



Digital
College

APLICAÇÃO WEB - HACKATHON



ABSTRACT

Criar uma aplicação web para gerenciamento de salas/espços para eventos (reuniões, treinamentos, conferências, etc).

A aplicação consiste em funcionalidades que permitem agendar **eventos** em **salas** com uma pessoa como **responsável**.

Num primeiro momento é notável a necessidade de pelo menos três entidades:

- **Sala:** espaço onde será realizado o *evento*;
- **Evento:** atividade que será realizada dentro de uma *sala*;
- **Responsável:** pessoa responsável pelo *evento* que acontecerá em uma determinada *sala*.

REQUISITOS MÍNIMOS

A aplicação deverá ter

- Telas principais:
 - Autenticação / Login (mesmo que não funcione)
 - Dashboard (com informações gerais sobre as quantidades de Salas, Eventos, Usuários, etc)
 - CRUD de Salas
 - CRUD de Eventos
 - CRUD de Usuários
- Documentação:
 - Um tutorial mínimo sobre como colocar para executar a aplicação
 - Uma documentação de funcionamento da API (se houver)
 - Um documento especificando as tecnologias utilizadas e suas versões
- Github/Gitlab:
 - Todo o código da aplicação deverá ser versionado via GIT e de preferência o trabalho em equipe deve ser através de Pull Requests / Merge Requests, ou seja, trabalhando com branches separados.
- Fluxo de desenvolvimento:
 - Cada time poderá escolher se vai ou não utilizar uma ferramenta de gestão das atividades, tais como, trello, github projects, github issues, Jira, etc.

STACK DE TECNOLOGIAS SUGERIDAS

O time de engenharia terá liberdade em escolher as tecnologias desde que as mesmas estejam dentro do ecossistema de linguagens de programação web de código aberto.

Preparamos uma lista de possíveis tecnologias a serem usadas:

Frontend:

- HTML
- CSS
 - puro sem bibliotecas
 - Bootstrap / Tailwind / Chakra
 - Material UI / AntDesign
 - AnimateCSS
 - outras bibliotecas também são permitidas.
- Javascript
 - VanillaJS (puro, sem frameworks)
 - ReactJS
 - AngularJS
 - VueJS
 - outras bibliotecas ou frameworks também são permitidas.

Backend:

- Javascript/NodeJS
 - puro sem frameworks/libraries
 - ExpressJS / NextJS / outras
- PHP
- Ruby on Rails
- etc

Banco de Dados:

- MySQL / MariaDB / PostgreSQL
- MongoDB / DynamoDB / CouchBase / Cassandra / outros
- Firebase

REQUISITOS DETALHADOS

CRUD de Usuários:

A aplicação deverá ter um ambiente onde seja possível gerenciar (CRUD) os **usuários** que poderão agendar **eventos** na sala.

Um usuário deve possuir no mínimo as informações:

- nome
- email
- telefone
- senha
- foto
- data de criação
- data de edição

Criar Usuário:

- O sistema deve permitir a criação de novos usuários.
- Os campos obrigatórios para criação de um usuário devem incluir nome, e-mail e senha.
- O sistema deve verificar se o e-mail fornecido já está sendo utilizado por outro usuário.
- O sistema deve garantir que a senha atenda a critérios mínimos de segurança, como comprimento mínimo e uso de caracteres especiais.

Detalhes do Usuário:

- O sistema deve permitir a visualização de informações de um usuário existente.
- Os usuários devem ser identificados de forma única, geralmente por um ID.
- As informações do usuário que podem ser exibidas devem incluir nome, e-mail, data de criação e outras informações relevantes.

Atualizar Usuário:

- O sistema deve permitir a atualização das informações de um usuário existente.
- Os campos que podem ser atualizados devem incluir nome, e-mail e senha.
- O sistema deve verificar se o novo e-mail fornecido já está sendo utilizado por outro usuário.
- O sistema deve permitir a atualização da senha apenas se a nova senha atender aos critérios mínimos de segurança.

Excluir Usuário:

- O sistema deve permitir a exclusão de um usuário existente.
- A exclusão de um usuário deve ser uma ação irreversível.

- A tela de excluir deverá ter um passo de confirmação, garantindo que a exclusão realmente precisa acontecer, por exemplo, "Deseja realmente excluir este usuário?"

Listar Usuários:

- O sistema deve fornecer uma funcionalidade para listar todos os usuários cadastrados, pode utilizar uma tabela ou lista para isso.
- A lista de usuários deve ser apresentada de forma clara e organizada, exibindo informações básicas de cada usuário, como nome e e-mail.
- A partir da listagem dos usuários deve-se poder escolher uma ação para cada usuário: Detalhar, Editar e Excluir.

Pesquisar Usuário:

- O sistema deve permitir a pesquisa de usuários com base em critérios específicos, como nome, pode ser um campo de buscar/filtro da tela de listar os usuários.
- Os resultados da pesquisa devem ser apresentados de forma clara e organizada, semelhante à listagem de usuários.

Autenticação e Autorização (recurso extra não obrigatório):

- O sistema deve fornecer um mecanismo de autenticação para permitir que os usuários façam login em suas contas.
- A autenticação pode ser baseada em nome de usuário e senha ou em outro método seguro, como autenticação de dois fatores.
- O sistema deve ter mecanismos de autorização para controlar o acesso às funcionalidades de CRUD de usuários, garantindo que apenas usuários autorizados possam executar ações específicas.

CRUD de salas:

A aplicação deverá ter também uma seção para gerenciamento de **Salas**, pois é onde acontecerão os **eventos** que serão agendados por alguém.

Uma sala deve possuir no mínimo as informações:

- nome
- capacidade máxima de pessoas
- recursos disponíveis
- data de criação

Criar Sala:

- O sistema deve permitir a criação de novas salas.
- Os campos obrigatórios para criação de uma sala devem incluir nome da sala, capacidade máxima e recursos disponíveis.
- A capacidade máxima deve ser um número inteiro que representa o número máximo de pessoas que a sala pode acomodar.

- Os recursos disponíveis podem incluir equipamentos audiovisuais, mobiliário, acesso à internet, entre outros.

Detalhar Sala:

- O sistema deve permitir a visualização das informações de uma sala existente.
- As salas devem ser identificadas de forma única, geralmente por um ID.
- As informações da sala que podem ser exibidas devem incluir nome da sala, capacidade máxima, recursos disponíveis e outras informações relevantes.

Atualizar Sala:

- O sistema deve permitir a atualização das informações de uma sala existente.
- Os campos que podem ser atualizados devem incluir nome da sala, capacidade máxima e recursos disponíveis.
- A capacidade máxima deve ser um número inteiro válido e positivo.
- Os recursos disponíveis podem ser modificados para refletir alterações na disponibilidade de equipamentos ou outros recursos.

Excluir Sala:

- O sistema deve permitir a exclusão de uma sala existente.
- A exclusão de uma sala deve ser uma ação irreversível.
- Ao excluir uma sala, o sistema deve verificar se existem eventos ou reservas associados (que ainda não aconteceram) a ela e fornecer opções para reatribuir esses eventos ou cancelá-los.

Listar Salas:

- O sistema deve fornecer uma funcionalidade para listar todas as salas cadastradas.
- A lista de salas deve ser apresentada de forma clara e organizada, exibindo informações básicas de cada sala, como nome da sala, capacidade máxima e recursos disponíveis.

Pesquisar Sala:

- O sistema deve permitir a pesquisa de salas com base em critérios específicos, como nome da sala, capacidade máxima ou recursos disponíveis.
- Os resultados da pesquisa devem ser apresentados de forma clara e organizada, semelhante à listagem de salas.

CRUD de Eventos/Reservas:

Um **evento** é o ponto chave da aplicação, pois o objetivo principal da mesma é garantir uma agenda integrada entre as **salas**, ou seja, sem duplicação de **eventos** ou sem o risco de dois ou mais **eventos** estarem reservados para uma mesma **sala** no mesmo **dia e horário**.

Criar Evento:

- O sistema deve permitir a criação de novos eventos relacionados a salas.

- Os campos obrigatórios para a criação de um evento devem incluir nome do evento, data e hora de início, data e hora de término, sala associada e organizador/responsável (usuário) do evento.
- O sistema deve garantir que a data e hora de término sejam posteriores à data e hora de início.
- O sistema deve verificar se a sala associada está disponível no horário especificado e se a capacidade da sala é suficiente para acomodar o número de participantes esperados.

Detalhar Evento:

- O sistema deve permitir a leitura das informações de um evento existente.
- Os eventos devem ser identificados de forma única, geralmente por um ID.
- As informações do evento que podem ser exibidas devem incluir nome do evento, data e hora de início, data e hora de término, sala associada, organizador do evento e outras informações relevantes.

Atualizar Evento:

- O sistema deve permitir a atualização das informações de um evento existente.
- Os campos que podem ser atualizados devem incluir nome do evento, data e hora de início, data e hora de término, sala associada e organizador do evento.
- O sistema deve garantir que a data e hora de término sejam posteriores à data e hora de início.
- O sistema deve verificar se a sala associada está disponível no horário especificado e se a capacidade da sala é suficiente para acomodar o número de participantes esperados.

Excluir Evento:

- O sistema deve permitir a exclusão de um evento existente.
- A exclusão de um evento deve ser uma ação irreversível.
- Ao excluir um evento, o sistema deve liberar a sala associada e fornecer opções para notificar os participantes sobre o cancelamento. (A notificação não precisa acontecer, apenas uma mensagem na tela informando isso para quem estiver excluindo o evento)

Listar Eventos:

- O sistema deve fornecer uma funcionalidade para listar todos os eventos cadastrados.
- A lista de eventos deve ser apresentada de forma clara e organizada, exibindo informações básicas de cada evento, como nome do evento, data e hora de início e sala associada.

Pesquisar Evento:

- O sistema deve permitir a pesquisa de eventos com base em critérios específicos, como nome do evento, data, sala associada ou organizador do evento.
- Os resultados da pesquisa devem ser apresentados de forma clara e organizada, semelhante à listagem de eventos.

DESIGN PARA ORIENTAÇÃO

Preparamos algumas telas que poderão auxiliar o time de engenharia no momento de desenvolvimento, não se limitando ao que está sendo apresentado, essa é apenas uma sugestão.

<https://drive.google.com/drive/folders/1IQzygyga1IGBWRvlfIS3UjPBmA8rrodh?usp=sharing>

