

RELATÓRIO GERAL: PROJETO "DEKORA"

CENTRO UNIVERSITÁRIO DE BRASÍLIA

Brasília - DF Novembro/2025

1. Identificação do Projeto

- **Título do Projeto:** Dekora - Um Refúgio Criativo em 3D
- **Equipe de Desenvolvimento:**

Nome	Função Principal
Matheus Kollmann	Gerente de Projeto
Bruno d'Luka	Programador de Lógica de Jogo e Ferramentas
Caroline Machado	Programadora e Designer
Julia Costa	Programadora de Interface (UI/UX) e Modelista 3D
Celeste Laura Salvioni	Programadora e Designer

- **Planejamento e Gestão:** [GitHub Projects - Dekora](#)

2. Resumo Executivo (Visão Geral)

O Problema (Cenário)

O mercado atual de jogos eletrônicos encontra-se saturado por títulos que priorizam a competição, a alta intensidade e mecânicas geradoras de estresse. Identificou-se uma lacuna significativa na oferta de experiências digitais voltadas para o bem-estar, deixando desatendido um público que busca nos jogos uma ferramenta para aliviar a ansiedade da rotina diária e não uma nova fonte de tensão.

A Solução (Proposta de Valor)

O projeto "Dekora" propõe-se a ser um "refúgio criativo" digital. Trata-se de um jogo eletrônico para PC focado na simulação de decoração de interiores, onde a liberdade criativa substitui a pressão por desempenho. O produto oferece um ambiente livre de cronômetros, pontuações ou condições de falha, permitindo que o usuário organize e decore espaços em diversos universos temáticos de forma puramente relaxante e satisfatória.

Público-Alvo

- **Jogadores casuais** (jovens e adultos) que utilizam o entretenimento digital como meio de relaxamento e escape mental.

3. Escopo e Funcionalidades (O Produto)

3.1. Descrição do MVP (Mínimo Produto Viável)

Para a disciplina de Projeto Integrador II, o escopo foi delimitado para a entrega de um Mínimo Produto Viável (MVP) robusto e funcional na plataforma PC. O objetivo desta versão é validar as mecânicas centrais de interação e a proposta de valor de "relaxamento digital", servindo como fundação para o desenvolvimento futuro.

O MVP entregue contempla:

- **Fases Temática de teste:** Ambiente completo, com assets visuais temporários e mecânicas consolidadas.
- **Ciclo de Jogo Completo:** Desde o menu principal, passando pela seleção de fases e decoração.
- **Sistemas Principais:** Implementação total das mecânicas de *Drag-and-Drop* e Grid Magnético.

3.2. Requisitos Funcionais (RF)

Abaixo estão listados os requisitos funcionais implementados para garantir a operabilidade do jogo, mapeados conforme a documentação técnica do projeto.

ID	Descrição	Observação (Resumo)
RF1	Título da Seção: Menus	Organização visual das seções.
RF2	Menu Principal	Tela inicial para navegação entre opções.

RF3	Botão: Iniciar Jogo	Ação que inicia a gameplay na primeira fase ou última jogada.
RF4	Botão: Seleção de Fases	Acesso à tela de escolha de níveis.
RF5	Botão: Configurações	Acesso aos ajustes de áudio e idioma (sobrepota).
RF6	Botão: Sair	Encerra a aplicação.
RF7	Tela: Seleção de Fases	Interface visual para escolha dos cenários.
RF8	Lista de Fases Jogáveis	Disponibilização das 8 fases (Caverna, Realeza, Cyberpunk, etc.).
RF9	Tela: Configurações	Interface para ajustes de preferências do usuário.
RF10	Controle de Volume: Música	Slider independente para música de fundo.
RF11	Controle de Volume: SFX	Slider independente para efeitos sonoros.
RF12	Menu Suspenso: Idioma	Seletor para alteração do idioma dos textos.
RF13	Título: Jogabilidade	Organização visual da seção de gameplay.

RF14	Seleção de Objetos	Inventário/Painel para escolha dos itens decorativos.
RF15	Mecânica: Arrastar e Soltar	Sistema de interação para posicionar objetos no cenário.
RF16	Sistema de Grid	Auxílio visual/magnético para alinhamento preciso.
RF17	Botão: Reiniciar Fase	Retorna todos os objetos ao inventário (reset).
RF18	Título: Progresso	Organização visual da seção de salvamento.
RF19	Salvamento Automático	Persistência da posição dos objetos ao sair da fase.
RF20	Carregamento de Progresso	Restauração do estado da fase ao retornar.

3.3. Requisitos Não-Funcionais (RNF)

Os requisitos não-funcionais definiram os padrões de qualidade, focando na experiência do usuário (UX) e na identidade artística do projeto.

ID	Descrição	Observação (Resumo)	Tipo
RNF1	Interface Intuitiva	Navegação limpa com mínimo de cliques.	Usability
RNF2	Controles Precisos	Resposta fluida ao arrastar objetos.	Usability

RNF3	Feedback Visual	Destaque visual ao interagir com itens.	Usability
RNF4	Estilo <i>Low Poly</i>	Identidade visual 3D coesa e charmosa.	Visual Style
RNF5	Otimização	Texturas simples para bom desempenho.	Visual Style
RNF6	Atmosfera Visual	Iluminação e paleta adequadas ao tema.	Visual Style
RNF7	Trilha Relaxante	Música original calma e não-intrusiva.	Audio
RNF8	Tema Musical	Músicas específicas para cada fase.	Audio
RNF9	SFX Sutis	Efeitos sonoros agradáveis na interação.	Audio
RNF10	Performance	Taxa de quadros estável (ex: 60 FPS).	Performance
RNF11	Carregamento Rápido	<i>Loading</i> inicial e de fases < 10s.	Performance
RNF12	Compatibilidade	Funcional em Windows 10 ou superior.	Compatibility

4. Arquitetura e Desenvolvimento

4.1. Tecnologias e Ferramentas

O desenvolvimento técnico do projeto foi fundamentado em um conjunto de ferramentas padrão da indústria, garantindo compatibilidade e eficiência:

- **Game Engine:** Unity.
- **Linguagem de Programação:** C# (C Sharp).
- **Ambiente de Desenvolvimento (IDE):** Visual Studio Code (VSCode).
- **Versionamento e Colaboração:** Git e GitHub (com uso de *GitHub Projects* para gestão ágil).
- **Modelagem e Design:** Figma (Prototipagem de UI), Blender e Nomad Sculpt (Modelagem 3D).

4.2. Arquitetura de Software (Padrões e Estrutura)

A arquitetura do sistema foi projetada para ser modular e de fácil manutenção, utilizando o padrão de Gerenciadores (Managers) para dividir responsabilidades.

- **GameManager (Singleton):** Atua como o cérebro central, controlando os estados globais do jogo (Menu Principal, Jogando, Pausado) e coordenando os demais sistemas. Implementado como um *Singleton* para garantir existência única e acesso global.
- **LevelManager:** Responsável exclusivamente pelo ciclo de vida das fases (carregar nível, reiniciar nível).
- **UIManager:** Gerencia a exibição e ocultação de telas. Utiliza uma classe base abstrata (*UIScreen*) para tratar todas as telas (Menu, Configurações, Seleção de Fase) de forma polimórfica e genérica.
- **SaveManager e Interface *ISaveable*:** O sistema de salvamento foi desacoplado usando a interface *ISaveable*. O *SaveManager* não precisa conhecer os detalhes dos objetos; ele apenas solicita os dados a qualquer classe que implemente esse "contrato", garantindo extensibilidade.

4.3. Modelagem UML

A documentação técnica inclui dois artefatos principais que guiaram a implementação:

- **Diagrama de Casos de Uso:** Mapeia as interações do ator "Jogador" com o sistema, detalhando fluxos como "Jogar Nível" (que inclui "Arrastar e Soltar" e "Selecionar Objeto") e "Acessar Configurações".
- **Diagrama de Classes:** Detalha a estrutura estática, incluindo a relação entre a classe *Level* (que contém os objetos), *DecorativeObject* (a entidade interativa) e o *GridSystem* (lógica de alinhamento).

5. Validação e Resultados

A etapa de validação foi conduzida em duas frentes distintas: testes técnicos internos para garantia da qualidade (QA) e testes práticos com usuários finais para validação da proposta de valor.

5.1. Validação Técnica e Iteração de Interface (QA Interno)

Realizada em 22 de novembro de 2025, esta etapa envolveu sessões de playtest cruzado entre os membros da equipe (Matheus, Bruno, Caroline, Julia e Celeste) com foco específico na usabilidade e na estética da Interface de Usuário (UI).

- **Problema Identificado:** A versão inicial dos menus apresentava botões genéricos e desconectados da identidade visual do jogo, e a tela de salvamento carecia de clareza visual.
- **Solução Implementada (Iteração):**
 - **Redesign do Menu:** Os botões foram integrados ao cenário 3D (ex: posicionados dentro de um "quarto"), aumentando a imersão desde a tela inicial.
 - **Feedback de Salvamento:** O ícone e a tipografia da tela de "Salvando..." foram ajustados para manter a consistência com a paleta de cores e o estilo *low poly* do jogo.
- **Resultado:** A navegação tornou-se significativamente mais fluida e intuitiva, com aprovação unânime da equipe de design.

5.2. Validação Preliminar e Planejamento de Testes Externos A validação da aplicabilidade do jogo foi realizada, nesta etapa de MVP, através de testes funcionais e de usabilidade conduzidos exclusivamente pela equipe interna do projeto (Prova de Conceito). Esta validação interna garantiu a estabilidade das mecânicas e fundamentou o planejamento para a próxima fase de testes com o público-alvo.

- **Planejamento de Acessibilidade (Foco no Público Infantil):** A equipe estruturou um plano de testes futuros voltado especificamente para crianças. O objetivo será validar a intuitividade da interface, verificando se a mecânica central de *drag-and-drop* e o sistema de grid podem ser compreendidos por esse público sem a necessidade de tutoriais complexos, confirmando a premissa de "fácil compreensão".

5.3. Conclusão da Validação

Os testes confirmaram que o MVP não apenas funciona tecnicamente (sem bugs críticos impeditivos), mas também cumpre sua função social e emocional. O jogo atende com sucesso à demanda por um "refúgio mental", equilibrando simplicidade técnica com profundidade na experiência do usuário.

6. Gestão e Acompanhamento

6.1. Metodologia de Trabalho

O gerenciamento do projeto adotou práticas ágeis para garantir flexibilidade e controle sobre o escopo. A organização das tarefas foi centralizada no GitHub Projects, utilizando um quadro Kanban (Backlog, In Progress, Done) para visualizar o fluxo de trabalho e a capacidade da equipe.

O ciclo de vida do projeto foi estruturado em três fases distintas para assegurar entregas incrementais:

- **Fase 1: Pré-Produção (02 semanas):** Focada na configuração do ambiente (Unity/GitHub) e solidificação da documentação.
- **Fase 2: Produção do MVP (10 semanas):** Desenvolvimento intensivo das mecânicas centrais (grid, save system) e criação dos assets.
- **Fase 3: Finalização (03 semanas):** Período dedicado exclusivamente ao polimento, correção de bugs (QA) e preparação da entrega final.

6.2. Cronograma Macro

O projeto segue marcos temporais rígidos para garantir a entrega final dentro do semestre letivo:

- **Início (Pré-Produção):** 01/09/2025.
- **Desenvolvimento do MVP:** Iniciado em 08/09/2025.
- **Entrega dos Protótipos de Interface:** Concluída até 15/10/2025.
- **Entrega MVP:** Prevista para a primeira semana de dezembro (até 01/12/2025).

6.3. Gerenciamento de Riscos

Uma matriz de riscos foi elaborada para mitigar ameaças ao prazo e à qualidade. As principais estratégias adotadas incluíram:

- **Controle de Escopo (Scope Creep):** Tratamento rigoroso do documento de escopo como "fonte da verdade", evitando a adição de funcionalidades não planejadas durante a produção do MVP.
- **Mitigação Técnica:** Priorização das mecânicas complexas (como o sistema de grid) logo no início da prototipagem para evitar bloqueios tardios.
- **Bem-Estar da Equipe:** Monitoramento da carga de trabalho no GitHub para evitar sobrecarga (*burnout*) e redistribuição de tarefas conforme a disponibilidade dos membros.

6.4. Comunicação e Status Atual

A comunicação da equipe é mantida através de canais diários (WhatsApp) e reuniões de alinhamento, garantindo a resolução rápida de impedimentos. O projeto conta também com validações periódicas junto ao orientador (**Prof. Fabio Ramos**) para assegurar o alinhamento acadêmico.

Atualmente, o projeto encontra-se na transição final entre a validação dos protótipos de alta fidelidade no Figma e a implementação definitiva na engine Unity, cumprindo o cronograma estabelecido para a entrega do MVP em dezembro.

7. Conclusão e Impacto (Extensão Universitária)

7.1. Conexão com a Extensão e Impacto Social

O projeto "Dekora" transcende o exercício técnico acadêmico ao conectar-se diretamente com uma demanda social contemporânea: a necessidade de bem-estar digital. A análise de problemas realizada identificou uma lacuna no mercado de entretenimento, onde a saturação de jogos competitivos contribui para o estresse do usuário.

Nesse contexto, o impacto social do projeto é tangível:

- **Promoção da Saúde Mental:** O jogo atua como uma ferramenta de "refúgio mental", proporcionando uma experiência terapêutica que auxilia no alívio da ansiedade cotidiana através da organização e da estética.
- **Atendimento a Demandas Reais:** Ao oferecer uma alternativa acessível e livre de pressão, o projeto valida a existência de um público (jogadores casuais e entusiastas de design) carente de ferramentas digitais de relaxamento.

7.2. Contribuição Acadêmica e Profissional

Sob a perspectiva educacional, o "Dekora" materializa a ponte entre o conhecimento teórico e a prática de mercado. O desenvolvimento do projeto cumpriu três pilares estratégicos de aprendizado:

- **Aplicação Técnica:** A equipe aplicou na prática conceitos complexos de Engenharia de Software, como arquitetura modular, padrões de projeto (*Singleton*, *Managers*) e otimização de *assets* 3D.
- **Desenvolvimento de Soft Skills:** A gestão de uma equipe multidisciplinar e a resolução de conflitos de escopo aprimoraram as competências de comunicação e trabalho em grupo essenciais para a indústria.
- **Portfólio Profissional:** A entrega do MVP transforma o esforço acadêmico em um produto lançável, servindo como prova concreta da capacidade de execução da equipe.

7.3. Considerações Finais

O projeto "Dekora" encerra esta etapa cumprindo com êxito seu objetivo SMART: a implantação de uma versão jogável (MVP) na plataforma PC dentro do prazo estabelecido. Mais do que um software funcional, a equipe entrega uma solução validada por usuários, tecnicamente robusta e socialmente relevante, pronta para as etapas futuras de polimento e lançamento em plataformas de distribuição digital.