NAAN MUDHALVAN

MERN STACK FUNDAMENTALS

TEAM MEMBERS:

Abinaya S – 2021506002 (5C7E00FD635182A870B435BD8F9FC37D)

Ellammal M A– 2021506022 (94FFEDE0F25783F9EFC4F5FA97801050)

Muthuvarshini S-2021506054
(FBF56B7DFDBA3DAA5AB4588099A6ED8A)

Sowmiya S – 2021506104 (67FA9860B02F928A26C4A99A7EFF1961)

Vikkiraman V- 2021506320 (DF8041D35798157BA5247F20A1DD0FD4)

INTRODUCTION

This report details the development of a grocery web application built using HTML, CSS, JavaScript, and MongoDB. The application aims to provide a seamless online shopping experience for users, enabling them to browse products, add items to their cart, view order details, and securely log in to their accounts, while ensuring efficient data management and user-friendly interfaces. It also incorporates responsive design to ensure compatibility across various devices for enhanced accessibility.

TECHNICAL IMPLEMENTATION

FRONTEND:

HTML:

- Structured the web pages, including the homepage, product listings, cart page, and login page.
- Implemented HTML forms for user input and interaction.

CSS:

- Styled the web pages to create a visually appealing and user-friendly interface.
- Used CSS to layout elements, apply colors, fonts, and spacing.

JavaScript:

- Handled dynamic interactions on the website, such as:
- Form validation
- AJAX requests to fetch and update data
- DOM manipulation to update the UI
- Event listeners for button clicks and other user actions
- Implemented client-side search functionality using JavaScript to filter products based on user input.

BACKEND:

MongoDB:

- Used MongoDB to store product information, user accounts, and order details.
- Created appropriate data structures to efficiently manage and retrieve data.
- Implemented CRUD operations (Create, Read, Update, Delete) to interact with the Database.

Server-Side Scripting (Optional):

- If required, used a server-side language like Node.js or Python to handle API requests, process user data, and interact with the MongoDB database.
- Implemented authentication and authorization mechanisms to protect user accounts and sensitive information.

FEATURES:

Homepage:

- Displays a curated selection of products.
- Includes a search bar to allow users to find specific products.

Product Listing:

- Presents products with detailed descriptions and images.
- Allows users to add products to their cart.

Cart:

- Shows a list of items added to the cart.
- Enables users to modify quantities and remove items.
- Calculates the total cost of the cart.

Checkout:

- Guides users through the checkout process.
- Collects shipping and payment information.
- Processes orders and updates inventory.

User Login:

- Provides a secure login system for registered users.
- Stores user credentials in the MongoDB database.
- Implements password hashing and salting for security.

Order History:

- Allows users to view their past orders and details.

FUTURE ENHANCEMENTS:

Mobile Optimization:

- Adapt the website for optimal viewing on mobile devices.

Payment Integration:

- Integrate with payment gateways to process online payments.

User Reviews and Ratings:

- Allow users to leave reviews and ratings for products.

Personalized Recommendations:

- Implement a recommendation system based on user purchase history and preferences.

Push Notifications:

- Send notifications for order updates, promotions, and new arrivals.

CONCLUSION

This grocery web application successfully combines HTML, CSS, JavaScript, and MongoDB to provide a robust and user-friendly online shopping experience. The application offers essential features like product browsing, cart management, secure login, and order processing. By incorporating future enhancements, this application can further elevate the user experience and expand its capabilities.