

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	Componentes	4
3.1	Player Movement	4
3.2	Interactor	4
3.3	Compass Changer	4
3.3.1	Funções	4
3.3.2	Variaveis públicas	4
3.4	Health	4
3.4.1	Funcões	5
3.4.2	Variaveis publicas	5
3.5	Mouse Look	5
3.5.1	Variaveis públicas	5
3.6	Weapon Switching	5
3.6.1	Funcões	5
3.6.2	Variaveis publicas	5
3.7	Gun	5
3.7.1	Variaveis publicas	6
4	Prefabs	7
4.1	Main Camera	7
4.1.1	Componentes	7
4.1.2	Childrens	7
4.2	Weapon Holder	7
4.2.1	Componentes	7
4.2.2	Childrens	7
4.3	Guns	7

4.3.1	Componentes	8
5	UI	9
6	IA	10
7	Interactable's Data System	11
7.1	Tool	11

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. Capítulo 3

Além dos Componentes tambem possui 3 childrens: a main Camera, o groundCheck, e o cilinder. Seção 2.2

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 3.1;
- *Interactor* : interação com objetos ao redor, Seção 3.2;
- *Compass Changer* : orientação da bussola, Seção 3.3;
- *Health* : vida, Seção 3.4.

2.2 Children

- Main Camera: camera first person Seção 4.1;
- Cylinder: ocupa o espaço da skin;
- GroundCheck: usado no player movement (Seção 3.1), para verificar se o player não está em queda livre.

Capítulo 3

Componentes

3.1 Player Movement

Responsavel pela simulação de fisicas.

3.2 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 7) 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.

3.3 Compass Changer

3.3.1 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.

3.3.2 Variaveis públicas

- *Compass* : referencia da bussola do player;

3.4 Health

Responsavel pela vida.

3.4.1 Funções

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

3.4.2 Variaveis publicas

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

3.5 Mouse Look

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

3.5.1 Variaveis públicas

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

3.6 Weapon Switching

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

3.6.1 Funções

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

3.6.2 Variaveis publicas

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

3.7 Gun

Funcionamento da arma.

3.7.1 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;

Capítulo 4

Prefabs

4.1 Main Camera

Camera do Player.

4.1.1 Componentes

- Mouse Look : rotação ad camera Seção 3.5.

4.1.2 Childrens

- Weapon Holder: sistema de armas Seção 4.2.

4.2 Weapon Holder

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

4.2.1 Componentes

- Weapon Switching : trocar de armas Seção 3.6.

4.2.2 Childrens

- Todas as armas que o player possui.

4.3 Guns

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

4.3.1 Componentes

- Gun : funcionamento da arma Seção 3.7.

Capítulo 5

UI

Capítulo 6

IA

Capítulo 7

Interactable's Data System

7.1 Tool