SIMON FRASER UNIVERSITY

Unity

Himanshu Arora, Ali Mahdavi Amiri

November 27, 2021

1 FIRST PERSON SHOOTER IN UNITY (15 + 5 POINTS)

You are to implement a simple first person shooter game in Unity. You are given a project, which has a player character that can shoot, reload the gun and move around the environment.
The player character moves using the *WASD* keys on your keyboard. It shoots the gun by pressing the left mouse button and reloads the magazine with the *R* key. You can rotate the character by rotating your mouse. There are some missing parts from the player codes, which we will complete in together class. The missing parts are completing the reload animation, putting bullet holes on the walls, instantiating the muzzle flash, and shoot sounds when firing the gun. You will lose points if these missing parts are not implemented in your final output. All of the player and the enemy actions should be implemented with animations.

1.1 IMPLEMENTATION

1.1.1 ENEMY CHARACTER (3 POINTS)

In this part, you are asked to implement the enemy character, which walks in a predefined path in the game world. Each character should have the ability to have a different path from the other enemies. The target positions are passed as an array to the enemy controller script (see *targets* node in framework as example). Note that when the enemy character reaches the last target, it should go back to the first target and start its path again from the first target point. Enemy characters should have a red uniform instead of blue ones. You should make a new material and use the texture provided *Soldier_Body_diffuse_red.png* in *shooter pack/swat.fbm* folder.

1.1.2 Detecting player (3 Points)

While the enemy walks in the path, it can detect the player character. This happens when the player moves into the enemy field of view. When the enemy detects the player, it should run towards the player until it reaches the distance of 10 meters from it. The enemy should also maintain the distance when the player moves away from the enemy.

1.1.3 Enemy shooting the player (3 Points)

When the enemy detects the player and reaches the distance of 10 meters to the player character, it starts shooting at the player. The enemy gun shooting should be implemented with *Physics.Raycast*. The enemy can shoot at most 5 bullets per second. You should also randomize the shooting vector, so that 20% of the times enemy bullets hit the player. When the player health becomes zero or less, you should run the death animation and move the camera to a fixed position.

1.1.4 Player shooting the enemy and health (3 Points)

For this part, you should complete the player's scripts. The player can shoot and kill the enemies and also loose health when the enemy shoots him. You should also add the player's health to the UI Canvas in the project. When the player kills the enemy, you should add a *rigidbody* and *collider* to the gun and make it an independent object from the dead enemy. Moreover, the enemy will detect the player when the player shoots him.

1.1.5 Game environment (3 Points)

The game environment should have multiple rooms and enemies. Each enemy should have a unique set of path targets. You should make an escape door in the last room. When the player reaches that position, it will win the game. The escape door should be implemented with a *trigger collider*. Also, when the game finishes, whether by escaping the last room or the player getting killed, the game should restart after 10 seconds.

1.2 Bonus

1.2.1 Ammunition supply (1 Point)

Create ammo crates, where the player can refill its ammunition by reaching it.

1.2.2 Enemies getting cover (1 Point)

When the enemy detects the player, instead of running towards the player, it goes to the nearest cover. The enemy should sit behind the cover and only uncover shortly while shooting and hiding again immediately.

1.2.3 Detecting body part hits (1 Point)

Instead of reducing the health by 20% for each hit, each body part has its own importance. Hitting the head would kill the player or enemy instantly. Hitting hands, chest and legs will reduce the health by 10, 30 and 20 percent, respectively.

1.2.4 Swapping guns (2 Points)

By pushing the *Q* key (or any other key of your choosing), the player will swap the primary weapon with another weapon. By pushing it again it will swap the secondary weapon with the primary one. For this part, there are no animations provided, so you should create or find the required animations yourself.