MERN Marketplace

Mesafnt Kalu, Samuel Wubeshet, Lina Akostka

A little bit of background

- Our website makes it easier for people to buy and sell goods online. The website provides a user-friendly interface that allows users to browse products, add products to their cart, and checkout. The application also provides a secure payment system that protects users' financial information.
- By creating a simple and user-friendly interface, we hope to encourage people with small businesses to sell online. This will extend their reach and allow them to tap into a wider market. We believe that this is a positive step for small businesses and the economy as a whole.
- We are also motivated by the desire to help those who are in need. As such, we have included the option for users to donate to charity when they make a purchase. We believe that this is a small way that we can make a difference in the world.

Making our plans come to life

- Users are able to access tabs such as: All Shops, My Shops, and My Profile.
- Customers are able to search by category of the product, or by using a search bar to narrow down their options. They are also able to see the latest product arrivals.
- Users are able to create their own shops and view them under “My Shops” tab.
- Users are also able to bookmark their favorite stores and products.

Challenges

- The most challenging aspect of our previous workflow was the manual process of posting products to the database and connecting with our Stride account to test transactions. This process was time-consuming and prone to errors. We have since automated this process, which has saved us time and improved the accuracy of our transactions.
- We also faced challenges in setting up and configuring the tabs on our website. We wanted to ensure that they were user-friendly and easy to navigate, while also being visually appealing. We are pleased with the final product, and we believe that it will provide a better experience for our users.

Highlight of Success

- A unified user interface that is consistent across all pages and sections of the website. This makes it easier for users to find what they are looking for and to navigate the website.
- A shopping cart that allows users to add items to their purchase and then check out at a later time. This makes it easier for users to shop online and to avoid having to make multiple trips to the website.
- A seller dashboard that allows sellers to view their sales, manage their inventory, and communicate with customers. This makes it easier for sellers to run their businesses online.
- A bookmarking feature that allows users to save their favorite stores and products. This makes it easier for users to find the products they are interested in and to come back to them later.

Features

- Use JSON Web Tokens (JWT) to authenticate users.
- Allow users to add, edit, view, and delete items in the store.
- Allow users to add and remove items from the cart.
- Display the total cost of the cart and update it as soon as the cart is updated.
- Store the JWT in Local Storage so that only logged-in users can buy items.
- Allow users to pay and checkout, which will create an order and empty the cart.
- Have a donate to charity option upon the check out

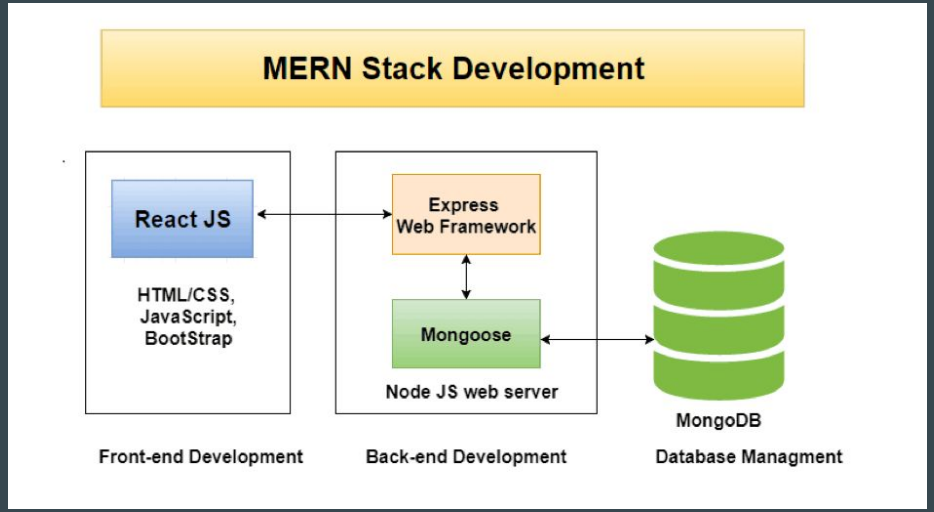
Frontend

- For the frontend, we have used React, a JavaScript library for building user interfaces. React is a component-based library, which means that it makes it easy to create reusable and maintainable code.
- We have also used Redux, a state management library that helps us to keep track of the state of our application in a centralized location. This makes it easier to reason about the state of our application and to make changes to it.
- Finally, we have used React Bootstrap, a library that provides React components for Bootstrap, a popular CSS framework. This makes it easier to style our application and to make it look like Bootstrap.

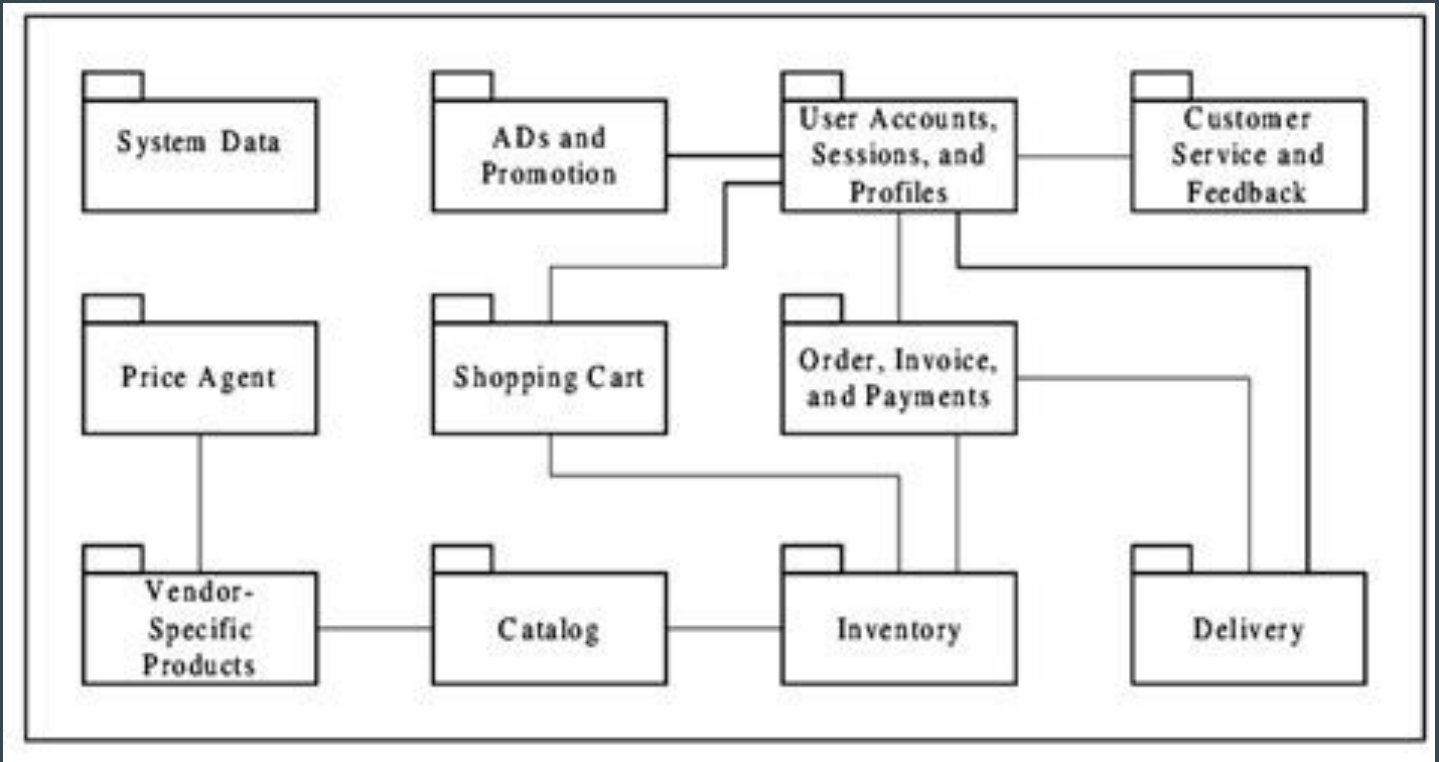
Backend

- For the backend, we have used Express, a web application framework for Node.js. Express is a fast, unopinionated, and minimalist framework that makes it easy to build web applications.
- We used Express to create REST APIs, and then incorporate these endpoints into our React frontend in order to interact with the backend.

- We also used MongoDB, a NoSQL database that stores data in documents in JSON format. MongoDB is a scalable and flexible database that is well-suited for storing large amounts of data.
- Finally, we used Mongoose, a MongoDB object modeling layer for Node.js. Mongoose makes it easy to interact with MongoDB databases and to model our data in a way that is easy to understand and maintain.



Logical Design



Future Plans

- A personalized user experience - We will use data to track customer behavior and preferences, and then using that information to create personalized shopping experiences.
- Making website Mobile friendly - We will make sure that our websites is optimized for mobile devices so that customers can have a seamless shopping experience.
- Adding more features - Adding more features and categories to add more diverse groups of users, and expand our scope.

Repo Links/Demo

https://github.com/mesafnt/Final_Project_CSC3916.git

https://final-paa6.onrender.com