**Full stack Engineering**

Project Report

Semester-VI (Batch-2022)

“TOMATO-A Food Delivery Website”

**A red and white sign

Description automatically generated with low confidence**

**Supervised By: Submitted By:**

Mr. Rahul Saloni,2210992231(G-22)

Saloni Kumari,2210992232(G-22)

Samreen Kaur,2210992240(G-22)

Samridhi Sachdeva,2210992243(G-22)

**Department of Computer Science and Engineering**

**Chitkara University Institute of Engineering & Technology,**

**Chitkara University, Punjab**



**Index**

**Sr. no Topic Page No**

1 Title of project 03

1. Objective & Key Learning’s 03

3 Problem Statement 03

4 Software Requirements 04

5 Proposed Design / Methodology 05

6 Core Features & Functionality 06

7 Code Snippets 07

8 Project Snippets 09

9 References 12



**1.Title of Project :**

"Tomato: Revolutionizing Convenient and Efficient Dining Experiences"

**2. Objective & Key Learning’s :**

**Objective:**

The primary objective of this project is to design and develop a food delivery application that provides users with a seamless experience for ordering meals. The system aims to:

* Support a diverse range of restaurant options.
* Offer real-time order tracking for enhanced transparency.
* Store and manage order history for easy reference.
* Display personalized recommendations based on user preferences.

**Key Learning’s:**

* Database Management: Storing user profiles, chat history, and real-time updates.
* Scalability: Designing a scalable system architecture for handling concurrent users.
* User Interface (UI) Design: Building an intuitive and user-friendly interface.
* Security: Implementing secure messaging features like end-to-end encryption, authentication, and data protection.



**3.Problem Statement:**

In today's fast-paced world, quick and reliable access to food is a critical necessity for individuals with busy lifestyles. Existing food delivery platforms often fall short in providing an intuitive, seamless experience. Common challenges include inefficient navigation, lack of real-time order tracking, limited restaurant options, and absence of personalized features that cater to individual preferences.

* There is a growing demand for a modern food delivery application that overcomes these

limitations by offering a user-friendly interface, a wide selection of restaurants, personalized recommendations, and instant updates on order status. By addressing these needs, the application can enhance the overall dining experience, providing convenience and efficiency to users.

**4. Software Requirements:**

**To develop the food delivery website using MERN Stack, the following software tools and technologies are required:**

**a. Frontend Technologies:**

* React.js: A widely used JavaScript library for building user interfaces with reactive
* HTML/CSS: For basic structure and styling.

**b. Backend Technologies:**

* Express.js: A lightweight and fast Node.js framework for building robust APIs and handling server-side logic.
* APIs: Used for facilitating communication between the frontend and backend, enabling seamless data exchange.
* Axios: A popular library for making HTTP requests to APIs, ensuring efficient communication between the client and server.
* Middleware: Implemented to handle tasks like authentication, request parsing, and error handling efficiently.



* MongoDB: A NoSQL database for storing and managing user profiles, order history, and menu data.

**5. Proposed Design / Methodology**

**Project Planning & Research**

* **Define Target Audience: Understand who the website is for (e.g., urban professionals, students, families) and their needs (convenience, variety, speed, cost).**
* **Market Research: Analyze competitors (UberEats, DoorDash, Grubhub) to identify features, design trends, and functionality.**
* **Establish Business Requirements: Determine key features like restaurant listings, delivery tracking, payment systems, etc.**
* **Set Milestones & Timeline: Break the project into phases: Discovery, Design, Development, Testing, Launch.**

**UX/UI Design**

* **Wireframing & Prototyping**:
  + **Tools**: Figma, Adobe XD, or Sketch.
  + Design wireframes for key pages: homepage, restaurant page, menu, cart, checkout, user profile, etc.
* **Information Architecture**: Organize content for intuitive navigation. Structure the content for easy browsing of restaurants, menus, and categories (cuisine type, ratings, prices).
* **User Flow Design**:
  + **Homepage** → browse restaurants → select items → add to cart → checkout → payment → delivery tracking → confirmation.
* **Visual Design**:
  + **Branding**: Choose color schemes, typography, logo design, and imagery .



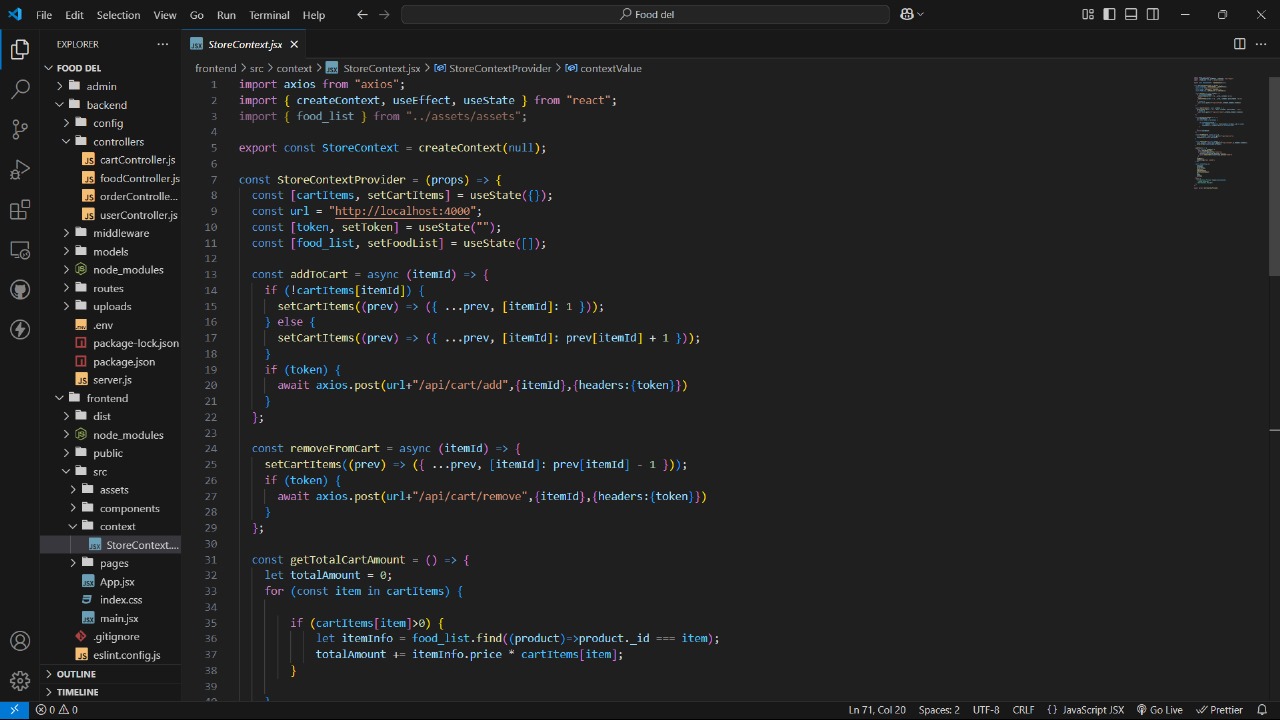
**6. Core Features & Functionality:**

* **Homepage & Search Bar**:
  + Search by restaurant, cuisine, location, or dish.
  + Filters for ratings, price range, and dietary preferences.
* **Restaurant Listings**:
  + Display a list of restaurants with thumbnails, ratings, cuisine type, and distance.
  + Option to sort/filter by popularity, delivery time, or price.
* **Restaurant Menu**:
  + Display menus with prices, images, and descriptions.
  + Allow customization options (e.g., add toppings, size, etc.).
* **Cart & Checkout**:
  + Users can view selected items, edit quantity, or remove items.
  + Option for promo codes, gift cards, and user rewards.
  + Integration with secure payment gateways (PayPal, Stripe, credit card options).
* **User Account**:
  + Sign up/login (using email or social accounts).
  + Order history, saved addresses, payment methods, and preferences.
  + Ability to track orders in real-time with delivery ETA.
* **Order Tracking**:
  + Real-time order status (placed, preparing, dispatched, delivered).
  + Map view of delivery progress.
* **Ratings & Reviews**:
  + Allow customers to rate and review restaurants and dishes.
  + Use reviews to enhance credibility and guide future customers.
* **Admin Panel**:
  + Dashboard to manage restaurant listings, menus, orders, payments, and customer feedback.

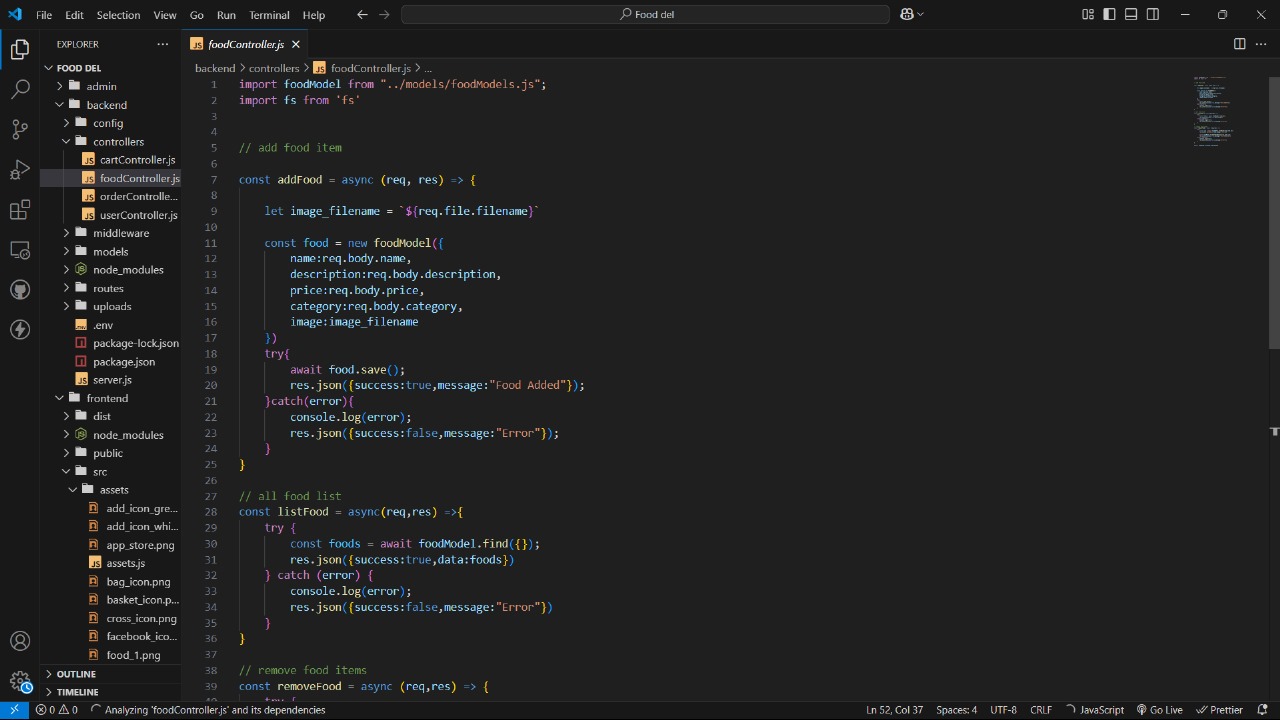


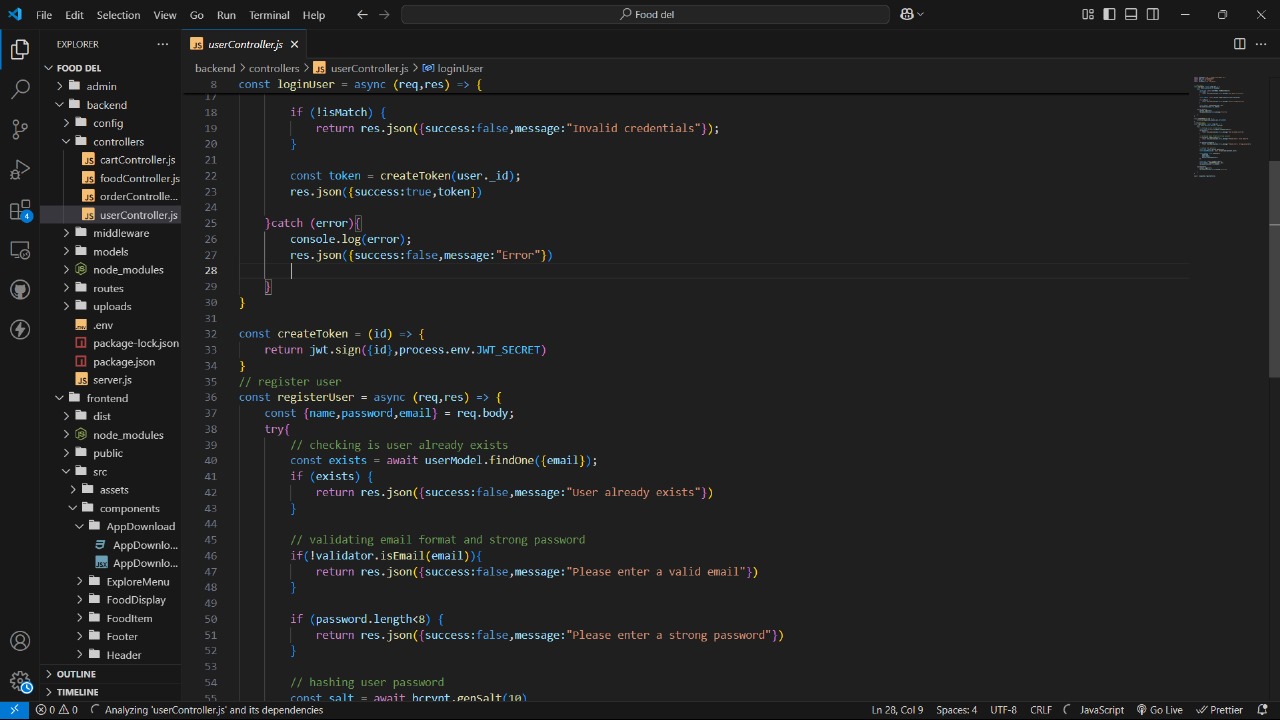
* **Notifications**:
  + Email or push notifications for order updates, delivery statuses, and promotional offers.

**7.Code Snippets:**

****



****

****



**8. Project Snippets:**

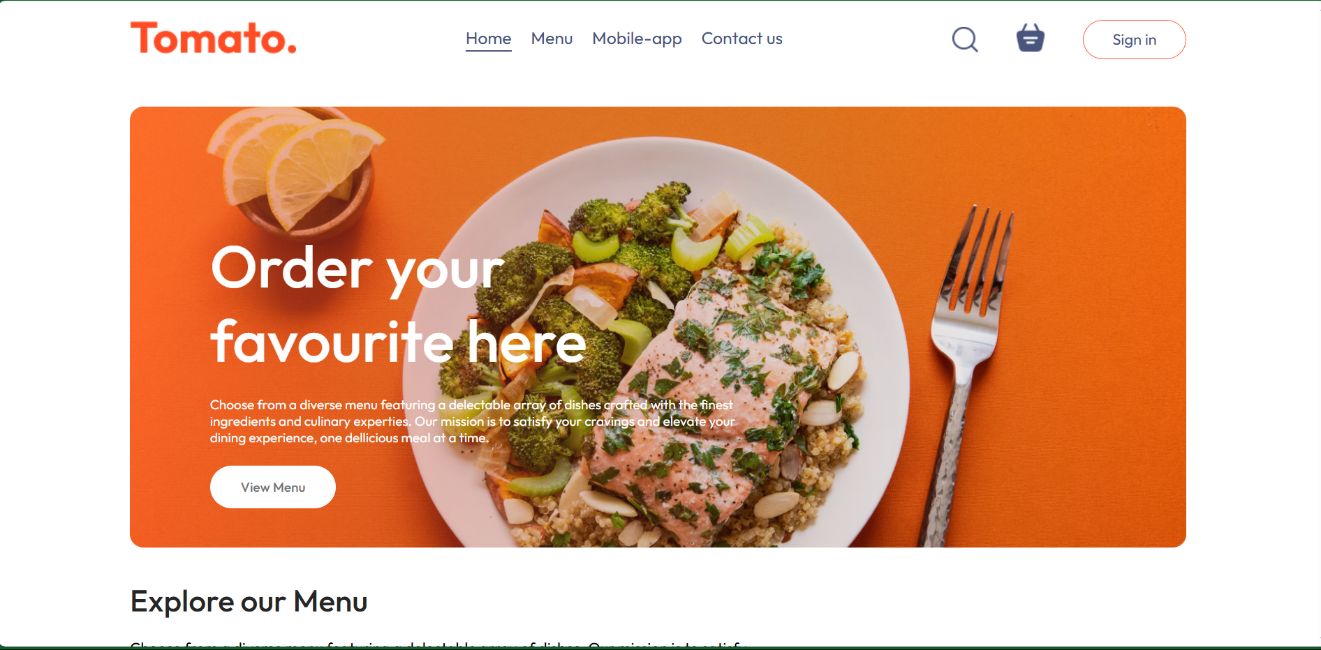
****

Figure Home page

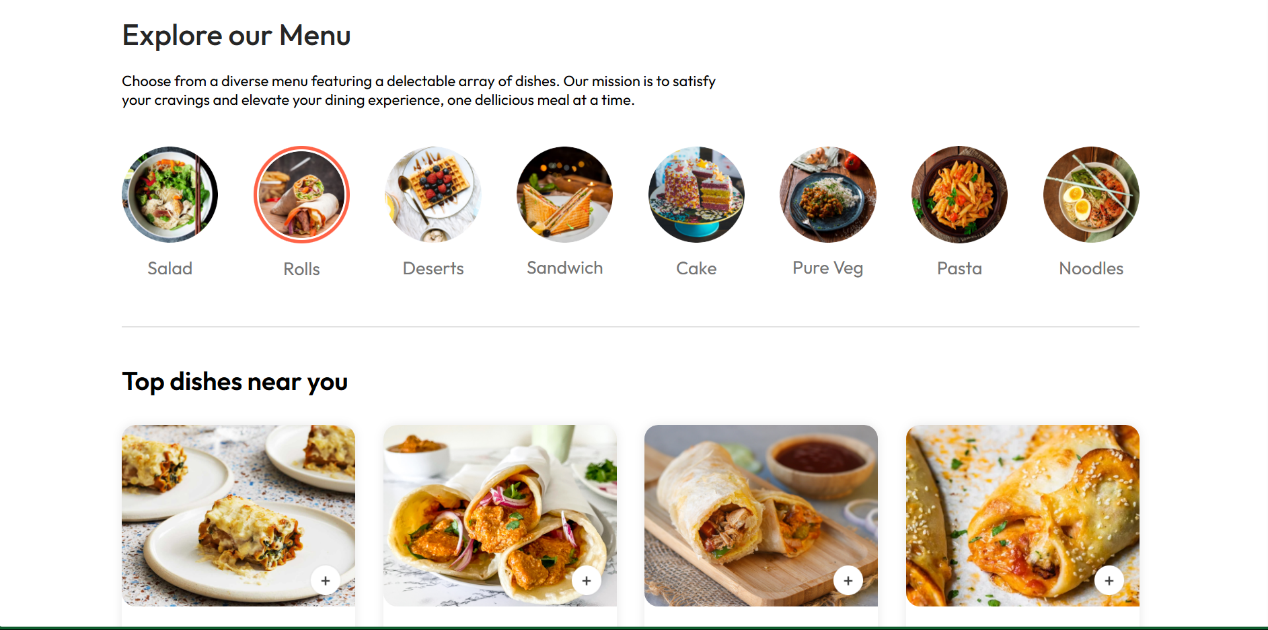
****

Figure Menu page



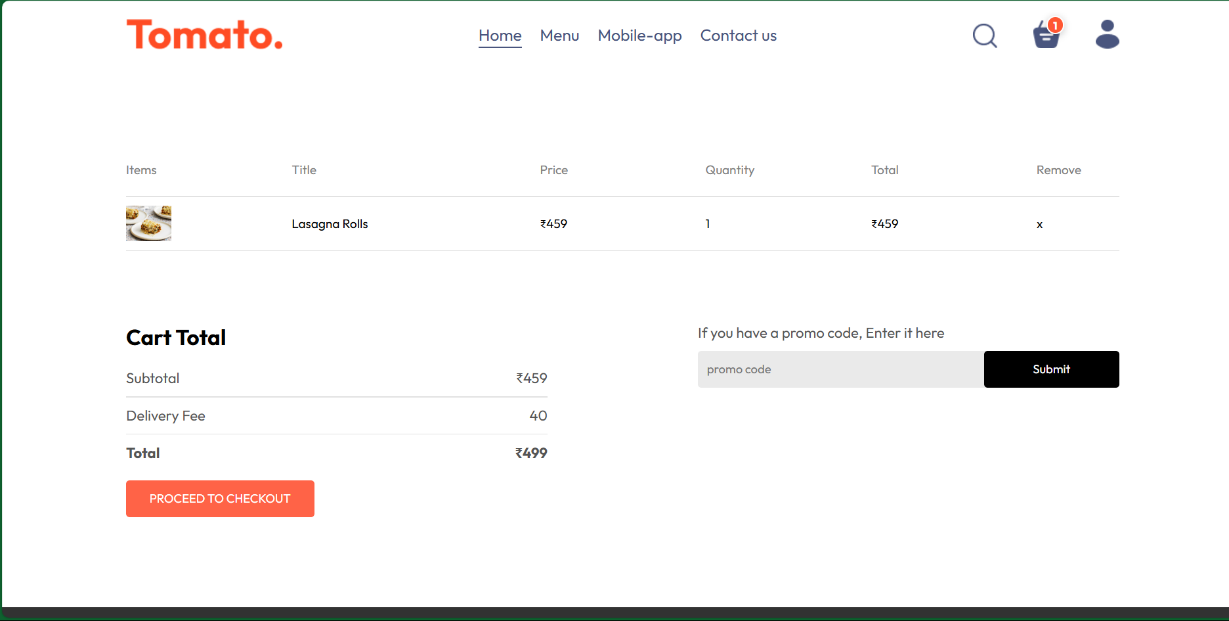
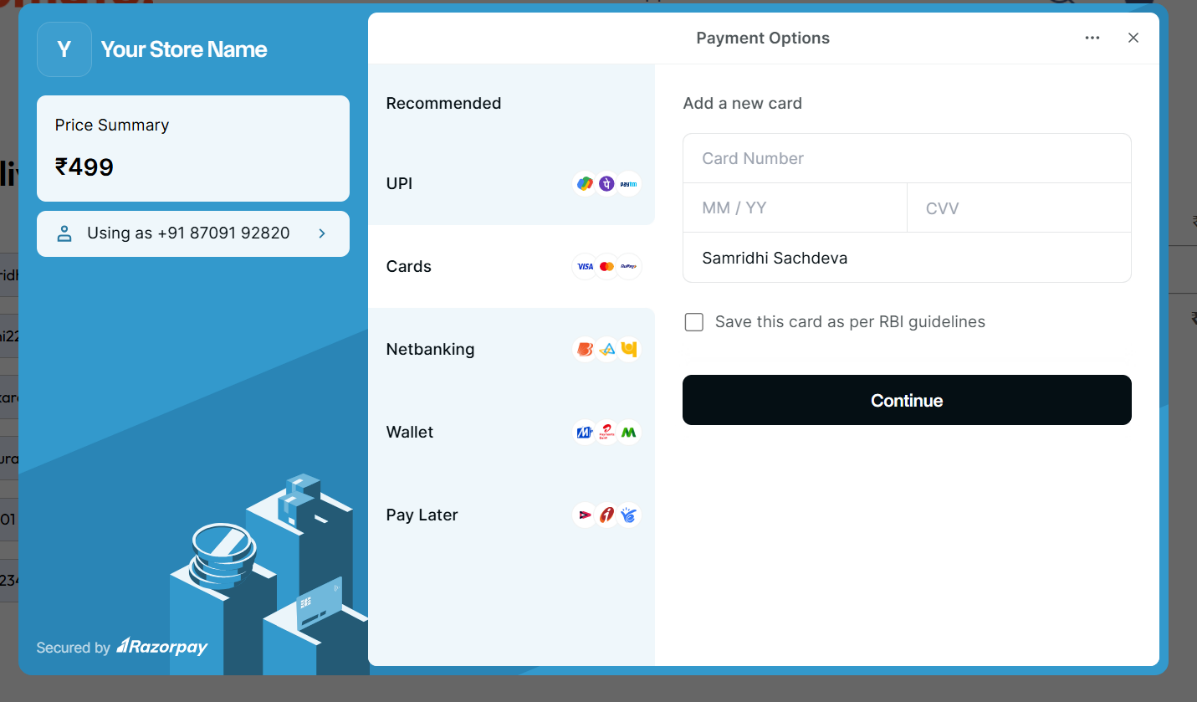
****

Figure Cart page

****

*Figure 4 Payment*



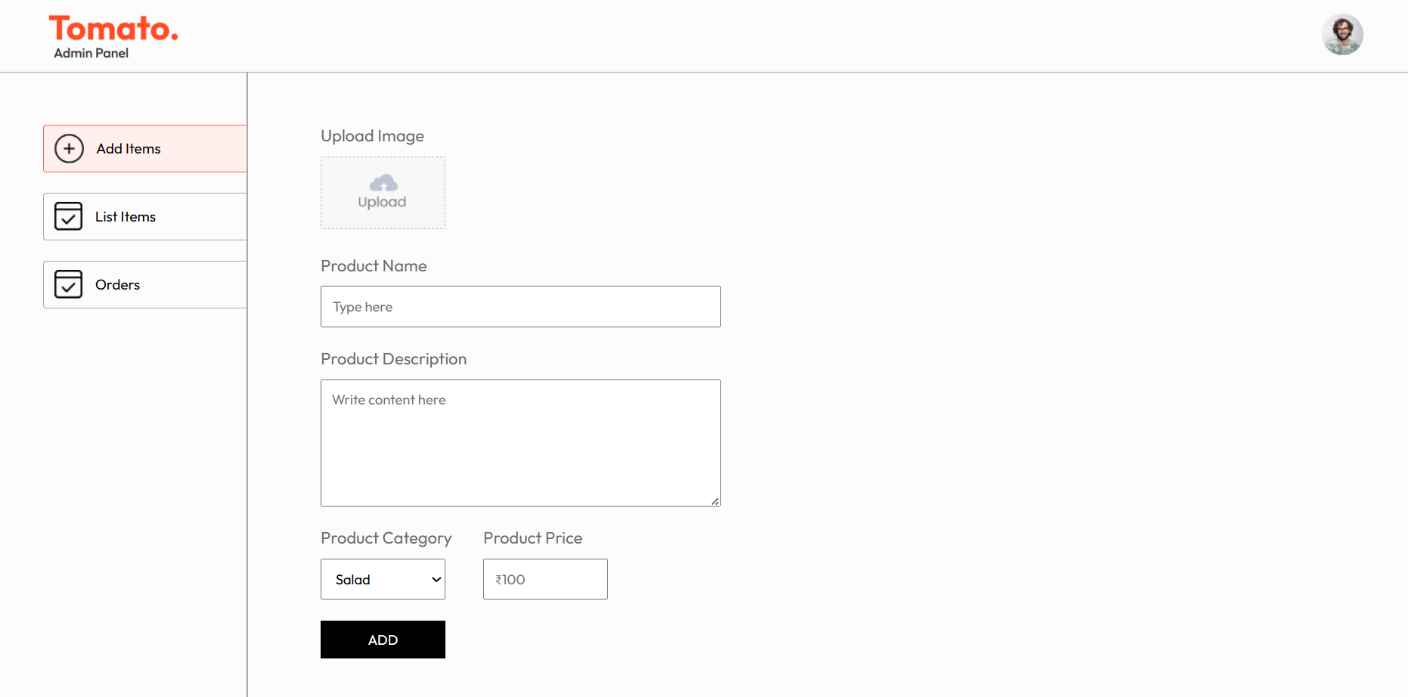
****

Figure Admin Pannel



**9. References**

* **Node.js Official Site:** Comprehensive documentation and guides for backend development using Node.js. ([nodejs.org](https://nodejs.org))
* **Express.js Documentation:** Official reference for using Express.js to build server-side logic and APIs. ([expressjs.com](https://expressjs.com))
* **MongoDB Documentation:** Resources for database integration and data management. ([mongodb.com/docs](https://www.mongodb.com/docs))
* **Axios GitHub Repository:** Details on Axios library for API requests. ([github.com/axios/axios](https://github.com/axios/axios))
* **React Documentation:** Guides and tutorials for building user interfaces. ([react.dev](https://react.dev))
* **W3Schools HTML/CSS Tutorials:** For foundational web development knowledge. ([w3schools.com](https://www.w3schools.com))