Лабораторная работа №28-29

Тема: Разработка игры Ant Smasher

Ход работы:

1.Выполнение работы

Были сделаны спрайты, UI ,префабы, счёт, анимации, спавны, меню.

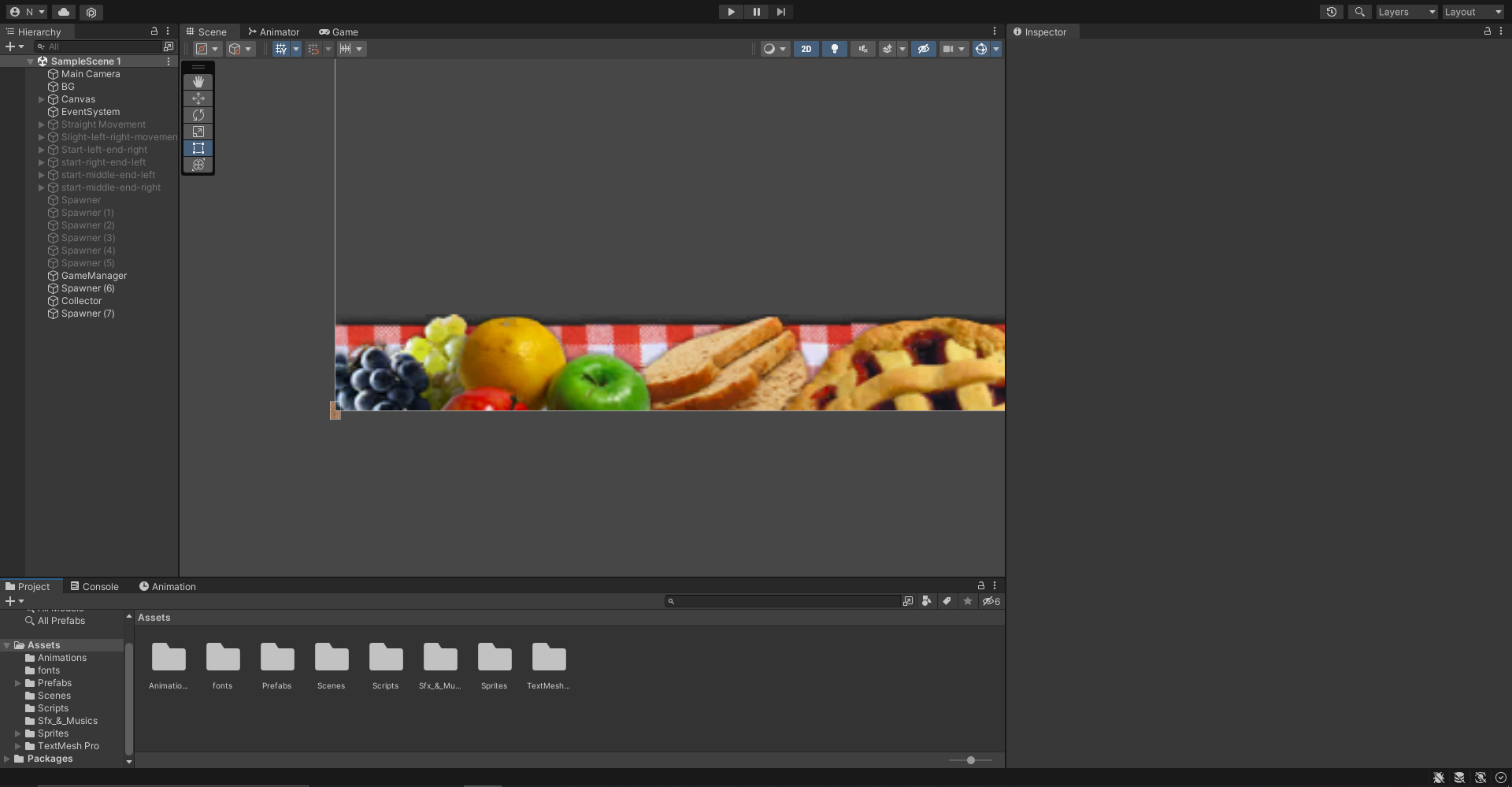


Рисунок 28.1 – Постройка сцены.

Был написан скрипт для анимации кнопок при их нажатии.

Листинг 28.1 Button\_Script.cs

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Button\_Script : MonoBehaviour

{

private Animator ButtonAnimator;

void Start()

{

ButtonAnimator = GetComponent<Animator>();

}

private void OnMouseUpAsButton()

{

ButtonAnimator.SetTrigger("playbt");

}

}

Был написан скрипт для реализации жизней игрока.

Листинг 28.2 LifeSystem.cs

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class LifeSystem : MonoBehaviour

{

public GameObject[] lifes;

public static int lifeCount;

void Start()

{

lifeCount = 2;

}

void Update()

{

if(lifeCount==5)

{

lifes[0].SetActive(true);

lifes[1].SetActive(true);

lifes[2].SetActive(true);

lifes[3].SetActive(true);

lifes[4].SetActive(true);

}

if (lifeCount == 4)

{

lifes[0].SetActive(true);

lifes[1].SetActive(true);

lifes[2].SetActive(true);

lifes[3].SetActive(true);

lifes[4].SetActive(false);

}

if (lifeCount == 3)

{

lifes[0].SetActive(true);

lifes[1].SetActive(true);

lifes[2].SetActive(true);

lifes[3].SetActive(false);

lifes[4].SetActive(false);

}

if (lifeCount == 2)

{

lifes[0].SetActive(true);

lifes[1].SetActive(true);

lifes[2].SetActive(false);

lifes[3].SetActive(false);

lifes[4].SetActive(false);

}

if (lifeCount == 1)

{

lifes[0].SetActive(true);

lifes[1].SetActive(false);

lifes[2].SetActive(false);

lifes[3].SetActive(false);

lifes[4].SetActive(false);

}

if (lifeCount == 0)

{

lifes[0].SetActive(false);

lifes[1].SetActive(false);

lifes[2].SetActive(false);

lifes[3].SetActive(false);

lifes[4].SetActive(false);

}

}

}

Был написан скрипт для подсчёта очков и изменения лучшего счёта.

Листинг 28.3 ScoreScript.cs

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using TMPro;

public class ScoreScript : MonoBehaviour

{

private TextMeshProUGUI scoreText;

public static int score = 0;

void Start()

{

scoreText = GetComponent<TextMeshProUGUI>();

}

void Update()

{

scoreText.text = score.ToString();

if (PlayerPrefs.GetInt("Best") < score)

{

PlayerPrefs.SetInt("Best", score);

}

}

}

Был написан скрипт, отвечающий за уничтожение врагов, их распознование, состояние жизни или смерти, а так же нанесение урона по жизни игрока.

Листинг 28.4 Ant\_Script.cs

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Ant\_Script : MonoBehaviour

{

private GameObject AliveAnt;

private Transform DeadAnt;

private Transform parent;

public bool isBee;

public bool isLife;

private GameManager gameManager;

void Start()

{

gameManager = FindObjectOfType<GameManager>().GetComponent<GameManager>();

if (!isLife)

{

AliveAnt = this.gameObject;

DeadAnt = transform.GetChild(0);

parent = transform.parent;

}

}

private void OnCollisionEnter2D(Collision2D other)

{

if(!isBee)

{

LifeSystem.lifeCount -= 1;

if (LifeSystem.lifeCount == 0)

{

gameManager.GameOver();

}

}

Destroy(parent.gameObject);

}

private void OnMouseUpAsButton()

{

if(isBee)

{

gameManager.GameOver();

}

if(isLife)

{

Destroy(gameObject);

LifeSystem.lifeCount += 1;

}

if (!isLife)

{

AliveAnt.GetComponent<SpriteRenderer>().enabled = false;

AliveAnt.GetComponent<BoxCollider2D>().enabled = false;

DeadAnt.gameObject.SetActive(true);

parent.GetComponent<Animator>().enabled = false;

if (!isBee)

{

ScoreScript.score += 1;

}

Destroy(parent.gameObject, 5);

}

}

void Update()

{

if(isLife)

{

Vector2 pos = transform.position;

pos.y -= 0.4f \* Time.deltaTime;

transform.position = pos;

}

}

}

Был написан скрипт для управления UI игрока, управления паузой и спавнерами противников, а так же было реализовано поражение игрока.

Листинг 28.5 GameManager.cs

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using TMPro;

public class GameManager : MonoBehaviour

{

public TextMeshProUGUI ScoreText, BestScoreText, PlayTimeText;

public GameObject GameOverPanel;

public GameObject[] spawner;

private float NextSpawnerTime;

void Start()

{

NextSpawnerTime = Time.time;

}

void Update()

{

SpawnerManagement();

}

public void GameOver()

{

Time.timeScale = 0;

GameOverPanel.SetActive(true);

ScoreText.text = ScoreScript.score.ToString();

BestScoreText.text = PlayerPrefs.GetInt("Best").ToString();

PlayTimeText.text = Time.time.ToString();

}

public void Pause()

{

Time.timeScale = 0;

}

public void Resume()

{

Time.timeScale = 1;

}

void SpawnerManagement()

{

if(Time.time>NextSpawnerTime)

{

int spr = Random.Range(0, 6);

if(spr==0)

{

spawner[0].SetActive(true);

}

if (spr == 1)

{

spawner[1].SetActive(true);

}

if (spr == 2)

{

spawner[2].SetActive(true);

}

if (spr == 3)

{

spawner[3].SetActive(true);

}

if (spr == 4)

{

spawner[4].SetActive(true);

}

if (spr == 5)

{

spawner[5].SetActive(true);

}

NextSpawnerTime = Time.time + 2;

Invoke("DeactivateAllSpawners", 2);

}

}

void DeactivateAllSpawners()

{

spawner[0].SetActive(false);

spawner[1].SetActive(false);

spawner[2].SetActive(false);

spawner[3].SetActive(false);

spawner[4].SetActive(false);

spawner[5].SetActive(false);

}

}

Был написан скрипт, который отвечает за функционирование спавнеров противников и дополнительных жизней игрока.

Листинг 28.6 Spawner.cs

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Spawner : MonoBehaviour

{

public bool isLife;

public GameObject[] ants;

public float minX, maxX, SpawnRate;

private float NextSpawn;

void Start()

{

if (!isLife)

{

NextSpawn = Time.time;

}

if (isLife)

{

NextSpawn = 60;

}

}

void Update()

{

Spawn();

}

void Spawn()

{

if (Time.time > NextSpawn)

{

Vector2 position = new Vector2(Random.Range(minX, maxX), transform.position.y);

Instantiate(ants[Random.Range(0, ants.Length)], new Vector2(position.x, position.y), transform.rotation);

NextSpawn = Time.time + SpawnRate;

}

}

}

2. Вывод.

В ходе выполненной работы была сделана игра Ant Smasher, в которой требуется защищать еду от муравьёв.