

Tanques da Terra

Resumo

Este artigo apresenta Tanques da Terra, um jogo educativo desenvolvido em Pygame que integra três fases temáticas – Sol, Água e Semente – cada uma com mecânicas distintas voltadas à conscientização ambiental. O jogador controla diferentes “tanques” elementais, enfrentando desafios de eliminar inimigos, apagar incêndios e tapar buracos no solo.

1. Introdução

Em um cenário de crescente preocupação com sustentabilidade, “Tanques da Terra” propõe uma experiência lúdica que mescla ação e educação ambiental. Cada fase representa um elemento natural traduzido em mecânicas de jogo: o Tanque Sol combate inimigos “incendiários”, o Tanque Água extingue chamas, e o Tanque Semente planta raízes para restaurar o solo. O objetivo deste trabalho é descrever brevemente o desenvolvimento do jogo, suas funcionalidades principais e avaliar a receptividade inicial.

2. Mecânica de Fases

Cada fase contém 3 dificuldades (Fácil, Médio e Difícil) cada uma delas modifica os atributos da fase para facilitar ou dificultá-las. Além disso, um quiz de aprendizado é apresentado ao final de cada fase caso o jogador consiga atingir o objetivo principal.

- **Fase Sol:** Controle de um tanque armado com projéteis solares para eliminar inimigos. Objetivo de 20-30 eliminações a cada 2 mortes, novos inimigos surgem.
- **Fase Água:** Por 20 segundos, posicione o tanque para disparar jatos contínuos de água sobre “picos de fogo” que surgem aleatoriamente em um cenário em loop infinito. Segure o botão de disparo por 1 - 1.5 s em cada fogo para extingui-lo. Conta-se quantos picos são apagados. É necessário apagar 15 picos de incêndio para a vitória.
- **Fase Semente:** Em 20 s, percorra um terreno repleto de buracos e inimigos voadores. Ao pressionar X próximo a um buraco, planta-se uma raiz que gera uma plataforma, permitindo passagem, além disso inimigos voadores atrapalham o jogador. A meta é plantar o maior número de sementes possível, sendo que para a vitória é no mínimo 10.

3. Parallax e Imersão

Todas as fases utilizam parallax scrolling com múltiplas camadas PNG, criando profundidade visual. A câmera segue horizontalmente o jogador, gerando a impressão de mundo contínuo.

4. Interface, HUD e áudio

- Controles simples: A/D para mover, X para atirar/plantar, Esc para sair ou voltar ao menu.
- Barra de vida em sprites (100 %, 60 %, 20 %) no Tanque Sol e HUD de vida do inimigo, exibidos logo acima de cada entidade, para que o jogador saiba instantaneamente seu estado de saúde e o dos oponentes.
- Contadores claros no canto superior (tempo restante, inimigos faltando, picos apagados ou sementes plantadas), com fonte pixelada e contorno para legibilidade mesmo sobre cenários complexos.
- Animações de efeito (lança-chamas, jato de água, GIF de fogo e semente) que deixam explícito quando o jogador está no alcance de dano ou extinguindo um foco, ajudando a posicionar corretamente o tanque.
- Sons de fogo crepitando enquanto o tanque está próximo de um pico de fogo; spray de água em loop ao apertar X; efeitos de lança-chamas, plantio e quiz; todos com volumes ajustados para não se sobreporem. Esses alertas auditivos reforçam situações de perigo e sucesso.
- Quiz final ao cumprir metas, reforçando conceitos ambientais antes de mostrar tela de vitória/derrota.

5. Implementação Técnica

A arquitetura do Tanques da Terra está organizada em módulos:

- **Bullet (bullet.py):** representa projéteis com posição (pygame.Rect), velocidade horizontal (dx), tamanho configurável e cor. Métodos update() movem a bala, e draw() renderiza-na ajustando pelo deslocamento da câmera.
- **Camera (camera.py):** mantém um offset que centraliza horizontalmente o jogador em uma janela de 800 px, limitado entre 0 e level_width – 800. A cada update(target_rect), recalcula esse deslocamento.

Fase Água:

- **Animações GIF:** função `load_gif_frames()` extrai frames de fogo e água via PIL e converte para superfícies Pygame, permitindo animações suaves.
- **Parallax:** camadas de fundo carregadas dinamicamente e ordenadas pelo nome do arquivo, com fatores de parallax de 0.2 a 1.1.
- **Gerenciamento de spikes:** lista dinâmica de Obstacle tipo 'fogo', com timers individuais para controle de dano (variando por dificuldade) e respawn até MAX_SPIKES.
- **Sons:** efeitos de fogo e spray de água reproduzidos via `pygame.mixer.Sound`, tocando em loop enquanto o jogador está no alcance.
- **HUD e fluxo final:** fontes "pixel.ttf" para renderizar tempo restante, picos apagados e pontuação, além de telas de vitória/derrota interativas com botões de "reiniciar" e "menu".

Fase Semente:

- **Buracos e plantio:** instanciam-se `Obstacle(kind='buraco')` e, ao pressionar X dentro de um alcance (`PLANT_RANGE`), o buraco é removido e substituído por um sprite de arbusto.
- **Mundo infinito:** configura-se `Camera(level_width=109)` para scroll contínuo, removendo limite horizontal real.
- **Inimigos verticais:** classe `Enemy` define voo/flutuação entre dois níveis de Y, com mudança de direção aleatória e velocidade ajustada por dificuldade. Colisões com o tanque resultam em derrota instantânea.

Fase Sol:

- **Spawn controlado:** função `spawn_enemies(n)` posiciona grupos de inimigos fora da tela (à esquerda/direita), evitando sobreposição mínima entre X's, e escala sprites via `pygame.transform.scale()`.

- **Mecânica de chama:** inimigos geram um `flame_rect` frontal; contato contínuo causa dano a cada intervalo (`FLAME_INTERVAL`), gerenciado pelo campo `touch_timer`.
- **HUD de vida:** sprites de vida do tanque (100 %, 60 %, 20 %) mostrados acima do tanque, e HUD de inimigo carregado do diretório `assets/hud`.
- **Som de lança-chamas:** canal dedicado em `pygame.mixer.Channel` para iniciar/parar o áudio conforme presença de qualquer chama ativa.

Conclusão

O protótipo Tanques da Terra demonstra como mecânicas simples de ação — disparo, extinção e plantio — podem ser integradas a conceitos ambientais de maneira didática e divertida. A combinação de parallax scrolling, feedback visual (HUD de vida, animações de fogo/água) e sonoro (sons de chamas, spray e quiz) cria uma experiência imersiva, enquanto o mundo “infinito” respeita um limite temporal que mantém o ritmo dinâmico. Os quizzes ao final de cada fase reforçam o aprendizado, encerrando cada rodada com um breve desafio cognitivo.

Bibliografia

- itch.io. *Itch.io — The Game Distribution Site*. Disponível em: <https://itch.io>
- Pygame Community. *Pygame Documentation*. Disponível em: <https://www.pygame.org/docs/>
- PyGame Tutorials. *Invent with Python — Making Games with Pygame*. Disponível em: <https://inventwithpython.com/pygame/>
- Real Python. *Game Development with Pygame*. Disponível em: <https://realpython.com/pygame-a-primer/>
- GitHub. *Pygame Source Repository*. Disponível em: <https://github.com/pygame/pygame>
- Stack Overflow. *Perguntas marcadas [pygame]*. Disponível em: <https://stackoverflow.com/questions/tagged/pygame>