COMSATS University Islamabad, Lahore Campus

CSC-495 Game Development

Lab 11-12

# Lab Manual: Creating a Unity Project with Night Terrain, FPS Controller, and Animated Character

# Learning Outcomes:

Upon completing this lab, students will be able to:

1. Create a new Unity project with a night terrain.

2. Implement an FPS controller with an MP7 gun.

3. Create a character with an animation controller that can idle, walk, and die.

4. Design and implement animation controllers for character movements.

# Lab Steps:

Step 1: Create a New Unity Project

1. Open Unity Hub and create a new project.

2. Choose the "3D" game template.

3. Name the project "Your Reg number\_FPS".

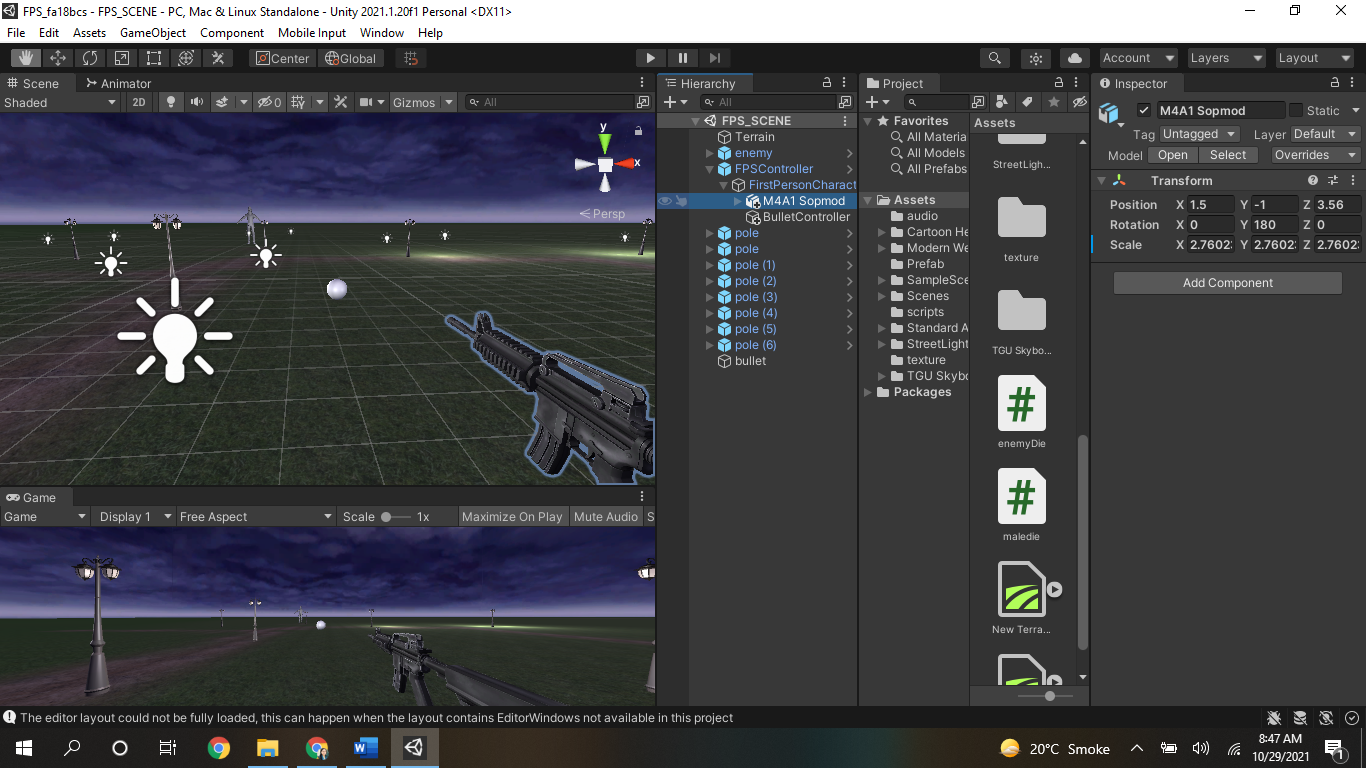
Step 2: Create a Night Terrain

1. Create a new terrain by going to "GameObject" > "3D Object" > "Terrain".

2. Adjust the terrain settings to create a night environment (e.g., dark colors, low lighting).

3. Add trees, rocks, and other environmental objects to enhance the terrain.

4. Now add a bullet in the scene, and instantiate it in a script added to BulletController (Empty Object).



1. Add a Machine to the FPS Character as its child. (Machine gun link: <https://assetstore.unity.com/packages/3d/props/guns/modern-weapons-pack-14233>)
2. Make sure BulletController is the child of FPS Character and position should be set from where you want to fire (instantiate) the bullet. For example give it same x, y, z position of the gun and move forward its z position a bit.
3. Add BulletLauncher.cs script to the BulletController where bullet should be passed as gameObject below the script in BulletController inspector.

Step 3: Create a Character with Animation Controller

1. Import a character model from a 3D modeling software or asset store.

2. Create a new animation controller by going to "Window" > "Animation" > "Animator Controller".

3. Design the animation controller to include the following states:

- Idle

- Walk

- Die

4. Create animations for each state using the "Animation" window as mentioned below:

1. **Create an Animator Controller**
   1. In Unity, go to Assets → Create → Animator Controller
   2. Name it EnemyAnimator
2. **Attach the Animator to the Enemy**
   1. Select your enemy GameObject.
   2. Add an Animator component (if not already added).
   3. Assign the EnemyAnimator controller to it.
3. **Define Animation States**
   1. In the Animator window (Window → Animation → Animator), add the following states:
   2. **Idle** → When the enemy is not moving.
   3. **Walk** → When the enemy moves toward the player.
   4. **Attack** → When the enemy is close enough to attack.
   5. **Die** → When the enemy is hit by a bullet and dies.
4. **Add Parameters**
   1. Go to the **Parameters** tab in the Animator window and add:
   2. isWalking (Bool) → true when enemy is walking.
   3. isAttacking (Bool) → true when enemy is attacking.
   4. isDead (Trigger) → Triggered when the enemy is hit and dies.
5. **Set Up Transitions**
6. **Idle → Walk**

Condition: isWalking == true

1. **Walk → Attack**

Condition: isAttacking == true

1. **Attack → Walk (Optional)**

Condition: isAttacking == false

1. **Any State → Die**

Condition: isDead (Trigger)

1. Make sure to **uncheck** "Has Exit Time" for transitions, so they happen instantly when conditions are met.

5. Attach the animation controller to the character model.

Step 5: Implement Character Movement and Death Animation

1. Write a script to control the character's movement using the animation controller.

2. Use Unity's built-in physics engine to detect collisions between the character and bullets.

3. When a bullet collides with the character, trigger the "Die" animation state.

Step 6: Test and Refine the Project

1. Test the project to ensure that all components are working correctly.

2. Refine the project by adjusting settings, animations, and scripts as needed.

# Assessment:

- Students will be assessed on the completion of the lab steps.

- The assessment will include:

- A working Unity project with a night terrain, FPS controller, and animated character.

- A well-designed animation controller with idle, walk, and die states.

- A script that controls character movement and death animation.

# Conclusion:

In this lab, students learned how to create a Unity project with a night terrain, FPS controller, and animated character. Students designed and implemented an animation controller with idle, walk, and die states, and wrote a script to control character movement and death animation. This lab provides a comprehensive understanding of Unity's animation and physics systems, and prepares students for more advanced game development projects.

Use following script:

BulletController.cs:

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class BulletLanucher : MonoBehaviour

{

public GameObject bullet;

// Start is called before the first frame update

void Start()

{

}

// Update is called once per frame

void Update()

{

if(Input.GetMouseButtonDown(0))

{

Instantiate(bullet, transform.position, transform.rotation);

}

}

}

BulletScript.cs:

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class bulletScript : MonoBehaviour

{

// Start is called before the first frame update

void Start()

{

}

// Update is called once per frame

void Update()

{

//transform.Translate(0, 0, 1);

Rigidbody r = GetComponent<Rigidbody>();

r.AddForce(transform.forward \*100);

}

private void OnCollisionEnter(Collision collision)

{

print("bullet collides");

Destroy(transform.gameObject);

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_END\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_