Green Mart

Project Presentation

Eby Kurian
Ebin J Alapatt
Gautham Suresh

Federal Institute of Science And Technology

Contents

Description

Our project is a web application that allows selling and buying organic products grown and manufactured at a household, which includes the following features:

- The main page of the website will provide an overview of the website, display featured products, and allow users to search for products based on various criteria such as product name, category, price, and location. The search results will display the relevant products with images, descriptions, and pricing information.
- Users can contact the sellers directly through the website and ask questions about the products or place orders. The website will provide a messaging system or a contact form to facilitate communication between buyers and sellers. Sellers can create their own profiles and list their products for sale, using a simple and intuitive interface to upload product images, descriptions, and pricing information. They can also manage their inventory and track their sales through their profile.
- Users can create their own accounts to save their personal information, shipping addresses, and order history. This feature will also allow users to leave feedback and ratings for sellers. In addition, the website will include a forum or a knowledge base where users can find answers to frequently asked questions about the products, the sellers, or the website. Users can also post their own queries and receive responses from the community or the website moderators.
- The website will provide detailed information about the sellers, including their location, contact information, ratings, and reviews. This feature will help users make informed decisions about buying from a particular seller. Overall, this web application will provide a convenient and reliable platform for selling and buying organic products grown and manufactured at a household, while also fostering a sense of community and trust among the users.

System Requirements

- **Server:** The web application requires a server to host the website and its database. The server should have a sufficient amount of storage space, processing power, and memory to handle user traffic and data storage.
- **Hosting:** The web application should be hosted on a reliable hosting service that provides sufficient bandwidth and storage space for the website and its database
- **Browser compatibility:** The web application should be compatible with different web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

User Requirements

- The web application should have a user-friendly interface that is easy to navigate.
- Users should be able to search for products based on various criteria such as product name, category, price, and location.
- Users should be able to view detailed information about the products, including images, descriptions, and pricing information.
- Users should be able to contact the sellers directly through the website and ask questions about the products or place orders.
- Users should be able to create their own accounts to save their personal information, shipping addresses, and order history.
- Users should be able to leave feedback and ratings for sellers.
- The web application should have a messaging system or a contact form to facilitate communication between buyers and sellers.
- Sellers should be able to create their own profiles and list their products for sale, using a simple and intuitive interface to upload product images, descriptions, and pricing information.
- The web application should have a forum or a knowledge base where users can find answers to frequently asked questions about the products, the sellers, or the website.
- The web application should provide detailed information about the sellers, including their location, contact information, ratings, and reviews.
- The web application should ensure the security of users' personal information and transactions.

Technologies Used

- **Front-End:**
- React.js: A popular front-end JavaScript library for building user interfaces and handling client-side logic.
- Redux.js: A predictable state container for managing application state across components.
- **Back-End:**
- Node.js: A server-side JavaScript runtime environment for building scalable and fast back-end applications.
- Express.js: A popular and lightweight web application framework for Node.js that simplifies building RESTful APIs.
- Database:mysql and mongodb

Implementation

- Set up the development environment: Install Node.js and MongoDB on your local machine. Set up a new project folder and initialize it with npm.
- Design the user interface: Create wireframes and design mockups using tools like Figma or Sketch. Use React to implement the UI components, and CSS for styling.
- Set up the server: Create an Express.js server and connect it to the MongoDB database using the Mongoose library.
- Create API endpoints: Define the routes and endpoints for the web application, including user authentication, product search, and seller registration.
- Integrate third-party APIs: Use APIs like Google Maps for location-based search or Stripe for payment processing.
- Test the application: Perform unit testing and integration testing to ensure the application functions as expected.
- Deploy the application: Deploy the web application to a hosting service like Heroku or AWS, and configure domain settings.