

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE  
SÃO PAULO**

**CRISTIANO DE ALMEIDA TOMAZ**

**DISCIPLINA: TÓPICOS AVANÇADOS I**

**PROFESSOR: PAULO GIOVANI DE FARIA ZEFERINO**

**CURSO: ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**JOGO: INSTITUTO DO TERROR**

**CAMPOS DO JORDÃO**

**2025**

## RESUMO

O presente trabalho descreve o desenvolvimento do jogo Instituto do Terror, criado como projeto da disciplina de Tópicos Avançados do curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – Câmpus Campos do Jordão. Trata-se de um jogo digital 2D elaborado em JavaScript, utilizando o framework Phaser 3 e ferramentas modernas de desenvolvimento web, com o objetivo de aplicar conceitos práticos de programação, lógica de jogos e manipulação de spritesheets.

Ambientado em um cenário distópico e pós-apocalíptico, o jogo apresenta um estudante do IFSP que realiza uma viagem no tempo em busca de restaurar a ordem do mundo. Sua missão consiste em coletar o maior número possível de “Notas 10”, ao mesmo tempo em que precisa desviar de inimigos representados por fantasmas de professores, simbolizando os desafios acadêmicos que surgem ao longo do percurso. A estética envolve elementos gráficos temáticos, como uma cidade destruída, efeitos visuais e animações criadas por meio de spritesheets.

O projeto foi desenvolvido pelo autor, incluindo concepção visual, estruturação das cenas, criação das mecânicas de movimento, colisão, pontuação, animação e lógica de progressão. A implementação demonstra a capacidade de integração entre técnicas de programação, modelagem de jogo e organização modular de código, proporcionando uma experiência completa de desenvolvimento. O resultado final evidencia a viabilidade de criar um jogo funcional e coerente narrativa e tecnicamente, atendendo aos requisitos propostos pela disciplina e contribuindo para o aprofundamento prático do aprendizado.

**Palavras-Chave:** desenvolvimento de jogos, phaser, javascript, game design, IFSP.

## ABSTRACT

This paper presents the development of Instituto do Terror, a 2D digital game created as a project for the Advanced Topics course of the Systems Analysis and Development program at the Federal Institute of Education, Science and Technology of São Paulo (IFSP) – Campos do Jordão campus. The game was built in JavaScript using the Phaser 3 framework and modern web development tools, with the aim of applying practical concepts related to programming, game logic, time-based animations, and spritesheet processing.

Set in a dystopian and post-apocalyptic future, the game follows a student from IFSP who travels through time in a mission to restore the collapsed world. The main objective is to collect as many “Grade 10” scrolls as possible while avoiding enemy entities, represented by ghost-like professors that metaphorically embody academic challenges. The visual design incorporates thematic elements such as a devastated city, animated spritesheets, and atmospheric effects that enhance player immersion.

The entire project was developed by the author, including graphic assets, scene structuring, movement mechanics, collision systems, scoring logic, and game progression. The implementation demonstrates the effective integration of programming principles, modular code organization, and essential game design techniques. The final result shows that it is feasible to create a functional and cohesive game that meets the requirements of the course and contributes significantly to the student’s practical learning experience.

**Keywords:** game development, phaser, javascript, game design, IFSP.

## SUMÁRIO

|              |                                   |           |
|--------------|-----------------------------------|-----------|
| <b>1</b>     | <b>Introdução</b>                 | <b>5</b>  |
| <b>1.1</b>   | <b>Objetivos</b>                  | <b>6</b>  |
| <b>1.2</b>   | <b>Justificativa</b>              | <b>7</b>  |
| <b>1.3</b>   | <b>Aspectos Metodológicos</b>     | <b>8</b>  |
| <b>1.4</b>   | <b>Aporte Teórico</b>             | <b>10</b> |
| <b>2</b>     | <b>Metodologia</b>                | <b>13</b> |
| <b>3</b>     | <b>Resultados Obtidos</b>         | <b>29</b> |
| <b>3.1</b>   | <b>Conclusão</b>                  | <b>30</b> |
| <b>3.1.2</b> | <b>Referências Bibliográficas</b> | <b>32</b> |

## 1 INTRODUÇÃO

O desenvolvimento de jogos digitais tem se consolidado como uma importante área dentro do campo da computação, incorporando conhecimentos de lógica de programação, matemática, estruturas de dados, design interativo, narrativa e processamento gráfico. No contexto da formação em Análise e Desenvolvimento de Sistemas, a criação de um jogo constitui uma atividade prática capaz de integrar diversos conteúdos estudados ao longo do curso, promovendo a aplicação de técnicas e tecnologias modernas em um ambiente lúdico e desafiador.

O presente trabalho descreve o processo de concepção e implementação do jogo Instituto do Terror, elaborado como projeto final da disciplina de Tópicos Avançados do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP), câmpus Campos do Jordão. O jogo foi desenvolvido com o framework Phaser 3, utilizando a linguagem JavaScript e o ambiente de desenvolvimento Vite, o que permitiu ao autor explorar recursos essenciais como criação de scenes, manipulação de spritesheets, física arcade, colisões, animações quadro a quadro, controle de game loop e gerenciamento de estados de jogo.

O enredo do jogo se passa em um cenário fictício, distópico e pós-apocalíptico, no qual um estudante do IFSP realiza uma viagem no tempo para impedir a destruição total do mundo. Para cumprir sua missão, o protagonista deve coletar “Notas 10”, elementos fundamentais para restaurar a linha temporal, enquanto enfrenta inimigos representados por espectros de professores. Essa narrativa foi concebida de forma a incorporar elementos simbólicos da vida acadêmica, transformando desafios educacionais em obstáculos lúdicos e interativos.

Além do aspecto narrativo, o projeto enfatiza a importância da organização modular do código, da estruturação das funcionalidades do jogo em diferentes cenas, e da construção de uma lógica de gameplay que integre movimento, desvio, coleta e pontuação. Dessa forma, o jogo não se limita ao entretenimento, mas se

torna uma ferramenta de aprendizado aplicada, permitindo ao estudante exercitar habilidades de desenvolvimento front-end, lógica computacional, otimização e design digital.

Assim, este relatório tem como objetivo apresentar o processo completo de desenvolvimento do jogo, abrangendo desde sua concepção visual e narrativa até sua implementação técnica e funcional. A documentação também busca evidenciar como o projeto contribuiu para o aprimoramento das competências práticas exigidas no curso, demonstrando a capacidade do aluno em integrar criatividade, raciocínio lógico e domínio de ferramentas tecnológicas no desenvolvimento de um produto digital interativo.

## 1.1 Objetivos

Desenvolver um jogo digital 2D utilizando JavaScript e o framework Phaser 3, aplicado como projeto prático da disciplina de Tópicos Avançados do curso de Análise e Desenvolvimento de Sistemas do IFSP – Câmpus Campos do Jordão, visando integrar conhecimentos de programação, lógica computacional, animação e design interativo em um produto funcional e coerente do ponto de vista técnico e narrativo. Foram ainda estabelecidos os seguintes objetivos específicos:

- Implementar um sistema completo de estruturas de cenas (scenes), incluindo carregamento de recursos (PreloadScene), menu inicial, gameplay e tela de encerramento (GameOverScene).
- Criar e configurar spritesheets, animações quadro a quadro e mecânicas de movimentação do personagem principal, garantindo fluidez na experiência do usuário.
- Desenvolver inimigos com comportamentos simples utilizando física arcade, colisões e movimentação aleatória, assegurando desafios ao jogador.

- Implementar a lógica de coleta de itens, contabilizando pontuação e exibindo informações em tempo real por meio de elementos de interface.
- Aplicar técnicas de gerenciamento de estado do jogo, incluindo temporizadores, controle de dificuldade e detecção de vitória ou derrota.
- Construir uma estética visual coerente com o enredo distópico, utilizando backgrounds temáticos, efeitos visuais e elementos gráficos e sonoros personalizados.
- Integrar práticas de organização modular do código, utilizando JavaScript moderno e ferramentas atuais de desenvolvimento (como Vite), favorecendo clareza, manutenibilidade e escalabilidade do projeto.
- Demonstrar a capacidade de unir narrativa, mecânicas e estrutura de código em um produto digital completo, reforçando competências adquiridas ao longo do curso.

## 1.2 Justificativa

O desenvolvimento de jogos digitais constitui uma atividade pedagógica altamente eficaz no contexto da formação tecnológica, pois integra simultaneamente aspectos de lógica de programação, estruturação de software, design de interfaces, uso de ferramentas modernas e criatividade aplicada. No curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de São Paulo (IFSP), tais atividades permitem ao estudante colocar em prática competências essenciais, ao mesmo tempo em que exploram soluções lúdicas e interativas para problemas computacionais.

A escolha de desenvolver o jogo Instituto do Terror fundamenta-se na oportunidade de aplicar conhecimentos adquiridos em disciplinas anteriores, empregando conceitos avançados como manipulação de spritesheets, uso de motores de jogo em JavaScript, colisões, física arcade, controle de cenas e

gerenciamento de estados. A prática favorece a consolidação do aprendizado, permitindo que o discente experimente, na prática, o ciclo completo de concepção, implementação, depuração e finalização de um produto digital interativo.

Além disso, o enredo escolhido - um cenário distópico envolvendo um aluno do IFSP em uma missão de viagem no tempo para coletar “Notas 10” enquanto enfrenta espectros de professores - funciona como uma metáfora criativa das vivências acadêmicas. Essa narrativa promove engajamento, representatividade e conexão emocional com o projeto, tornando o processo de desenvolvimento mais significativo. Ao utilizar elementos simbólicos do cotidiano estudantil, o jogo aproxima teoria e prática, reforçando a compreensão dos desafios enfrentados pelos próprios discentes.

Outro ponto relevante é que o sistema foi desenvolvido pelo próprio aluno, que possui experiência prévia com projetos autorais e atuação prática em desenvolvimento de sistemas. Isso permite validar sua capacidade técnica, sua autonomia e sua competência em integrar diferentes tecnologias e áreas do conhecimento, consolidando sua evolução acadêmica e profissional.

Assim, a elaboração deste jogo se justifica pela sua contribuição direta para o aprendizado ativo, pelo desenvolvimento de habilidades técnicas essenciais na área de computação e pelo potencial de demonstrar a capacidade do estudante em criar soluções completas, criativas e tecnicamente robustas, atendendo plenamente às expectativas da disciplina de Tópicos Avançados e aos padrões de excelência do IFSP.

### **1.3 Aspectos Metodológicos**

A metodologia adotada para o desenvolvimento do jogo Instituto do Terror fundamenta-se em práticas de engenharia de software aplicadas ao contexto de jogos digitais, utilizando ferramentas contemporâneas de desenvolvimento web e técnicas específicas de criação de jogos 2D. O processo metodológico foi dividido em etapas progressivas, que envolveram

desde a concepção narrativa e artística até a implementação técnica, programação, testes e refinamentos finais.

O projeto foi conduzido pelo próprio autor, que possui experiência prática com desenvolvimento de aplicações e atuação na área de fotografia. Essa vivência prévia contribuiu significativamente para a definição da estética visual, para a criação dos sprites personalizados e para a seleção de elementos gráficos coerentes com a proposta narrativa distópica. Além disso, o autor trabalha há sete anos com Ensaios Fotográficos, o que facilita a compreensão de composição visual, ambientação e direção artística, aspectos essenciais para a construção da identidade visual do jogo.

A implementação técnica utilizou o framework Phaser 3, devido à sua robustez, documentação acessível e ampla utilização em projetos acadêmicos e profissionais. Para gerenciamento de ambiente e execução, adotou-se o Vite, que proporciona carregamento rápido, hot-reload e organização modular do código. A linguagem utilizada foi JavaScript moderno (ES6), seguindo boas práticas de modularização por meio da estrutura de cenas (scenes) independentes: PreloadScene, MenuScene, GameScene e GameOverScene, cada uma responsável por uma etapa específica do fluxo lógico do jogo.

A criação dos spritesheets, incluindo o personagem principal e os inimigos, foi realizada a partir de editores externos, resultando em arquivos PNG otimizados e padronizados. Esses assets foram integrados ao jogo por meio de spritesheet e animações quadro a quadro, configuradas diretamente no Phaser através de funções como anims.create() e generateFrameNumbers(). O background, ilustrando a cidade destruída, também foi criado pelo autor e incorporado à lógica de renderização como tileSprite, permitindo movimento contínuo e efeito de parallax.

Para implementar as mecânicas, recorreu-se à física arcade do Phaser, especialmente para movimentação, detecção de colisões e manipulação de grupos de objetos, como notas e inimigos. Os sistemas de pontuação,

controle de tempo e progressão foram desenvolvidos com timers internos (`this.time.addEvent()`), métodos de gerenciamento de estado e tratamento de eventos dentro da classe GameScene.

A metodologia adotada também incluiu testes iterativos, nos quais cada parte implementada como movimentação, animação, colisões, transição de cenas foi avaliada individualmente e refinada. Esse processo incremental garante maior estabilidade, facilita correções e reduz o risco de erros estruturais. Todas as cenas e fontes de código foram disponibilizadas individualmente, permitindo análise e validação modular.

Por fim, buscando validar a eficácia do sistema não apenas como exercício acadêmico, mas como produto funcional, o autor testou o jogo em diferentes momentos do processo, com o objetivo de identificar oportunidades de melhoria, avaliar a jogabilidade e garantir que o produto final estivesse alinhado às expectativas de qualidade e coerência visual. O feedback obtido contribuiu para ajustes finais e para a consolidação do projeto como uma solução robusta, criativa e tecnicamente fundamentada.

## 1.4 Aporte Teórico

O desenvolvimento de jogos digitais envolve um conjunto multidisciplinar de conhecimentos que integram áreas da computação, matemática, design, narrativa e interação humano-computador. No contexto acadêmico, compreender os fundamentos teóricos que sustentam a criação de jogos é essencial para justificar as escolhas técnicas e metodológicas feitas ao longo do projeto. Nesta seção, são apresentados os principais conceitos teóricos que embasam o desenvolvimento do jogo Instituto do Terror, distribuídos entre teoria de jogos, programação orientada a eventos, engines e frameworks de jogos, lógica de sprites e animações, física simplificada, narrativa e imersão.

### **1.4.1 Jogos Digitais como Ferramenta de Aprendizado**

Segundo estudos da área de Computação e Educação, jogos digitais se destacam como ferramentas de aprendizagem ativa, pois combinam resolução de problemas, feedback imediato, experimentação e motivação intrínseca. A criação de um jogo coloca o estudante no centro do processo de desenvolvimento, exigindo que ele aplique conceitos de lógica, estrutura de dados, modelagem e criatividade. Esse caráter interdisciplinar justifica a utilização de jogos como projetos finais em disciplinas que buscam consolidar competências técnicas, como é o caso da disciplina de Tópicos Avançados do IFSP.

### **1.4.2 Arquitetura de Jogos 2D e o Paradigma de Scenes**

Motores de jogo 2D, como o Phaser 3 utilizado neste projeto, adotam uma arquitetura baseada em scenes (cenas). Cada cena representa um estado lógico do jogo, por exemplo: carregamento, menu, jogo em execução, e tela de encerramento. Tal abordagem segue princípios da engenharia de software, como modularização, encapsulamento e separação de responsabilidades.

Essa estratégia permite que:

- cada parte do jogo seja desenvolvida independentemente;
- o código seja mais organizado e tenha fácil manutenção;
- a execução siga um fluxo controlado por máquina de estados.

A divisão do jogo em PreloadScene, MenuScene, GameScene e GameOverScene reflete exatamente esse paradigma.

### **1.4.3 Programação Orientada a Eventos e Game Loop**

O desenvolvimento de jogos digitais se apoia fortemente na programação orientada a eventos, na qual determinadas ações ocorrem em resposta a interações do jogador, colisões, temporizadores ou mudanças no cenário. O Phaser implementa um game loop interno que atualiza a lógica e a renderização do jogo a cada quadro, permitindo:

- movimentação contínua;
- detecção de entradas do usuário (teclado, mouse);

- execução de animações quadro a quadro.

Esse modelo de execução contínua é essencial para jogos de ação como o Instituto do Terror.

#### **1.4.4 Sprites, Spritesheets e Animações Quadro a Quadro**

Spritesheets são estruturas amplamente utilizadas em jogos 2D devido à sua eficiência gráfica e facilidade de manipulação. Trata-se de uma única imagem contendo múltiplos quadros de animação organizados em linhas e colunas. Engines como o Phaser permitem “fatiar” esses quadros e gerar animações fluídas utilizando técnicas de keyframe-based animation.

O uso de spritesheets no jogo — tanto para o personagem principal quanto para os inimigos — está alinhado às boas práticas de otimização gráfica, pois:

- reduz o número de requisições de imagem;
- melhora o desempenho da renderização;
- facilita a criação de movimentos contínuos como voo, caminhada e ataques.

#### **1.4.5 Física Arcade e Colisões em Jogos 2D**

A física arcade, disponibilizada pelo Phaser, utiliza modelos simplificados para simular movimento, aceleração, colisões e respostas físicas. Embora não siga padrões realistas, ela é ideal para jogos 2D rápidos, pois oferece:

- baixo custo computacional;
- funções prontas para detecção de colisões;
- controle de velocidade, aceleração e direção;
- grupos de objetos com comportamento próprio.
- Isso fundamenta teoricamente a implementação dos inimigos, do movimento do player e das mecânicas de coleta de itens no jogo.

#### **1.4.6 Câmera, Background e Imersão Visual**

O uso de tileSprites para o background cria a ilusão de movimento contínuo,

técnica comum em jogos 2D side-scroll e top-down. Essa estratégia de parallax adiciona profundidade e reforça o ambiente narrativo do jogo. No caso de Instituto do Terror, o background mostra uma cidade destruída, coerente com a ambientação distópica proposta, reforçando o papel da estética como parte fundamental da experiência do jogador.

#### **1.4.7 Narrativa e Storytelling Aplicado ao Game Design**

A teoria de narrativa aplicada a jogos (Game Narrative) estabelece que a ambientação, o enredo e os elementos simbólicos contribuem para a imersão e motivação do jogador. Transformar professores em fantasmas e transformar a coleta de “Notas 10” em objetivo central reforça a metáfora acadêmica do IFSP.

Esse tipo de narrativa simbólica:

- contextualiza o gameplay;
- cria identidade visual e temática;
- reforça a conexão emocional com o usuário;
- aumenta a motivação e a compreensão do propósito do jogo.

#### **1.4.8 Integração entre Arte, Código e Jogabilidade**

O desenvolvimento de jogos exige a integração entre arte digital (sprites, cenários), lógica de programação (movimento, colisões, animações), estrutura de software (scenes, módulos) e design de mecânicas (desafios, ritmo, dificuldade). Essa integração é fundamentada em princípios teóricos de game design, que sugerem que elementos mecânicos, estéticos e narrativos devem atuar de maneira coordenada.

## **2 Metodologia**

A metodologia adotada para o desenvolvimento do jogo Instituto do Terror seguiu um conjunto de etapas organizadas e fundamentadas nas boas práticas de engenharia de software, desenvolvimento de jogos 2D e uso de ferramentas tecnológicas contemporâneas. O processo contemplou desde a fase de concepção

visual e narrativa até as estratégias de implementação, testes incrementais e validação funcional. As etapas metodológicas foram estruturadas da seguinte forma:

## 2.1 Planejamento e Concepção Narrativa

A etapa inicial do projeto envolveu a definição do enredo, da temática e da identidade visual do jogo, elementos fundamentais para estruturar todas as demais fases de desenvolvimento. O jogo Instituto do Terror foi idealizado tomando como inspiração um projeto anterior do próprio aluno, o jogo Jornada IFSP, no qual o protagonista utilizava um skate voador para superar desafios acadêmicos. Essa referência serviu como base criativa, não apenas pela familiaridade do autor com a mecânica, mas também pela afinidade do público com o estilo visual e a dinâmica fluida proporcionada pelo veículo.

No entanto, o presente projeto foi desenvolvido sob a diretriz temática Fantasia, estabelecida pela disciplina de Tópicos Avançados. Assim, enquanto Jornada IFSP possuía uma abordagem mais aventureira e cotidiana, o Instituto do Terror buscou reinterpretar o uso do skate voador dentro de um universo fantástico, distópico e pós-apocalíptico. O protagonista, agora um viajante do tempo, enfrenta uma missão crucial: coletar o maior número possível de Notas 10, fragmentos temporais essenciais para restaurar a realidade devastada.

O cenário é ambientado em uma versão destruída do IFSP – Campus Campos do Jordão, simbolizando o colapso de múltiplas linhas temporais. Os inimigos aparecem como fantasmas de professores, representações metafóricas dos desafios, pressões e exigências acadêmicas que o estudante enfrenta ao longo de sua formação. A presença desses espectros reforça o clima de fantasia sombria, ao mesmo tempo em que estabelece uma conexão simbólica com o cotidiano escolar.

Essa concepção narrativa — unindo distopia, fantasia, viagem no tempo e metáforas acadêmicas — orientou toda a estética visual do jogo, influenciando a escolha de cores, sprites, animações e trilhas sonoras. Também serviu como eixo estruturante para as mecânicas de gameplay e para o aspecto emocional da experiência, garantindo coerência temática e identidade própria ao projeto.

## 2.2 Criação e Tratamento dos Assets Visuais

A produção dos recursos visuais utilizados no jogo Instituto do Terror envolveu um conjunto diversificado de ferramentas especializadas em edição, geração e manipulação gráfica. Os principais elementos visuais — incluindo sprites, spritesheets, inimigos, notas, background e capa do jogo — foram desenvolvidos combinando tecnologias de inteligência artificial com editores avançados de imagens, de modo a garantir um resultado visual coerente com a narrativa distópica e com as exigências técnicas do framework Phaser 3.

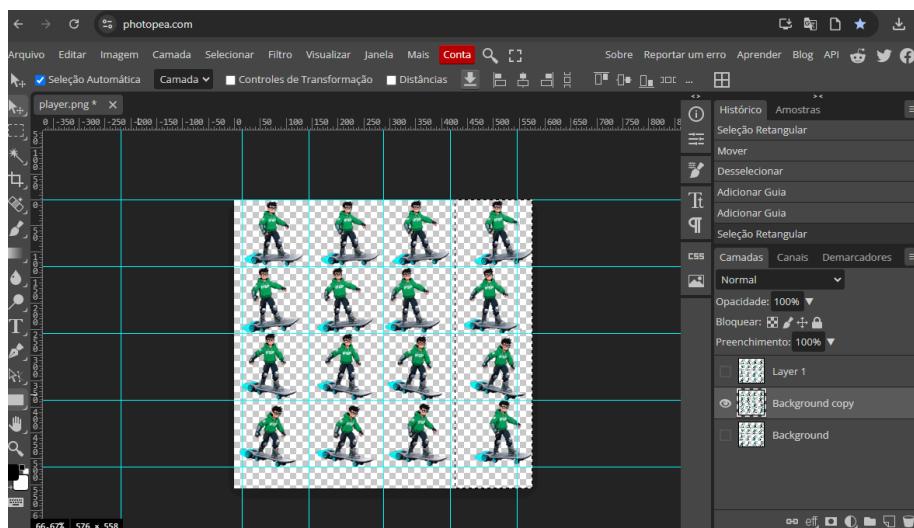
Uma parte significativa dos spritesheets do jogador (player) e dos inimigos (enemy) foi inicialmente gerada utilizando a ferramenta Ludo AI (<https://ludo.ai/>), que permite criar personagens animados com inspiração em jogos 2D. Entretanto, os spritesheets produzidos pela plataforma não estavam imediatamente prontos para uso dentro da engine. Foi necessário realizar um trabalho detalhado de reposicionamento, alinhamento e ajuste quadro a quadro para assegurar a fluidez das animações. Esse processo foi conduzido no editor Photopea (<https://www.photopea.com/>), que oferece um conjunto robusto de funcionalidades semelhantes ao Adobe Photoshop.

Nesse ambiente, cada frame dos spritesheets foi cuidadosamente realinhado para garantir consistência visual entre as poses do personagem e dos inimigos. A padronização dos frames, tanto em proporções quanto em espaçamentos, foi essencial para viabilizar animações corretas dentro do Phaser, especialmente para movimentos contínuos, como o voo do protagonista e o deslocamento dos inimigos.

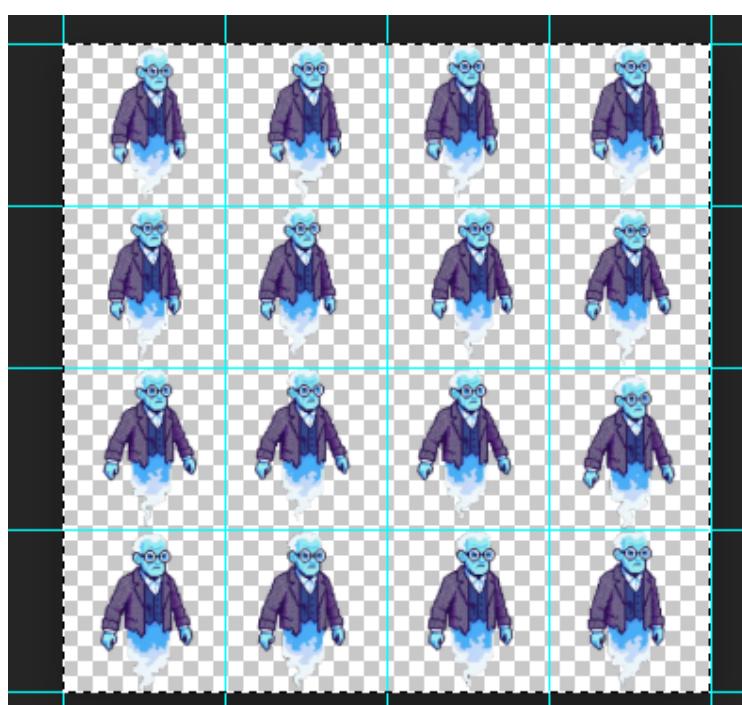
As demais imagens utilizadas no jogo, como o background distópico, a capa do jogo, as Notas 10 e elementos auxiliares, foram criadas com apoio de ferramentas de IA generativa, incluindo o Google Gemini e o ChatGPT. Essas plataformas permitiram a experimentação de composições visuais alinhadas ao ambiente pós-apocalíptico desejado, bem como a elaboração de variações estilísticas rapidamente. Após a geração inicial, todo o material visual foi refinado e redimensionado no Photopea, ajustando resoluções, removendo artefatos e

garantindo transparências adequadas ao formato PNG.

O tratamento final das imagens possibilitou sua integração precisa ao pipeline do Phaser 3, possibilitando a criação de backgrounds utilizáveis como tileSprite, spritesheets padronizados para animações quadro a quadro e ícones com resolução adequada para HUD e interface. Esse cuidado permitiu que o jogo exibisse boa coerência estética, fluidez visual e compatibilidade técnica com o ambiente web.



(captura de tela)



(captura de tela)

### 2.3 Implementação Técnica com Phaser 3

A fase de implementação utilizou o framework Phaser 3, que oferece recursos completos para jogos 2D, incluindo:

- Gerenciamento de scenes
- Sistema de física arcade
- Renderização gráfica em WebGL e Canvas
- Manipulação de spritesheets e animações
- Eventos de entrada do teclado
- Criação de grupos, colisões e sobreposições

A estrutura do jogo foi dividida em quatro cenas principais:

#### 2.3.1 PreloadScene

Responsável por carregar todos os assets e registrar a animação principal do jogador. Aqui, spritesheets, background e notas são processados para garantir que estarão disponíveis nas demais cenas.

#### 2.3.2 MenuScene

Implementa a tela inicial, onde o jogador visualiza o título, instruções e o background. A cena aguarda a entrada do usuário (Enter ou clique) para iniciar o jogo.

#### 2.3.3 GameScene

É o núcleo da implementação, contendo:

- Sistema de movimentação do jogador;
- Aplicação da física arcade;
- Mecânica de coleta de "Notas 10";
- Inimigos com movimento aleatório;
- HUD com pontuação e tempo restante;
- Parallax do cenário via tileSprite;

- Detecção de colisões e sobreposições;
- Lógica de vitória e derrota.

#### **2.3.4 GameOverScene**

Exibe o resultado final da partida (vitória ou derrota), permitindo retorno ao menu principal ou reinício da fase.

### **2.4 Desenvolvimento da Lógica de Gameplay**

Durante a implementação, o jogador recebeu:

- Velocidade controlada por teclado (setas ou WASD)
- Flip horizontal (setFlipX) baseado na direção
- Animação contínua de voo (player\_fly)
- Colisão com inimigos
- Coleta de itens com pontuação dinâmica
- Sistema de temporizador para término da fase
- Os inimigos foram configurados como entidades com:
- Velocidade aleatória
- Movimento com bounce
- Colisão entre si
- Deadzone para aleatoriedade

### **2.5 Modelagem e Organização do Código**

Para manter clareza e modularidade, o código foi organizado de forma que cada scene tivesse responsabilidades bem definidas. Foram utilizados:

- Classes ES6;
- Modularização nativa (import e export);
- Instâncias independentes de sprites, grupos e HUD;
- Funções de callback separadas para coleta e dano;
- Controle de estado com variáveis internas (pontuação, tempo).

## 2.6 Tratamento de Áudio

Para compor a ambientação sonora:

- As músicas foram obtidas a partir da biblioteca Pixabay;
- Realizou-se o corte e ajuste de áudio com Clideo (<https://clideo.com/pt/cut-audio>).

## 2.7 Testes, Iteração e Validação

O método de desenvolvimento adotou caráter incremental, onde cada recurso implementado era testado isoladamente:

- Testes de colisão;
- Testes de movimento e resposta ao teclado;
- Testes de carregamento de assets;
- Testes de animação e fluidez do jogador;
- Testes de progressão e temporizador.

Erros encontrados durante o processo, como caminhos incorretos, tamanhos inconsistentes de spritesheets e ausência de texturas, foram identificados e corrigidos, garantindo estabilidade ao jogo final.

## 2.8 Consolidação e Preparação do Produto Final

Após a implementação completa, foram realizados ajustes estéticos, reorganização de arquivos, refinamento da velocidade dos inimigos, correção de flips, calibragem de tempo e verificação visual do cenário. Finalizado o processo, o jogo mostrou-se funcional, estável e alinhado aos objetivos propostos na disciplina.

## 2.9 Ferramentas Utilizadas

O desenvolvimento do jogo Instituto do Terror demandou o uso de um conjunto diversificado de ferramentas digitais, abrangendo ambientes de programação, frameworks especializados para jogos, plataformas de edição gráfica,

geradores de imagens por IA e softwares de tratamento de áudio. A seguir, são apresentadas as principais ferramentas utilizadas ao longo do projeto, bem como suas contribuições específicas para cada etapa da produção.

### 2.9.1 Ferramentas de Desenvolvimento e Execução

- Phaser 3: O framework Phaser 3 foi a base tecnológica para implementação do jogo. Trata-se de uma biblioteca JavaScript consolidada no desenvolvimento de jogos 2D, oferecendo recursos essenciais como física arcade, gerenciamento de scenes, animações quadro a quadro, manipulação de spritesheets, tratamento de colisões e renderização acelerada por WebGL. Sua robustez e documentação rica permitiram o desenvolvimento de um jogo fluido e modular.
- Vite: O ambiente Vite foi utilizado como servidor de desenvolvimento e empacotador do projeto. Sua velocidade e suporte nativo a módulos ES6 aceleraram significativamente os testes, a recarga automática e a estruturação dos arquivos do jogo.
- JavaScript (ES6): Toda a lógica de programação — controles de movimento, animações, colisões, HUD, pontuação, tempo e lógica de cenas — foi implementada em JavaScript moderno, utilizando sintaxe modular com import e export, classes ES6 e organização estrutural compatível com aplicações web contemporâneas.
- Node.js e npm: O Node.js e o npm foram essenciais para gerenciamento de dependências, instalação do Phaser, execução do Vite e manutenção da arquitetura modular do projeto.

### 2.9.2 Ferramentas de Edição e Criação Visual

- Ludo AI (<https://ludo.ai/>): Os spritesheets iniciais — tanto do personagem principal quanto dos inimigos — foram criados a partir da plataforma Ludo AI, que gera animações e personagens para jogos 2D.

No entanto, as imagens produzidas não estavam prontas para uso imediato no Phaser. Os quadros estavam desalinhados ou apresentavam variações indesejadas de posição, exigindo refinamento manual.

- Photopea (<https://www.photopea.com/>): O editor online Photopea, semelhante ao Adobe Photoshop, desempenhou papel central no polimento gráfico do projeto. Ele foi utilizado para: alinhar e reajustar cada quadro dos spritesheets criados pela Ludo AI, corrigir espaçamentos irregulares; padronizar dimensões, tratar transparências, redimensionar imagens, compor arquivos PNG otimizados. O ajuste preciso desses spritesheets foi essencial para garantir o efeito de animação fluida dentro do Phaser, resultando em movimentos corretos de voo e deslocamento dos personagens.
- Google Gemini e ChatGPT (<https://gemini.google.com/>) e (<https://chatgpt.com/>): A criação de artes complementares — como o background da cidade destruída, a capa do jogo, as Notas 10 e variações adicionais — foi realizada com ajuda de ferramentas de IA generativa, especialmente Google Gemini e ChatGPT. Esses modelos forneceram conceitos visuais ricos, variações estilísticas e elementos artísticos inspirados na estética pós-apocalíptica e fantástica do jogo. Posteriormente, todos os materiais gerados foram refinados no Photopea para ajustar tamanho, resolução, proporção e cortes.

### 2.9.3 Ferramentas de Áudio

- Pixabay - biblioteca de músicas (<https://pixabay.com/>): As músicas e efeitos sonoros utilizados no jogo foram obtidos através da biblioteca gratuita do site Pixabay, que oferece conteúdos livres de direitos autorais e adequados ao uso acadêmico. As trilhas adicionaram atmosfera ao gameplay, reforçando o clima de fantasia sombria.
- Clideo - corte e adaptação de áudio (<https://clideo.com/pt/cut-audio>): Para ajustar a duração e intensidade das trilhas de fundo, utilizou-se o Clideo, uma ferramenta online para edição e corte de áudio. O recurso permitiu adaptar músicas selecionadas ao tempo do jogo.

### 2.9.3 Ferramentas Auxiliares

- Visual Studio Code (VS Code): O VS Code foi o editor principal do projeto. Através dele foram realizados todos os ajustes de código, organização de diretórios, testes diretos, execução do Vite e utilização de extensões úteis, como formatação automática e integração com JavaScript moderno.
- Navegador Web (Chrome / Edge): Os testes foram executados diretamente no navegador, uma vez que o Phaser é compatível com WebGL e Canvas. Isso permitiu depuração rápida via ferramentas de desenvolvedor (DevTools).

## 2.10 Personagens e imagens de fundo

### 2.10.1 Player



(Player- Personagem principal)

O personagem principal do jogo Instituto do Terror representa um estudante do IFSP Campus Campos do Jordão que, em um futuro distópico e pós-apocalíptico, assume o papel de viajante do tempo responsável por restaurar as linhas temporais

corrompidas. Utilizando seu característico skate voador — um elemento já presente em projetos anteriores do próprio autor, como o jogo Jornada IFSP. O protagonista desloca-se com agilidade pelo cenário arruinado, reforçando a estética fantástica e futurista definida pela temática da disciplina.

Do ponto de vista técnico e artístico, o personagem foi construído a partir de um spritesheet gerado inicialmente pela plataforma Ludo AI, contendo múltiplos quadros de animação distribuídos em linhas e colunas. No entanto, o material exportado não estava pronto para uso direto no framework Phaser 3. Foi necessário realizar um trabalho meticoloso de edição, alinhamento e padronização de cada frame no editor Photopea, garantindo uniformidade nos tamanhos, centragem adequada e espaçamento regular entre os quadros.

Esse cuidado foi fundamental para possibilitar a criação de uma animação fluida do movimento de voo, essencial para reforçar a identidade do personagem e a sensação de deslocamento constante através do tempo. A animação final — integrada ao jogo por meio de frame slicing e comandos como `generateFrameNumbers()` e `anims.create()` — permitiu ao protagonista exibir um comportamento visual coerente e natural, transmitindo dinamismo e leveza ao gameplay.

Narrativamente, o personagem simboliza a resistência acadêmica em meio ao caos temporal: cabe a ele coletar o maior número possível de Notas 10, fragmentos responsáveis por restabelecer a ordem, ao mesmo tempo em que precisa escapar dos fantasmas de professores, representações metafóricas das dificuldades enfrentadas no ambiente educacional. Seu skate voador funciona tanto como recurso estético quanto mecânico, reforçando a fantasia tecnológica do universo do jogo e oferecendo uma movimentação característica, rápida e responsiva.

Assim, o personagem principal do jogo integra de forma equilibrada elementos narrativos, visuais e mecânicos, resultando em uma figura iconográfica que representa o espírito do projeto: criatividade, resiliência e domínio das tecnologias aplicadas em jogos digitais.

### 2.10.2 Os Fantasmas dos Professores



(Fantasmas do professores - enemy)

Os inimigos presentes no jogo Instituto do Terror foram concebidos como fantasmas de professores, representações simbólicas dos desafios e pressões acadêmicas enfrentadas pelos estudantes ao longo de sua trajetória. Em vez de adotar inimigos tradicionais ou criaturas fantásticas genéricas, optou-se por uma abordagem narrativa mais contextualizada, integrando elementos do cotidiano escolar ao universo distópico do jogo. Essa escolha reforça o caráter fantasioso do projeto, conectando a experiência lúdica ao ambiente institucional do IFSP.

Visual e artisticamente, os inimigos foram criados a partir de um spritesheet gerado com apoio de inteligência artificial, especificamente por meio da plataforma Ludo AI, a mesma utilizada na criação do personagem principal. O arquivo bruto continha múltiplos frames distribuídos em grade, representando diferentes poses e movimentos dos fantasmas. No entanto, assim como ocorreu com o spritesheet do player, o material exportado pela Ludo AI não estava imediatamente pronto para uso dentro do framework Phaser 3.

Foi necessário um processo minucioso de edição manual no editor Photopea, onde cada quadro da animação foi realinhado, corrigido e padronizado para garantir consistência visual e compatibilidade com o tamanho final do spritesheet. Esse

refinamento incluiu:

- ajuste de dimensões do quadro completo, margens e espaçamentos;
- correção do posicionamento vertical e horizontal dos frames;
- adequação da área útil de cada quadro para evitar recortes indesejados;
- padronização das dimensões para permitir animações estáveis.

Esse trabalho foi fundamental para assegurar que a animação tivesse fluidez e que os inimigos apresentassem um comportamento visual coerente quando movimentados no ambiente do jogo.

Do ponto de vista técnico, os inimigos foram integrados ao jogo como entidades dotadas de movimento aleatório utilizando a física arcade do Phaser 3. Cada fantasma recebe velocidade variada e colisão com as bordas da tela, além de colisão entre si, permitindo que o comportamento emergente crie situações imprevisíveis durante o gameplay. Embora não possuam animações complexas, seus deslocamentos erráticos combinados com o design visual espectral reforçam a ambientação de fantasia sombria e pós-apocalíptica do cenário.

Narrativamente, os fantasmas dos professores funcionam como obstáculos simbólicos: representam a cobrança, a exigência e a dificuldade que os estudantes costumam associar ao ambiente acadêmico. A presença desses inimigos no cenário destruído reforça o clima de tensão e urgência da narrativa, onde o protagonista — um viajante do tempo — precisa escapar dessas entidades enquanto coleta as Notas 10 que restaurarão a ordem temporal.

Assim, os inimigos cumprem simultaneamente funções estéticas, técnicas e narrativas, contribuindo diretamente para a ambientação do jogo e para a experiência de desafio enfrentada pelo jogador, consolidando-se como parte essencial da identidade visual e mecânica de Instituto do Terror.

### 2.10.3 Capa do Jogo e Background



(Capa do Jogo)



(Tela de fundo - Background do Jogo)

As imagens de background e capa (cover) desempenham um papel fundamental na construção da identidade visual e narrativa do jogo Instituto do

Terror, funcionando como elementos introdutórios para a estética distópica, fantástica e pós-apocalíptica que caracteriza o projeto. Ambas as imagens foram desenvolvidas utilizando ferramentas de inteligência artificial gerativa, e posteriormente refinadas para atender às exigências técnicas do motor gráfico Phaser 3.

A criação inicial dos conceitos visuais do cenário destruído e da capa do jogo foi realizada com apoio dos modelos generativos Google Gemini e ChatGPT, que permitiram produzir composições artísticas coerentes com o storytelling do jogo: uma versão devastada da cidade de Campos do Jordão, refletindo o colapso das linhas temporais que o protagonista, um viajante do tempo, deve restaurar. Esses elementos visuais reforçam o clima de fantasia sombria, contribuindo para a imersão do jogador desde a primeira interação com o jogo.

No entanto, as imagens geradas por IA não estavam imediatamente prontas para uso. Ambas passaram por um processo de edição, adequação e otimização no editor Photopea, que possibilitou:

- ajuste de resolução e proporções para o modo landscape utilizado no jogo;
- corte adequado para evitar distorções ao preencher a tela inteira;
- correção de bordas, remoção de artefatos e aprimoramento da nitidez;
- padronização da paleta de cores para harmonizar com o restante dos assets;
- preparação do background para funcionar como elemento estático ou tileSprite.

A imagem de background foi posicionada de maneira centralizada na GameScene, ocupando a totalidade do espaço de renderização para formar o cenário base em que o jogador se movimenta. Já a imagem de cover foi utilizada na MenuScene, desempenhando função estética e informativa ao introduzir o nome do jogo e reforçar sua temática.

O tratamento rigoroso dessas imagens garantiu compatibilidade com o

framework Phaser 3, mantendo desempenho adequado e uma apresentação visual coerente com os demais elementos gráficos do projeto. Em conjunto, background e cover contribuem significativamente para a experiência narrativa do jogo, funcionando como o primeiro contato sensorial do usuário com o universo de Instituto do Terror.

#### **2.10.4 Notas 10: ítems coletáveis**



(Nota 10)

As Notas, representadas visualmente como Notas 10, constituem o principal item coletável do jogo Instituto do Terror e desempenham um papel central tanto na narrativa quanto na mecânica de progressão. No enredo, as notas simbolizam fragmentos essenciais da linha temporal que o protagonista — um viajante do tempo — precisa recuperar para restaurar o equilíbrio do mundo pós-apocalíptico.

As imagens das notas foram geradas com auxílio de ferramentas de inteligência artificial, e posteriormente ajustadas no editor Photopea para adequação de tamanho, nitidez e transparência, garantindo a compatibilidade com o sistema de renderização do Phaser 3. Cada nota é posicionada aleatoriamente no cenário durante o início da fase, incentivando movimento constante e exploração por parte do jogador.

Do ponto de vista mecânico, quando o player colide com uma nota, essa é imediatamente coletada e removida da tela, incrementando a pontuação do jogador. A coleta de todas as notas disponíveis na fase representa um dos critérios de vitória, funcionando como objetivo principal do gameplay. Assim, as notas atuam como elementos motivadores, reforçando a dinâmica de progressão e mantendo o jogador

engajado durante toda a partida.

### 3 Resultados Obtidos

O desenvolvimento do jogo Instituto do Terror permitiu alcançar resultados significativos tanto em termos técnicos quanto pedagógicos, demonstrando a consolidação das competências previstas na disciplina de Tópicos Avançados do curso de Análise e Desenvolvimento de Sistemas do IFSP. O produto final apresenta um conjunto coerente de mecânicas, estética visual consistente, estabilidade funcional e integração eficiente entre os recursos programáticos e artísticos empregados ao longo do projeto.

Do ponto de vista técnico, o jogo atingiu plenamente os requisitos estabelecidos. As quatro scenes principais — PreloadScene, MenuScene, GameScene e GameOverScene — foram implementadas com sucesso, compondo uma estrutura modular organizada que facilita a leitura e manutenção do código. Os arquivos enviados pelo aluno, como GameScene.js, MenuScene.js, PreloadScene.js e GameOverScene.js, demonstram organização funcional, uso correto de classes ES6 e integração adequada com o framework Phaser 3. A lógica de animação do player, baseada em um spritesheet de 16 quadros, foi corretamente processada e aplicada, resultando em um movimento de voo fluido e responsivo.

A mecânica principal do jogo — coletar as Notas 10 enquanto o jogador desvia dos fantasmas dos professores — foi implementada de maneira eficiente e funcional. Os itens coletáveis são distribuídos aleatoriamente no cenário, e seu sistema de detecção, remoção e atualização de pontuação opera sem falhas. A detecção de colisões com os inimigos também foi configurada de forma adequada, acionando corretamente as condições de derrota. O timer global e o sistema de HUD, que apresentam tempo restante e pontuação, funcionam de forma sincronizada, contribuindo para a clareza da dinâmica de jogo.

Os inimigos, representados pelos fantasmas dos professores, demonstram movimentação autônoma com velocidades aleatórias e colisões entre si, o que promove um comportamento imprevisível e aumenta o desafio para o jogador. A

física arcade do Phaser foi aplicada corretamente, garantindo movimentação suave e coerente com a proposta do jogo. O protagonista, por sua vez, responde imediatamente aos comandos de teclado e altera sua orientação horizontal conforme a direção do movimento, reforçando a sensação de controle fluido.

No campo visual, os resultados também foram expressivos. As imagens de background, notas, inimigos e capa do jogo foram geradas via IA (Gemini e ChatGPT) e posteriormente refinadas no Photopea, resultando em elementos visuais de boa qualidade estética e perfeitamente adequados às dimensões e necessidades do jogo. Os spritesheets criados inicialmente pela plataforma Ludo AI foram devidamente ajustados manualmente, o que garantiu consistência nos quadros da animação e evitou distorções durante a execução no Phaser.

Em relação à experiência do usuário, o jogo apresenta uma tela inicial funcional, com instruções claras para iniciar a partida, e uma tela de game over coerente com o tema proposto, oferecendo feedback adequado sobre vitória ou derrota e incentivando o reinício da fase. A performance geral durante os testes mostrou-se estável, sem travamentos, erros críticos ou inconsistências de renderização.

Por fim, o projeto demonstrou que é possível integrar ferramentas modernas de edição gráfica, IA generativa, bibliotecas de desenvolvimento de jogos e boas práticas de programação para produzir um jogo funcional, esteticamente consistente e alinhado aos objetivos acadêmicos. O resultado evidencia não apenas o domínio técnico do estudante, mas também sua capacidade criativa e sua competência em transformar uma proposta temática em um produto final jogável e bem estruturado.

#### **4. Conclusão**

O desenvolvimento do jogo Instituto do Terror permitiu integrar conhecimentos práticos e teóricos adquiridos ao longo do curso de Análise e Desenvolvimento de Sistemas do IFSP, especialmente no contexto da disciplina de Tópicos Avançados. A partir do uso de ferramentas modernas de criação gráfica, plataformas de IA generativa, edição de spritesheets e implementação técnica com o framework

Phaser 3, foi possível construir um produto funcional, coerente e alinhado ao tema proposto de Fantasia.

A realização do projeto evidenciou a capacidade do estudante de estruturar um jogo completo utilizando conceitos de scene management, física arcade, animações, manipulação de objetos, colisões, lógica de progressão, organização modular do código e integração de múltiplos assets. O jogo apresenta narrativa consistente, ambientação visual marcante e mecânicas estáveis, demonstrando domínio das tecnologias empregadas e habilidade em transformar uma ideia criativa em uma aplicação interativa e jogável.

Além disso, o projeto reforçou competências essenciais como resolução de problemas, tratamento de erros, refinamento de assets gráficos, pesquisa de ferramentas, adaptação de códigos, compreensão de pipelines de renderização e uso eficiente de bibliotecas JavaScript. A construção dos spritesheets, em especial, demandou precisão técnica e rigor no alinhamento e padronização dos quadros, o que elevou a maturidade do processo de desenvolvimento.

Mesmo cumprindo os objetivos acadêmicos, o jogo possui potencial significativo para aprimoramentos e expansões. Algumas sugestões de melhorias incluem:

- Inclusão de novas fases, com cenários diferentes, aumento gradual de dificuldade e variações temáticas;
- Criação de novos tipos de inimigos, com comportamentos distintos, padrões específicos de movimento ou habilidades especiais;
- Implementação de power-ups, como velocidade temporária, invencibilidade, magnetismo para coletar notas, entre outros;
- Sistema de vidas ou saúde, oferecendo maior profundidade estratégica ao gameplay;
- Cutscenes narrativas, reforçando o storytelling da viagem no tempo e da distopia;
- Sistema de pontuação global, com leaderboard local ou online;
- Criação de uma fase final (boss) representando um professor fantasma mais poderoso;

- Versão mobile responsiva, ajustando a interface e controles para telas sensíveis ao toque;
- Menu de configurações, com ajustes de volume, sensibilidade e acessibilidade;
- Transições animadas entre cenas, aumentando a polidez do produto final.

Essas melhorias não apenas agregariam valor ao jogo, como também ampliariam o aprendizado do desenvolvedor, permitindo a exploração de novos recursos técnicos e criativos.

Em síntese, o jogo Instituto do Terror cumpriu seu propósito dentro da disciplina, demonstrando competência técnica, criatividade, autonomia e capacidade de execução do estudante. Ao mesmo tempo, estabelece uma base sólida para evoluções futuras, podendo transformar-se em um projeto ainda mais rico, complexo e expressivo dentro do portfólio acadêmico e profissional.

## **Referências Bibliográficas**

BRASIL. Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – IFSP. Slides da disciplina de Tópicos Avançados. Curso de Análise e Desenvolvimento de Sistemas. Campos do Jordão, 2025.

PHASER. Phaser 3 Framework: documentação oficial. Disponível em:  
<https://phaser.io/>. Acesso em: 24 nov. 2025.

PHOTOPEA. Photopea – Editor de imagens online. Disponível em:  
<https://www.photopea.com/>. Acesso em: 24 nov. 2025.

LUDO AI. Ferramenta de criação de sprites e conceitos para jogos. Disponível em:  
<https://ludo.ai/>

. Acesso em: 24 nov. 2025.

GOOGLE. Gemini – IA generativa para criação de imagens e conteúdos visuais.

Disponível em: <https://gemini.google.com/>

. Acesso em: 24 nov. 2025.

OPENAI. ChatGPT – Ferramenta de geração de conteúdo e imagens. Disponível

em: <https://chatgpt.com/>

. Acesso em: 24 nov. 2025.

PIXABAY. Biblioteca gratuita de músicas e efeitos sonoros. Disponível em:

<https://pixabay.com/>

. Acesso em: 24 nov. 2025.

CLIDEO. Ferramenta de corte e edição de áudio online. Disponível em:

<https://cliffeo.com/pt/cut-audio>

. Acesso em: 24 nov. 2025.

VITE. Vite – Build tool para desenvolvimento Front-End. Disponível em:

<https://vitejs.dev/>

. Acesso em: 24 nov. 2025.

MOZILLA DEVELOPER NETWORK. Documentação JavaScript (ES6). Disponível

em: <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>

. Acesso em: 24 nov. 2025.

CHROME DEVELOPERS. Ferramentas de desenvolvimento (DevTools). Disponível

em: <https://developer.chrome.com/docs/devtools/>

. Acesso em: 24 nov. 2025.