**чу Поо «столичный БИЗНЕС КОЛлЕДЖ»**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Курсовая работа**

**По предмету**

**«Технология разработки ПО»**

**На тему**

**«Выбор среды разработки приложения»**

**Выполнила: студентка гр. 1304**

**Алейчик Виктория Леонидовна**

**Проверил: преподаватель  
Киселев Леонид Александрович**

**Москва2021**

Содержание

[Постановка задачи 3](#_Toc89871000)

[Описание спрайтов 3](#_Toc89871001)

[Схема взаимодействий объектов 6](#_Toc89871002)

[Описание процесса разработки 7](#_Toc89871003)

[Разработка игрового уровня 7](#_Toc89871004)

# Постановка задачи

В игре должны присутствовать следующие элементы: вид на игру сбоку; управление клавишами WASD; пауза (при нажатии кнопки ESC); запись количества собранных кристаллов игроком; количество жизней; враг; бонус «Вишня» который дает выше прыжок; выход (дверь в домике).

# Описание спрайтов

Одним из главных элементов игры является персонаж «Лисенок» (рис. 1); имеет 4 анимации: стоять, прыжок, бег и падение. Анимация «стоять» имеет 4 кадра спрайта (рис. 2), «прыжок» имеет 1 кадр спрайт (рис. 3), «бег» имеет 6 кадров спрайтов (рис. 4), «падение» имеет 1 кадр спрайт (рис. 5).

|  |  |  |  |
| --- | --- | --- | --- |
| Рисунок 1. Персонаж «Лисенок» | | | |
|  |  |  |  |

Рисунок 2. анимация «Стоять»

|  |
| --- |
| Рисунок 3. анимация «Прыжок» |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Рисунок 4. анимация «Бег»

|  |
| --- |
| Рисунок 5. анимация «Падение» |

Враг «Опоссум» ходит по заданной траектории «туда-сюда» (рис. 6).

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Рисунок 6. Враг «Опоссум»

Спрайты кристаллов (рис. 7); жизни игрока (рис. 8); Бонус «Вишня» (рис. 9).

|  |  |  |
| --- | --- | --- |
| Рисунок 7. Кристалл | Рисунок 8. Жизнь игрока | Рисунок 9. Бонус «Вишня» |

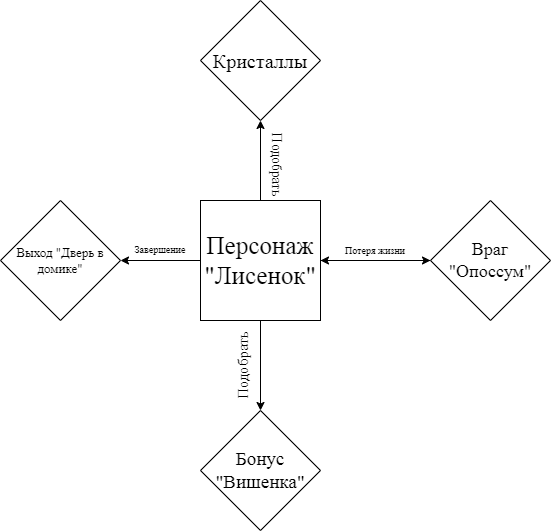
Сцена построена через «TileSet», так же есть отдельные декорации на заднем фоне (рис.10-12).

|  |
| --- |
| Рисунок 10. TileSet Рисунок 11. Декорации |
| Рисунок 12. Сцена с уровнем |

Выход (дверь в домике) (рис. 13)

|  |
| --- |
| Рисунок 13. Выход |

# Схема взаимодействий объектов



Персонаж «Лисенок» подбирает кристаллы и Бонус «Вишенка».

Персонаж «Лисенок» теряет жизнь с объектом «Опоссум» и наоборот.

Персонаж «Лисенок», касаясь «двери в домике» завершает уровень.

# Описание процесса разработки

## Разработка игрового уровня

Для начало игрового уровня был создан объект «Grid» с объектами «GroundAktiv» и «GroundPasiv». В «GroundAktiv» и «GroundPasiv» создаем компонент «TileMap», вместе с ним создается компонент «TileMap Renderer». В объекте «GroundAktiv» создаем компонент «TileMap Collider 2D», это нужно для того, чтоб персонаж не проваливался сквозь текстуры. Дальше открываем в верхней панели Window – 2D – Tile Palette. В Tile Palette добавляем TileSet и рисуем уровень и дополняем все это декорациями (рис. 14).



Рисунок 14. Создание «Grid»

Далее создаем объект «Player» с объектом «isGround», это будущий персонаж «Лисенок» (рис. 15). К объектам добавляем компонент «Capsule Collider 2D». К объекту «Player» добавляем еще «Sprite Renderer» и «Rigidbody 2D». И добавляем скрипты «Controller» и «Complited Menu» (рис. 16-19). В скрипте «Controller» описываем движение и взаимодействия с другими объектами персонажа «Лисенок». Создаются переменные (рис. 20). Метод Void Star остается пустым, в методеVoid HeartLoss создаем отнятие жизней персонажу (рис. 21). В методе private void MakeJump создаем прыжок персонажу, с помощью добавления силы. В методе private void PlayerMove создается передвижение персонажа с помощью скорости по оси Y. В методе private void PlayerFlip создаем повороты по оси Y. В методе void FixedUpdate вызываются методы PlayerMove и PlayerFlip. Метод PlayerMove вызывает

движение, и если оно больше 0 и персонаж не повернут в право или персонаж повернут в право и меньше 0, то вызывается метод PlayerFlip (рис. 22). В методах private void OnTriggerEnter2D и OnTriggerExit2D описываем пересечение коллайдеров с другими объектами (рис. 23).

В скрипте Complited Menu описываем завершение уровня. Метод Void Star остается пустым. В методе OnTriggerEnter2D описываем пересечение коллайдера и вызова объекта (рис. 24). В методах public void Next и public void Menu создаются переходы в другие сцены (рис. 25).



Рисунок 15. Объект «Player»

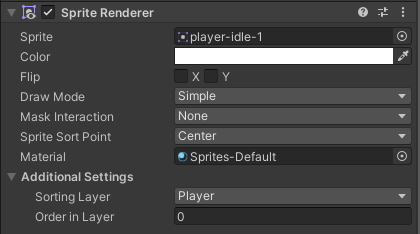


Рисунок 16. Компонент «Sprite Renderer»

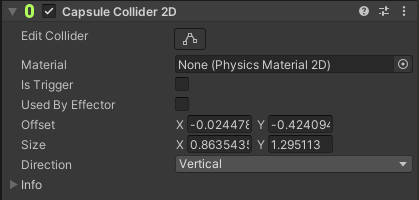


Рисунок 17. Компонент «Capsule Collider 2D»

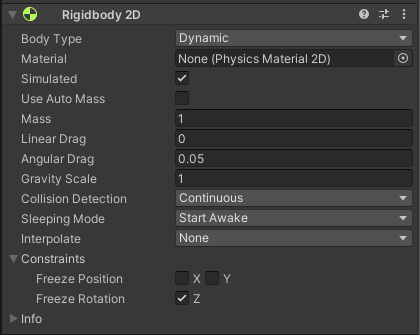


Рисунок 18. Компонент «Rigidbody 2D»

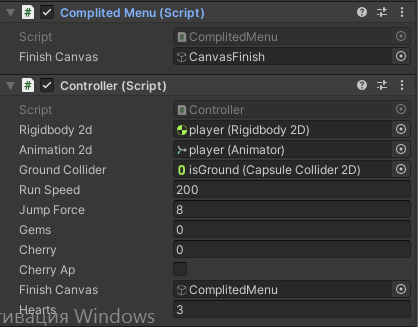


Рисунок 19. Скрипты «Controller» и «Complited Menu»

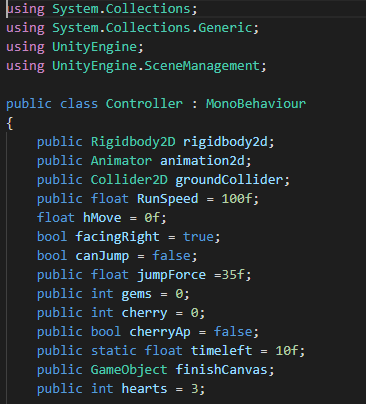


Рисунок 20. Создание переменных.

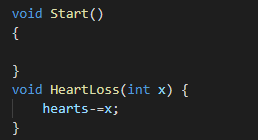


Рисунок 21. Start и HeartLoss



Рисунок 22. Движение персонажа

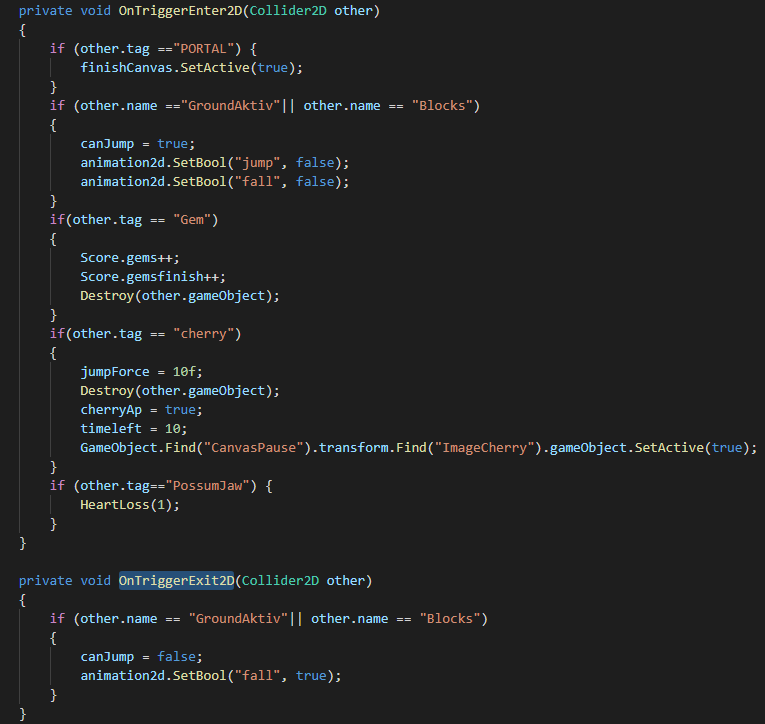


Рисунок 22. Методы OnTriggerEnter2D и OnTriggerExit2D

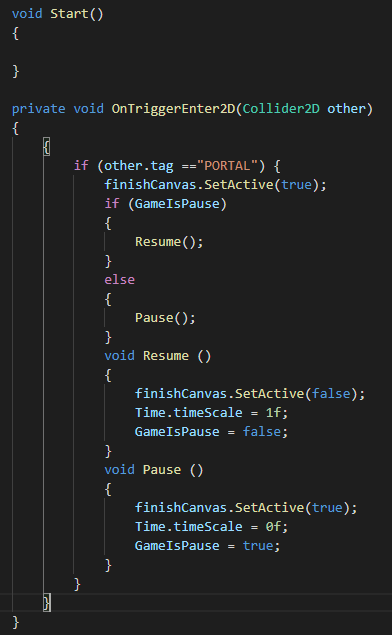


Рисунок 24. Методы Start и OnTriggerEnter2D

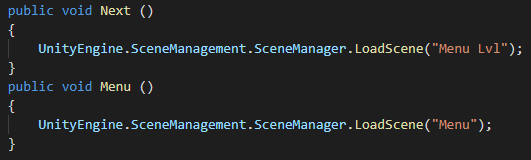


Рисунок 25. Методы Next и Menu

Далее создаем кристаллы и бонус «Вишенка». Для этого создаем объект «Gems», там будут Prefab кристаллов и «Вишенки». Дальше создаем еще один объект под названием «ScoreGem», тут будет скрипт подсчета кристаллов (рис. 26). К Prefab кристаллов и «Вишенки» добавляем компоненты «Sprite Renderer» и «Circle Collider 2D» (рис. 27). К Prefab «Вишенки» добавляем еще скрипт «CherryCont» (рис. 28). К объекту ScoreGem добавляем скрипт «Score» (рис. 29).



Рисунок 26. Создание объекта Gems

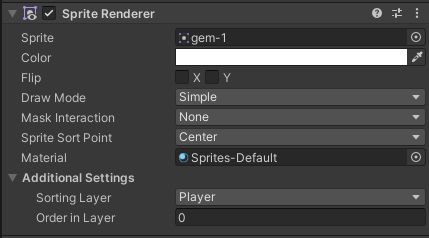


Рисунок 27. Компонент «Sprite Renderer»

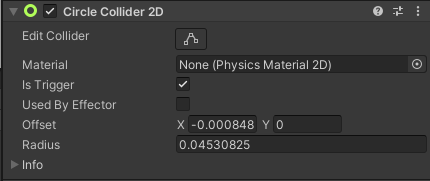


Рисунок 28. Компонент «Circle Collider 2D»