Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки

ЗВІТ

з лабораторної роботи №2

з навчальної дисципліни «Розробка ігрових застосувань. Unity рішення»

Тема:

«Дослідження базового патерну ігрового рушія Unity на прикладі тривимірного ігрового застосунку»

Виконав:

Голік В.М., група ІП-94

Перевірив:

доц. Катін П. Ю.

Київ 2022

**Мета роботи**: полягає у набутті знань, умінь та навичок з технології розроблення основ проекту з використанням обраної мови програмування у обраній парадигмі. Надається досвід створення репозиторію у системі контролю версій і знання елементів середовища розробки і основи вихідного коду для управління грою. Також лабораторна робота дає основні навички розробки з використанням IDE ігрового рушія.

**Вхідні дані**

Студент: Голік Владислав

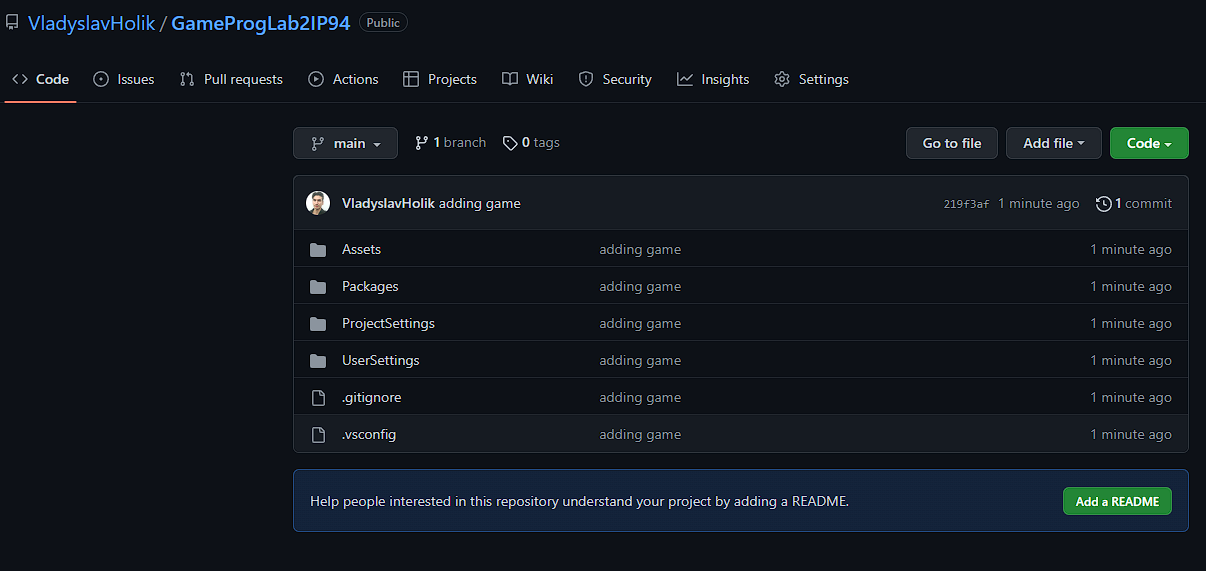
Група: ІП-94

Скорочена назва факультету: ФІОТ

Варіант - 4, примітив - шар, assert – 4 (https://assetstore.unity.com/packages/3d/vegetation/trees/2022-pbr-xfrogplants-sampler-229007).

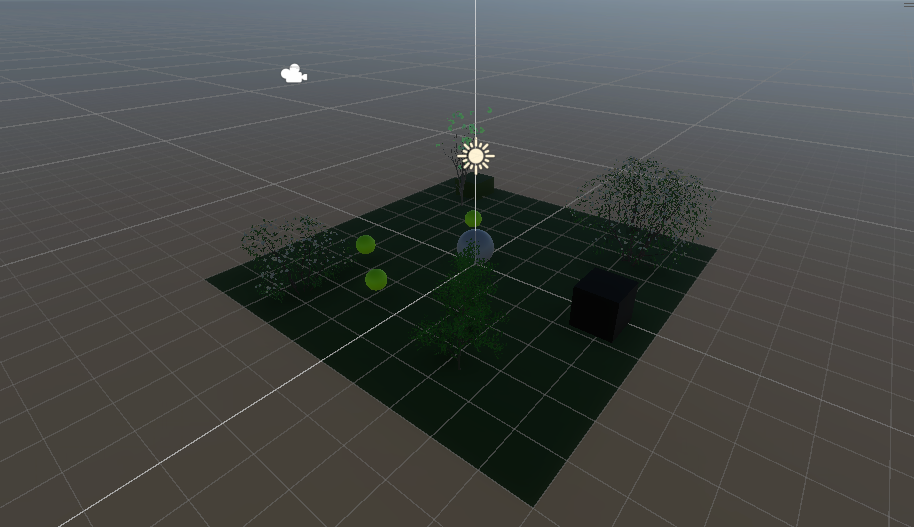
**Хід роботи**

Був створений проект репозиторію https://github.com/VladyslavHolik/GameProgLab2IP94

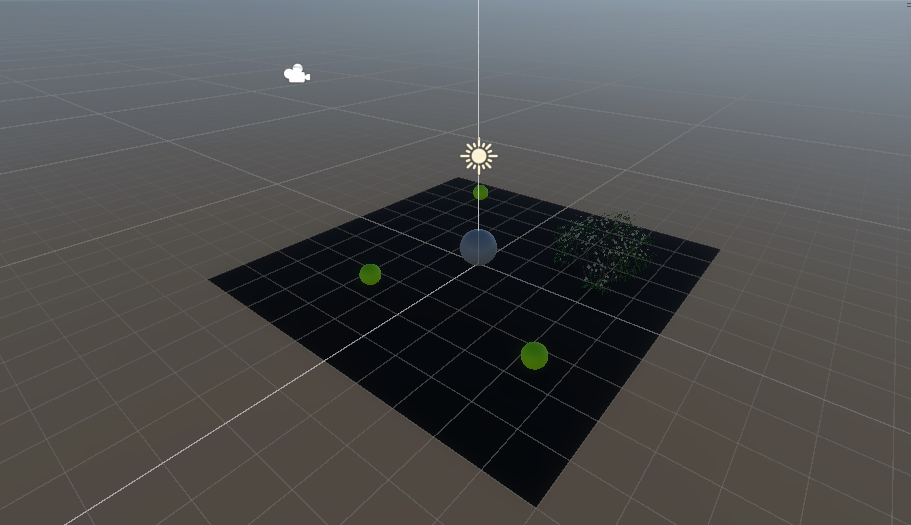


Проект 3D, має дві сцени. В сценах присутні різні види об’єктів, з якими ігровий об’єкт може взаємодіяти, та ні.

Сцена №1:



Сцена №2:



Примітив ігрового персонажу – сфера. Присутній інтерфейс на якому показуються очки гравця. Збираючи маленькі сфери – токени, гравець збільшує свій score. Токени дають різну кількість очок. Присутній об’єкт тригер, при взаємодії з яким, завантажується друга сцена.

Були використані ассерти для дерев згідно з варіантом.

Вихідний код:

PlayerController відповідає за пересування ігрового об’єкта.

public class PlayerController : MonoBehaviour

{

[SerializeField] private LayerMask groundLayers;

[SerializeField] private Transform groundCheck;

[SerializeField] private float moveThrust = 5f;

[SerializeField] private float jumpThrust = 5f;

private Rigidbody rigidBody;

void Start()

{

rigidBody = GetComponent<Rigidbody>();

}

private void Update()

{

if (OnGround() && Input.GetButtonDown("Jump"))

{

Jump();

}

}

private void FixedUpdate()

{

Move();

}

private void Move()

{

var horizontal = Input.GetAxis("Horizontal");

var vertical = Input.GetAxis("Vertical");

var direction = new Vector3(horizontal, 0, vertical);

var moveVector = transform.TransformDirection(direction) \* moveThrust;

rigidBody.velocity = new Vector3(moveVector.x, rigidBody.velocity.y, moveVector.z);

}

private void Jump()

{

rigidBody.AddForce(Vector3.up \* jumpThrust, ForceMode.VelocityChange);

}

private bool OnGround()

{

return Physics.CheckSphere(groundCheck.position, 0.05f, groundLayers, QueryTriggerInteraction.Ignore);

}

}

TokenController відповідає за збільшення очок гравця при взаємодії.

public class TokenController : MonoBehaviour

{

[Range(1, 10)]

[SerializeField] private int points = 1;

private InterfaceController interfaceController;

private void OnTriggerEnter(Collider other)

{

interfaceController = FindObjectOfType<InterfaceController>();

if (other.gameObject.CompareTag("Player"))

{

interfaceController.Add(points);

Destroy(gameObject);

}

}

}

InterfaceController слугує для відображення поточного значення очок гравця.

public class InterfaceController : MonoBehaviour

{

[SerializeField] private TextMeshProUGUI scoreText;

private int \_score = 0;

private void Awake()

{

SetScore(0);

}

public void Add(int points)

{

SetScore(\_score + points);

}

private void SetScore(int score)

{

\_score = score;

scoreText.text = $"Score: {\_score}";

}

}

PortalController відповідає за завантаження другої сцени при взаємодії з порталом.

public class PortalController : MonoBehaviour

{

private void OnTriggerEnter(Collider other)

{

if (other.gameObject.CompareTag("Player"))

{

SceneManager.LoadScene(1);

}

}

}

**Висновок:** під час виконання даної лабораторної роботи була виконана її мета, а саме: набуття знань, умінь та навичок з технології розроблення основ проекту з використанням обраної мови програмування у обраній парадигмі, отримання досвіду створення репозиторію у системі контролю версій і знання елементів середовища розробки і основи вихідного коду для управління грою, набуття основних навичок розробки з використанням IDE ігрового рушія. Я створив 3D проект, ознайомився з TextMesh, написав контроллери для відображення, обчислення очок гравця та руху ігрового об’єкту, контроллер для переходу на іншу сцену.