A PROJECT ON

Food Delivery Management System

SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR THE COURSE OF DIPLOMA IN ADVANCED COMPUTING FROM CDAC

SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY

Hinjawadi

SUBMITTED BY:

Pushkar Yogesh Kasar

UNDER THE GUIDENCE OF:

Mrs. Pooja Jaiswal

Faculty Member

Sunbeam Institute of Information Technology, Pune

ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mr. Yogesh Kolhe (Course Coordinator, SIIT, Pune).

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least, we thank the entire faculty and the staff members of Sunbeam Institute of Information Technology, Pune for their support.

Pushkar Yogesh Kasar

0225 PG-DAC

SIIT Pune

CERTIFICATE

This is to certify that the project work under the title ‘Food Delivery Management System’ is done by Pushkar Yogesh Kasar in partial fulfillment of the requirement for award of Diploma in Advanced Computing Course.

Mrs. Pooja Jaiswal

Project Guide

Mr. Yogesh Kolhe

Course-Coordinator

Date: 11/08/2025

INTRODUCTION

In today's fast-paced world, the demand for quick, efficient, and user-friendly services has transformed the way businesses operate, especially in the food industry. Traditional methods of ordering food often involve long queues, slow service, and limited communication between customers and service providers. The Online Food Ordering System addresses these challenges by offering a modern, technology-driven solution that streamlines the ordering process and enhances the overall customer experience.

This system allows customers to browse menus, place orders, and track their purchases with just a few clicks, eliminating the need for physical waiting lines. Through a secure and personalized account system, each customer's information—such as name, contact details, and order history—is stored safely, ensuring privacy and easy accessibility. The use of unique IDs and passwords provides an additional layer of security, protecting sensitive user data.

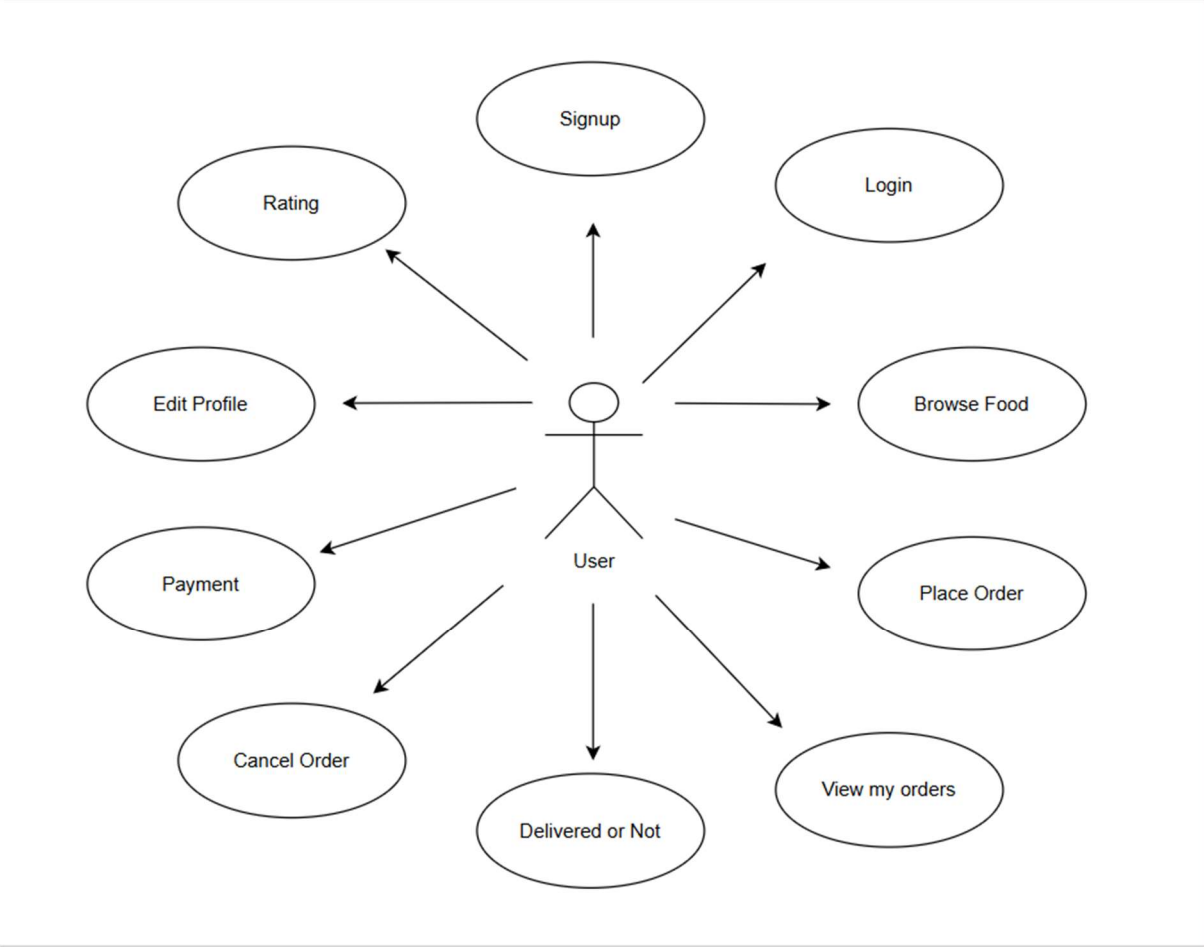
Flexibility in payment methods is another core advantage of the system. Customers can opt for online payment through integrated gateways for speed and convenience or choose the cash-on-delivery option for a more traditional approach. Real-time order tracking keeps customers informed about the status of their orders, enhancing transparency and trust.

Ultimately, the Online Food Ordering System not only improves service efficiency for businesses but also delivers unparalleled convenience for customers. By merging technology with food service, it ensures faster order processing, better customer satisfaction, and a more organized way of managing operations in the competitive food industry.

REQUIREMENTS

2.1 FUNCTIONAL REQUIREMENTS

- **User :**



The Online Food Ordering System is designed to provide customers with a seamless, efficient, and user-friendly platform for ordering food from their preferred restaurants. This system eliminates the limitations of traditional ordering methods by integrating features such as online browsing, secure payment, and real-time order tracking. The use case diagram illustrates the various interactions a user can have with the system, enabling a smooth flow from account creation to order delivery.

This section explains the different functionalities represented in the use case diagram, highlighting the role of the User and their possible actions within the system. The User is the primary actor in the Online Food Ordering System. A user can register for a new account, log in with valid credentials, browse through available food items, place and manage orders, make secure payments, and provide ratings for food and services. The system is designed to cater to all the needs of the user, from account creation to order completion, while ensuring a smooth and secure experience.

The **Signup** process enables a new user to register an account by providing essential personal details such as name, email, phone number, and password. The system validates this data for correctness and securely stores it in the database to ensure privacy and account protection.

Once registered, the **Login** use case allows the user to access their account using their unique ID and password. The system authenticates the entered details before granting access to the user dashboard, maintaining account security.

The **Browse Food** feature allows users to view the available menu items along with their categories, descriptions, prices, and ongoing offers. To enhance the user experience, the system includes search and filter options, enabling customers to quickly locate their desired food items.

When a user is ready to purchase, the **Place Order** feature lets them confirm their cart and submit the order. The system records the details, generates a unique order ID, and sends a confirmation notification to the user.

The **View My Orders** feature enables users to check both their current and past orders, including details such as items purchased, quantity, total price, payment status, and estimated delivery time.

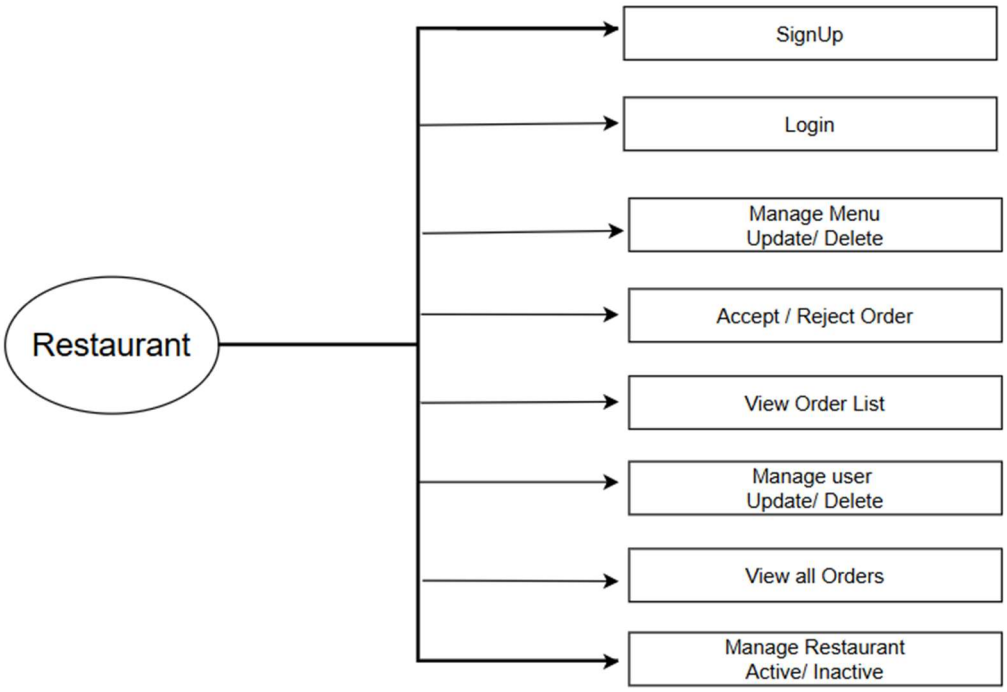
With the **Delivered or Not** feature, users can track the real-time status of their orders. The system updates the status to reflect whether the order is pending, in preparation, out for delivery, or completed.

The **Cancel Order** option allows users to cancel an order before it has been dispatched. If the payment was made online, the system initiates the refund process as per the set policies.

For transactions, the **Payment** feature provides multiple options including credit/debit cards, online wallets, UPI, and cash on delivery. The use of secure payment gateways ensures that all transactions are processed safely and efficiently.

Through the **Edit Profile** feature, users can update their personal details such as contact information, delivery addresses, and passwords, ensuring that their account remains accurate and up to date.

- **Admin :**



The Admin Module in the Online Food Ordering System plays a crucial role in ensuring smooth operations from the service provider's side. While the user module focuses on ordering and tracking food, the restaurant module ensures that the backend processes—such as menu management, order processing, and account administration—are handled efficiently. The restaurant acts as the primary service provider and interacts with the system to update offerings, manage customer orders, and maintain the quality of service.

The use case diagram illustrates the different operations available to the restaurant within the system. Each operation ensures that the restaurant can maintain control over its profile, manage its offerings, interact with customer requests, and oversee the entire ordering process.

For admin that wish to partner with the platform, the **Signup** process is the initial step. The restaurant provides essential details such as business name, contact information, location, food license number, and password. Once the system verifies the details, the restaurant account is created, enabling them to access their control panel. This process ensures that only verified and legitimate businesses are onboarded, maintaining platform credibility.

Once registered, the admin can use the **Login** functionality to securely access its account. The system authenticates the credentials provided before granting access. This ensures that only authorized staff can make changes to menus, manage orders, and update restaurant information.

A restaurant's menu is one of its most critical components in the system. Through the Manage Menu feature, restaurants can add new dishes, update existing menu items, modify prices, and remove unavailable products. The ability to instantly update the menu ensures that customers always see the most accurate and current offerings, reducing confusion and improving service reliability.

Once a customer places an order, it is sent to the restaurant for review. Using the **Accept / Reject Order** function, the restaurant can confirm the order if it can fulfill it or reject it if items are unavailable or operational constraints exist. This prevents delays and ensures that customers receive only confirmed orders.

The **View Order List** feature provides restaurants with a real-time display of all pending orders, along with relevant details such as customer name, order items, quantity, delivery instructions, and payment status. This organized view helps in prioritizing preparation and ensuring timely deliveries.

The **Manage User** functionality allows restaurant administrators to update or remove user accounts associated with their restaurant operations. This is particularly useful for managing employee access, ensuring that only active and authorized personnel can log in and manage the restaurant's orders and menu.

While the View Order List feature focuses on pending orders, the **View All Orders** option provides a complete historical record of all customer transactions, including completed, cancelled, and refunded orders. This data is useful for performance analysis, revenue tracking, and resolving customer disputes.

The admin module is designed to provide **control**, **efficiency**, and **flexibility** to service providers. By allowing instant updates to menus and operational statuses, restaurants can adapt quickly to changes in demand or availability. The order management system minimizes errors, improves order fulfillment speed, and ensures clear communication with customers. Additionally, detailed records of past transactions support better business planning and customer service.

The use case diagram for the restaurant module showcases how the system empowers restaurants to manage their business efficiently within an online ordering platform. From onboarding and menu management to order processing and operational control, each functionality works together to create a streamlined and reliable service. This not only benefits restaurants by improving operational workflows but also enhances the overall customer experience by ensuring accuracy, timeliness, and professionalism in service delivery.

The Online Food Ordering System offers numerous benefits to both customers and service providers. Convenience is one of its main advantages, allowing users to order food anytime and from anywhere. The system ensures efficiency by processing orders faster than traditional methods, reducing waiting times and improving service speed.

2.2 NON-FUNCTIONAL REQUIREMENTS

1. **Interface :**

   Go to Appendix B for user interfaces.

2. **Other Requirements :**

- **Hardware Requirements**

  The SPMS is expected to function on Intel PIII 900 MHz Processor equivalent or above,128 MB RAM, 20 GB HDD.

- **Software Requirements**

  The SPMS shall work on MS Windows operating systems family (MS Windows 98, MS Windows NT Workstation, MS Windows 2000, MS Windows XP). It configures to work with Oracle database. This System works on Apache Tomcat server. It uses browser IE 5.0 & above. It uses IIS 5.0 server.

DESIGN

- **Database Design**

Table 1: User

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null (1=Yes;0=No)
3	id	int	4	0
0	email	varchar2	255	1
0	name	varchar2	255	1
0	Password	varchar2	255	1

Table 2: Food

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null (1=Yes;0=No)
3	id	int	4	0
0	category	varchar2	255	1
0	description	varchar2	255	1
0	image	varchar2	255	1
0	name	varchar2	255	1
0	price	double	8	1

Table 3: Orders

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null (1=Yes;0=No)
3	id	int	4	0
0	city	varchar2	255	1
0	country	varchar2	255	1
0	postal_code	varchar2	255	1
0	state	varchar2	255	1
0	street	varchar2	255	1
0	amount	double	255	0
0	date	datetime	6	1
0	payment	bit	1	0
0	status	varchar2	255	1
0	user_id	int	4	1

Table 4: Order_Items

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null (1=Yes;0=No)
3	order_id	int	4	0
0	name	varchar2	255	1
0	price	double	8	1
0	quantity	int	4	1

Table 5: User_Cart

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null (1=Yes;0=No)
3	user_id	int	4	0
0	quantity	int	4	1
0	Item_id	int	4	0

E-R Diagram,Dataflow diagram and Class Diagram:

Go to Appendix A

CODING STANDARDS IMPLEMENTED

Naming and Capitalization

Below summarizes the naming recommendations for identifiers in Pascal casing is used mainly (i.e. capitalize first letter of each word) with camel casing (capitalize each word except for the first one) being used in certain circumstances.

Identifier	Case	Examples	Additional Notes
Class	Pascal	Person, BankVault, SMSMessage, Dept	Class names should be based on "objects" or "real things" and should generally be **nouns**. No '_' signs allowed. Do not use type prefixes like 'C' for class.
Method	Camel	getDetails, updateStore	Methods should use **verbs** or verb phrases.
Parameter	Camel	personName, bankCode	Use descriptive parameter names. Parameter names should be descriptive enough that the name of the parameter and its type can be used to determine its meaning in most scenarios.
Interface	Pascal with "I" prefix	Disposable	Do not use the '_' sign
Property	Pascal	ForeColor, BackColor	Use a noun or noun phrase to name properties.
Associated private member variable	_camelCase	_foreColor, _backColor	Use underscore camel casing for the private member variables
Exception Class	Pascal with "Exception" suffix	WebException	

Comments

- Comment each type, each non-public type member, and each region declaration.
- Use end-line comments only on variable declaration lines. End-line comments are comments that follow code on a single line.
- Separate comments from comment delimiters (apostrophe) or // with one space.
- Begin the comment text with an uppercase letter.
- End the comment with a period.
- Explain the code; do not repeat it.

TEST REPORT

Another group called Linux did the testing and the report of the testing is given here under.

SR-NO	TEST CASE	EXPECTED RESULT	ACTUAL RESULT	ERROR MESSAGE
1	Register Page	Redirected to Next page	OK	Nothing
2	Login Page	Pop-up will come	Ok	Please enter username and password again.
3	Reset login	Only user's password will be reseted	Ok	Nothing
4	Quick search food	Gives all flight details	Ok	Nothing
5	Ordering Food	All the fields should be filled for submission	Ok	Nothing
6	Checking login or not	User is logged in or not	Ok	Nothing
7	Add user's details	Add information according to no of seats allocated	Ok	Nothing
8	Goto order page	Set added information about person	Ok	Nothing
9	Add information in order table	Save this all data into booking table	Ok	Nothing
10	Transaction	On back it should be reverted to previous page	Ok	Nothing
11	View transaction done	It shows you all transactions done previously	Ok	Nothing
12	Logout	It will logout from user profile.	Ok	Nothing

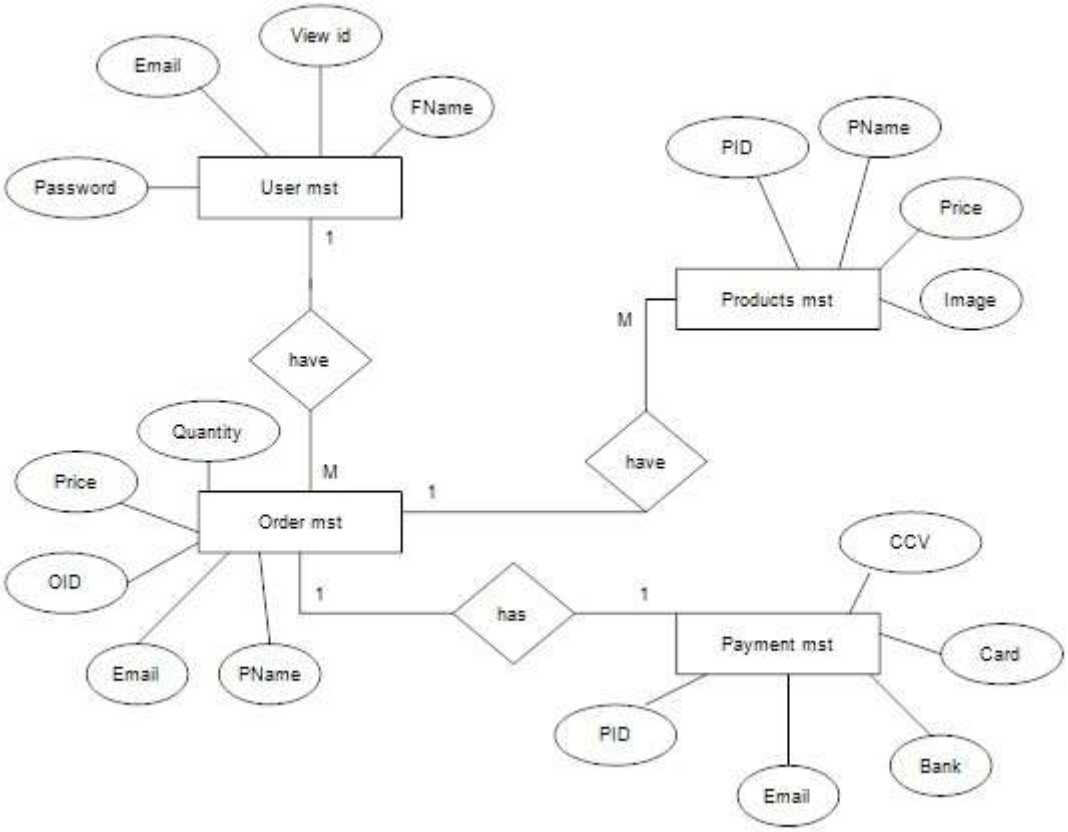
PROJECT MANAGEMENT RELATED STATISTICS

DATE	WORK PERFORMED		Additional Notes
JULY 11, 2025	Project Allotment and User Requirements Gathering	Feasibility Study	Our team met the client Mr. Nitin Kudale (CEO, SIIT Pune) to know his requirements.
JULY 21, 2025	Initial SRS Document Validation And Team Structure Decided	Requirement Analysis (Elicitation)	The initial SRS was presented to the client to understand his requirements better
JULY 22, 2025	Designing the use-cases, Class Diagram, Collaboration Diagram, E-R Diagram and User Interfaces	Requirement Analysis & Design Phase	Database Design completed
JULY 23, 2025	Business Logic Component Design Started	Design Phase	--
JULY 24, 2025	Coding Phase Started	Coding Phase	70% of Class Library implemented.
JULY 25, 2025	Implementation of Web Application and Window Application Started	Coding Phase	Class Library Development going on.
JULY 26, 2025	Implementation of Web Application and Window Application Continued	Coding Phase and Unit Testing	Class Library Modified as per the need.
JULY 27, 2025	Off	Off	Off
JULY 28, 2025	Implementation of Web Application and Window Application Continued	Coding Phase and Unit Testing	--

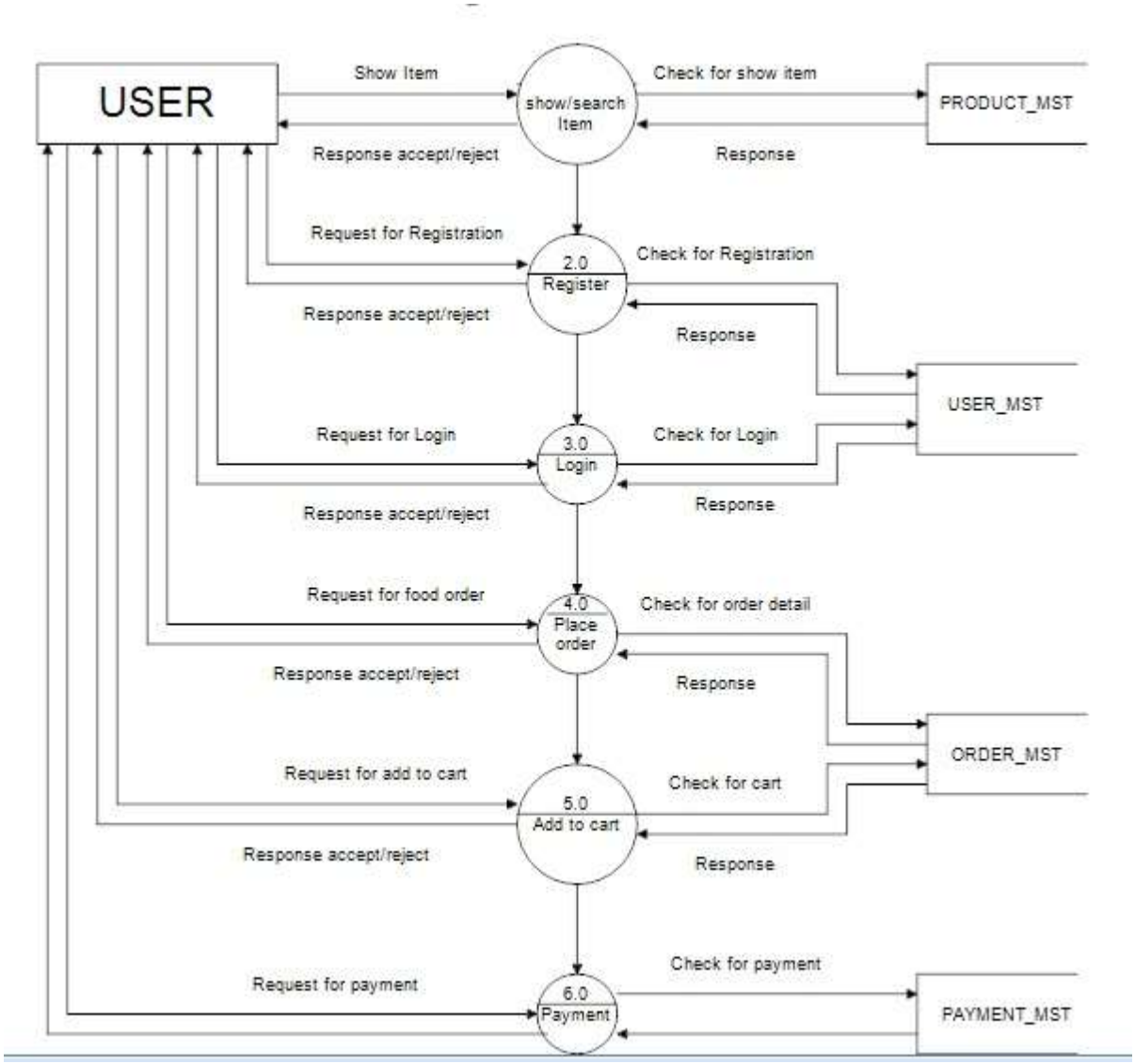
JULY 29 – AUG 03, 2025	After Ensuring Proper Functioning the Required Validations were Implemented	Coding Phase and Unit Testing	Module Integration was done by the Project Manager
AUG 04 - 05, 2025	The Project was Tested by the respective Team Leaders and the Project Manager	Testing Phase (Module Testing)	--
AUG 06 - 08, 2025	The Project was Submitted to Other Project Leader of Other Project Group for Testing	Testing Phase (Acceptance Testing)	The Project of Other Team was Taken up by the Team for Testing
AUG 09 - 10, 2025	The Errors Found were Removed	Debugging	The Project was complete for submission
AUG 11, 2025	Final Submission of Project		--

APPENDIX A

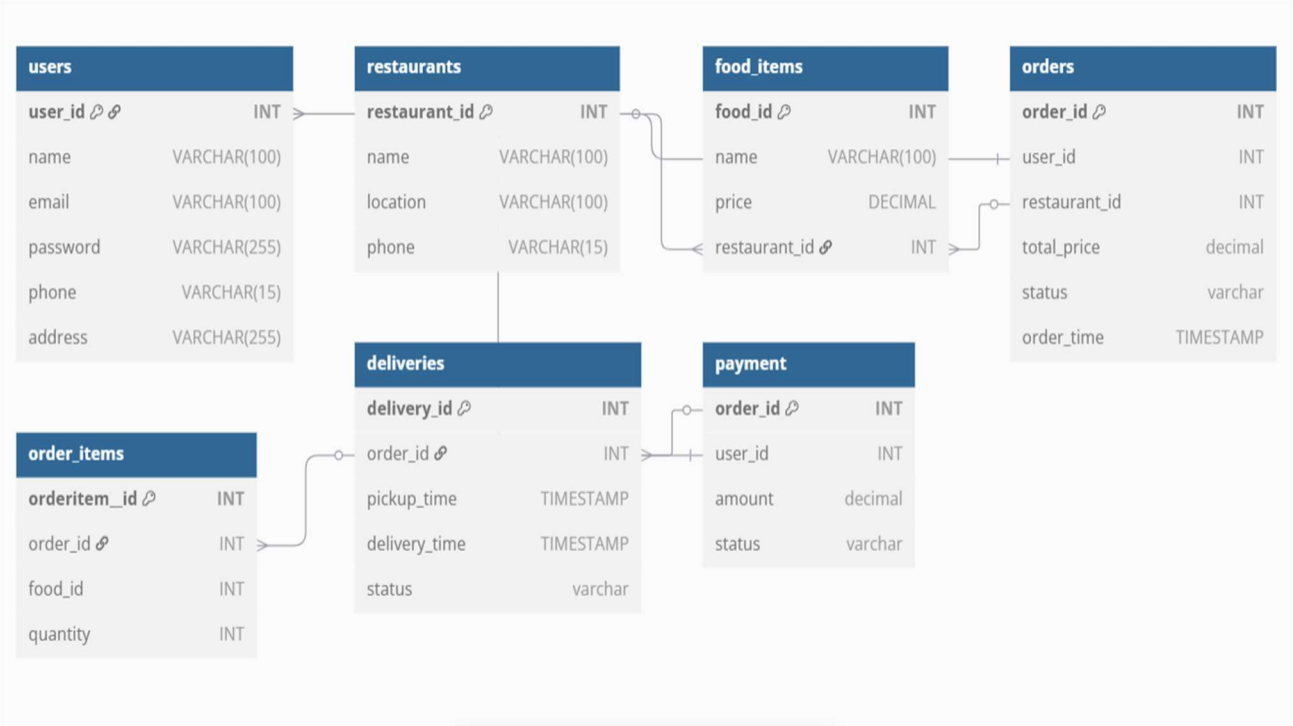
- **Entity Relationship Diagram**



- **Data Flow Diagram**

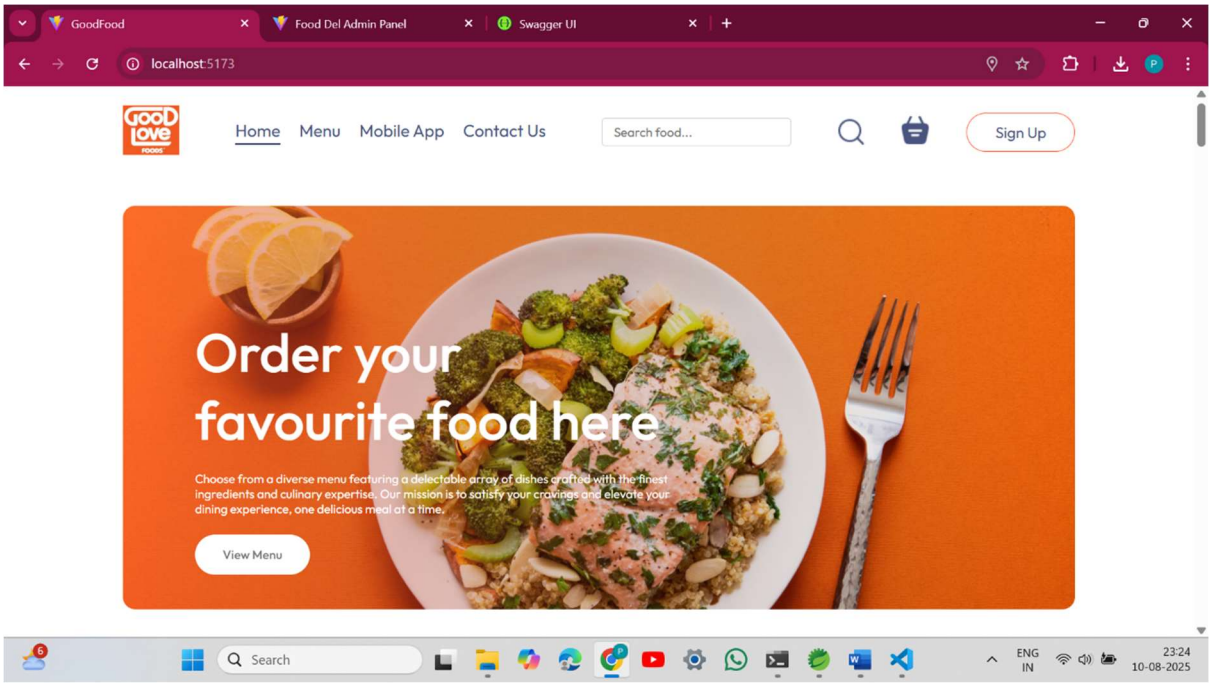


- **Class Diagram**

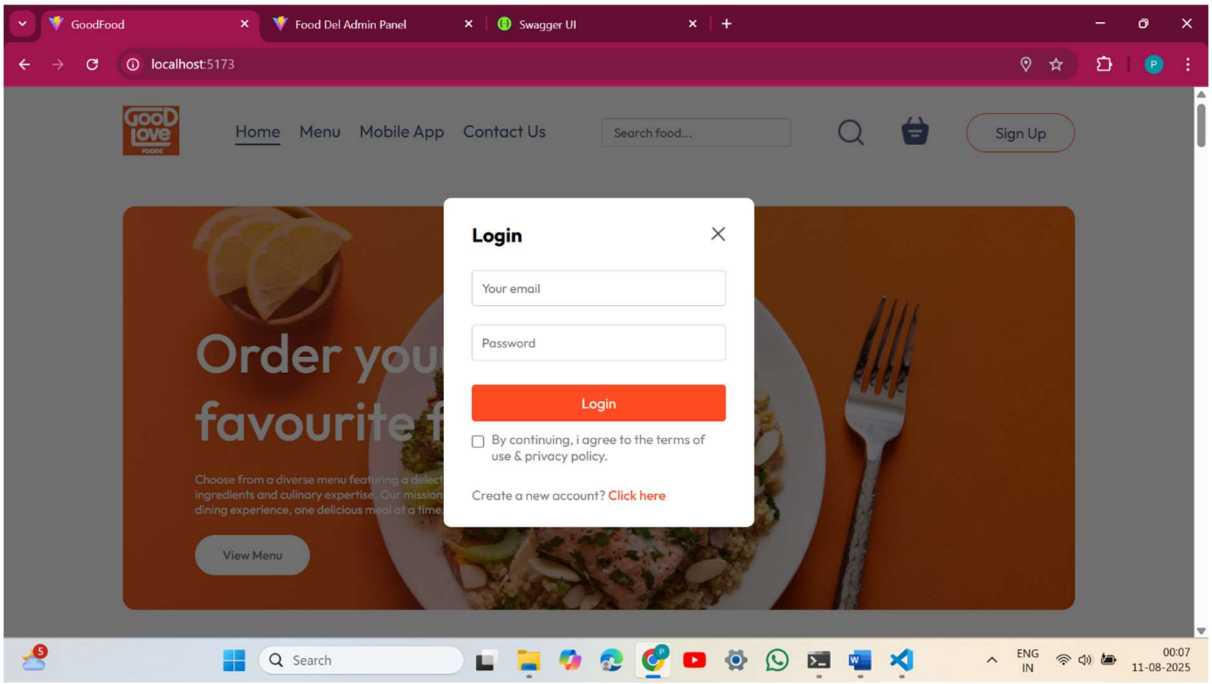


APPENDIX B

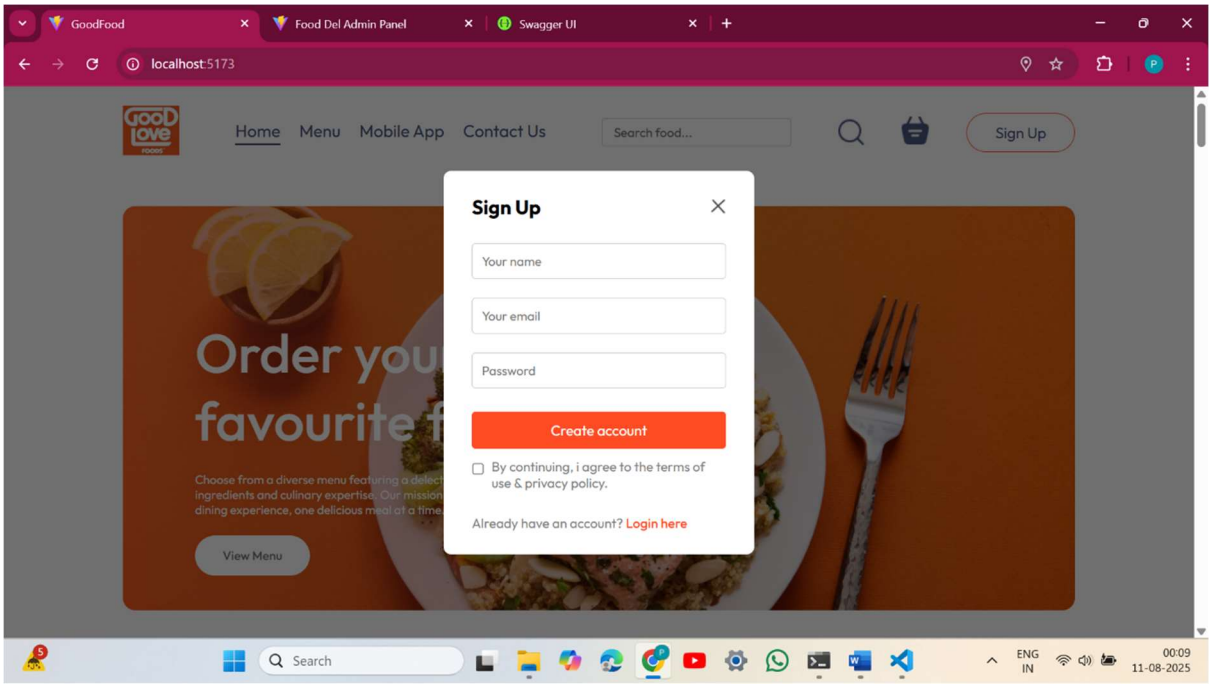
Homepage:



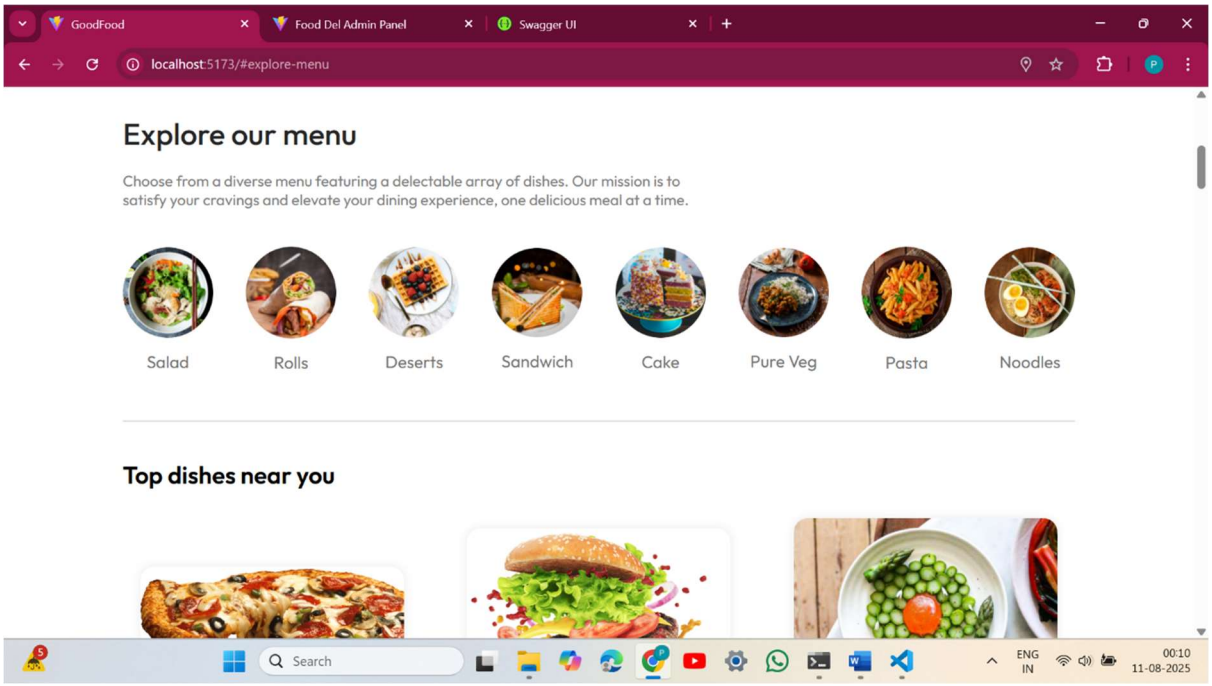
Login:



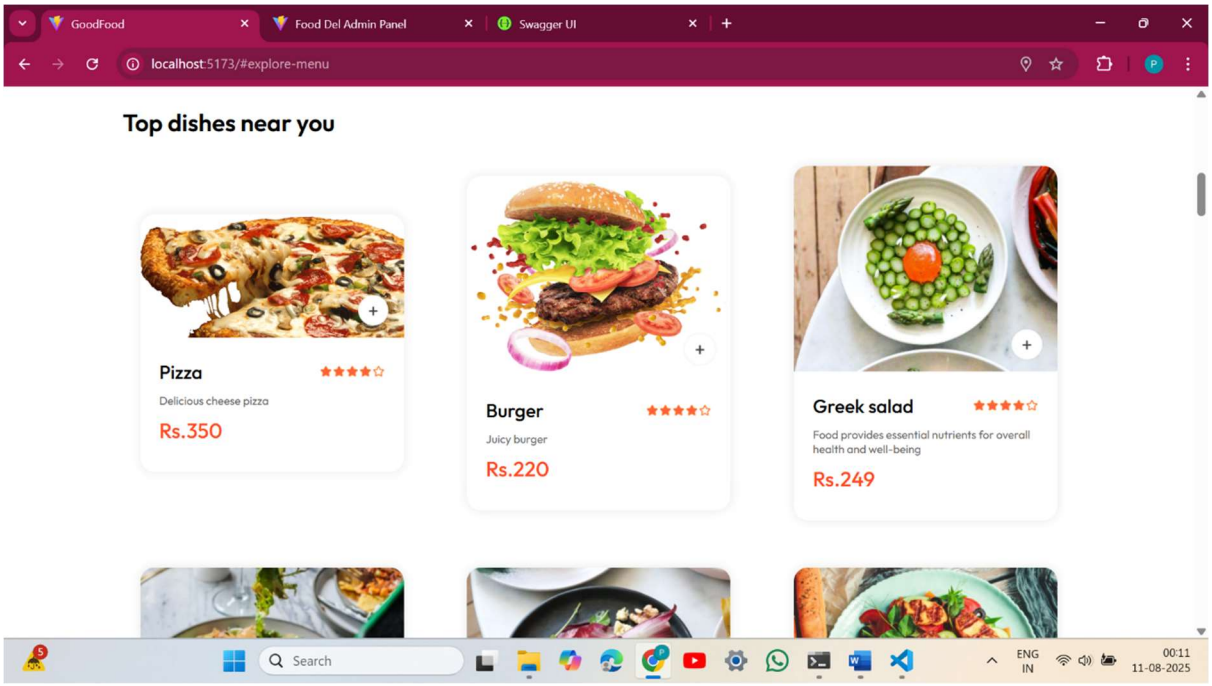
Register:



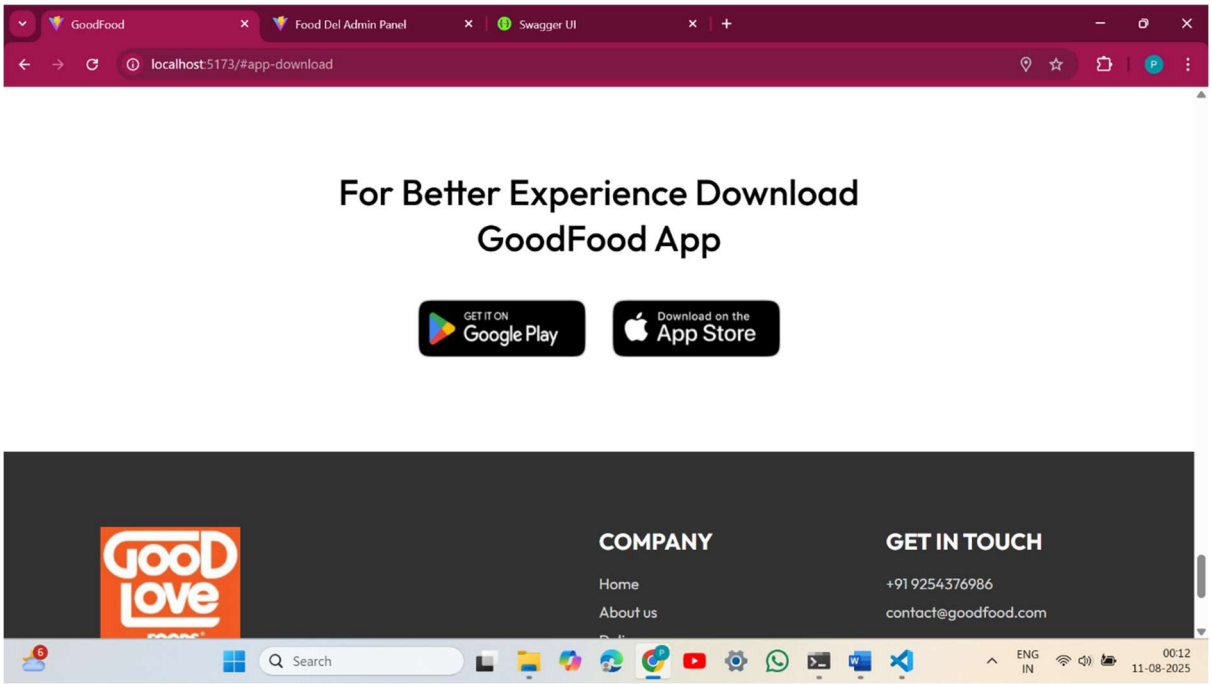
Explore Menu:



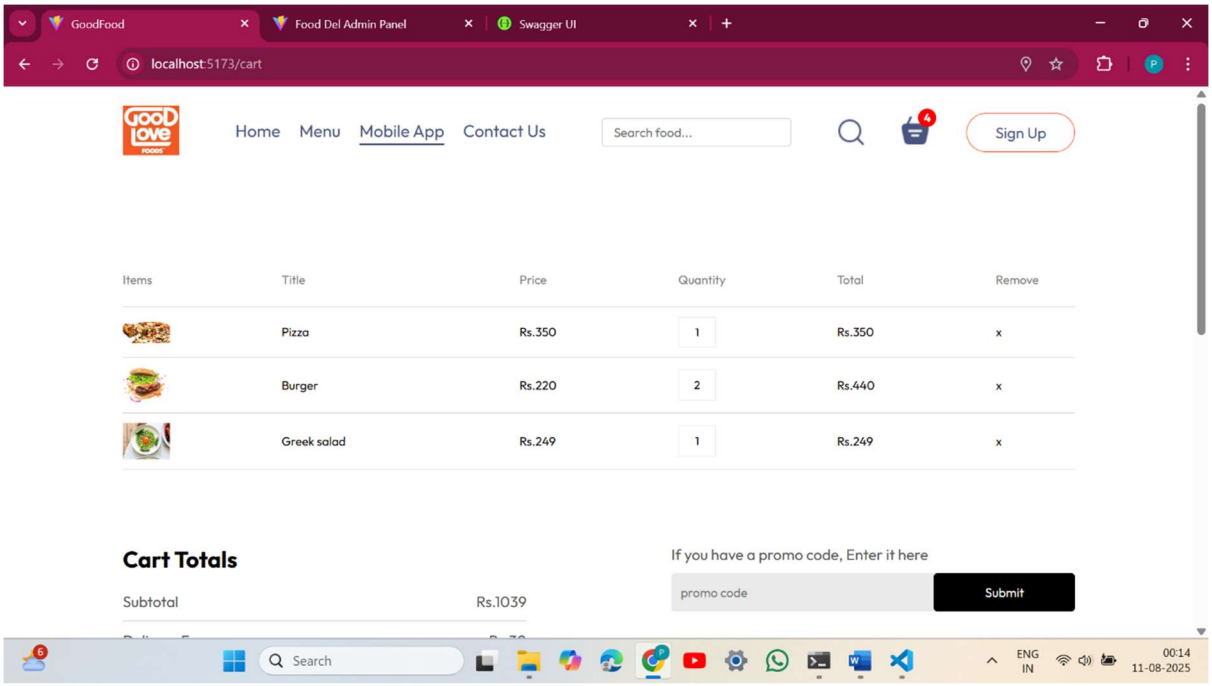
Food Item:



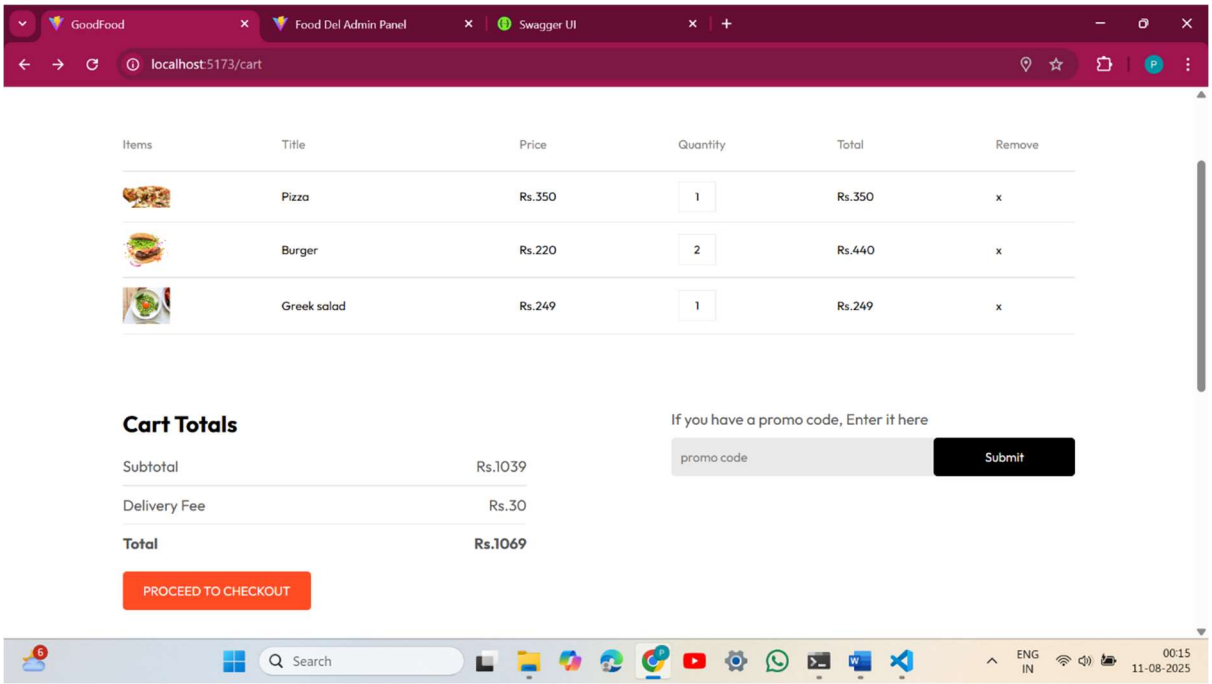
Mobile App:



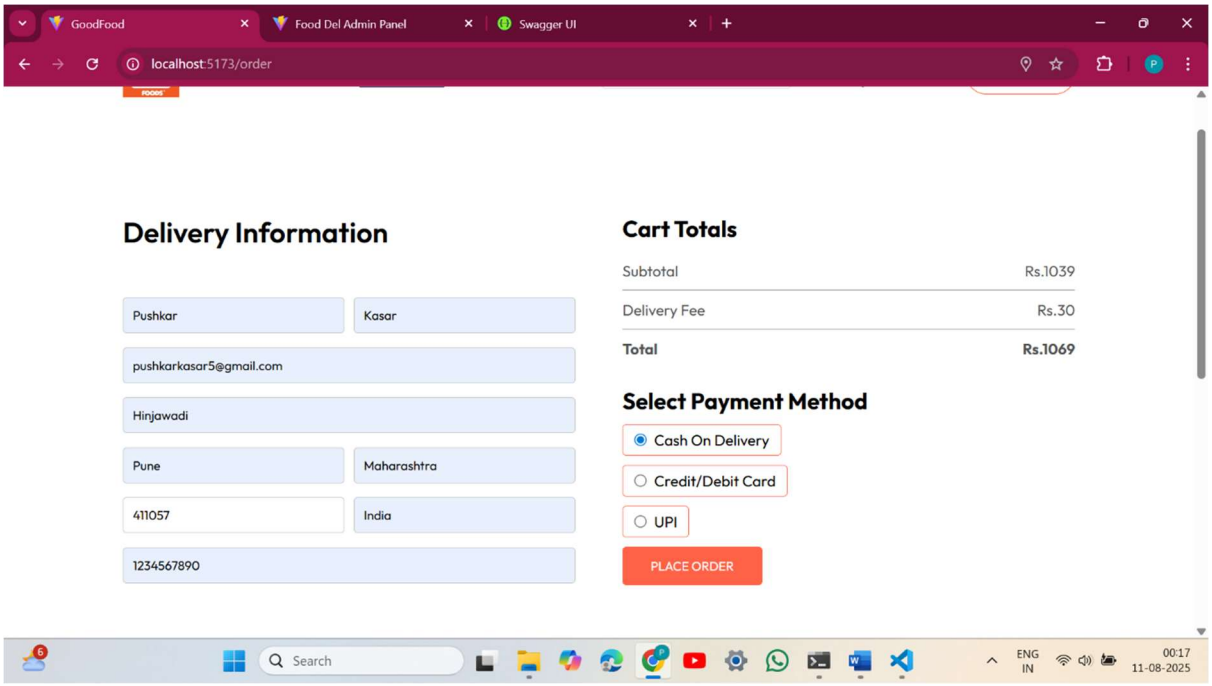
Order List:



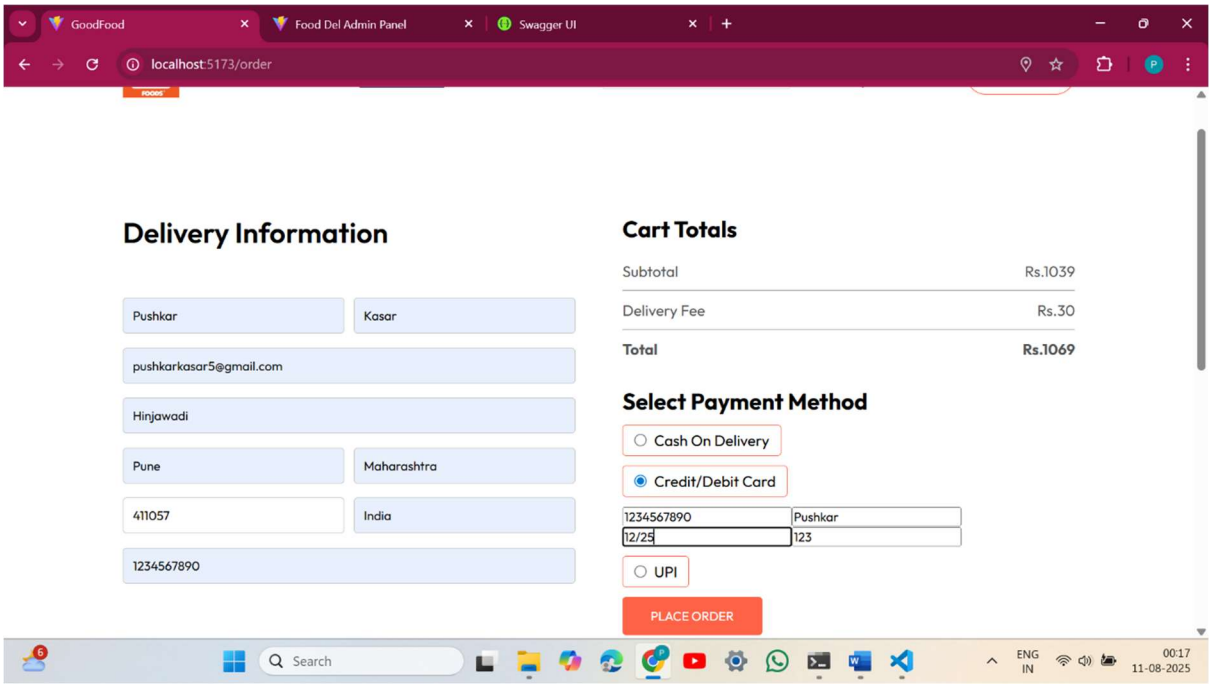
Cart:



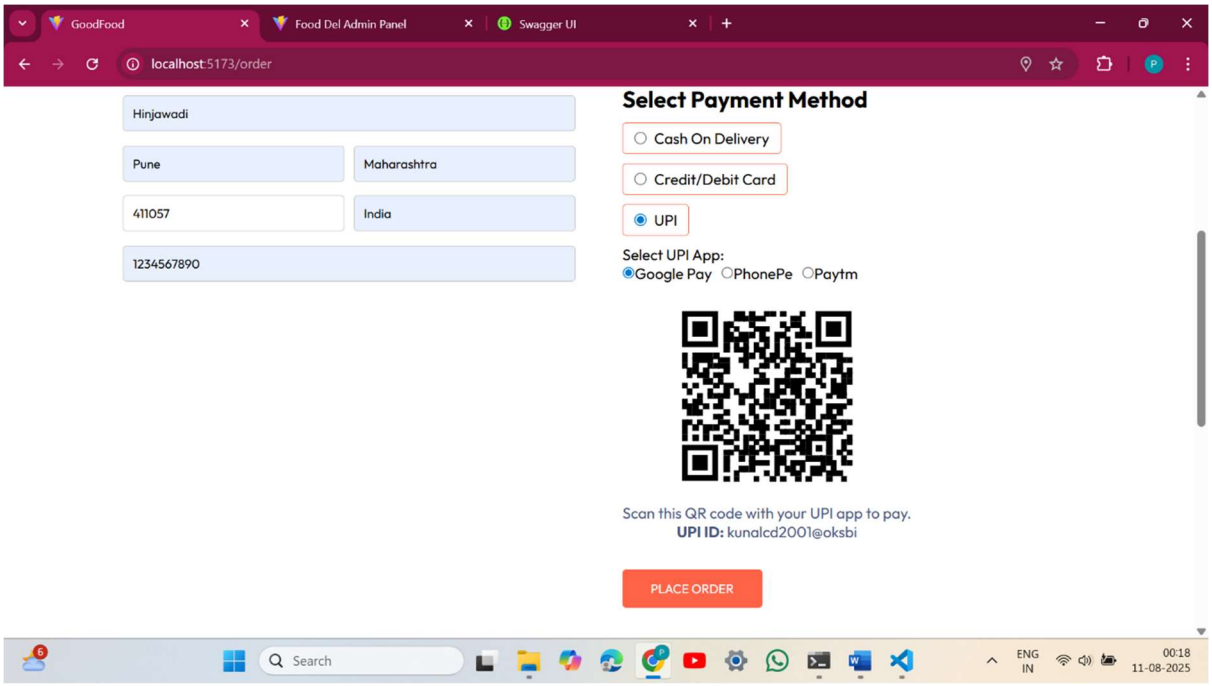
Payment Cash Gateway:



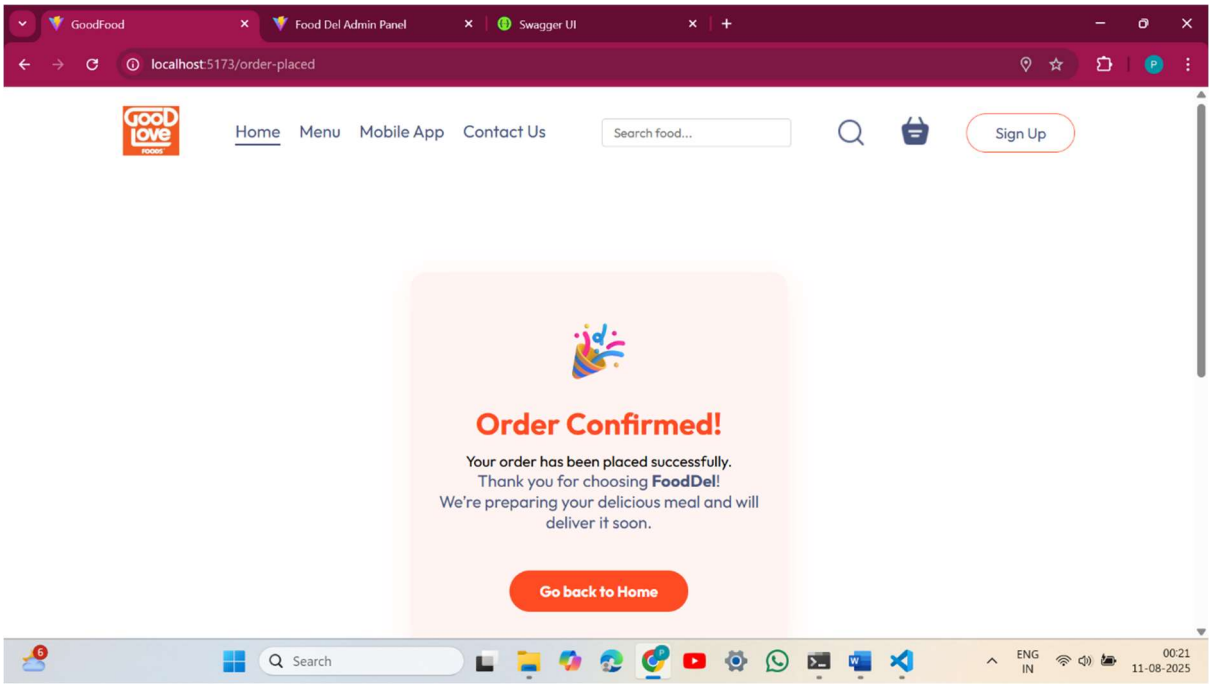
Payment Card Gateway:



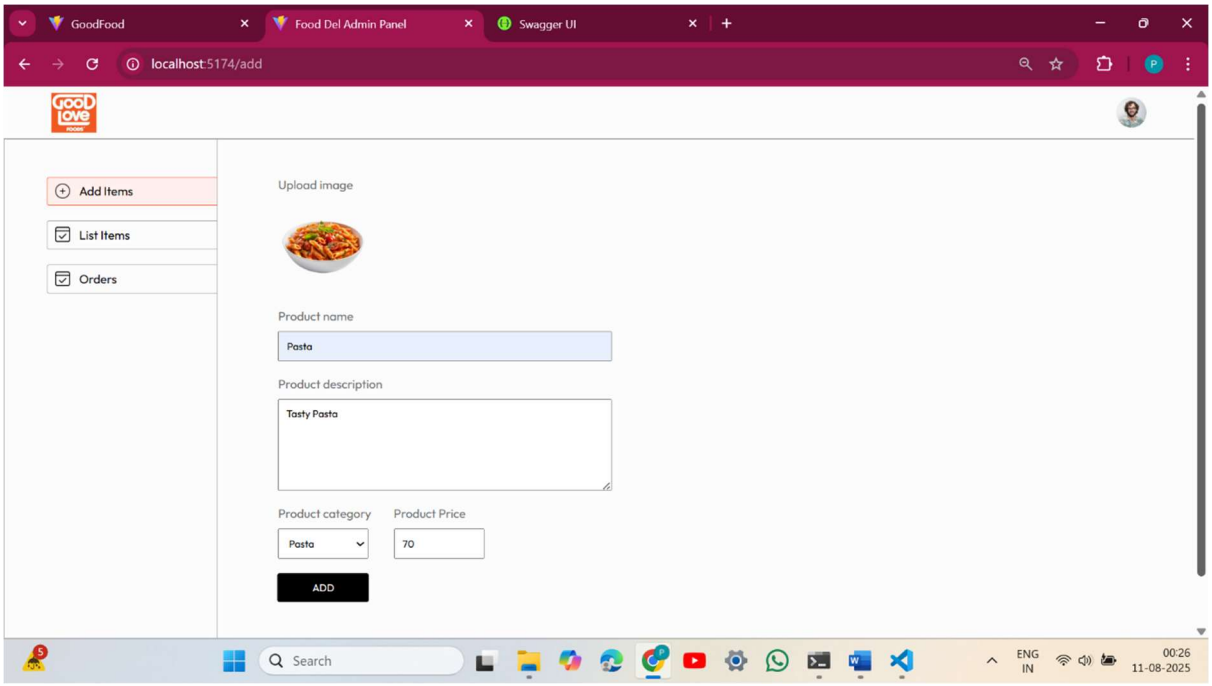
Payment UPI Gateway:



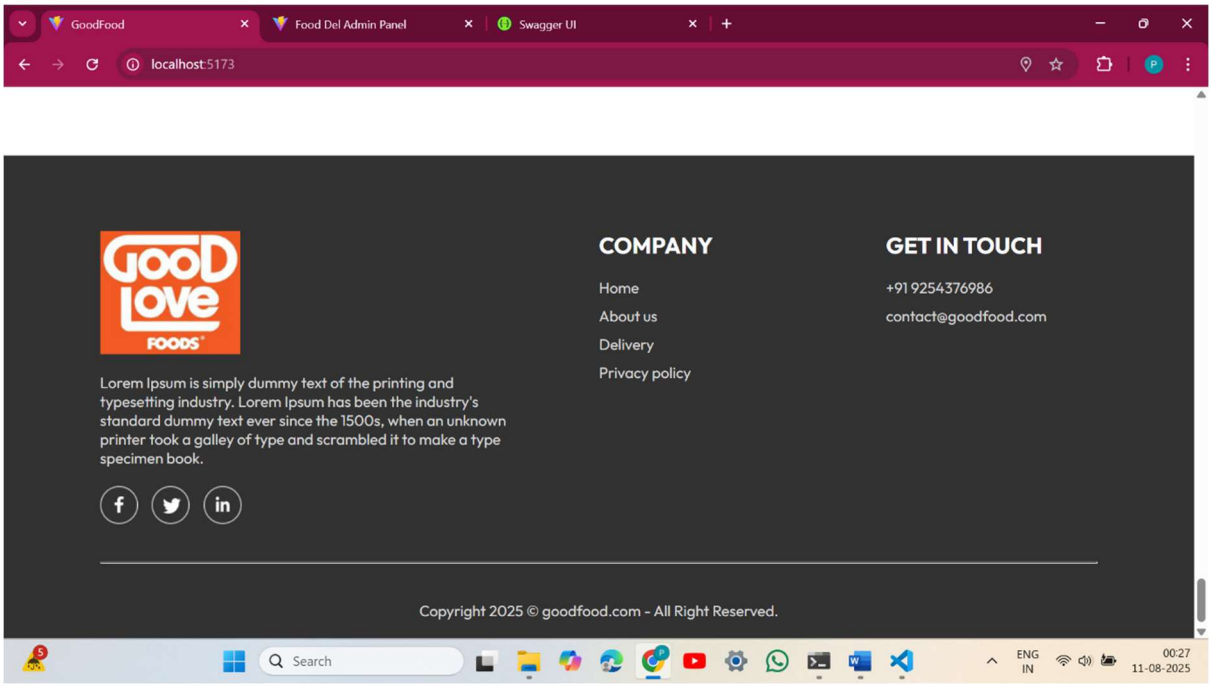
Order Confirmed:



Add Food:



Footer:



REFERENCES

https://arxiv.org/abs/2209.04614

https://www.researchgate.net/publication/363501951

https://arxiv.org/abs/2002.01713

https://www.uber.com/blog/engineering/