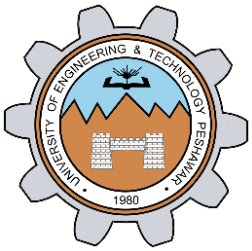
Unity Animations and Animator

LAB # 9



Fall 2024

CSE-411L Intro to Game Development Lab

Submitted by: **Ali Asghar**

Registration No.: **21PWCSE2059**

Class Section: **A**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Submitted to:

Engr. Abdullah Hamid

Date:

25th December 2024

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Objective:

In this lab we further explored the Unity API.

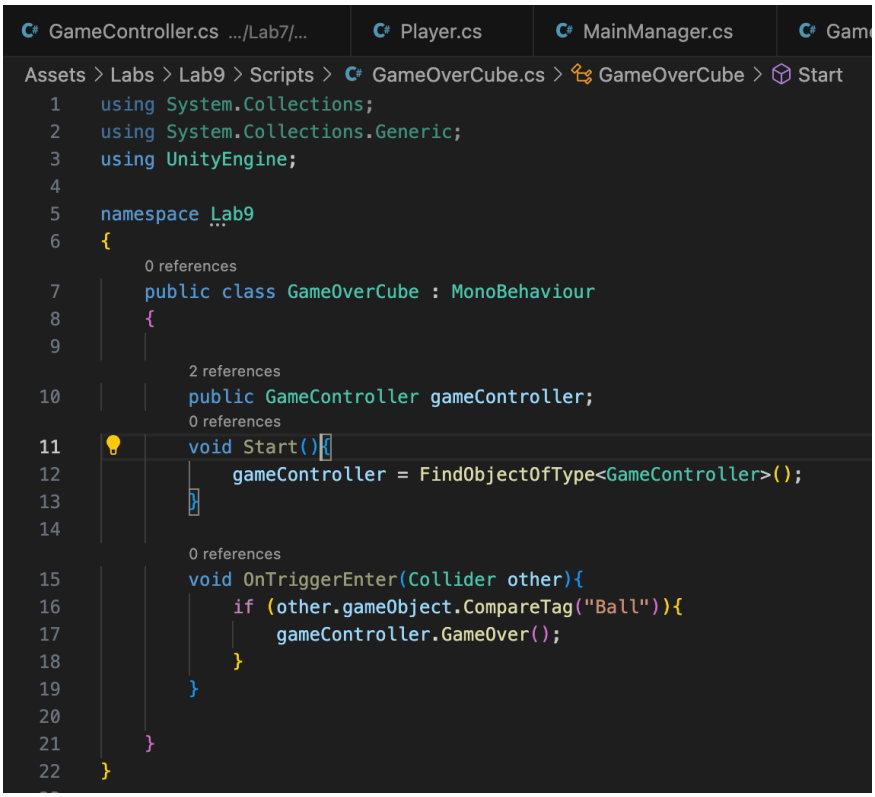
Tasks:

1) **Set Up the Scene**:
    a) Open or create a new scene in Unity.
    b) In this scene, create your game layout and add a main panel with a “Play” button.
2) **Pause the Game on Start**:
    a) Ensure the game is paused when it starts.
    b) When the “Play” button is clicked:
    c) Animate the main panel to slide up and out of the main camera’s view.
    d) Resume the game.
3) **Download Assets**:
    a) Go to Mixamo and download a character model along with basic animations like **walk** and **idle**.
4) **Convert to Humanoid**:
    a) Convert the character and animations to **Humanoid** rig in Unity.
    b) (See conversion method on the next page).
5) **Set Up Animator Controller**:
    a) Create a new **Animator Controller** for the character.
    b) This character will act as the main player in your game.
6) **Idle and Walk Animation**:
    a) Set the default animation state of the player to **Idle**.
    b) Configure the animator so that pressing W, A, S, or D will:
    c) Trigger the **Walk** animation.
    d) Move the player in the corresponding direction.
7) **Design the Game Scene**:
    a) Add a **Plane** as the ground.
    b) Place the player and a **Ball** on the plane.
    c) Apply a **Rigidbody** and **Bouncy Physics Material** to the ball.
    d) Ensure the player can push the ball towards a goal on the plane.
8) **Goal Mechanic**:
    a) Add a **Goal** object to the scene.
    b) The goal should have a **color-changing animation**.
    c) When the ball reaches the goal:
    d) Display the UI message: **“Game Complete!”**
    e) End the game.
9) **Game Over Mechanic**:
    a) Add an **Invisible Collider** underneath the plane.
    b) If the ball falls off the plane and collides with this object:
    c) Display the UI message: **“Game Over, you lose!”**
    d) End the game.

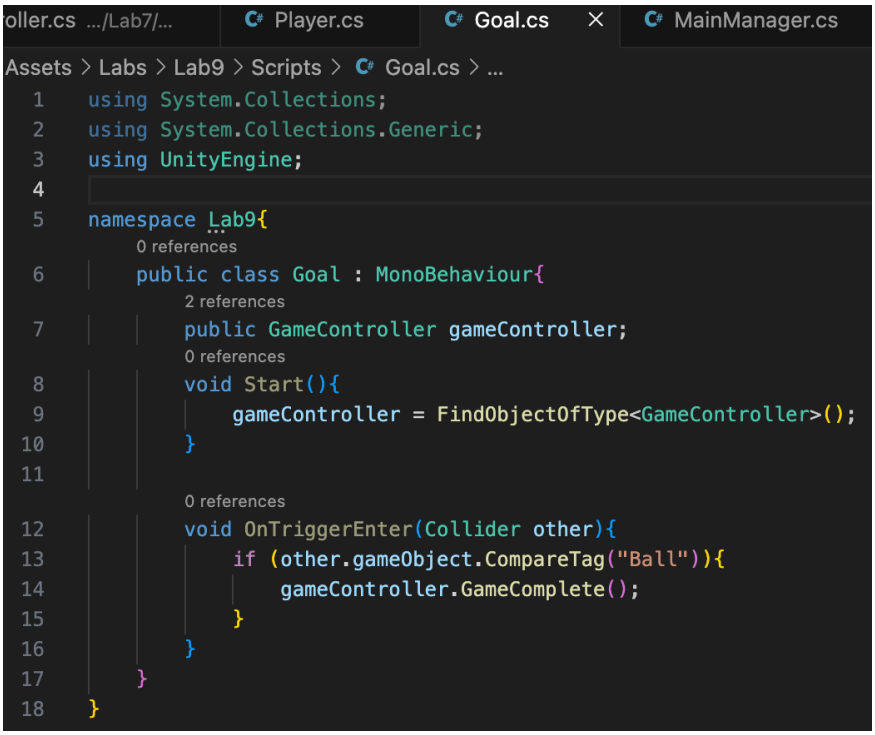
Code:

GameController class

GameOverCube class



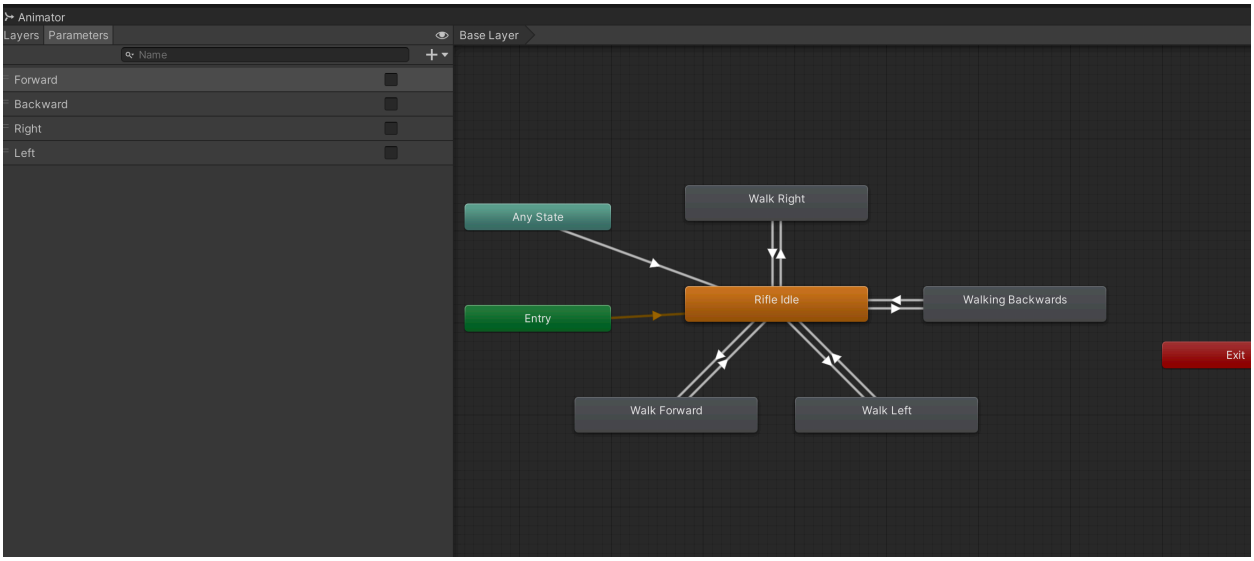
Goal class



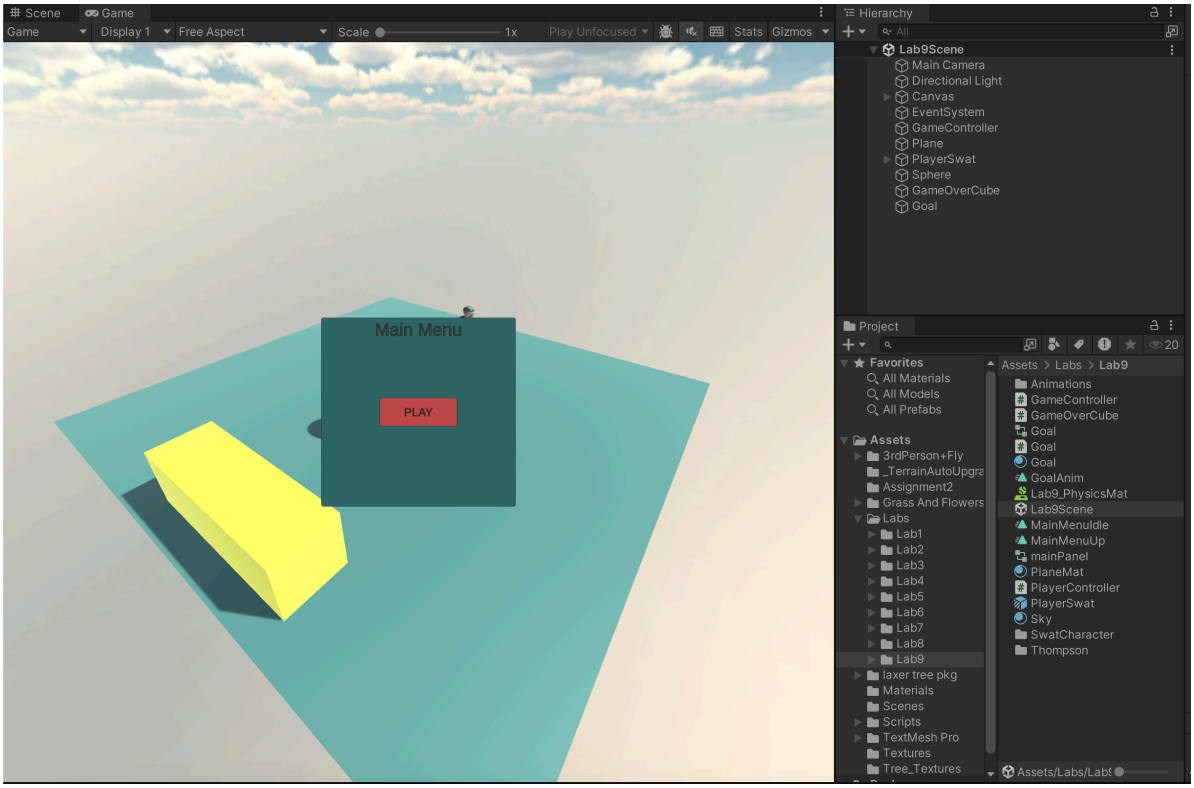
PlayerController class

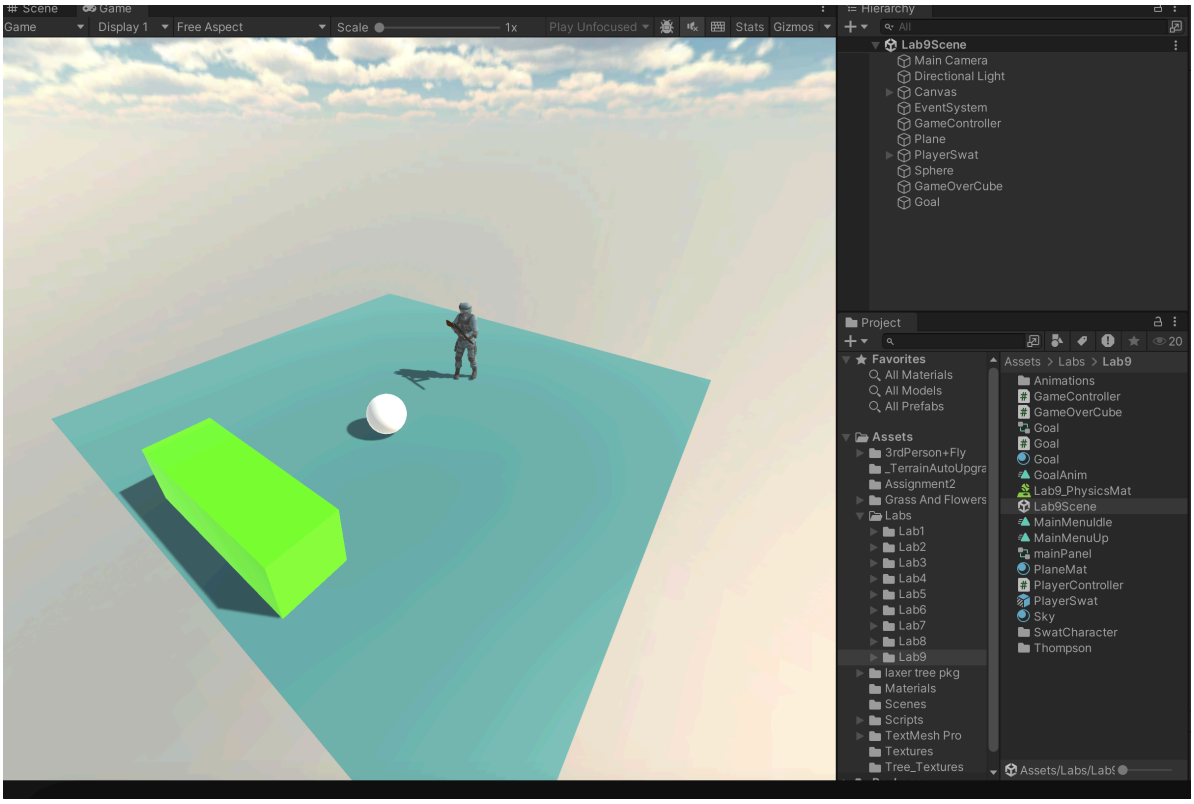
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

namespace Lab9
{

0 references
public class PlayerController : MonoBehaviour
{
    [SerializeField]
    10 references
    private Animator playerAnimator;

    [SerializeField]
    4 references
    private float speed = 1f;
    // Start is called before the first frame update
    0 references
    void Start()
    {
        if (playerAnimator == null){
            playerAnimator = GetComponent<Animator>();
        }

    }

    // Update is called once per frame
    0 references
    void Update()
    {
        playerAnimator.SetBool("Forward", false);
        playerAnimator.SetBool("Backward", false);
        playerAnimator.SetBool("Left", false);
        playerAnimator.SetBool("Right", false);

        if (Input.GetKey(KeyCode.W)){
            transform.Translate(Vector3.forward * Time.deltaTime * speed);
            playerAnimator.SetBool("Forward", true);
        }
        if (Input.GetKey(KeyCode.S)){
            transform.Translate(Vector3.back * Time.deltaTime * speed);
            playerAnimator.SetBool("Backward", true);
        }
        if (Input.GetKey(KeyCode.A)){
            transform.Translate(Vector3.left * Time.deltaTime * speed);
            playerAnimator.SetBool("Left", true);
        }
        if (Input.GetKey(KeyCode.D)){
            transform.Translate(Vector3.right * Time.deltaTime * speed);
            playerAnimator.SetBool("Right", true);
        }
    }
}
}
```

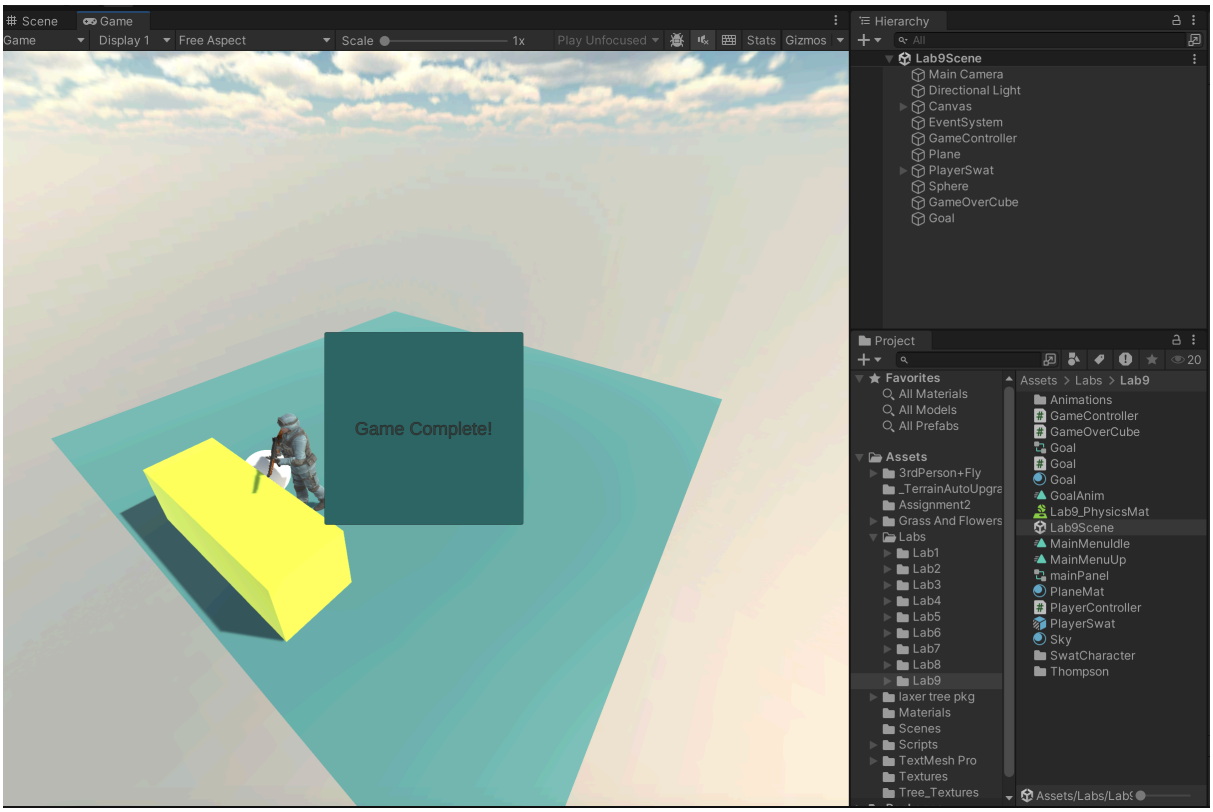
Animator FSM:

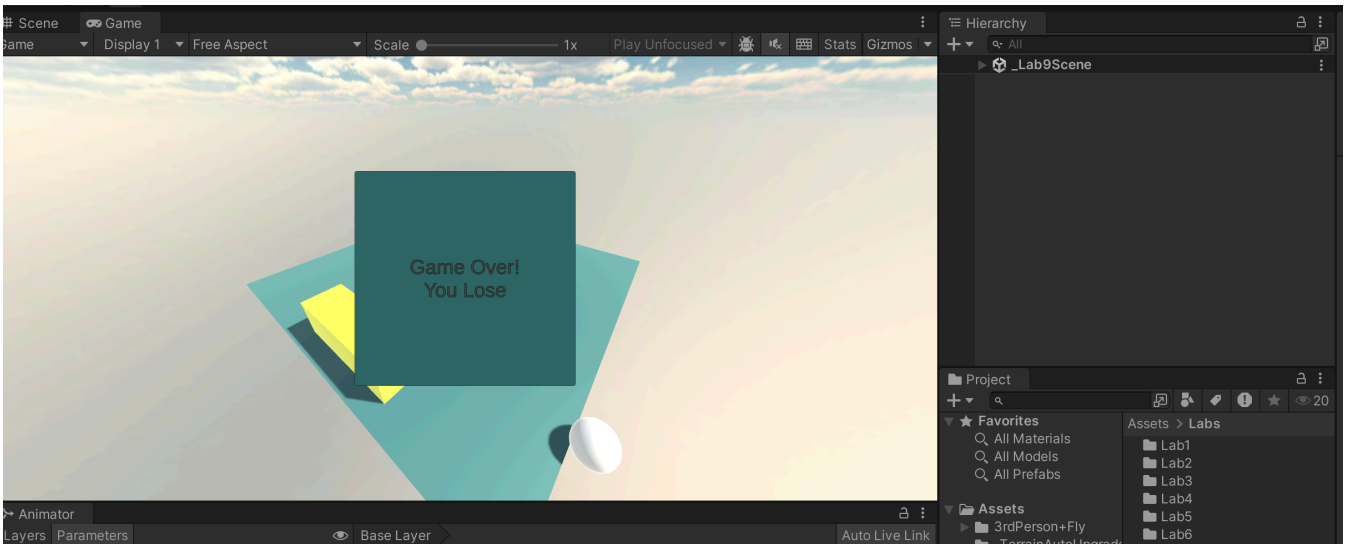


Output:




Scene
Game
Game
Display 1
Free Aspect
Scale
1x
Play Unfocused
Stats
Gizmos
Hierarchy
All
Lab9Scene
Main Camera
Directional Light
Canvas
EventSystem
GameController
Plane
PlayerSwat
Sphere
GameOverCube
Goal
Project
20
Favorites
All Materials
All Models
All Prefabs
Assets
3rdPerson+Fly
_TerrainAutoUpgra
Assignment2
Grass And Flowers
Labs
Lab1
Lab2
Lab3
Lab4
Lab5
Lab6
Lab7
Lab8
Lab9
laxer tree pkg
Materials
Scenes
Scripts
TextMesh Pro
Textures
Tree_Textures
Assets > Labs > Lab9
Animations
GameController
GameOverCube
Goal
Goal
Goal
GoalAnim
Lab9_PhysicsMat
Lab9Scene
MainMenuIdle
MainMenuUp
mainPanel
PlaneMat
PlayerController
PlayerSwat
Sky
SwatCharacter
Thompson
Assets/Labs/Lab


Scene
Game
Game
Display 1
Free Aspect
Scale
1x
Play Unfocused
Stats
Gizmos
Game Complete!
Hierarchy
All
Lab9Scene
Main Camera
Directional Light
Canvas
EventSystem
GameController
Plane
PlayerSwat
Sphere
GameOverCube
Goal
Project
20
Favorites
All Materials
All Models
All Prefabs
Assets
3rdPerson+Fly
_TerrainAutoUpgra
Assignment2
Grass And Flowers
Labs
Lab1
Lab2
Lab3
Lab4
Lab5
Lab6
Lab7
Lab8
Lab9
laxer tree pkg
Materials
Scenes
Scripts
TextMesh Pro
Textures
Tree_Textures
Assets > Labs > Lab9
Animations
GameController
GameOverCube
Goal
Goal
Goal
GoalAnim
Lab9_PhysicsMat
Lab9Scene
MainMenuIdle
MainMenuUp
mainPanel
PlaneMat
PlayerController
PlayerSwat
Sky
SwatCharacter
Thompson
Assets/Labs/Lab


Scene
Game
Display 1
Free Aspect
Scale
1x
Play Unfocused
Stats
Gizmos
Game Over!
You Lose
Animator
Layers
Parameters
Base Layer
Auto Live Link
Hierarchy
All
_Lab9Scene
Project
Favorites
All Materials
All Models
All Prefabs
Assets
3rdPerson+Fly
Assets > Labs
Lab1
Lab2
Lab3
Lab4
Lab5
Lab6