**OrganicStaples React Website Starter Kit**

**Project Overview**

**OrganicStaples** is a React-based front-end and Node.js back-end website starter kit tailored for food brands. It includes key components for building a food-related website, such as a product listing page, category filters, hero sliders, and more. This project is ideal for beginners looking to explore React, frontend development, or full-stack applications using Node.js and PostgreSQL.

This project includes two main directories:

* frontend: The React frontend of the website.
* backend: The Node.js backend API for user registration, login, and product management.

**Features**

**Frontend (React)**

1. **Announcement Bar** – Display special announcements and promotions.
2. **Top Navigation Bar** – Contains brand logo and links for navigation.
3. **Hero Section** – A slider to showcase promotional banners with the react-slick package.
4. **Content Section** – Informative sections with images and text about the brand.
5. **Product Category Listing** – Displays categories of products with clickable links.
6. **Best Selling Products Section** – A slider to highlight featured products.
7. **Collage Section** – For displaying blog links or additional content.
8. **Newsletter Section** – Captures user input for subscribing to newsletters.
9. **Footer Section** – Contains brand logo, quick links, social media icons, and additional information.
10. **Products Page** – Shows a list of products, filterable by categories.
11. **Login and Register Pages** – User authentication pages for managing customer accounts.

**Backend (Node.js)**

* **User Authentication** – Includes user login, registration, password hashing (using bcryptjs), and session management.
* **Product Management** – API for handling product data.
* **PostgreSQL Database** – Backend integration with a PostgreSQL database for storing users, products, and other data.

**Project Structure**

The project is split into two main directories:

1. **frontend/** – Contains all the React frontend code.
2. **backend/** – Contains the Node.js and Express API for backend functionality.

java

Copy code

organicstaples-react-website-starter-kit/

│

├── frontend/

│ ├── public/

│ ├── src/

│ └── package.json

│

├── backend/

│ ├── config/

│ ├── controllers/

│ ├── models/

│ ├── routes/

│ └── package.json

│

├── .env.example

├── README.md

└── LICENSE

**Installation and Setup**

To get started with this project, follow the steps below.

**Prerequisites**

* Node.js (v14 or later)
* PostgreSQL database
* A modern web browser

**Cloning the repository**

git clone https://github.com/your-username/organicstaples-react-website-starter-kit.git

cd organicstaples-react-website-starter-kit

**Frontend Setup**

1. Navigate to the frontend folder and install the dependencies:

cd frontend

npm install

1. Start the development server:

npm start

This will run the app in development mode. Open http://localhost:3000 to view it in the browser.

**Backend Setup**

1. Navigate to the backend folder and install the dependencies:

cd ../backend

npm install

1. Run the backend server:

npm run dev

The backend server will run on http://localhost:5000.

**Running the Full Stack Application**

To run both the frontend and backend together, make sure both servers are running in separate terminal windows:

* Frontend: npm start (in the frontend directory)
* Backend: npm run dev (in the backend directory)

**Dependencies**

**Frontend (React)**

* [**react-slick**](https://www.npmjs.com/package/react-slick): For creating image sliders in the hero and best-selling product sections.
* [**slick-carousel**](https://www.npmjs.com/package/slick-carousel): Supporting CSS for the react-slick component.
* [**styled-components**](https://www.npmjs.com/package/styled-components): For styling components using tagged template literals.
* [**axios**](https://www.npmjs.com/package/axios): For making HTTP requests to the backend API.
* [**react-router-dom**](https://www.npmjs.com/package/react-router-dom): For handling routing and navigation between pages.

**Backend (Node.js)**

* [**pg**](https://www.npmjs.com/package/pg): PostgreSQL client for Node.js used in the backend to interact with the database.
* [**express**](https://www.npmjs.com/package/express): For building the backend API server.
* [**bcryptjs**](https://www.npmjs.com/package/bcryptjs): For hashing and validating passwords.

**License**

This project is licensed under the MIT License - see the <LICENSE> file for details.

**Contributions**

Contributions are welcome! Please feel free to open an issue or create a pull request with any improvements.

**Contact**

If you have any questions, feel free to reach out:

* Email: your-nethragovinda@gmail.com