

## Requisitos Funcionais

### 1. Gestão de Usuários:

- **Cadastro de Usuários:** O sistema deve permitir o cadastro de usuários com os seguintes atributos:
  - id (int)
  - name (String)
  - email (String)
  - password (String)
- **Alteração de Dados:** Usuários devem poder alterar seus dados pessoais e senha.
- **Visualização de Dados:** Usuários devem poder visualizar seus dados cadastrados.

### 2. Administração:

- **Gestão de Usuários:** Administradores devem poder:
  - Deletar usuários.
  - Listar todos os usuários registrados.
  - Criar novos usuários administradores.
- **Gestão de Eventos:** Administradores devem poder:
  - Adicionar novos eventos.
  - Deletar eventos existentes.

### 3. Gestão de Eventos:

- **Cadastro de Eventos:** O sistema deve permitir o cadastro de eventos com os seguintes atributos:
  - id (int)
  - creationDate (Date)
  - eventName (String)
  - description (String)
  - eventDate (Date)
  - eventType (String)
  - capacity (int)
  - price (float)
- **Registro em Eventos:** Usuários comuns devem poder se registrar em eventos.
- **Cancelamento de Registro:** Usuários comuns devem poder cancelar sua inscrição em eventos.

### 4. Certificados:

- **Emissão de Certificados:** Após a conclusão dos eventos, o sistema deve emitir certificados contendo:
  - id (int)
  - user (CommonUser)
  - event (Event)
  - dateIssue (Date)

- description (String)

## Requisitos Não Funcionais

### 1. Tecnologia:

- **Backend:** Spring Framework.
- **Frontend:** React.js.
- **Banco de Dados:** PostgreSQL.

### 2. Segurança:

- **Autenticação e Autorização:** Implementar autenticação e autorização para proteger os dados dos usuários e garantir que apenas administradores possam realizar ações administrativas.

### 3. Desempenho:

- **Escalabilidade:** O sistema deve ser escalável para suportar um grande número de usuários e eventos.

### 4. Usabilidade:

- **Interface Intuitiva:** A interface do usuário deve ser intuitiva e fácil de usar, garantindo uma boa experiência para os usuários.

## Estrutura do Banco de Dados

### ● Tabela de Usuários:

- id (Primary Key)
- name
- email
- password

### ● Tabela de Administradores:

- id (Primary Key)
- user\_id (Foreign Key para a tabela de Usuários)

### ● Tabela de Usuários Comuns:

- id (Primary Key)
- user\_id (Foreign Key para a tabela de Usuários)

### ● Tabela de Eventos:

- id (Primary Key)
- creationDate
- eventName
- description
- eventDate
- eventType
- capacity

- price
  - creator\_id (Foreign Key para a tabela de Administradores)
- **Tabela de Registros:**
  - id (Primary Key)
  - user\_id (Foreign Key para a tabela de Usuários Comuns)
  - event\_id (Foreign Key para a tabela de Eventos)
  - registrationDate
- **Tabela de Certificados:**
  - id (Primary Key)
  - user\_id (Foreign Key para a tabela de Usuários Comuns)
  - event\_id (Foreign Key para a tabela de Eventos)
  - dateIssue
  - description