

Trabalho: Biblioteca Raylib

Leandro Torres Mocelin

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)

Curso: Análise e Desenvolvimento de Sistemas

Disciplina: Programação Orientada a Objetos

Professor: Paulo Giovani de Faria Zeferino

Matrícula: CJ302556X

Data de Entrega: 06 de outubro de 2024

1 Introdução

Raylib é uma biblioteca open-source dedicada a simplificar o desenvolvimento de jogos e aplicações gráficas. Criada com o objetivo de oferecer uma ferramenta acessível para iniciantes, Raylib reduz a complexidade associada à programação gráfica, permitindo que até mesmo pessoas com pouca experiência possam criar suas próprias aplicações interativas. Ao longo dos anos, Raylib se destacou por sua facilidade de uso, ao mesmo tempo em que oferece recursos avançados para usuários mais experientes.

Desenvolver com Raylib é uma experiência que combina aprendizado e prática, tornando-a ideal tanto para estudantes quanto para desenvolvedores independentes que buscam uma solução ágil e descomplicada para criar desde pequenos projetos até jogos mais elaborados.

2 Onde é Utilizada

A Raylib é bastante versátil e pode ser usada em uma variedade de contextos. Originalmente focada no desenvolvimento de jogos 2D e 3D, a biblioteca também é amplamente utilizada para criar ferramentas gráficas e simuladores interativos. Além disso, sua compatibilidade com diversas plataformas, como Windows, macOS, Linux, e até mesmo dispositivos embarcados como o Raspberry Pi, a torna uma escolha popular em projetos educacionais e experimentais.

Graças à sua simplicidade e código limpo, a Raylib tem sido uma ferramenta comum em escolas e universidades ao redor do mundo, onde professores a utilizam para ensinar conceitos de computação gráfica e desenvolvimento de jogos de maneira prática e didática.

3 Quem Desenvolveu

A Raylib foi desenvolvida por Ramon Santamaria, um programador espanhol, em 2013. A motivação por trás da criação da Raylib foi a necessidade de uma ferramenta educacional acessível para ensinar programação gráfica a estudantes. Ao perceber que as bibliotecas gráficas tradicionais eram muito complexas e intimidantes para iniciantes, Santamaria desenvolveu uma solução mais amigável, mas que não deixava de ser poderosa.

Desde então, a Raylib se tornou um projeto de código aberto com uma comunidade ativa de desenvolvedores que colaboram continuamente para expandir suas funcionalidades e mantê-la atualizada com as últimas tecnologias.

4 Exemplos de Código

Um exemplo simples de código escrito com Raylib que cria uma janela e desenha um círculo em movimento é mostrado a seguir. Este exemplo básico demonstra como a biblioteca facilita a criação de gráficos interativos com poucas linhas de código.

```
[language = C++]

#include "raylib.h"

int main() {
    InitWindow(800, 600, "Exemplo Raylib");
    Vector2 ballPosition = { 400, 300 };

    while (!WindowShouldClose()) {
        if (IsKeyDown(KEY_RIGHT)) ballPosition.x += 2.0f;
        if (IsKeyDown(KEY_LEFT)) ballPosition.x -= 2.0f;

        BeginDrawing();
        ClearBackground(RAYWHITE);
        DrawCircleV(ballPosition, 50, RED);
        EndDrawing();
    }
}
```

```

    }

    CloseWindow ();
    return 0;
}

```

Nesse código, podemos ver como a Raylib lida facilmente com a criação de janelas, manipulação de entradas do usuário e a renderização de gráficos simples.

5 Exemplos de Jogos Criados com Raylib

Muitos jogos independentes e experimentais foram criados com a Raylib, justamente por sua facilidade de uso e rápida prototipação. Abaixo estão alguns exemplos de jogos desenvolvidos com a biblioteca:

- **Pong Clone:** Um jogo que recria o clássico Pong, sendo um dos primeiros projetos de quem aprende Raylib.
- **Asteroids:** Uma versão simplificada do famoso jogo de arcade "Asteroids", mostrando como a Raylib pode ser utilizada para desenvolver jogos 2D com física e colisões.

Esses projetos demonstram o quão flexível a Raylib é, permitindo que iniciantes consigam resultados rápidos, e que desenvolvedores experientes possam criar jogos completos sem a necessidade de ferramentas mais complexas.

6 Referências Bibliográficas

- Raylib Documentation: <https://www.raylib.com/>
- GitHub Raylib: <https://github.com/raysan5/raylib>
- Raylib Wiki: <https://github.com/raysan5/raylib/wiki>