



# TERMINAL INVADERS

Fight the bugs

## **SOBRE ESTE JOGO**

---

Combata as levas de bugs que querem destruir o seu programa!

Prepare-se para um desafio épico de sobrevivência! Em 'Terminal Invaders', você será lançado em uma arena repleta de bugs que aparecem em ondas. Sua única esperança é lutar para proteger seu programa, recolhendo armas poderosas e formando estratégias inteligentes para enfrentar essa invasão mortal. Você tem o que é preciso para se tornar o mestre da programação? Enfrente as ondas de bugs, teste suas habilidades de combate e descubra até onde você pode chegar em 'Terminal Invaders'!

- compatível com mouse e teclado.

## **PLANEJAMENTO**

---

### **Requisitos mínimos do projeto**

#### **•Arena:**

Em uma arena 2D, o jogador irá batalhar contra inimigos que apareceram em "ondas", suscetivelmente mais difíceis. A arena representa um terminal, sendo o seu tamanho dinâmico, podendo o jogador escolher entre aumenta-lo entre cada round.

#### **•Jogador:**

O jogador possui uma barra de status indicando suas habilidades, armas, vida e estamina. A vida representa a quantidade de dano que o jogador pode levar, sendo fixa por round, isso é, a cada round o jogador tem que manejá-la para não atingir a tela de "game over" tendo a opção de se curar uma vez que o round acaba, já a estamina indica se o personagem pode realizar a ação de esquivar, que o deixa invulnerável a dano por um pequeno período, recuperando-a com o passar do tempo. Além disso, esse possui armas e habilidades, que são obtidos a partir de certos rounds, começando apenas com a habilidade de esquivar e com uma espada.

#### **•Armas:**

As armas são obtidas pelo jogador conforme esse avança pelos rounds, cada uma tendo características próprias e dano que podem ser modificados entre os rounds. Cada arma abre estratégias diferentes para o jogador, sendo cada uma adequada para situações diferentes. Lista das armas:

- Espada: Causa dano ao entrar em contato com o inimigo, tem pequeno alcance e causa dano alto.
- Pistola: Dispara um projétil que causa dano médio em um inimigo.
- Shotgun: Dispara cinco projeteis de uma vez que causam dano baixo em um inimigo.
- Sniper: Dispara um projétil que causa dano alto em múltiplos inimigos de uma vez, sem desaparecer após entrar em contato com um, tendo uma frequência lenta de disparo.

#### **•Inimigos:**

Os inimigos possuem status variados entre eles, tendo cada um os tributos de vida, dano, padrão de ataque e velocidade. Eles aparecem durante as ondas em grupos, sendo a variabilidade e quantidade deles definida pelo round em que o jogador se encontra. Lista dos inimigos:

...

- Zumbi: Causa dano médio em contato com o jogador, tendo baixa velocidade e vida média.
- Morcego: Causa dano médio em contato com o jogador, tendo alta velocidade e vida baixa.
- Slime: Possui três estágios, quando leva uma certa quantidade de dano, entra no próximo estágio se separando em outros dois slimes. Possui dano e vida variados entre seus estagio sendo maiores no primeiro e diminuindo nos próximos, sendo o contrario para a sua velocidade.
- Mago: Lança feitiços no jogador que o causam dano médio e debuffs, evitando contato com esse para lançarem seus feitiços.
- Dragão: Possui um ataque lento e de longa duração que ocorre após uma animação que o torna vulnerável a dano, e depois lança uma coluna de chamas pela tela que causa alto dano no jogador.

- **Dinâmica:**

O jogador começa com uma espada e um status base de vida, estamina e habilidades, dentro de uma arena onde aparecem hordas de inimigos. Quando todos os inimigos de uma horda são derrotados pelo jogador, é exibida uma tela de escolha entre buffs aleatórios, iniciando após a escolha, um novo round. Conforme o jogador progride, mais inimigos aparecem por round, sendo novos tipos introduzidos em determinados rounds juntos de novas opções de armas. Uma vez atingido o número máximo de rounds, o jogador entra no modo arena, onde a geração das hordas se torna aleatória e acaba as escolhas de buffs.

- **Banco de Dados:**

Uma vez que o jogador tenha entrado no modo arena será registrado o tempo que esse levou para chegar nessa fase e o seu score obtido na arena, junto da quantidade de inimigos eliminados. Quando o jogador morrer, lhe será mostrado um menu para registro do nome de usuário vinculado a sua pontuação, que será ranqueado dentro de um banco de dados local.

## **Modelagem do Programa**

### **Modelo MVC:**

- **Models**

Jogador	Inimigos	Arma
<pre>+ nome: string + vida: integer + estamina: integer + velocidade: integer + armas: list(Arma) + pontuação: integer + pos_jogador: tuple(integer, integer) + skin: Image  + atacar(Arma): void + esquivar(integer): integer + trocar_arma(integer): integer + trocar_vida(integer): integer + movimentar(tuple, integer): tuple + aumentar_pontuação(integer): integer</pre>	<pre>+ vida: integer + velocidade: integer + dano: integer + skin: image + pontos: integer + pos_inimigo: tuple(integer, integer)  + movimentar(tuple, integer): tuple + atacar(integer): void</pre>	<pre>+ dano: Integer + velocidade: Integer + recarga: Integer + skin: Image + direcao: tuple(integer, integer)  + atacar(integer, integer, integer, tuple): void</pre>

• Viewers:

Tela_de_Jogo
<pre>+ exibir_arena(image); void + exibir_pontuação(integer); void + exibir_tempo(string); void + exibir_round(integer); void + exibir_jogador(image); void + exibir_status(integer, integer); void + exibir_inimigos(list(image)); void</pre>

• Controllers:

Arena	Round
<pre>+ round: list(Round) + player: Jogador + game_over: boolean + tempo_jogo: string + area_de_jogo: image + tamanho_arena: tuple(integer, integer)  + trocar_tamanho(tuple); tuple + salvar_pontuação(Jogador, string); void</pre>	<pre>+ horda: list(Inimigo) + num_round: integer + menu_buffs: list(string)  + summonar_horda(integer); list(Inimigo) + escolha_buff(integer); string</pre>

## Diagrama de Classes - UML



