



UNIVERSIDADE DA AMAZÔNIA - UNAMA CIÊNCIAS DA COMPUTAÇÃO

DÉLCIO FARIAS DIAS NETO - 04183547 / PATRÍCIA MELISSA DA CRUZ SARAIVA - 04179602
/ ROSÂNGELA DIAS - 04188826

****RELATÓRIO TÉCNICO: DISK DEFRAG**** ****Desenvolvimento de Jogo Educacional com
Biblioteca Pygame****

BELÉM/PA 2025

DÉLCIO FARIAS DIAS NETO - 04183547 / PATRÍCIA MELISSA DA CRUZ SARAIVA - 04179602 /
ROSÂNGELA DIAS - 04188826

****RELATÓRIO TÉCNICO: DISK DEFRAG**** ****Desenvolvimento de Jogo Educacional com
Biblioteca Pygame****

Trabalho apresentado à disciplina de Tópicos Especiais do curso de Ciências da
Computação da Universidade da Amazônia - UNAMA, como requisito parcial para
obtenção de nota. Orientador: Prof. Wilson Cosmos

BELÉM/PA 2025

****RESUMO**** O presente relatório detalha o desenvolvimento do jogo "Disk Defrag", um projeto acadêmico para a disciplina de Tópicos Especiais. O jogo, criado em Python com a biblioteca Pygame, simula a manutenção de um disco rígido de forma lúdica. O desenvolvimento seguiu a restrição de não utilizar ativos visuais externos, empregando apenas primitivas geométricas. A configuração do jogo é externa, via arquivo `.ini`, e a jogabilidade inclui mecânicas como reparo de setores defeituosos, neutralização de ameaças e um sistema de combos. O projeto cumpriu todos os requisitos, demonstrando competência técnica na aplicação de lógica de programação e manipulação de bibliotecas gráficas. ****Palavras-chave:**** Pygame, Desenvolvimento de Jogos, Python, Lógica de Programação.

****ABSTRACT**** This report details the development of the game "Disk Defrag", an academic project for the Special Topics discipline. The game, created in Python with the Pygame library, simulates hard drive maintenance in a playful way. The development followed the restriction of not using external visual assets, employing only geometric primitives. The game's configuration is external, via an `.ini` file, and the gameplay includes mechanics such as repairing bad sectors, neutralizing threats, and a combo system. The project met all requirements, demonstrating technical competence in applying programming logic and handling graphics libraries.

****Keywords:**** Pygame, Game Development, Python, Programming Logic.

****SUMÁRIO**** 1. INTRODUÇÃO 2. ESPECIFICAÇÕES TÉCNICAS 2.1. Tecnologias Utilizadas 2.2. Arquitetura de Configuração 2.3. Renderização "Vanilla" 3. DESENVOLVIMENTO E MECÂNICAS 3.1. Dinâmica de Jogo 3.2. Sistema de Combos e Recompensas 3.3. Persistência de Dados (I/O) 4. MANUAL DO USUÁRIO 4.1. Lore (Contexto) 4.2. Controles 4.3. Elementos Visuais 5. CONCLUSÃO REFERÊNCIAS

1. INTRODUÇÃO

O presente relatório descreve o desenvolvimento do software "Disk Defrag", um jogo eletrônico desenvolvido como parte da avaliação da disciplina de Tópicos Especiais. O projeto tem como objetivo a aplicação prática de lógica de programação e manipulação de bibliotecas gráficas utilizando a linguagem Python e o framework Pygame.

O escopo do projeto foi definido pelas diretrizes do "PyGaming Hub", que exigem a criação de um jogo sem o uso de ativos externos (imagens ou sprites), utilizando exclusivamente primitivas geométricas para a renderização visual. Além disso, o

software deve obedecer a rigorosos padrões de configuração externa via arquivo `conf.ini`, garantindo a separabilidade entre código e configuração.

O “Disk Defrag” simula, de forma gamificada, a tarefa de manutenção de um disco rígido, onde o jogador atua como um protocolo de defesa contra a fragmentação de dados e ataques de malwares.

2. ESPECIFICAÇÕES TÉCNICAS

O desenvolvimento do software respeitou integralmente as restrições impostas pelo projeto:

2.1. Tecnologias Utilizadas

- **Linguagem:** Python 3.12.
- **Biblioteca Gráfica:** Pygame (versão Community Edition).
- **Controle de Versão:** Git e GitHub (Fluxo de Fork e Pull Request).

2.2. Arquitetura de Configuração

O sistema não possui controles ou resoluções fixas no código-fonte “hardcoded”. Conforme a norma técnica do projeto, o jogo realiza a leitura do arquivo `conf/conf.ini` através da biblioteca `configparser`. O software implementa um mapeamento dinâmico que converte as strings do arquivo de configuração (ex: “w”, “return”) em constantes de evento do Pygame (`pygame.K_w`, `pygame.K_RETURN`).

2.3. Renderização “Vanilla”

Não foram utilizados arquivos de imagem (.png, .jpg). Todos os elementos visuais — incluindo o cursor do jogador, setores do disco, inimigos, partículas de explosão e a interface de usuário (HUD) — são gerados em tempo real utilizando funções nativas como `pygame.draw.rect()` e renderização de fontes do sistema.

3. DESENVOLVIMENTO E MECÂNICAS

O jogo foi estruturado sob o paradigma de Máquina de Estados (State Machine), alternando entre MENU, GAME (Execução) e BSOD (Tela de Erro/Game Over).

3.1. Dinâmica de Jogo

O “Disk Defrag” opera sobre uma matriz (Grid) de 8x6 setores. O jogador controla um cursor de leitura/gravação que deve navegar pelos setores para corrigir falhas.

- **Setores Defeituosos (Bad Blocks):** Blocos vermelhos que aumentam a fragmentação do disco.
- **Ameaças Ativas (Rootkits):** Blocos roxos que atuam como “bombas lógicas”. Possuem um temporizador interno; se não forem reparados a tempo, explodem e corrompem os setores adjacentes (área 3x3).

3.2. Sistema de Combos e Recompensas

Para incentivar o desempenho (“Flow”), foi implementado um sistema de Combo. Acertos consecutivos sem erros aumentam o multiplicador de pontuação.

- **Mecânica de Nuke:** Ao atingir um combo de 50x, o sistema recompensa o jogador instantaneamente com um setor especial (Ciano) que, ao ser ativado, limpa todas as ameaças da tela.

3.3. Persistência de Dados (I/O)

O jogo implementa um sistema de Highscore local. As 5 melhores pontuações são serializadas e armazenadas em um arquivo `system.log`. Ao final de cada partida, o sistema verifica se houve um novo recorde, oferecendo feedback visual ao usuário.

4. MANUAL DO USUÁRIO

4.1. Lore (Contexto)

“O servidor central entrou em colapso lógico. Você é o protocolo de defesa final encarregado de impedir a perda total de dados. Navegue pela matriz do HD, repare setores defeituosos e neutralize Rootkits explosivos em tempo real. Sua missão é simples: mantenha a integridade do disco e evite a Tela Azul da Morte (BSOD).”

4.2. Controles

Os controles são definidos no arquivo `conf.ini`, mas o padrão desenvolvido é:

- **Navegação:** Teclas W, A, S, D (ou Setas).
- **Reparar Setor (Ação A):** Tecla O.
- **Pausar Sistema (Ação B):** Tecla P.
- **Encerrar Execução:** Tecla Enter (Return).

4.3. Elementos Visuais

- **Verde:** Setor Saudável / Cursor.
- **Vermelho:** Setor Danificado (Repare rápido).
- **Roxo:** Rootkit (Prioridade Máxima - Explode em 3s).
- **Cinza:** Ransomware Lock (Armadilha - Não toque).
- **Azul Escuro:** Firewall Shield (Proteção contra dano).
- **Ciano:** System Purge (Limpa a tela).

5. CONCLUSÃO

O projeto "Disk Defrag" atendeu a todos os requisitos propostos na ementa da disciplina. A restrição de não utilizar sprites externos fomentou a criatividade na utilização de formas geométricas e cores para comunicar estados de jogo (perigo, bônus, dano).

A implementação técnica demonstrou competência no uso de laços de repetição, manipulação de dicionários para gerenciamento de entidades, leitura de arquivos externos e lógica de colisão em grade. O resultado final é um software funcional, robusto e alinhado com a temática de Tecnologia da Informação.

REFERÊNCIAS

- [1] COSMO, W. Guia de Contribuição: PyGaming Hub. Documento de Requisitos do Projeto. 2025.
- [2] COSMO, W. Tópicos Especiais - Projeto Pygame. Material Didático/Slides. 2025.