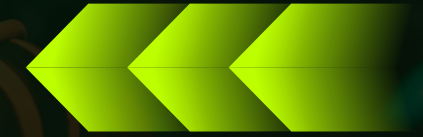




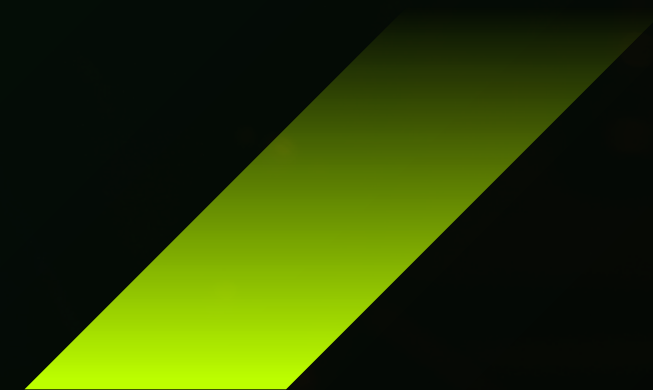
# MINOR PROJECT

## TENNIS GAME



- BY: ANKIT TIRKEY
- ROLL NO: 20222714
- GITHUB LINK:

GITHUB REPOSITORY





# INTRODUCTION



- "A Unity-based tennis simulation with a player-controlled character and AI-driven bot, featuring realistic physics and animations."

## Key elements:

- Player and AI interactions.
- Real-time ball physics.
- Seamless animations.







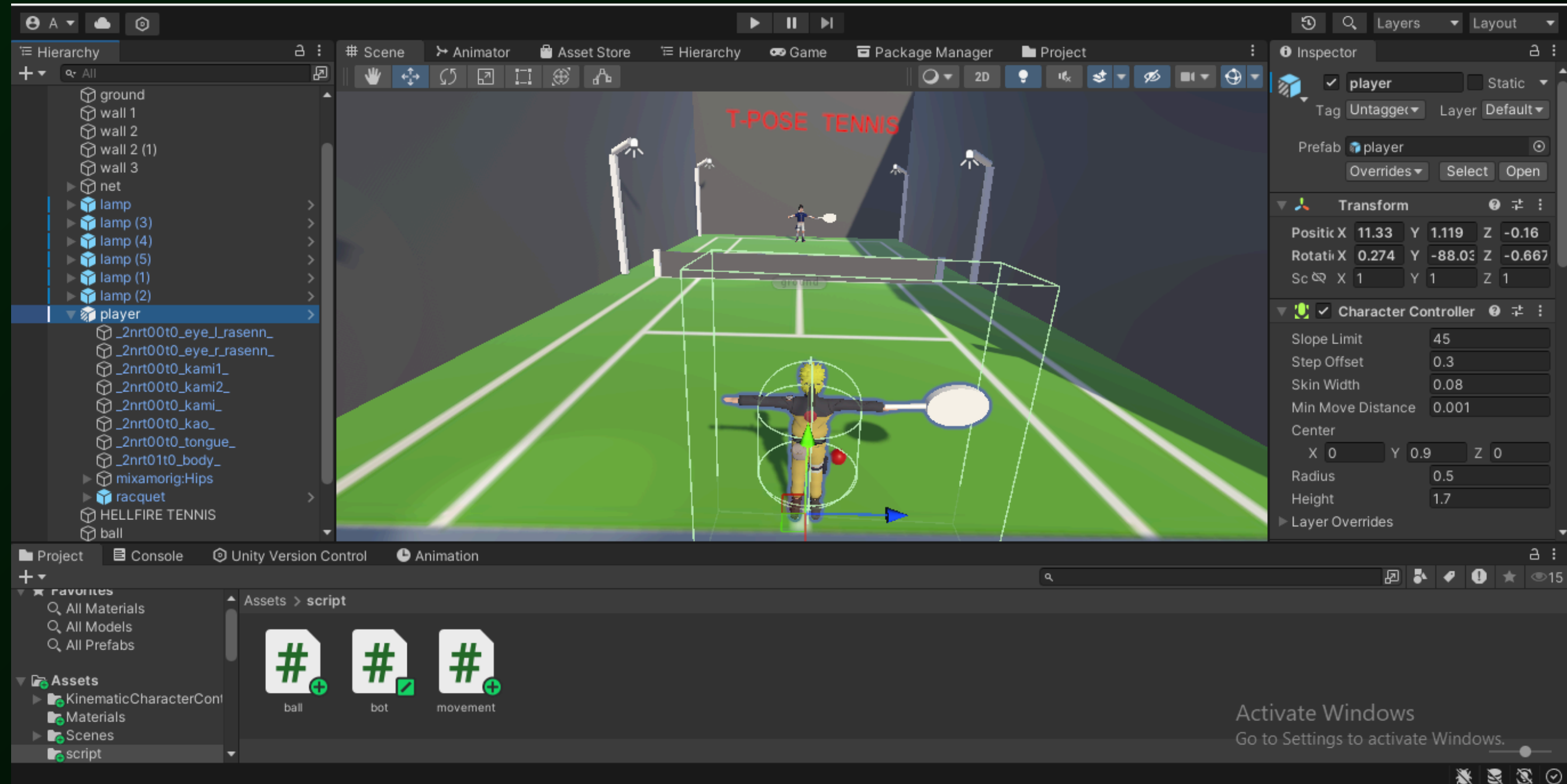
# OBJECTIVE OF THE GAME

## Primary Goal:

Keep the ball in play by hitting it back and forth with precision and timing.

## Features:

- Realistic player movement.
- AI bot as a challenging opponent.
- Physics-based ball interactions.





# TECHNOLOGY AND SCRIPTS

## ► Technologies Used:

- Unity Physics: Realistic interactions.
- Animator: Smooth animations.
- Input System: Player controls.

## Key Scripts:

1. Movement Script: Player control and hitting mechanics.
2. Bot Script: AI movement and aiming logic.
3. Ball Script: Ball position reset for seamless gameplay.

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

0 references
public class Bot : MonoBehaviour
{
    1 reference
    float speed = 40;
    2 references
    Animator animator;
    2 references
    public Transform ball;
    1 reference
    public Transform aimTarget;

    2 references
    public Transform[] targets;

    1 reference
    float force = 13;
    3 references
    Vector3 targetPosition;

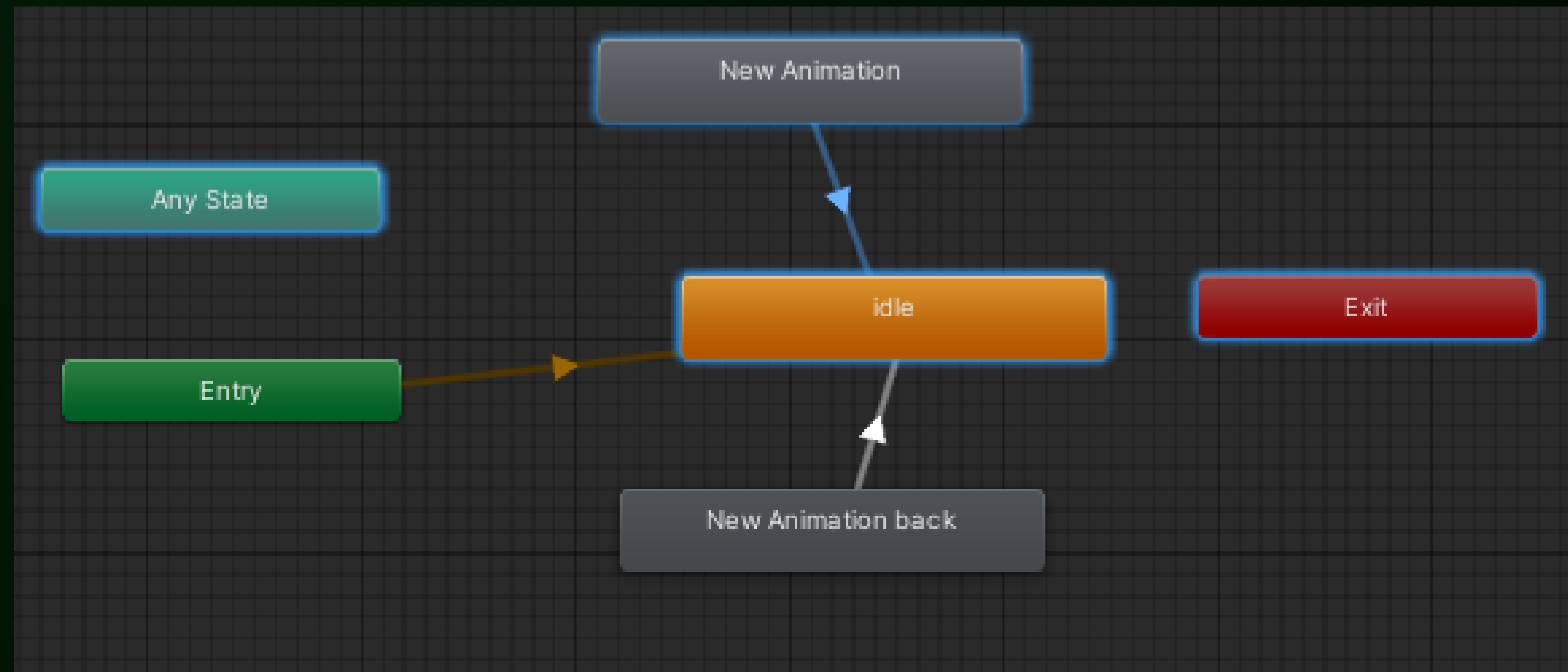
    0 references
    void Start()
    {
        targetPosition = transform.position;
        animator = GetComponent<Animator>();
    }
}
```



# SYSTEM ARCHITECTURE

## Integration:

- Ball behavior: Resets on collision.
- Bot behavior: Tracks and hits the ball.
- Player movement: Dynamic control and aiming.



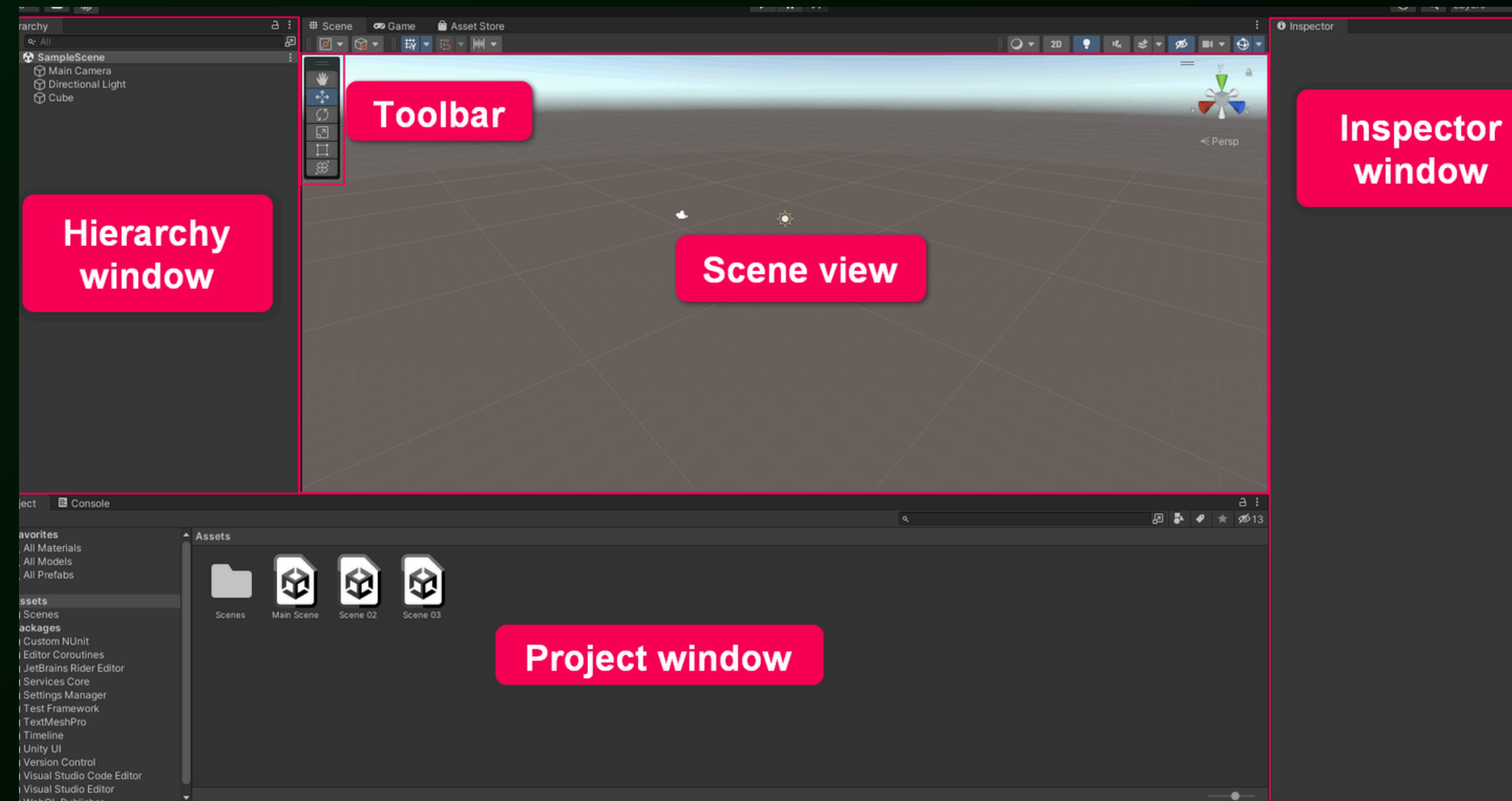
Data Flow Diagram (DFD):  
Player Inputs → Movement  
Script → Animator → Actions.



# HOW TO RUN THE GAME

1. Open Unity Hub: Load the project.
2. Verify Objects: Ensure Player, Bot, Ball, etc., are present.
3. Configure Scripts: Assign values in the Unity Inspector.
4. Press Play: Start playing!

- **Option for pre-built game:**  
GAME





# CONCLUSION AND FUTURE SCOPE

## **Conclusion:**

"This Unity-based project demonstrates interactive gameplay, realistic AI, and smooth animations. It serves as a solid foundation for further development."

## **Future Scope:**

- Improved AI for diverse behaviors.
- Multiplayer functionality.
- Additional levels and customization.