Domain : Service and Capacity Management.

Problem Statement : Restaurant reservation and capacity management for dine-ins'.

Team Name : 21_DineTime-Devs

Team Members :-

S20240010031	Bharat . T
S20240010032	B. Tharun Reddy
S20240010063	D. Naga Varshith
S20240010103	K. Tejdeep Reddy
S20240010153	M. Mahanth Naik

Table Of Contents

Preface

In today's fast-paced digital world, software systems play a vital role in simplifying everyday activities and improving service quality. Restaurants, especially during peak hours, often face challenges such as overcrowding, long waiting times, inefficient table usage, and poor coordination between staff and customers. These issues not only affect customer satisfaction but also reduce operational efficiency. A well-designed software solution can help address these problems by providing structured, reliable, and easy-to-use support for restaurant operations and customer interactions.

This project, **Restaurant Reservation and Capacity Management Platform**, is designed to study and model a real-world problem using fundamental software engineering principles. The focus is on clearly understanding the problem domain, identifying stakeholders, and defining system requirements before moving toward design. By following a systematic and disciplined approach—as emphasized in standard software engineering textbooks—this project highlights the importance of requirement analysis, modelling , and consistency. The proposed system aims to improve dining experiences, optimize restaurant capacity, and demonstrate how well-engineered software can effectively solve practical business problems.

1.Introduction

Restaurants increasingly rely on software systems to manage dine-in reservations, customer flow, and table utilization, especially during peak hours. Manual or fragmented reservation processes often lead to overbooking, inefficient capacity usage, long waiting times, and poor coordination between customers and restaurant staff. From a software engineering perspective, such problems require a well-defined system that is clearly specified, analysed, and modeled before implementation.

This project focuses on the **Restaurant Reservation and Capacity Management Platform**, which supports advance table prebooking, real-time availability tracking, customer check-in, and operational coordination within a restaurant. The system serves multiple stakeholders, including customers, restaurant staff, restaurant managers, and external services such as payment gateways and location services.

The primary objective of this project is to **analyse and model the system requirements** using standard Software Engineering practices. The emphasis is on **requirements engineering and system modelling**, rather than technology selection or implementation. Functional requirements describe the services the system must provide, while non-functional requirements specify quality constraints such as performance, reliability, and security.

To ensure clarity and correctness, the system is represented using **UML diagrams**, including use case, activity, sequence, and class diagrams. These models help in visualizing system behaviour, defining system boundaries, and ensuring that requirements are complete, consistent, and understandable to all stakeholders.

This document follows a structured approach aligned with standard software engineering textbooks and presents the system requirements and UML models for the proposed platform.

2.User Requirements

User requirements describe the services that the system should provide to its users and are expressed in a form that is understandable by stakeholders.

Customer (Diner) Requirements

- The customer shall be able to browse available restaurants.
- The customer shall be able to view real-time table availability.
- The customer shall be able to reserve a dining table for a selected date and time.
- The customer shall be able to modify or cancel an existing reservation.
- The customer shall receive a reservation confirmation after successful booking.
- The customer shall be able to check in at the restaurant using the reservation.

Restaurant Staff Requirements

- Restaurant staff shall be able to view daily reservations.
- Restaurant staff shall be able to update table status (available, reserved, occupied).
- Restaurant staff shall be able to verify customer check-in.
- Restaurant staff shall be able to mark reservations as completed or no-show.

Restaurant Manager Requirements

- The restaurant manager shall be able to manage restaurant profile details.
- The restaurant manager shall be able to configure restaurant operating hours and table capacity.

- The restaurant manager shall be able to monitor reservation and occupancy status.

External System Requirements

- The system shall interact with a payment gateway for reservation payments.
- The system shall integrate with location services for restaurant discovery.
- The system shall support external review and rating services.

3. System Requirements

Functional Requirements

1. User Management

- The system shall allow customers to **register and log in** using valid credentials.
- The system shall allow restaurant managers and staff to **log in with role-based access**.

2. Browse Restaurants

- The system shall allow customers to **browse restaurants** based on location.
- The system shall display **restaurant details** such as name, address, cuisine type, operating hours, and ratings.
- The system shall integrate **GPS location services** to show nearby restaurants.

3. Check Real-Time Availability

- The system shall allow customers to **check real-time table availability** for a selected restaurant.
- The system shall display available **date, time slots, and table capacity**.

4. Table Reservation

- The system shall allow customers to **reserve tables** for a selected date and time.
- The system shall validate reservation requests against **restaurant capacity constraints**.
- The system shall generate a **reservation confirmation** after successful booking.

5. Modify or Cancel Booking

- The system shall allow customers to **cancel a reservation** within the allowed policy time.

6. Customer Check-In Verification

- The system shall support **customer check-in verification** at the restaurant.
- The system shall update reservation status after successful check-in.

7. Restaurant Profile Management

- The system shall allow restaurant managers to **manage restaurant profiles**.
- The system shall allow managers to update **menu, images, and restaurant details**.

8. Configure Restaurant Operations

- The system shall allow restaurant managers to **configure operating hours and time slots**.
- The system shall allow managers to **define table layout and seating capacity**.

9. Table Status Management

- The system shall allow restaurant staff to **update table status** (Available, Reserved, Occupied).

- The system shall reflect table status changes in **real-time availability**.

10. Food Ordering Integration

- The system shall allow customers to **place food orders** after check-in.

11. Payment Processing

- The system shall integrate with a **payment gateway** for reservation.
- The system shall confirm payment status before finalizing the booking.

12. External Services Integration

- The system shall integrate **third-party review and rating services**.
- The system shall display restaurant ratings and customer feedback.

Non-Functional Requirements

1.Performance Requirements

- The system shall respond to **availability search requests within a short period**.
- The system shall support **multiple concurrent reservation requests** without data inconsistency.

2. Availability Requirements

- The system shall be available **anytime**, except during scheduled maintenance.

3. Security Requirements

- The system shall enforce **secure authentication and authorization**.
- The system shall encrypt **user credentials and payment information**.

- Only authorized users shall be allowed to **modify restaurant configurations**.

4. Usability Requirements

- The system shall provide a **user-friendly interface** for customers and staff.
- The system shall be accessible through **web and mobile devices**.
- The booking process shall be completed in **minimal steps**.

5. Reliability Requirements

- The system shall ensure **no double-booking of tables**.
- The system shall maintain **data consistency** during concurrent transactions.

6. Scalability Requirements

- The system shall support **multiple restaurants** simultaneously.
- The system shall handle increased users during **peak hours**.

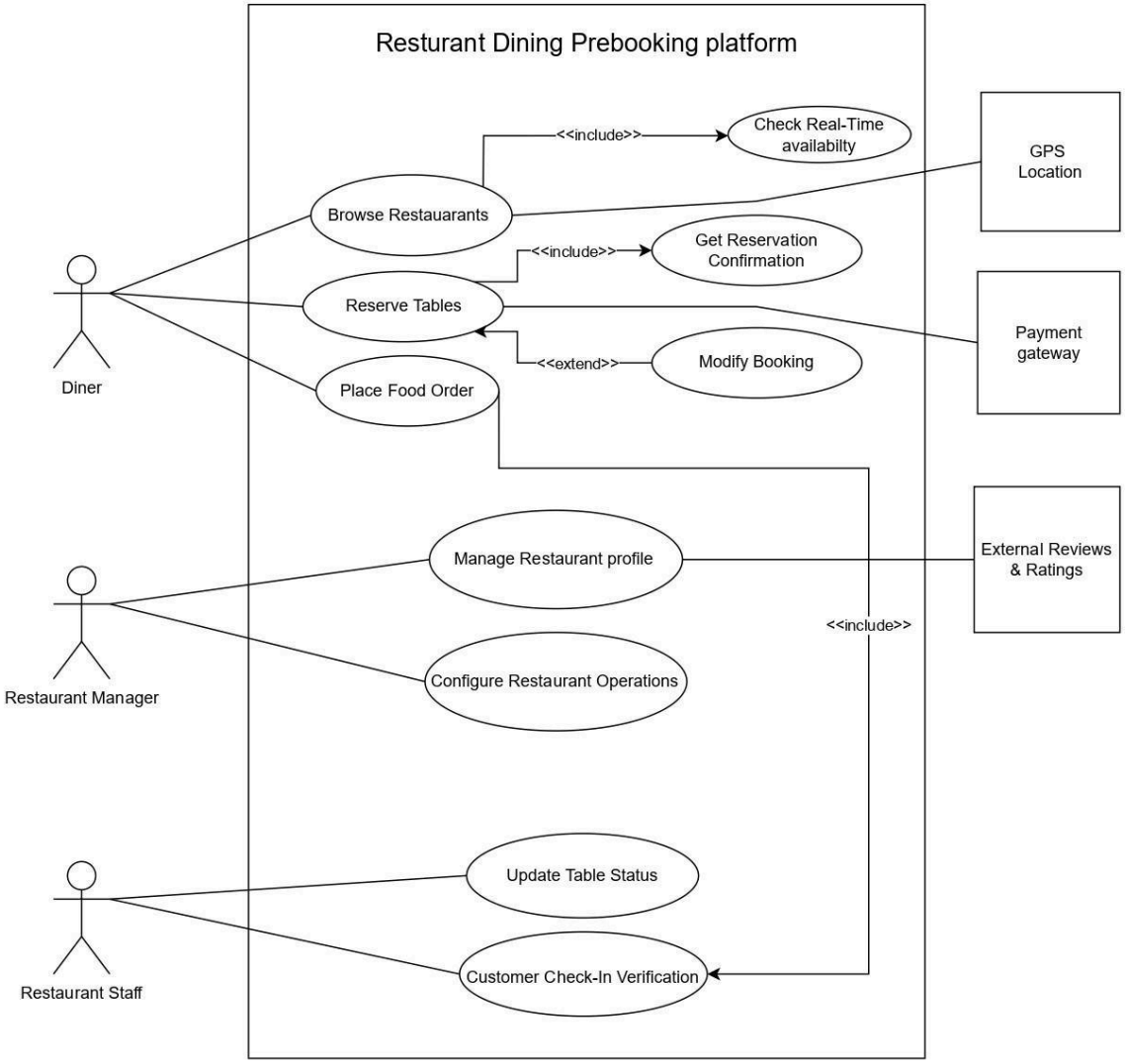
7. Maintainability Requirements

- The system shall be divided into separate **modules**, **each handling a specific function**
- The system shall allow easy updates to **restaurant policies and configurations**.

8. Compatibility Requirements

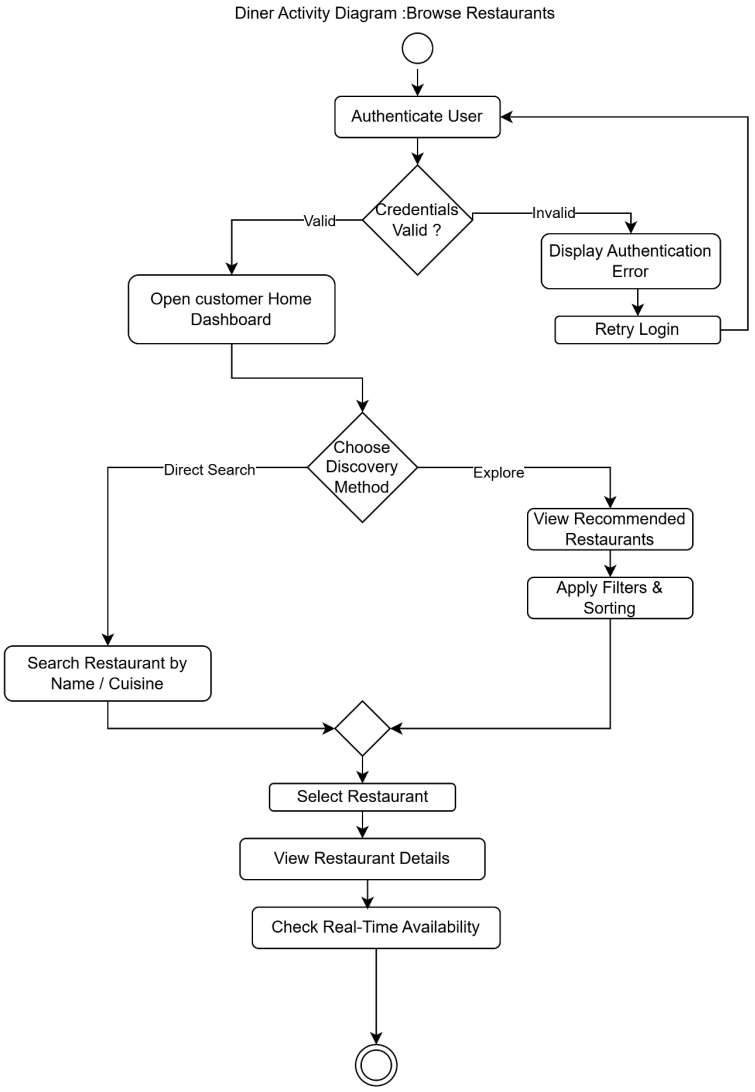
- The system shall integrate with **external payment gateways, GPS services, and kitchen systems**.
- The system shall be compatible with **modern web browsers**.

USE-CASE DIAGRAM

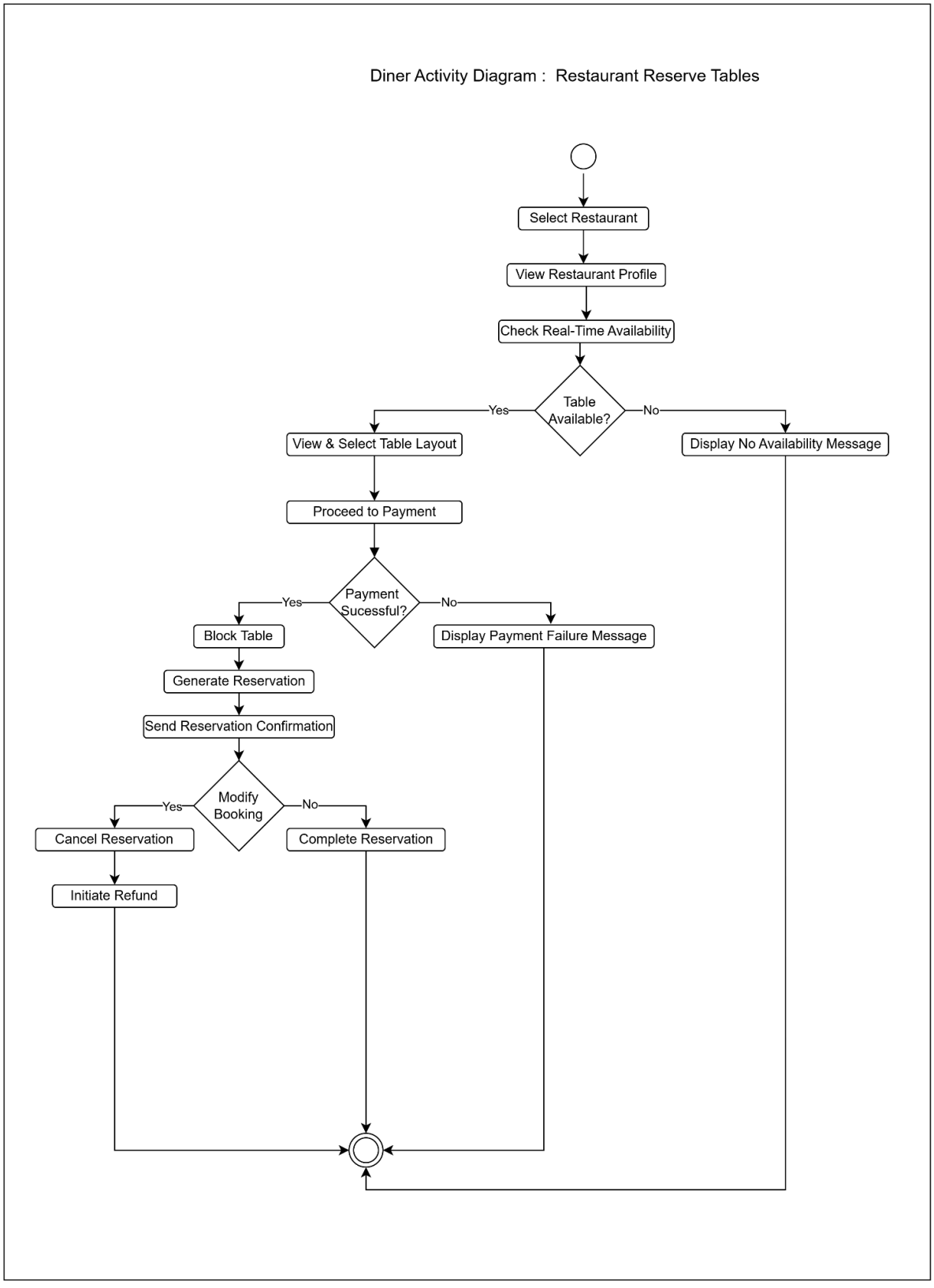


ACTIVITY DIAGRAMS:

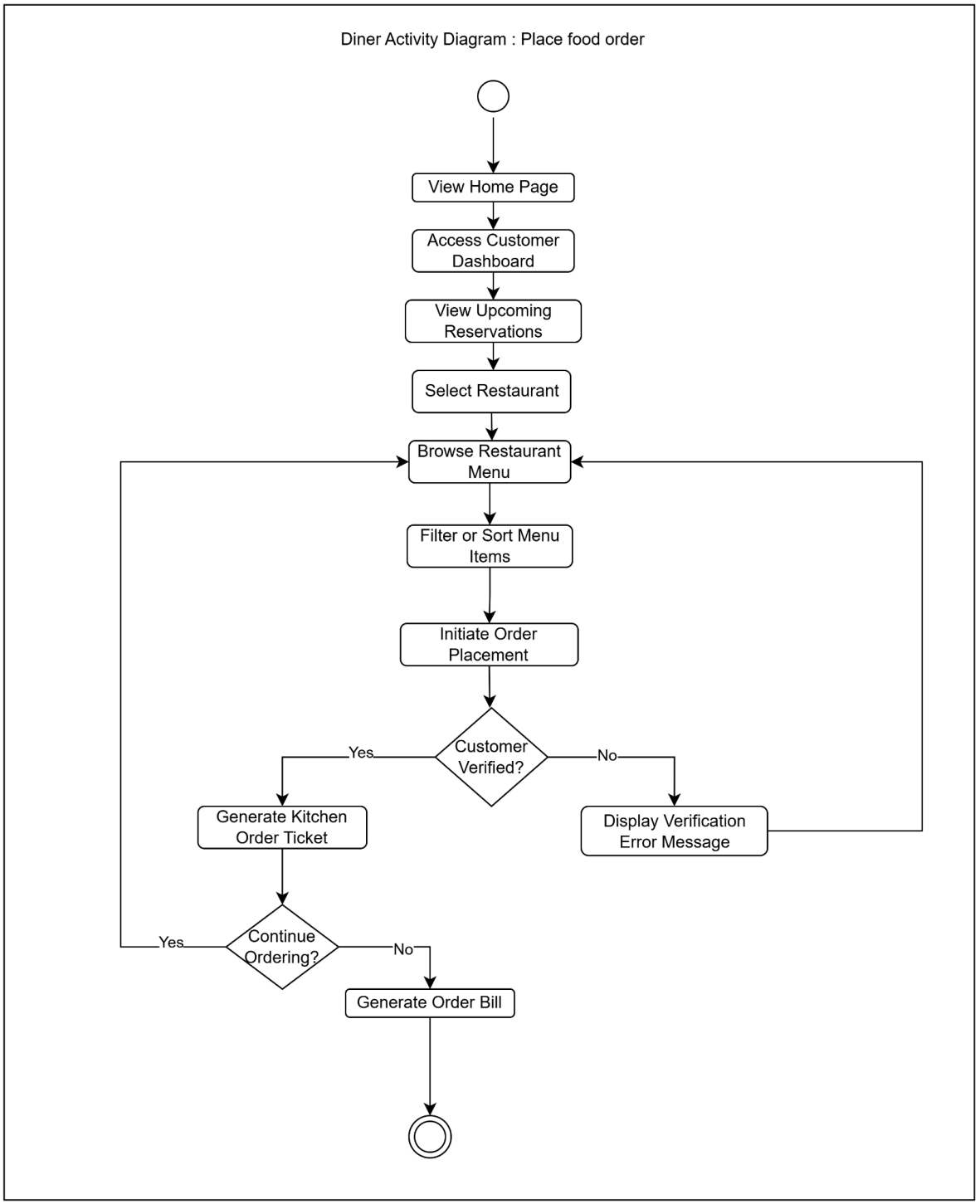
Browse Restaurants



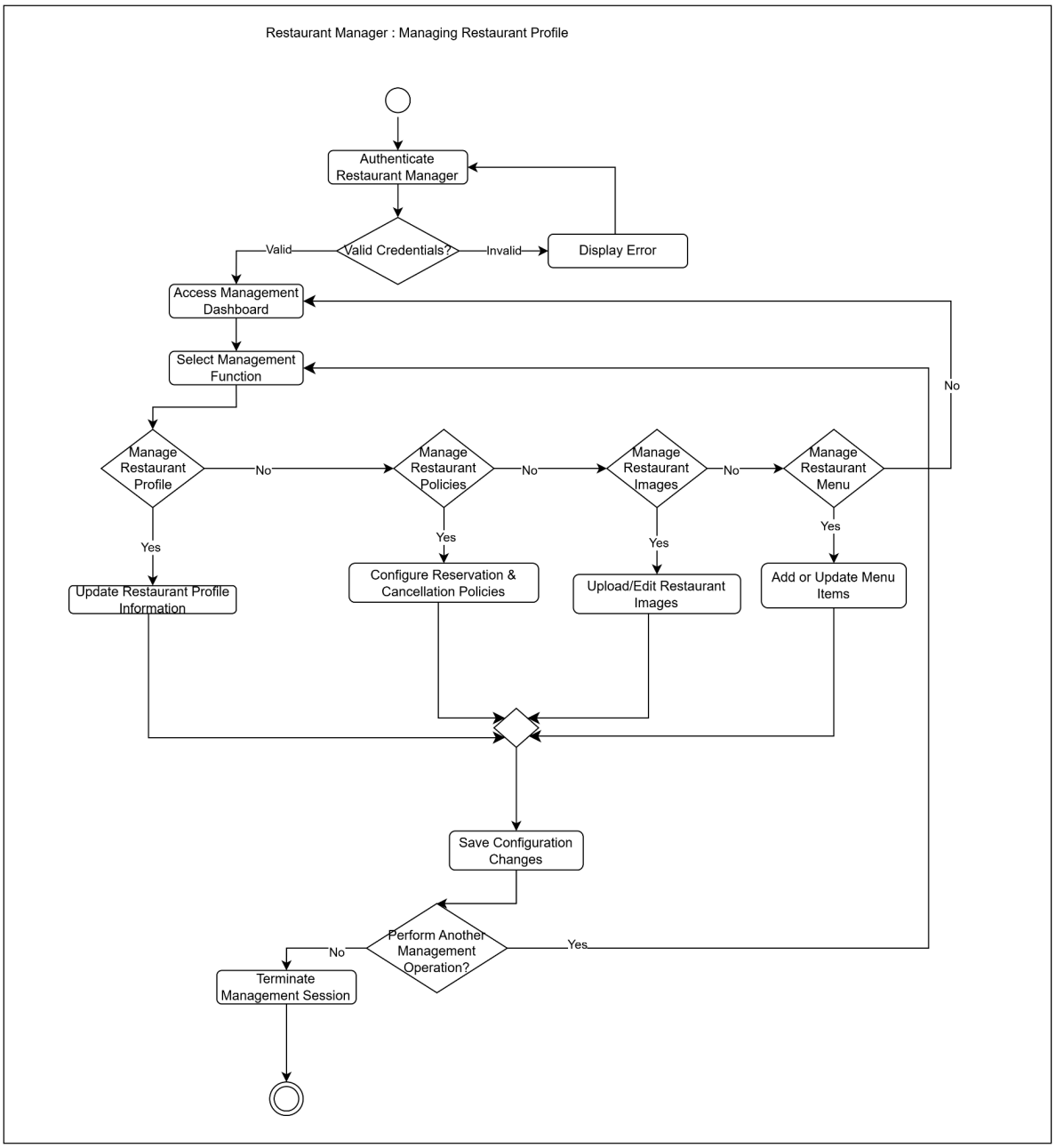
Reserve Tables



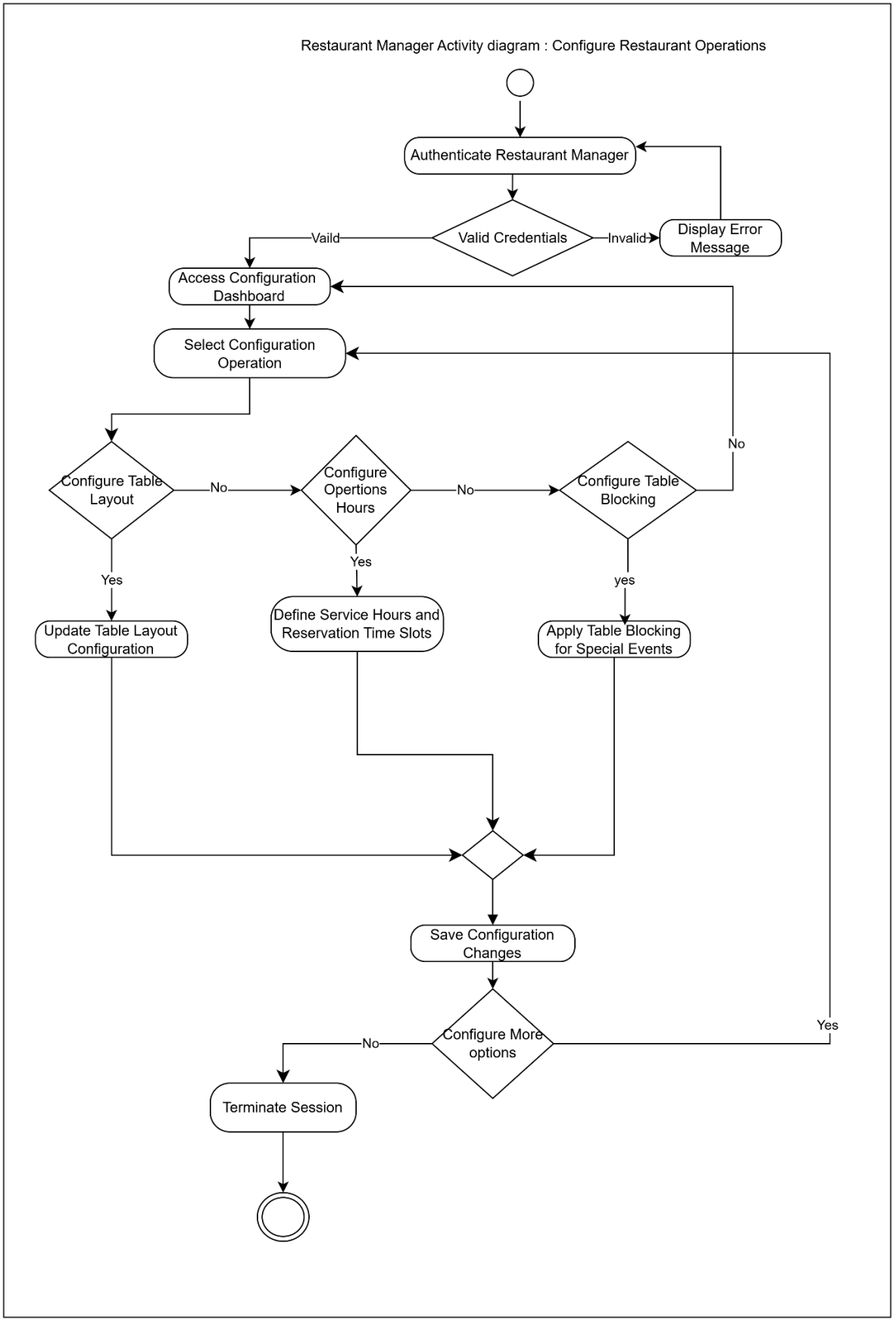
Place Food Order



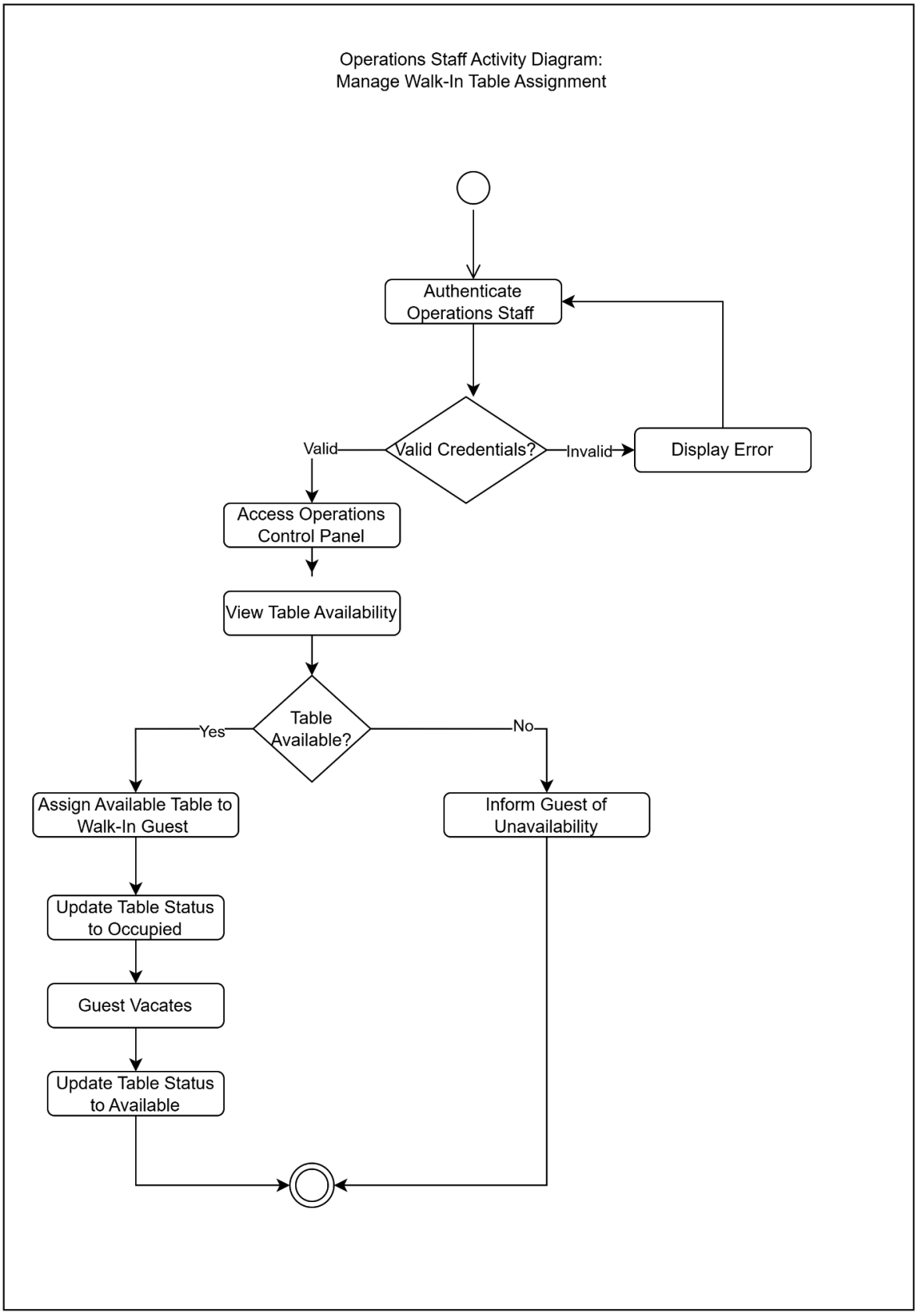
Manage Restaurant Profile



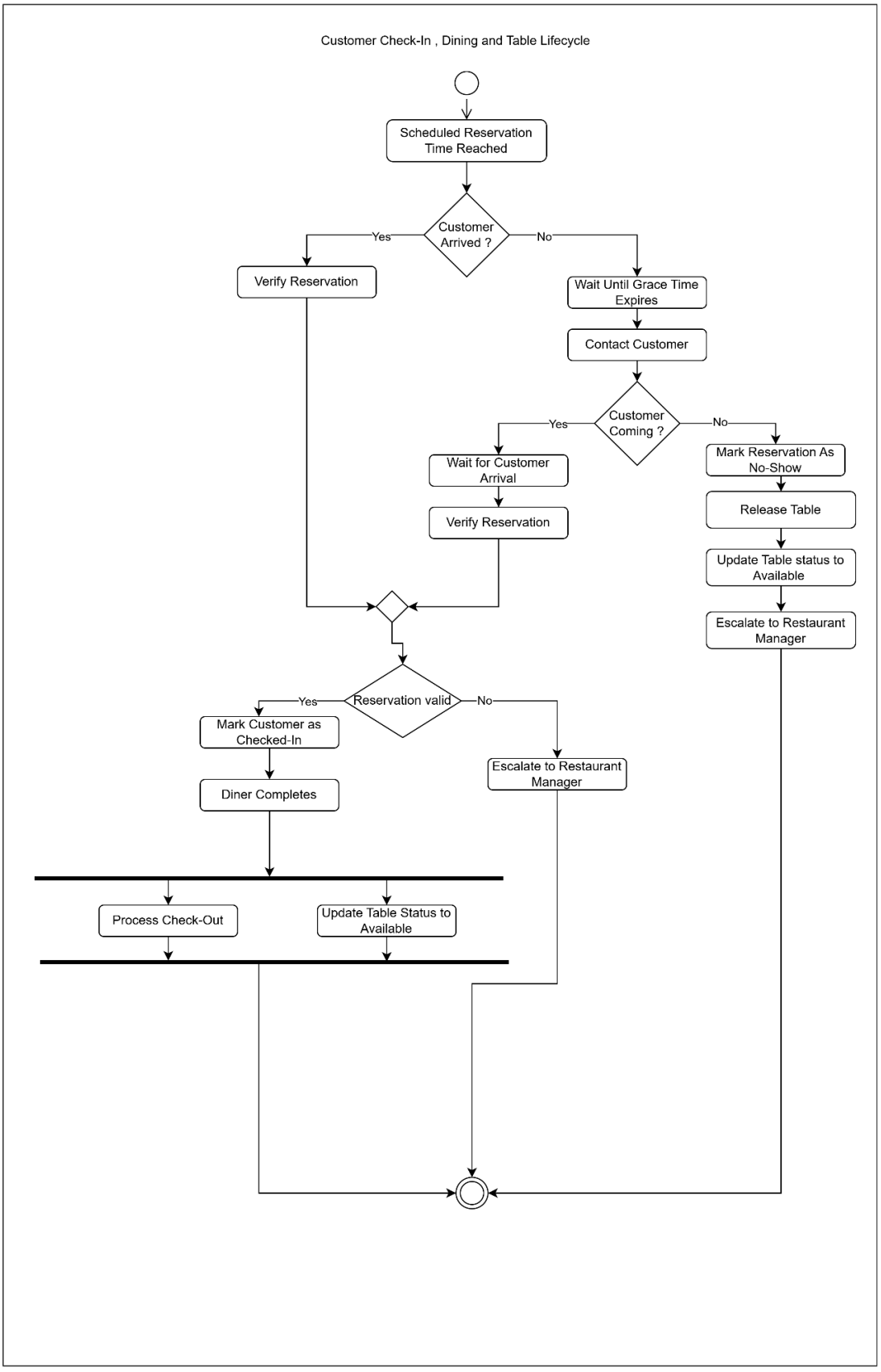
Configure Restaurant Operations



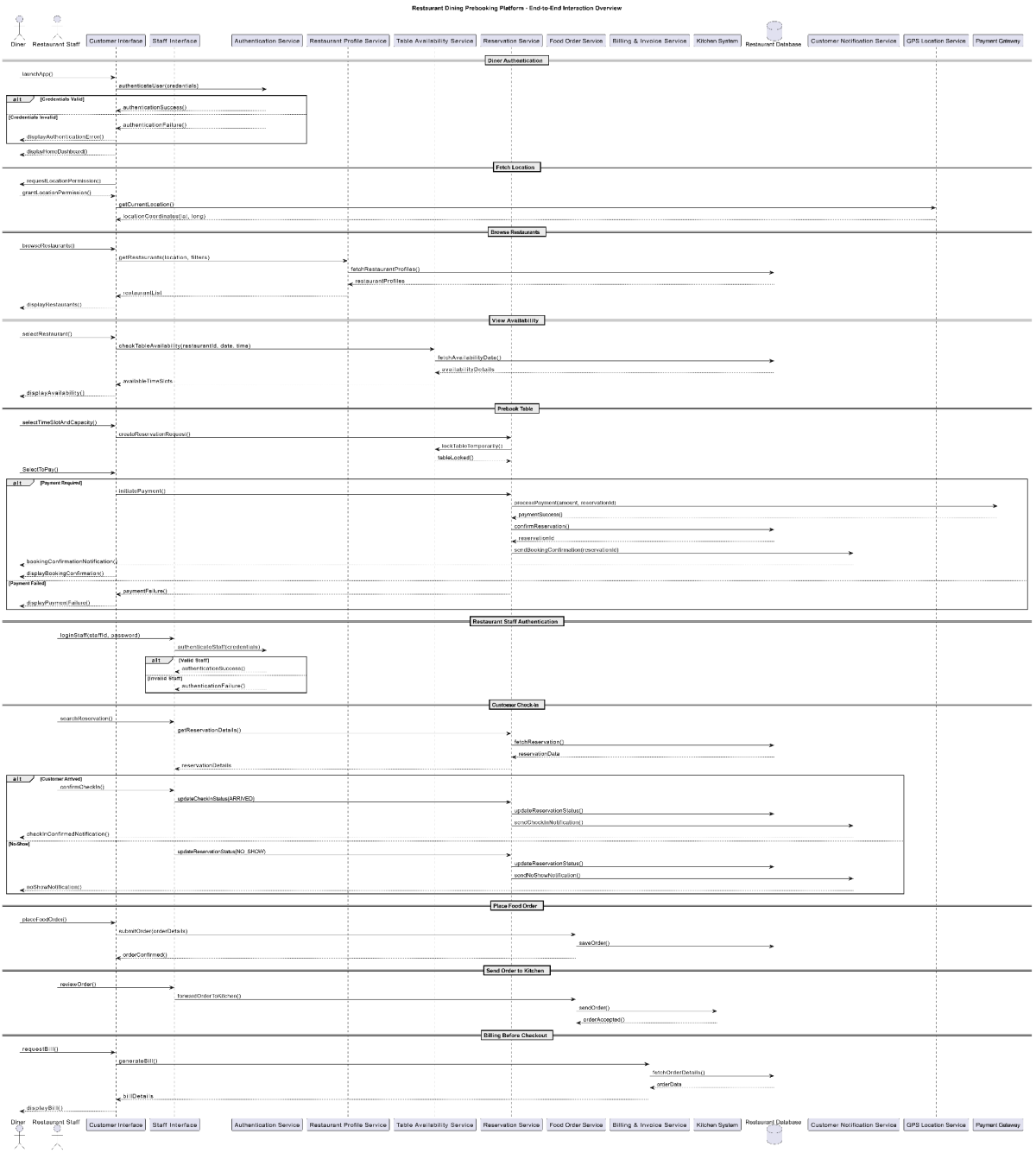
Update Table Status



Customer Track Check-In



Sequence Diagram



Draw.io Links

Use Case Diagram:-
https://drive.google.com/file/d/1HdAyNpKAY2q7Vt25ixUrZCcwxIJT5XGj/view?usp=sharing

Activity Diagrams:-

Browse Restaurants:-

https://drive.google.com/file/d/1Ned93Ol5qJ3mUZlooQBXALAENb3Q8jSm/view?usp=sharing

Reserve Table :-

https://drive.google.com/file/d/1D5GnivZUP-MvAy1-nmFLNh-e7WuKiIEg/view?usp=sharing

Place Order :-

https://drive.google.com/file/d/1cLniABwVp3DSdvBxmvy9F4WU3p1sLsra/view?usp=sharing

Manage Restaurant Profile:-

https://drive.google.com/file/d/1UOX5zPS7-fqfwtdsqlFwD1qXuaXsksJv/view?usp=sharing

Configure Restaurant Operations:-

https://drive.google.com/file/d/15EhsnM3U7p-DAEyfjE26S2WyvTO_oq93/view?usp=sharing

Update Table Status:-

https://drive.google.com/file/d/1s13TR2s7J_b2SYUtGI2IFbxZPi4GSnmF/view?usp=sharing

Check-In Verifications:-

https://drive.google.com/file/d/1K46GrFO7LSJpMg4PkeKzs-mAoolU5dgO/view?usp=sharing

Sequence Diagram:-

https://drive.google.com/file/d/17xIv11a13j6mQwnxKdhjSS0EoKRLg_dX/view?usp=sharing