**FOOD DEL**

**A MINI PROJECT REPORT**

***Submitted by***

**RETHINAATH S (220701222)**

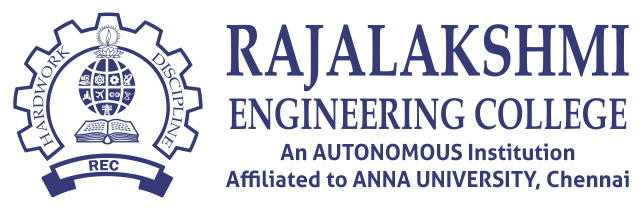
***in partial fulfillment of the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**in**

**COMPUTER SCIENCE AND ENGINEERING**



**RAJALAKSHMI ENGINEERING COLLEGE**

**RAJALAKSHMI NAGAR**

**THANDALAM**

**CHENNAI – 602 105**

**NOVEMBER 2024**

**RAJALAKSHMI ENGINEERING COLLEGE**

**CHENNAI - 602105**

**BONAFIDE CERTIFICATE**

Certified that this mini project report “**FOOD DEL**” is the bonafide work of “**RETHINAATH S (220701222)**” who carried out the project work for the subject CS19542-Internet Programming under my supervision.

**MR. DEEPAK KUMAR,**

Assistant Professor (SG),

Department of

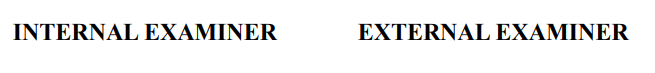
Computer Science and Engineering

Rajalakshmi Engineering College

Rajalakshmi Nagar

Thandalam

Chennai – 602105

Submitted to Project and Viva Voce Examination for the subject CS19542-Internet Programming held on \_\_\_\_\_\_\_\_\_\_.

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS** | | |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** | **iv** |
|  | **ACKNOWLEDGEMENT** | **v** |
| **1.** | **INTRODUCTION** | **1** |
|  | 1.1 INTRODUCTION | 1 |
|  | 1.2 SCOPE OF THE WORK | 2 |
|  | 1.3 EXISTING SYSTEM | 2 |
|  | 1.4 AIM AND OBJECTIVES OF THE PROJECT | 2 |
| **2.** | * 1. **SYSTEM SPECIFICATIONS** | **4** |
|  | 2.1 HARDWARE SPECIFICATIONS | 4 |
|  | 2.2 SOFTWARE SPECIFICATIONS | 4 |
| **3.** | **ARCHITECTURE DIAGRAM** | **5** |
| **4.** | **MODULE DESCRIPTION** | **5** |
|  | 4.1 MENU BROWSING MODULE | 5 |
|  | 4.2 CART MODULE | 5 |
|  | 4.3 USER INTERFACE MODULE | 5 |
| **5.** | **SYSTEM DESIGN** | **6** |
|  | 5.1 USE CASE DIAGRAM | 6 |
|  | 5.2 ER DIAGRAM | 6 |
|  | 5.3 DATA FLOW DIAGRAM | 7 |
|  | 5.4 ACTIVITY DIAGRAM | 7 |
| **6.** | **SCREENSHOTS** | **8** |
| **7.** | **CONCLUSION** | **10** |
|  | **REFERENCES** | **11** |

**ABSTRACT**

The **“Food Del”** focuses on developing a food delivery web application that allows users to browse various food items, add selected items to their cart, and place orders conveniently. Built using React, the application provides an intuitive and responsive user interface optimized for both desktop and mobile platforms. It integrates essential features for a food delivery service, including user authentication, dynamic menu browsing, order tracking, and secure checkout. This project showcases best practices in modern web development and offers a scalable solution that can be extended with additional features in the future.

**ACKNOWLEDGEMENT**

I express my sincere thanks to my beloved and honourable chairman **R.S.MEGANATHAN** and the chairperson **DR.M.THANGAM MEGANATHAN** for their timely support and encouragement. I am greatly indebted to my respected and honourable principal **Dr. S.N.MURUGESAN** for his able support and guidance. No words of gratitude will suffice for the unquestioning support extended to us by my head of the department **Dr. P. KUMAR**, and my Academic Head **Dr.R.SABITHA** for being ever supporting force during my project work. I also extend my sincere and hearty thanks to my internal guide **Mr. DEEPAK KUMAR** for his valuable guidance and motivation during the completion of this project. My sincere thanks to my family members, friends and other staff members of Computer Science and Engineering.

**Rethinaath S (220701222)**

**CHAPTER 1**

**INTRODUCTION**

* 1. **INTRODUCTION**

The food delivery industry has witnessed exponential growth in recent years, driven by the convenience of ordering meals online. With the increasing demand for quick and reliable food delivery, having a dedicated and user-friendly platform is essential. This project, a food delivery web application, aims to simplify the process of browsing menus, placing orders, and tracking deliveries. Users can access the application from any device, making it easy to order food from their favorite restaurants in just a few clicks.

This application is designed to provide an enhanced user experience with seamless navigation, responsive design, and real-time updates. By using React, a powerful front-end framework, the application achieves smooth, dynamic interactions, ensuring a hassle-free experience for the user. Essential features like menu browsing, cart management, and order tracking are all integrated into a streamlined interface. These features allow users to complete their transactions swiftly, catering to the fast-paced lifestyle of today’s consumers.

Beyond the user experience, this application is built with scalability and maintainability in mind. With modular components and a flexible architecture, it can adapt to future updates and additional features, making it a sustainable solution for long-term use. By separating user interface components from business logic and data handling, the application achieves a clean and modular structure, which simplifies both initial development and future enhancements.

In addition, security and privacy are prioritized in the development of this platform. User data, including personal information and order details, is securely handled to maintain confidentiality. Secure authentication protocols are implemented, ensuring that each user has a protected account for safe transactions. Overall, this food delivery application not only meets the immediate needs of customers but also adheres to industry standards for web development, providing a secure, scalable, and efficient solution.

* 1. **SCOPE OF THE WORK**
* User authentication and session management.
* An interactive and filterable menu browsing experience.
* Cart functionality to add, modify, or remove items.
* Order processing with checkout, payment, and order tracking features.
* Notifications and order status updates. This project will be responsive across devices, allowing users to place orders from desktops, tablets, or mobile devices.
  1. **EXISTING SYSTEM**

Many existing food delivery systems face issues related to slow response times, complex navigation, and poor customization options. Additionally, some platforms are not optimized for mobile devices, making the ordering process cumbersome. This project aims to address these challenges by creating a streamlined, intuitive, and highly responsive platform specifically designed to simplify food ordering.

* 1. **AIM AND OBJECTIVES OF THE PROJECT**

**Aim:**

The aim of this project is to develop a robust, user-friendly food delivery application.

**Objectives:**

* To create a smooth, visually appealing UI that enhances user experience.
* To provide secure user authentication, including password encryption and session handling.
* To enable users to browse and search for food items with ease.
* To offer cart functionality, allowing users to review and modify their orders before checkout.
* To ensure that the application is optimized for all devices, especially mobile.

**CHAPTER 2**

**SYSTEM SPECIFICATIONS**

* 1. **HARDWARE SPECIFICATIONS**

The system requires basic hardware to host the website, which can be adjusted based on user load. Recommended hardware specifications include:

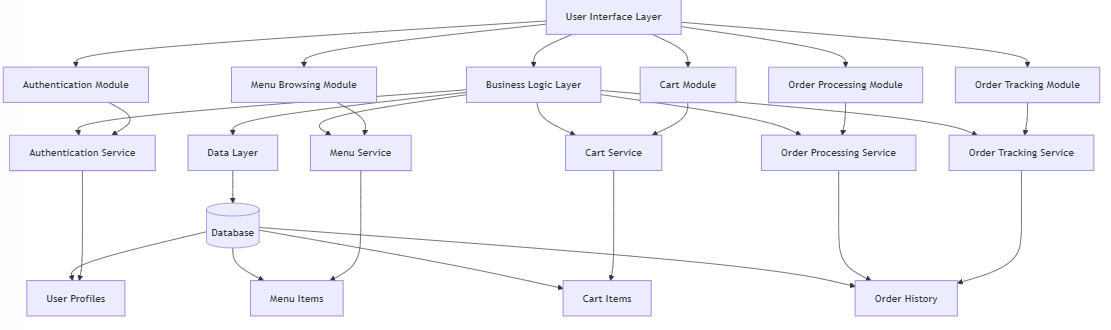
* Processor: Intel Core i5 or higher
* RAM: Minimum 8 GB for development, but 4 GB is sufficient for user devices
* Storage: 500 MB for development and deployment; server storage may vary based on user data
* Network: Stable internet connection, especially for API calls and backend interactions

For local development, any modern laptop or desktop with the above specifications will suffice. However, for production, cloud hosting is recommended to ensure scalability and accessibility.

* 1. **SOFTWARE SPECIFICATIONS**
* Operating System: Windows, macOS, or Linux
* Development Environment: Node.js (version 14 or above), React, and a package manager like NPM or Yarn
* Database: MongoDB, Firebase, or SQL-based database for storing user profiles, menu items, and orders

**CHAPTER 3**

**ARCHITECTURE DIAGRAM**



**CHAPTER 4**

**MODULE DESCRIPTION**

* 1. **Menu Browsing Module:**

Displays food items categorized by type (e.g., appetizers, main courses, desserts) with filtering and sorting options. This module enables users to view details, prices, and add items to the cart.

* 1. **Cart Module:**

Allows users to add, remove, or adjust quantities of food items before proceeding to checkout. The cart module provides real-time updates on total costs.

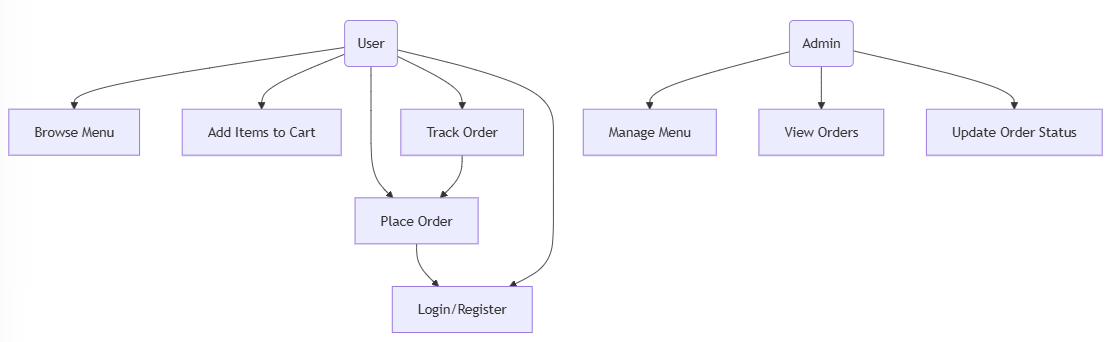
* 1. **User Interface Module:**

Ensures a cohesive and responsive design across the platform. This module is responsible for maintaining a consistent look, handling navigation, and managing user interactions.

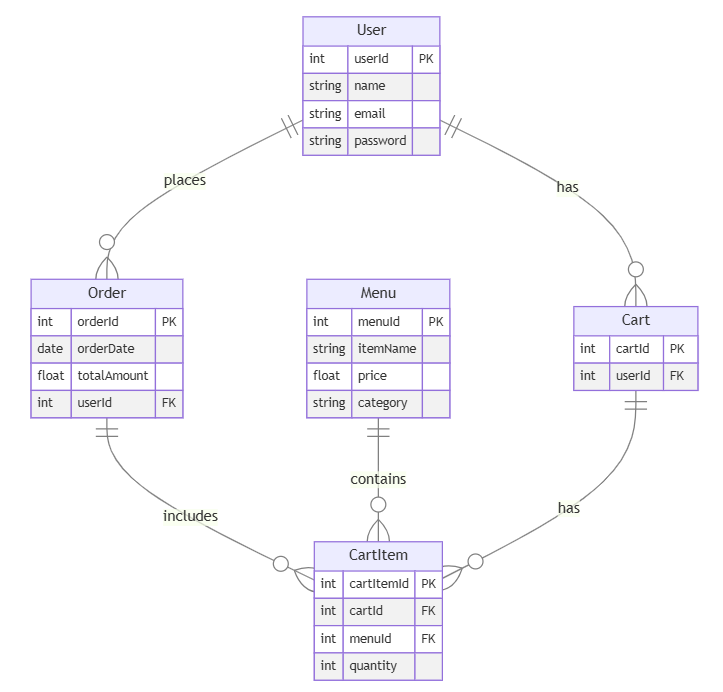
**CHAPTER 5**

**SYSTEM DESIGN**

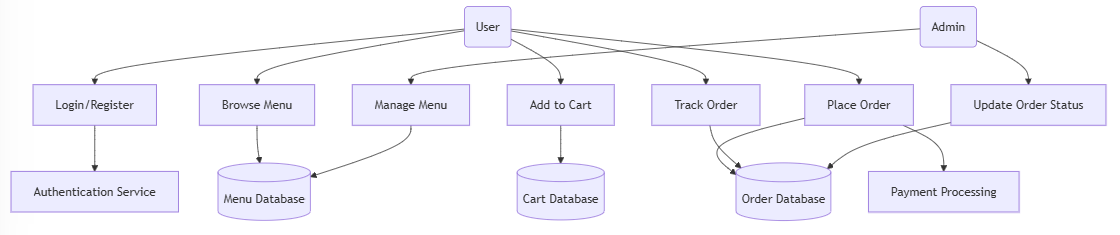
* 1. **USE CASE DIAGRAM**



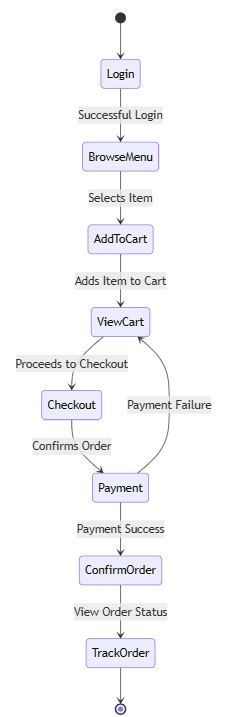
* 1. **E-R DIAGRAM**



* 1. **DATA FLOW DIAGRAM**

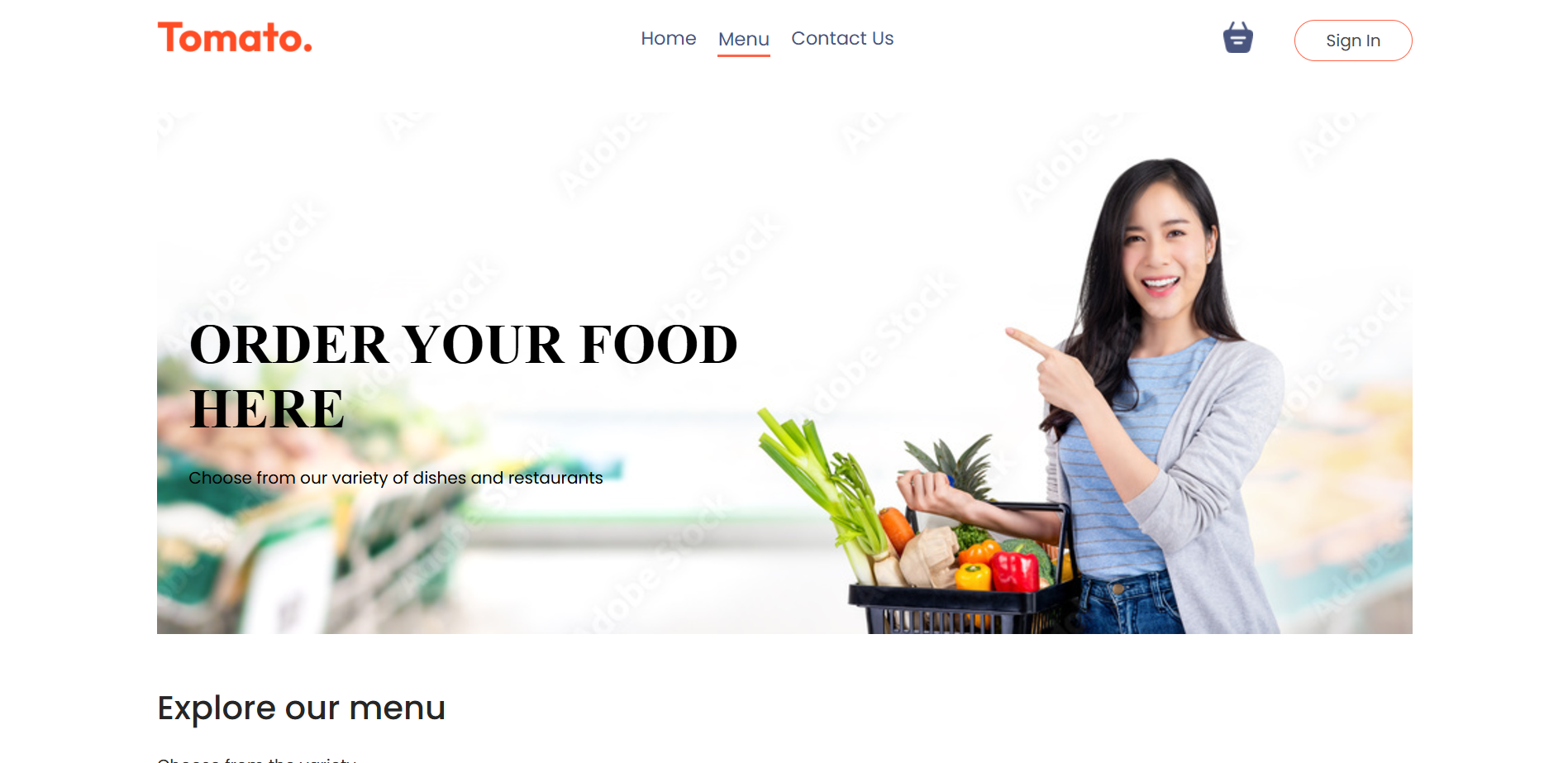


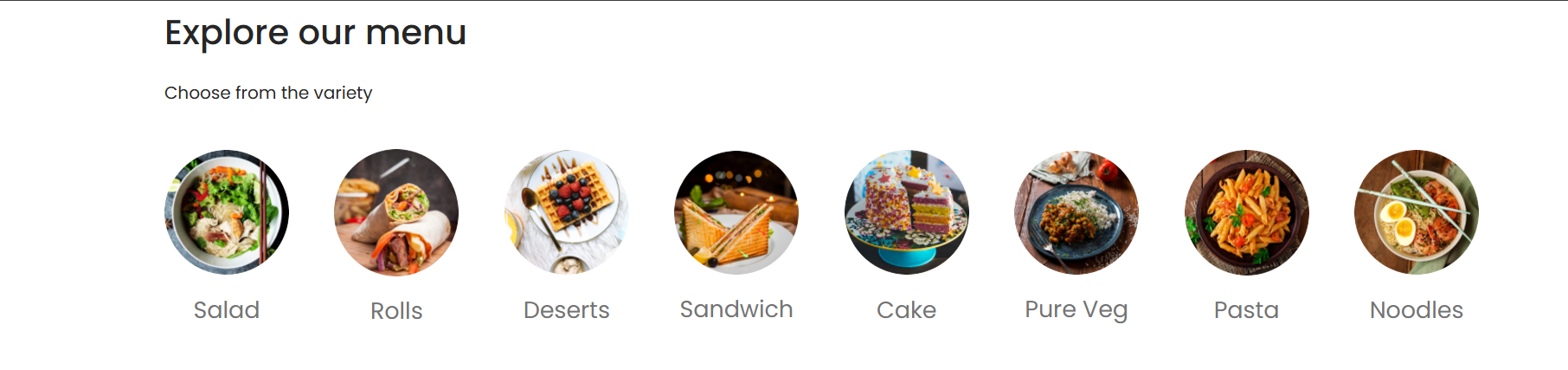
* 1. **ACTIVITY DIAGRAM**

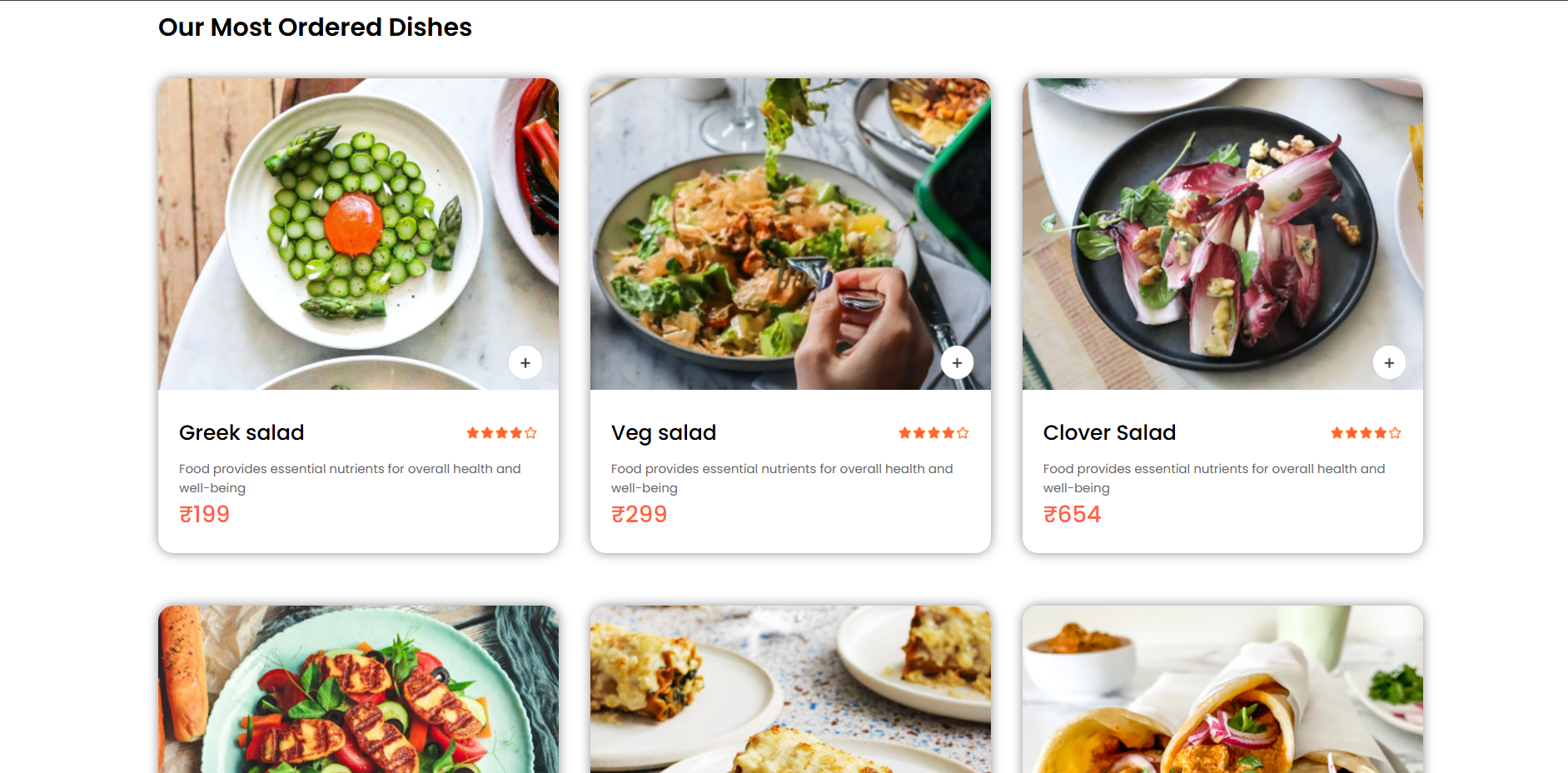


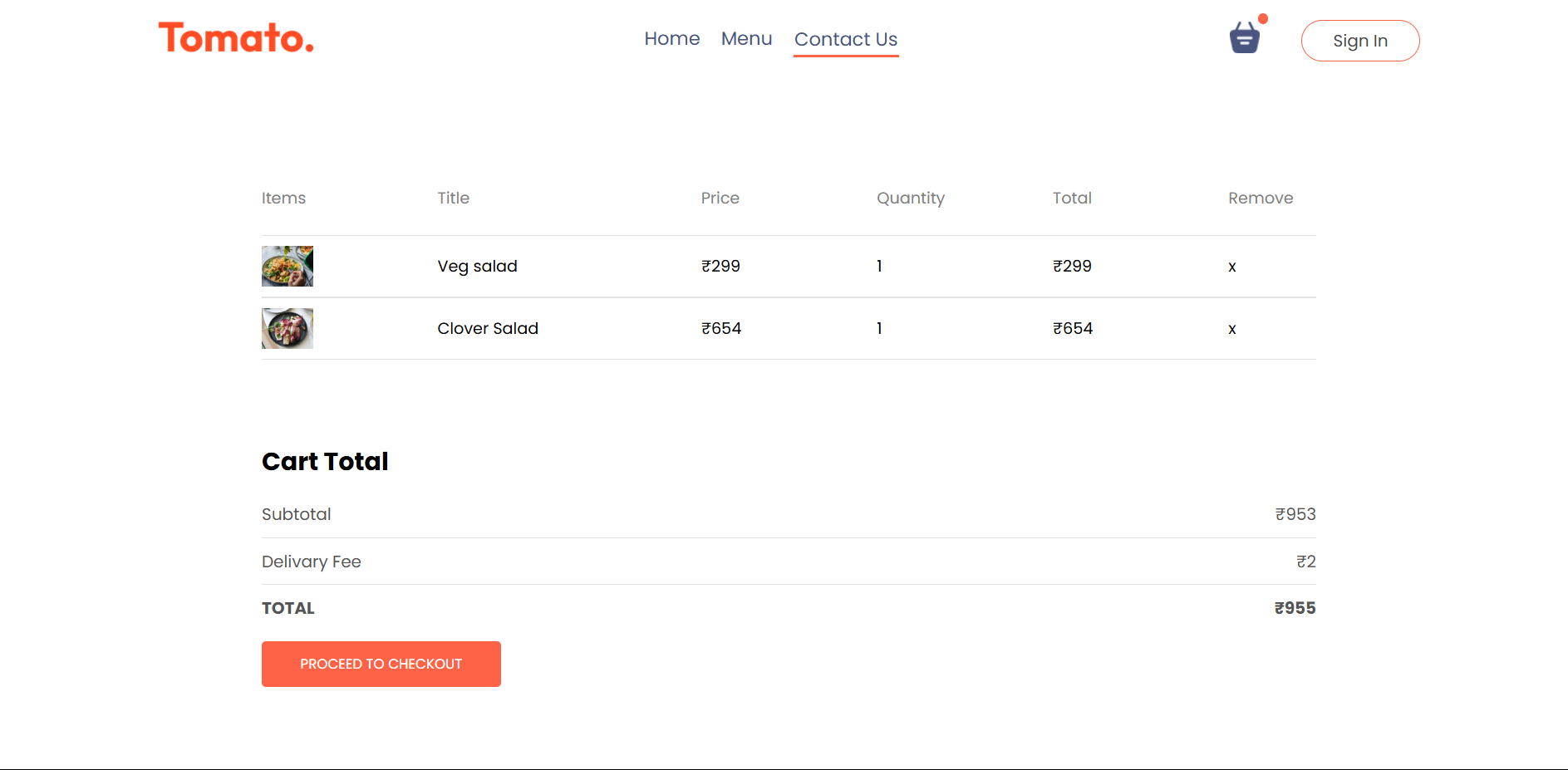
**CHAPTER 6**

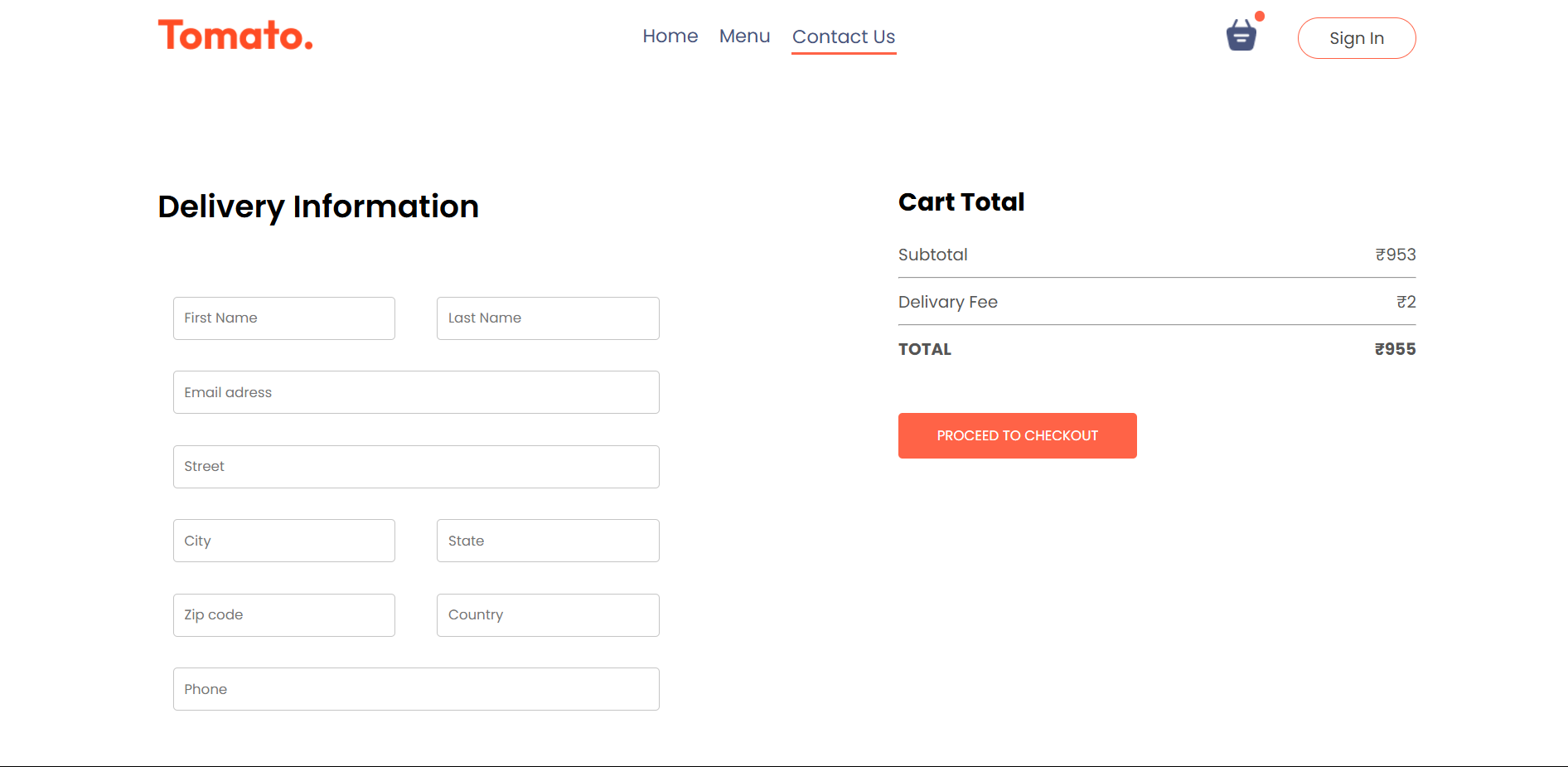
**SCREENSHOTS**











**CHAPTER 7**

**CONCLUSION**

The **“Food Del”** application successfully addresses the core needs of modern users by providing a seamless and intuitive platform for ordering meals. Utilizing React for a responsive front-end interface, the application ensures that users can browse menus, manage their cart, and place orders with ease. The project adheres to best practices in web development, focusing on a modular architecture that enhances scalability and maintainability. The implemented features, including secure user authentication, real-time order tracking, and a streamlined checkout process, demonstrate a comprehensive approach to building a reliable and user-friendly food delivery service.

Through this project, we have also emphasized security, user privacy, and data integrity by incorporating robust backend processes and database management practices. This application is well-positioned for future expansion, with the potential to integrate additional features like personalized recommendations, real-time delivery tracking, and in-app promotions. Overall, this project demonstrates a practical application of internet programming principles, delivering a complete food delivery solution that meets both current market demands and technical standards.

**REFERENCES**

* Node.js Documentation

[https://nodejs.org/en/docs/](https://nodejs.org/en/docs/" \t "_new)

* MongoDB Documentation

[https://www.mongodb.com/docs/](https://www.mongodb.com/docs/" \t "_new)

* React.js Documentation

[https://reactjs.org/docs/getting-started.html](https://reactjs.org/docs/getting-started.html" \t "_new)

* Stripe API Documentation

[https://stripe.com/docs](https://stripe.com/docs" \t "_new)

* Express.js Documentation

[https://expressjs.com/en/starter/installing.html](https://expressjs.com/en/starter/installing.html" \t "_new)

* GitHub Repository

[https://github.com/your-username/food-delivery-app](https://github.com/your-username/food-delivery-app" \t "_new)

* MongoDB Atlas

[https://www.mongodb.com/cloud/atlas](https://www.mongodb.com/cloud/atlas" \t "_new)