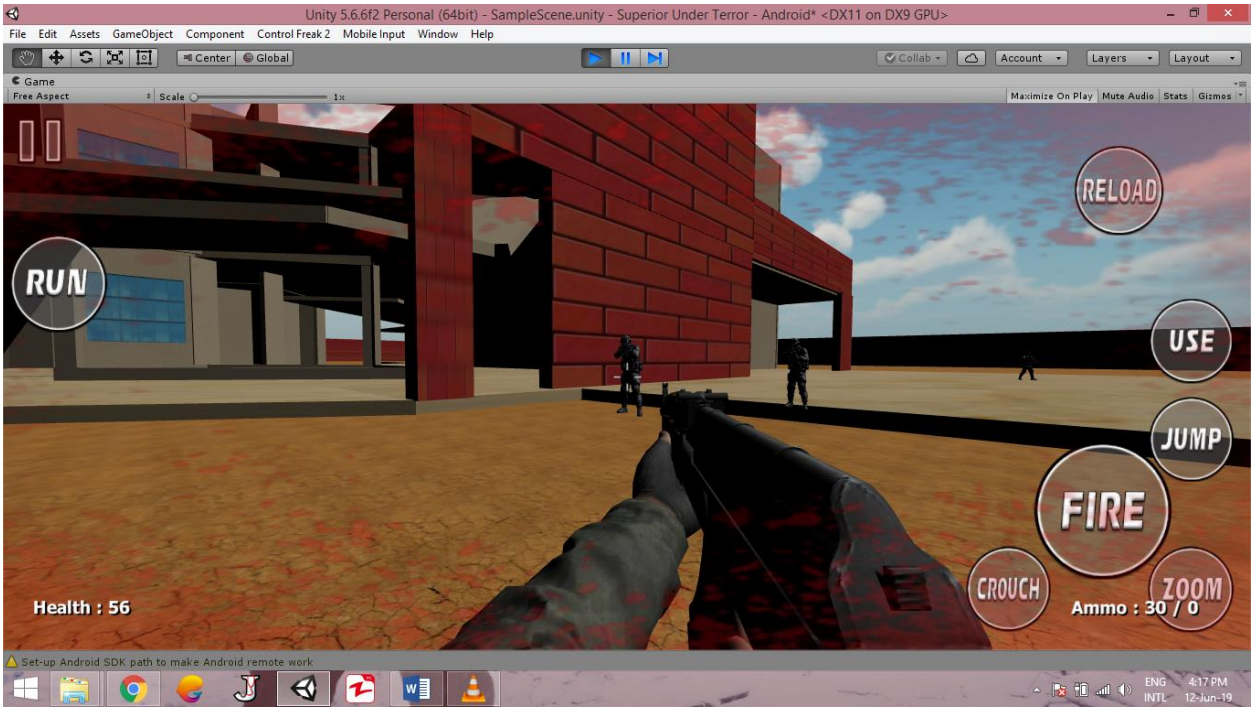


Figure 5. 14: Health Reduction

5.4 Tools and Techniques

Tools	Outcomes

Unity	Game Environment Scripts
MS Project	Gantt Chart
MS Visio	UML Digrams
MS Word	Documentation

Table 5. 1: Tools and Techniques

5.5 Tools and Techniques

5.5.1 Unity3D

(Unity 3D, n.d.) is a powerful cross-platform 3D engine and a user friendly development environment. Easy enough for the beginner and powerful enough for the expert; Unity should interest anybody who wants to easily create 3D games and applications for mobile, desktop, the web, and consoles.

The Unity application is a complete 3D environment, suitable for laying out levels, creating menus, doing animation, writing scripts, and organizing projects. The user interface is well organized and the panels can be fully customized by dragging and dropping.

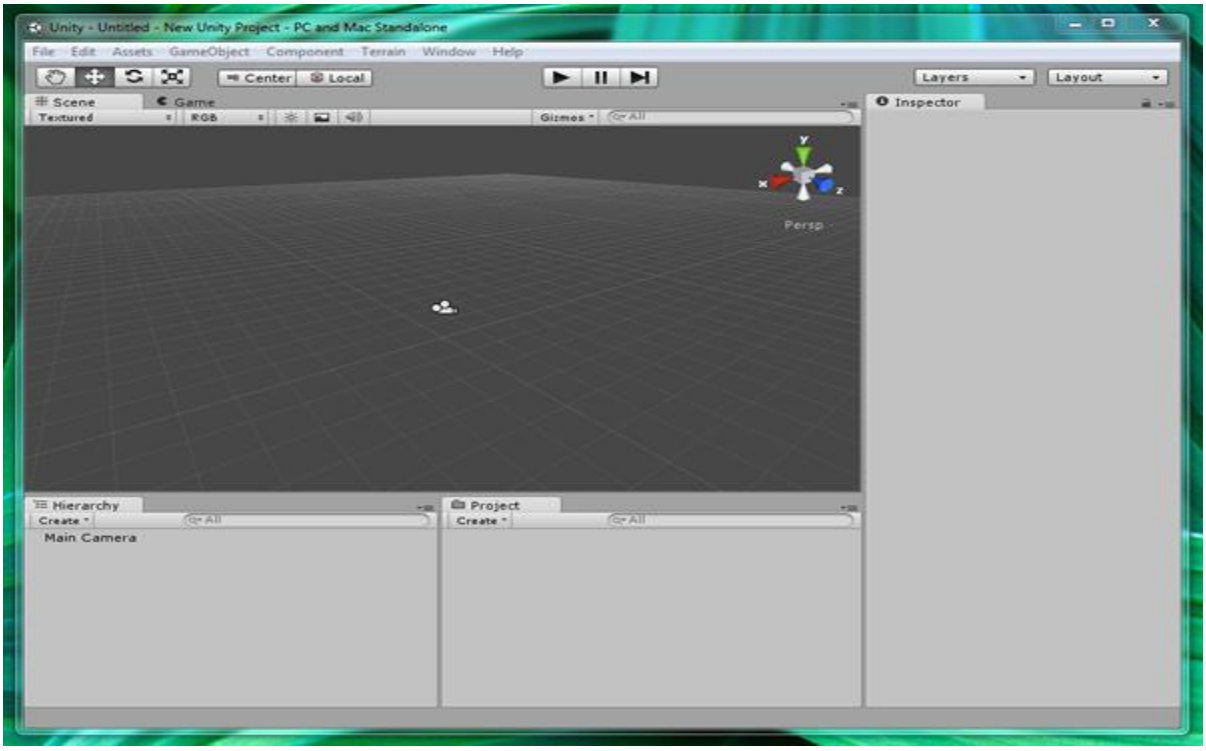


Figure 5. 15: Unity Environment

The Project panel is where all the assets within a project are stored. When assets are imported, they will first appear here.

The hierarchy panel is where assets are organized in a scene. Assets from the Project panel can be dragged into the Hierarchy panel to add them to the current scene.

The Inspector panel lets you inspect and adjust all the attributes of a selected asset. Everything, from its position and rotation whether it's affected by gravity or able to cast a shadow.

The Scene panel is a 3D viewport where you can physically arrange assets by moving them around in 3D space. You can navigate the viewport by panning, rotating, and zooming the view.

When it comes to running your game, it couldn't be simpler. Just press the play button. To stop it, press the play button again. You can even pause your game during play to inspect your scene.

A Unity project is an ordinary folder containing every resource that belongs to your game. Creating a new project is a straightforward affair.

Click File > New Project

Click the Create New Project tab

Browse to a suitable folder

Click Create

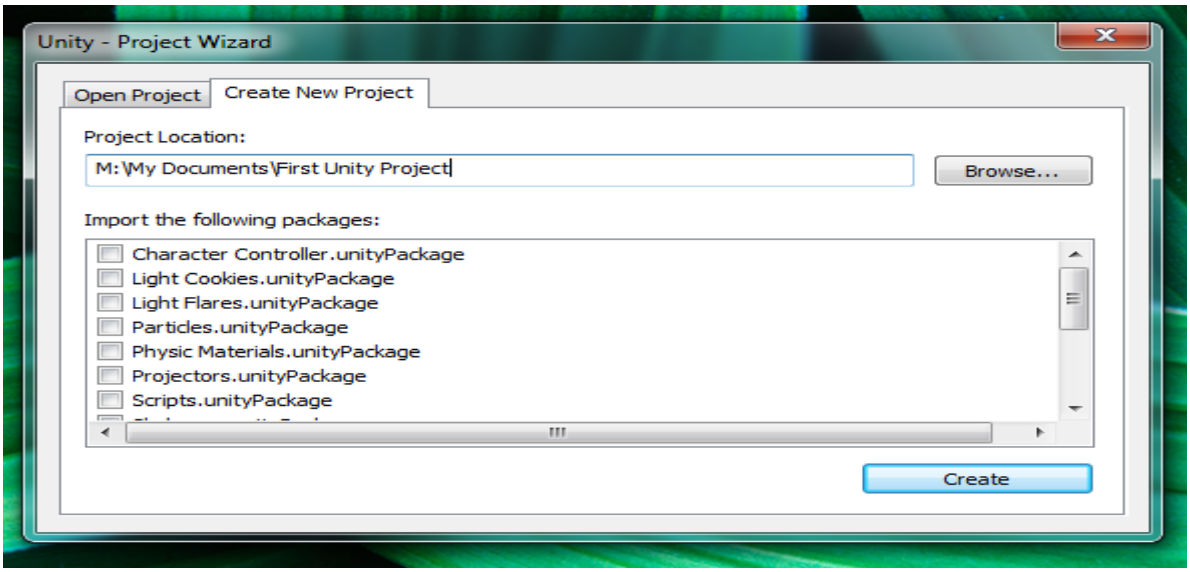


Figure 5. 16: Creating Project in Unity

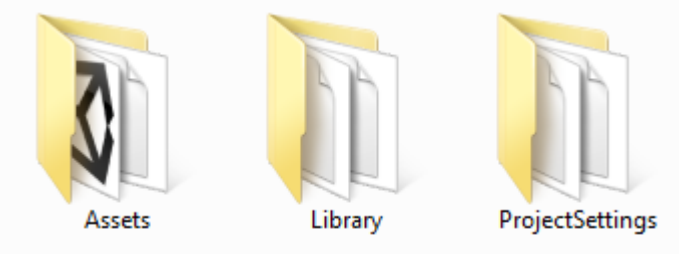


Figure 5. 17: Unity

Assets

Assets are any resource your game uses. These include 3D models, materials, textures, audio, scripts, and fonts, to name a few. Other than a few simple objects such as cubes and spheres, Unity can't actually create most of these assets. Instead, they must be created externally using 3D modeling applications and painting tools and then imported into Unity.

Note: Unity has an Asset Store where you can purchase 3D models, characters, textures, sound effects, music, tools, and even scripts. The Unity Asset Store has quickly become an invaluable resource for game developers and a money making venture for artists and tool developers.

Scenes

Scenes are where you can drag in project assets and arrange them to make levels and game screens. The Hierarchy panel represents the contents of the current scene in a tree-like format. While the Scene panel is ideal for arranging your scene's assets in 3D space, the Hierarchy is where you'll spend most of your time actually organizing your scenes and keeping them tidy.

When you start a new project, Unity automatically creates a new scene for you. Scenes start out with nothing but a camera. If you were to run the game now, you won't see anything but the background color. To give us something to look at:

Scenes are assets and should be saved in your project just like other assets. To save your scene:

- Click File > Save Scene
- Navigate to your project's Assets folder
- Name your scene Main
- Click Save

Scripting

Scripts, known in Unity as behaviors, let you take assets in your scene and make them interactive. Multiple scripts can be attached to a single object, allowing for easy code reuse. Unity supports three different programming languages; Unity Script, C#, and Boo. Unity Script is similar to JavaScript and Action Script, C# is similar to Java, and Boo is similar to Python. Depending on your background you may feel more comfortable with one or the other.

5.5.2 MS Word

(Microsoft Word, 2013) Or simply Word is a word processor developed by Microsoft. It was first released on October 25, 1983 under the name *Multi-Tool Word* for Xenix systems. Subsequent versions were later written for several other platforms including IBM
PCs running DOS (1983), Apple Macintosh running the Classic Mac OS (1985), AT&T Unix
PC (1985), Atari ST (1988), OS/2 (1989), Microsoft Windows (1989), SCO Unix (1994),
and macOS (formerly OS X; 2001).

5.5.3 MS Visio

(Microsoft Visio, 2013) is a diagramming and vector graphics application and is part of the Microsoft Office family. The product was first introduced in 1992, made by the Shapeware Corporation. It was acquired by Microsoft in 2000.

5.5.4 MS Project

(Microsoft Project, 2015) is a project management software product, developed and sold by Microsoft. It is designed to assist a project manager in developing a plan, assigning resources to tasks, tracking progress, managing the budget, and analyzing workloads.

5.1. Best Practices / Coding Standards

Coding standards are a set of guidelines, best practices, programming styles and conventions that developers adhere to when writing source code for a project. In our project we fully struggle to meet maximum number of coding standards

- Avoid complexities.
- Write efficient code.
- Follow OOP concepts.
- Avoid gold plating.

5.2. Version Control

Unity supports version control integration with Perforce and Plastic SCM. Refer to those pages for specific information regarding your choice of version control.

Using a version control system makes it easier for a user (or multiple users) to manage their code. It is a repository of files with monitored access. In the case of Unity, this is all the files associated with a Unity project. With version control, it is possible to follow every change to the source, along with information on who made the change, why they made it, and what they changed. This makes it easy to revert back to an earlier version of the code, or to compare differences in versions. It also becomes easier to find out when a bug first occurred, and with what changes might have caused it.

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

This chapter includes some test cases for the game to check if the game works properly in various situations. We are giving four test examples for eight different situations here. In this whole chapter we check each aspect of game according to its performance and appearance. Testing each part of the game part give us a detailed report of our game which gives us a better understanding of game and its life cycle.

6.1. Use Case Testing

In use case testing, we generally test our game covering all aspects.

User Interface(UI)	TEST in Landscape mode
	TEST for animation, movement of character, graphics, Zoom In/Out (all gestures) etc.
	There should not be any clipping (cut background)
	Test if there is no overlapping of objects
	Character should not move out of the screen/specified area
	Font displayed (color, size etc)
Performance during game	TEST the loading time of a game
Core structure of game	TEST game area, game logic
	Menu Settings
Game Settings	Check the game Graphics and other settings etc.
Device , Android	Check in supported screen sizes and Android versions (basically depend upon Client requirement)

Table 6. 1: Use Case Testing

6.2. Equivalence partitioning

Invalid	**Valid**	**Invalid**
Player Health < 1	Health in between 1-100	Player Health > 100
Move speed < 100	Move Speed = 100	Move Speed >100
Shot Speed > 50	Shot Speed = 50	Shot Speed < 50
Spawn Points in level >1	Spawn Points in level 1	Spawn Points in level <1
Fire Damage > 10	Fire Damage = 10	Fire Damage < 10

Table 6. 2: Equivalence Partitioning

6.3. Boundary value analysis

Invalid (min-1)	Valid (min, +min, -max, max)	Invalid (max+1)
Player Health (-1)	Health in between (1-100)	Player Health (101)
Move speed (1-99)	Move Speed = 100 (100)	Move Speed (101)
Shot Speed (-49)	Shot Speed (50)	Shot Speed (51)
Spawn Points in level (1)	Spawn Points in level (0-1)	Spawn Points in level (1)
Fire Damage (9)	Fire Damage (10)	Fire Damage (11)

Table 6. 3: Boundary Value Analysis

6.4. Unit testing

With the help of unity testing you can avoid from stupid bugs and write a sustainable code for your application. By using unit testing you get the early feedback through series of tests and avoid any type of regression in game and enrooted to object oriented design and create during documentation of your code.

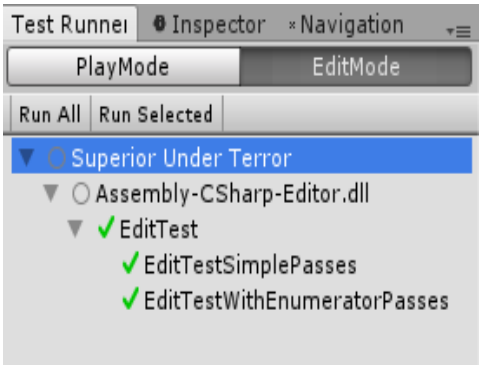


Figure 6. 1: Unit Testing

Unity itself provides the features of unit testing of game in its tool. So we ran our script tests for the betterment and find there is no error to be fixed in this code all the scripts to be tested and unity made them clear.

6.5. Data flow testing

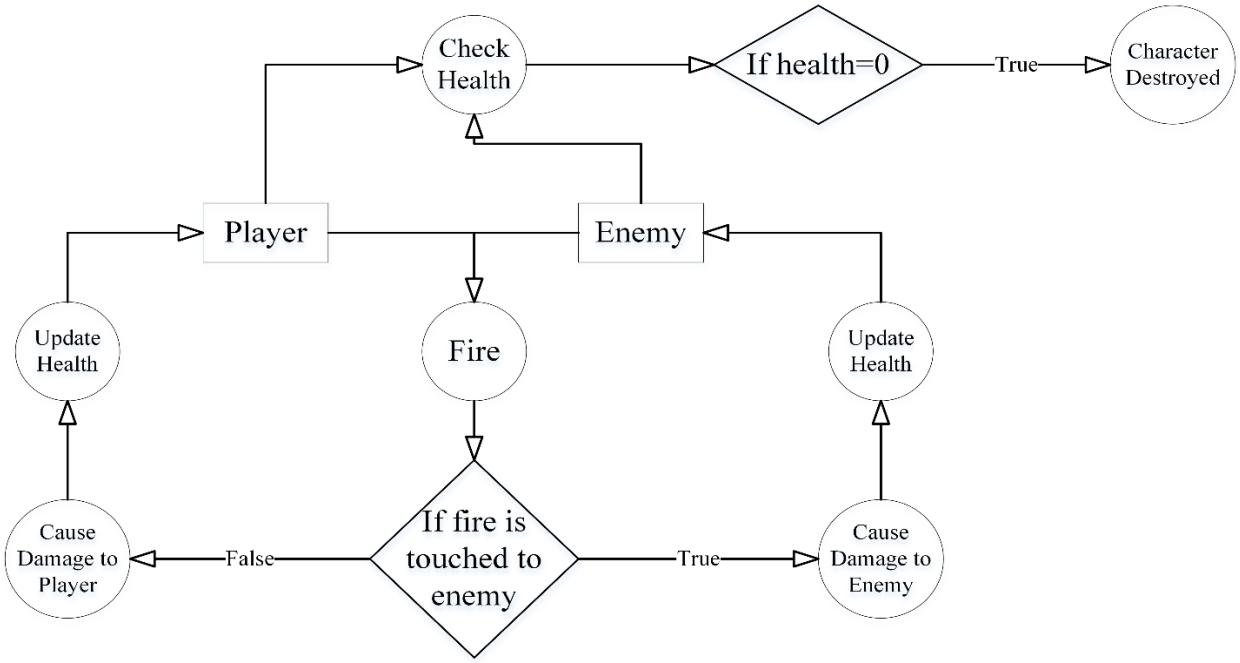


Figure 6. 2: Data Flow Testing

6.6. Integration testing

Our game consists of total 5 scenes that are given below. We test all the scenes step by step:

Figure 6. 3: Game Scenes

In first Scene that is based on UI we test all the buttons and their respective trigger action.

Figure 6. 4: Main Menu

Next scene is level selection which is triggered when play button is touched.

Figure 6. 5: Level Selection

After selecting the level, next scene is loading scene.

Figure 6. 6: Loading Scene

After loading scene, game, after some time, moves to selected level screen.

Figure 6. 7: Gameplay Scene

6.7. Performance testing

For testing the performance of our game unity, itself, provide a feature to test the performance of your game using profiler. It can test the GPU and CPU requirement of your game and displays

the result right away when you play it in the unity. We can test the CPU usage, rendering unit, memory consumption and many other related things.

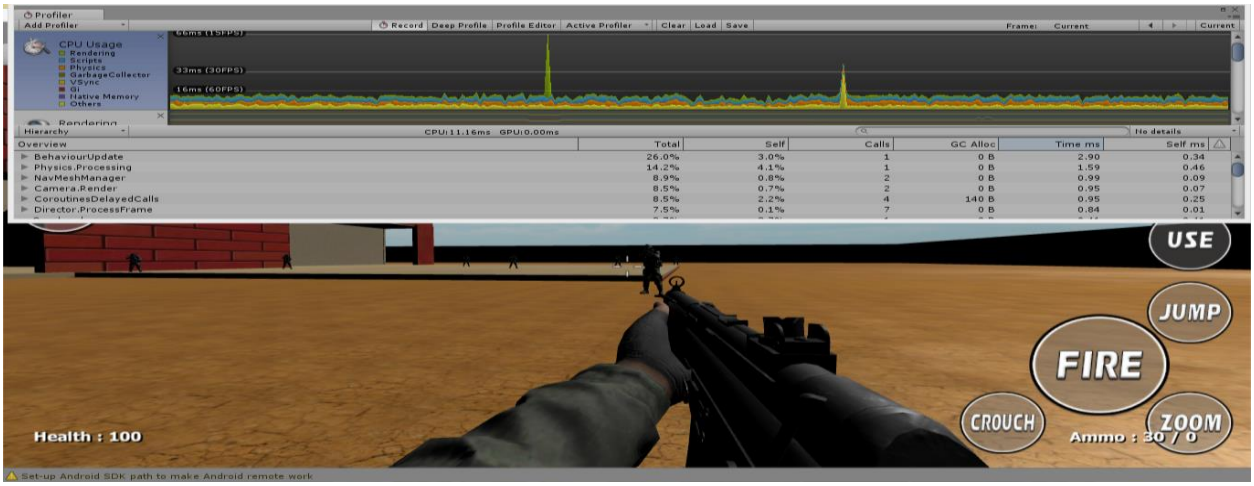


Figure 6. 8: Performance Testinng

6.8. Stress Testing

In this testing we test the strength of a software by pushing to its limits so that we can came to know about how much this software goes far. In our game we ran a series of test in the unity to test the stability of game and conclude that how much our system bear the system load in the game we ran through different devices.

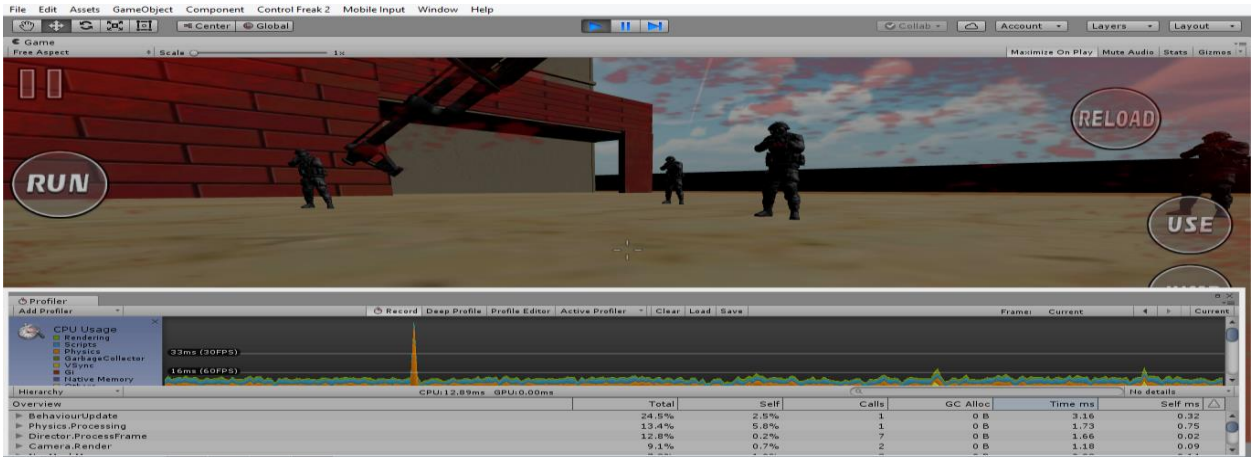


Figure 6. 9: Stress Testing

Above image show that the all the enemies are attacking perfectly and causing no critical or serious issue.

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

We are making game on unity named **"Superior under Terror"**. The game is based on a situation where superior university is hijacked by the terrorists. In our game there is only one character with multiple features and actions. While playing game we can select only one mission at a time to play. Our game is an action game. Our game is, basically, a first person shooting game which has some sort of a story line. In this game user can also learn how to survive in difficult situations. The player will fight against different enemies at different locations in superior university where enemies will locate. The game will be played locally android mobiles.

7.2. Achievements and Improvements

During this process of game development we learnt a lot of skills like:

- Unity Engine Understanding
- Working with a 3D objects
- Creating an Environments
- Coding practices
- Physics
- Dimensions
- C# scripting
- Building and running mobile games in various environments
- Exposure in a gaming field
- Logic behind gaming
- Creative thinking

These are the achievements and improvements we think we learn so far from this final project and that takes our interest in gaming to take pursue our future with this field.

7.3. Critical Review

Superior under Terror is first of its kind that provide the environment of superior university for the shooting. As it is for the first time we used unity and were not well known about the

environment development in unity or game coding. So it was difficult for us to develop the game and there are some lags in the game.

7.4. Lessons Learnt

We absorb very much from this project. This project sharpens our skills in **Unity,** designing tools and many management concepts as well as how to deal with a problem and how to stick for finding the solution of any problem until you found. As well as technical skills this project also enhance our personal development skills such as team working, dedication. We learnt various types of techniques in gaming development for the creation of game. Our passion of making a unique game that contains the enjoyment element in an innovative environment and we succeeded it. **We continue our work after this to make it through worldwide.** This project is a stepping stone for us in the beginning we don't even know that if we accomplished it or not but due to our hardworking and research we do it and it's an achievement for us that motivates us to do better and more.

7.5. Future Enhancements/Recommendations

So generally, there are many enhancements we want to put in our game i.e.

- Enhance more the Multiplayer functionality of game.
- Adding various Interactive maps in game.
- Adding a better survival mode and techniques.
- Adding more functionalities of our players and enemies.
- Increase the performance of game.
- Make our network servers in a large scale according to user rate.
- Adding new enemies with their specific skins same as our player skins.
- Create a better competition environment between players with their enemies.
- Enhance the visual of game
- Enhance the visual of game
- Adding a survival mode
- Create a competition between players.

Appendices

Appendix A: User Manual

In this appendix section we describe the different phases of user interface and also describe how user can use our project. Secondly we describe our promotional plan and promotional materials like broacher, banner, standee and other marketing materials. We also describe other interfaces of our game project.

A. 1: Game Screen

A.1. Fire

This icon is for firing. As you select any weapon, touching this icon will fire the bullets through selected weapon. If the bullets are empty, this icon will hit the enemy with the selected weapon.

A.2. Reload

This icon is for reloading the weapon.

A.3. Use

This icon is for picking up the weapon.

A.4. Jump

This icon is used to jump the player.

A.5. Zoom

This icon is for sniping or zooming for aim accuracy.

A.6. Crouch

This icon is used to crouch.

A.7. Health

This label shows the current health of player.

A.8. Pause

This icon is used to pause the game.

A.9. Run

This icon is for moving character.

Appendix B: Administrator Manual

There is no administration manual in our game “Superior under Terror”.

Appendix C: Information / Promotional Material

C.1. Broacher

Figure C. 1: Broacher

C.2. Flyer

Figure C. 2: Flyer

C.3. Banner

Figure C. 3: Banner

C.4. Standee

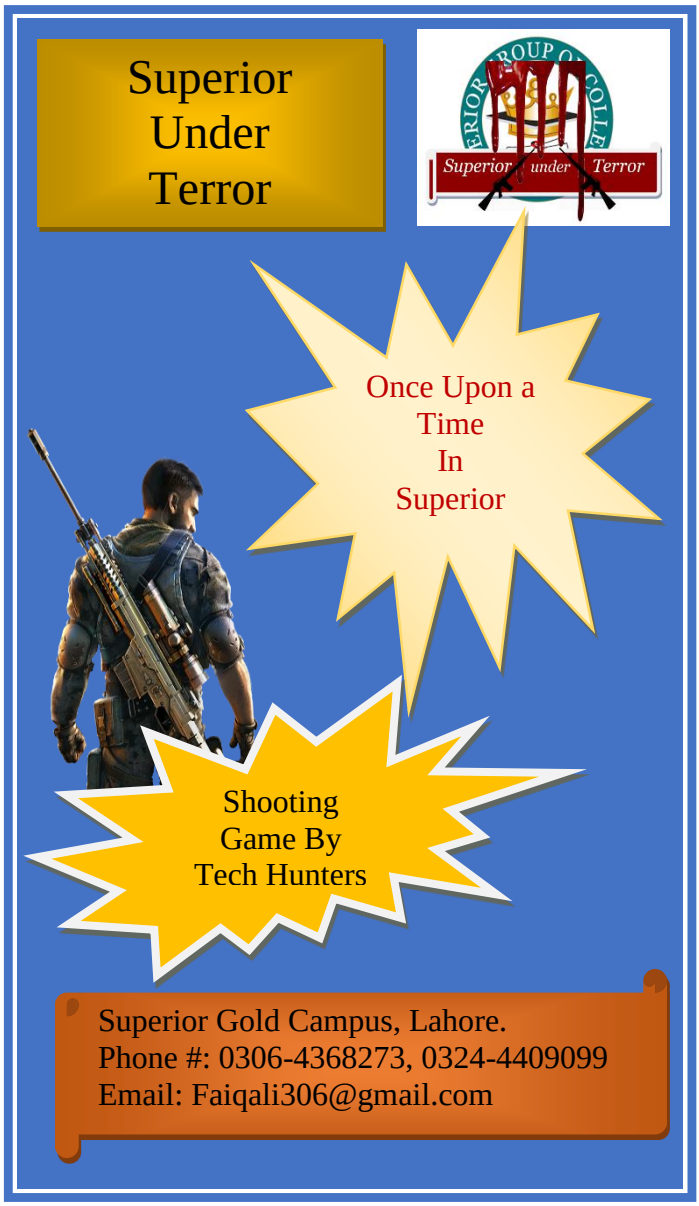


Figure C. 4: Standee

C.5. Logo

Figure C. 5: Logo

Reference and Bibliography

References and Bibliography

(n.d.).
Android. (n.d.). Retrieved from Wikipedia:
https://en.wikipedia.org/wiki/Android_(operating_system)
Google. (n.d.). Retrieved from www.google.com
Microsoft Project. (2015, January 11). Retrieved from Wikipedia:
https://en.wikipedia.org/wiki/Microsoft_Project
Microsoft Visio. (n.d.).
Microsoft Visio. (2013, January 11). Retrieved from Wikipedia:
https://en.wikipedia.org/wiki/Microsoft_Visio
Microsoft Word. (2013, January 11). Retrieved from Wikipedia:
https://en.wikipedia.org/wiki/Microsoft_Word
The Superior University. (n.d.). Retrieved from Superior Group: www.superior.edu.pk
Thomos Malone. (1952). Retrieved from Wikipedia:
https://en.wikipedia.org/wiki/Thomas_W._Malone
Unity 3D. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Unity_(game_engine)