

SHIBO JI

COSC829-FALL 2020

MULTI-USER ACTIVE SHOOTER RESPONSE

OVERVIEW

- Project Introduction
- Implementation:
 - 3D modeling
 - Audio
 - Animation
 - Sensors
 - User interactive
 - AI
 - Multi-User
- Conclusion

PROJECT INTRODUCTION

This project aims to design a simulation game of school shooting incident to strengthen students' understanding and prevention of campus attack incidents through various behaviors of players in the game.

The simulation game designed in this project builds a 3D campus model through Unity.

MODELING

- 3D buildings from Unity Asset Store



MODELING

Avatars: Students, teachers, policemen, enemy.



Teacher



Police



Enemy



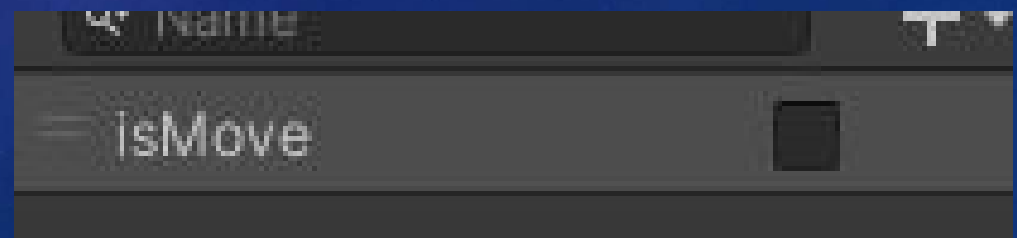
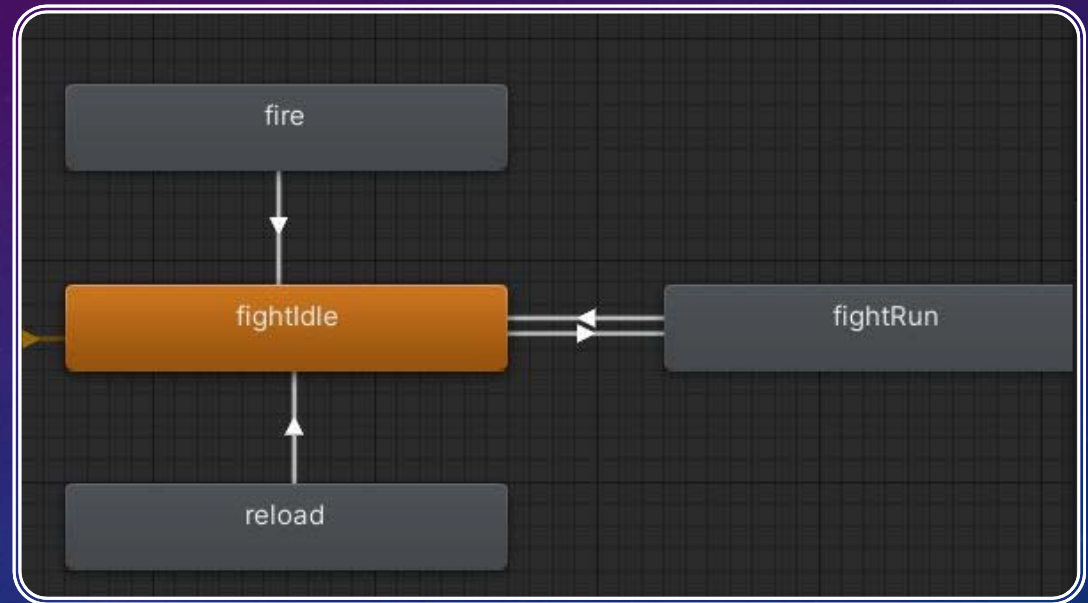
Student

AUDIO

- Sounds used to enhance experience.
 - Clop-clop Sound
 - Shoot Sound
 - Engine Sound
 - Alert Sound

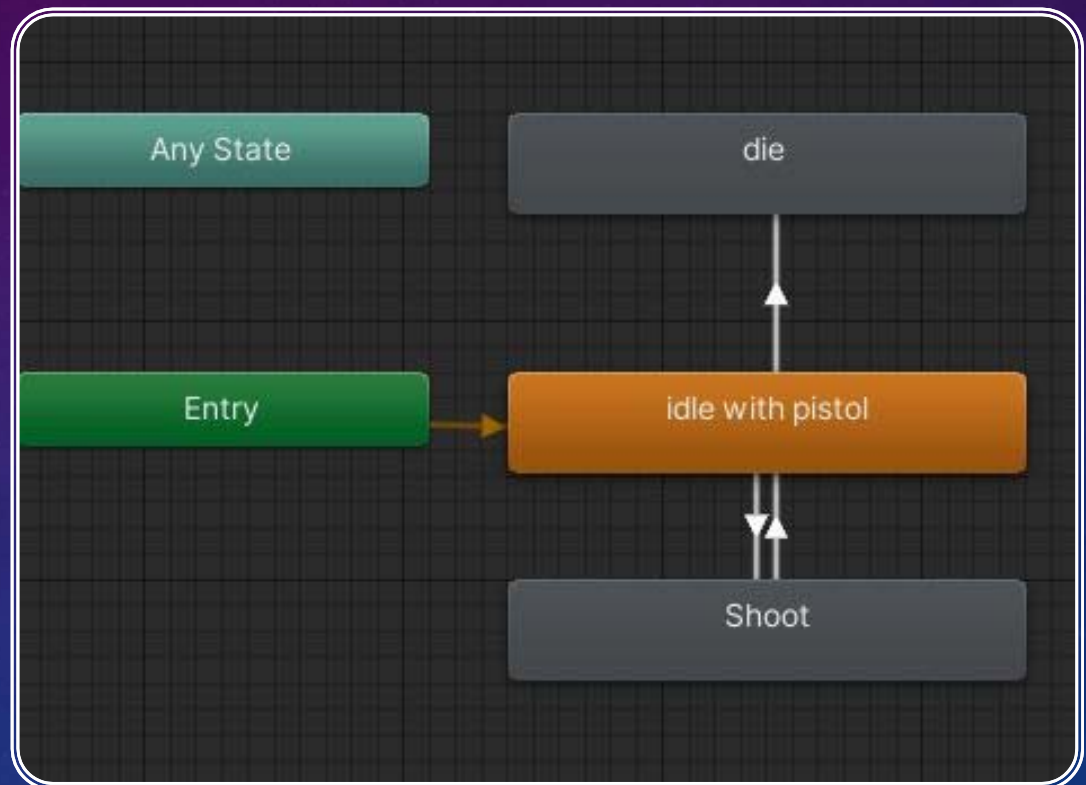
ANIMATION

- Character Animation: In this part, it include the avatar walk, fire(shoot) and jump.
- From the figure, we can know that, 'idle' is the standby animation, 'fire' is the shoot animation, 'fightrun' is the running animation, and 'reload' is reloading bullet animation.



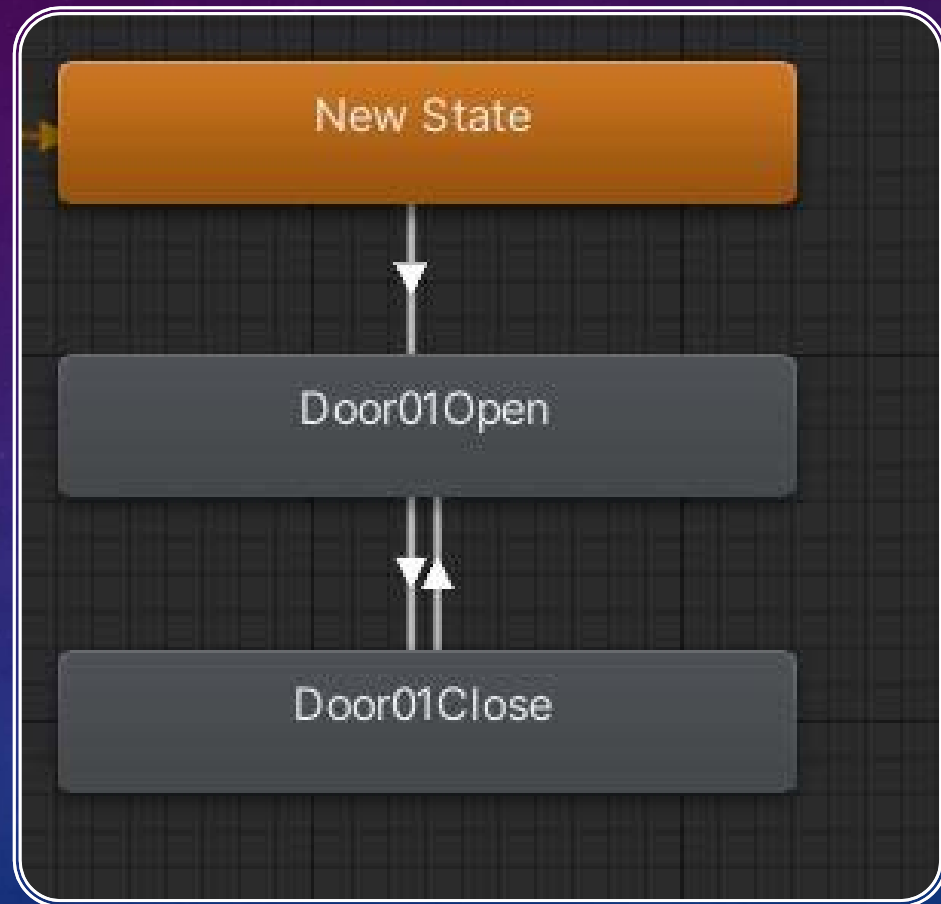
ANIMATION

- Enemy Animation:
- For the enemy animation, it is also use animator to control. The animation include standby animation 'idel', the death animation 'die', and fire animation 'shoot'.



ANIMATION

- The default status for the door is closed. I use a bool variable 'isOpen', if it is true it will open the door, else it will close the door. I also use a sensor for the door open/close. I will talk it later in the sensor part.



SENSORS

The first sensor is the sensor. When the user press 'K' on the keyboard, the event will be triggered. The time will cut down. I use a 'timer', let 'timer -= Time.deltaTime'. We can see the figure is a part of code how to do the countdown timer.

```
if (isEventStart && timer>0 && isFinished == false)
{
    timer -= Time.deltaTime;
    currentTimerText.text = "CountDown:" + ((int)timer).ToString()+"S";
}
```

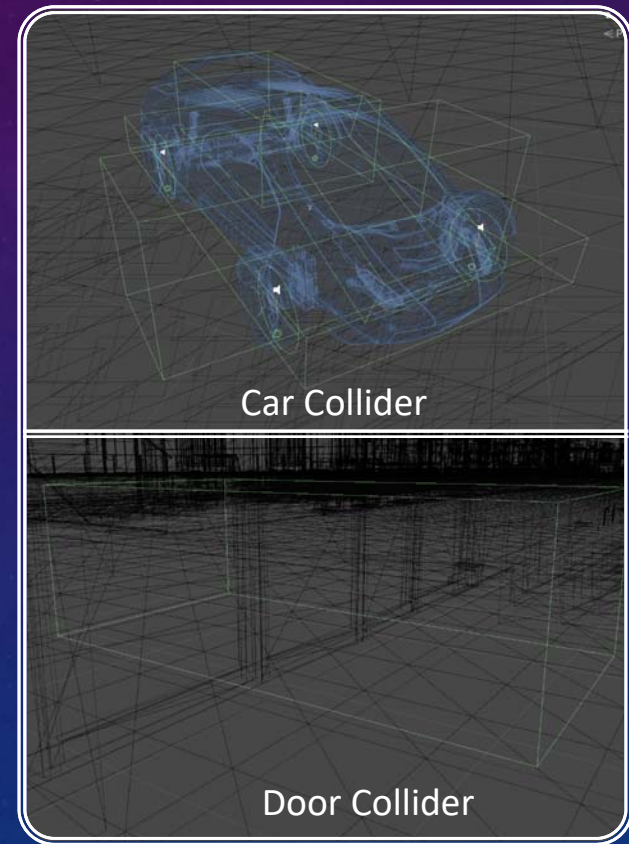
SENSORS

- I put a collider on the enemy object, it is used to detect the users. From figure we can see that , if the user is in the detect range, the enemy will shoot the user.



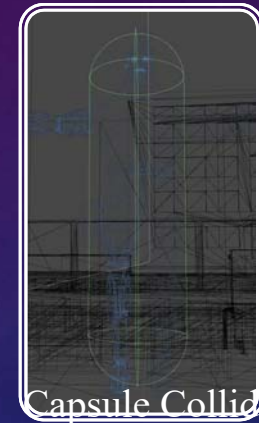
SENSOR

- For the car, I also use collider to detect if any users are near the car, it detect the user, the user can use 'K' to drive the car. I add a box collider for the car.
- I add a trigger for the doors. It also a box collider. When the user is in the range, it will open the door and close the door when they are out of the range.



SENSOR

- I add trigger for other avatars in the game, it is used to detect if the user is in the detect area and then they can have a conversation. I use capsule collider and sphere for the teacher and student. I use Sphere Collider for the police.



Capsule Collider



Sphere Collider

USER INTERACTIVE

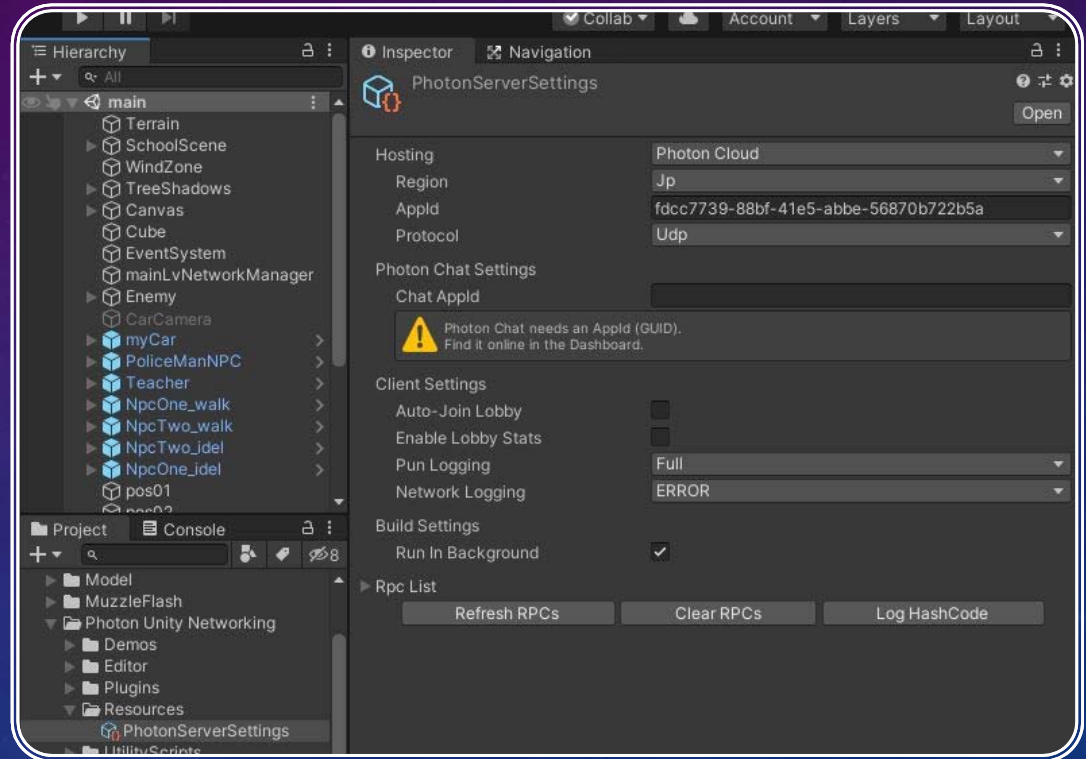
- 1) Talk with NPC in the game. The user can talk with the student, police and teacher in the game. In this game I use collider to detect if the user is near the NPC, if detect the user, and it will notice the user'Use F to talk with the NPC'.
- 2) Open the door. The second interactive is to open the door. I set up a trigger to detect the user. If detect the user it will notice the user" Press F to open the door".
- 3) Drive the car is the third interactive. I also use a trigger for the car module. If detect the user, it will notice "press k to drive the car". When the user is in the car, it will disable the camera and use the carcamera to follow the car.
- 4) Trigger the event. In this game when you press K the event will be triggered.

AI

- 1) When the event is triggered, all the NPC will go to the gym automatically. The Police will stand at the door to protect the people.
- 2) The enemy can detect the user and attack the user automatic.

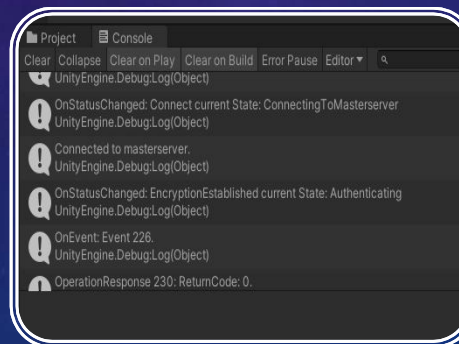
MULTI-USER

- For this game I use Photon Unity Network (PUN) to realize multi user environment.
- We need to connect the server. You should have the AppId you apply from the PUN website,



- Then you need to test if you connect the server successfully. You can use the code in figure to connect the server. If you succeed you will see “connected to masterserver”

MULTI-USER



```
Project Console
Clear Collapse Clear on Play Clear on Build Error Pause Editor
UnityEngine.Debug.Log(Object)
! OnStatusChanged: Connect current State: ConnectingToMasterserver
UnityEngine.Debug.Log(Object)
! Connected to masterserver.
UnityEngine.Debug.Log(Object)
! OnStatusChanged: EncryptionEstablished current State: Authenticating
UnityEngine.Debug.Log(Object)
! OnEvent: Event 226.
UnityEngine.Debug.Log(Object)
! OperationResponse 230: ReturnCode: 0.
```

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

```
void Start()
{
    PhotonNetwork.ConnectUsingSettings("0.0.1");
}
```

MULTI-USER

- For PUN it use component 'PhotonView' to synced the variables.
- Add component 'Photon TransformView'
- Add the component 'PhotonAnimatorView'



MULTI-USER

- How the server sent information to the client?

```
[PunRPC]
public void OnEventStart ()
{
    isEventStart = true;
    tipText.text = "The enemy appeared! ";
    dangerAudio.Play ();

    enemmyGo01.SetActive (true);
    enemmyGo02.SetActive (true);
    enemmyGo03.SetActive (true);
    enemmyGo04.SetActive (true);
}
```

```
PhotonView pv = GetComponent<PhotonView> ();
pv.RPC ("OnEventStart", PhotonTargets.AllBuffered);
```

CONCLUSION

- For this game, I was to do a campus active shoot response. The users are in the campus and when the event was triggered, all the avatars in the game will go to a safe space(the gym) to take refuge. The police will stand by the door to protect the people in the gym.