

## Vaga de estágio para Dev. Full Stack

Já pensou em fazer parte de um time de desenvolvimento de grande impacto no mercado? Você poderá fazer parte de uma equipe habilidosa e comprometida em desenvolver o melhor para seus clientes e parceiros.

A UX Software, empresa de desenvolvimento do grupo Alan Araujo Engenharia de Telecomunicações, está com vagas de estágio abertas. E para conseguir o seu passaporte, temos um desafio em forma de linhas de código para você.

## Instruções para Front e Back-End

Neste projeto avaliaremos como os desenvolvedores criam, estruturam e organizam os códigos, bem como o domínio dos conceitos gerais das tecnologias sugeridas.

💡 Dica: Lembre de focar nas funcionalidades da aplicação.

## Tecnologias sugeridas:

- ReactJS ou NextJS;
- Node.js com Typescript.

## Descrição:

**Situação:** Imagine que você foi contratado como desenvolvedor para criar uma aplicação web para uma academia, esta academia precisa de um software para cadastrar seus administradores, colaboradores e clientes, podendo ainda cadastrar treinos. Os clientes da academia, por sua vez, devem possuir uma página para *logar* e visualizar seus treinos.

### Requisitos para a aplicação web:

#### Visão do personal:

##### *Tela de Login*

- A plataforma terá uma tela de Login;
- Na tela de Login deverá ser visível os campos de email e senha;
- Quando for feito o login, deverá ser redirecionado para a tela Home

##### *Tela de Home*

- Na *Home* deverá ser visível a listagem dos usuários e treinos, ou botões para iniciar tais fluxos;

##### *Tela de Treino*

- Na tela de treino deve ser possível listar, cadastrar, editar e apagar treinos;
- Na listagem de treinos deve ser possível;
  - Ver o nome do treino,
  - e botões para: Editar treino e Excluir treino
- No cadastro do treino precisaremos adicionar:
  - Um nome para aquele treino(ex: iniciante)
  - uma lista de exercícios:
    - Nome do exercício (ex. Agachamento);
    - Imagem do exercício
    - Descrição do exercício(ex. Você deve se agachar);
    - Quantidade de repetições(ex. 3 vezes);
    - Quantas execuções por repetição(ex. 15 vezes)
    - Intervalo deste exercício em segundos(ex. 30s)
- Na edição de treino deve ser possível editar todos os dados cadastrados dos treinos;

#### *Tela de Usuários*

- Na tela de usuários deverá ser possível fazer o CRUD dos clientes;
- Na listagem de Usuários deve ser possível ver:
  - Foto do usuário;
  - Nome do cliente;
  - Número de telefone;
  - email;
  - e ações:
    - Adicionar dicas: Onde deve ser possível adicionar dicas aos clientes e listar as dicas também(Dica: Use um modal com um campo e um local listando elas)
    - Ver treino: que mostre o treino deste cliente(todos os dados e quais dias ele realizou o treino) ;
    - Editar usuário: pode editar todos os dados adicionados no cadastro;
    - Excluir usuário.
- No cadastro de usuário deveremos receber os seguintes dados:
  - Foto do usuário;
  - Email;
  - Telefone;
    - O campo de telefone deve ter uma máscara: ( \_\_ ) \_\_\_\_ - \_\_\_\_
  - Senha do usuário;
  - Treino que deve ser atrelado àquele usuário(Dica: pode usar um campo select que pega todos os treinos)
- Opção de Logout
  - Retornar ao login

#### **Visão do cliente:**

##### *Tela de Login*

- A plataforma terá uma tela de Login;

- Na tela de Login deverá ser visível os campos de email e senha;
- Quando for feito o login, deverá ser redirecionado para a tela Home

#### *Tela de Home*

- Na *Home* deverá ser visível a listagem dos treinos/treino do cliente, ou botões para iniciar tais fluxos;
- Na listagem de treinos deve ser possível;
  - Ver o nome do treino,
  - e botões para: Ver treino completo
  - deve ser possível adicionar os dias em que o treino foi realizado;ex(um local com um calendário para que ele possa marcar os dias que o treino foi realizado)
- Opção de Logout
  - Retornar ao login

## API'S sugeridas:

Ordenando estados por nome:

<https://servicodados.ibge.gov.br/api/v1/localidades/estados?orderBy=nome>

Ordenando as cidades pelo estado ordenando por nome:

<https://servicodados.ibge.gov.br/api/v1/localidades/estados/CE/municipios?orderBy=nome>

## Métodos importantes para uma API:

- Get (Consultas/Solicitações)
- Post (Criar/Adicionar)
- Put (Atualização de dados)
- Delete (Excluir dados)

## Inspirações para o front-end:

<https://drive.google.com/drive/folders/17jM9H5qJddfn1CCiwl1589UefbVjsjn0?usp=sharing>

## Dúvidas?

E-mail: [softwareux@gmail.com](mailto:softwareux@gmail.com)

WhatsApp: [+55 88 99314-0667](https://api.whatsapp.com/send?phone=5588993140667) - Alice Sousa