

ENGENHARIA DA COMPUTAÇÃO

Gabriel Rocha Monaco | RA: 171092

Raphael Augusto Santos | RA: 200361

André Luiz dos Santos Leite | RA: 200036

Willians Ferreira | RA: 190520

**Desenvolvimento Web**

**Moriya Sushi**

Orientador: Profª Angelina Vitorino de Souza Melaré

**Sorocaba**

**Junho, 2022**

Sumário

[1. PROPOSTA 3](#_gjdgxs)

[2. INTERFACE 3](#_30j0zll)

[3 RECURSOS/ FERRAMENTAS / FRAMEWORKS / PACOTES UTILIZADOS 5](#_1fob9te)

[4 REFERÊNCIAS 6](#_3znysh7)

# 1. PROPOSTA

O projeto consiste em um site de um restaurante fictício chamado Moriya Sushi, no qual o usuário realiza pedidos através das opções determinadas pelo site, com opções de entrega delivery para endereço e escolha da forma de pagamento. No backend do site, foi criada uma API na qual recebe informações enviadas do site, podendo até salvar informações caso o usuário tenha permissões para isto.

O site foi criado utilizando a biblioteca React, com a utilização de componentes e a API foi criada usando nodemon e express.

O site tem uma homepage, com menus que recebem dados do banco de dados. Usuários cadastrados podem fazer pedidos através do site, que verifica se o usuário está logado no sistema, caso esteja este pode fazer pedidos na página específica para isto. A tela de pedido depende dos valores existentes no banco de dados, que podem ser alterados ou adicionados pelo usuário administrador. Dados dos usuários, além dos dados de pratos e pedidos feitos, são armazenados utilizando MongoDB.

# 2. INTERFACE

Ao entrar no site, o usuário é apresentado com a homepage, que é exemplificada pela figura 1.

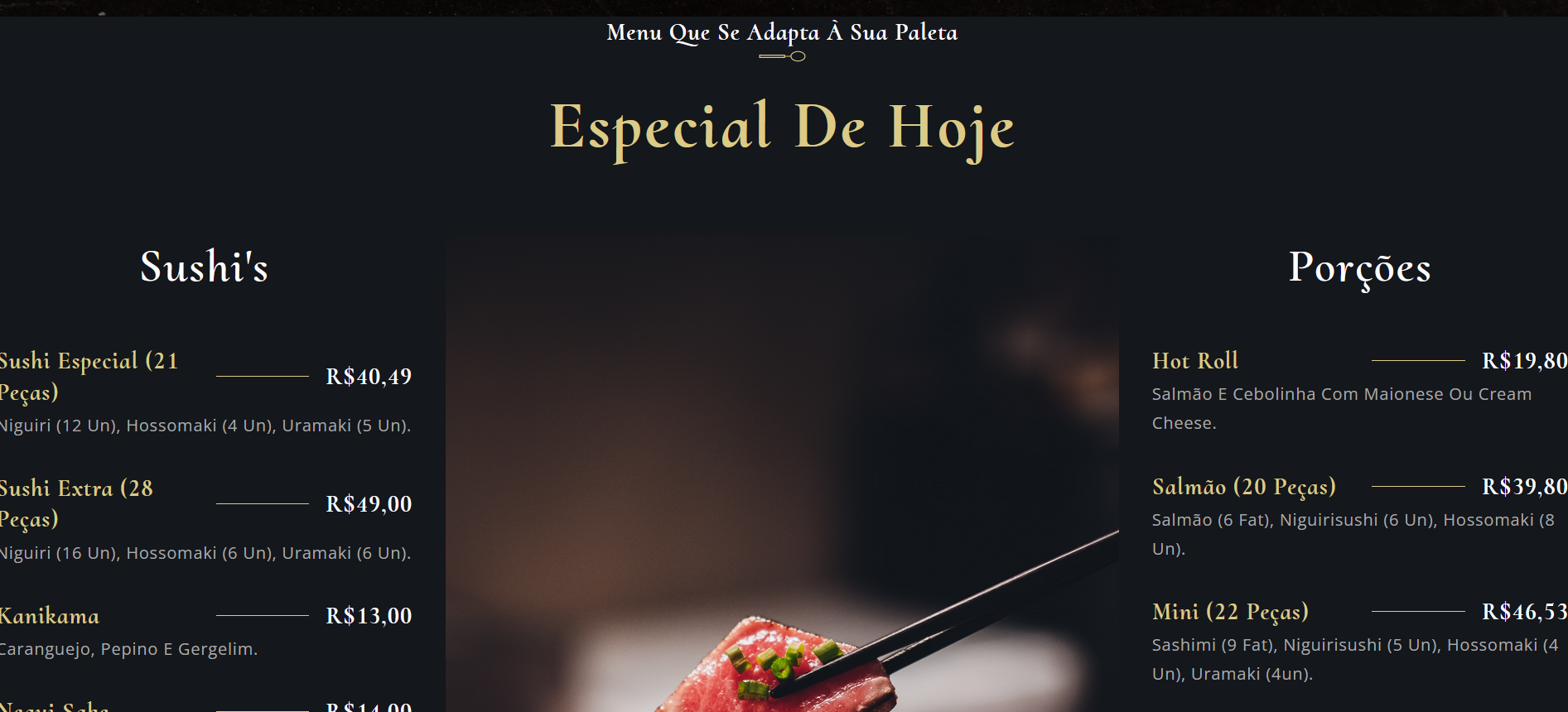
Figura 1: Homepage



Fonte: Autoria Própria (2022)

Ao navegar para baixo no mesmo site, o usuário é apresentado pelo menu, que é demonstrado pela figura 2. Este menu apresenta os primeiros 5 itens do menu, que são enviados pela API.

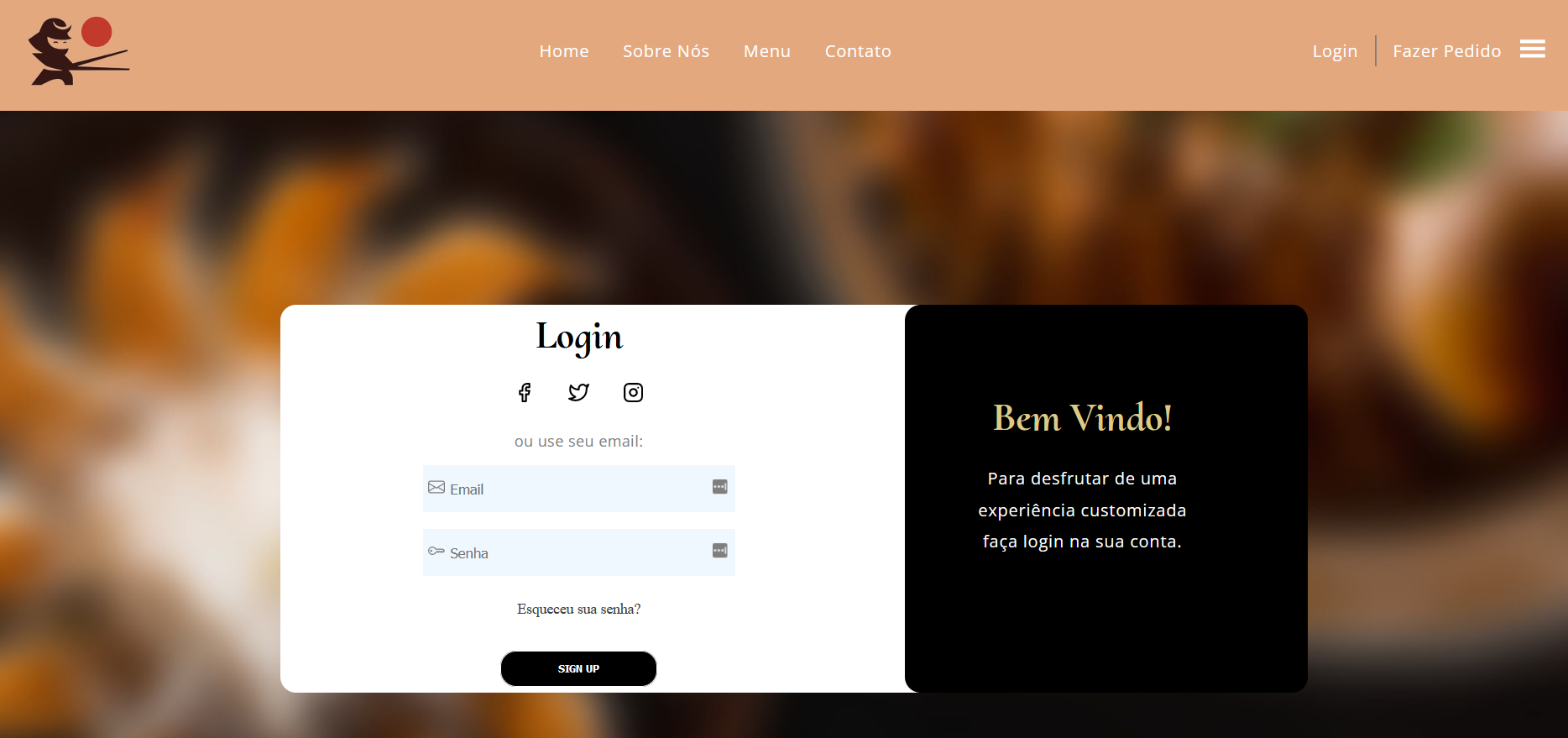
Figura 2: Componentes de menu dinâmico



Fonte: Autoria Própria (2022)

O usuário pode fazer o login, e ao tentar fazer, o usuário é apresentado pela seguinte tela, demonstrada pela figura 3. É possível acessar essa tela, também, pela rota ***/login***.

Figura 3: Tela de Login



Fonte: Autoria Própria (2022)

Quando o usuário acessa o menu de pedidos ou usa a rota ***/pedir***, a tela de pedidos é apresentada. Nesta interface, o usuário pode efetuar os pedidos.

Figura 4: Menu de pedidos

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Fonte: Autoria Própria (2022)

Após efetuar o pedido, o usuário é apresentando a tela de revisão de pedido, que é demonstrado pela figura 5. O endpoint dessa tela é ***/pedir/efetuado***.

Figura 5: Revisão do pedido efetuado

Tela de jogo de vídeo game

Descrição gerada automaticamente com confiança média

Fonte: Autoria Própria (2022)

# 3 RECURSOS/ FERRAMENTAS / FRAMEWORKS / PACOTES UTILIZADOS

Entre as ferramentas utilizadas, estão o Visual Studio Code - ambiente de desenvolvimento principal -, e o Postman, que foi utilizado para auxiliar no desenvolvimento da API.

Para a persistência de dados, foi utilizado um banco de dados não relacional MongoDB na versão 5.0.8. Dentro do MongoDB, é criado um database chamado “sushi”, no qual contém as collections “admins”,

“opcaodepratos”, “pedidos” e “usuarios”.

Dentro de “admins”, cada documento, tem o e-mail de cada usuário que tem permisão de administrador no sistema.

A colection “opcaodepratos” é composta de documentos que contém o nome do prato, o tipo para determinar se é sushi, porção ou bebida, o preço do prato em questão e a descrição deste

Na collection “pedidos”, cada documento contém um array chamado “opcoes” que contém cada opção selecionada pelo cliente, com cada item mostrando nome do prato, a quantidade, valor unitário do prato e valor total da opção selecionada. Ainda dentro do documento de “pedidos”, existe o e-mail do cliente que fez o pedido, o endereço do cliente, a forma de pagamento e o preço total do pedido.

Na collection “usuarios”, cada documento contém o nome do usuário, a senha deste e o e-mail utilizado pelo usuário.

Os pacotes e dependências utilizados serão listados a seguir:

Para o desenvolvimento da API, foi utilizado a biblioteca cors na versão 2.8.5 para configurar as regras de recebimento das requisições, a biblioteca express da versão 4.18.1 e nodemon da versão 2.0.16 para a criação da API e a biblioteca mongoose da versão 6.3.3 para comunicação do banco de dados MongoDB.

Para a elaboração geral do site, foi utilizado a biblioteca “react” na versão 18.1.0. Para elaboração de formulários, foi utilizado a biblioteca “formik” na versão 2.2.9 e para o envio dos dados para a API foi utilizado a biblioteca “axios” na versão 0.27.2.

Foi utilizado o “react-router-dom” na versão 6.3.0 para criação das rotas do navegador, navegação automática e envio de informações através de componentes.

Para avisar o usuário de erros ou modificações executadas, foi usado o “react-toastify” na versão 9.0.4.

# 

# 4 REFERÊNCIAS

* W3SCHOOLS. Bootstrap Templates. Disponível em: https://www.w3schools.com. Acesso em: 9 jun. 2022.
* ALURA. Z-index: Como utilizar essa propriedade do CSS. Disponível em: https://www.alura.com.br. Acesso em: 5 jun. 2022.
* GETBEM. Block Element Modifier. Disponível em: https://www.w3schools.com. Acesso em: 9 jun. 2022.
* MONGOOSE. Gettin Started. Disponível em: https://www.w3schools.com. Acesso em: 9 jun. 2022.