**TDD**

**Unity3D**

**План:**

* Подготовка ресурсов (анимаций, префабов)
* Редактор характеристик кораблей
* Редактор расположения кораблей противника
* Игровые состояния на основе FSM
* Разработка HUD
* Окно выбора корабля
* Окно игры (отображение статистики игрока и текущей цели)
* Окно окончания игры
* Система уровней
* Система бонусов
* Система движения корабля игрока и противников
* Внедрение звуковой составляющей
* Внедрение 2D физики (для коллизий)

**Дополнительно (эти пункты вы должны реализовать по своему усмотрению):**

* Система частиц для динамического космоса (звезд)
* Больше чем одно вооружение у корабля игрока
* Ракета с тепловым наведением (с захватом цели)
* Лазер (с использованием анимационных текстур)
* Больше чем одно вооружение у кораблей противника
* AI атаки вражеского корабля (корабль противника атакует корабль игрока отделившись от основного отряда)
* Система повреждений (численное отображение повреждений количества жизней вражеского корабля)

**Требования:**

* Разрешение игры 1024х768, camera orthographic size 7.68, camera clear color black, все текстуры должны быть без mipmap уровней
* Для всех моделей кораблей необходимо создать материалы
* Обязательным является использование Objects Pools
* Использовать версию Unity 2018.1.0f2
* Срок выполнения – 2 календарных недели
* Приложение должно быть собрано под PC (**в финальной папке проекта должен находиться \*.exe** для запуска игры).

**Немного Game Design-а:**

В игре присутствуют пять видов кораблей игрока:

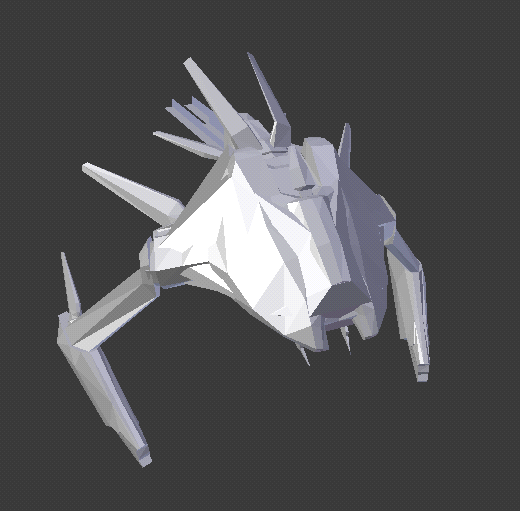
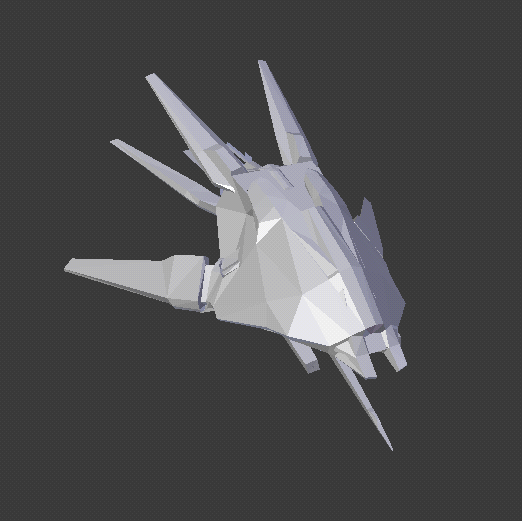
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Player 1 | Player 2 | Player 3 | Player 4 | Player 5 |
|  |  |  |  |  |

Также в игре присутствуют три расы противников по 5 видов кораблей каждой расы:

**Уровни кораблей противника:**

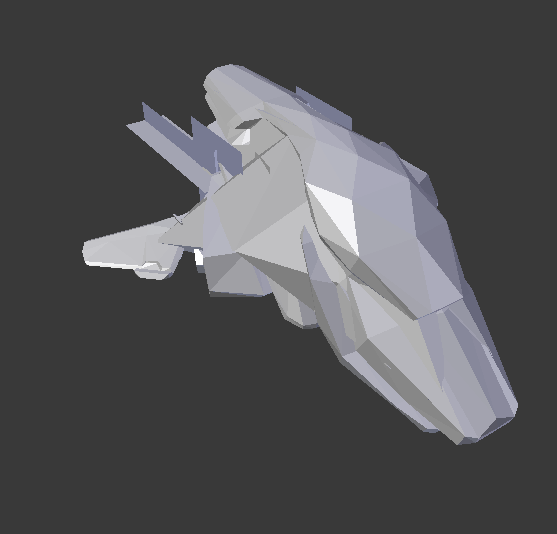
* **Enemy 1**

Level 1 Level 2 Level 3 Level 4 Level 5



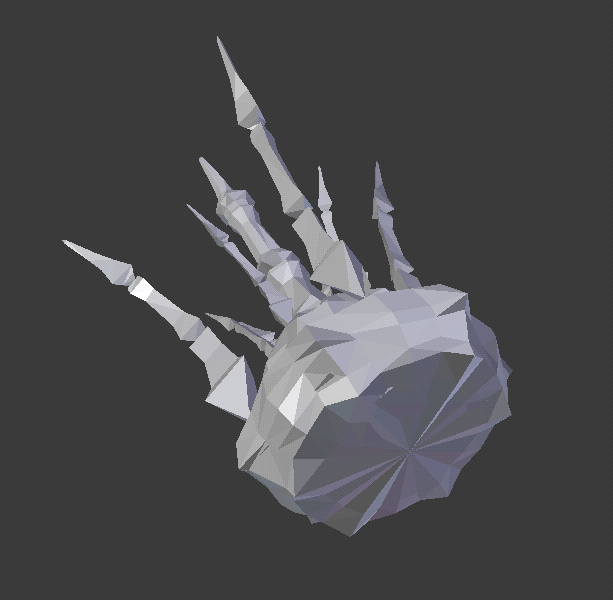
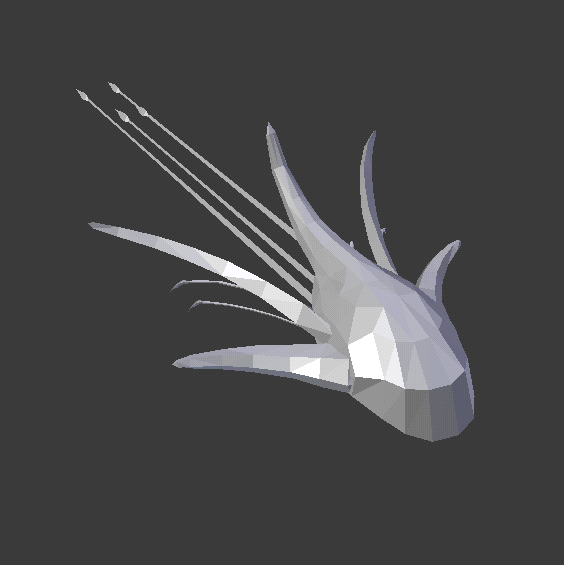
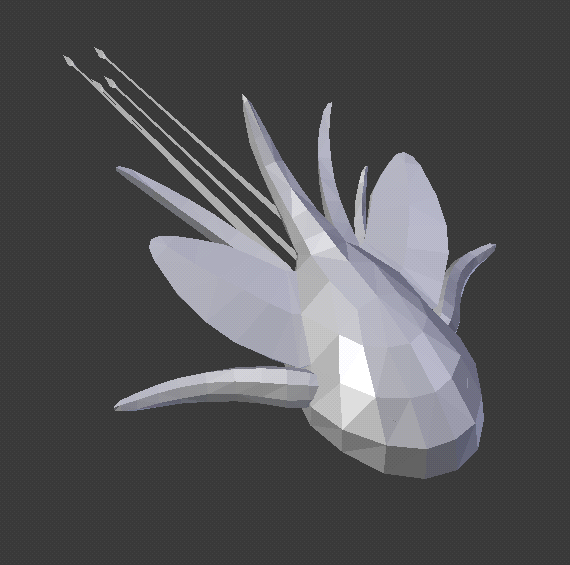
* **Enemy 2**

Level 1 Level 2 Level 3 Level 4 Level 5



* **Enemy 3**

Level 1 Level 2 Level 3 Level 4 Level 5



**В игре присутствует один тип вооружения (не включая пункт дополнительно)**

Missile 1 Missile 2 Missile 3



**Базовые характеристики кораблей (определяются для Игрока и Врагов):**

* BaseHealth (количество урона который можно нанести кораблю) – например 100
* BaseShootingSpeed (периодичность выстрела) – например 1 выстрел в 500 мс
* BaseMovingSpeed (скорость полета корабля игрока или врага вне звена (AI атаки)) – например 10 ед/мс
* BaseMobility (вероятность уклонения) – например 15%

**Множители характеристик кораблей (определяется для всех типов кораблей отдельно):**

* HealthMultiplier – например x2
* ShootingSpeedMultiplier – например x0.5
* MovingSpeedMultiplier – например x3
* MobilityMultiplier – например x1.5
* WeaponDamageMultiplier – например x4

**Финальная формула вычисления характеристики:**

* CurrentHealth = BaseHealth \* Ships[ShipType].HealthMultiplier;
* CurrentShootingSpeed = BaseShootingSpeed \* Ships[ShipType].ShootingSpeedMultiplier;
* CurrentMovingSpeed = BaseMovingSpeed \* Ships[ShipType].MovingSpeedMultiplier;
* CurrentMobility = BaseMobility \* Ships[ShipType].MobilityMultiplier;

**Финальная формула уклонения от снаряда:**

Percent = 100% - CurrentMobility;

Chance = Random[0, 100];

If(Chance > Percent ) { // No damage } else { // Damage }

**Финальная формула получения урона:**

Damage = Weapons[WeaponType].Damage\*Ships[ShipType]. WeaponDamageMultiplier

**Пример финальных характеристик корабля:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Health  Multiplier | Shooting Speed  Multiplier | Moving Speed  Multiplier | Mobility  Multiplier |
|  |  | **Х2** | **Х3** | **Х4** | **Х5** |
| Base Health | **100** | **200** |  |  |  |
| Base  Shooting Speed | **1** |  | **3** |  |  |
| Base Moving Speed | **10** |  |  | **40** |  |
| Base Mobility | **15** |  |  |  | **75** |

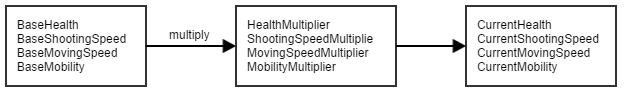
**Подготовка ресурсов (анимаций, префабов):**

Подготовить префабы и анимации для кораблей всех типов, подготовить префабы бонусов, подготовить префабы для GUI.

**Редактор характеристик кораблей:**

Редактор характеристик предназначен для управления балансом игры и заполнения данными основной игровой модели.

**Логические связи базовых характеристик и множителей характеристик:**



**Пример окна редактора характеристик:**





**Редактор расположения кораблей противника:**

На сцене для дизайнера (чистой сцене) расположить корабли противника и используя ScriptableObject сохранить расположения кораблей в \*.asset файл.

**Игровые состояния на основе Finite State Machine (FSM):**

В игре должны присутствовать игровые состояния, реализованные через FSM. Примеры состояний: загрузка ресурсов, главное меню, экран игры.

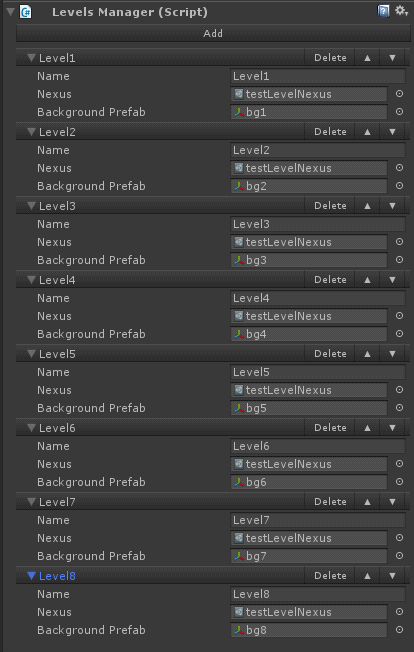
**Разработка HUD:**

Необходимо реализовать HUD на основе стандартной системы GUI предоставленной Unity 3D начиная с версии 4.6. В игре должны присутствовать три окна, такие как: “Выбор корабля игрока”, “Игровой HUD”, “Окно окончания игры”.

**Система уровней:**

Для настройки последовательности уровней в игре необходимо разработать систему уровней и редактор для нее.

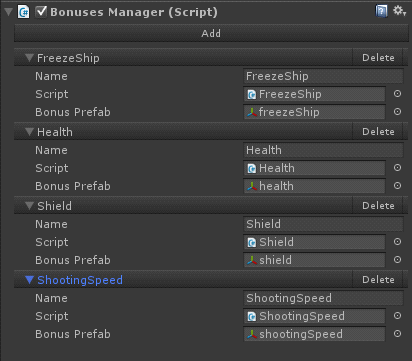
**Пример редактора системы уровней:**



**Система бонусов:**

В игре должна присутствовать система бонусов, а также для этой системы необходимо создать редактор.

**Пример редактора системы бонусов:**

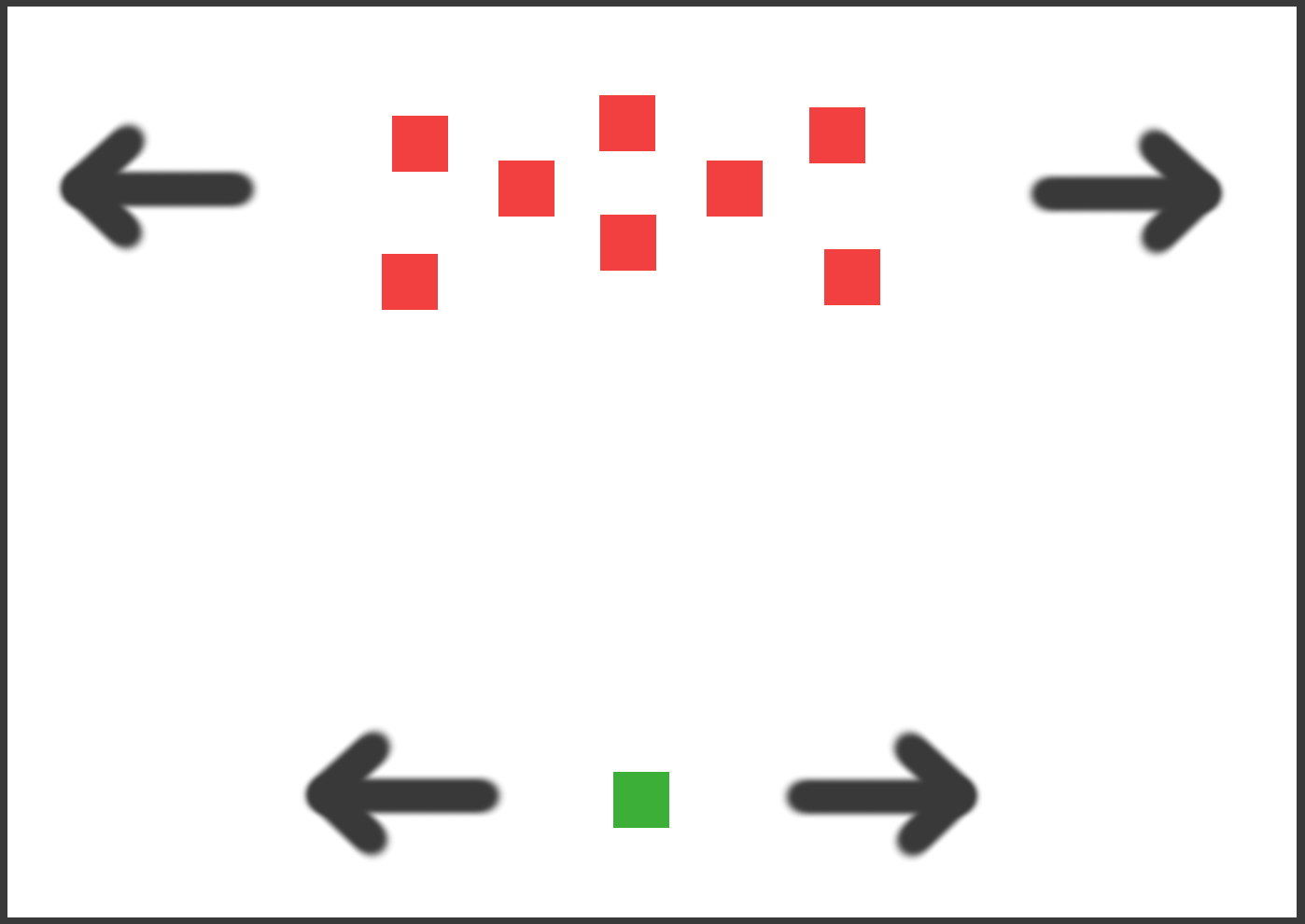


**Система движения корабля игрока и противников:**

Корабли противника должны перемещаться общей группой от правой границы к левой, движение группы кораблей противника вниз **не является обязательным**.

Корабль игрока также должен перемещаться от левой границы к правой.

Также **не является обязательным** инерциальное затухание скорости при остановки движения.



**Внедрение звуковой составляющей:**

В игре должно присутствовать звуковое сопровождение для следующих элементов:

* Бэкграунд музыка отдельно для каждого уровня
* Звук пуска ракеты (если больше, чем одно вооружение - отдельно для каждого типа вооружения)
* Звук для выпавшего бонуса
* Звук подбора бонуса один для всех типов вооружения
* Звук взрыва корабля

**Внедрение физики (для коллизий):**

Для определения столкновений необходимо использовать стандартные механизмы, предоставленные в Unity3D.

**PS:**

Управление кораблем игрока может быть реализовано используя контроллеры по Вашему выбору (мышка, клавиатура и.т.д.). Приложение должно быть собрано под PC (**в финальной папке проекта должен находиться \*.exe** для запуска игры).