

Matheus Lopes - CJ3023494

Street Racing - Corrida de Rua

1- Ideia Geral

Street Racing é um jogo retrô de corrida desenvolvido em C++ com Raylib. O jogador controla um carro e precisa desviar de veículos inimigos. O objetivo é sobreviver o máximo de tempo possível e acumular pontos.

2- Estrutura do Projeto

- main.cpp: controle do jogo, menu, colisões, score.
- jogador.cpp/hpp: movimentação e renderização do jogador.
- inimigo.cpp/hpp: lógica dos veículos inimigos.
- gerador_inimigos.cpp/hpp: criação dinâmica dos inimigos.
- Assets: sprites, sons, fonte pixelada.

3- Funcionamento do Jogo

1. Jogador insere o nome, vê os recordes do jogo e acessa o menu principal.



Inserção do nome do jogador



Tela de Recordes do Jogo



Menu Principal

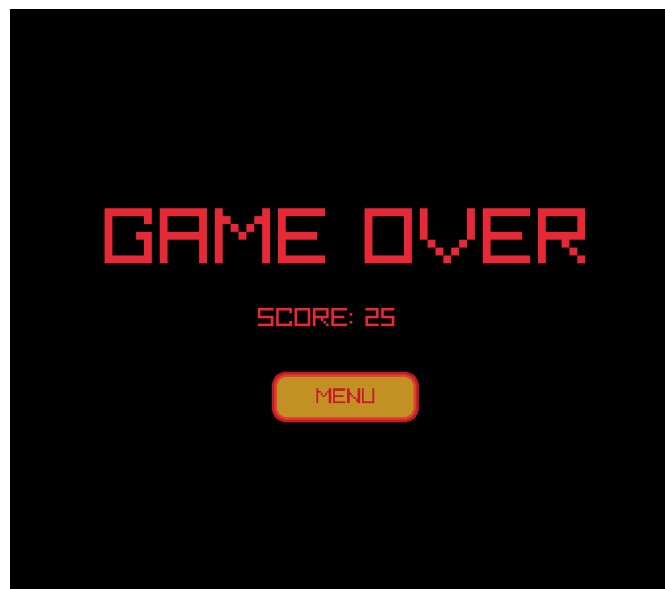
2. Ao clicar em 'Jogar', a corrida começa com o carro do jogador, o carro roxo. Sua pontuação é de acordo com o tempo vivo no jogo.



Jogo iniciado

Obs: O jogador pode se mover para os lados, para frente e para trás (setas direcionais). Os cantos da tela possuem limites, ou seja, o jogador não sai dela. Além disso o carro possui uma buzina (espaço).

3. Carros inimigos aparecem em faixas aleatórias e o jogador deve desviar deles para se manter vivo.
4. Ao colidir com um inimigo, o jogo encerra, há um barulho de batida, e exibe a tela de Game Over com o score do jogador.



5. A pontuação é salva e exibida na tela de recordes, que pode ser acessada através do botão Recordes no Menu Principal.



4- Colisões e Recordes

As colisões são detectadas com CheckCollisionRecs entre o carro do jogador e os inimigos.

Os dados de nome e pontuação são salvos em 'recordes.txt' com fstream.

Na tela de recordes são exibidos os 10 maiores scores.

5- Sons e Estética

O jogo utiliza efeitos sonoros de buzina, colisão, motor e música de fundo, que retirei do site Epidemic Sounds (sem copyright).

O jogo tem um estilo visual retrô com botões arredondados e fontes pixeladas, além de ter texturas variadas para carros e rodovia.

6- Tecnologias Usadas

- Linguagem: C++
- Biblioteca: Raylib
- STL: vector, fstream
- Organização modular com headers e cpp
- Animação 2D e detecção de colisão

7- Finalização

O jogo demonstra o conhecimento adquirido nos estudos de programação orientada a objetos, manipulação de arquivos e design de interface.

A arquitetura modular facilita a expansão do jogo para futuras melhorias.