

Jornada IFSP

Atire nos Obstáculos

 Jogo 2D desenvolvido em C++

 Utilizando a biblioteca **RayLib**

 Projeto de **Programação Orientada a Objetos**

 C++

 RayLib

 POO



Estudante IFSP

Pronto para a jornada!



Objetivo Principal

Aplicação prática dos conceitos de POO

Aplicar os conceitos de **Programação Orientada a Objetos** na construção de um jogo digital 2D, utilizando a linguagem **C++** e a biblioteca **RayLib**.



Funcionalidade

- ✓ Controle de personagem
- ✓ Sistema de disparo
- ✓ Obstáculos dinâmicos



Modelagem POO

- ✓ Classes e objetos
- ✓ Herança e polimorfismo
- ✓ Encapsulamento



Recursos Multimídia

- ✓ Sprites personalizados
- ✓ Efeitos sonoros
- ✓ Música de fundo



Mecânicas

- ✓ Detecção de colisões
- ✓ Sistema de vida
- ✓ Reinício do jogo

Ferramentas Utilizadas



C++

Linguagem de programação principal



RayLib

Biblioteca gráfica para jogos 2D



Code::Blocks

IDE com MinGW para desenvolvimento



ChatGPT + I Love IMG

Criação e edição de sprites

Processo de Desenvolvimento

1

Planejamento POO

Definição de classes e relacionamentos

2

Implementação Incremental

Desenvolvimento e teste por partes

3

Depuração e Testes

Execução contínua e correção de erros

4

Refinamento Estético

Aplicação de recursos gráficos e sonoros

5

Documentação

Relatório técnico e explicações

Estrutura Modular do Projeto



main.cpp

Loop principal e controle de estados



game.h

Declarações de classes e protótipos



game.cpp

Implementação das funcionalidades



Encapsulamento

Proteção de Dados

Variáveis privadas e métodos públicos controlam o acesso

Exemplo

Atributos hp, speed protegidos nas classes



Herança

Classe Base

GameObject com posição, cor e velocidade

Classes Derivadas

Player, Projectile, Obstacle



Polimorfismo

Interface Comum

Métodos Update() e Draw() virtuais

Implementação Específica

Cada classe implementa seu próprio comportamento



Hierarquia de Classes

GameObject

+ Rectangle rect
+ Color color
+ int speed
+ bool active

+ virtual Update()
+ virtual Draw()



Player

+ int hp
+ Texture2D texture

+ TakeDamage()
+ LoadTexture()

Projectile

Movimento horizontal
Detecção de colisão

+ Update()
+ Draw()

Obstacle

+ Texture2D texture
Sprites aleatórios

+ LoadTexture()
+ UnloadTexture()



Personagem Principal



Estudante IFSP

Skate voador + Estilingue

Controles:

- ➡ Setas direcionais para movimento
- ➡ Barra de espaço para atirar
- ↺ Tecla R para reiniciar

Funcionalidades



Sistema de Vida

3 pontos iniciais



Deteccão de Colisão

Projéteis vs Inimigos



Game Over

Vida zerada



Áudio Imersivo

Música + Efeitos

⚠ Obstáculos da Jornada Estudantil

Representam os desafios reais enfrentados pelos estudantes do IFSP



Ansiedade

Nervosismo
antes das
provas



Frio

Clima de
Campos do
Jordão



Sono

Cansaço
dos
estudos



Prof. Chato

Aulas
difíceis



Prova

Avaliações
complexas

📄 Comportamento dos Obstáculos

- ➡ Aparecem aleatoriamente na tela
- ➡ Movem-se da direita para a esquerda
- ❤ Causam dano ao tocar o jogador
- 🎯 Podem ser eliminados com projéteis



Arquitetura Modular



main.cpp

Ponto de entrada e controle principal



game.h

Declarações e definições



game.cpp

Implementação das funcionalidades

main.cpp

Responsabilidades:

- Inicialização da janela
- Loop principal do jogo
- Controle de entrada
- Estados de Game Over
- Limpeza de recursos

```
// Loop principal
while (!WindowShouldClose()) {
    if (!player.active) {
        // Game Over
    }
    UpdateGame();
    DrawGame();
}
```

game.h

Responsabilidades:

- Definição de classes
- Constantes globais
- Protótipos de funções
- Variáveis externas
- Includes necessários

```
class GameObject {
public:
    Rectangle rect;
    virtual void Update() = 0;
    virtual void Draw() = 0;
};
```

game.cpp

Responsabilidades:

- Implementação das classes
- Lógica de movimentação
- Detecção de colisões
- Gerenciamento de sons
- Renderização gráfica

```
void Player::Update() {
    // Limites da tela
    if (rect.x < 0) rect.x = 0;
    if (rect.y < 0) rect.y = 0;
}
```

Fluxo de Execução



Inicialização

Janela, áudio, texturas



Loop Principal

60 FPS contínuo



Entrada

Teclado e controles



Atualização

Lógica do jogo



Renderização

Desenho na tela

Campos do Jordão

Cenário inspirado na cidade do IFSP

 Casas Europeias

 Araucárias

 Montanhas

Ferramentas de Criação



ChatGPT

Geração de sprites via IA

Criação dos personagens e inimigos através de prompts específicos



I Love IMG

Redimensionamento de imagens

Ajuste das dimensões para tamanhos adequados ao jogo



Pixabay

Biblioteca de áudio gratuita

Trilha sonora e efeitos sonoros livres de direitos autorais

Assets Visuais

Personagens



Estudante IFSP
Protagonista



Ansiedade
Obstáculo



Frio
Obstáculo

Recursos Sonoros



Música de Fundo

Loop contínuo



Som de Disparo

Estilingue

Som de Impacto

Acerto no inimigo



Som de Dano

Jogador atingido

Características Técnicas

Formatos Visuais:

- PNG para sprites
- Transparência suportada
- Resolução otimizada

Formatos Sonoros:

- MP3 para música
- WAV para efeitos
- Qualidade balanceada



✓ Principais Resultados Alcançados



Jogo Funcional

- Loop principal estável
- Controles responsivos
- Sistema de reinício
- Performance otimizada



POO Aplicada

- Herança implementada
- Encapsulamento correto
- Polimorfismo funcional
- Código reutilizável



Experiência Imersiva

- Sprites personalizados
- Trilha sonora envolvente
- Efeitos sonoros
- Tema identificável

🎓 Impacto Educacional

Consolidação de Conhecimentos

Aplicação prática dos conceitos de POO em um projeto real e funcional

Desenvolvimento de Habilidades

Programação em C++, uso de bibliotecas externas e gerenciamento de projetos

Experiência Multimídia

Trabalho com recursos gráficos, sonoros e interação em tempo real

🚀 Melhorias Futuras



Sistema de Pontuação

Ranking e competição entre jogadores



Dificuldade Progressiva

Aumento gradual da velocidade dos inimigos



Múltiplos Níveis

Diferentes fases e cenários



Efeitos Visuais

Explosões e animações de impacto



Menu Interativo

Tela inicial com opções de configuração

★ Conclusão Final

O projeto "**Jornada IFSP: Atire nos Obstáculos**" foi uma experiência enriquecedora que demonstrou como os fundamentos da **Programação Orientada a Objetos** podem ser aplicados de forma eficiente em aplicações interativas e orientadas a eventos.



Técnico

Domínio de C++ e RayLib



Criativo

Design de gameplay envolvente



Social

Identificação com o público

