

Федеральное агентство связи
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет
телекоммуникаций и информатики»

РГР по теме «Бегущий»

Выполнил: студент 3 курса

ИВТ, гр. ИП-713

Михеев Н.А.

Проверила:

ассистент кафедры ПМиК

Павлова У.В.

Новосибирск, 2020 г.

1. Постановка задачи

В рамках выполнения расчетно-графической работы необходимо было реализовать игровое приложение с помощью кроссплатформенной среды разработки игр Unity.

Был использован за основу предложенный вариант игры «Бегущий» с небольшими изменениями для разнообразия геймплея. Имеется фигурка игрока, который собирает монетки бегая и прыгая по различным платформам. Его задача собрать как можно больше монет по пути на финиш и не упасть в пропасть.

2. Описание

Игра была реализована полностью с использованием среды Unity, для логики работы игры так же были прописаны скрипты на языке C#.

Игровая логика включает в себя:

- создание игровых сцен (меню, уровень)
- реализацию движения игрока по нажатию на специальные кнопки на дисплее смартфона
- логику «получения» монеток, учет соответствующих коллизий
- перезапуск игры в случае провала игрока

Все игровые элементы – спрайты, имеют специальные коллайдеры для регистрации взаимодействия друг с другом.

После запуска приложения – демонстрируется загрузочная заставка и игрок попадает в меню, где он может начать новую игру или закрыть приложение. Начав новую игру, перед игроком предстает интерфейс с элементами управления и отображением текущего счета монет. После прохождения подготовленного уровня, игроку показывается сообщение об окончании и его счет.

3. Скриншоты работы игры



Рис.1 – Загрузочная заставка.



Рис.2 – Меню игры.

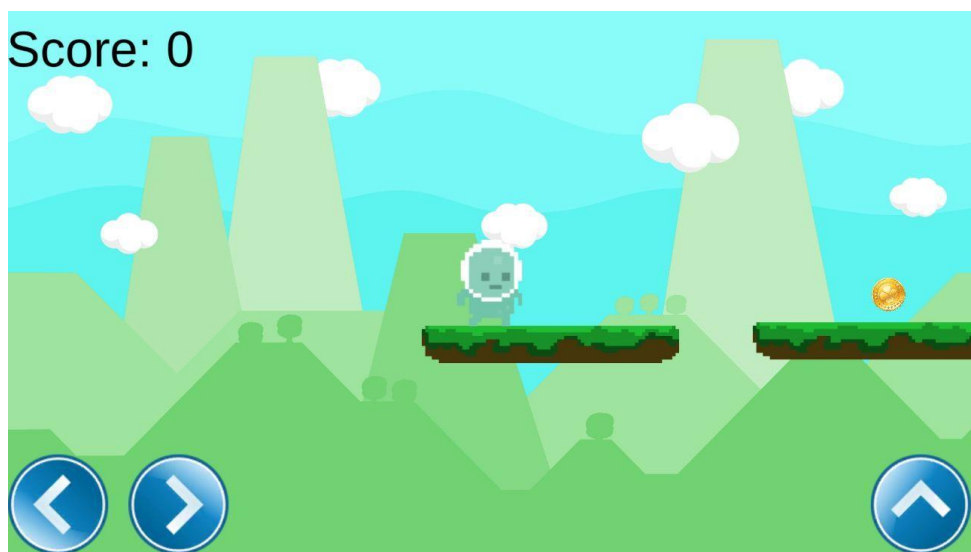


Рис.3 – Старт уровня – на экране персонаж, клавиши управления, текущий счет.

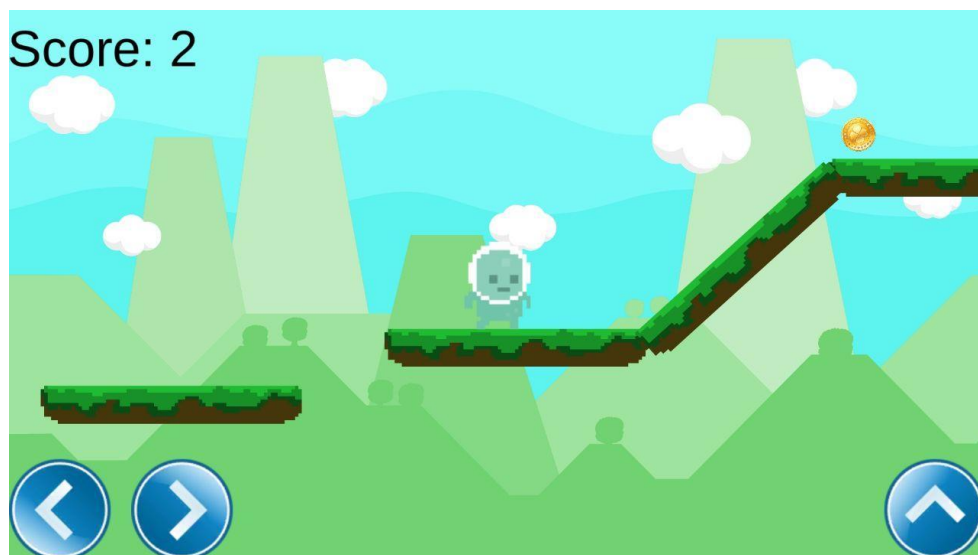


Рис.4 – Дальнейшая демонстрация уровня.

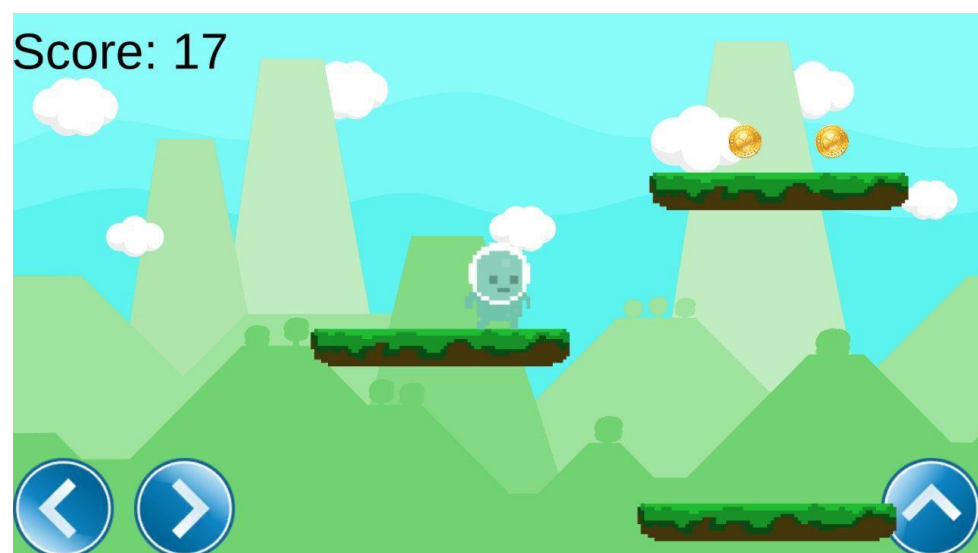


Рис. 5 – Дальнейшая демонстрация уровня.

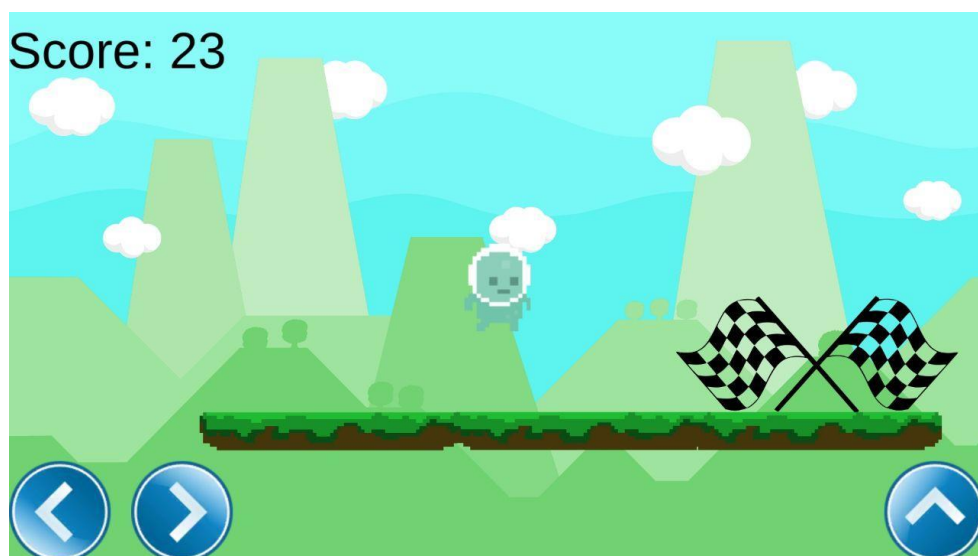


Рис. 6 – Демонстрация финала уровня.



Рис.7 – Окно об успешном окончании игры.

4. Листинг игровых скриптов

Скрипт меню для старта игры:

```
using UnityEngine;
using UnityEngine.Events;
using UnityEngine.SceneManagement;

public class StartGame : MonoBehaviour
{
    public void NextScene()
    {
        SceneManager.LoadScene("SampleScene");
    }

    public void ExitPressed()
    {
        Application.Quit();
        Debug.Log("Exit pressed!");
    }
}
```

Скрипт по управлению игроком и проверкой для сбора монет:

```
using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class PlayerController : MonoBehaviour
{
    public Text txt;
    public Rigidbody2D rb2d;
    public float playerSpeed;
    public float jumpPower;
    public int directionInput;
```

```

public bool groundCheck;
public bool facingRight = true;
public int coin = 0;

void Start()
{
    rb2d = GetComponent<Rigidbody2D>();
}

void OnTriggerEnter2D(Collider2D other) {
    if(other.gameObject.tag == "Collectable") {
        coin++;
        txt.text = "Score: " + coin.ToString();
        //other.gameObject.SetActive(false);
        Destroy(other.gameObject);
    }
}

void Update()
{
    if ((directionInput < 0) && (facingRight))
    {
        Flip();
    }

    if ((directionInput > 0) && (!facingRight))
    {
        Flip();
    }
    groundCheck = true;
}

void FixedUpdate()
{
    rb2d.velocity = new Vector2(playerSpeed * directionInput, rb2d.velocity.y);
}

public void Move(int InputAxis)
{
    directionInput = InputAxis;
}

public void Jump(bool isJump)
{
    isJump = groundCheck;

    if (groundCheck)
    {
        rb2d.velocity = new Vector2(rb2d.velocity.x, jumpPower);
    }
}

```

```

    }

    void Flip()
    {
        facingRight = !facingRight;
        Vector3 theScale = transform.localScale;
        theScale.x *= -1;
        transform.localScale = theScale;
    }
}

```

Скрипт для плавного движения камеры за персонажем:

```

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

public class CamMovement : MonoBehaviour
{
    public GameObject player;
    void Update()
    {
        transform.position = new Vector3 (player.transform.position.x, player.trans
        form.position.y, -10f);
    }
}

```

Скрипт для реализации зон проигрыша:

```

using UnityEngine;
using UnityEngine.SceneManagement;

public class DieSpace : MonoBehaviour
{
    public GameObject respawn;
    private void OnTriggerEnter2D (Collider2D other) {
        if (other.tag == "Player") {
            SceneManager.LoadScene("SampleScene");
        }
    }
}

```