

Requisitos Funcionais para Jogo de Batalha Pokémon na Web

Filipe Sampaio, Vinicius Cantanhede, Rafael Meireles

October 23, 2024

1 Introdução

Este documento apresenta os requisitos funcionais para o desenvolvimento de um **Jogo de Batalha Pokémon na Web**, utilizando **JavaScript** para a implementação e **MongoDB** como banco de dados. O foco principal do sistema é permitir que treinadores batalhem com Pokémons pré-definidos e que o desempenho seja registrado e exibido em leaderboards.

2 Requisitos Funcionais

2.1 Cadastro e Autenticação de Usuários

2.1.1 Descrição da Funcionalidade

Permitir que novos treinadores se cadastrem no jogo e que treinadores existentes possam autenticar-se para acessar suas contas.

2.1.2 Detalhes

- **Cadastro de Treinador:** O sistema deve permitir que novos treinadores criem uma conta fornecendo informações básicas como nome de usuário, email e senha.
- **Autenticação:** Treinadores cadastrados devem conseguir fazer login utilizando seu email e senha.

- **Recuperação de Senha:** Funcionalidade para recuperação de senha via email.
- **Verificação de Email:** Enviar email de verificação após o cadastro para ativar a conta.

2.1.3 Dados Necessários

- **Usuários:**
 - `usuario_id` (UUID)
 - `nome_usuario` (string, único)
 - `email` (string, único)
 - `senha_hash` (string)
 - `data_cadastro` (timestamp)
 - `ultimo_login` (timestamp)
 - `email_verificado` (boolean)

2.2 Sistema de Batalhas Pokémon

2.2.1 Descrição da Funcionalidade

Permitir que os treinadores iniciem e participem de batalhas Pokémon contra outros treinadores ou NPCs (Personagens Não Jogadores), utilizando Pokémons pré-definidos.

2.2.2 Detalhes

- **Início de Batalha:** Treinadores podem desafiar outros treinadores ou enfrentar NPCs.
- **Mecânica de Batalha:** Implementação das regras de batalha, seleção de ataques, uso de itens (se aplicável), e processamento do resultado.
- **Resultado da Batalha:** Determinar o vencedor, atualizar estatísticas dos treinadores, e conceder recompensas (como pontos de experiência).
- **Histórico de Batalhas:** Registrar detalhes das batalhas realizadas para consulta futura.

2.2.3 Dados Necessários

- **Batalhas:**

- `batalha_id` (UUID)
- `treinador1_id` (referência ao documento de Usuários)
- `treinador2_id` (referência ao documento de Usuários ou NPC)
- `pokemon1_id` (referência ao documento de Pokémons pré-definidos)
- `pokemon2_id` (referência ao documento de Pokémons pré-definidos)
- `estado_batalha` (string: em andamento, concluída)
- `resultado` (string: treinador1 venceu, treinador2 venceu, empate)
- `data_batalha` (timestamp)
- `recompensas` (subdocumento com pontos de experiência ou outros benefícios)

- **Pokémons Pré-definidos:**

- `pokemon_id` (UUID)
- `nome` (string)
- `especie` (string)
- `nivel` (inteiro)
- `habilidades` (array de strings)
- `status` (subdocumento com atributos como HP, ataque, defesa, etc.)
- `imagem_url` (string)

2.3 Leaderboards (Rankings)

2.3.1 Descrição da Funcionalidade

Registrar e exibir rankings dos treinadores com base em seu desempenho nas batalhas, promovendo a competitividade e o engajamento dos jogadores.

2.3.2 Detalhes

- **Rankings Globais:** Listar os melhores treinadores com base em pontos acumulados, vitórias, etc.
- **Rankings por Categoria:** Rankings específicos, como por número de vitórias, pontos de experiência, etc.
- **Atualização Dinâmica:** Atualizar os rankings em tempo real conforme os treinadores participam de batalhas e acumulam pontos.
- **Visualização de Rankings:** Interface para que os treinadores possam visualizar seu posicionamento e o de outros treinadores.

2.3.3 Dados Necessários

- **Pontuações:**
 - `pontuacao_id` (UUID)
 - `usuario_id` (referência ao documento de Usuários)
 - `batalha_id` (referência ao documento de Batalhas)
 - `pontos` (inteiro)
 - `data` (timestamp)
- **Rankings:**
 - **Derivados a partir dos dados de Pontuações:** Agregando os pontos por usuário e ordenando de forma decrescente para determinar a posição no leaderboard.

3 Considerações Finais

Os requisitos funcionais simplificados acima focam nas funcionalidades essenciais para o funcionamento do **jogo de batalha Pokémon na web**, garantindo uma experiência básica porém completa para os treinadores. A utilização do **MongoDB** permite uma modelagem flexível dos dados, facilitando a gestão das batalhas, usuários e rankings de maneira eficiente.