

DOCUMENTO DE REQUISITOS DO SISTEMA

Apresentação do documento

Este documento descreve os requisitos funcionais e não funcionais do sistema ODSlize, um jogo digital de slide puzzle sobre os 17 Objetivos de Desenvolvimento Sustentável.

Visão geral do projeto

Nome do projeto: ODSlize

Equipe do projeto: GEISA MORAIS GABRIEL

LEONARDO INACIO GUILHERME DANTAS

TIAGO AMARO NUNES

Objetivos

- Facilitar o ensino-aprendizagem dos ODS por meio de um jogo Web.
- Aumentar o engajamento e estimular a adoção de práticas sustentáveis no dia a dia por meio da gamificação.
- Promover a educação e aumentar a conscientização sobre os Objetivos do Desenvolvimento Sustentável.
- Organizar e controlar as mudanças no software a fim de evitar retrabalho e facilitar atualizações e melhorias no sistema.

Histórico de registros

Versão	Data	Autor	Descrição
0.1	25/09/2025	Geísa M. Gabriel	Descrição de requisitos funcionais e não funcionais
0.2	22/11/2025	Geísa M. Gabriel	Atualização da descrição de requisitos funcionais
0.3	06/12/2025	Geísa M. Gabriel	Atualização da descrição de requisitos funcionais

REQUISITOS FUNCIONAIS DO SISTEMA

[RF01] - Iniciar partida

Descrição:	O sistema deve permitir que o jogador inicie uma nova partida, com o tabuleiro embaralhado automaticamente de forma que possa ser resolvido.
Prioridade:	Essencial.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador acessa o sistema. 2. O sistema exibe a tela inicial do jogo. 3. O jogador seleciona a opção "jogar". 4. O sistema exibe os níveis abertos. 5. O jogador seleciona o nível de jogo liberado.
Fluxo alternativo:	

[RF02] - Movimentar peças

Descrição:	O sistema deve permitir que o jogador mova uma peça para a casa vazia adjacente por meio de um clique ou gesto (arrastando).
Prioridade:	Essencial.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador inicia o jogo. 2. O sistema embaralha o tabuleiro.
Fluxo alternativo:	

[RF03] - Reiniciar partida

Descrição:	O sistema deve permitir ao jogador reiniciar a partida atual reembaralhando o jogo.
Prioridade:	Essencial.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O sistema exibe um botão "reiniciar" após o início da partida. 2. O jogador seleciona o botão. 3. O jogo é reiniciado na configuração atual.
Fluxo alternativo:	

[RF04] - Contabilizar tempo e movimentos	
Descrição:	O sistema deve contabilizar o tempo decorrido desde o início da partida e a quantidade de movimentos realizados por rodada.
Prioridade:	Importante.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador inicia a partida. 2. O sistema inicia a contagem do tempo e de movimentos enquanto o jogador não concluir a partida.
Fluxo alternativo:	

[RF05] Possuir níveis de dificuldade	
Descrição:	O sistema deve permitir que o jogador avance no jogo por meio de níveis de dificuldade que devem variar conforme o tamanho do tabuleiro (ex.: 3x3, 4x4).
Prioridade:	Importante.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O sistema exibe os níveis do jogo. 2. O jogador avança no jogo a medida que conclui um nível.
Fluxo alternativo:	1. O jogador pode rejogar um nível já realizado.

[RF06] Armazenar resultados do jogo	
Descrição:	O sistema deve armazenar os resultados do jogo para os jogadores que estão logados.
Prioridade:	Importante.
Ator(es):	Usuário.
Entradas e pré-condições:	1. O sistema está na homepage do usuário. 2. O usuário ter concluído uma partida.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador inicia a partida. 2. O sistema armazena o resultado de cada nível atingido pelo jogador.
Fluxo alternativo:	

[RF07] Exibir resultados do jogo

Descrição:	O sistema deve exibir o melhor tempo de jogo e a menor quantidade de movimentos daquele jogador para cada fase a cada partida finalizada.
Prioridade:	Importante.
Ator(es):	Usuário.
Entradas e pré-condições:	1. O sistema está na homepage do usuário. 2. O usuário ter concluído uma partida.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador inicia a partida. 2. O sistema informa ao jogador o tempo da partida e a quantidade de movimentos.
Fluxo alternativo:	

[RF08] Informar a vitória do jogador	
Descrição:	O sistema deve verificar automaticamente se as peças estão na ordem correta, e em caso afirmativo, deve informar a vitória do jogador.
Prioridade:	Essencial.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador resolve o <i>slide puzzle</i> . 2. O sistema exibe o estado de vitória do jogador para aquele nível de jogo.
Fluxo alternativo:	

[RF09] Explicar o ODS	
Descrição:	Após o estado de vitória, o sistema deve retornar uma explicação sobre o ODS associado àquele slide puzzle, promovendo o conhecimento ao jogador.
Prioridade:	Importante.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador resolve o <i>slide puzzle</i> . 2. O sistema exibe uma breve explicação sobre o ODS associado ao <i>slide puzzle</i> resolvido.
Fluxo alternativo:	

[RF10] Apresentar resultado esperado	
Descrição:	O sistema deve apresentar o resultado esperado para que o jogador visualize a organização esperada das peças.

Prioridade:	Essencial.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador inicia a partida. 2. O sistema exibe a imagem esperada para o <i>slide puzzle</i> proposto.
Fluxo alternativo:	

[RF11] Apresentar <i>ranking</i>	
Descrição:	O sistema deve apresentar a posição do usuário no ranking para cada nível de jogo em relação aos demais jogadores.
Prioridade:	Desejável.
Ator(es):	Usuário.
Entradas e pré-condições:	1. O sistema está na homepage do usuário. 2. O usuário possuir conta no sistema.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador inicia a partida. 2. O jogador resolver um nível do <i>puzzle</i> . 3. O sistema retorna a posição do jogador para aquele nível em relação aos demais jogadores.
Fluxo alternativo:	

[RF12] Realizar cadastro	
Descrição:	O sistema deve permitir que o usuário crie uma conta inserindo e-mail e senha para que possa fazer login usando essas informações.
Prioridade:	Desejável.
Ator(es):	Usuário.
Entradas e pré-condições:	O sistema está na homepage do usuário.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador acessa o sistema. 2. O sistema exibe a opção de realizar cadastro criando uma conta.
Fluxo alternativo:	

[RF13] Confirmar cadastro	
Descrição:	O sistema deve enviar um e-mail de confirmação para que o usuário possa realizar a autenticação do seu cadastro no site do jogo.
Prioridade:	Desejável.
Ator(es):	Usuário.

Entradas e pré-condições:	1. O sistema está na homepage do usuário. 2. O jogador ter criado conta.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador cria conta inserindo nome de usuário, e-mail e senha. 2. O sistema envia um e-mail de confirmação de criação de conta contendo um código para que o jogador realize a autenticação da conta criada no site do jogo.
Fluxo alternativo:	

[RF14] Deletar usuário	
Descrição:	O sistema deve permitir que um usuário realize a exclusão da sua conta no site do jogo, perdendo todo o progresso obtido.
Prioridade:	Desejável.
Ator(es):	Usuário.
Entradas e pré-condições:	1. O sistema está na homepage do usuário. 2. O jogador ter pelo menos uma conta criada no sistema.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador acessa sua conta (login). 2. O sistema exibe um opção de excluir conta no perfil do jogador.
Fluxo alternativo:	

[RF15] Recuperar senha	
Descrição:	O sistema deve permitir que um usuário recupere senha.
Prioridade:	Importante.
Ator(es):	Usuário.
Entradas e pré-condições:	1. O sistema está na homepage do usuário. 2. O jogador ter pelo menos uma conta criada no sistema.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O jogador clica sobre a opção de recuperar senha antes de realizar o login.
Fluxo alternativo:	

REQUISITOS NÃO FUNCIONAIS SISTEMA

[RNF01] - Usabilidade (operabilidade)	
Descrição:	A interface do sistema deve ser responsiva, adaptando-se automaticamente a diferentes tamanhos de tela.
Atores:	Usuário, Sistema.

Prioridade:	Essencial.
Entradas e pré-condições:	1. O sistema deve estar funcional e em execução. 2. A conexão de rede deve estar estável.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O usuário acessa o sistema. 2. O sistema apresenta a tela inicial do jogo ao usuário.
Fluxo alternativo:	

[RNF02] - Usabilidade (aprendizibilidade)	
Descrição:	O sistema deve informar as instruções de como se joga, facilitando o aprendizado do usuário.
Atores:	Usuário, Sistema.
Prioridade:	Importante.
Entradas e pré-condições:	1. O sistema deve estar funcional e em execução. 2. A conexão de rede deve estar estável.
Saídas e pós-condições:	
Fluxos de eventos	
Fluxo principal:	1. O usuário acessa o sistema. 2. O sistema apresenta a tela inicial do jogo contendo as instruções ao usuário.
Fluxo alternativo:	