

Especificação de Requisitos de Software (ERS)

Aplicativo: Linha do Tempo de Eventos Institucionais

1. Visão Geral

Aplicativo web que gera uma linha do tempo interativa a partir de arquivos `.md`, exibindo eventos organizados por ano. Cada evento pode ser do tipo acadêmico ou cerimônia de premiação, com múltiplos participantes. A aplicação permite navegação, visualização detalhada e filtros avançados.

2. Funcionalidades Principais

- Anos: agrupam múltiplos eventos.
- Tipos de eventos: Acadêmico e Cerimônia de Premiação.
- Participantes: cada evento possui participantes com funções diferentes.

3. Eventos Acadêmicos

- Atributos: Nome, Tema, Local, Data, Temática.
- Participantes:
 - * Palestrantes: com foto, biografia e resumo da palestra.
 - * Ouvintes: com foto e biografia.

4. Cerimônias de Premiação

- Atributos: Nome, Local, Data, Temática.
- Participantes: Foto, Biografia, Prêmio recebido (de uma lista predefinida).

5. Visualização e Filtros

- Linha do tempo por ano.
- Visualização expansível por evento.
- Página descritiva de cada evento.
- Modal ou seção para detalhes do participante.
- Filtros por:
 - * Ano
 - * Nome do participante
 - * Nome do evento

Especificação de Requisitos de Software (ERS) - Linha do Tempo de Eventos Institucionais

* Nome do prêmio

* Título/resumo da palestra

6. Entrada de Dados

Os dados serão gerados a partir de arquivos `.md` com metadados estruturados.

7. Tecnologias e Arquitetura

- Frontend: JavaScript orientado a objetos (OOP), HTML, CSS.
- Arquitetura: MVC (Modelo - Visão - Controlador).
- Dados: Arquivos `.md`, parseados com bibliotecas como `gray-matter`.
- Deploy: Vercel (hospedagem estática).

8. Requisitos Funcionais

- Gerar linha do tempo dinâmica por ano.
- Carregar eventos de arquivos `.md`.
- Exibir detalhes de eventos e participantes.
- Filtro dinâmico no frontend.
- Design responsivo.
- Navegação fluida entre vistas.
- Ordenar eventos cronologicamente.

9. Requisitos Não Funcionais

- Acessibilidade básica (etiquetas semânticas, ARIA).
- Bom desempenho na carga.
- Boas práticas de codificação (modularidade, reutilização).
- Estrutura de pastas organizada (`/models`, `/views`, `/controllers`, etc.).