

SECRET COMMANDO: A THIRD PERSON SHOOTING GAME

BY

**KAZI SAFAET ULLAH
ID: 161-15-7569**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering
Supervised By

Moushumi Zaman Bonny
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Ahmed Al Marouf
Lecturer
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2019

APPROVAL

This Project/internship title “Secret Commando: A Third Person Shooting Game”, submitted by Kazi Safaet Ullah, ID No: 161-15-7569 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 07.12.2019.

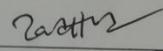
BOARD OF EXAMINERS



Dr. Syed Akhter Hossain
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

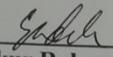
Chairman



Md. Zahid Hasan
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

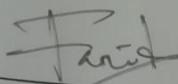
Internal Examiner



Sadekur Rahman
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Dewan Md. Farid
Associate Professor

Department of Computer Science and Engineering
United International University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Moushumi Zaman Bonny, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



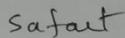
Moushumi Zaman Bonny
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:



Ahmed Al Marouf
Lecturer
Department of CSE
Daffodil International University

Submitted by:



Kazi Safaet Ullah
ID: 161-15-7569
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First I express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

I really grateful and wish our profound my indebtedness to **Moushumi Zaman Bonny**, **Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Game Development*” to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

I would like to express our heartiest gratitude to Prof Dr. Syed Akhter Hossain and Head, Department of CSE, for his kind help to finish my project and also to other faculty member and the staff of CSE department of Daffodil International University.

I would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

ABSTRACT

This report is intended as a guide for users as users get a better idea about the game. For Example, which types game it is, how to play it, why it play and what can be learn. A user can control the player in front of screen. From this report users get user manual, how to shoot bullet, how to throw grenade, search enemies and identify enemies. User also gets idea on how to attack enemies and kill them, when player will be die and also how to complete game level etc. An user obtain better idea on commando attack by playing this game. User can also know about some technique in this game. From this report user can learn many things and when users play this game, they can pass their leisure time efficiently. So our young generation can pass their time better than the before, so they will be right way.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	I
Declaration	II
Acknowledgements	III
Abstract	IV
CHAPTER	
CHAPTER 1: INTRODUCTION	1-2
1.1 Introduction	1
1.2 Motivation	1
1.3 Objective	1
1.4 Expected Outcome	2
1.5 Report Layout	2
CHAPTER 2: BACKGROUND	3-5
2.1 Introduction	3
2.2 Related Works	3
2.3 Comparative studies	4
2.4 Scope of the Problem	5
2.5 Challenge	5
CHAPTER 3: REQUIREMENT SPECIFICATION	6-9
3.1 Business Process Modeling	6
3.2 Requirement Collection and Analysis	6
3.3 Use Case Modeling And Description	7
3.4 Logical Data Model	8
3.5 Design Requirement	9

CHAPTER 4: DESIGN SPECIFICATION	10-22
4.1 Front-End Design	10
4.1.1 Player	10
4.1.2 Joykoli stick	11
4.1.3 Bullet	11
4.1.4 Crouch	12
4.1.5 Grenade	12
4.1.6 Reload	13
4.1.7 Scope	13
4.1.8 Bullet Amount	14
4.1.9 Radar	14
4.1.10 Player HP	14
4.1.11 Aim Button	15
4.1.12 Jump Button	15
4.1.13 Enemy	16
4.2 Back-End Design	18
4.2.1 Player Design	18
4.1.2 Animator	19
4.2.3 Enemy Design	20
4.2.4 Nav Mesh Agent	21
4.2.5 Game Level	22
4.3 Interaction Design and UX	23
4.4 Implementation Requirement	23
CHAPTER 5: IMPLEMENTATION AND TESTING	24-29
5.1 Implementation of Database	24
5.2 Implementation of front-end design	24
5.3 Implementation of Interactions	28
5.4 Testing Implementation	29
5.5 Test Results and Reports	29

CHAPTER 6: CONCLUSION AND FUTURE SCOPE	30
6.1 Discussion and Conclusion	30
6.2 Scope of Further Development	30
REFERENCE	31
APPENDIX	32-34

LIST OF FIGURES

FIGURURE	PAGE
Figure 2.1.1: Business Process Modeling	6
Figure 2.3.1: Use Case Modeling	7
Figure 3.4.1: UML	8
Figure 3.5.1: Unity	9
Figure 3.5.2: Visual Studio	9
Figure 4.1.1 Player	10
Figure 4.1.2: Joykoli stick	11
Figure 4.1.3: Bullet	11
Figure 4.1.4: Crouch	12
Figure 4.1.5: Grenade	12
Figure 4.1.6: Reload	13
Figure 4.1.7: Scope	13
Figure 4.1.8: Bullet Amount	14
Figure 4.1.9: Radar	14
Figure 4.1.10: Player HP	14
Figure 4.1.11: Aim Button	15
Figure 4.1.12: Jump Button	15
Figure 4.1.13: Enemy	16
Figure 4.1.14: Front-End Design	17
Figure 4.2.1: Player Design	18
Figure 4.2.2: Player Animator	19
Figure 4.2.3: Enemy	20
Figure 4.2.4 Enemy Animator	20
Figure 4.2.5: Nav Mesh Agent	21
Figure 4.2.6: Game Level	22

Figure 4.3.1: User Interaction Design and UX	23
Figure 5.2.1: Player movement implementation	25
Figure 5.2.2: Player Grenade implementation	25
Figure 5.2.3: Player scope implementation	26
Figure 5.2.4: Player die implementation	26
Figure 5.2.5: Enemy die implementation	27
Figure 5.2.6: Player shoot bullet implementation	27
Figure 5.3.1: Interaction implementation	28

CHAPTER 1

Introduction

1.1 Introduction

This report describes the process and user manual in making a 3D game using unity game Engine. I make this game where, the user search enemies, attack enemies and kill them. In this report, I discuss on game overview including game description and game play. Then I focus on the game design, development, implementation, testing and expected outcome from the game.

1.2 Motivation

Our country has faced a terrorist attack in the recent years. According to intelligence report, many terrorist organizations are very active in our country. So we want make a game using this concept to protect for protecting our country. So, we will develop a game for planning an attack into terrorist secret areas secretly and destroy their arms and collect their wealth. So when our next generations will play this game, they can make their time enjoyable and they would be make distance between their and other illegal activities in the society.

1.3 Objective

This is an offline game where user can go for terrorist hunt and go ahead for searching arms and enemies. User can determine his right route by using terrorist secret sign and different hints. User can watch out around him for protecting himself from enemy. User arms would be limited, so he should use them properly. In this way he can also find out terrorist secret doors, tunnel stairs etc. to reach his goal. After reaching his goal, he must destroy terrorist arms and collect wealth and come back from enemy haunt. Player also should find out end zone of the game so that player can upgrade in next level.

1.4 Expected Outcome

User will get knowledge about commando raids with a third person shooter. He also obtains experience on many sign and better hints. By playing this, user will be happy to control a military man against the enemy. User will discover this game is more attractive than others. User should kill all enemies and then user will find out a end zone. In the end zone there have a trigger collider. When users go into this collider, the level will be end then user can play next level. Our young generations get inspiration against terrorism. User will receive it as a funny game.

1.5 Report Layout

This part is an important part of the report. Because this part shows report full overview. In this part report is formally structure. So when anyone needs something they can easily jump straight to their parts. In this report, I described my project motivation, objective and expected outcome in chapter 1. In chapter 2, I discussed on my project related other project, compare my project to other, scope of the problem and my challenges to make this project. In chapter 3, I designed Business Process Modeling of my project, Use Case Model, Logical Data Model etc. for my project. In Chapter 4, I fully described of my project front-end design, back-end design, and UI/UX design. I also implement my project database, front-end design and testing implementation, in Chapter 5. At last, in my project report, I discussed on my future target and features of my game and how would it work. And how the next update will be work with the ohter

CHAPTER 2

Background

2.1 Introduction

Secret Commando: A Third person Shooting Game is a well-designed game in android game development sector. Because other developers develop their game in different concept. And they give users some basic feature. Some of their feature is locked. That means when user want to this feature, they should pay for it. In my project, I tried my best to give users more features so that they can get more activity in their leisure time and enjoy themselves without any lock feature. So users can use this game in free and feel fresher than before.

2.2 Related Work

In Play store, there are some games like my project. For example, army battle field, mission impossible, cover shooter, gun fire etc. Those game their main concept is shooting enemies, throw grenade on enemies, gun reload, search enemies etc. They offer user to add user database for more efficient. Because they update their game, several time, so uses get new things in new update.in my project, i use some things shoot bullet throw grenade search enemies and kill enemies in my project. I add some new feature like as, player jumping, movement, knee down and navigation mesh agent for enemies. I tried my best to use better graphics in this game. In my project next update i add vehicles system and more level for users to give them more flexibility. I also play Player Unknown Battle Ground (PUBG). This game is online multiplayer games. There also give many features to the user. Those feature also so attractive for any user. So I will try my best to give users more feature so that they get this game so attractive.

2.3 Comparative Studies

Last year i played this types Game. One is "real army rapid commando "and another is mission impossible". Those games were FPS (first person shooter) Games. As a fps games. When user play those games, Users couldn` t see the player and player movement. Users just see front View of the game. In my project, "secret commando: A Third Person Shooting Game" is a TPS (third person shooter) games. As a tps game, when someone play this game, he can see player movement, front view also, there are no grenade in "Real Army Rapid Commando" but in my project i add grenade system. When User throw grenade, it will be blast after 4 sec. If there have any enemy near grenade blast area, this enemy will be die. I also add more user input, which is less in "Real Army Rapid Commando", enemy will attack player, when player close to the enemy else nothing to do. If player away from this enemy, he could not attack and follow player. In my project I used Nav Mesh Agent in enemy so that enemy can follow and attack the player. At this stage, player can get knee down; so that enemy cannot target the player. But in "Real Army Rapid Commando", there had no knee down option for player to protect himself from enemies. In "Real Army Rapid Commando", User can change gun, in play time, but in my project gun must be selected before play time, in my project, uses can see blood effect when player shoot bullet on enemy. But in "Real Army Rapid Commando", there have no blood effect.

2.4 Scope of the Problem

In previous part, I discuss on, how to other developer create their game and which kinds of feature they add in their game. But there is a major problem in user input and player movement in the front of camera view. Because in the previous they use a small number of user input. On the other hand, when user move player using joykoli stick press straight forward, at the mean time player will not target any one and he can run so fast. But when joykoli stick handle bar move forward with an angle, player cannot run fast but he can target enemy. So user would be confused. But I think, I use proper documentation so that user will get it friendlier.

2.5 Challenge

This project was too much challenging for me, in its first to last. When I make animation for Player that it is so challenging to run player with the combinations of animation, locomotion, animator coding. Then player's gun, bullet shoot throw grenade, player movement using joykoli stick, player jumping, player scope, bullet reload, player knee down etc. are also challenging. Then Identification of enemy and find out enemies by using Nav Mesh Agent was so critical. At last when all enemies die, I should print preview of congratulations and level complete. But for show congratulation message, I should use a collider in an empty game object, when Player goes into this collider, congratulation message will be print but there also had a problem. When an enemy go into this collider, editor print this congratulation message also. So I use in unity tags option for Player, Enemies and other game object so that, when only Player tag object go into this collider then it print congratulation message.

CHAPTER 3

Requirement Specification

3.1 Business Process Modeling

Business process Modeling is a strong process to describe any software or games to others. In my games, there are only one admin. He will control all thing in this game like this.

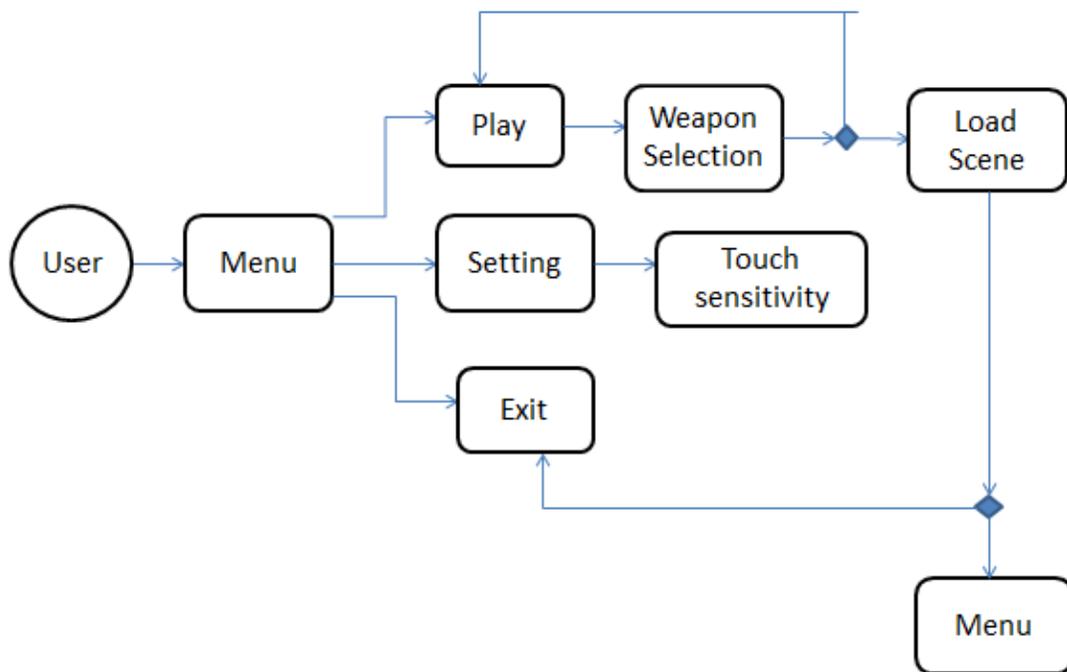


Figure 3.1.1: Business Process Modeling

3.2 Requirement Collection and Analysis

In this project, I used just software no hardware. I analysis on project concept and all requirement for it. I also collected some expectation from people. I also analysis market view of my project. And Which Types of feature should I offer in my project.

3.3 Use Case Modeling

Use case data model is very strong process for understanding any project. In figure 3.3.1, there is only one actor. Because in my games one user can access it at a time in a mobile. So when user press play button it brings user next layout weapon selection. Player can also select setting and exit button. After select any weapon from weapon selection player can see scene and play the game.

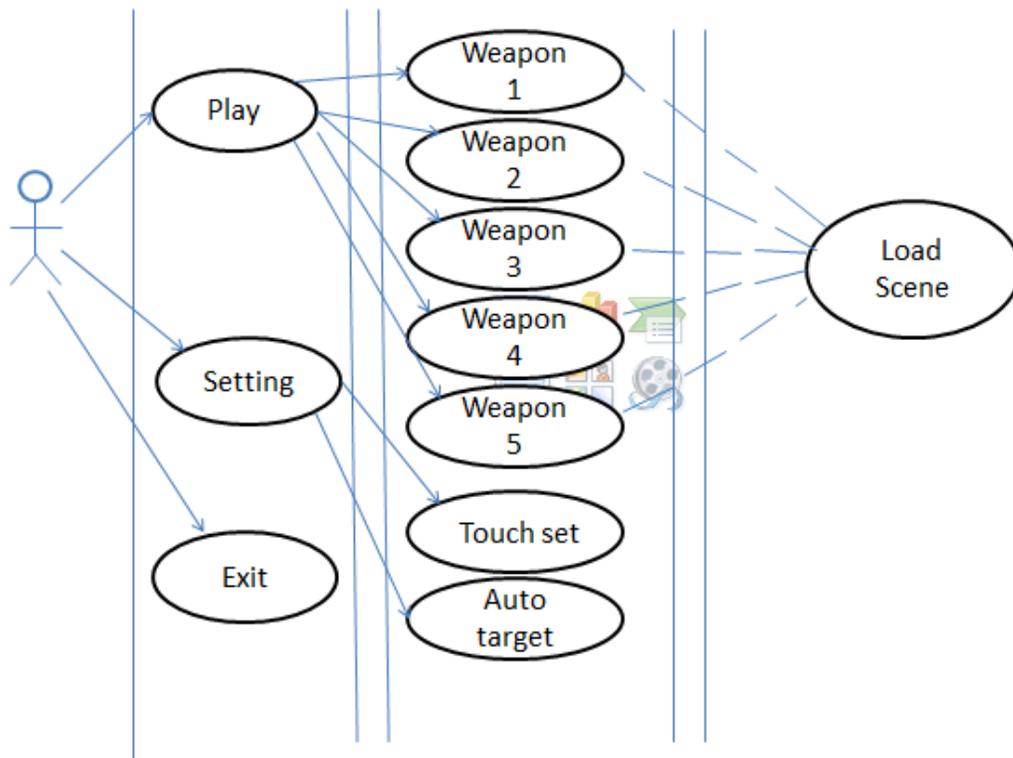


Figure 3.3.1: Use Case Modeling

3.4 UML

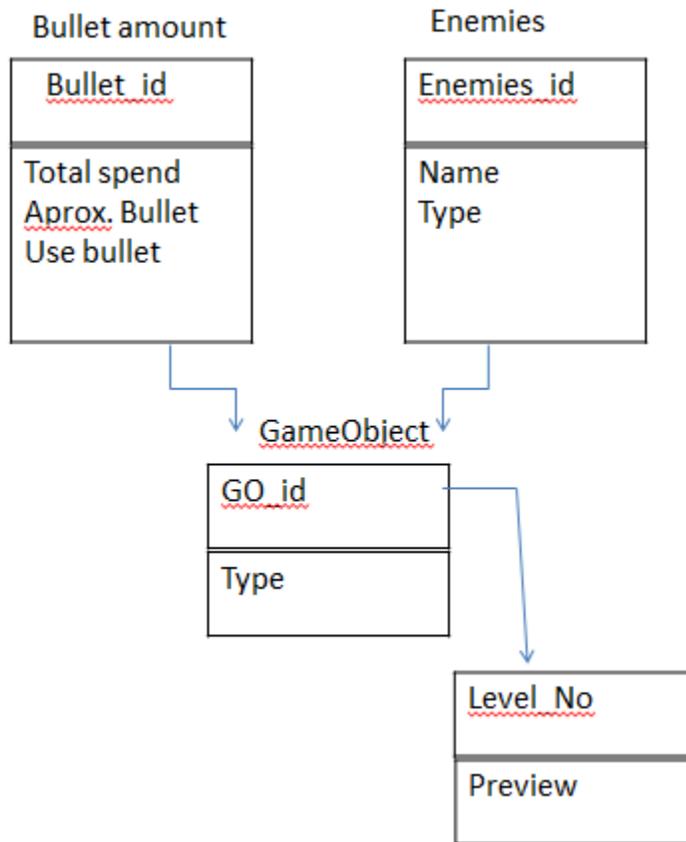


Figure: 3.4.1 UML

In my project, there is no external database. There are only internal data store system for total current enemies, total bullet which is store in a empty object so that when all enemies die then current level will update into next level.

3.5 Design Requirement

In my project, I use some Software, some package, many texture etc. I used Unity game Engine for design my games front view. I use Visual Studio for code in my games. Because Unity Engine just work for game design, but it cannot give proper view without coding. In Unity, I install Probuilder so that I can design different model in small time. For light map data in terrain, I use auto light map data on. I also used Text Mesh Pro for making text in UI more easily.



Figure 3.5.1: Unity



Figure 3.5.2: Visual Studio

CHAPTER 4

Design Specification

4.1 Front-End Design

In My Project, I tried hard so that User can find this game more friendly and attractive. So, I first look in animation of player. Because, when a user play this game he first look in the player. He also Look input system of the game. User also looks at graphics. So when graphics will be better in a game, then this game is more attractive for a user. Now I describe my project frond-end design with figure.

4.1.1 Player: In My project, I try my best to create Player more excellent. I also make Player animation fairer.



Figure 4.1.1 Player

4.1.2 Joykoli stick: In play time, Player can move change direction, change current look at position by using joykoli stick.

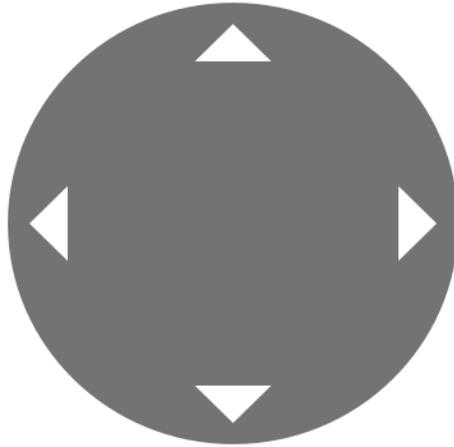


Figure 4.1.2: joykoli stick

4.1.3 Bullet: When Player presses this button, it will be fire bullet.



Figure: 4.1.3 Bullet

4.1.4 Crouch: When User pressed this button, player will be knee down position to protect himself from enemy and attack position on enemy.



Fig 4.1.4 Crouch

4.1.5 Grenade: When User pressed it, Player will be throw grenade on enemy and it will be blast after 3 sec, if there have any enemy near blast area, enemy will be die.



Figure 4.1.5 Grenade

4.1.6 Reload: When reload button is pressed then it will go if statement. If current load amount is 30 then it will not apply for reload. Else amount is less than 30 then it plays reload animation.



Figure 4.1.6: Reload

4.1.7 Scope: Scope is use for zoom in camera to see closely enemies to shoot them. When scope button is pressed, User only see zoom in view he cannot see the player. At that time user can shoot enemies more easily.

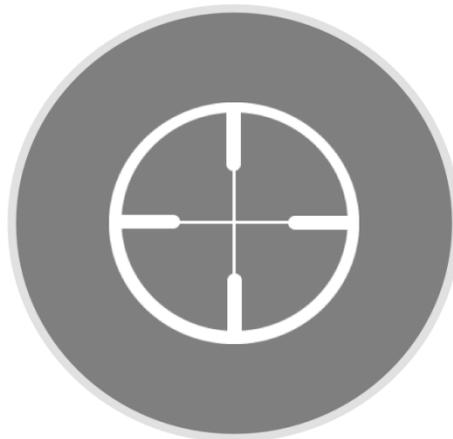


Fig 4.1.7 Scope

4.1.8 Bullet Amount: This text will show current bullet amount and approximate bullet amount right.



Fig 4.1.8 Bullet count

4.1.9 Radar: Enemy will be shown in radar with tags enemy. When an enemy will die then radar able object also remove from the radar.



Figure 4.1.9: Radar

4.1.10 Player HP: When enemy fire bullet to enemy, Player hp will be decrease.



Figure 4.1.10: Player HP

4.1.11 Aim Button: Aim button is like a target point. In this point bullet will be hit.



Fig 4.1.11 Aim button

4.1.12 Jump Button: Player will be jumped when it pressed.



Fig 4.1.12 Jump button

Player will be jumping, when player want to pass a railing or some small wall. If Player want to jumping down, he fall in down with smooth animation.

4.1.13 Enemy: For enemy, I use same body just change in enemy controller and its behavior. In Enemy I add nav mesh agent so that enemy can find out if there have any player close to the enemy. Then he can attack and kill the player. When player run away from the enemy, the enemy can follow the player.

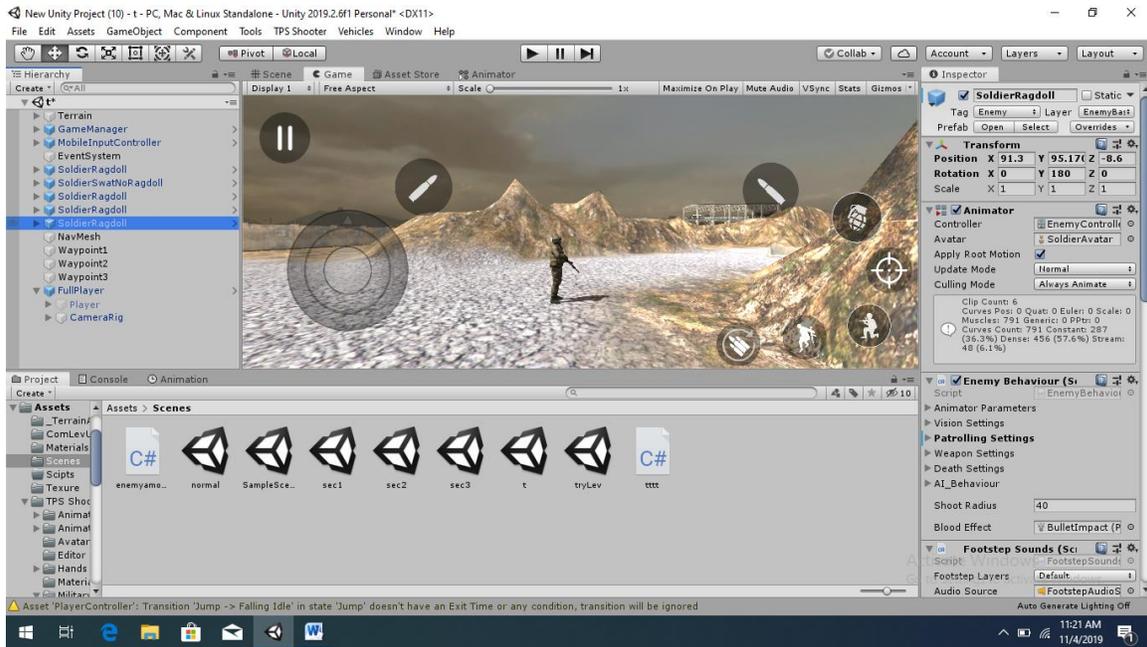


Figure 4.1.13 Enemy

Full view of my project front-end design is



Fig 4.1.13 Front-End Design

4.2 Back-End Design

In back-end design, I use unity editor to design. I also use Text Mesh Pro and Probuilder for design text and small game object. On the other hand, I use Visual Studio to code.

4.2.1 Player Design: For Player design I use Character controller, animation, animator, Player Behaviors script, Player HP script, Player Weapon menu Script etc. I also use player footstep sound as it makes different sound in different layers.



Figure 4.2.1: Player Design

4.1.2 Animator: In animator I use different Layer for Player and Enemy.

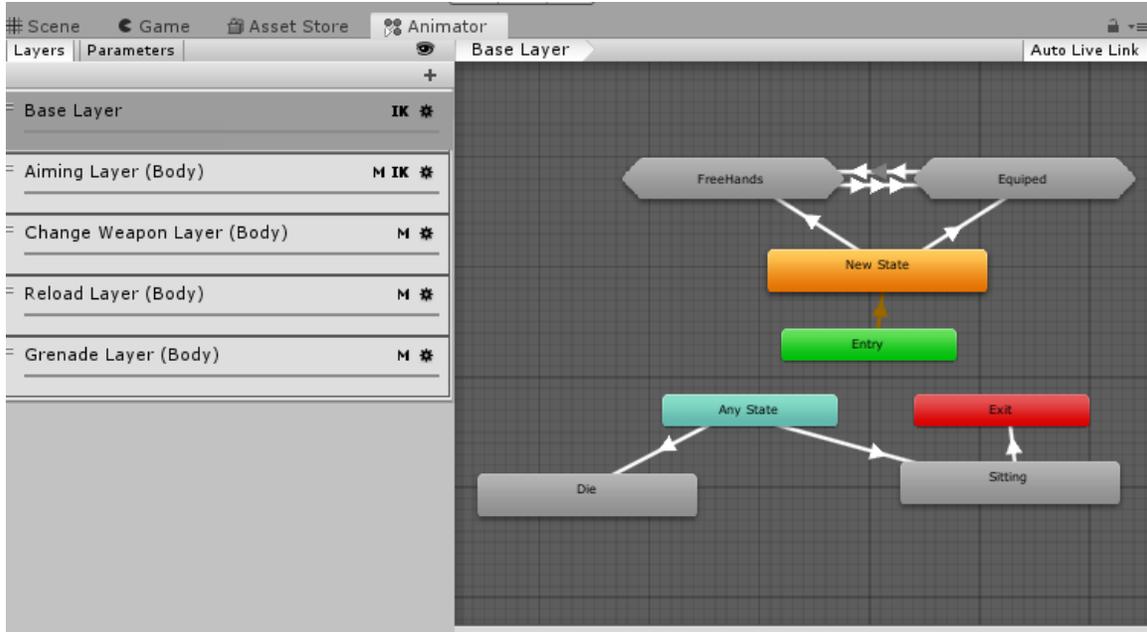


Figure 4.2.2: Player Animator

4.2.3 Enemy Design: For enemy design, I also use same body. But made different in behavior for enemy.

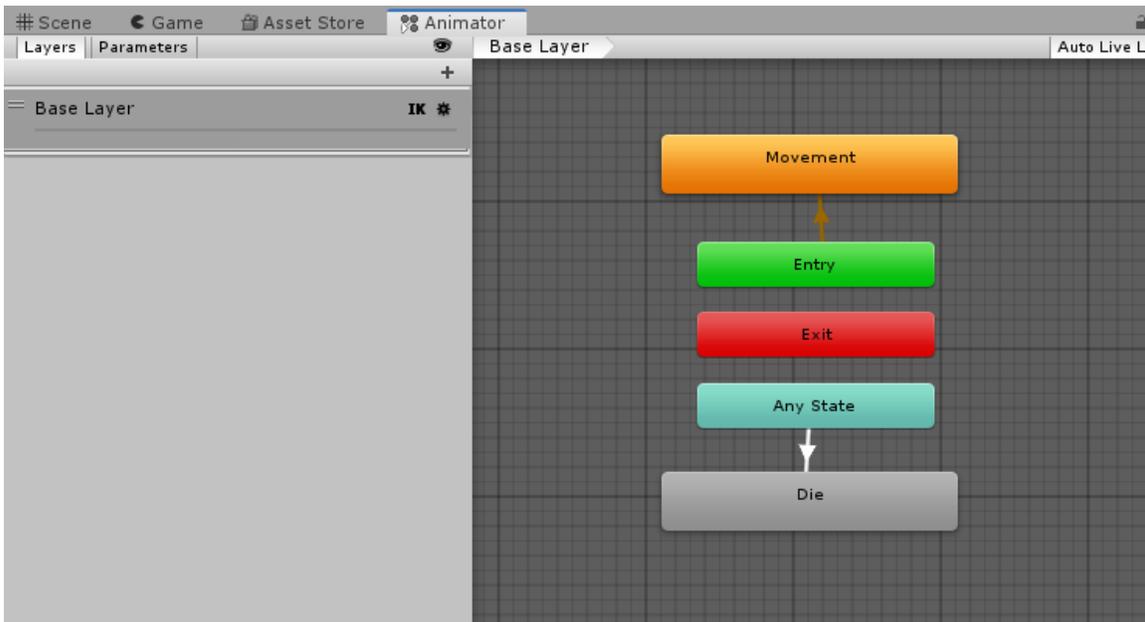
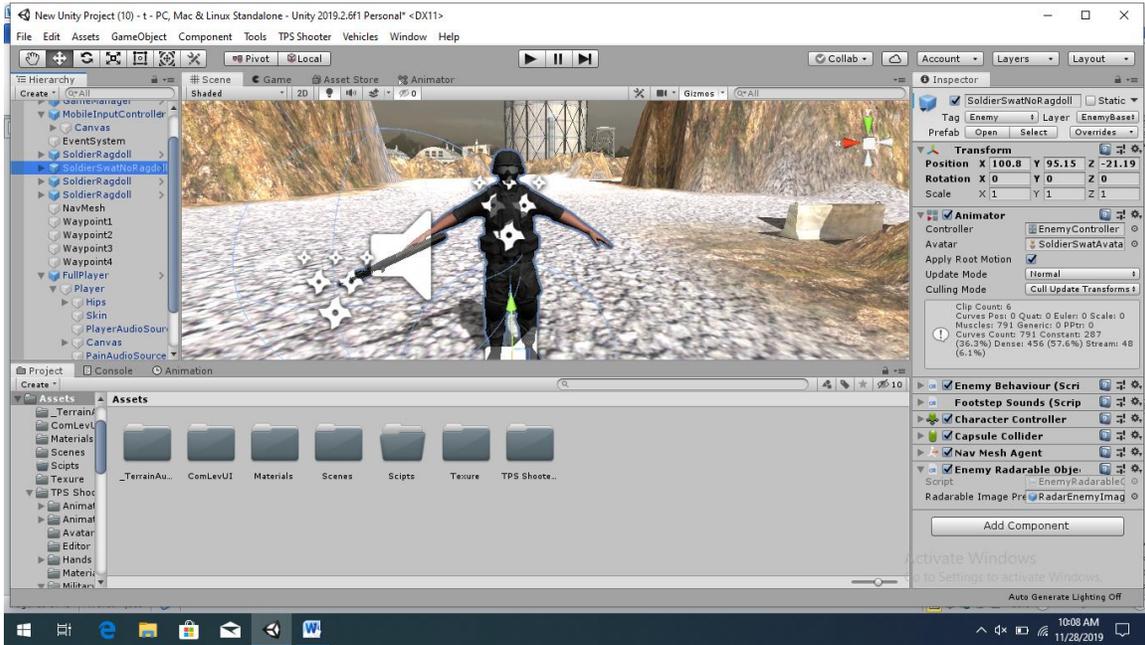


Figure 4.2.3 & 4.2.4 Enemy and its animator

4.2.4 Nav Mesh Agent:

Nav Mesh Agent helps enemy to find out player current position. It also helps enemy to follow player continuously.

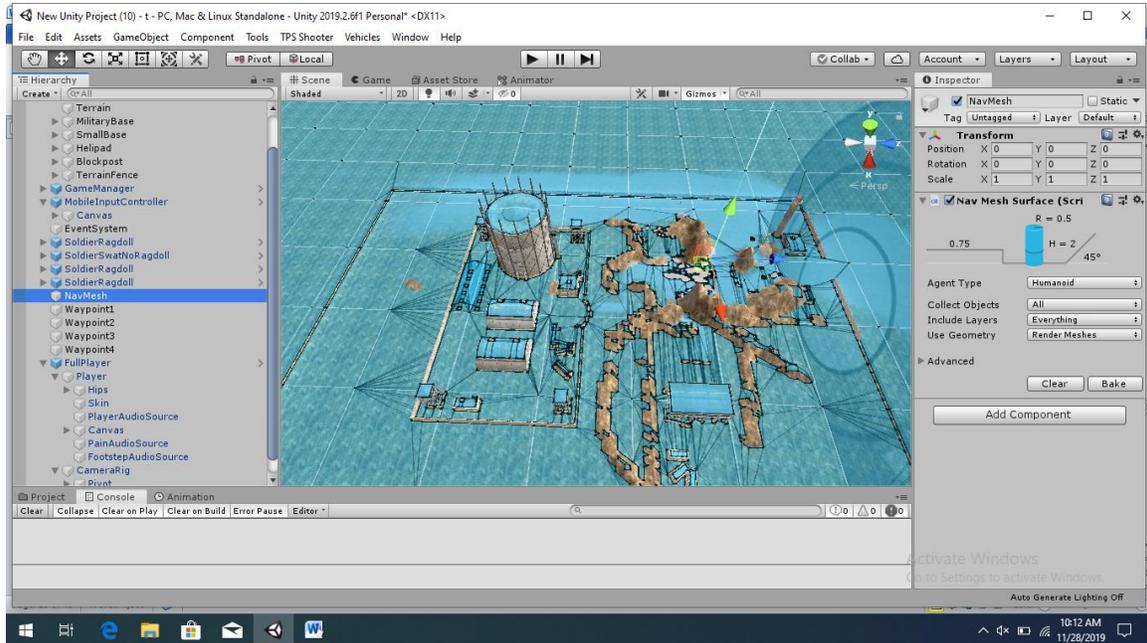


Figure 4.2.5: Nav Mesh Area

4.2.5 Game Level: I design my game level like as a military camp. Firstly I create a terrain. Then I established a military camp on it.

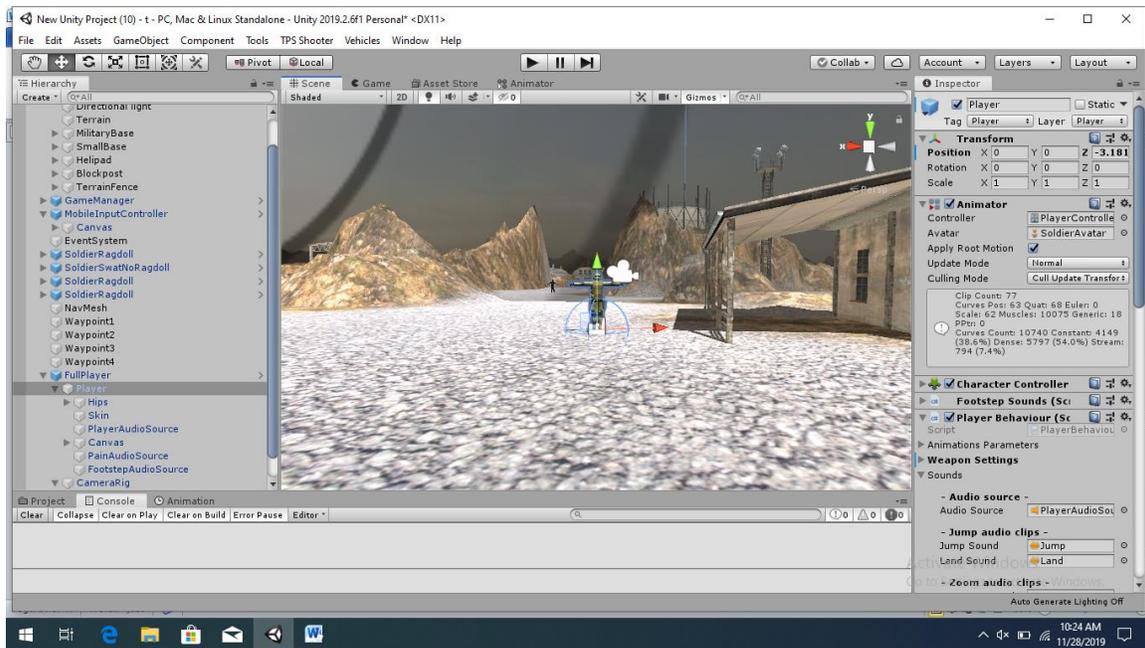


Figure 4.2.6: Game Level

4.3 Interaction Design and UX

In this stage, I use Text Mesh Pro for design Interaction Text. I also use visual studio for code in this section. When User pressed play button then it brings user to next step by using On Click Event. Then User should choose his weapon for play. After choose weapon, if Player click next, it will load next scene by using On Click Event.

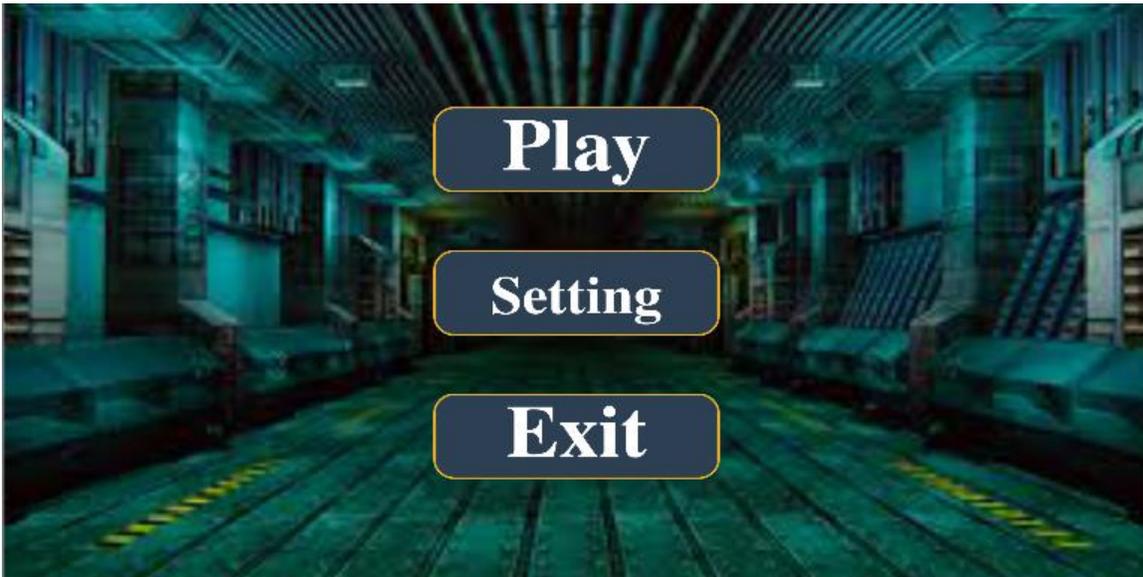


Figure 4.3.1: User Interaction Design and UX

4.4 Implementation Requirement

For implementation of my project, I must need some specific things. Like as long-term goal, plan, move and action of organization to achieve goal. I have a plan and long term goal for my project to achieve my goal regarding this project.

CHAPTER 5

Implementation and Testing

5.1 Implementation of Database

Implementation and testing in my project, I use small database, I just collect player bullet amount. How much player shoot bullet. I also collect Enemies database in my project. I just count enemies total amount. When an enemy dies, I decrease one from enemy database. When enemy amount is zero a collider is open. When Player goes into this collider, it will show congratulation message.

5.2 Implementation of Front-End Design

After complete my project end design, I check User interface in front end design firstly. I check jokoli stick. Is player movement continuing in right way. After, get a good, Performance, I focus on two user input bullet trigger. I check when bullet trigger is pressed, it shoot bullet. Then I Check grenade button is pressed, player throw grenade. I also check on think. When player throw grenade he cannot shoot bullet. I check it with Script. How to code work and how function work, how on click event work. I also check enemy die function and Player die function.

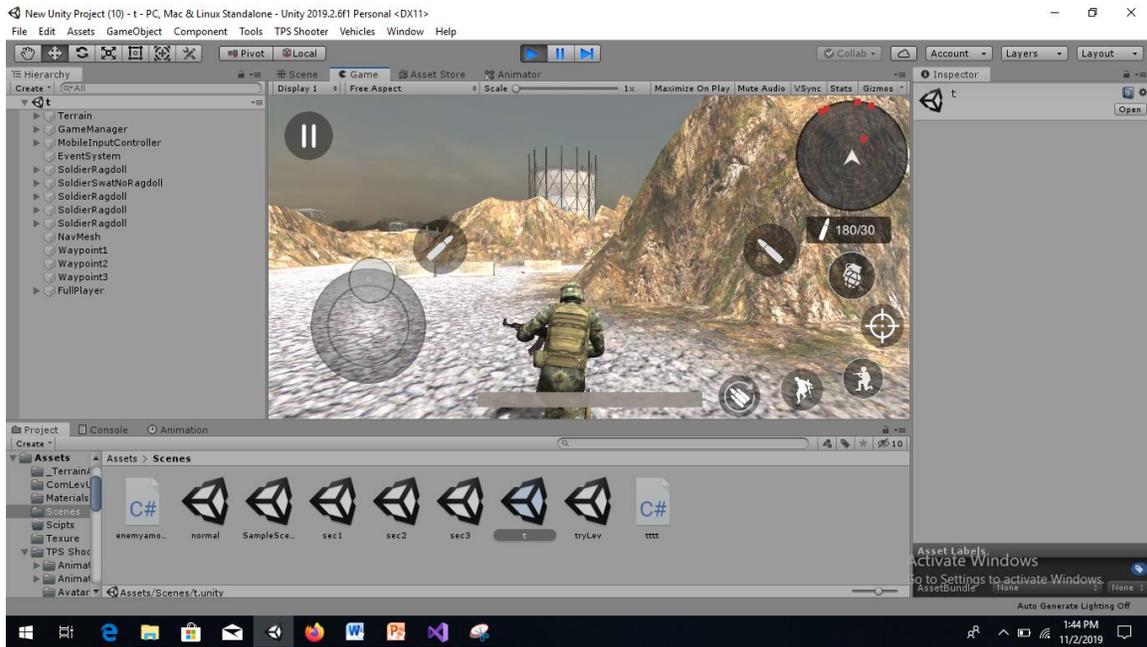


Figure 5.2.1: Player Move Implementation

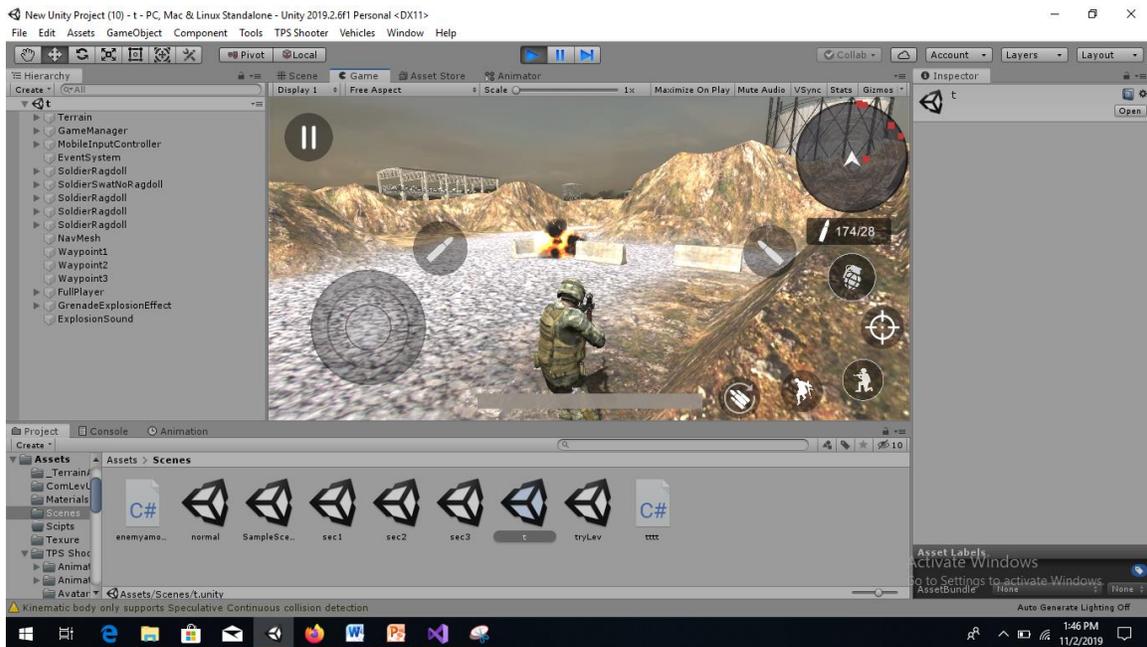


Figure 5.2.2: Player grenade implementation

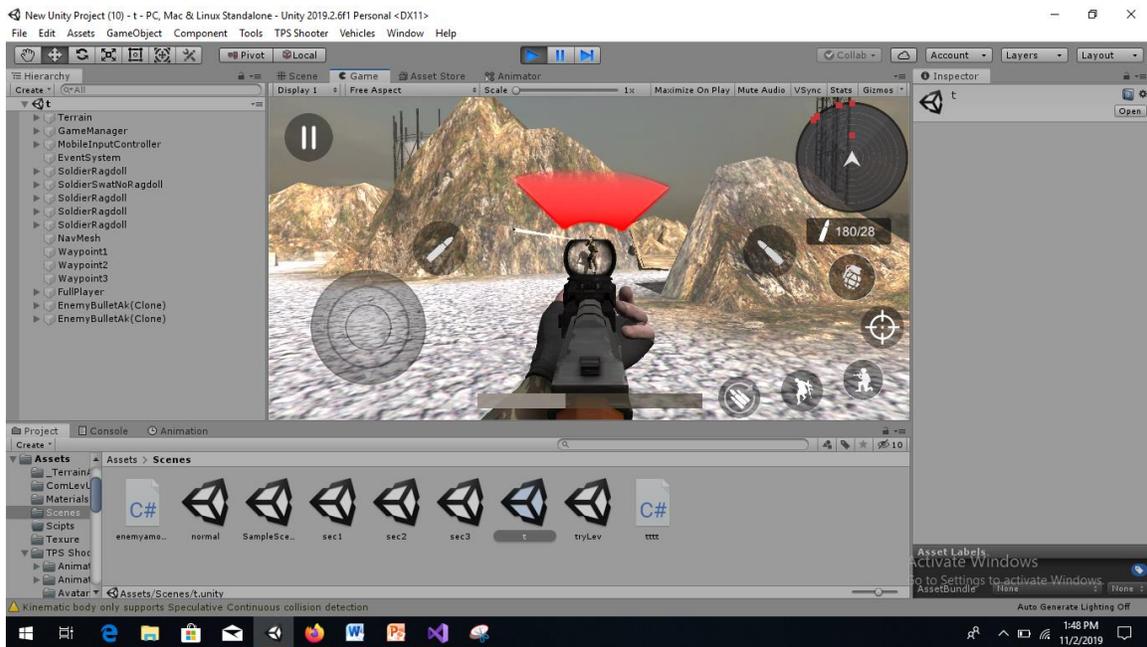


Figure 5.2.3: Player scope implementation

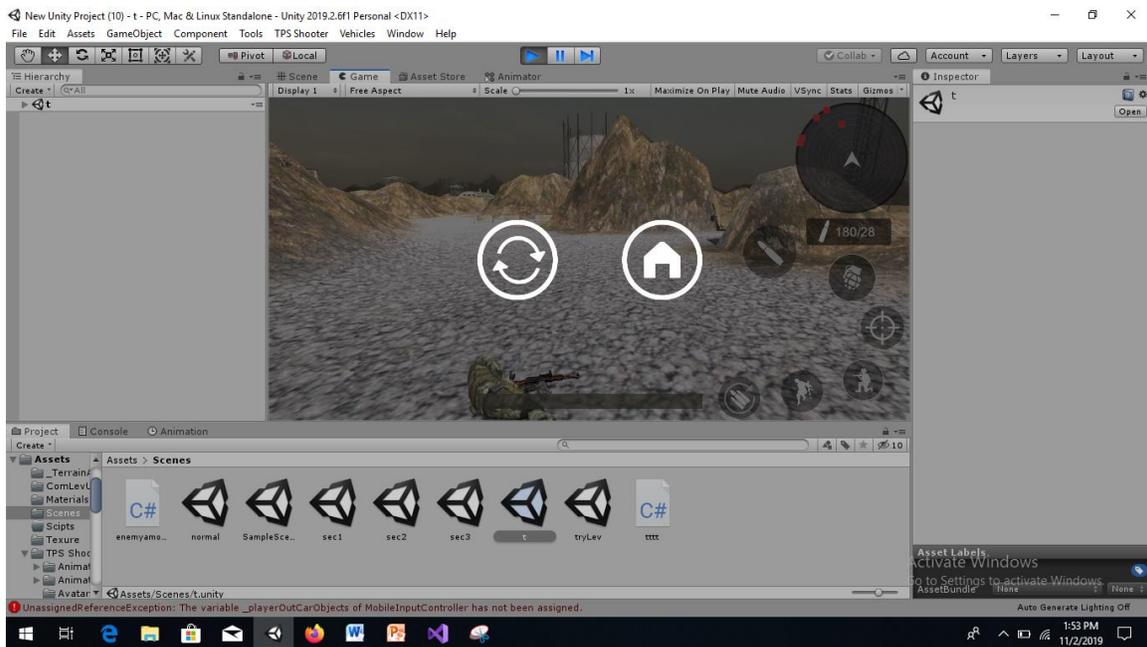


Figure 5.2.4: Player dies implementation

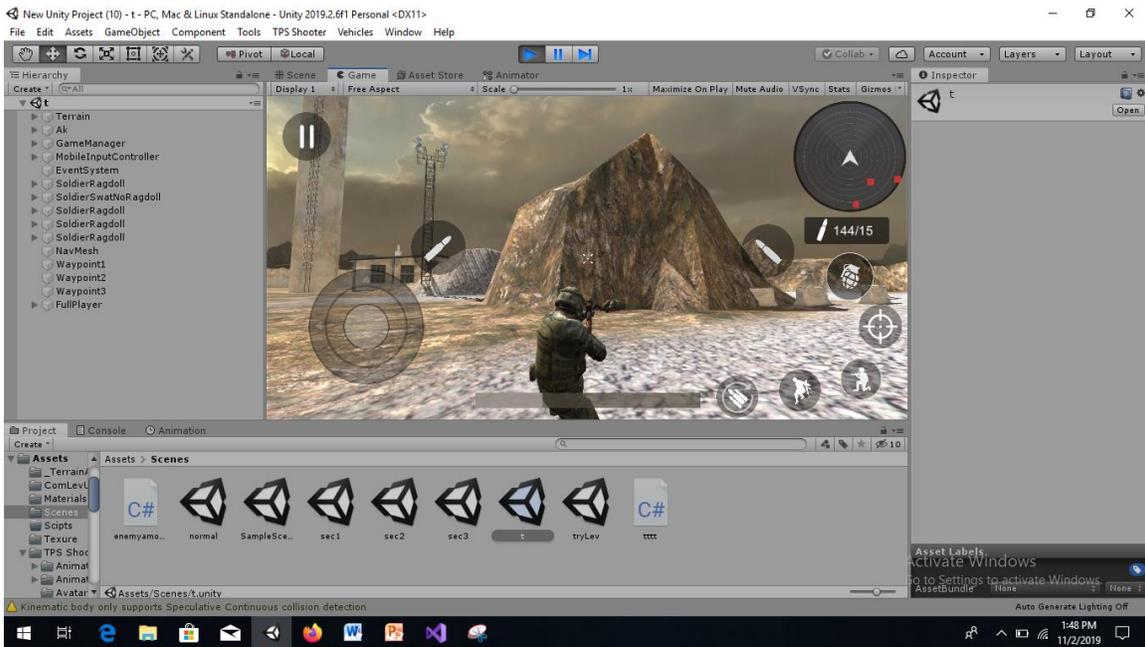


Figure 5.2.5: Enemy die implementation

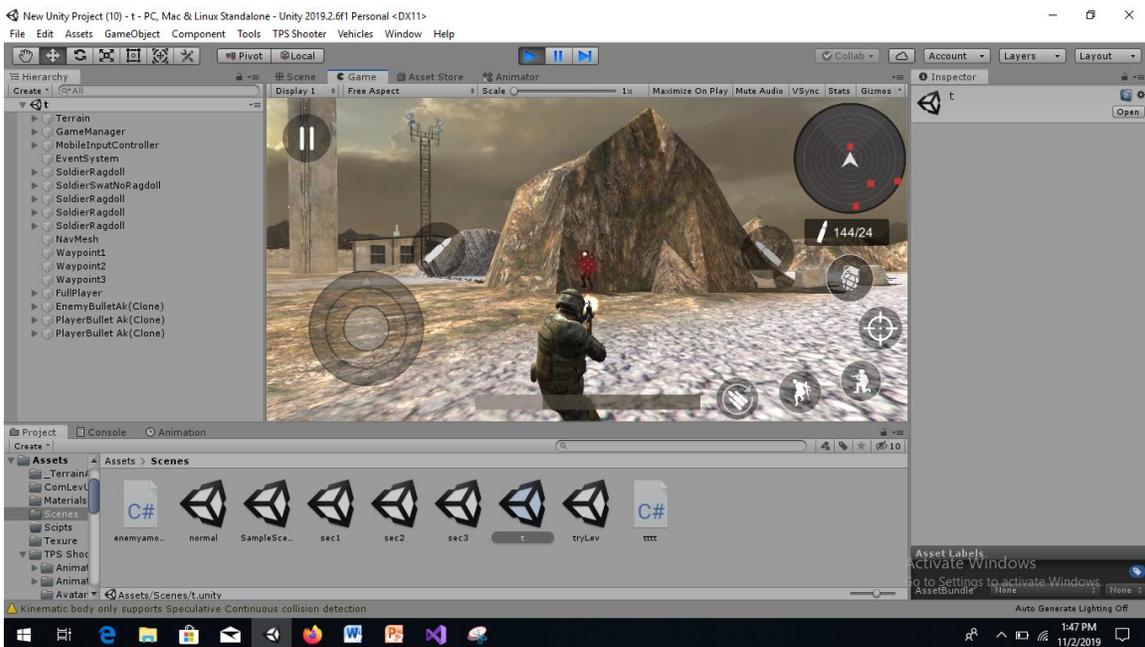


Figure 5.2.6: Player shoots bullet implementation

5.3 Implementation of Interaction

After complete, implementation of front -design, i check my games interaction. Secret commando has there interaction. Play button, setting button and exit button. This three button has also other part. When i pressed play button. It brings me to next layout weapon selection preview for playing. After select weapon button it load scene for playing game.

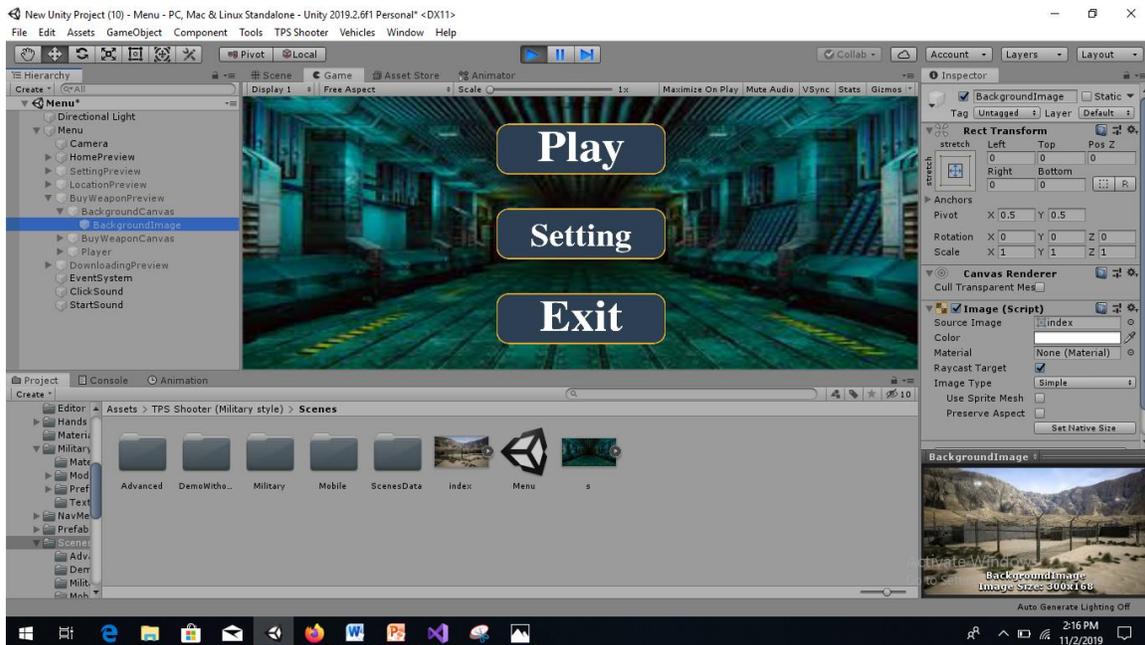


Figure 5.3.1: Interaction Implementation

In weapon button, there are five parts. There are different types of weapon. Player can select their weapon in this preview. There is a button also to go previous preview to select another level. In setting preview user can select

5.4 Testing Implementation

Implementation is a long term goal plan to achieve goals. I along with some of my friend testing this. They test GUI and games level. They check all User interface. Is all input work properly. Is player move properly. Then they check radar and radar able object which was tag with enemy. Then they check, can change one layout to another layout easily. Then they enter the final collider to cheek, without killing enemy. And next time they check same things after killing all enemies.

5.5 Test Results and Reports

After testing, all result comes to me positive. That means all user input work properly. Player controller control player properly. Radar can be identifying game object with tags enemy. They can change one layout to another layout easily. Without killing all enemy they cannot go into collider. When they kill all enemies, then they can enter this collider and it preview Congratulation message to Users.

CHAPTER 6

Conclusion and Future Scope

6.1 Discussion and Conclusion

This Report Show all activity in this project. When User plays this they can easily move Player. They can also change player look at direction by touching phone screen. User will get Player movement in two types. When Player Run Straight forward, Player can run very fast and run very straight. But at this time Player cannot target any enemies. On the other hand, when player run forward but not straight, Player cannot run very fast. Player runs slowly with targeting enemies in this time. This is main different in player movement in forward. User can run player with joystick forward, at the time player run straight and early. When player need to turn himself, user should scroll left or right on the screen. So player can run forward easy and fast.

6.2 Scope for Further Development

In the future, I will update my game. I have some plan on it. I will add in this game vehicles ability system. That means user can use vehicles in this game when they go to one place to another place. When user comes to close near vehicles around 5 meters then a drive ui will show in the screen. When user selects this he can drive this vehicles and there is another preview to drive the vehicle. At this time player will not able to do his normal activity like as shoot bullet, throw grenade, scope etc. When player drive vehicle user can move one place to another place. User only pushes enemies to kill them. Another feature I will think, that is I will add multiplayer system in this game. User can invite their friend to play with them. So they can make fun with friends though they stay at home. They can also contact each other at play time.

Reference

- [1] “Unity assets Text Mesh Pro” [Online] Available at <<<https://assetstore.unity.com/packages/essentials/beta-projects/textmesh-pro-84126>>>[Last Accessed on 5 5 2019 at 11 pm].
- [2] “Unity assets Third person Shooting” [Online] Available at <<<https://assetstore.unity.com/.../systems/third-person-controller-shooter-template-84483>>>
- [3] “Surface field texture” [Online] Available at <<<https://www.bing.com/images/search?view=detailV2&ccid=Cyuli6DA&id=1A37D75EED1ED7E9819D4126191869B7ACE88C66&thid=OIP%26s%3d612x612%26w%3d0%26h%26e&simid=607994663415647955&selectedIndex=61&ajaxhist>>>[Last access at 10 2 2019 at 12.07 am]
- [4] “Unity assets probuilder” [Online]. Available at<<<https://unity3d.com/unity/features/worldbuilding/probuilder>>>> [Last Accessed on 27 4 19 at 1.00 am].

Appendices

Appendix A: Some source code

```
using System.Collections;
using UnityEngine;
using UnityEngine.UI;

namespace TPSSHooter
{
    [RequireComponent(typeof(FootstepSounds))]
    [RequireComponent(typeof(CharacterController))]
    [RequireComponent(typeof(Animator))]
    public class PlayerBehaviour : MonoBehaviour
    {
        public AnimationParameters animationsParameters;
        public PlayerWeaponSettings weaponSettings = new PlayerWeaponSettings();
        public PlayerSounds sounds = new PlayerSounds();
        public PlayerIKSettings ikSettings = new PlayerIKSettings();
        public PlayerCrouchSettings crouchSettings = new PlayerCrouchSettings();
        public PlayerMovementSettings movementSettings = new
PlayerMovementSettings();

        private InputController _inputController;
        private CharacterController _characterController;
        private Animator _animator;

        private float _characterControllerInitialHeight;
        private Vector3 _characterControllerInitialCenter;

        public bool IsRunning { get; private set; }
        public bool IsJumping { get; private set; }
        public bool IsCrouching { get; private set; }
        public bool IsThrowingGrenade { get; private set; }
        public bool IsScoping { get; private set; }
        public bool IsDrivingVehicle { get; private set; }
        public bool IsGrounded { get; private set; }
        public bool IsAiming { get; private set; }
        public bool IsReloading { get; private set; }

        private void Update()
        {
            if (_isAlive && !IsDrivingVehicle)
            {
                UpdatePlayerActions();
            }
        }

        private void LateUpdate()
        {
            if (!IsAlive)

```

```

        return;

        IsAiming = CanAim();
        _animator.SetBool(animationsParameters.aimingBool, IsAiming);

        if (IsAiming)
            PositionSpine();
    }

private void OnAnimatorIK(int layerIndex)
{
    if (!IsAlive)
        return;
    if (IsScoping && CurrentWeaponBehaviour.ScopeSettings.UseCustomHands)
    {
        _animator.SetIKPositionWeight(AvatarIKGoal.LeftHand, 1);
        _animator.SetIKRotationWeight(AvatarIKGoal.LeftHand, 1);
        _animator.SetIKPosition(AvatarIKGoal.LeftHand,
CurrentWeaponBehaviour.ScopeSettings.LeftHandIk.position);
        _animator.SetIKRotation(AvatarIKGoal.LeftHand,
CurrentWeaponBehaviour.ScopeSettings.LeftHandIk.rotation);
    }
    else if (CurrentWeaponBehaviour && !IsReloading && !IsThrowingGrenade
&& !IsUnarmedMode && !IsChangingWeapon)
    {
        _animator.SetIKPositionWeight(AvatarIKGoal.LeftHand, 1);
        _animator.SetIKRotationWeight(AvatarIKGoal.LeftHand, 1);
        _animator.SetIKPosition(AvatarIKGoal.LeftHand,
CurrentWeaponBehaviour.LeftHandIk.position);
        _animator.SetIKRotation(AvatarIKGoal.LeftHand,
CurrentWeaponBehaviour.LeftHandIk.rotation);
    }
    else
    {
        _animator.SetIKPositionWeight(AvatarIKGoal.LeftHand, 0);
        _animator.SetIKRotationWeight(AvatarIKGoal.LeftHand, 0);
    }
}

#endregion

private void UpdatePlayerActions()
{
    GroundedAction();
    JumpAction();

    AirControlAction();

    WalkAction();
    RunAction();

    CrouchAction();

    ApplyGravity();

//weaoon

```

```

        ShootAction();
        ReloadAction();
        ScopeAction();
        DropWeaponAction();
        ChangeWeaponAction();

        GrenadeAction();
    }

    private void GroundedAction()
    {
        IsGrounded = CheckGrounded();
        _animator.SetBool(animationsParameters.groundedBool, IsGrounded);
    }

    #region Walk_Run

    private float _forward;
    private float _strafe;
    private readonly float _movementLerpSpeed = 15f;

    private void RunAction()
    {
        if (_inputController.IsRun && ((!IsUnarmedMode && _forward > 0.3f) ||
(IsUnarmedMode))&& !IsReloading && !IsThrowingGrenade &&
!_inputController.IsFirePressed && !IsJumping && !IsCrouching && !IsScoping)
            IsRunning = true;
        else
            IsRunning = false;

        _animator.SetBool(animationsParameters.runBool, IsRunning);
    }

private void ShootAction()
    {
        if (_inputController.IsFirePressed && CurrentWeaponBehaviour &&
!IsThrowingGrenade && CurrentWeaponBehaviour.CanShoot &&
CurrentWeaponBehaviour.BulletsInMag > 0)
            Shoot();
    }

    private void Shoot()
    {
        if (_inputController.PositionWhereToFire == Vector3.zero)
            CurrentWeaponBehaviour.Fire(IkSettings.LookAt.position);
        else
            CurrentWeaponBehaviour.Fire(_inputController.PositionWhereToFire);

        UpdateBulletsCountText();
    }
}

```

Secret Comando

ORIGINALITY REPORT

1%

SIMILARITY INDEX

Freedom
0.15-12-19
0%

INTERNET SOURCES

0%

PUBLICATIONS

1%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to Daffodil International University

Student Paper

1%

2

Submitted to Heriot-Watt University

Student Paper

<1%

3

Submitted to Informatics Education Limited

Student Paper

<1%

Exclude quotes

Exclude matches

Exclude bibliography