

# Desafio final



Rocketseat

Nesse desafio você irá construir uma aplicação completa envolvendo back-end com **NodeJS**, front-end com **ReactJS** e mobile com **React Native**.

A aplicação consiste em um app de controle de pedidos por delivery de uma pizzeria. O aplicativo móvel será utilizado pelos clientes da pizzeria para realizar pedidos e a versão web será utilizada para controle e registro dos pedidos realizados.

Nem todas funcionalidades da aplicação estarão descritas nesse documento, utilize de sua criatividade para tomar algumas decisões que possam deixar a aplicação mais acessível e com uma melhor usabilidade.

## Regras gerais

- O back-end deve ser uma API REST com o framework em NodeJS de sua preferência.
- O front-end deve utilizar ReactJS configurado com o create-react-app;
- O app mobile deve utilizar React Native configurando com React Native CLI;

## Detalhes da aplicação

A aplicação chamada “Pizza Don Juan” será utilizada para realizar e controlar pedidos de delivery. A aplicação não suportará mais restaurantes sendo feita apenas para essa pizzeria.

A aplicação móvel será mais completa contendo todas páginas necessárias para um cliente realizar um pedido na pizzeria, selecionar sabor, tamanho e fechar pedido. O cliente também poderá visualizar seu histórico de pedidos e todo cardápio da pizzeria.

A versão web da aplicação será utilizada apenas para monitorar os pedidos conforme forem chegando e não terá versão para o cliente fazer pedido, apenas para administradores consultarem os pedidos já feitos.

\* Todas imagens e layout da aplicação estão em anexo junto com esse PDF





## Back-end (NodeJS)

No back-end da aplicação você pode utilizar o framework NodeJS que preferir, mas sempre criando uma API REST. Além disso, utilize banco relacional (SQL).

### Usuário / autenticação

- Deve ser possível o usuário se autenticar e criar sua conta na aplicação;
- O usuário pode ser administrador ou cliente, você pode adicionar uma coluna com o tipo de usuário na tabela de usuários;
- Quando um usuário se cadastra ele é automaticamente um cliente;
- Não deve ter cadastro de usuário administradores, crie um usuário administrador manualmente para utilizar a versão web do sistema;

### Cardápio

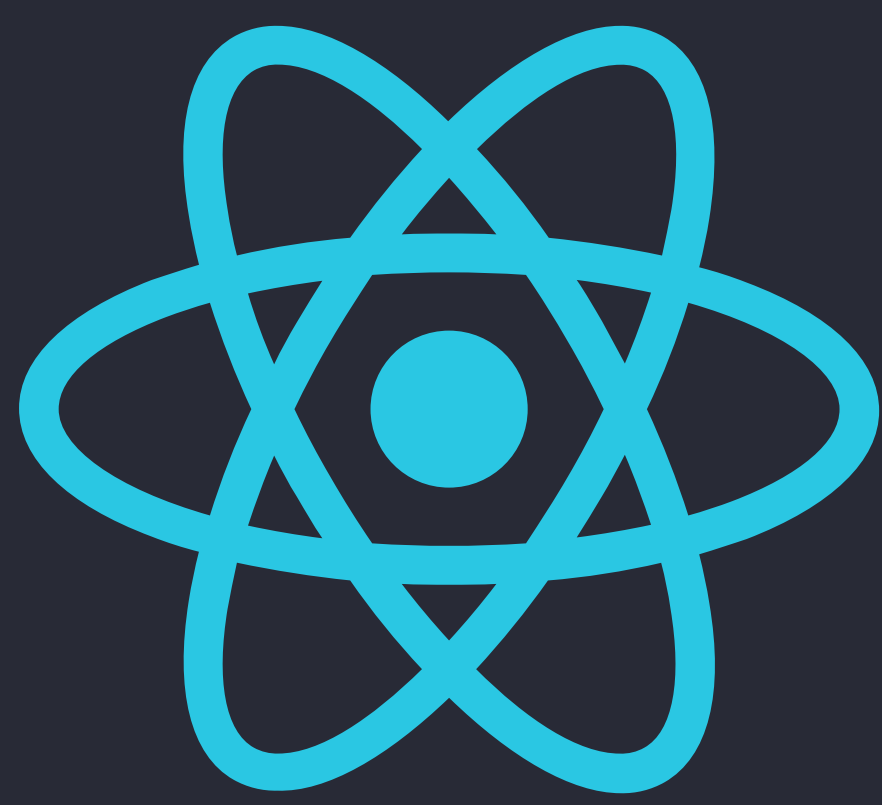
- O cardápio deve ser cadastrado através de API, você deve ter tabelas para armazenar produtos (pizza/bebidas/massas), tipo de produtos (sabor de pizza/marca de refrigerante) e tamanho do produto (cm para pizzas/ml para bebidas/P,M ou G para outros);
- Não existirá uma interface para cadastro dessas informações, mas você pode cadastrá-las usando o próprio Insomnia.

- **O relacionamento ficará da seguinte forma:**

1 produto tem N tipos

1 tipo tem N tamanhos

- O **relacionamento** entre tipo e tamanho define o valor do produto, ou seja, uma pizza média de 4 queijos pode custar diferente de uma pizza média de calabresa.

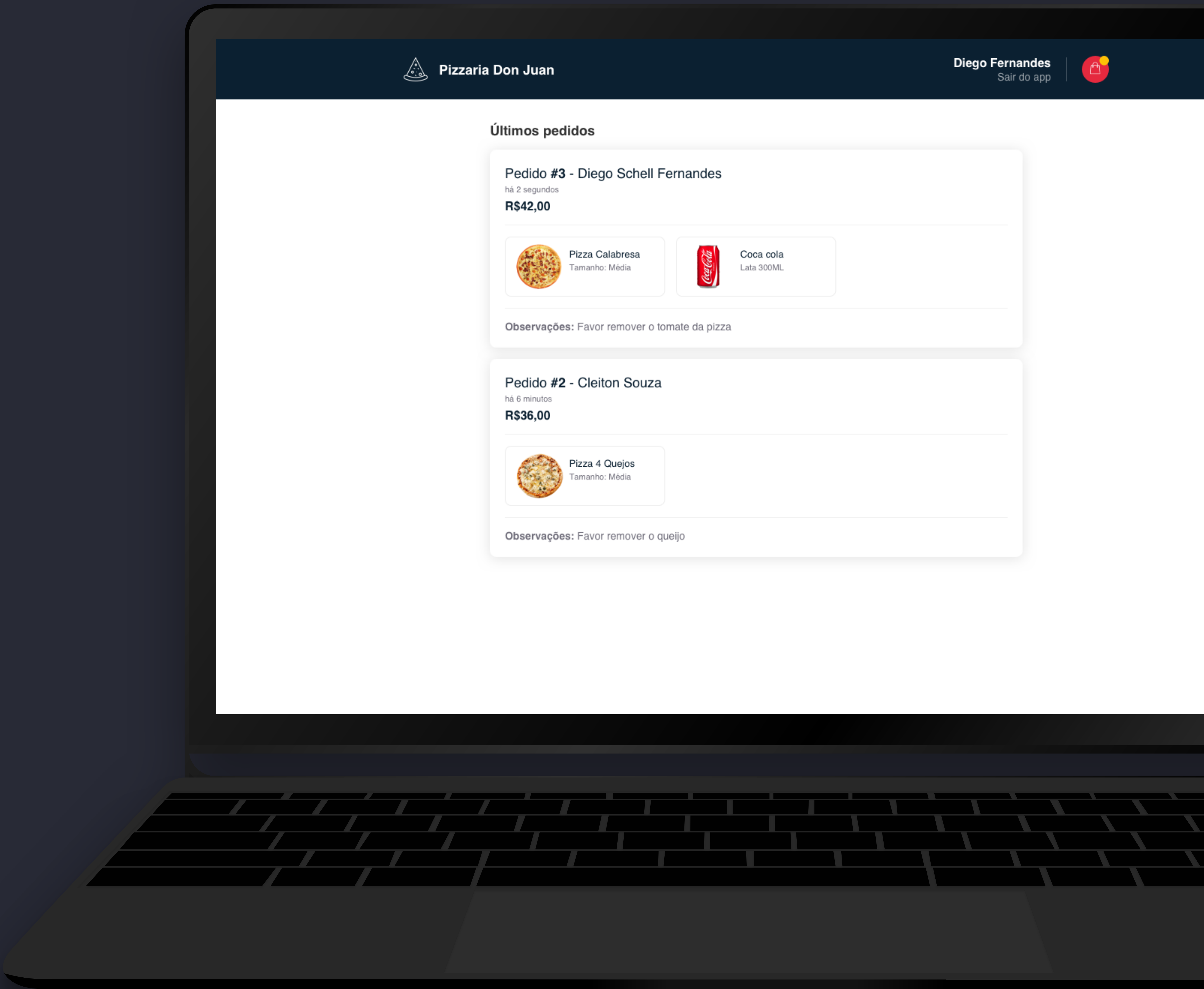


# Front-end (ReactJS)

No front-end da aplicação utilize o create-react-app, estilização através de Styled Components, Redux, Redux Saga.

## Telas da aplicação

- Login;
- Pedidos

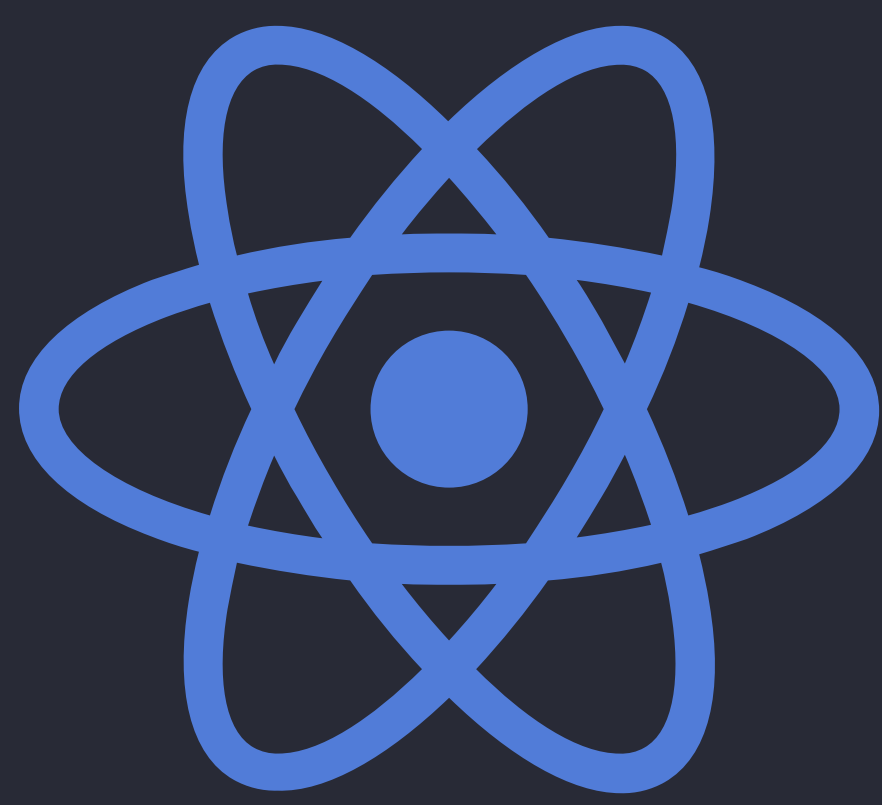


## Detalhes

- Não permita que clientes loguem na aplicação web, apenas administradores.

Todas imagens e layouts dessa aplicação, assim como cores, tamanhos de fonte, espaçamentos podem ser visualizados através do arquivo “index.html” na pasta “Mockup” incluída com esse PDF.



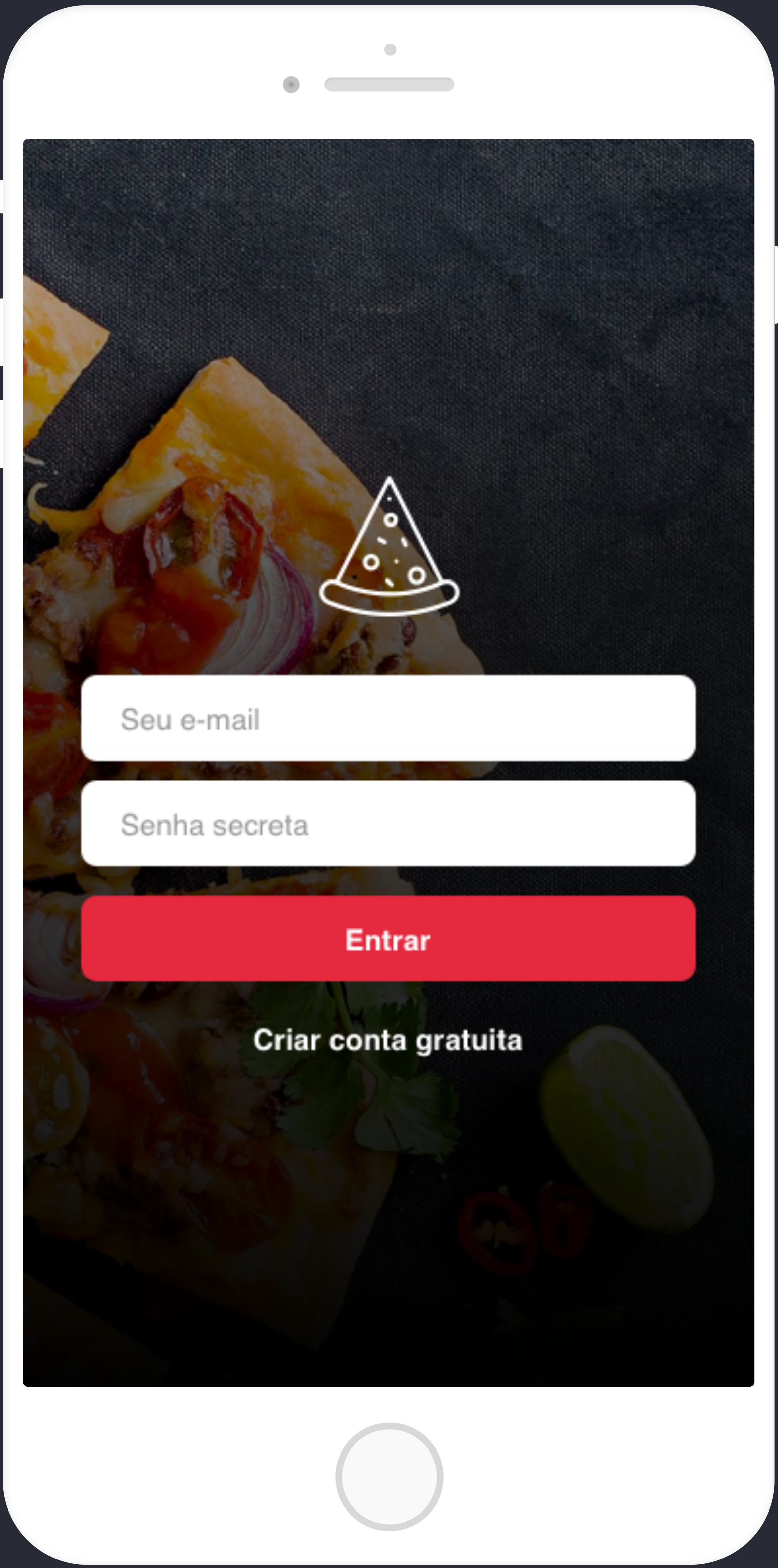


# Mobile (React Native)

No mobile da aplicação utilize o React Native CLI, estilização através de Styled Components, React Navigation, Redux, Redux Saga.

## Telas da aplicação

- Login;
- Cadastro;
- Cardápio;
- Histórico de pedidos;
- Selecionar tipo;
- Selecionar tamanho;
- Carrinho;
- Finalizar pedido;



## Detalhes

- Na finalização do pedido, ao informar o CEP, o restante dos campos devem ser informados de forma automática;
- Não permita que administradores loguem no aplicativo mobile.

Todas imagens e layouts dessa aplicação, assim como cores, tamanhos de fonte, espaçamentos podem ser visualizados através do arquivo “index.html” na pasta “Mockup” incluída com esse PDF.