Вид:

2д - вертикальный скроллшутер.

----------------------------------------------------------------------------------------------------

Основные фичи:

-2-3 вида врагов.

-подбор предметов, выпавших с врагов

-1-2 уровня (геймплея 2-3 минуты)

-4 вида оружия:

-прямая турель (энергитические снаряды стреляют вперед без разброса).

-ракеты (с рандомной самонаводкой).

-турель (энерг. снаряды стреляют вразброс) (3)

-ракеты (без самонаводки)

-энергит. щит в начале уровня. (заряжается с врагов) (1)

-две статусные полоски. ХП и щит.

-ультимативная способность после накопления энергии с убитых врагов:

-лазер (максимум 5 уровней заряда). (2)

-у каждого корабля 2 своих оружия. ЛКМ и ПКМ стрельба из них.

-Босс (если останется время)

Игры откуда взяты идеи:

1. Revolution Ace (PC)

https://www.youtube.com/watch?v=YdrGfqY2bkE

2. Border Down (Dreamcast)

https://www.youtube.com/watch?v=zHNesI94CQs

3. Jamestown

<https://www.youtube.com/watch?v=4lI3EmLSTg4>

----------------------------------------------------------------------------------------------------

Описание сущностей:

Player

{

Spaceship spaceship

Lives

Event OnPlayerDeath()

TakeDamage(int damage)

Event OnTakeDamage()

ChangeSpaceship()

}

Spaceship

{

Enum SpaceshipType

Weapon mainWeapon

Weapon altWeapon

horizSpeed

vertSpeed

shieldCapacity

shieldEnergy //от 0 до shieldCapacity

dashSpeed

ultraMoveEnergy //от 0 до 100

ultraMoveLevel //от 0 до 5

InitializeSpaceship(SpaceshipType type)

Move()

Dash(Vector2 direction)

ActivateShield()

DeactivateShield()

FireMainWeapon()

FireAltWeapon()

FireUltraMove()

}

Weapon

{

Enum WeaponType

damage

bulletSpeed

fireRate

InitializeWeapon(WeaponType type)

Fire()

}

EnemySpawner

{

Spawn(int count, EnemyType type, EnemyType e\_type, BehaviourType b\_type, EnemyFireType f\_type)

}

Enemy

{

enum EnemyType

{

Enemy1,

Enemy2,

Enemy3

}

enum EnemyFireType

{

Radian,

Direct,

}

enum BehaviourType

{

S\_type,

Z\_type,

}

HP,

Damage,

Firerate

Speed

InitializeEnemy(EnemyType e\_type, BehaviourType b\_type, EnemyFireType f\_type)

TakeDamage()

OnDeathEvent()

Fire(Vector2 direction)

}

PrefabsDictionary

{

Dictionary<string, GameObject> weaponPrefabs

Dictionary<string, GameObject> ammunitionPrefabs

Dictionary<string, GameObject> spaceshipPrefabs

GetWeaponPrefab(string name)

GetAmmoPrefab(string name)

GetSpaceshipPrefab(string name)

}

GameManager

{

Score

Player player

OnEnemyDeathEventHandler(Enemy.EnemyType type)

OnPlayerDeathEventHandler()

ToMainMenu()

ExitGame()

NextLevel()

Pause()

}