

iFOOD

iFood Online Ordering System Final Presentation

“Revolutionising the food industry with seamless online ordering experience.”

Presented by: **Group Alpha**

Group: **M2**

Supervisor Signature: 

All UML diagrams are too small. Must re-draw before presenting



Group Members

M.S. Pietersz
Team Leader

J. Sarujan
Team Member

V. Vijenthuja
Team Member

M.B. Kishor Jahan
Team Member



Introduction to the Project

Project Overview

- designed to improve the food ordering experience for customers by providing a seamless, efficient, and user-friendly platform.
- includes features such as menu browsing, order placement, and payment processing



Project Objectives

- Improve order accuracy and minimise communication errors.
- Enhance delivery efficiency and address verification to reduce failed deliveries.
- Simplify payment processing for secure and efficient transactions.
- Provide real-time visual order confirmation for users.



Functional Requirements

- ★ **User Account Management** (Register, login, logout, and manage profiles.)
- ★ **Restaurant and Menu Browsing** (Browse restaurants, view menus, and filter by location, cuisine, and ratings.)
- ★ **Order Tracking** (Real-time order status updates and delivery tracking using location services.)
- ★ **Restaurant Management** (Manage restaurant profiles, menus, and order statuses.)
- ★ **Admin Panel** (Admin interface for managing users, restaurants, and system settings.)

Non-Functional Requirements

❖ Performance & Load

- Supports 900 - 1000 concurrent users; scales for 500-600 new users.
- Transaction time: 5-10 seconds.

❖ Compatibility

- HTML5; Chrome, Firefox, Safari, Edge (latest versions).
- Platforms: Windows 10/11, macOS 10.15+.

❖ Security

- Secure login and data encryption (SSL/TLS).
- End-to-end data protection.

Non-Functional Requirements

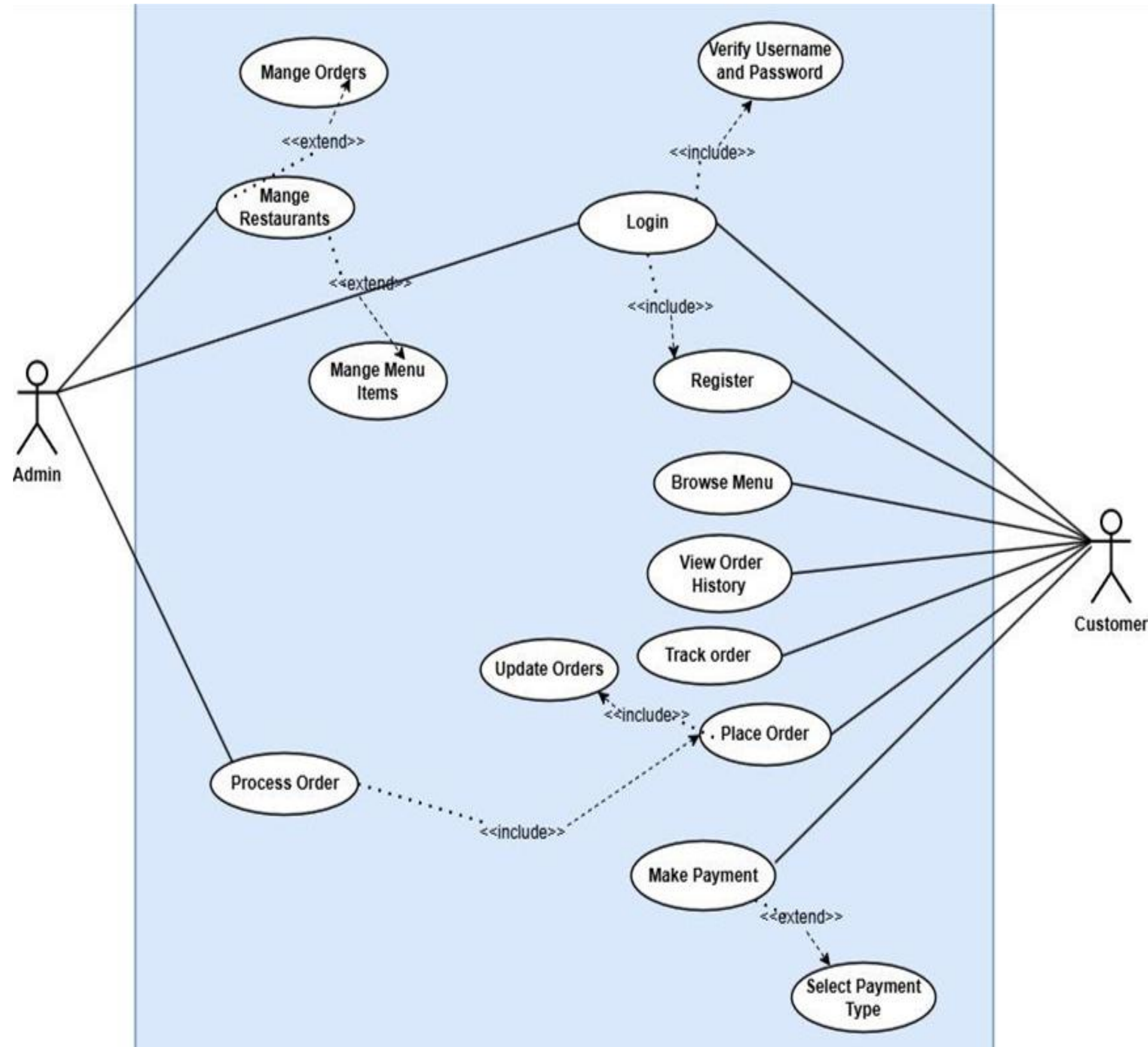
❖ Quality Assurance

- Testing: Unit, Integration, System, Performance, UAT.
- Tools: Selenium, JMeter.

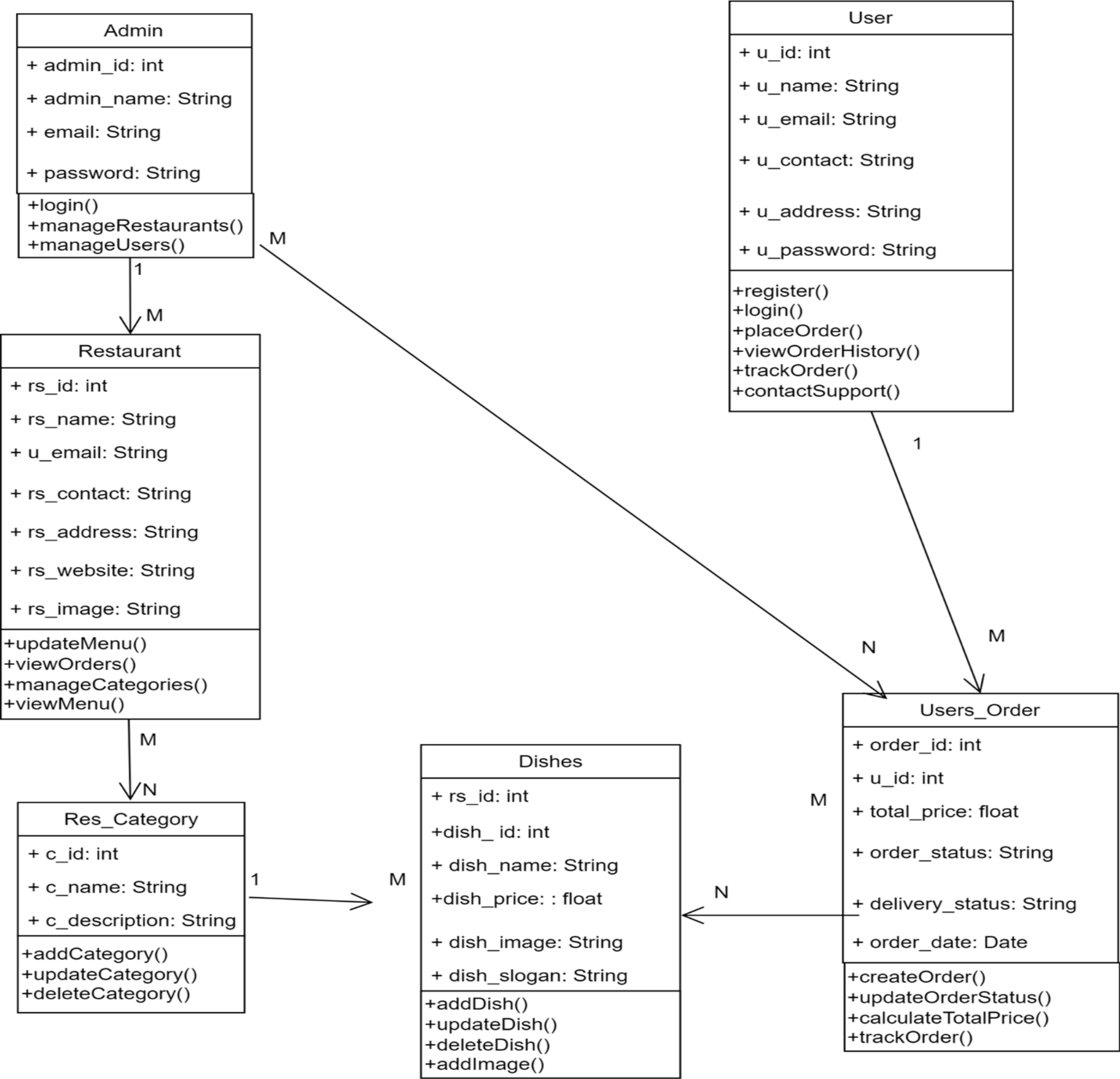
❖ Usability

- Responsive design; major browser support.
- Optimized for PC/laptop use.

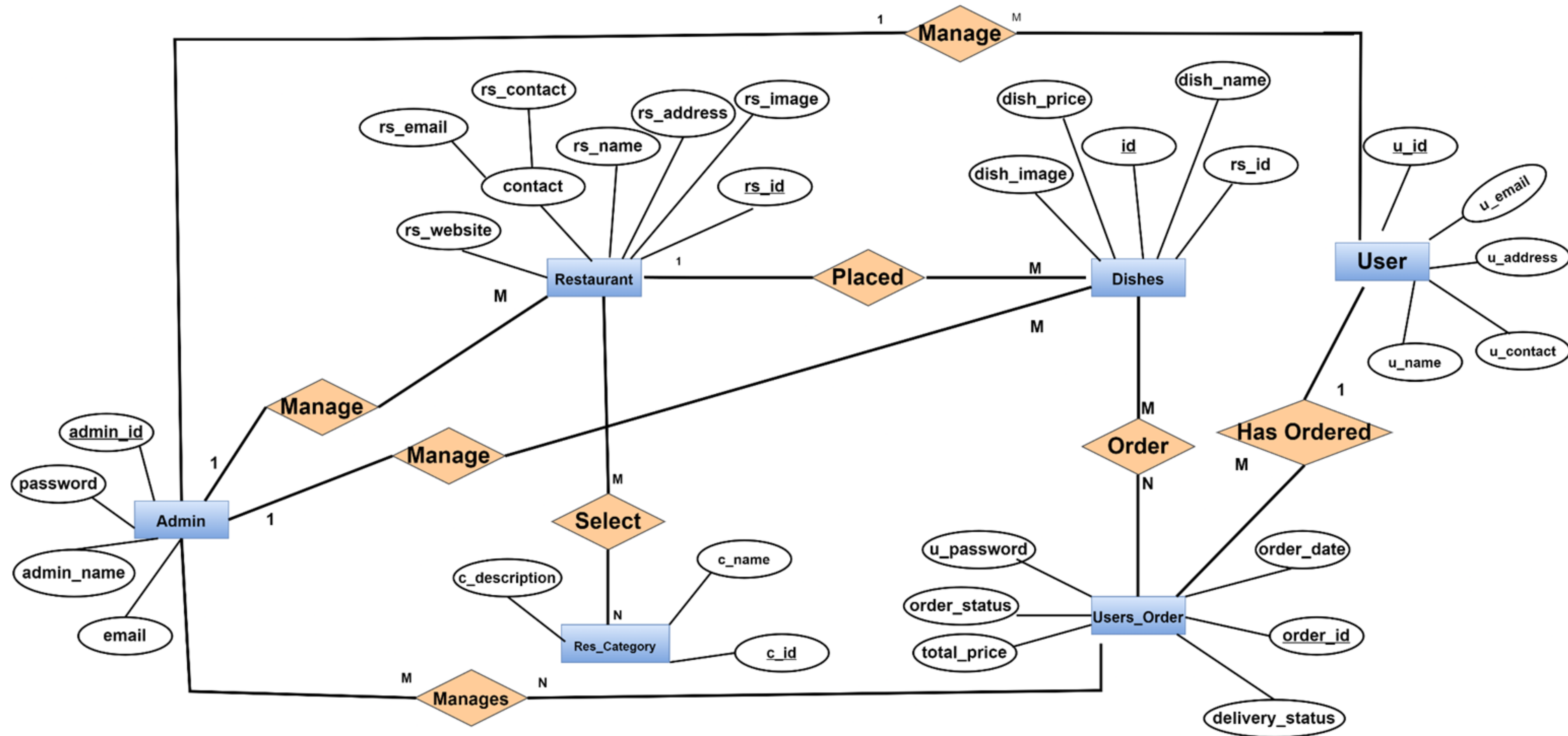
Use Case Diagram



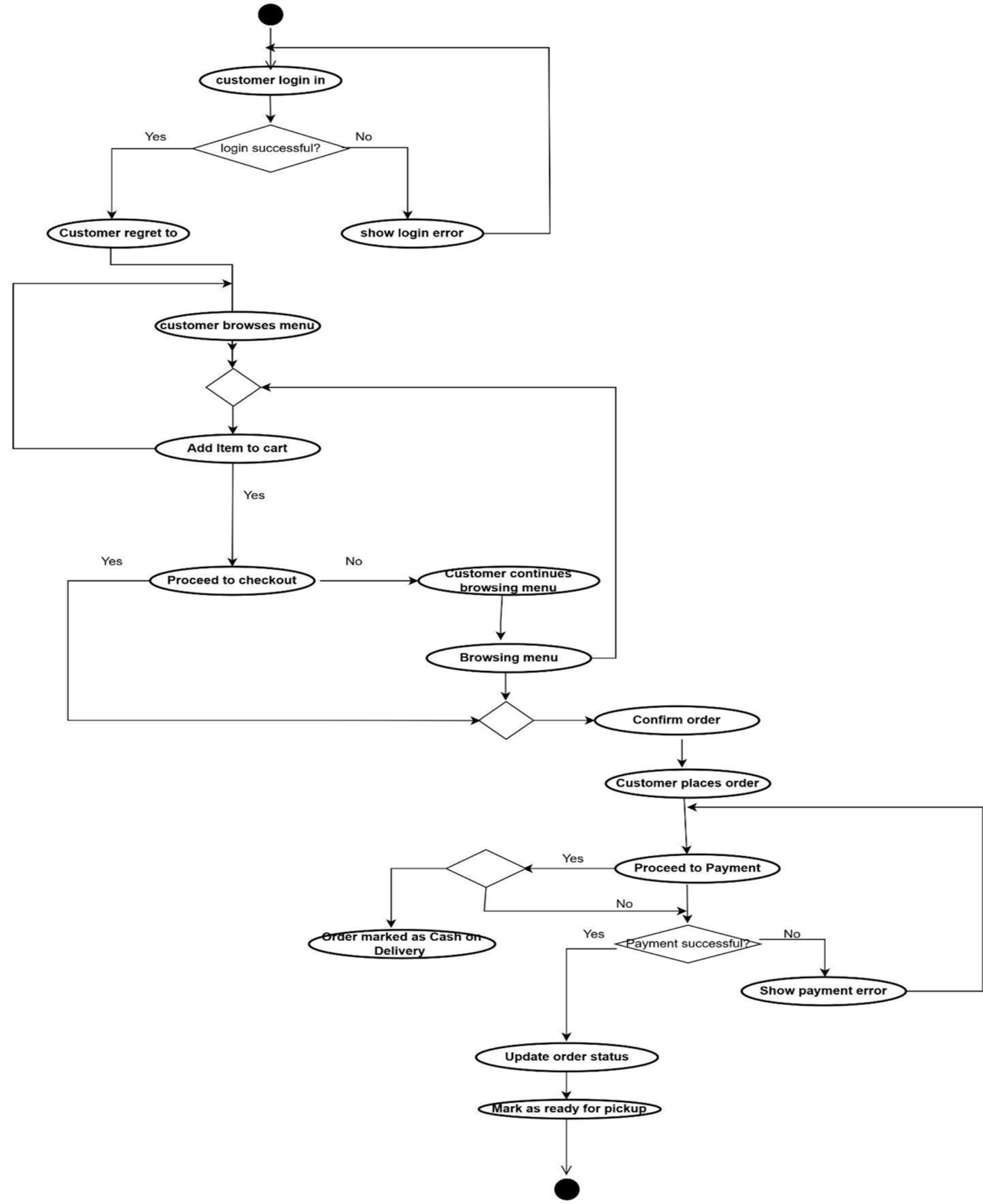
Class Diagram



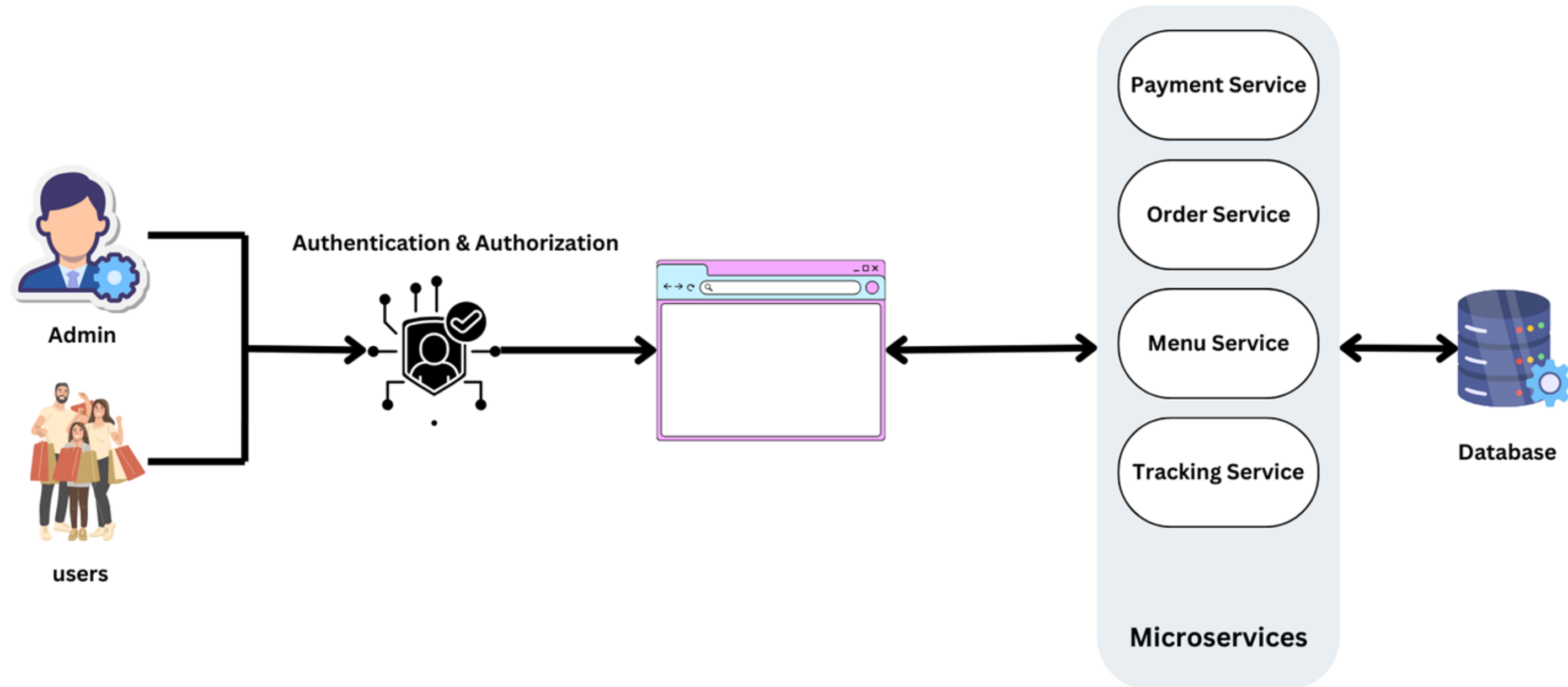
ER Diagram



Activity Diagram



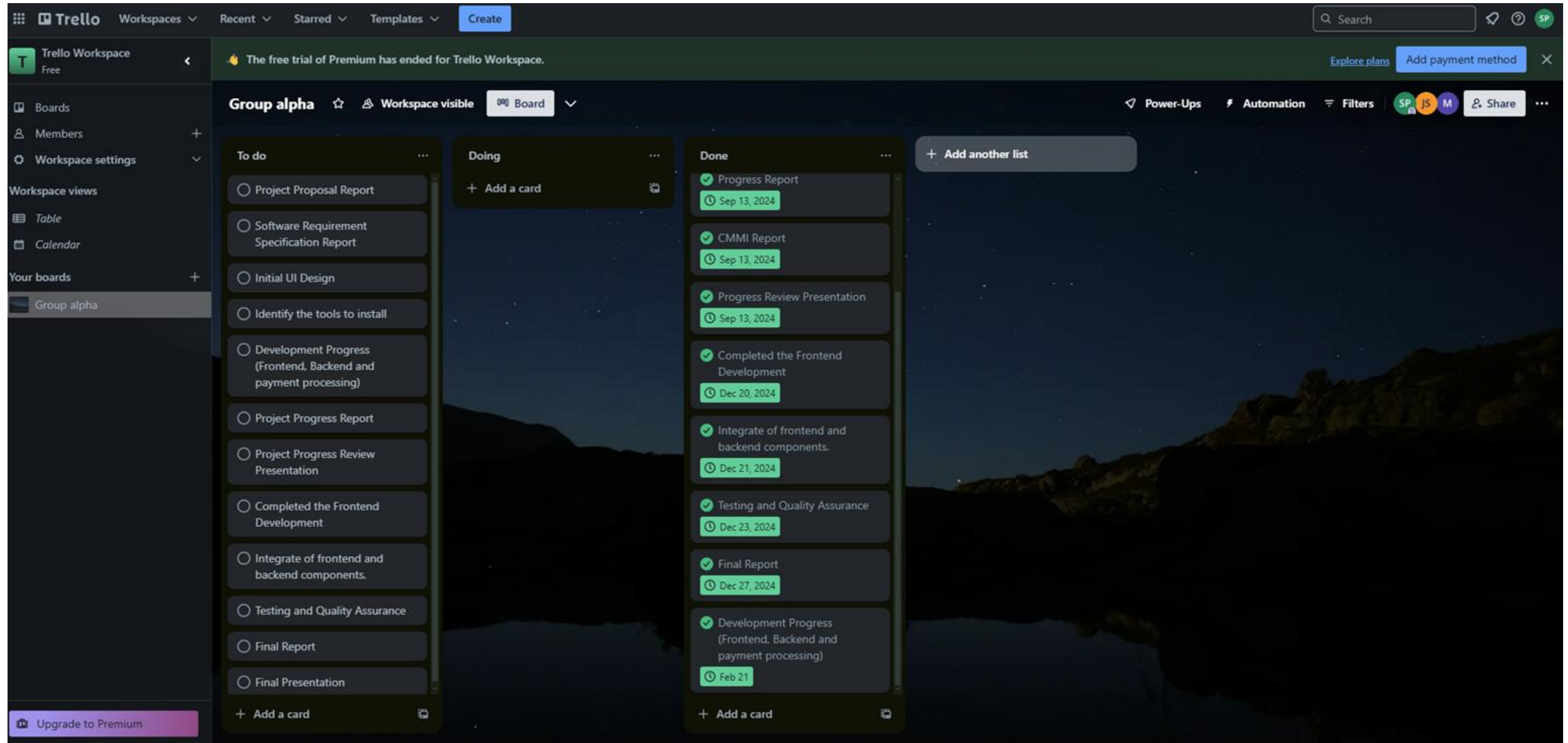
Overall architecture-block diagram



Test Case Document and Test results





Test Case ID	Test Case Description	Pre-Conditions	Test Steps	Expected Result	Actual Result	Test Result
#001	Verify the "User Login" Page	The system is up and running.	Open login page. Enter valid login credentials. Click "Login" button.	The user is logged in successfully, and the home page is displayed.	The user login was successful, and the home page was displayed.	Pass
#002	User View each restaurant menu items	The system is up and running.	Navigate to the "Restaurant menu items" section.	The User can view each all-Restaurant menu items.	The user is presented with each all-Restaurant menu items.	Pass
#003	Validate that users can add items to their cart.	The system is up and running.	Browse the menu. Select an item and click "Add to Cart." Verify the cart is updated.	Selected item appears in the cart.	cart updated correctly	Pass
#004	Confirm users can place orders successfully.	The system is up and running.	Add items to the cart. Proceed to check out. Complete payment.	Order confirmation page is displayed	Order placed successfully	Pass
#005	Verify that order details are correctly displayed in the "My Order" table.	The system is up and running.	Navigate to the "My Orders" section. Verify that the table displays the correct order details.	The "My Order" table accurately displays all order details with the correct data	The my orders table displayed all order details as expected, including item name, quantity, price, status, date, and action options	Pass
#006	Verify the "Admin Login" Page	The system is up and running.	Open login page. Enter valid login credentials. Click "Login" button.	The admin is logged in successfully, and the home page is displayed.	The admin login was successful, and the home page was displayed.	Pass
#007	Verify that the admin can edit restaurant details.	The system is up and running.	Navigate to the " Restaurants" section. Select all restaurants and click the "Edit" button. Modify details. Save changes.	The admin can edit restaurant information.	The admin edited restaurant information.	Pass
#008	Verify that the admin can add a restaurant category.	The system is up and running.	Navigate to the " Restaurants" section. Select add category. Enter category details and save.	New category appears in the list.	Category added successfully	Pass
#009	Verify that the admin can edit a restaurant category.	The system is up and running.	Navigate to the " Restaurants" section. Select add category. and click the "edit " button. Modify details. Save changes.	The admin can edit restaurant category information.	The admin edited restaurant category information.	Pass
#010	Verify that the admin can add a new restaurant	The system is up and running.	Navigate to the " Restaurants" section. Select add Restaurant Enter category details and save.	New restaurant appears in the list.	Restaurant added successfully.	Pass
#011	Verify that the admin can edit menu details	The system is up and running.	Navigate to the "Manage Menus" section. Select a menu item. Modify details. Save changes.	Updated menu details are saved and displayed correctly.	Menu details updated successfully.	Pass
#012	Verify that the admin can add a new menu item	The system is up and running.	Navigate to the "Manage Menu" section. Click "Add Menu." Enter menu item details and save.	New menu items appear in the list.	menu item added successfully.	Pass
#013	Verify that the admin can update the order status	The system is up and running.	Navigate to the "Orders" section. Change the status to one of the predefined options. Save changes.	Updated status is displayed in the order list and reflected for the customer.	Order status updated successfully.	Pass

Trello Board Screenshot



GIT Account

The screenshot displays the GitHub interface for an organization named "Group-Alpha-Project". The top navigation bar includes a search bar and icons for repository management. The main navigation tabs are Overview, Repositories (1), Projects, Packages, Teams, People (4), and Settings. The left sidebar lists organization management options: Organization permissions, Members (4), Outside collaborators, Pending collaborators, Invitations, Failed invitations (2), and Security Managers. The main content area shows the "People" tab with a search bar and buttons for "Export" and "Invite member". A notification banner states: "You are the only owner of this organization! We recommend a minimum of two people within each organization have the owner role." Below this, a table lists the organization's members.

Members		2FA	Membership
<input type="checkbox"/>	 Kishor Jahan Mohamed Badurdeen s92067570kj	× 2FA	Private Member 0 teams 0 roles
<input type="checkbox"/>	 Sarujan-03	× 2FA	Private Member 0 teams 0 roles
<input type="checkbox"/>	 Stefenis	✓ 2FA	Private Owner 0 teams 0 roles
<input type="checkbox"/>	 Vijenthuja	× 2FA	Private Member 0 teams 0 roles

Monthly Retrospective

For what month you are going to fill your Retrospective *

- ☐ May
- ☒ June
- ☐ July
- ☐ August
- ☐ September
- ☐ October

What went well in last four weeks? *

Mention success of something you have done within last four weeks with respect to your project. You can keep doing these things in your project.

Stakeholder participation was necessary for the project's requirements collecting, which produced a well-organized proposal with goals, a problem definition, a suggested solution, and expected outcomes.

What could be improved? *

Mention something you have done, but could be done better.

Add google maps to successfully find the accurate location.

What went badly? *

Mention something you have done badly which you should not continue.

Since these objectives were not effective, we have removed both 'improving delivery efficiency' and 'enhancing operational efficiency' from the project proposal.

For what month you are going to fill your Retrospective *

- ☐ May
- ☐ June
- ☒ July
- ☐ August
- ☐ September
- ☐ October

What went well in last four weeks? *

Mention success of something you have done within last four weeks with respect to your project. You can keep doing these things in your project.

The SRS report for iFood Online Ordering System effectively captured functional and non-functional requirements, ensuring alignment with user needs, restaurant management, and delivery personnel, and was well-structured for stakeholder approval.

What could be improved? *

Mention something you have done, but could be done better.

While the basic interface has been developed, it could benefit from further refinement and user testing to ensure it is simple and user-friendly.

What went badly? *

Mention something you have done badly which you should not continue.

There were delays in implementing feedback from the Software Requirements Specification report review| due to miscommunication between team members.

Project Limitations

- **No Multi-Language Support:** We couldn't implement support for multiple languages in the system.
- **No Multi-Payment Options:** The system currently lacks the capability to handle multiple payment gateways.
- **No Customer Support Messaging:** We were unable to integrate a customer support messaging feature into the system.
- **No Offline Functionality:** The system does not support offline operations, making it entirely dependent on active internet connectivity.



Conclusion

- ★ The system simplifies online food ordering.
- ★ Secure payments ensure safe transactions.
- ★ Real-time tracking improves order transparency.
- ★ Restaurants process orders faster and efficiently.
- ★ Future updates will enhance system features.
- ★ The project was completed on time.
- ★ It provides a competitive edge in the market.
- ★ The platform is reliable and scalable.

Thank You

Any Questions?

