STOPPER'S SHOP

14.01.2024

—

CODE TECHIES

Rishav (2110991165),
Riya (2110991186),
Ankit (2110991966)
BE- CSE
Group: G19
Submitted To – Ashwani Dubey Sir

Introduction

Stopper's Shop stands out as the preferred e-commerce destination, boasting a diverse array of high-quality products tailored for men, women, and children. The user-friendly site ensures a seamless shopping experience, enabling the effortless addition of preferred items to the cart. Across all categories, customers can make conscious choices, promoting a sustainable and stylish lifestyle through mindful shopping at Stopper's. The online store prides itself on offering thoughtfully curated products, contributing to both individual preferences and a greener future. At Stopper's Shop, every selection reflects a commitment to quality, style, and sustainability, making it a unique and distinguished shopping destination.

Objectives

1. Optimized User Experience: Prioritize an intuitive and user-friendly interface for a seamless and enjoyable shopping experience.

2. Sustainable Lifestyle Promotion: Encourage conscious and sustainable choices across diverse product categories, aligning with environmental awareness.

3. Technical Excellence and Scalability: Implement a robust technical foundation using React, MongoDB, and Express.js to ensure top-tier performance, security, and scalability.

4. Personalized Customer Interaction: Provide users with a personalized dashboard and effective communication channels for managing orders, tracking shipments, and gathering valuable feedback.

5. Effortless Navigation: Implement a navigation system that allows users to easily explore products, add items to the cart, and manage their preferences effortlessly.

Tech Stack Used

React:

React is used for building the user interface, providing a dynamic and responsive design. Its component-based architecture facilitates modular development, allowing for efficient management and scalability. React ensures a smooth and interactive user experience.

Bootstrap:

Bootstrap is utilized for styling the user interface. As a front-end framework, Bootstrap streamlines the styling process, offering pre-built components and a responsive grid system. This results in a visually appealing and consistent design across the website.

MongoDB:

MongoDB serves as the backend database for efficient data management. MongoDB's NoSQL approach provides flexibility in handling diverse product data, crucial for accommodating varying attributes. It offers scalability and seamless integration with the application.

Express.js:

Express.js handles server-side logic and API development. As a minimal and flexible Node.js framework, Express.js simplifies server-side development, ensuring efficient data processing and enhancing the overall performance of the application.

Node.js:

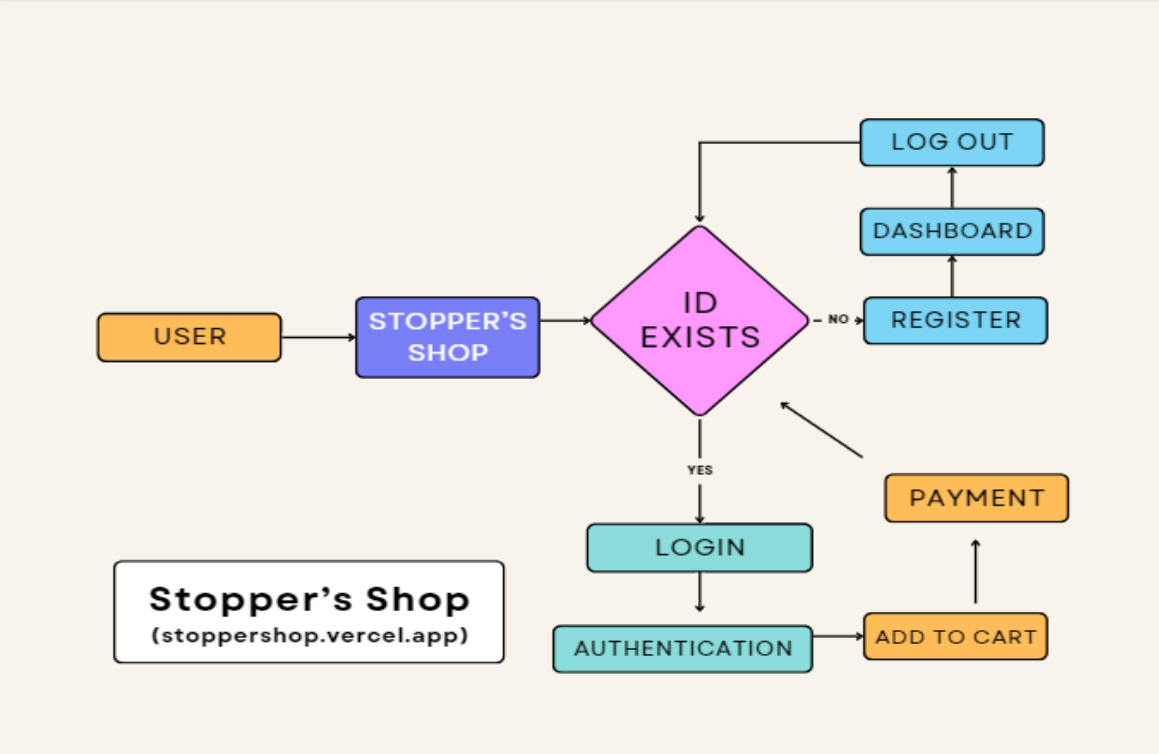
Node.js is the runtime environment for executing server-side JavaScript code. It unifies server-side and client-side

development, providing a seamless JavaScript environment. Node.js enhances the efficiency of server-side processes and contributes to the overall performance of the application.

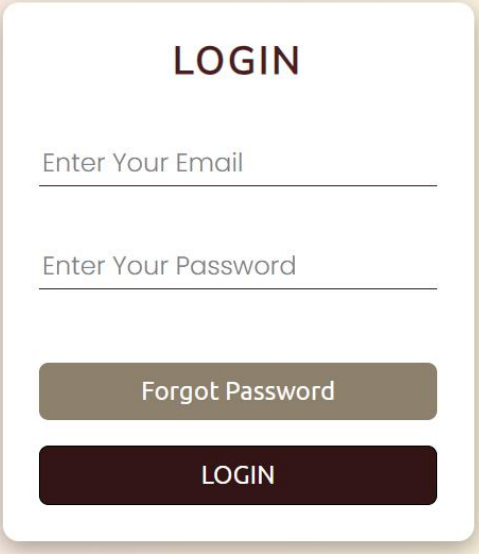
This integrated tech stack combines the strengths of React for the user interface, Bootstrap for styling, MongoDB for data management, and Express.js and Node.js for server-side logic, creating a powerful foundation for Stopper's Shop.

Project-Flow

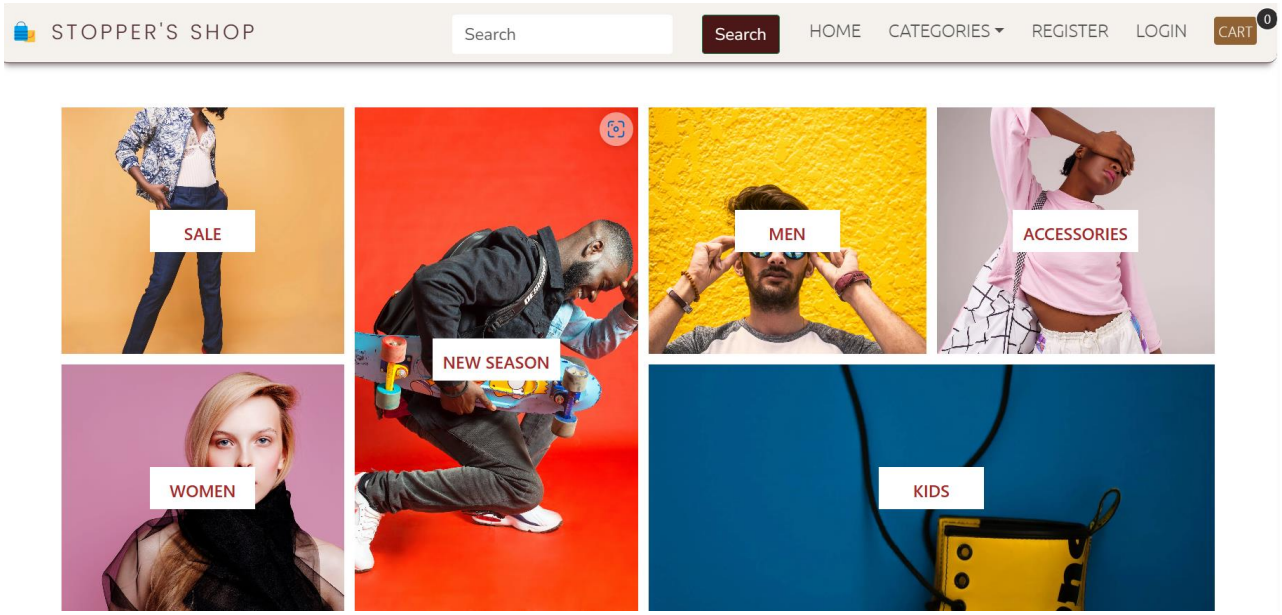
The flow of the application is in the following way-



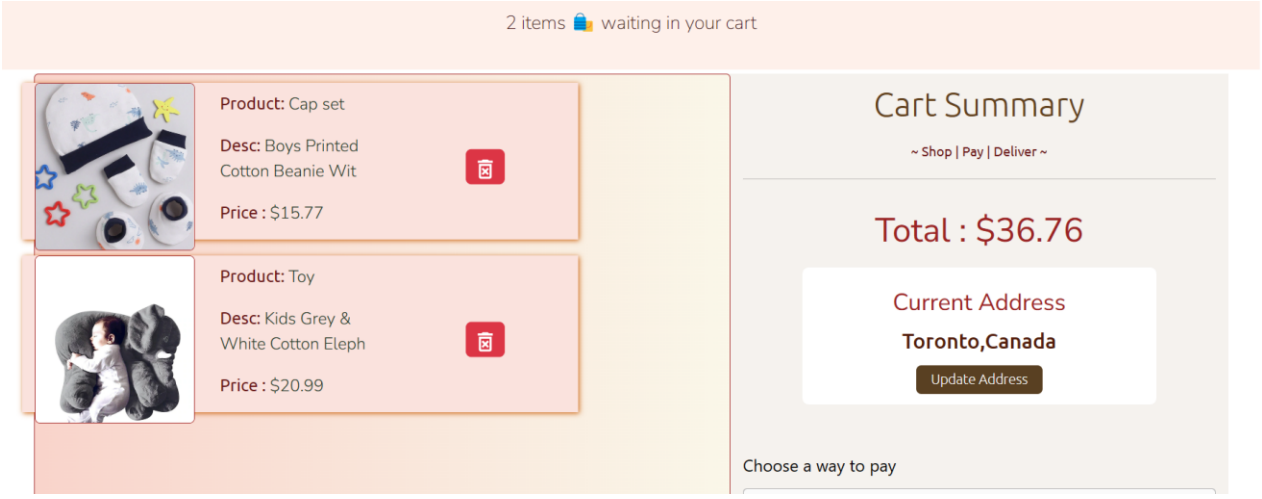
EXPLANATION



The Login component utilizes React, Axios, and state management to handle user authentication. It captures user email and password, sends a POST request to the server, and updates state upon successful login. The layout integrates with a navigation system and displays a simple form for user login.

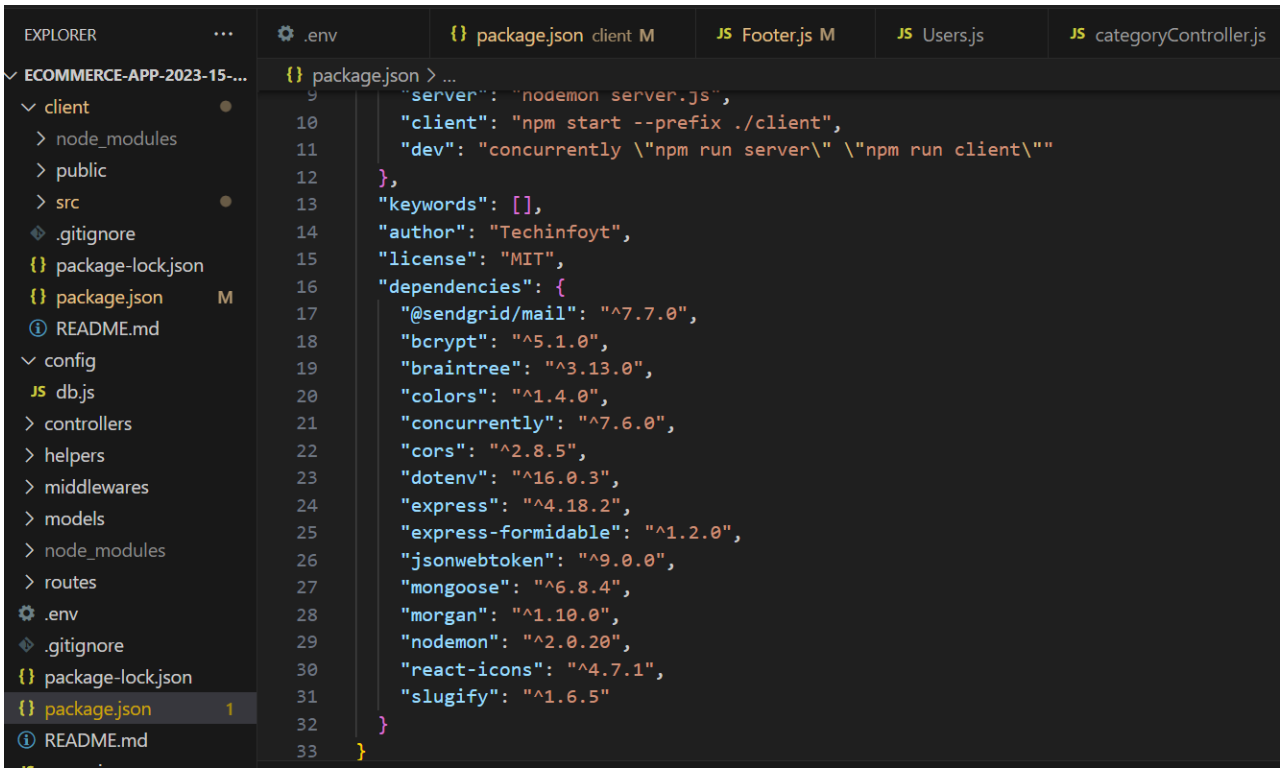


The ‘useCategory’ hook, implemented with React and Axios, manages the state for categories in a functional component. It initializes with an empty array and utilizes the ‘getCategories’ function, which makes an asynchronous request to retrieve category data from the server using Axios. The ‘useEffect’ hook ensures that the categories are fetched when the component mounts. The `categories` state is then updated with the received data, and the hook returns the current array of categories. This hook can be employed in other components to seamlessly manage and retrieve category data.



The CartPage component in React manages the shopping cart, utilizing context for state management and various dependencies. It displays cart items, allowing removal, and presents a summary with the total price. Users can update their address, and if logged in, proceed to checkout. The component integrates with Braintree for payment processing, displaying a loading state during payment. Upon successful payment, the cart is cleared, and users are redirected to their orders. The component ensures a seamless shopping experience, handling various scenarios like login prompts, address updates, and payment processing. Styling is enhanced with Bootstrap and custom CSS.

DEPENDENCIES USED



The project relies on various npm packages to enhance its functionality. These dependencies serve specific purposes within the application. Notable ones include '@sendgrid/mail' for handling email communication, 'bcrypt' for secure password hashing, and 'braintree' for integrating a payment gateway. Additionally, packages like 'concurrently' enable running multiple commands concurrently, while 'express' and 'mongoose' streamline server-side development and database interactions. The inclusion of middleware such as 'cors' and 'morgan' enhances security and facilitates logging. Overall, these npm packages collectively contribute to building a robust and feature-rich web application.