Info

Игра Asteroids 1979, воссозданная на движке Unity. Главное, чего я пытался добиться немного осовременить игру и передать основные механики игры.

В осовременивание вошли такие вещи как:

* Расширение границ с квадрата до прямоугольника с соотношением сторон (16:9).
* Измененная анимация спавна корабля.
* Добавлена анимация поворота ракеты.
* Добавлена разрушаемость корабля и пришельца на собственные части.
* Изменены все текстуры.

**Документация для разработчиков**

1. Обзор проекта в Unity
   1. Расположение ассетов
   2. Построение сцены
2. Контроллер сцены
   1. GameController
3. Интерфейс
   1. Canvases для разных состояний игры
   2. UIController
4. Игрок (ship)
   1. Ship (Script)
   2. Ship Fire
   3. Animation Controller
   4. Части коробля (Ship Parts)
5. Астероиды (asteroid)
   1. Asteroid (Script)
   2. Части астероида (Asteroids Parts)
6. Пришелец (alien)
   1. Alien (Script)
   2. Alien Fire (Script)
   3. Alien Parts
7. Звуки в игре
   1. AudioController

Обзор проекта в Unity

Kuva, joka sisältää kohteen teksti

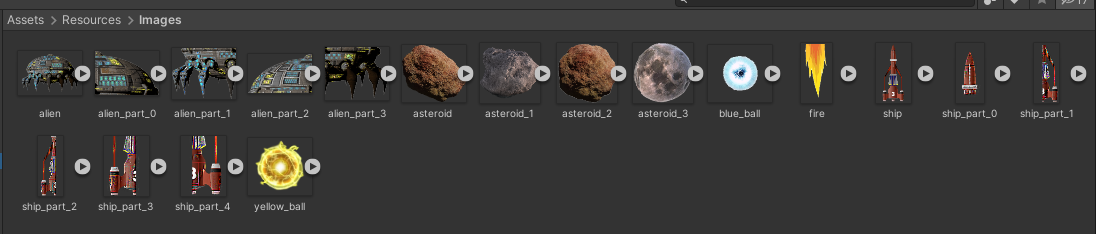
Kuvaus luotu automaattisesti**Расположение ассетов**

Все префабы (Prefabs), звуки (Sounds) и изображения (Images) находятся в папке Reources, для быстрого использования в скриптах.

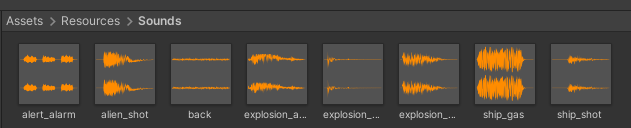
Всего в проекте 15 префабов.



18 изображений (Texture).



8 аудио клипов.



**Построение сцены**

Kuva, joka sisältää kohteen teksti

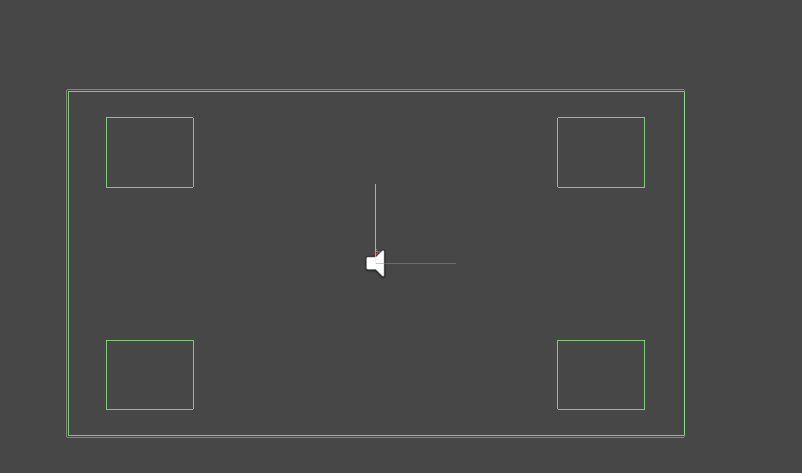
Kuvaus luotu automaattisesti

В процессе разработки я принял решение, что не буду добавлять загрузочную сцену на которой будет спавниться неуничтожаемый GameController, так как я делал игру по примеру видео и своего игрового опыта с сайта [Asteroids](https://www.echalk.co.uk/amusements/Games/asteroidsClassic/ateroids.html). У нас будет всего один бесконечный уровень и других уровней не подразумевается, так что мы будем просто перезагружать сцену для повторной игры после проигрыша.

**GameController**

GameController на сцене.

Сам гаме контроллер содержит коллайдер для спавна внутри него астероидов, так же в нем находятся дочерние объекты (которые тоже содержат коллайдер) для спавна в определенных зонах пришельцев.

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti, näyttö, näyttökuva, tietokone

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Одна из важных функций — это состояние игры (GameCondition). в зависимости от него контроллер спавнит игрока и противников. Так же в зависимости от состояния показыветься нужны интерфейс (UI).

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Сам контроллер регулирует всю игровую активность. Он спавнит игрока, астероиды и пришельцев. Считает жизни, астероиды и очки.

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisestiKuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

**SpawnShip ()**

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

с помощью этой функции мы спавним корабль (который мы загружаем из ресурсов) игрока на центр поля и включаем у него режим «NonTarget», когда у игрока отключен коллайдер и ему нельзя нанести урон. Он продлиться 1.5 секунды.

**AsteroidSpawn()**

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

С помощью этой функции мы спавним астероид (который мы загружаем из ресурсов). Используем функцию GetRandomPosition() для определения позиции спавна астероида.

**GetRandomPosition()**

Считаем размеры коллайдера и берем случайное значение X и Y, создаем Vector2 и возвращаем его.

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

**AlienSpawn()**

Работает точно так же как и AsteroidSpawn().

Во время спавна начинается проигрываться предупреждающий звук.

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

**GetRandomPositionAlien()**

Работает точно так же как и GetRandomPosition(), но теперь у нас есть лист с объектами, которые содержат коллайдер для определения места возможного спавна.



Kuva, joka sisältää kohteen teksti

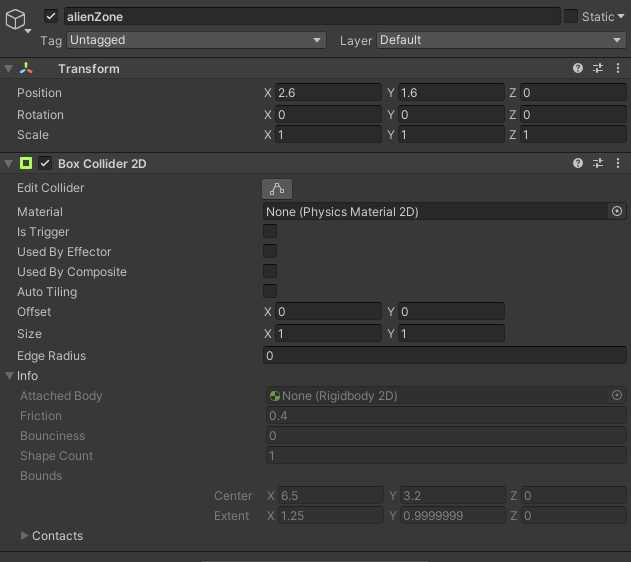
Kuvaus luotu automaattisesti

Сам GameController содержит 4 объекта-зоны, которые располагаются в каждом угле игрового поля.

Kuva, joka sisältää kohteen pöytä

Kuvaus luotu automaattisesti Kuva, joka sisältää kohteen teksti, elektroniikka

Kuvaus luotu automaattisesti



GameController так же отвечает за подсчет всех объектов на поле, кроме \*\_parts (так как они уничтожаются самостоятельно и не влияют на подсчеты).

Список функций и их содержание:

1. AddAsteroid()
2. DeleteAsteroid()
3. AddScore(int sc)
4. MinusLife(int li)
5. GetLife()
6. GetScore()
7. GetShipPosition()
8. GetGameCon()
9. SetGameCon(GameCondition game)

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti, näyttökuva, valkokangas

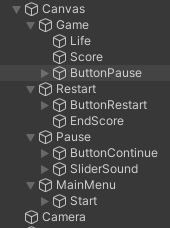
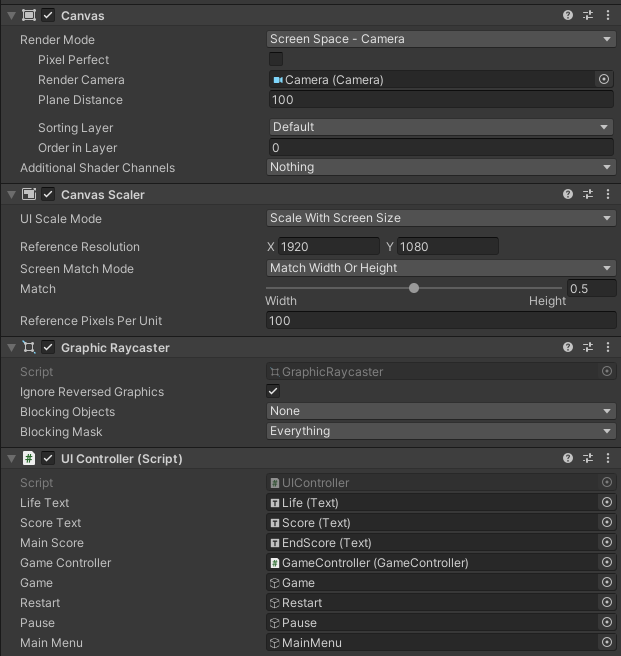
Kuvaus luotu automaattisesti Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisestiKuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

**Интерфейс**

У каждого игрового состояния есть свой объект с UI. Во время одного состояния только один объект является активным, остальные неактивные.

**UIController**

Контроллер обрабатывает все события для интерфейса.

А именно:

1. Отображение счета и жизней игрока.
2. Так же активирование объекта и остановка времени.

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

так же есть функции для кнопок:

1. SetPause()
2. SetPauseOff()
3. SetGame()
4. SetRestart()

Kuva, joka sisältää kohteen teksti

Kuvaus luotu automaattisesti

Заслуживает внимание SetRestart(), в котором как было приведено ранее мы перезагружаем целиком первую сцену.