

Project ILIA

OpenPlay

Isac Cruz

Project ILIA

Game Development
OpenPlay

Dezembro 2019

Resumo

Agradecimientos

Conteúdo

1	Introdução	1
1.1	Objetos no unity	1
1.2	Player	1
1.3	Game System	2
2	Payer	3
2.1	Componentes	3
2.2	Children	3
3	Base	4
3.1	Componentes	4
3.2	Children	4
4	Componentes	5
4.1	Player Movement	5
4.2	Interactor	5
4.3	Compass Changer	5
4.3.1	Funções	5
4.3.2	Variaveis públicas	5
4.4	Health	5
4.4.1	Funcões	6
4.4.2	Variaveis publicas	6
4.5	Mouse Look	6
4.5.1	Variaveis públicas	6
4.6	Weapon Switching	6
4.6.1	Funcões	6
4.6.2	Variaveis publicas	6
4.7	Gun	6
4.7.1	Variaveis publicas	7
4.8	Base Parent	7
4.8.1	Funcões	7
4.8.2	Variaveis publicas	7
4.9	Computer	7

5	Prefabs	8
5.1	Main Camera	8
5.1.1	Componentes	8
5.1.2	Childrens	8
5.2	Weapon Holder	8
5.2.1	Componentes	8
5.2.2	Childrens	8
5.3	Guns	8
5.3.1	Componentes	9
5.4	Computer	9
5.4.1	Componentes	9
5.4.2	Childrens	9
5.5	Door	9
6	UI	10
7	IA	11
8	Interactable's Data System	12
8.1	Tool	12

Capítulo 1

Introdução

1.1 Objetos no unity

No unity todos os objectos usados no jogo são compostos por componentes.

Por exemplo um cubo que é colocado no mapa possui varios componentes, cada um responsavel por uma caraterística. Neste caso específico os componentes são:

- *Box Collider* : responsavel pela colisão;
- *Mesh Render* : responsavel pela renderização do objeto na tela;
- *Rigid Body* : responsavel pela simulação da fisica do objeto(peso, queficiente de atrito, gravidade, ect..)
- *Transform* : responsavel por guardar e modificar as informações de *Location*, *Rotation* e *Scale*

Estes são os componentes que um cubo tem por defeito, mas é possivel adicionar mais componentes quer sejam eles parte do unity o criados à parte em C#.

Por exemplo: criar um componente Health que adiciona um valor de vida ao cubo.

1.2 Player

O *First Person Player* é um *Charater Controller* que te possui varios componentes, para interagir com outros objetos, ter um sistema de vida e dano, simular gravidade entre outros. ??

Além dos Componentes tambem possui 3 childrens: a main Camera, o groundCheck, e o cilinder. Capítulo 2

1.3 Game System

Game System possui um conjunto de bases espalhadas no mapa e um *Game-Manager* também no mapa, porém só pode haver um *GameManager*.

Capítulo 2

Player

2.1 Componentes

Os Componentes programados à parte tem funções bem específicas:

- *Player Movement* : físicas, Seção 4.1;
- *Interctor* : interação com objetos ao redor, Seção 4.2;
- *Compass Changer* : orientação da bussola, Seção 4.3;
- *Health* : vida, Seção 4.4.

2.2 Children

- Main Camera: camera firs person Seção 5.1;
- Cylinder: ocupa o espaço da skin;
- GroundCheck: usado no player movement (Seção 4.1), para verificar se o player não está em queda livre.

Capítulo 3

Base

3.1 Componentes

Os Componentes programados à parte tem funções bem específicas:

- *Base Parent* : Responsavel pelo funcionamento da base Seção 4.8.

3.2 Children

- estrutura da base: ocupa o espaço da mesh e da skin;
- HitBox: trigger que detecta se o player está dentro da base.
- computer: permite ao player interagir com a base Seção 5.4;
- Door: Porta da base Seção 5.5.

Capítulo 4

Componentes

4.1 Player Movement

Responsavel pela simulação de fisicas.

4.2 Interactor

Guarda a referencia do objeto proximo com o qual se possa interagir. Possui as funções:

- *setInteract(<Tool>)* que aceita como argumento, um objeto do tipo *Tool*(Capítulo 8) e muda a referencia do objeto de interação(ex. ammo, gun, usado para interagir com trigger boxes de items, e objectos iterativos);
- *resetInteract()* que muda a referencia para null, não aceita argumentos.

4.3 Compass Changer

4.3.1 Funções

- *PointTo(<Transform>)*: Permite mudar a direção para onde a bussola aponta, aceita como argumento um *Transform*. A bussola passará a apontar para a localização do que for passado como argumento.

4.3.2 Variaveis públicas

- *Compass* : referencia da bussola do player;

4.4 Health

Responsavel pela vida.

4.4.1 Funções

- *TakeDamage(<float>)*: retira vida, na quantidade passada como argumento;
- *Die()*: destroy o objeto;

4.4.2 Variaveis publicas

- *float health* vida atual;
- *float maxhealth* vida maxima;

4.5 Mouse Look

Responsavel pela rotação da camera. Faz com que a camera seja controlada pelo rato.

4.5.1 Variaveis públicas

- (*FLOAT*) Mouse Sensivity: Sensibilidade do rato (valor base 100);
- (*TRANSFORM*) Player Body: Referencia do Player.

4.6 Weapon Switching

Responsavel pela troca das armas a para usar(ex. trocar da pistola para a shotgun).

4.6.1 Funções

- *GetActualGun()*: Retorna a arma atual a ser usada;

4.6.2 Variaveis publicas

- (*Transform*)Gun List: Weapon Holder;

4.7 Gun

Funcionamento da arma.

4.7.1 Variaveis publicas

- (*FLOAT*)Damage: Dano da arma;
- (*FLOAT*)Impact force: força de impacto da bala;
- (*FLOAT*)Range: Alcance da arma;
- (*FLOAT*)Fire Rate: Numero de balas por segundo;
- (*INT*)Ammo: Numero de balas no pente da arma;
- (*INT*)Max Ammo: Numero maximo de balas no pente da arma;

4.8 Base Parent

Responsavel por comunicar ao game master as ações do jogador.

4.8.1 Funções

- *ComputerActivated()*: Activa a base;
- *ResetComputer()*: desativa o computador permitindo que este seja reativado;
- *Unlock()*: Destranca a porta;
- *Sequence()*: Pede ao game master para avançar com o jogo;

4.8.2 Variaveis publicas

- (*STRING*)id: Nome de identificação da base;
- (*FLOAT*)health: vida da base;
- (*DOOR*)Door: porta da base;
- (*FLOAT*)R: ;

4.9 Computer

Capítulo 5

Prefabs

5.1 Main Camera

Camera do Player.

5.1.1 Componentes

- Mouse Look : rotação ad camera Seção 4.5.

5.1.2 Childrens

- Weapon Holder: sistema de armas Seção 5.2.

5.2 Weapon Holder

Responsavel por todas as armas que o player usa, e pela capacidade de trocar entre elas.

5.2.1 Componentes

- Weapon Switching : trocar de armas Seção 4.6.

5.2.2 Childrens

- Todas as armas que o player possui.

5.3 Guns

Responsavel por todas as armas que o player usa, e pela capacidade de trocar entre elas.

5.3.1 Componentes

- Gun : funcionamento da arma Seção 4.7.

5.4 Computer

Responsavel pela interção player/base.

5.4.1 Componentes

- computer : Permite ao jogador interagir com a base ??.

5.4.2 Childrens

- HitBox: Detecta se o player está proximo, para interagir.

5.5 Door

Capítulo 6

UI

Capítulo 7

IA

Capítulo 8

Interactable's Data System

8.1 Tool