

# Técnologo em Análise e Desenvolvimento de Sistemas

## Tópicos em Desenvolvimento de Aplicativos

### Trabalho III

#### Objetivo

Criar uma aplicação mobile utilizando **React Native** e **Expo** para interagir com a API disponibilizada em <https://simple-api-ngvw.onrender.com>. O aplicativo deve implementar as funcionalidades descritas nos endpoints da API e seguir as boas práticas de desenvolvimento mobile.

---

#### Requisitos Funcionais

##### 1. Autenticação de Usuários

###### 1.1. Cadastro de Usuário

- O aplicativo deve permitir que novos usuários criem contas fornecendo:
  - Nome (**name**)
  - E-mail (**email**)
  - Senha (**password**)

###### 1.2. Login de Usuário

- O aplicativo deve permitir que usuários façam login utilizando:
    - E-mail
    - Senha
  - Após o login, o **JWT de autenticação** recebido deve ser salvo localmente e utilizado nas requisições protegidas.
- 

##### 2. Gerenciamento de Usuários

###### 2.1. Listagem de Usuários

- Exibir a lista de usuários cadastrados na API.
  - A listagem deve ser paginada, enviando os parâmetros **limit** e **page** como query na requisição.
  - O JWT de autenticação deve ser enviado para acesso ao endpoint.
- 

##### 3. Gerenciamento de Posts

###### 3.1. Criar Post

- O usuário autenticado deve poder criar um post enviando os seguintes dados:
  - Título (**title**)
  - Conteúdo (**content**)
  - Foto (arquivo de imagem)
- O envio deve ser feito em formato **form-data**.

### 3.2. Listar Posts

- Exibir a listagem de posts existentes.
- A listagem deve ser paginada, enviando os parâmetros **limit** e **page**.
- As imagens dos posts devem ser carregadas diretamente do link do bucket S3 retornado pela API.
- O JWT de autenticação deve ser enviado para acesso ao endpoint.

### 3.3. Excluir Post

- O usuário autenticado deve poder excluir seus posts utilizando o **id** do post como parâmetro na rota.

---

## Requisitos Não Funcionais

- **Design Responsivo:** A interface deve ser adaptada para diferentes tamanhos de tela.
- **Gerenciamento de Estado:** Utilize **Redux** ou **Context API** para gerenciar o estado global.
- **Armazenamento Local:** Salve o token JWT localmente utilizando o AsyncStorage do Expo.
- **Boas Práticas:** Seguir padrões de componentização, organização de código, e nomeação de variáveis.

---

## Entrega até 16/06/2025

### Estrutura da Aplicação

A aplicação deve seguir a seguinte estrutura de funcionalidades:

1. Tela de Cadastro
2. Tela de Login
3. Tela de Listagem de Usuários
4. Tela de Listagem de Posts
5. Tela de Criação de Posts
6. Tela de Excluir Post (confirmar exclusão antes de apagar)