Curso Técnico

de Programação de Jogos Digitais

Aula 45

índice

[1. Sistema de knockback 3](#_Toc170135942)

[2. Exemplo de script com knockback 3](#_Toc170135943)

[3. Knockback junto a outros efeitos 4](#_Toc170135944)

[4. Referencias 8](#_Toc170135945)

# Sistema de knockback

O sistema de knockback em um jogo refere-se à mecânica onde um personagem ou objeto é empurrado ou lançado para trás ao ser atingido por um ataque ou ao colidir com outro objeto. Esse efeito é amplamente utilizado em muitos tipos de jogos, especialmente em jogos de luta, ação e plataformas, para aumentar a dinâmica e o impacto das interações entre personagens e objetos.

Principais Elementos do Sistema de Knockback

Força de Knockback: Define a magnitude da força aplicada ao personagem ou objeto quando é atingido. Esta força pode variar dependendo do tipo de ataque, do inimigo, ou de outros fatores.

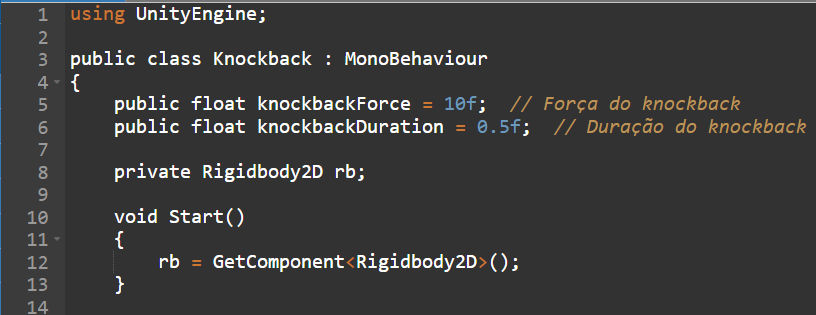
Duração do Knockback: O tempo durante o qual o efeito de knockback é aplicado. Isso pode incluir tanto o tempo em que o personagem está se movendo devido à força de knockback quanto qualquer tempo de "stun" ou incapacitação que o personagem possa sofrer.

Interação com Física: Em jogos que utilizam sistemas de física, o knockback pode ser integrado para interagir com a gravidade, colisões e outras forças físicas, proporcionando um comportamento mais realista.



A direção na qual o personagem ou objeto é empurrado. Normalmente, esta direção é oposta ao ponto de impacto do ataque.

# Exemplo de script com knockback



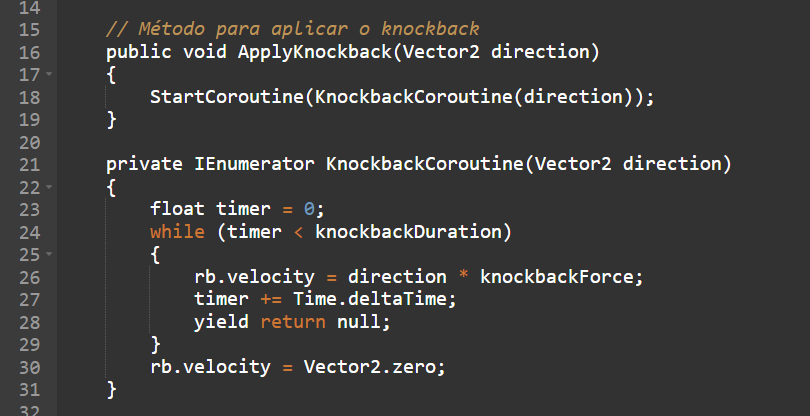
**Declaração de Variáveis:**

knockbackForce: Define a força do knockback.

knockbackDuration: Define quanto tempo o efeito de knockback deve durar.

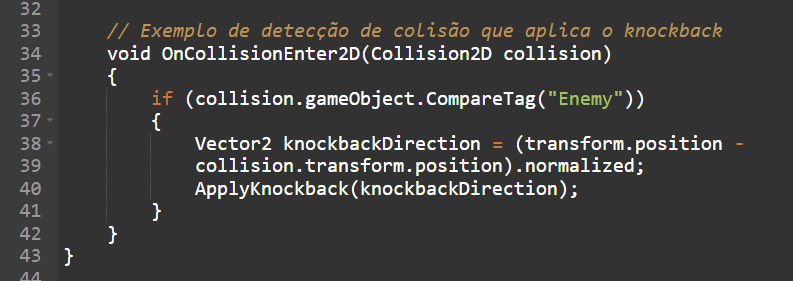
**Inicialização:**

rb: Referência ao componente Rigidbody2D do objeto, necessário para aplicar a física do knockback.



**Método ApplyKnockback:** Recebe uma direção como parâmetro e inicia uma coroutine para aplicar o knockback.

**KnockbackCoroutine:** Aplica uma força na direção especificada durante a duração do knockback, depois zera a velocidade do objeto.



**Detecção de Colisão:** Quando o objeto colide com um inimigo (um objeto com a tag "Enemy"), calcula a direção do knockback (do ponto de impacto para o personagem) e aplica o knockback.

# Knockback junto a outros efeitos

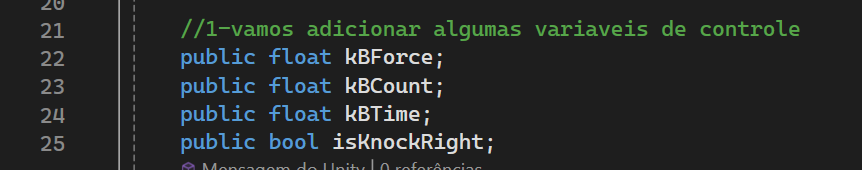
O knockback pode ser combinado com efeitos de stun, onde o personagem não pode realizar outras ações enquanto está sendo empurrado.

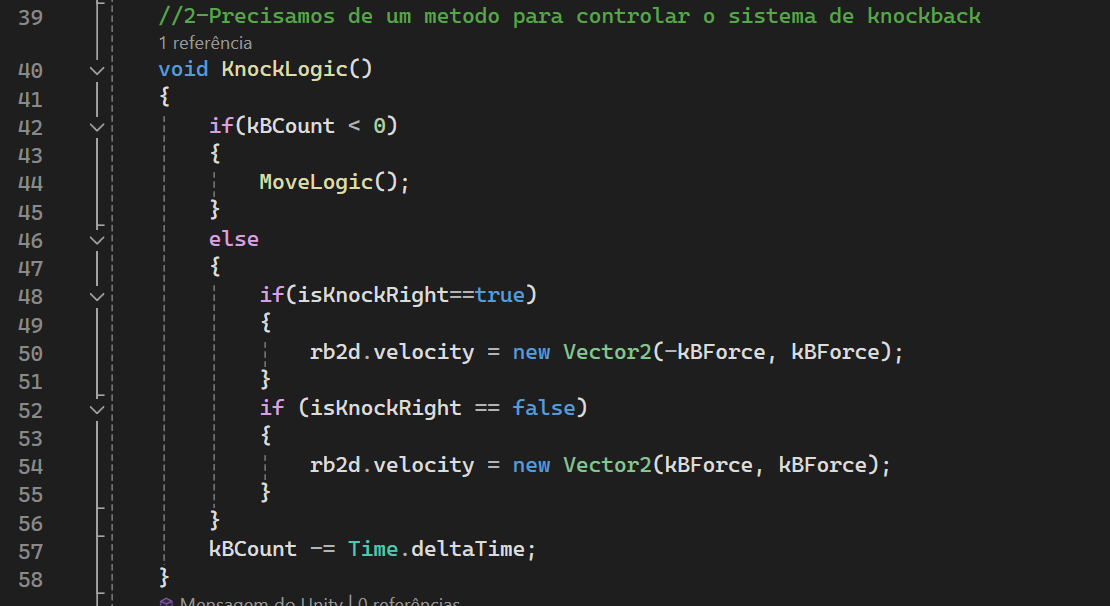
Animações podem ser sincronizado com animações com o knockback para melhorar o feedback visual.

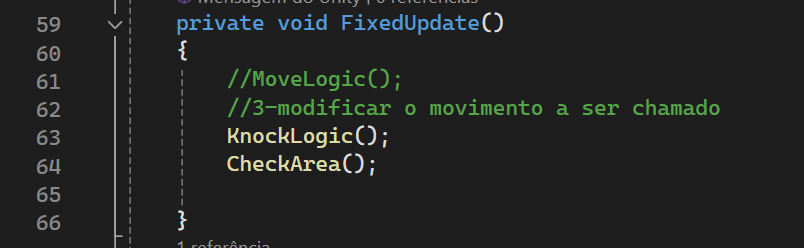
Alem de poder se ajustar a força e a direção do knockback dinamicamente com base em fatores como a força do ataque, o ângulo de impacto e as resistências do personagem.

Implementar um sistema de knockback eficaz pode adicionar profundidade e realismo ao seu jogo, tornando as interações entre personagens e objetos mais impactantes e divertidas.

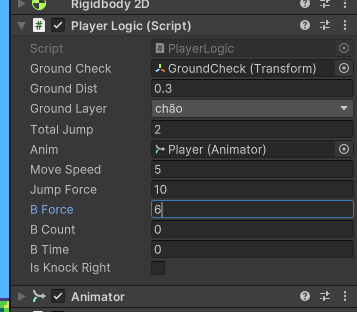
Para iniciar vamos ao script PlayerLogic.





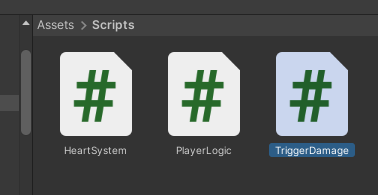


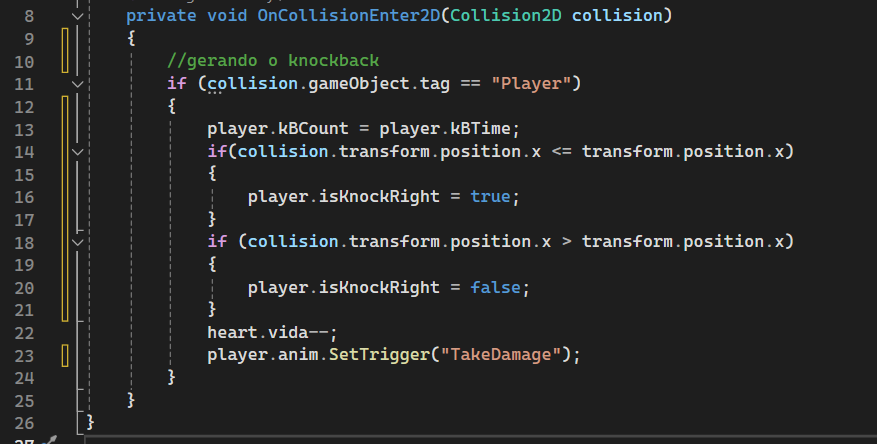
Salve e retorne a Unity. Selecione o player e no inspector no componente script player logic Altere o kBForce para 6



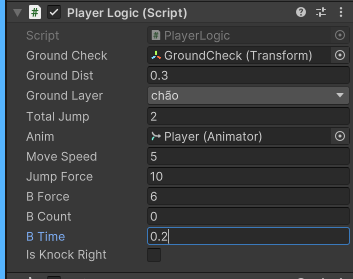
Se testarmos agora o game podemos perceber que o player leva um empurrão para direita. Vamos gerar uma lógica para que isso aconteça com o toque do inimigo.

Vamos ao script TriggerDamage

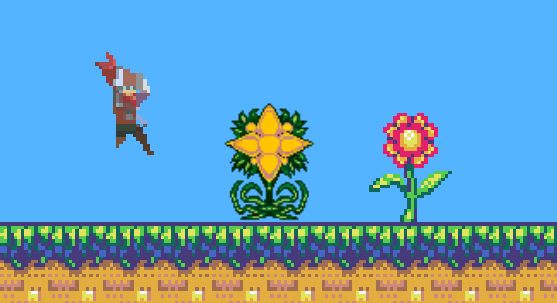




Salve e retorne para Unity e selecione o player. Em B Time coloque 0.2



O player receberá agora um empurrão toda vez que tocar o inimigo



Os códigos usados em nossas aulas estão disponíveis em nosso repositório no GitHub.



https://github.com/rildexter/pjd/tree/main/djm

# Referencias

1. Unity Technologies. (2020). **Unity User Manual.** Unity Technologies.
2. [C#, 2020] **Visual C# Developer Center**, Microsoft Docs.

Microsoft Corporation. (2020).

1. Geig, M. (2018). **Unity 2018 Game Development in 24 Hours**, Sams Teach Yourself. Sams Publishing.
2. Hocking, J. (2015). **Unity in Action: Multiplatform Game Development in C#.** Manning Publications.