

Roteiro Detalhado de Testes – Jogo Godot (Expo Tech 2025)

Testes Unitários

◇ *Teste 1 – Redução de Vida do Jogador*

- **Objetivo:** Garantir que a função `take_damage(valor)` reduza corretamente a vida.
- **Entradas:** Vida inicial = 100, dano = 20
- **Passos:**
 - Instanciar o jogador com 100 de vida
 - Aplicar `take_damage(20)`
- **Resultado esperado:** Vida final = 80
- **Aprovado se:** `player.health == 80`

◇ *Teste 2 – Reinício do Jogo*

- **Objetivo:** Verificar se a função `restart_game()` reseta o estado.
- **Entradas:** Game Over acionado
- **Passos:**
 - Acionar Game Over
 - Executar `restart_game()`
- **Resultado esperado:** Cena inicial carregada, vida = 100, pontos = 0
- **Aprovado se:** Tudo reiniciado corretamente

◇ *Teste 3 – Coleta de Itens*

- **Objetivo:** Validar que um item é adicionado ao inventário após coleta.
- **Entradas:** Jogador colide com item “Chave”
- **Passos:**
 - Colidir jogador com item
 - Verificar lista de inventário

- **Resultado esperado:** Item “Chave” presente
- **Aprovado se:** Inventário contém o item

◇ *Teste 4 – Movimentação do Personagem*

- **Objetivo:** Testar se o personagem se move com entrada do teclado.
- **Entradas:** Tecla “seta direita”
- **Passos:**
 - Pressionar tecla
 - Observar posição no eixo X
- **Resultado esperado:** X aumenta
- **Aprovado se:** Posição X mudou positivamente

◇ *Teste 5 – Cálculo de Dano no Ataque*

- **Objetivo:** Verificar que o dano do ataque corresponde ao status do personagem.
- **Entradas:** Ataque padrão
- **Passos:**
 - Executar `player.attack()`
 - Observar valor retornado
- **Resultado esperado:** Valor entre 10 e 20
- **Aprovado se:** Valor esperado retornado

Testes Funcionais (Usabilidade)

◇ *Teste 6 – Botão “Jogar” Funciona*

- **Objetivo:** Verificar se o botão inicia o jogo corretamente.
- **Passos:**
 - Clicar no botão “Jogar”
 - Observar se a cena principal é carregada
- **Resultado esperado:** Cena principal aberta
- **Aprovado se:** Cena do jogo inicia

◇ *Teste 7 – Pausar e Retomar Jogo*

- **Objetivo:** Testar funcionalidade da tecla ESC
- **Passos:**
 - Pressionar ESC
 - Verificar se o jogo pausou
 - Pressionar ESC novamente
- **Resultado esperado:** Jogo pausa e retorna
- **Aprovado se:** O estado muda corretamente

◇ *Teste 8 – Exibição da Vida*

- **Objetivo:** A UI de vida deve refletir corretamente a vida atual
- **Passos:**
 - Jogador recebe dano
 - Observar barra de vida
- **Resultado esperado:** Barra diminui proporcionalmente
- **Aprovado se:** Valor visual condiz com valor lógico

◇ *Teste 9 – Navegação por Menus*

- **Objetivo:** Garantir acessibilidade com teclado e mouse
- **Passos:**
 - Navegar com mouse
 - Navegar com teclado (setas, Enter)
- **Resultado esperado:** Menus funcionam em ambos modos
- **Aprovado se:** Navegação completa em ambas formas

◇ *Teste 10 – Tela de Vitória*

- **Objetivo:** Verificar se a cena de vitória é acionada corretamente
- **Passos:**
 - Completar o objetivo final do jogo

- **Resultado esperado:** Cena de “Vitória” aparece
- **Aprovado se:** Cena correta é carregada

Testes de Performance

◇ *Teste 11 – Estabilidade de FPS*

- **Objetivo:** Verificar se o FPS se mantém estável com 10 inimigos ativos
- **Passos:**
 - Iniciar cena com 10 inimigos
 - Monitorar FPS por 30 segundos
- **Resultado esperado:** FPS ≥ 30
- **Aprovado se:** FPS não cai abaixo de 30

◇ *Teste 12 – Tempo de Carregamento*

- **Objetivo:** Medir tempo para transição entre cenas
- **Passos:**
 - Trocar de cena (ex.: menu → gameplay)
 - Cronometrar tempo
- **Resultado esperado:** Menor que 3s
- **Aprovado se:** Tempo ≤ 3 segundos