

# Aplicação 1: cadastro usuário WEB

## Nível 1 (Backend/Fullstack/Frontend)

Fazer uma aplicação na web com cadastro de usuários.

- Dados do cadastro:
  - nome
  - email
  - endereço do usuário
    - País
    - Estado
    - Município
    - CEP
    - Rua
    - Número
    - Complemento
  - CPF
  - PIS
  - Senha
- Enquanto o usuário não estiver “logado”, apresentar uma página que diga “Olá visitante” que:
  - Apresente campo de login por email, CPF ou PIS + senha
  - Apresente botão que leva para página de cadastro de usuário
- Se o usuário estiver “logado”, ao invés da página do “Olá Visitante”, apresentar “Olá {NOME\_DO\_USUARIO}”. Apresentar:
  - botão de logout
  - botão que leve para a página de edição dos dados cadastrais. Nessa página apresentar botão de remoção do usuário que o desloga em seguida.

## Nível 1 (Bônus):

- Use variáveis de ambiente
- Use Docker
- Suba a aplicação(back e front) no heroku ou similar.

## Nível 2 (Front/Fullstack)

Criar frontend que realize as funcionalidades acima em SPA JS - React/Vue/Angular/Svelte/sei-la-o-q-mais-estiver-na-moda

## **Nível 2 (Backend/Fullstack)**

- Criar API REST completa para manipulação dos usuários
- Realizar autenticação via token JWT

## **Nível 2 (Bônus):**

- Torne o front adaptado para web e mobile.
- Escreva um arquivo de docker-compose contendo a configuração dos 3 serviços(back, banco, front). Torne o banco de dados acessível apenas para o back.

## **Nível 3 (Backend/Fullstack)**

- Implementar autenticação OAUTH2 usando uma conta gratuita do Auth0
  - Login por github
  - Login por gmail (não o email)
  - Login por email, CPF ou PIS + senha

## **Nível 3 (Front)**

- Implementar autenticação com firebase

Sua aplicação tem os seguintes requisitos

- validar os dados de entrada do usuário adequadamente
- armazenar as informações em banco de dados relacional
- armazenar a sessão do usuário em cookie
- apresentar testes unitários