

Restaurant Management System Requirements Specification

Version 3.0

March 14, 2025

Table of Contents

Contents

1. Executive Summary	3
1.1 Project Overview	3
2. Product/Service Description.....	3
2.1 Product Context	3
2.2 Assumptions.....	4
2.3 Constraints and Dependencies.....	4
3. Requirements	4
3.1 Functional Requirements.....	4
3.2 Non-Functional Requirements	6
3.2.1 Product Requirements	8
4. Use Scenarios/ Use cases.....	10
5. Diagrams	31
5.1 ER Diagram	31
5.2 Use Case Diagram (general).....	33
5.3 Activity Diagram.....	34
5.4 Class Diagram	62
5.5 State Diagram	63
5.6 Sequence Diagram	80
5.7 Collaboration Diagram	103
6. Design Patterns	116

1. Executive Summary

1.1 Project Overview

This Restaurant Management System (RMS) is designed to streamline restaurant operations, including order management, menu updates, billing, and staff coordination. It ensures efficient customer service, reduces wait times, and integrates with payment systems.

2. Product/Service Description

The RMS optimizes restaurant operations by automating order processing, billing, inventory tracking, and staff coordination. It improves efficiency, and lowers errors. The system tracks inventory, facilitates staff coordination, integrates with payment gateways, permits dynamic menu updates, and supports order management for dine-in, takeout, and delivery. The RMS assists restaurants in providing quicker, more effective service by lowering manual labor and enhancing communication.

2.1 Product Context

The RMS will be used by restaurants of various sizes to manage their daily activities. It will be accessible via a web-based platform and mobile application. The system will integrate with third-party delivery services and support multiple payment gateways.

User Characteristics

- Administrator:
 - Role: Manages the entire system, user roles, and settings.
 - Experience: Restaurant operations and management.
 - Technical Expertise: Comfortable with system settings and reports.
 - Other Characteristics: Needs access to analytics and financial data to make decisions.
- Restaurant Manager:
 - Role: Oversees staff, inventory, and sales reports.
 - Experience: Several years in restaurant management.
 - Technical Expertise: Skilled with reporting tools and payment systems.
 - Other Characteristics: Requires effective dashboards for updates in real time.
- Waiter/Server:
 - Role: Takes orders and interacts with the kitchen.
 - Experience: From beginner to experienced service staff.
 - Technical Expertise: Should require minimal training.
 - Other Characteristics: Needs simple, user-friendly interface for quick order processing.
- Chef/Kitchen Staff:
 - Role: Receives and prepares orders.
 - Experience: Professional chefs and kitchen assistants.
 - Technical Expertise: Minimal interaction with the system.
 - Other Characteristics: Needs a simple, effective method for receiving and tracking orders.
- Inventory Manager:
 - Role: Monitors stock levels, places restocking orders, and ensures inventory availability.
 - Experience: From beginner to experienced service staff.
 - Technical Expertise: Should require minimal training.
 - Other Characteristics: Needs real-time inventory tracking, low-stock alerts, and an easy-to-use interface for updating stock levels.

Restaurant Management System Requirements Specification

- Customer:
 - Role: Places online orders, books tables, and makes payments.
 - Technical Expertise: All users should be able to easily navigate and understand the system.
 - Other Characteristics: Gives priority to quick checkout, secure payment methods, and ease of use.

2.2 Assumptions

- Both desktop and mobile devices with internet access will be able to access the RMS.
- Users will know the fundamentals of using common restaurant management tools and digital systems.
- Third-party delivery services and payment gateways will be integrated with the system.
- For smooth operation, the restaurant will have reliable internet access.
- Common web browsers (Chrome, Firefox, and Safari) will be supported by the RMS.
- The necessary hardware, such as computers, tablets, or smartphones, will be available to users.
- Restaurants of all sizes, from tiny to massive operations, will be able to use the system.

2.3 Constraints and Dependencies

- System Integration: The system needs to work with the restaurant's existing payment system to ensure smooth transactions.
- User Security and Access: Some features, like inventory management and payment processing, should only be accessible to authorized users.
- Data Backup and Logging: Every day, the system has to automatically backup customer, inventory, and sales data. Logs must also be kept for auditing purposes.
- In order for order processing to work properly, inventory management must be set up.
- In order to avoid slowdowns during peak hours, the system must be able to manage busy hours effectively.

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_RMS_01	The system must allow users to log in / log out securely.	Entry point for all other functions.	1	3/14/25	
BR_RMS_02	The system must handle table reservations (create, view).	Core booking workflow.	1	3/14/25	
BR_RMS_03	The system must support bill generation and multi-method payment .	Accurate checkout is critical.	1	3/14/25	
BR_RMS_04	Inventory staff must be able to increase ingredient stock on delivery.	Reflects restocking.	1	3/14/25	
BR_RMS_05	Inventory staff must be able to decrease ingredient stock (usage / waste).	Keeps counts in sync.	1	3/14/25	

Restaurant Management System Requirements Specification

BR_RMS_06	Waiters must be able to place customer orders .	Order capture.	1	3/14/25	
BR_RMS_07	Admins must be able to add, update, remove menu items .	Dynamic menu management.	1	3/14/25	
BR_RMS_08	The system must enforce role-based access & MFA for sensitive roles.	Security baseline.	1	3/14/25	
BR_RMS_09	Managers must be able to view sales & revenue reports .	Business KPI tracking.	1	3/14/25	
BR_RMS_10	Customers must be able to cancel reservations online .	Reduces no-shows.	2	3/14/25	
BR_RMS_11	Kitchen staff must update order status (Preparing → Ready → Served).	Keeps FOH & customers informed.	1	3/14/25	
BR_RMS_12	Customers can submit feedback and ratings after dining.	Service-quality input.	3	3/14/25	
BR_RMS_13	Managers must be able to create & adjust shift schedules .	Optimizes staffing.	2	3/14/25	
BR_RMS_14	The system must track waste events & ingredient expiry .	Compliance & cost control.	2	3/14/25	
BR_RMS_15	Customers can view order history and reorder favorites.	Convenience & upsell.	3	3/21/25	
BR_RMS_16	Admins must support employee role assignment / changes .	Keeps permissions current.	1	3/21/25	
BR_RMS_17	Inventory manager must monitor supplier performance KPIs .	Data-driven purchasing.	2	3/21/25	
BR_RMS_18	Waiters must declare tips ; system auto-distributes per policy.	Transparency & fairness.	2	3/21/25	
BR_RMS_19	Customers must be able to book special events with custom options.	High-value reservations.	2	3/21/25	
BR_RMS_20	Customers can request valet parking ; valet staff manage vehicles.	Premium service.	3	3/21/25	
BR_RMS_21	The system must track utility consumption (water, gas, electricity).	Cost & sustainability metrics.	2	3/21/25	
BR_RMS_22	Managers must define & track a cleaning-task schedule .	Health-code compliance.	2	3/21/25	

Restaurant Management System Requirements Specification

BR_RMS_23	All security-camera footage accesses must be logged & traceable.	Security & audit.	1	3/21/25	
BR_RMS_24	Managers must be able to schedule employee training sessions.	Staff development.	2	3/21/25	
BR_RMS_25	Staff must report tableware damage for replacement tracking.	Controls breakage costs.	3	3/21/25	
BR_RMS_26	The system must monitor temperature zones and alert on breach.	Food-safety critical.	1	3/21/25	
BR_RMS_27	The system must schedule & log dish-washer maintenance.	Equipment uptime.	2	3/21/25	
BR_RMS_28	Staff must register children in the play area with guardian consent.	Safety & liability.	3	3/21/25	
BR_RMS_29	Customers must acknowledge food-allergy warnings before ordering.	Legal requirement.	1	3/21/25	
BR_RMS_30	The system must warn as dining-time limits approach and allow override.	Table-turn efficiency.	2	3/21/25	

3.2 Non-Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_RMS_01	The system must have an intuitive UI for both staff and customers.	Ensures ease of use and better user experience.	1	3/14/25	
BR_RMS_02	The system should support multiple languages.	Increases accessibility for diverse customer base.	2	3/14/25	
BR_RMS_03	The system must handle up to 100 concurrent users.	Supports scalability for high-traffic periods.	1	3/14/25	
BR_RMS_04	The system should process orders within 2 seconds per transaction.	Ensures fast and responsive user interactions.	2	3/14/25	

Restaurant Management System Requirements Specification

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_RMS_05	The system must be available 99.9% of the time.	Ensures high availability and minimal downtime.	1	3/14/25	
BR_RMS_06	Scheduled maintenance should be notified in advance.	Helps users plan around downtime and improves transparency.	2	3/14/25	
BR_RMS_07	Customer and payment data must be encrypted.	Protects sensitive customer and payment information.	1	3/14/25	
BR_RMS_08	The system should maintain access logs for security audits.	Enables auditing and ensures system integrity.	2	3/14/25	
BR_RMS_09	The system must comply with restaurant industry standards.	Ensures industry-standard practices for operation.	1	3/14/25	
BR_RMS_10	Staff must undergo training before using the system.	Ensures proper usage and reduces errors in system operation.	1	3/14/25	
BR_RMS_11	The payment gateway must comply with PCI-DSS security standards.	Ensures secure payment processing and compliance.	1	3/14/25	
BR_RMS_12	The system should integrate with third-party food delivery services.	Expands order fulfillment options and reaches more customers.	2	3/14/25	
BR_RMS_13	The system should use 256-bit encryption for storing sensitive data like passwords and payments.	Data encryption: Ensures secure storage of sensitive information.	1	3/19/25	
BR_RMS_14	The system should allow easy addition of new restaurant branches without reconfiguration.	Branch expansion: Simplifies scaling without extra setup.	2	3/19/25	
BR_RMS_15	The system should provide training and onboarding materials for employees.	Employee training: Helps staff quickly adapt to the system.	3	3/19/25	
BR_RMS_16	The system should be compatible with government tax regulations, automatically calculating VAT or service charges.	Tax compliance: Automates VAT and service charge calculations.	1	3/19/25	
BR_RMS_17	The system should allow cross-platform access, working on Windows, macOS, and Linux.	Cross-platform access: Ensures usability on multiple operating systems.	2	3/19/25	
BR_RMS_18	The system should allow quick restoration from backup within 10 minutes.	Quick backup restoration: Minimizes downtime and data loss.	1	3/19/25	

Restaurant Management System Requirements Specification

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_RMS_19	After a system failure, the chef should be able to see orders that went through before the crash.	Order recovery: Prevents lost orders after system crashes.	1	3/19/25	
BR_RMS_20	The system should include search and filter options to help users find menu items, orders, and reports quickly.	Search and filter: Improves efficiency in finding key data.	2	3/19/25	

3.2.1 Product Requirements

3.2.1.1 Usability Requirements

Include any specific usability requirements, for example,

- Both employees and clients should find the system's user interface (UI) easy to use.
- New staff should be able to learn the system within a day.
- The system should support multiple languages.

3.2.1.2 Performance Requirements

- Each transaction should take no more than two seconds to place and process an order.
- The system should handle up to 100 users at once.
- The system should support a database of at least 10,000 menu items.

3.2.1.3 Availability

- The system should be available 99.9% of the time.
- The system shall provide availability to all users across regions.
- Scheduled maintenance shall be communicated to users at least 48 hours in advance.
- The system shall maintain a Mean Time Between Failures (MTBF) of at least 1,000 hours. The maximum permitted number of failures shall be no more than 2 failures per 1,000 operational hours.

3.2.1.4 Security

- All sensitive data, including customer payment information, personal details, and order history, must be encrypted.
- All user interactions, system access, data modifications, and transaction history must be documented in the system's extensive activity logs.
- Order management, payment processing, user authentication, and other system module communications must be encrypted and closely watched for security purposes.
- All transactions and data modifications will automatically undergo data integrity checks by the system. The system administrator will receive an alert if any unauthorized changes or data inconsistencies are found.

3.2.2 Organizational Requirements

- Staff should undergo training on how to use the system.
- The system must follow all applicable restaurant industry standards and laws, such as data protection, payment processing, and health and safety guidelines.
- The system must implement and follow the organization's backup and disaster recovery procedures.

3.2.3 External Requirements

- The payment system must follow security standards to protect payment information.
- The system must be able to connect with third-party delivery platforms for order fulfillment.
- The system must protect customer data and ensure privacy.

4. Use Scenarios/ Use cases

UC_1 – Gloria Traja	User Login
Summary	Allows users to log into the system with appropriate access based on their role.
Dependency	None
Actors	Primary Actor: Registered User (Admin, Inventory Manager, Customer)
Preconditions	The user must be registered in the system and have valid credentials.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. User navigates to the login page. 2. User enters a valid username and password. 3. System verifies the credentials. 4. System grants access and redirects to the dashboard.
Description of the Alternative Sequence	1a. If credentials are invalid, the system displays an error message and prompts the user to retry.
Non-Functional Requirements	<ul style="list-style-type: none"> - Security – Encrypted password storage, role-based access control, and login attempt monitoring - Performance – Login authentication must be completed within 2 seconds - Availability – System must always be available for registered users - Interface – Clean, responsive, and user-friendly login page - Access Control – Users are only shown modules and functionalities permitted by their role: <ul style="list-style-type: none"> • Admin: Full system access including user management and settings • Inventory Manager: Access to stock management and restocking modules • Customer: Access to reservation, order, and profile modules only
Postconditions	User is logged into the system and can access authorized functionalities.

Restaurant Management System Requirements Specification

UC_2 – Gloria Traja	Table Reservation
Summary	Allows customers to reserve a table online.
Dependency	None
Actors	Primary Actor: Customer
Preconditions	The system must be operational.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Customer navigates to the "Reserve Table" page. 2. Customer selects date, time, and number of guests. 3. System checks table availability. 4. Customer confirms the reservation. 5. System saves the reservation and sends confirmation.
Description of the Alternative Sequence	3a. If no table is available, the system suggests alternative time slots.
Non-Functional Requirements	<ul style="list-style-type: none"> - Performance – Reservation should be processed within 2 seconds - Scalability – Supports multiple simultaneous reservations - Notifications – Sends confirmation and reminders - Interface – User-friendly reservation process
Postconditions	Table is successfully reserved and confirmation is sent.

UC_3 – Gloria Traja	Bill Generation & Payment Processing
Summary	Allows the system to generate a bill and process payments.
Dependency	An order must be completed.
Actors	Primary Actor: Customer, Cashier
Preconditions	The customer must have completed an order.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. System generates a bill based on the items ordered. 2. Customer reviews the bill. 3. Customer selects a payment method (Cash, Credit, Digital Payment). 4. System processes the payment. 5. System confirms the transaction and generates a receipt.
Description of the Alternative Sequence	4a. If payment fails, the system prompts the customer to retry or select a different method.
Non-Functional Requirements	- Performance – Payment should process within 3 seconds

Restaurant Management System Requirements Specification

	<ul style="list-style-type: none"> - Security– Ensures safe transactions and encrypted data - Scalability– Handles multiple transactions simultaneously - Interface– Clear and easy payment flow
Postconditions	<i>Payment is completed, and a receipt is issued.</i>

UC_4 – Gloria Traja	Ingredient Stock Management Increase
Summary	<i>Allows the system to increase ingredient stock levels when new inventory is received.</i>
Dependency	None
Actors	<i>Primary Actor: Inventory Manager, Admin</i>
Preconditions	<i>The system must be operational, and the user must be logged in with appropriate privileges.</i>
Main Sequence	<ol style="list-style-type: none"> 1. <i>Inventory manager logs into the system.</i> 2. <i>System displays current stock levels.</i> 3. <i>Inventory manager selects the ingredient to update</i> 4. <i>Inventory manager inputs the quantity of new stock received.</i> 5. <i>System updates the stock level accordingly.</i> 6. <i>System logs the update and confirms the new stock level.</i>
Alternative Sequence	<p>4a. <i>If input quantity is invalid (e.g., negative or non-numeric), the system displays an error and requests correction.</i></p> <p>6a. <i>If the update fails due to system error, the system notifies the admin for manual intervention.</i></p>
Non-Functional Requirements	<ul style="list-style-type: none"> - Real-Time Updates – Stock increase must reflect instantly. - Validation – System must validate the input quantity before processing. - Scalability – Should support bulk updates for multiple ingredients. - Interface – Must provide a user-friendly interface for entering stock quantities.
Postconditions	<i>Stock levels are successfully updated and recorded in the system log.</i>

Restaurant Management System Requirements Specification

UC_6 – Alisa Tozaj	Place Order
Summary	Allows customers to place orders, categorize as dine-in, takeout, or delivery, with real time kitchen notifications and order tracking.
Dependency	The system must have a functional menu management system and a real-time kitchen notification system.
Actors	Customer, Chef, Waiter
Preconditions	Customer is logged into the system; menu is available; kitchen is online and receiving notifications.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Customer browses the menu. 2. Customer selects items and places the order. 3. The system categorizes the order (dine-in, takeout, or delivery). 4. The kitchen receives the order in real-time. 5. Chef updates the order status (e.g., preparing, ready). 6. Customer tracks the status of the order.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 1. If the menu is unavailable, the customer cannot place an order. 2. If the kitchen system fails, the chef cannot update the order status.
Non functional requirements	<ul style="list-style-type: none"> - Responsiveness – Real-time order updates - Availability – System must function smoothly during peak hours - Scalability – Must handle high order volume efficiently - Interface – Simple UI for customers, waiters, and kitchen staff
Postconditions	The order is placed, categorized, and notified to the kitchen. The customer can track the order status.

UC_7 – Alisa Tozaj	Menu Management
Summary	Allows admins to add, update, or remove menu items and upload multimedia content, with scheduled updates.
Dependency	Admin must have access to the system; menu items and images should be uploaded properly.
Actors	Administrator

Restaurant Management System Requirements Specification

Preconditions	<i>Admin is logged into the system; menu data is available for updates.</i>
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Admin logs into the system. 2. Admin selects the option to add, update, or remove menu items. 3. Admin uploads images or videos for the items (if applicable). 4. Admin saves changes and the system updates the menu. 5. Menu is updated in real-time for customers.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 1. If an image fails to upload, the system will notify the admin with an error message. 2. If no changes are made, the admin can exit without saving.
Non functional requirements	<ul style="list-style-type: none"> - Navigation – Intuitive for admins to update menus - Multimedia Support – Supports images/videos for menu items - Scalability – Handles large menus without performance issues Scheduled Updates – Allows updates without disrupting service
Postconditions	<i>The menu is updated in the system and is available for customers to view.</i>

UC_8 – Alisa Tozaj	User Access & Security
Summary	<i>The system ensures role-based access and protects sensitive data through secure login and multi-factor authentication for admins.</i>
Dependency	<i>The system must have secure authentication mechanisms and role-based access control.</i>
Actors	<i>Administrator, Restaurant Manager, Waiter, Inventory Manager</i>
Preconditions	<ol style="list-style-type: none"> 1. User attempts to log into the system. 2. The system verifies user credentials (username/password). 3. For admins, the system prompts for multi-factor authentication.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The system checks the user's role (Admin, Restaurant Manager, Waiter/Server, Inventory Manager). 2. The system grants access based on user role.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 1. If credentials are incorrect, the system denies access and prompts for re-entry. 2. If multi-factor authentication fails, the system denies access and notifies the user.

Restaurant Management System Requirements Specification

Non functional requirements	<ul style="list-style-type: none"> - Security – Encrypted credentials, role-based access control - Multi-Factor Authentication – Required for admins - Availability – Always accessible for authorized users - Interface – Simple and secure login process
Postconditions	<i>The user is granted access to the system based on their role.</i>

UC_9 – Alisa Tozaj		Sales & Reports
Summary	<i>The system generates daily sales reports and provides insights on customer preferences.</i>	
Dependency	<i>The system must have a functioning sales tracking mechanism.</i>	
Actors	<i>Administrator, Restaurant Manager</i>	
Preconditions	<i>Sales data is available; the system is operational.</i>	
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Admin or restaurant manager logs into the system. 2. They navigate to the reports section. 3. The system generates a daily sales report. 4. The system provides insights on customer preferences based on sales data. 	
Description of the Alternative Sequence	<ol style="list-style-type: none"> 1. If there is an issue generating the report (e.g., system failure), the system notifies the user. 2. If no sales data is available for the day, the system displays a "No data" message. 	
Non functional requirements	<ul style="list-style-type: none"> - Performance – Reports should generate within seconds - Scalability – Supports large datasets as business grows - Data Insights – Provides meaningful analytics - Interface – Clear dashboard for restaurant managers 	
Postconditions	<i>A daily sales report is generated, and insights on customer preferences are provided.</i>	

Restaurant Management System Requirements Specification

UC_10 – Jodi Xure		Table Cancellation
Summary		Allows customers to cancel an existing table reservation online.
Dependency		Table Reservation must exist.
Actors		Primary Actor: Customer
Preconditions		<ul style="list-style-type: none"> - The customer must have an existing reservation. - The system must be operational.
Description of the Main Sequence		<ol style="list-style-type: none"> 1. Customer navigates to the "My Reservations" or "Cancel Reservation" page. 2. Customer selects the reservation to cancel. 3. System prompts for cancellation confirmation. 4. Customer confirms the cancellation. 5. System cancels the reservation and sends a cancellation confirmation.
Description of the Alternative Sequence		<ol style="list-style-type: none"> 2a. If the reservation is not found or already cancelled, the system notifies the customer. 4a. If the customer cancels the cancellation process, the system returns to the reservation page.
Non-Functional Requirements		<ul style="list-style-type: none"> - Performance: Cancellation should be processed within 2 seconds. - Security: Only the reservation holder can cancel. - Notifications: Sends cancellation confirmation instantly. - Interface: Clear and user-friendly cancellation process.
Postconditions		<ul style="list-style-type: none"> - The reservation is successfully cancelled. - Cancellation confirmation is sent to the customer.

UC_11 – Joldi Xure		Update Order Status
Summary		Allows kitchen staff to update the status of a customer's food order.
Dependency		Order must already be placed.
Actors		Primary Actor: Kitchen Staff
Preconditions		<ul style="list-style-type: none"> - Order has been placed and is pending in the kitchen system. - System must be operational.
Main Sequence		<ol style="list-style-type: none"> 1. Kitchen staff logs into the kitchen interface. 2. Staff views the list of active/pending orders. 3. Staff selects an order. 4. Staff updates the order status (e.g., "In Preparation", "Ready for Pickup").

Restaurant Management System Requirements Specification

	5. System saves the updated status and notifies relevant parties (e.g., waitstaff or customer).
Alternative Sequence	3a. If the order is not found, system notifies the staff. 4a. If status update fails due to system error, a retry prompt is displayed.
Non-Functional Requirements	<ul style="list-style-type: none"> - Performance: Status update should reflect system-wide within 1 second. - Reliability: Must handle simultaneous status updates from multiple staff. - Interface: Simple, quick-access UI optimized for a kitchen environment. - Notifications: Sends real-time updates to waitstaff/customer interface.
Postconditions	<ul style="list-style-type: none"> - Order status is updated and saved. - Notification is sent to the relevant stakeholders.

UC_12 - Joldi Xure	Customer Feedback & Ratings
Summary	Allows customers to submit feedback and rate their experience (food, service, ambiance) after dining or placing an order.
Dependency	A completed order or reservation must exist.
Actors	Primary Actor: Customer
Preconditions	<ul style="list-style-type: none"> - Customer has completed an order or dined in. - System must be operational. - Feedback form must be available.
Main Sequence	<ol style="list-style-type: none"> 1. Customer logs into their account or uses a feedback link. 2. Customer navigates to the “Feedback & Ratings” page. 3. System displays a list of recent completed orders/reservations. 4. Customer selects the visit/order to review. 5. Customer rates experience (e.g., 1–5 stars) and optionally leaves written feedback. 6. Customer submits the feedback. 7. System saves the feedback and optionally sends a thank-you message.

Restaurant Management System Requirements Specification

Alternative Sequence	<p>4a. If no completed orders are found, system displays a message: "No recent visits to review."</p> <p>6a. If submission fails, system shows an error and prompts to retry.</p>
Non-Functional Requirements	<ul style="list-style-type: none"> - Usability: Easy-to-use interface for quick ratings and comments. - Performance: Feedback should be submitted within 2 seconds. - Analytics: Feedback data should be stored and accessible for reporting. - Interface: Mobile-friendly form layout.
Postconditions	<ul style="list-style-type: none"> - Feedback and rating are saved to the database. - Admin and/or manager can view aggregated feedback reports. - Optionally, a thank-you email or offer is sent to the customer.

Shift Scheduling for Staff	
UC Name 13 – Jodi Xure	
Summary	Enables managers to create, assign, modify, and view staff work schedules, ensuring optimal shift coverage.
Dependency	Staff members must be registered in the system.
Actors	Primary Actor: Restaurant Manager Supporting Actor: Staff
Preconditions	<ul style="list-style-type: none"> - Restaurant Manager is logged into the system. - Staff profiles exist and are up to date. - System calendar is synchronized.
Main Sequence	1. Manager navigates to the "Shift Scheduling" module. 2. Manager selects a date range or specific day. 3. Manager assigns staff members to time slots based on roles (e.g., waiter, chef, cleaner). 4. Manager saves the schedule. 5. System notifies assigned staff of their shifts.
Alternative Sequence	3a. If a staff member is unavailable (e.g., time conflict or time-off request), the system alerts the manager and suggests alternatives. 5a. If notifications fail, the system logs the issue and allows for manual communication.
Non-Functional Requirements	<ul style="list-style-type: none"> - Performance: Scheduling actions should complete within 3 seconds. - Scalability: Supports scheduling across multiple branches or large teams. - Notifications: Staff receive instant notifications via app, email, or SMS. - Interface: Calendar-based drag-and-drop UI for efficient shift planning.

Restaurant Management System Requirements Specification

Postconditions	- Shift schedule is saved and visible to both managers and staff.- Staff are notified and can view or request changes if necessary.
-----------------------	---

UC_14 – Ester Qershori	<i>Waste & Expiry Tracking</i>
Summary	<i>Tracks ingredient usage, expiration dates, and waste events to ensure food quality and reduce unnecessary loss.</i>
Dependency	<i>Ingredient Stock Management module must be active and up to date.</i>
Actors	<i>Primary Actor: Inventory Manager Supporting Actor: Kitchen Staff</i>
Preconditions	<ul style="list-style-type: none"> - Ingredients are registered in the inventory with expiry dates. - System is operational and synced with stock updates.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. System automatically monitors ingredient expiry dates. 2. Inventory Manager views upcoming expiry reports. 3. Kitchen staff logs waste events (e.g., spoiled or overcooked items). 4. System deducts wasted items from stock. 5. Inventory Manager reviews waste and expiry trends in the dashboard.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 2a. If expired items are detected, the system triggers an alert to the Inventory Manager. 3a. If waste entry fails (e.g., connectivity issue), the system stores locally and syncs later.
Non functional requirements	<ul style="list-style-type: none"> - Real-Time Monitoring - Automatic checks on expiry status every hour. - Alerts - Sends expiry/waste alerts via email or app notifications. - Analytics Generates monthly reports on waste trends and loss estimates. - Interface - Intuitive logging system for quick kitchen entries.
Postconditions	<ul style="list-style-type: none"> - Expired or wasted ingredients are removed from available stock. - Historical data is stored for reporting and future planning. - Inventory accuracy and food safety are maintained.

Restaurant Management System Requirements Specification

UC_15 – Ester Qershori	<i>Customer Order History viewer</i>
Summary	<i>Allows customers to view their previous orders, including item details, total amount, order date, and reorder options.</i>
Dependency	<i>Customer must be logged in and have at least one completed order in the system.</i>
Actors	<i>Customer</i>
Preconditions	<ul style="list-style-type: none"> - <i>Customer is logged into their account.</i> - <i>Previous orders exist in the database.</i>
Description of the Main Sequence	<ol style="list-style-type: none"> 1. <i>Customer logs into their account.</i> 2. <i>Customer navigates to the "Order History" section.</i> 3. <i>System displays a list of previous orders with basic info (date, total, status).</i> 4. <i>Customer clicks on an order to view detailed information (items, instructions, payment method, etc.).</i> 5. <i>Customer may optionally click a "Reorder" button to place the same order again.</i>
Description of the Alternative Sequence	<p>3a. <i>If no previous orders exist, the system displays a message: "No past orders found."</i></p> <p>5a. <i>If an item from a past order is no longer available, the system warns the user and suggests alternatives.</i></p>
Non functional requirements	<ul style="list-style-type: none"> - <i>Performance - Orders must load in under 2 seconds.</i> - <i>Interface Clean, responsive design with date filters and search capability.</i> - <i>Data Integrity - Orders must reflect exactly what was placed and paid for.</i> - <i>Security Only the logged-in user can access their order history.</i>
Postconditions	<ul style="list-style-type: none"> - <i>The customer has viewed their past orders.</i> - <i>A new order may be initiated via the reorder option.</i>

Restaurant Management System Requirements Specification

UC_16 – Ester Qershori	<i>Employee Role Assignment</i>
Summary	<i>Allows the administrator to assign, update, or change roles for any registered employee in the system (e.g., Chef, Waiter, Cashier). Role changes dynamically update the employee's system permissions and dashboard access.</i>
Dependency	<i>Employee profiles must already exist in the system.</i>
Actors	<i>Primary Actor: Administrator</i>
Preconditions	<ul style="list-style-type: none"> - Administrator is logged into the system. - The target employee exists in the employee database.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Admin navigates to the "Employee Management" section. 2. Admin selects an employee from the list. 3. System displays the employee profile and current role. 4. Admin selects a new role from the predefined list (e.g., Waiter, Chef, Inventory Manager). 5. System saves the new role and automatically adjusts the employee's permissions. 6. The employee is notified of the role update.
Description of the Alternative Sequence	<p>2a. If the selected employee does not exist or is inactive, the system shows an error message.</p> <p>5a. If role update fails (e.g., due to database error), the system prompts the admin to retry.</p>
Non functional requirements	<ul style="list-style-type: none"> - Security - Only admins can access and modify employee roles. - Performance - Role update and permission refresh must complete in under 2 seconds. - Scalability - Supports large staff databases across multiple locations. - Auditability - Logs every role change for accountability.
Postconditions	<ul style="list-style-type: none"> - The employee's role is updated in the system. - Access permissions are immediately adjusted. - Change is logged for auditing.

UC_29 – Ester Qershori	<i>Record Customer Allergy Warning</i>
Summary	<i>Allows customers (or waiters) to input, update, and save known allergies in their profile for future order warnings.</i>

Restaurant Management System Requirements Specification

UC_29 – Ester Qershori	Record Customer Allergy Warning
Dependency	<i>Customer profile must exist in the system.</i>
Actors	<i>Primary Actor: Customer Secondary Actor: Waiter (if entering on behalf of customer)</i>
Preconditions	<ul style="list-style-type: none"> - <i>Customer must be logged into the system.</i> - <i>System must have allergy data fields available.</i>
Description of the Main Sequence	<ol style="list-style-type: none"> 1. <i>Customer logs into their account.</i> 2. <i>Customer navigates to “Allergy Settings” or is prompted during order start.</i> 3. <i>System prompts for known allergies.</i> 4. <i>Customer enters allergy information.</i> 5. <i>System asks for confirmation.</i> 6. <i>Customer confirms allergy list.</i> 7. <i>System saves allergies, logs with timestamp, and updates profile</i>
Description of the Alternative Sequence	<p><i>5a. If customer declines to confirm, system allows editing allergy input.</i> <i>7a. If system fails to save data, display error and prompt retry.</i></p>
Non functional requirements	<ul style="list-style-type: none"> - <i>Performance: Allergy data must be saved within 2 seconds.</i> - <i>Data Integrity: Allergy records must be accurately stored and not lost.</i> - <i>Security: Only authorized users can view or modify allergy information.</i> - <i>Auditability: All allergy updates must be timestamped for legal tracking.</i>
Postconditions	<ul style="list-style-type: none"> - <i>Customer’s allergy information is saved to their profile.</i> - <i>System uses the allergy data to warn during future order processes.</i>

UC_20 – Erdi Perhati	Valet Parking Request Management
Summary	Customers request valet service, and staff coordinate car pickup and return.
Dependecny	None.
Actors	Primary: Customer Secondary: Valet Staff
Preconditions	Valet service must be enabled and configured in the system.
Description of the Main Sequence	<p>Customer submits valet request upon arrival or checkout. System records the vehicle details and timestamp. Valet staff are notified with the request. Customer is notified once car is ready.</p>

Restaurant Management System Requirements Specification

Description of the Alternative Sequence	If valet service is unavailable, the system informs the customer and cancels the request.
Nonfunctional Requirements	Requests must reach valet staff terminals within 5 seconds of submission.
Postconditions	The valet service is logged and tracked for service quality and coordination.

UC_21 – Erdi Perhati	Utility Consumption Tracking
Summary	The system logs utility consumption (water, electricity, gas) for each day or billing cycle.
Dependency	None.
Actors	Primary: Admin / Manager Secondary: System (Automated)
Preconditios	Utility meters or logs must be integrated or manually updated into the system.
Description of the Main Sequence	System receives or allows entry of utility usage data. Data is logged daily and categorized by type (e.g., kitchen gas, lighting, water). Reports are generated weekly/monthly for review.
Description of the Alternative Sequence	If values are missing or inconsistent, system highlights gaps for correction.
Nonfunctional Requirements	Data entry and reports must support unit conversions (kWh, liters, m ³).
Postconditions	Utility data is stored, available for auditing, and usable in cost-efficiency analysis.

Restaurant Management System Requirements Specification

UC_22 – Erdi Perhati	Cleaning Schedule Management
Summary	The manager creates and assigns cleaning tasks and schedules for different restaurant zones.
Dependecny	UC13 – Shift Scheduling
Actors	Primary: Manager Secondary: Cleaning Staff
Preconditions	Cleaning zones and staff must be registered in the system.
Description of the Main Sequence	Manager opens the cleaning module and selects time/date/area. Assigns staff and sets task descriptions (e.g., “sanitize kitchen,” “wipe front glass”). System notifies assigned staff and logs task status.
Description of the Alternative Sequence	If a staff member is unavailable, the system suggests alternatives.
Nonfunctional Requirements	Tasks must be markable as complete via mobile/staff interface.
Postconditions	Scheduled cleaning tasks are tracked and recorded for compliance and monitoring.

UC_23 – Erdi Perhati	Security Camera Footage Log
Summary	The system logs timestamps and access history of security footage playback.
Dependecny	None.
Actors	Primary: Manager / Security Staff
Preconditions	Security cameras must be integrated and system storage available.
Description of the Main Sequence	Manager accesses the Security tab. Selects camera, date, and time range. System logs who accessed which footage and when. Playback history is archived.
Description of the Alternative Sequence	If access permissions are invalid, system denies access and logs the attempt.
Nonfunctional Requirements	Footage logs must be encrypted and securely stored for at least 30 days.
Postconditions	All security access is logged and reviewable in the audit history.

Restaurant Management System Requirements Specification

UC_24 – Keisi Loci	Training Session Scheduling
Summary	Managers schedule employee training sessions, assign staff, and track attendance.
Dependency	UC16 – Employee Role Assignment
Actors	Primary: Manager / Trainer Secondary: Employees
Preconditions	Training content and roles must be predefined in the system.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Manager creates a training session with topic, trainer, and time. 2. Assigns employees to the session. 3. System notifies employees and tracks attendance during/after.
Description of the Alternative Sequence	If employee cannot attend, manager may reassign to future sessions.
Non functional requirements	- Attendance status must be markable in real time and viewable by admins.
Postconditions	- Training records are saved and linked to employee profiles.

UC_25 – Keisi Loci	Tableware Damage Reporting
Summary	Staff reports damaged or broken tableware to track incidents and manage replacements.
Dependency	None
Actors	Primary: Waiter / Busser
Preconditions	User must be logged in with access to damage report module.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Staff selects “Report Damage” option. 2. Enters item type (plate, glass, utensil), quantity, and description. 3. Report is logged and sent to the manager for inventory updates or review.
Description of the Alternative Sequence	If report is incomplete, system prompts for missing information.

Restaurant Management System Requirements Specification

UC_25 – Keisi Loci	Tableware Damage Reporting
Non functional requirements	Reports must be timestamped and available in monthly inventory summaries.
Postconditions	Damage incidents are recorded and used to track costs or restocking needs.

UC_27 – Keisi Loci	Dishwashing Machine Maintenance Scheduling
Summary	The system schedules and logs maintenance sessions for dishwashing machines.
Dependency	None
Actors	Admin / Maintenance Staff
Preconditions	Machine data and maintenance history must be available in the system.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Admin accesses the maintenance module. 2. Selects dishwashing machine and sets maintenance date, time, and staff. 3. System schedules a reminder and logs the task upon completion
Description of the Alternative Sequence	If maintenance is missed, the system escalates with priority notifications to management.
Non functional requirements	Maintenance logs must be exportable and retained for at least 6 months.
Postconditions	Scheduled maintenance is completed and logged, ensuring equipment upkeep.

UC_28 – Keisi Loci	Play Area Access Registration
Summary	The system records children's play area access, tracking entry and exit times and guardian consent.
Dependency	None
Actors	Primary: Front-of-house Staff Secondary: Guardian / Parent
Preconditions	Play area must be active and monitored. Guardian consent must be given.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Staff registers child with name, age, and guardian's name/contact. 2. Entry time is recorded; guardian signs digital or physical consent form. 3. Upon exit, staff logs departure time and confirms with guardian.
Description of the Alternative Sequence	If the play area is at capacity, the system notifies the staff and offers a waitlist.
Non functional requirements	Access records must be stored securely and visible only to authorized personnel.
Postconditions	Play area access is registered and traceable for safety and auditing.

UC_30 – Arlis Arapi	Manage Dining Time Limits Warnings
Summary	The system tracks table usage duration and issues warnings as dining time limits approach.
Dependency	UC2 – Table Reservation UC6 – Place Order
Actors	Primary: System Secondary: Waiter / Host
Preconditions	Dining time limits must be configured per reservation or table type.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. System starts a dining session timer upon table activation. 2. When time reaches threshold (e.g., 15 mins remaining), it issues an alert to the waiter. 3. Final notification is sent at or near the limit expiration.
Description of the Alternative Sequence	If customers request more time, manager can extend the dining limit through override.

Restaurant Management System Requirements Specification

Nonfunctional Requirements	Time tracking must be accurate to the minute; alerts must be non-disruptive but timely.
Postconditions	Dining session is monitored, and warnings are issued appropriately.

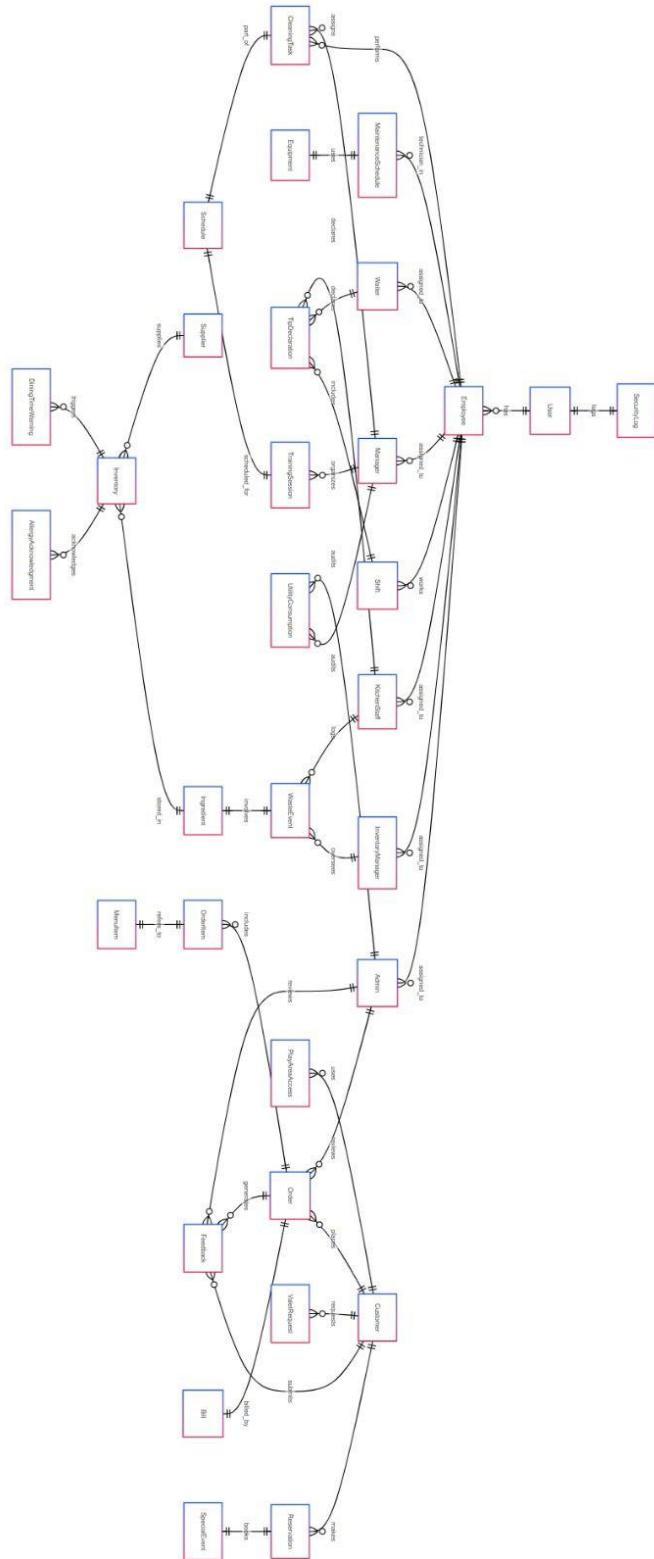
UC_17 – Arlis Arapi	Monitor Supplier Performance
Summary	Enables the inventory manager to track and evaluate supplier performance based on delivery times, order accuracy, and fulfillment rate. Helps inform future ordering decisions.
Dependency	Supplier order data must be logged in the system.
Actors	Primary: Inventory Manager
Preconditions	Supplier profiles and past order records must exist in the system. Delivery logs and order statuses must be consistently updated.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Inventory Manager navigates to the 'Supplier Performance' dashboard. 2. System displays a list of active suppliers with summary KPIs (on-time delivery %, fulfillment accuracy, average delay, etc.). 3. Manager selects a supplier to view detailed performance history. 4. System displays historical data including past orders, delivery dates, issues logged, and average delivery time. 5. Manager uses this information to adjust future orders or rate suppliers.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 2a. If no data is available for a supplier, the system displays a message: "No performance data available yet." 4a. If a data entry is inconsistent or missing (e.g., missing delivery time), system flags the entry for manual review.
Nonfunctional Requirements	<ul style="list-style-type: none"> - Analytics: Dashboards should offer visual graphs and sortable tables. - Performance: Dashboard should load within 3 seconds. - Data Integrity: Accurate, tamper-proof supplier performance metrics. - Interface: User-friendly with filters by supplier, date, or performance metric.
Postconditions	<ul style="list-style-type: none"> - Inventory manager gains insights into supplier reliability. - Data can be exported or used in reports for procurement decisions.

UC_19 – Arlis Arapi	Special Event Booking
Summary	Allows customers to request a reservation for special events (e.g., birthdays, anniversaries, corporate dinners), with options to customize menu, seating layout, decorations, and additional services.
Dependency	The table reservation and menu systems must be functional.
Actors	Primary Actor: Customer Supporting Actor: Restaurant Manager
Preconditions	<ul style="list-style-type: none"> - Customer is logged in or provides contact details. - Desired date/time must be available. - Event requests must be submitted in advance (e.g., 48 hours prior).
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Customer navigates to the 'Special Event Booking' section. 2. Customer selects desired date, time, event type, and number of guests. 3. System checks availability of tables and event capacity. 4. Customer selects optional customizations (menu, decorations, music, seating layout, etc.). 5. Customer submits the request. 6. Restaurant Manager reviews the request and confirms or contacts the customer for clarification. 7. System sends booking confirmation and details.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 3a. If no event slots are available, system suggests alternative dates/times. 6a. If manager rejects the request (e.g., due to overcapacity), the system notifies the customer and proposes modifications.
Nonfunctional Requirements	<ul style="list-style-type: none"> - Interface: Intuitive form with customization options and calendar view. - Performance: Event availability should load in under 2 seconds. - Notifications: Email/SMS confirmation and reminders for both customer and manager. - Data Handling: Stores event details securely for future reference.
Postconditions	<ul style="list-style-type: none"> - Event reservation is recorded and assigned to the manager for preparation. - Customer receives confirmation and a summary of the booking.

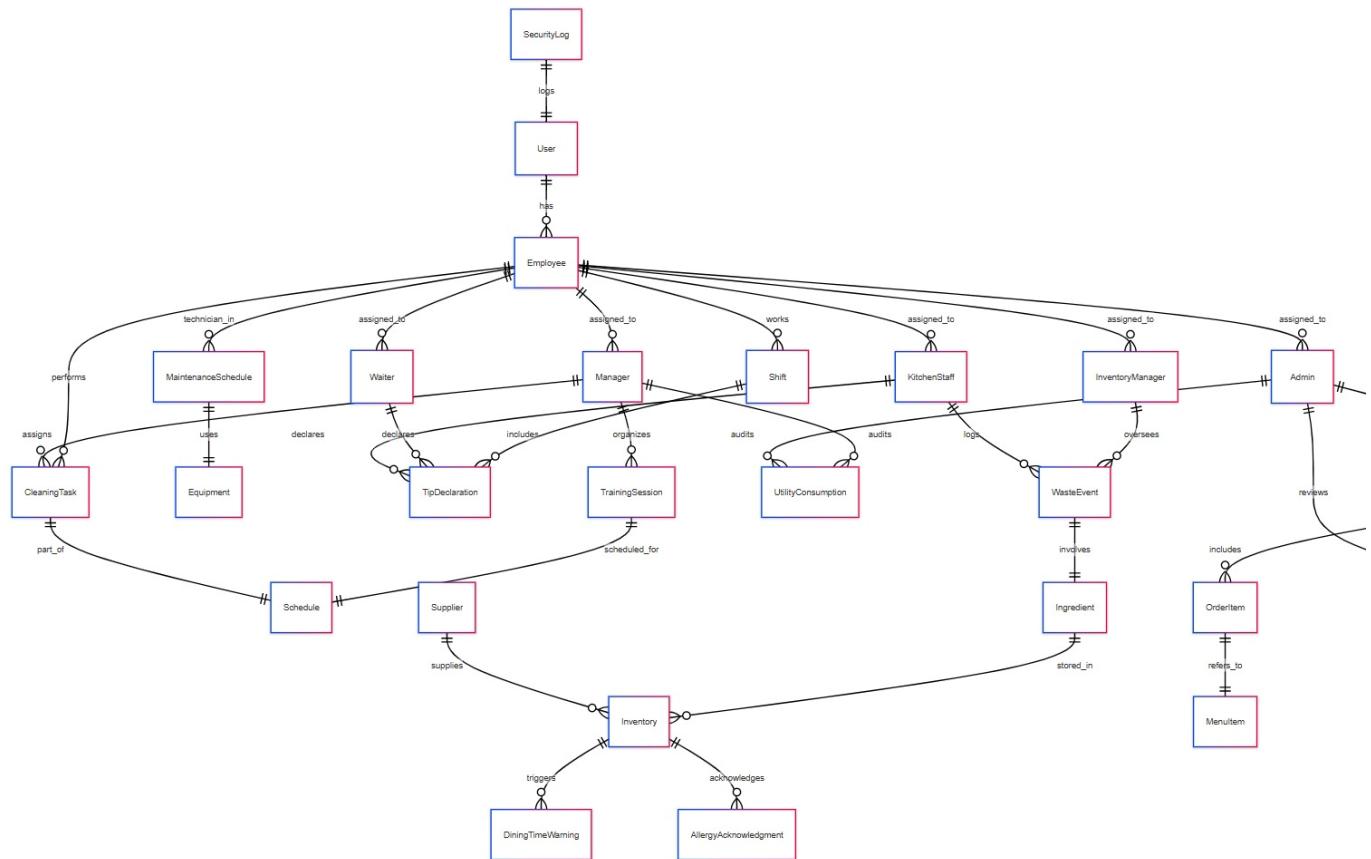
UC_18 – Arlis Arapi	Declare Tips & Automated Distribution
Summary	Allows waiters to declare tips received during their shift, after which the system automatically calculates and distributes the tip amount among eligible staff (e.g., bussers, kitchen) based on predefined rules.
Dependency	<ul style="list-style-type: none"> - Waiter's shift must be completed or closed. - Tip distribution rules must be configured in the system.
Actors	Primary Actor: Waiter Secondary Actor: System
Preconditions	<ul style="list-style-type: none"> - Waiter is logged in and has a completed shift. - The system must have access to the role-based distribution policy. - Tip declaration is enabled for the shift.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. Waiter logs into the system and navigates to the “Declare Tips” section. 2. Waiter enters the total tips received (e.g., cash or manual card tips). 3. System validates the entry and confirms shift completion. 4. System retrieves active tip distribution policy. 5. System calculates the distribution of the declared tips among roles (e.g., 60% to waiter, 25% to bussers, 15% to kitchen). 6. System saves the declaration and distribution record. 7. Notification is sent to all involved roles (e.g., bussers and kitchen staff) with their allocated share.
Description of the Alternative Sequence	<ol style="list-style-type: none"> 2a. If the shift is still open or no sales were registered, the system blocks the declaration and displays a warning. 5a. If distribution rules are missing, the system notifies the admin and holds the declaration for manual review.
Nonfunctional Requirements	<ul style="list-style-type: none"> - Performance: Tip distribution must be calculated in under 2 seconds. - Transparency: Each staff member can view their share and its breakdown. - Security: Only the declaring waiter can submit tips for their shift. - Auditability: All declarations and distributions are logged for admin review.
Postconditions	<ul style="list-style-type: none"> - Tip is declared, calculated, and distributed based on policy. - Records are stored and accessible for employees and managers. - All involved parties are notified.

5. Diagrams

5.1 ER Diagram

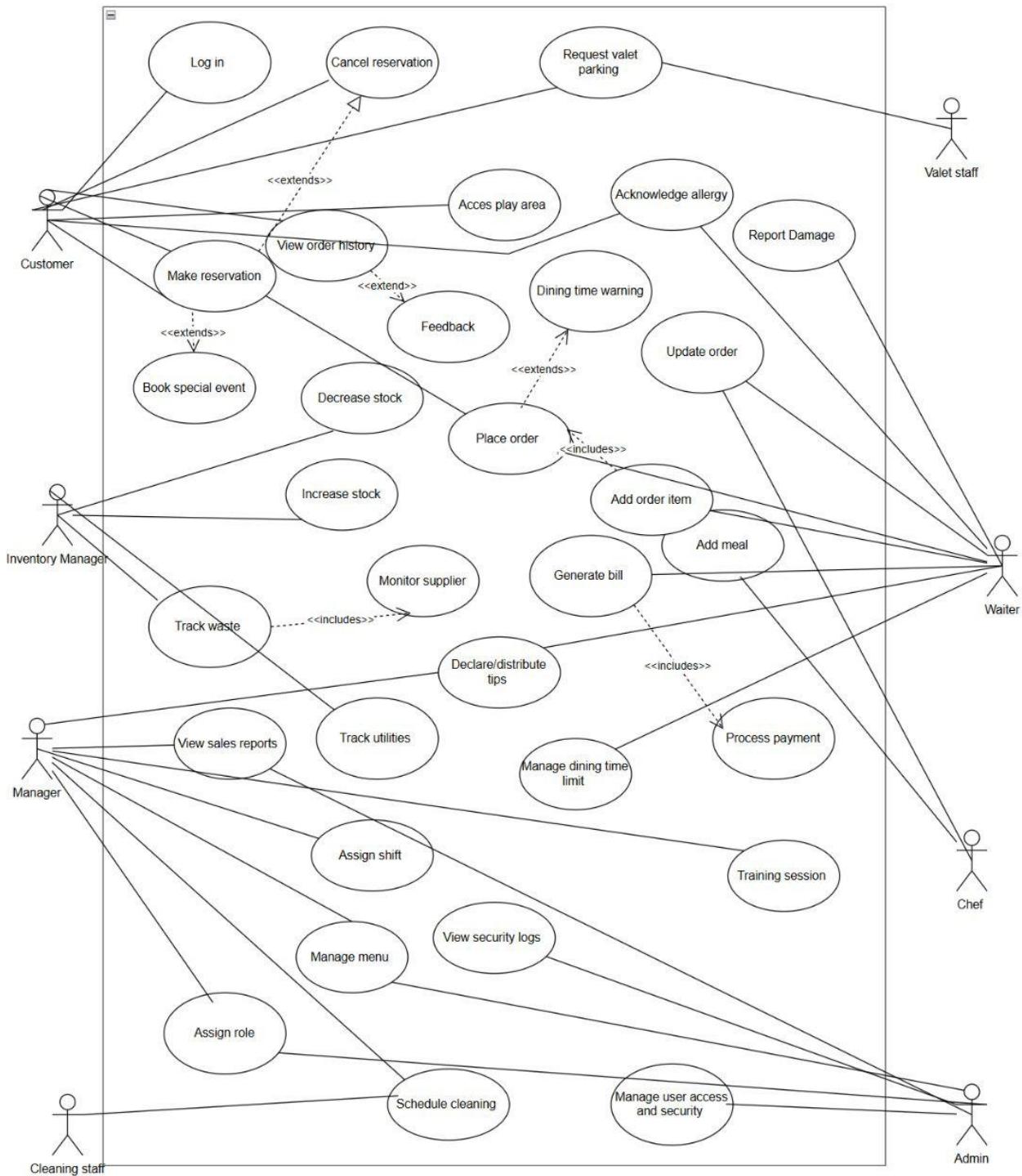


Restaurant Management System Requirements Specification



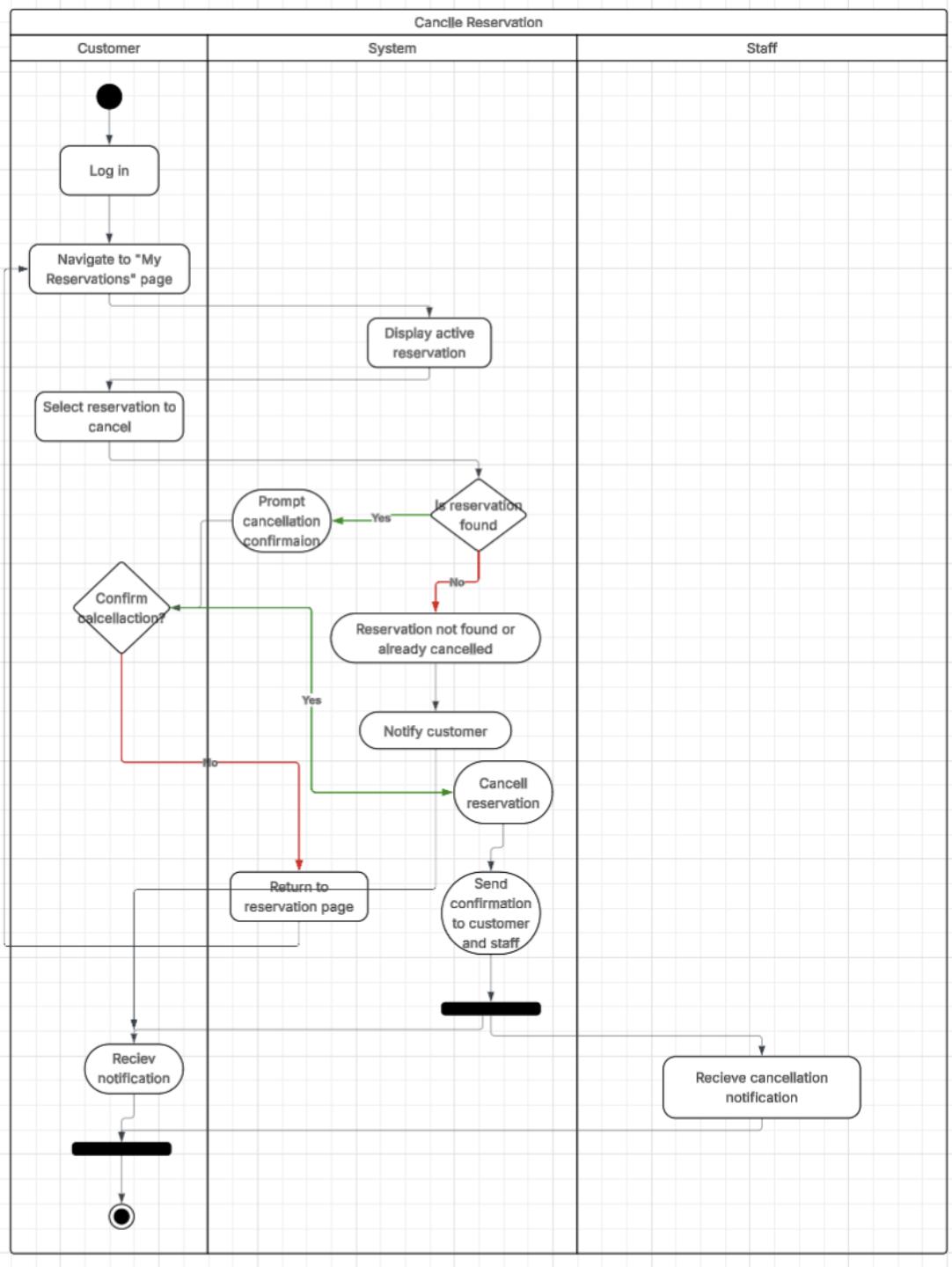
Restaurant Management System Requirements Specification

5.2 Use Case Diagram (general)

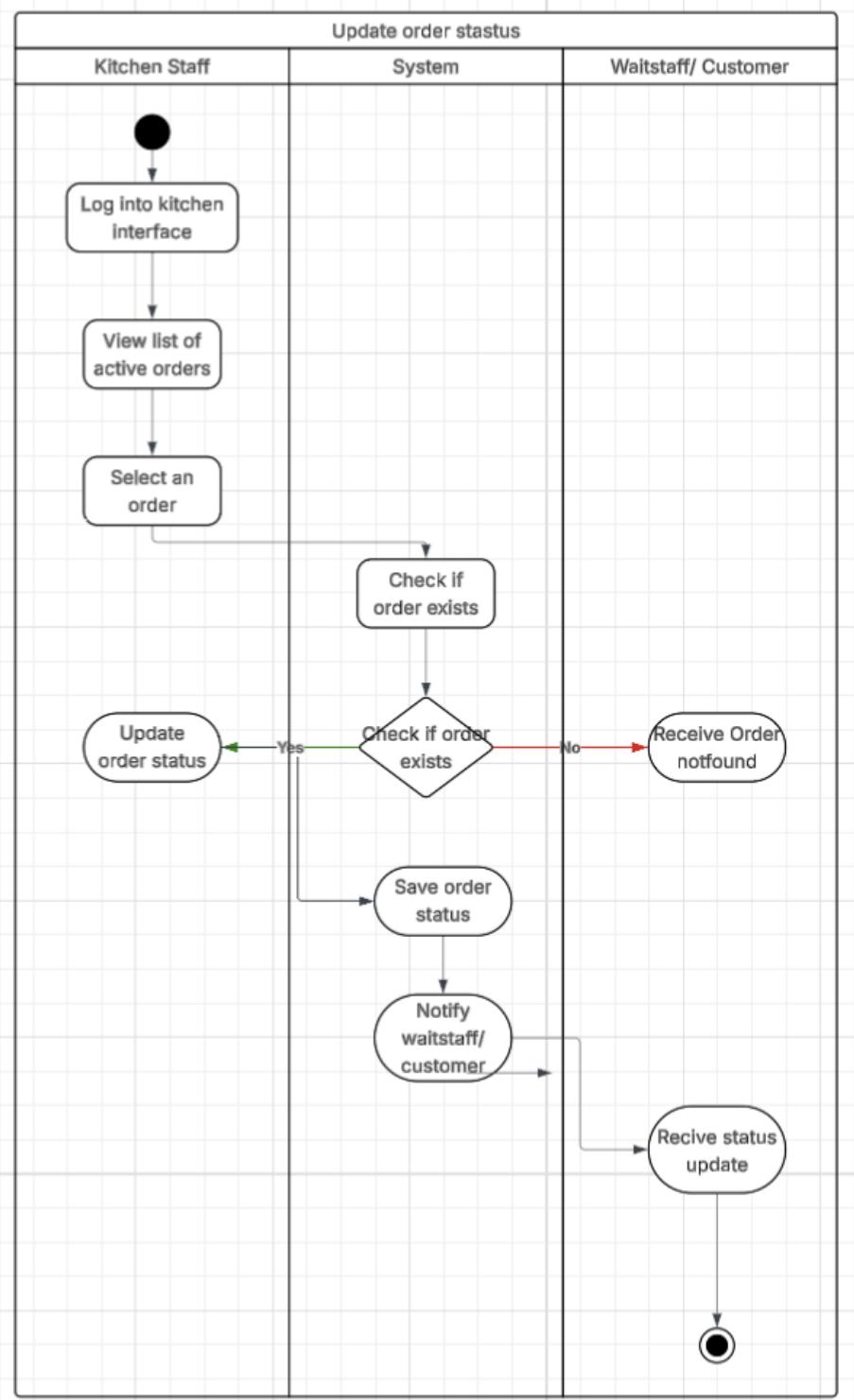


5.3 Activity Diagram

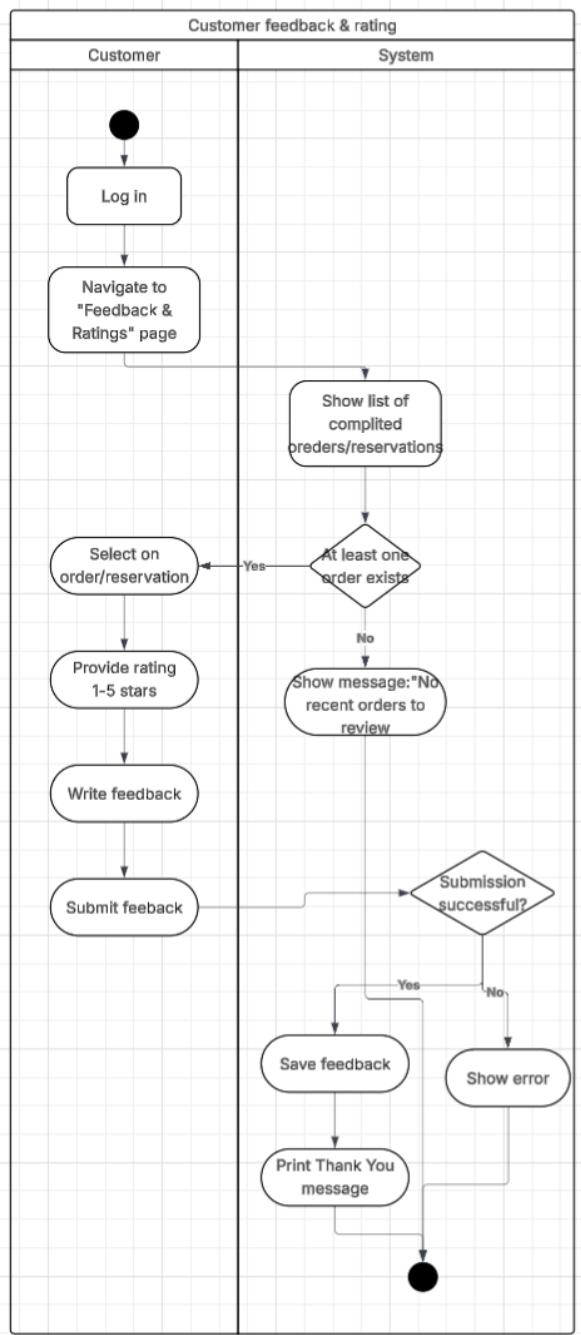
AC_10 Table Cancellation – Joldi Xure



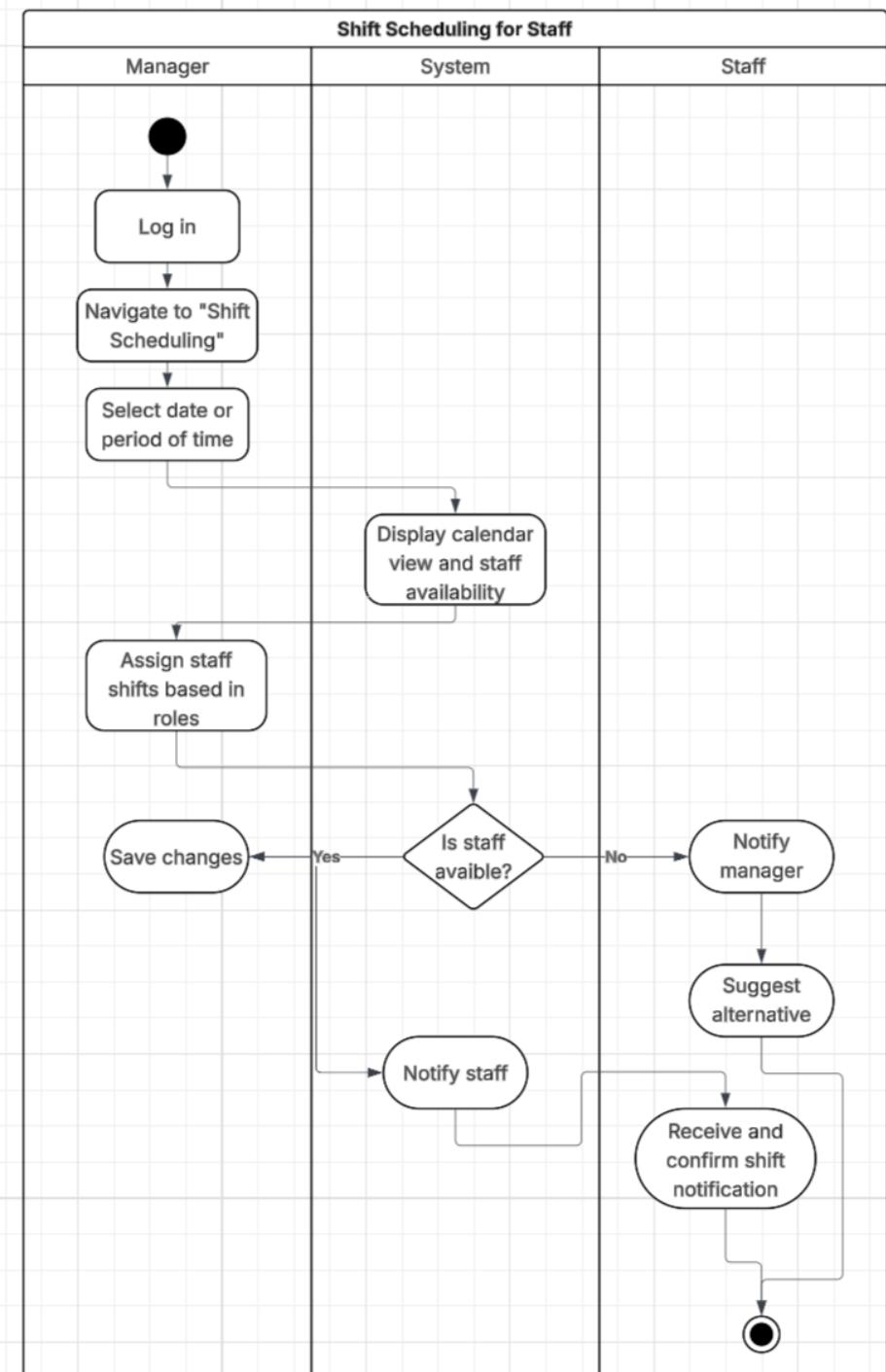
AC-11 Update Order Status – Joldi Xure



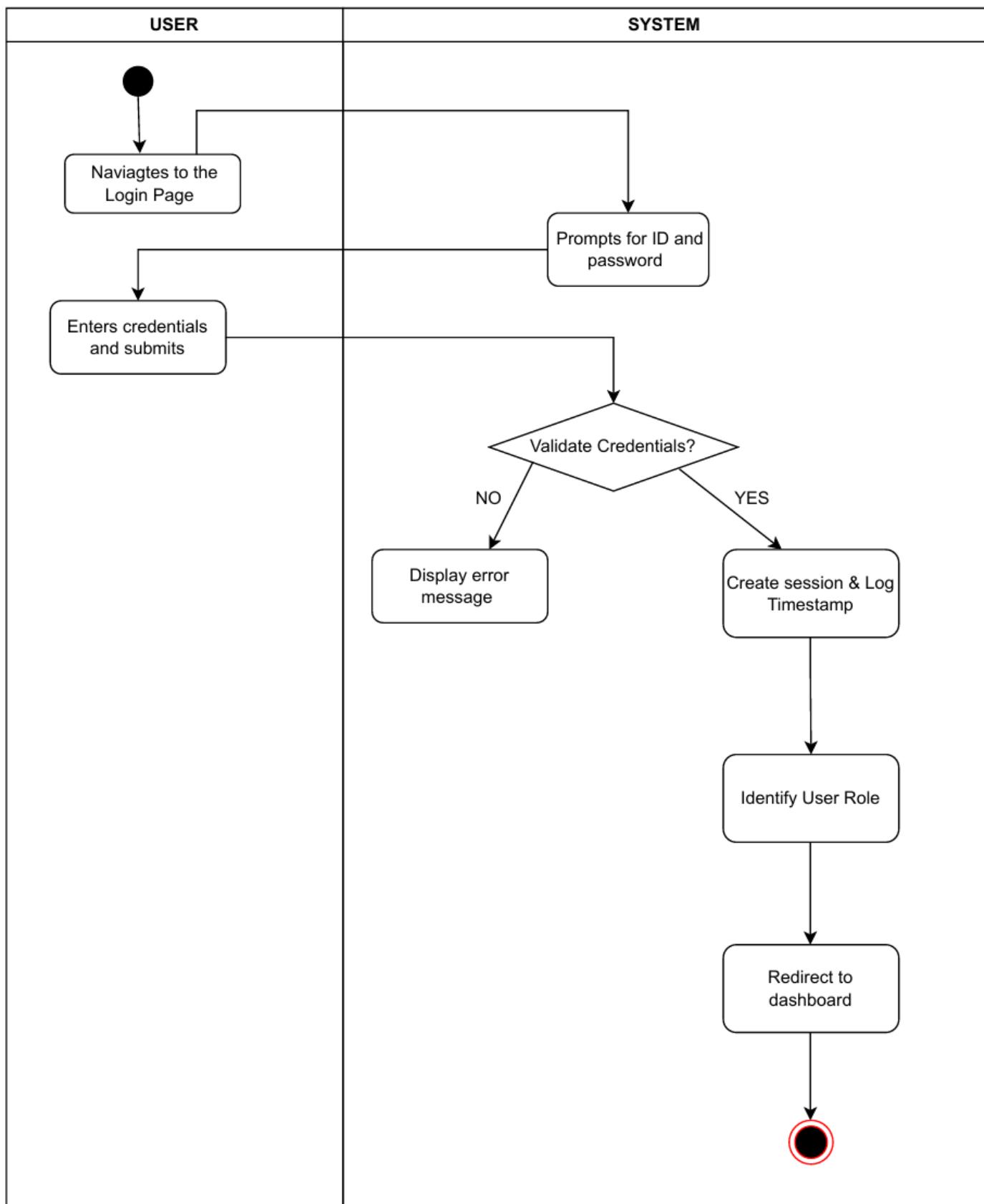
AC-12 Customer Feedback & Ratings– Joldi Xure



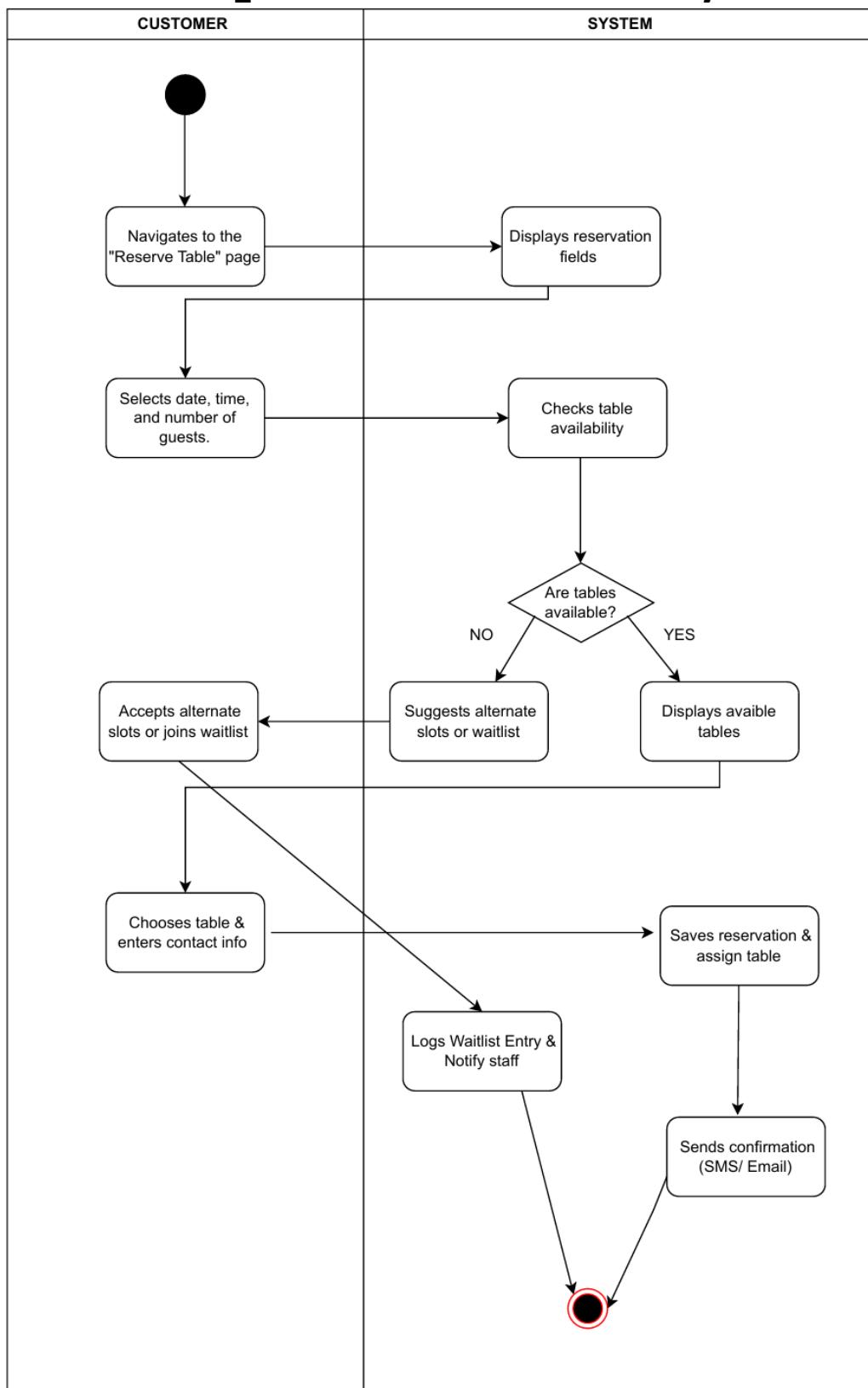
AC-13 Shift Scheduling for Staff– Joldi Xure



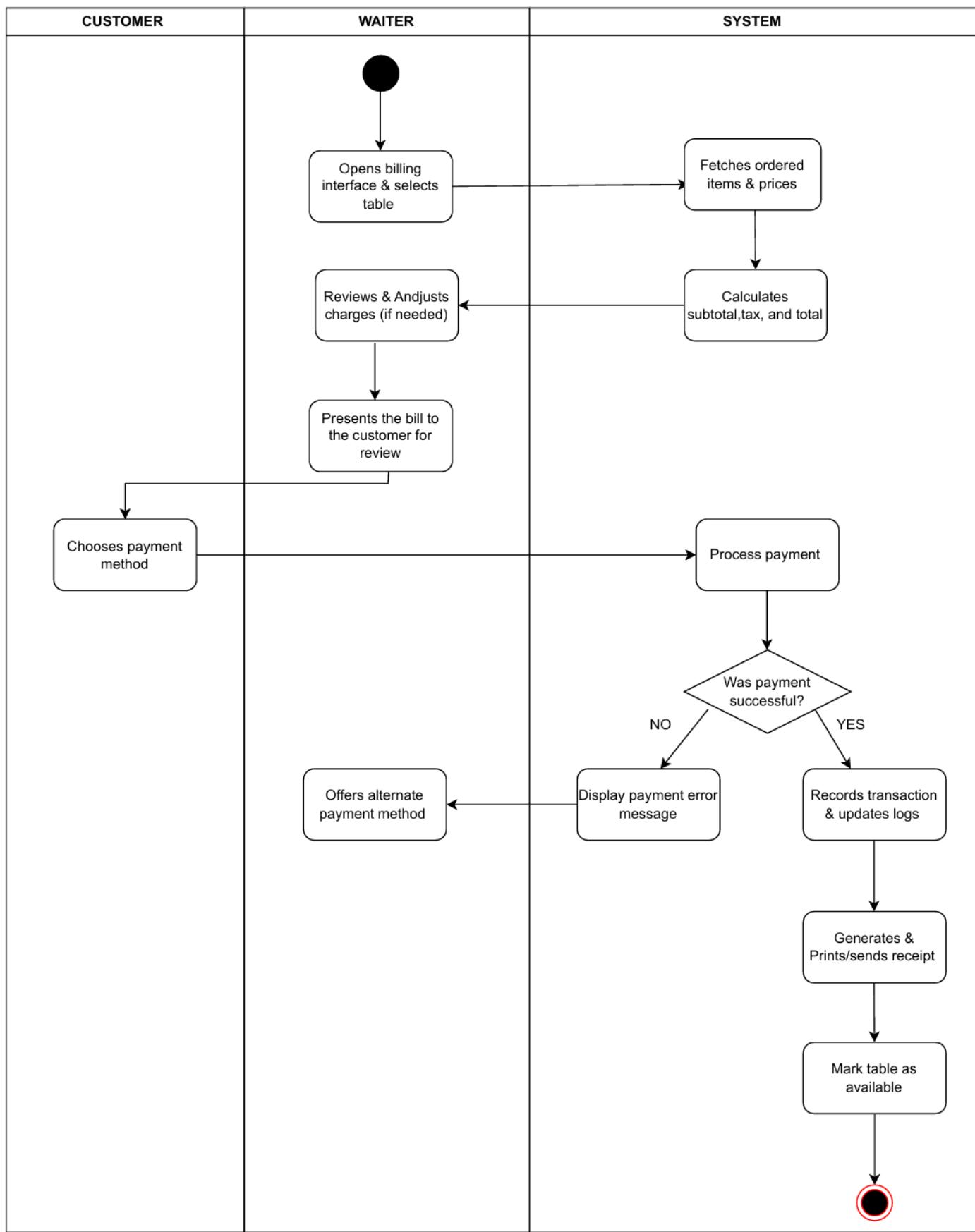
AC_01 – User Login– Gloria Traja



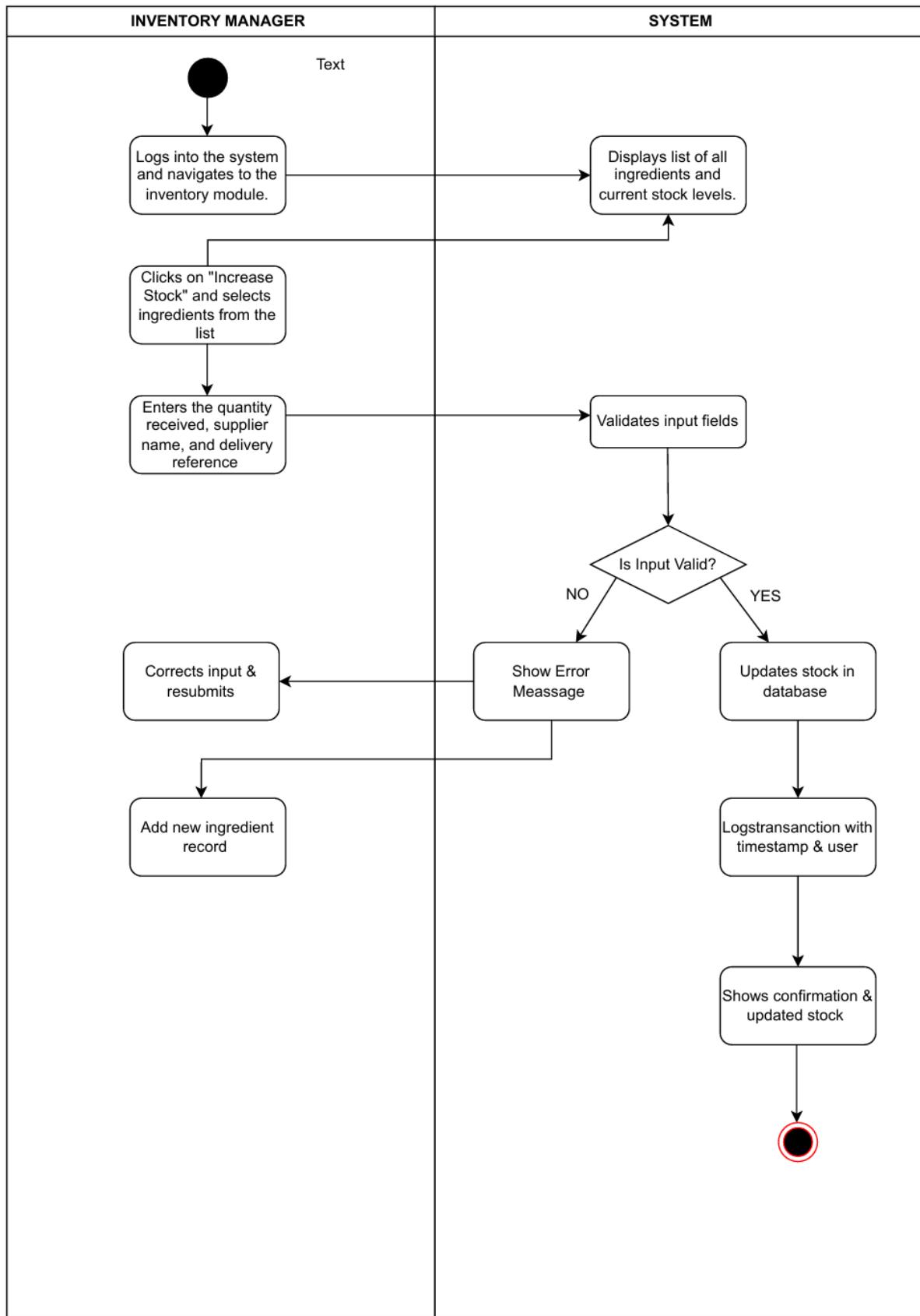
AC_02 – Table Reservation – Gloria Traja



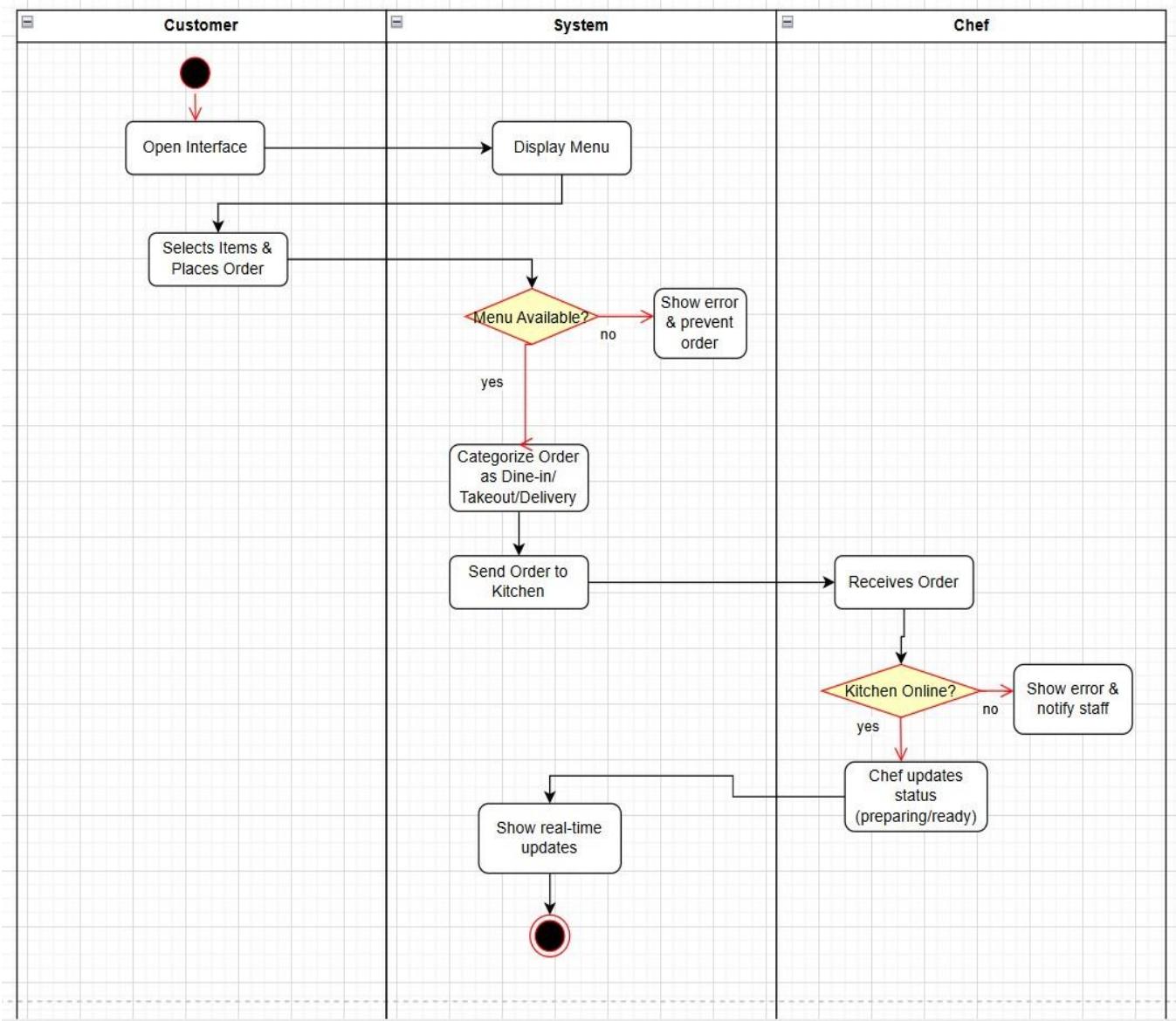
AC_03 – Bill Generation & Payment Processing – Gloria Traja



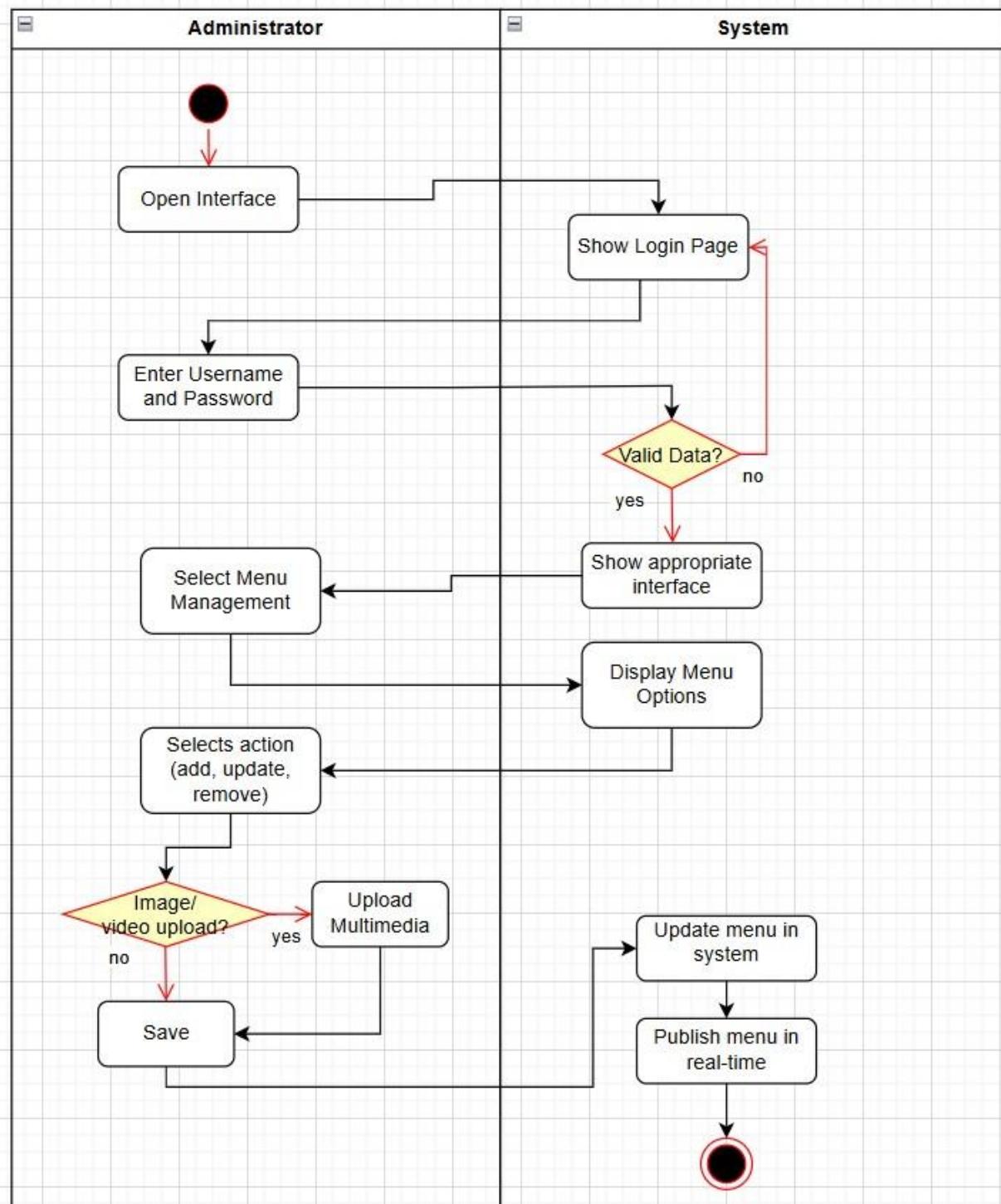
AC_04 – Ingredient Stock Management Increase – Gloria Traja



AC_06: Place Order - Alisa Tozaj

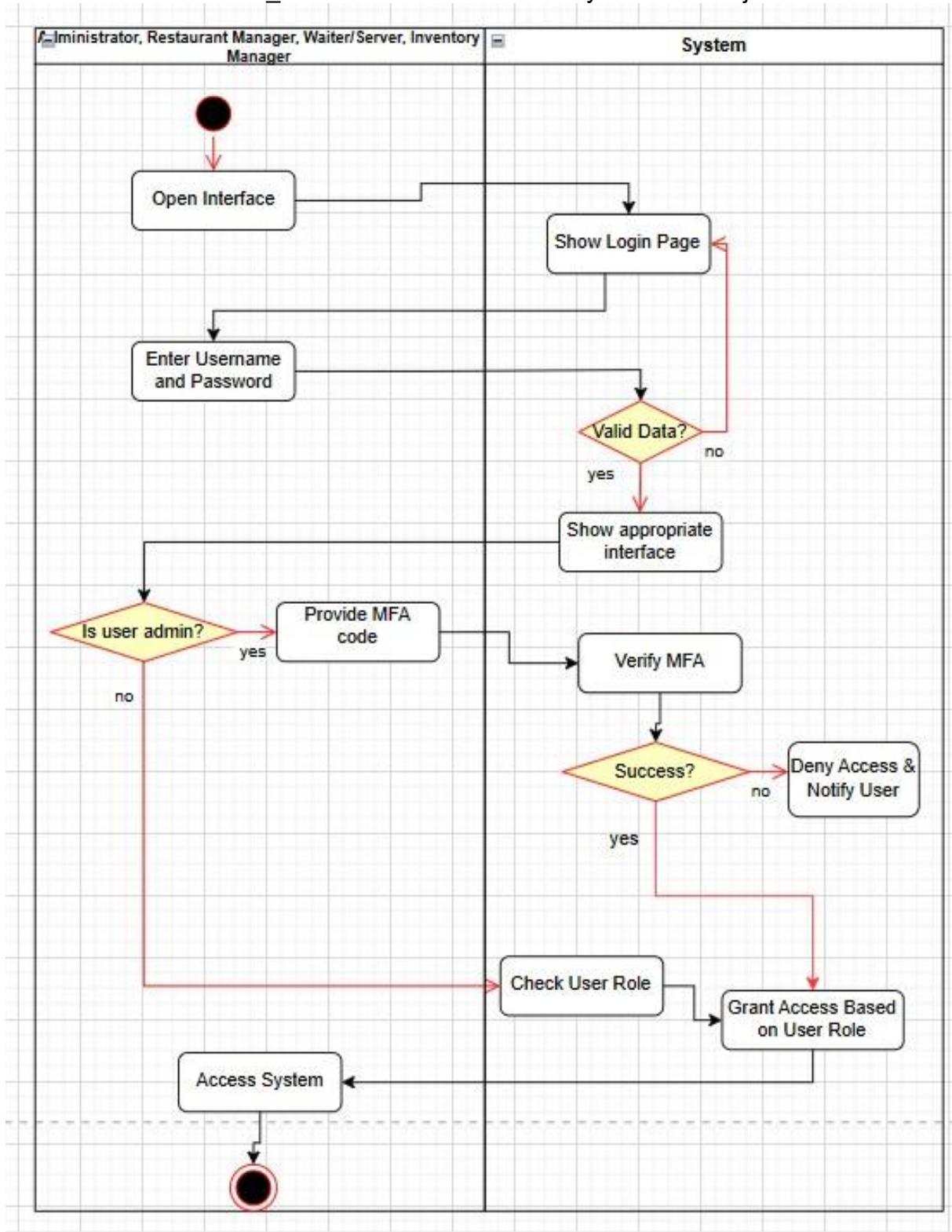


AC_07 – Menu Management – Alisa Tozaj



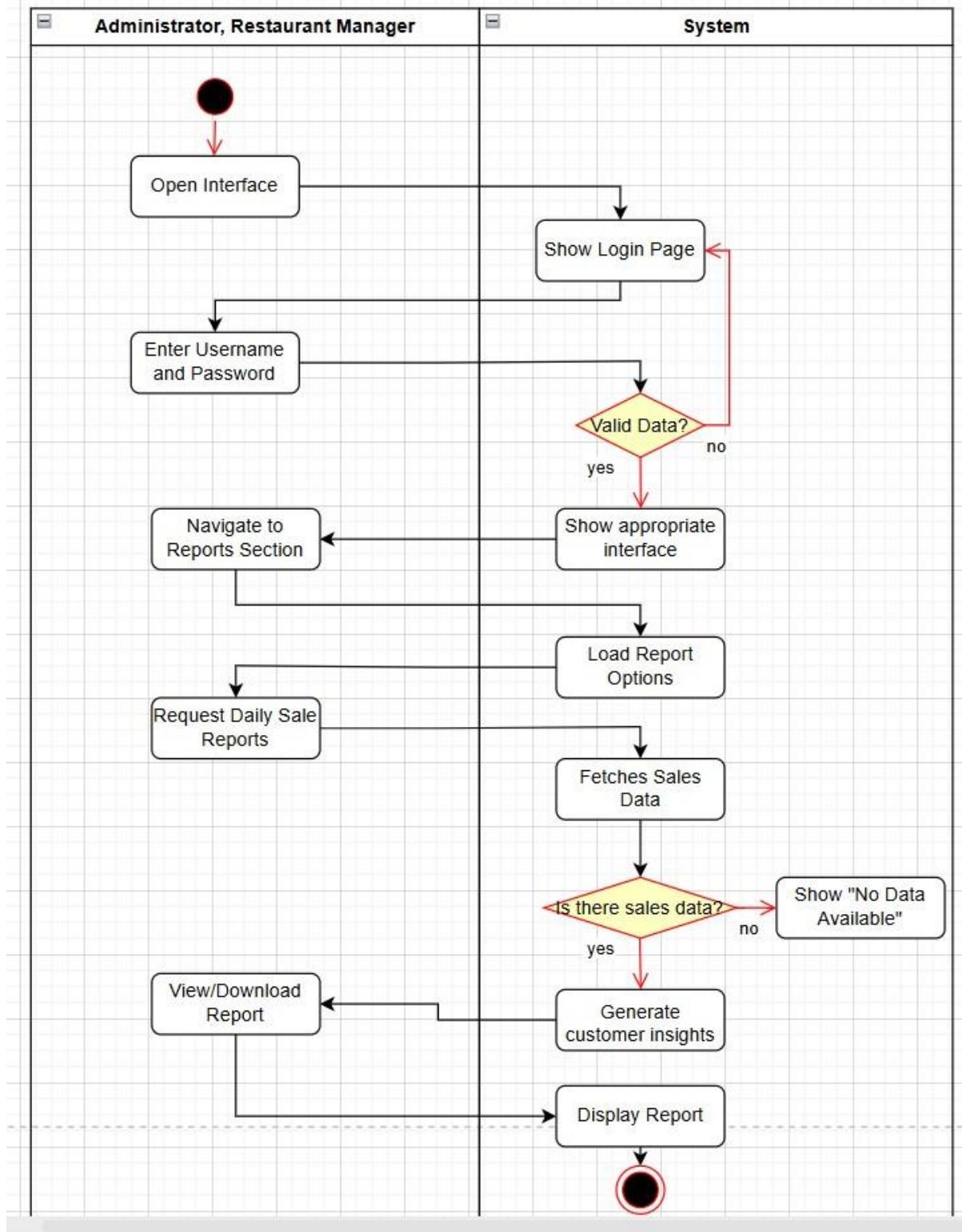
Restaurant Management System Requirements Specification

AC_08 – User Access & Security – Alisa Tozaj



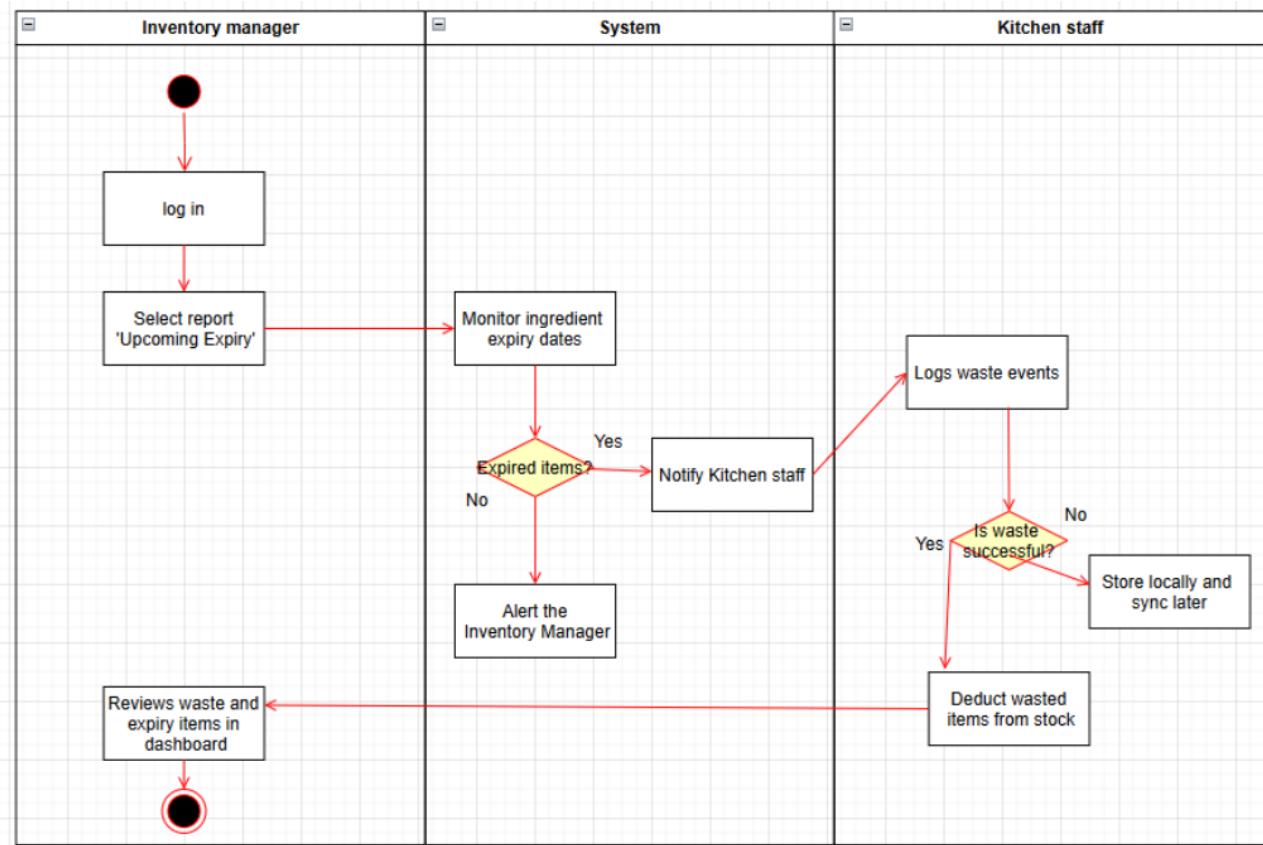
Restaurant Management System Requirements Specification

AC_09 – Sales & Reports – Alisa Tozaj



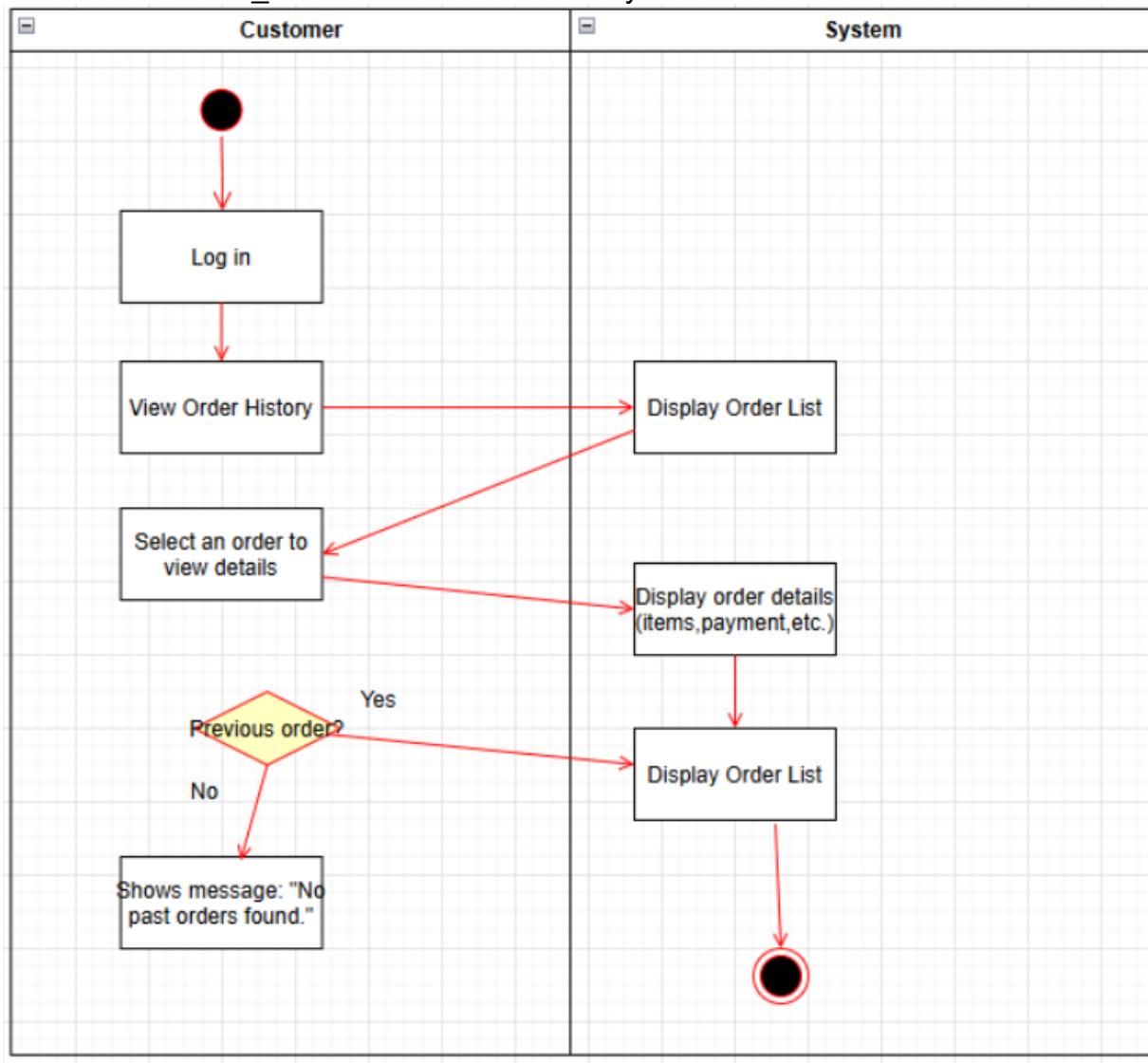
Restaurant Management System Requirements Specification

Ester Qershori – AC_14: Waste & Expiry Tracking



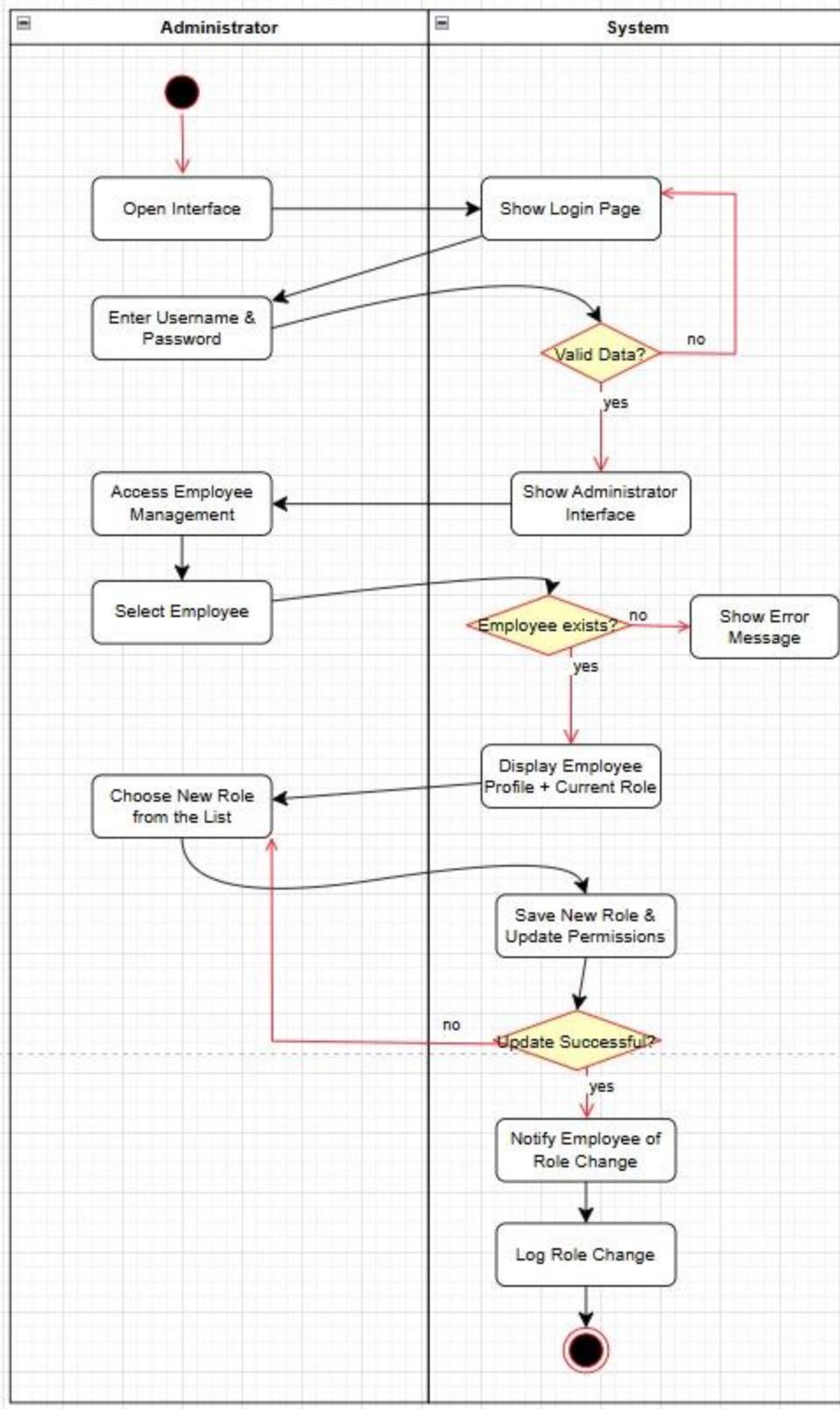
Restaurant Management System Requirements Specification

AC_15: Customer Order History Viewer – Ester Qershori



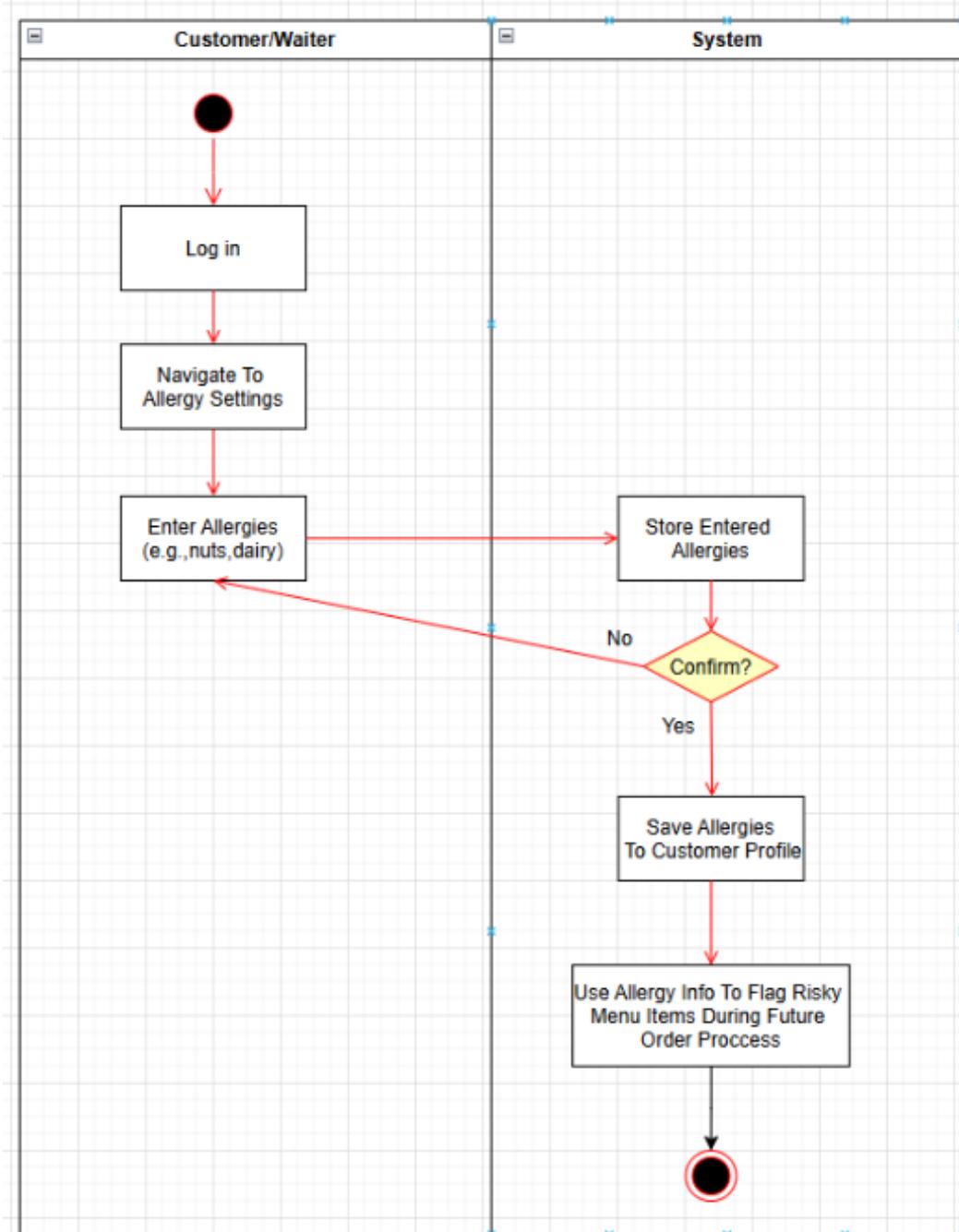
Restaurant Management System Requirements Specification

AC_16: Employee Role Assessment – Ester Qershori



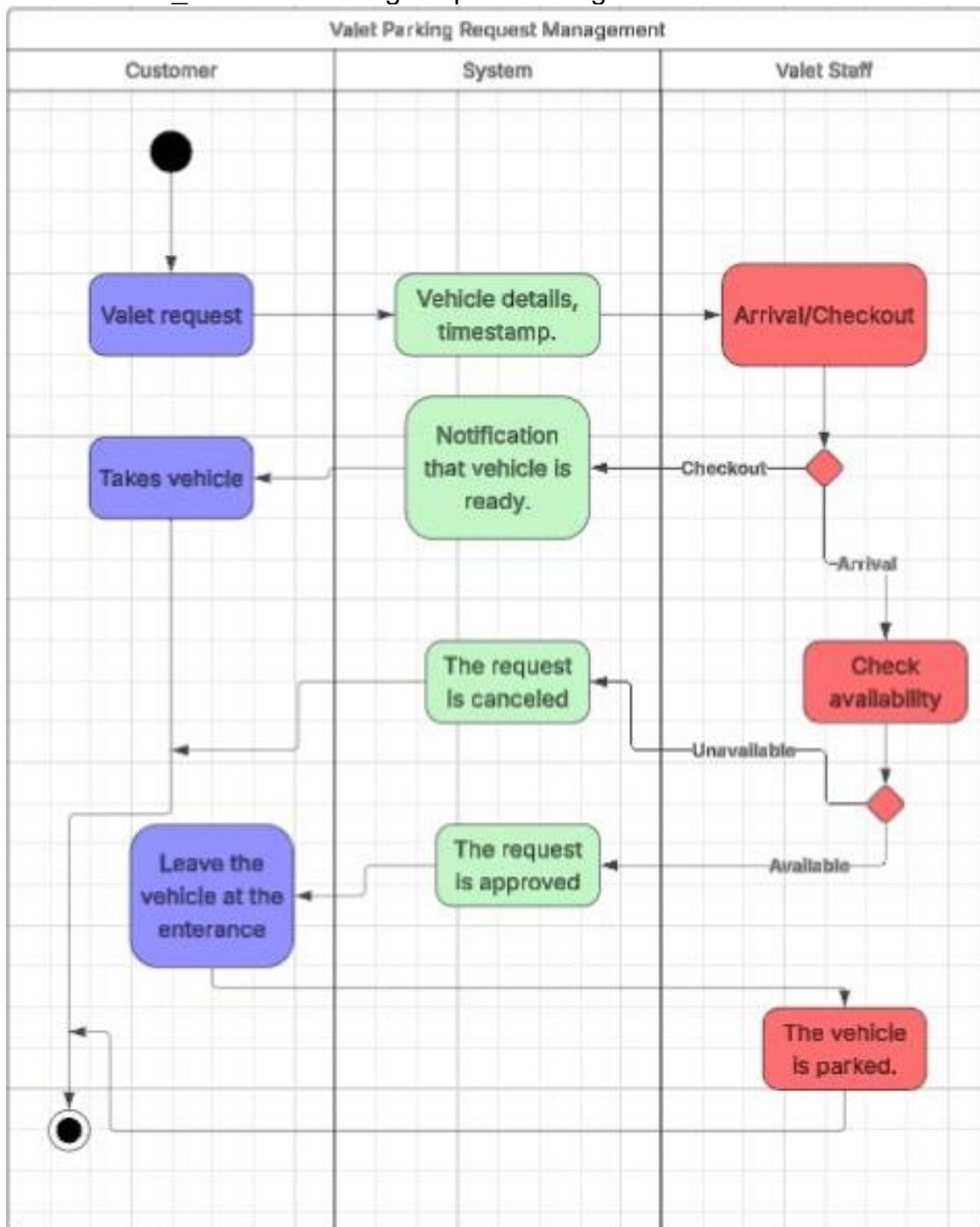
Restaurant Management System Requirements Specification

AC_29: Record Customer Allergy Warning – Ester Qershori



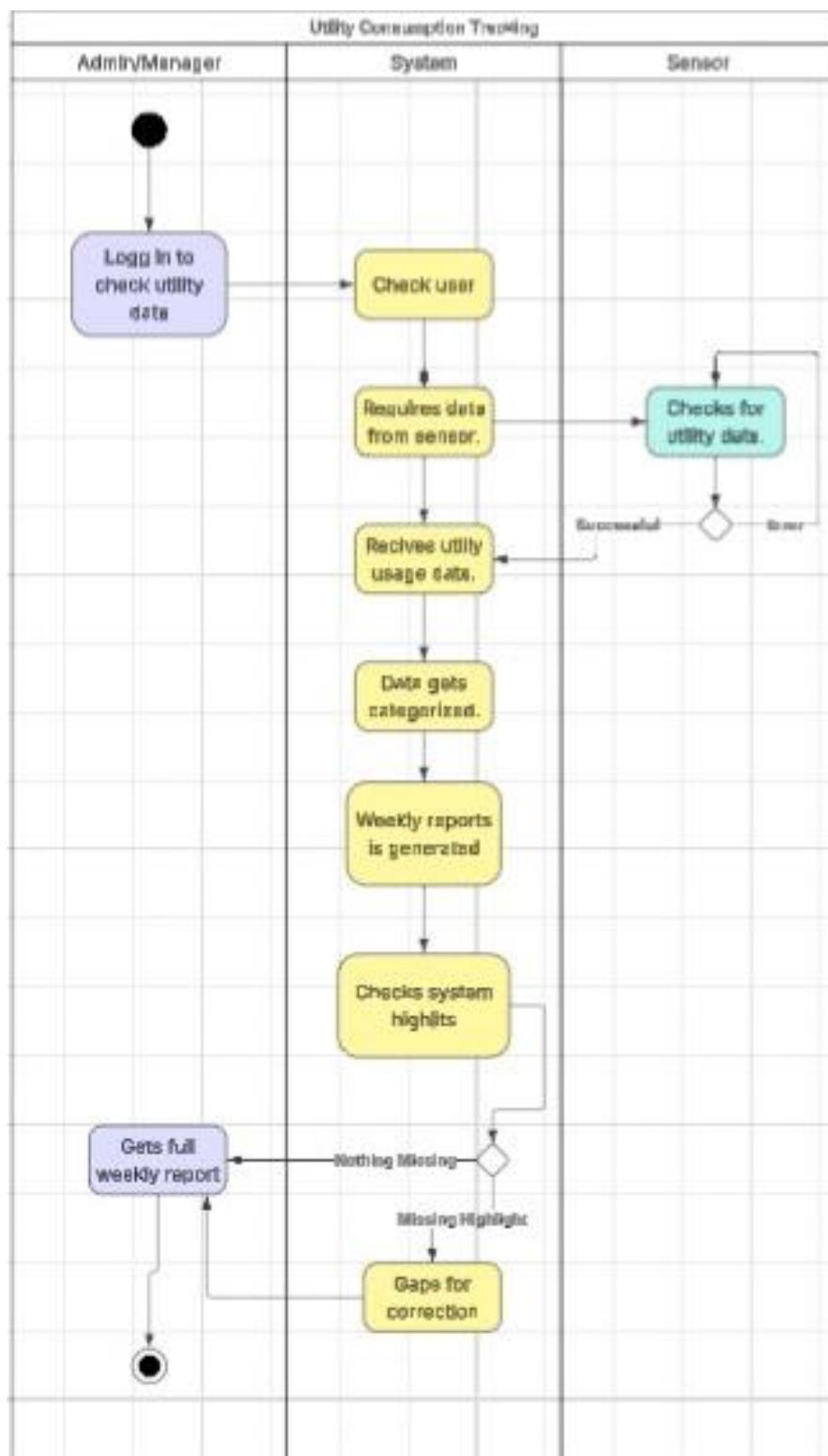
Restaurant Management System Requirements Specification

AC_20: Valet Parking Request Management – Erdi Perhati



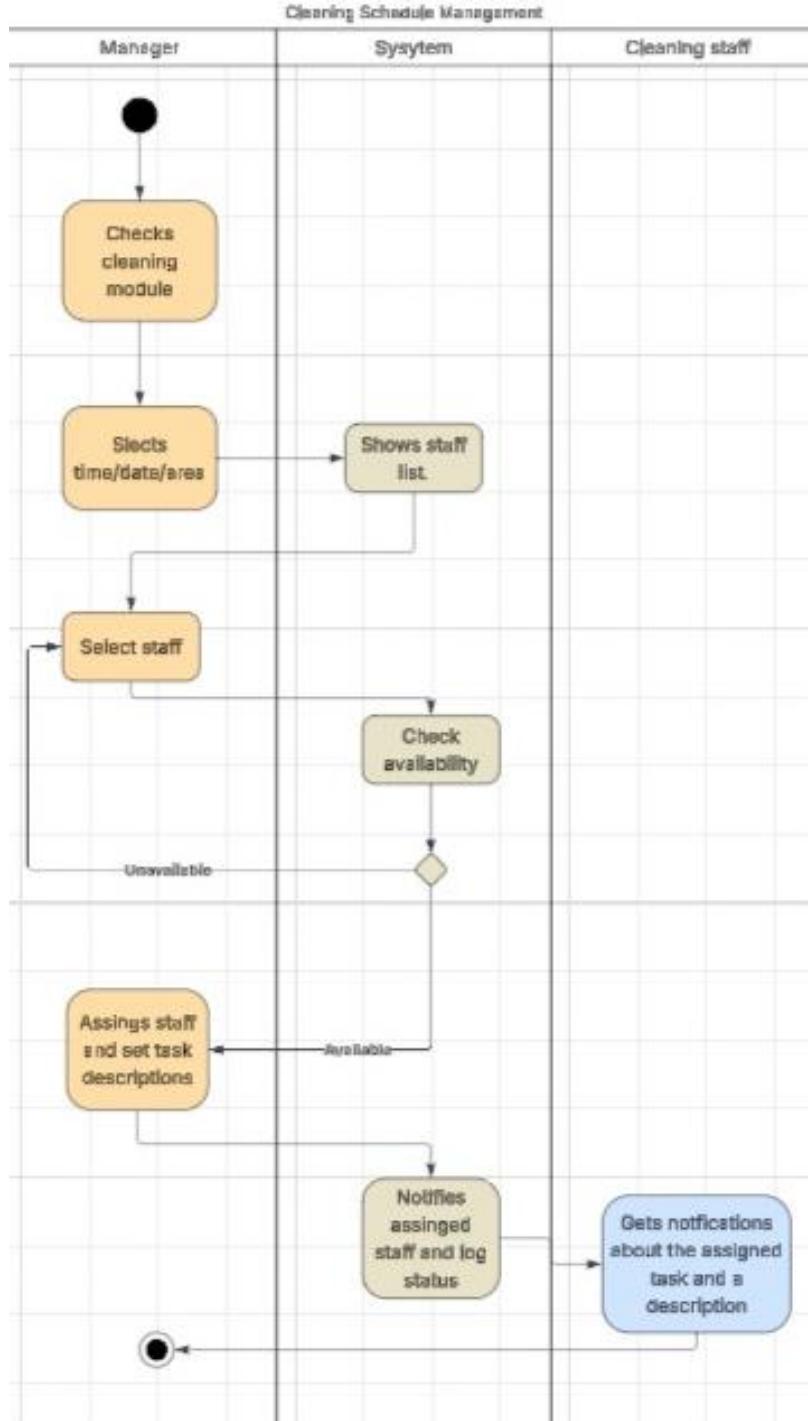
Restaurant Management System Requirements Specification

AC_21: Utility Consumption Tracking – Erdi Perhati

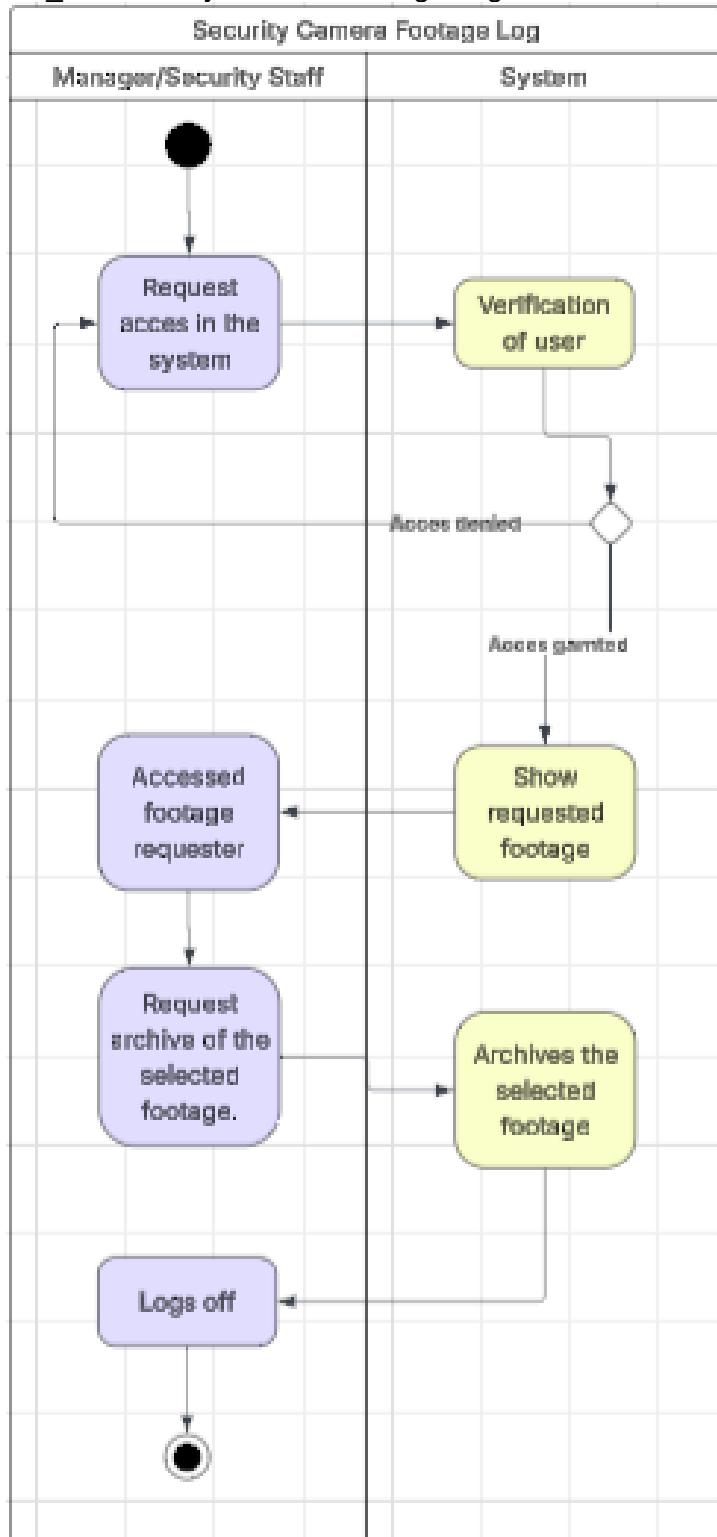


Restaurant Management System Requirements Specification

AC_22: Cleaning Schedule Management – Erdi Perhati

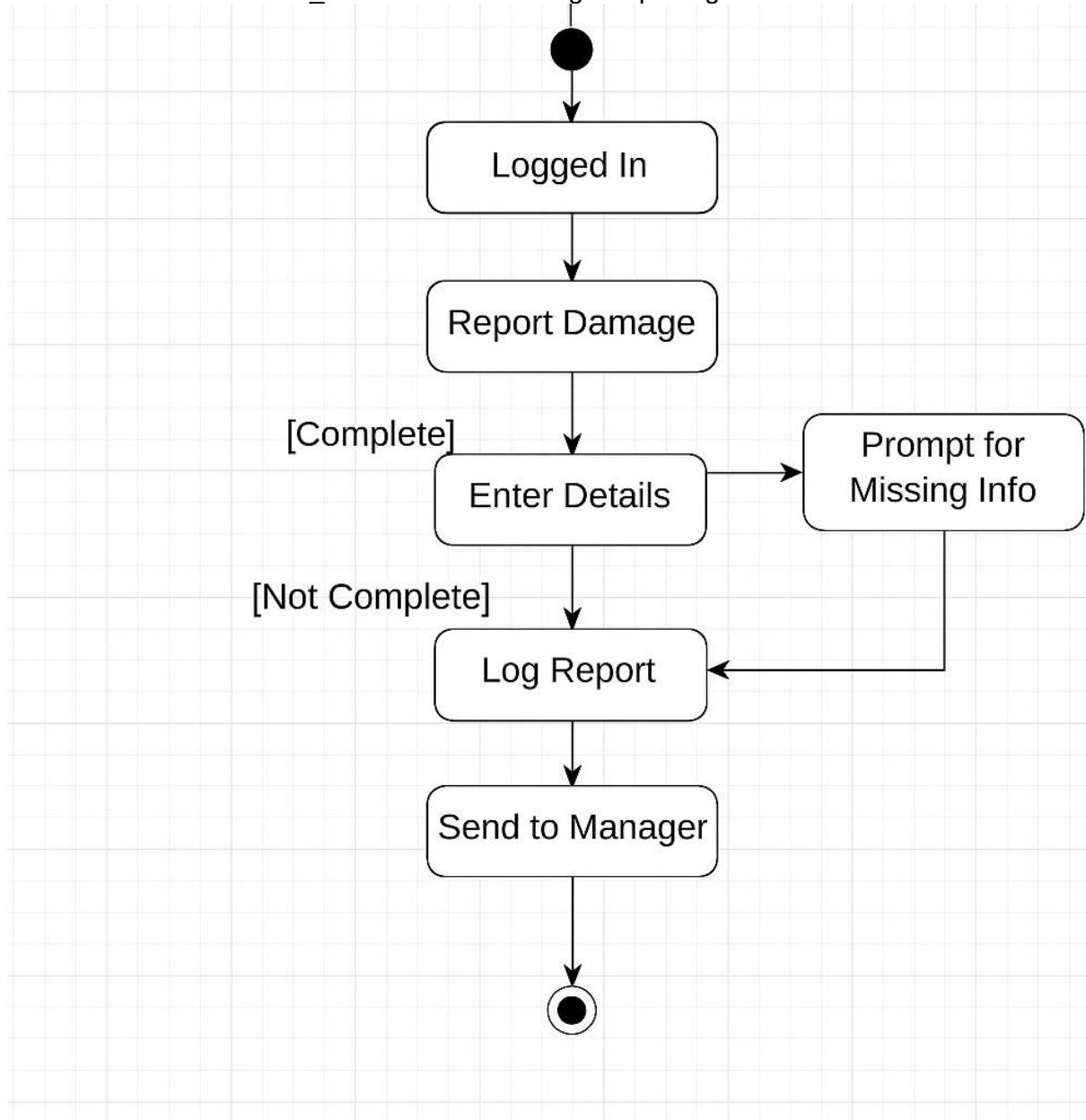


AC_23: Security Camera Footage Log – Erdi Perhati

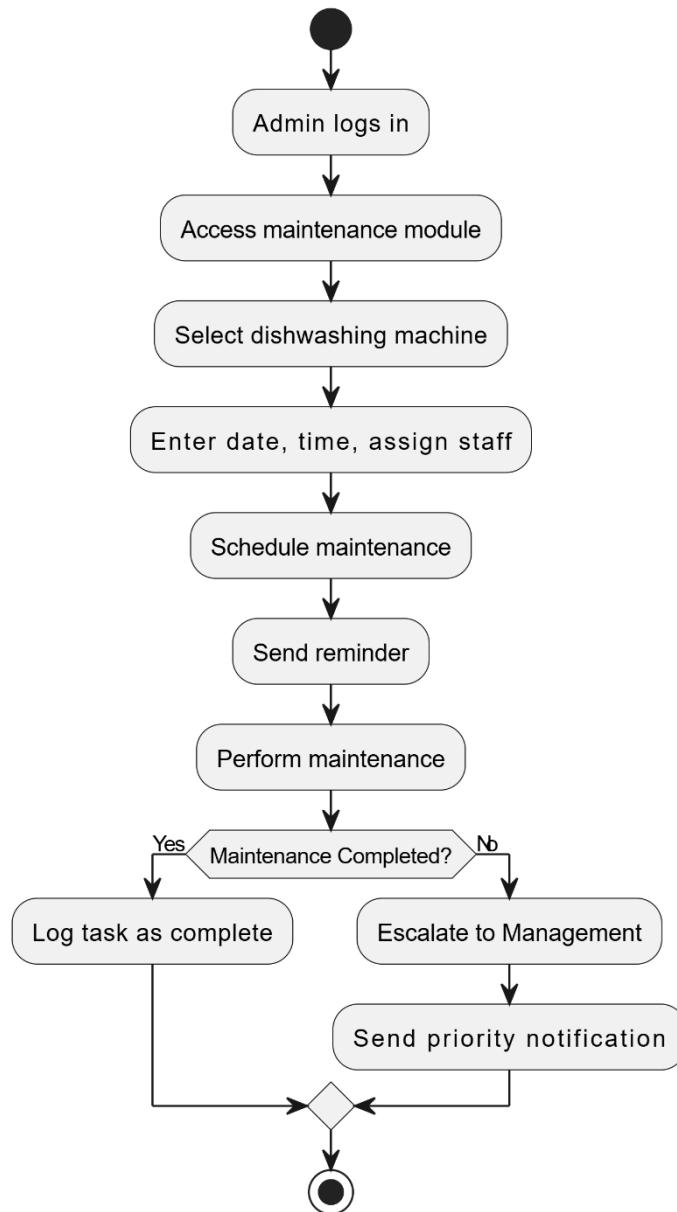


Restaurant Management System Requirements Specification

AC_25: Tableware Damage Reporting – Keisi Loci

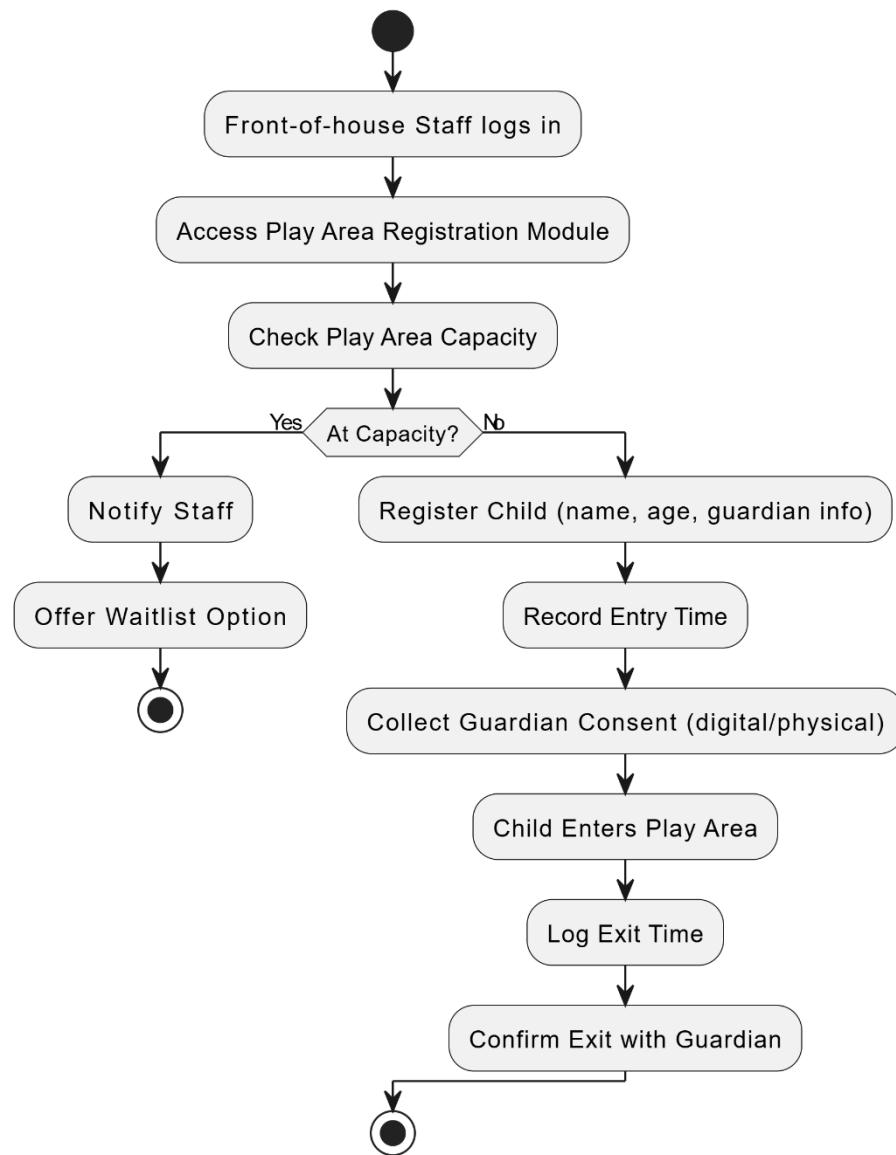


Restaurant Management System Requirements Specification
AC_27: Dishwashing Machine Maintenance Scheduling – Keisi Loci

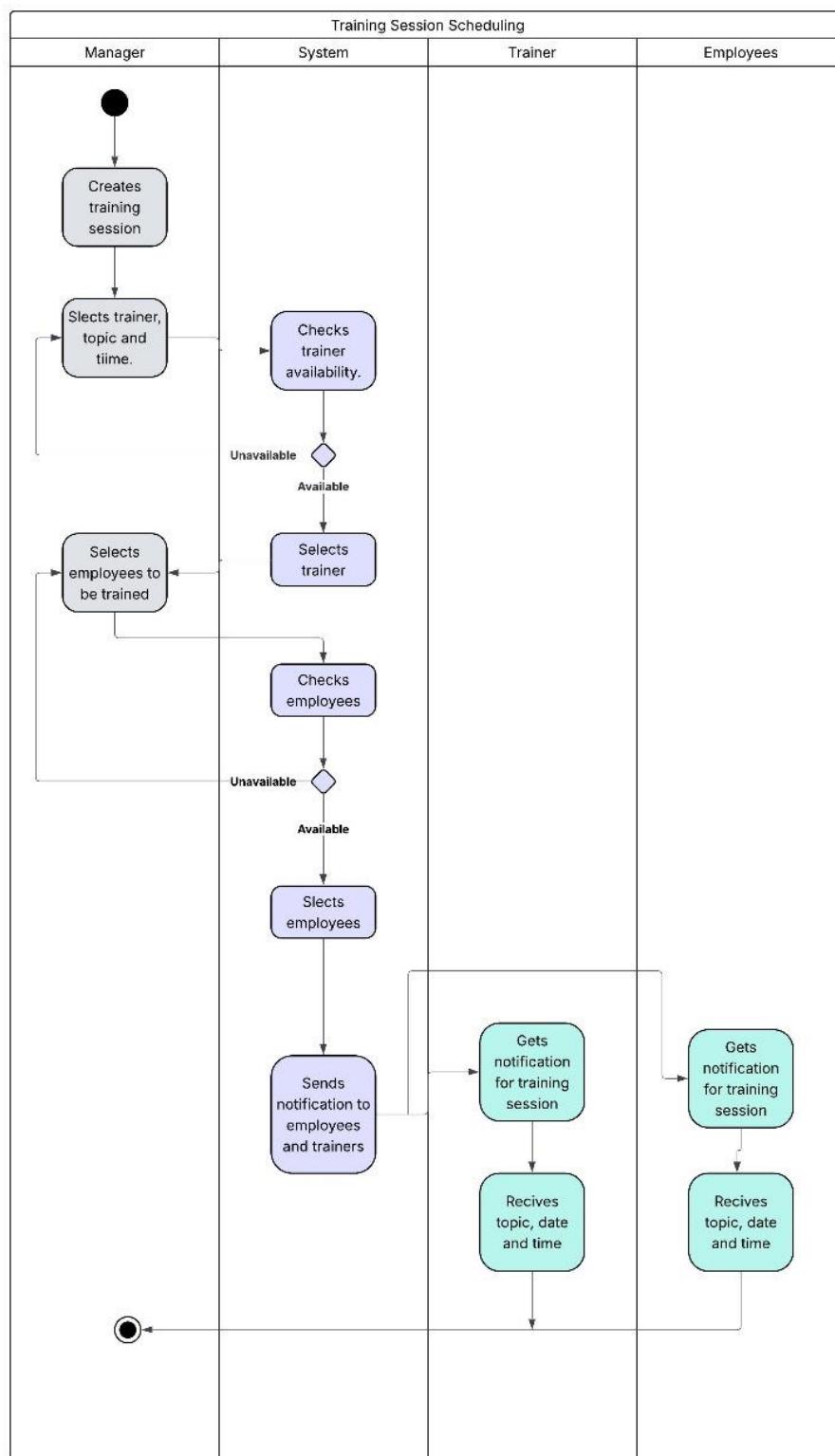


Restaurant Management System Requirements Specification

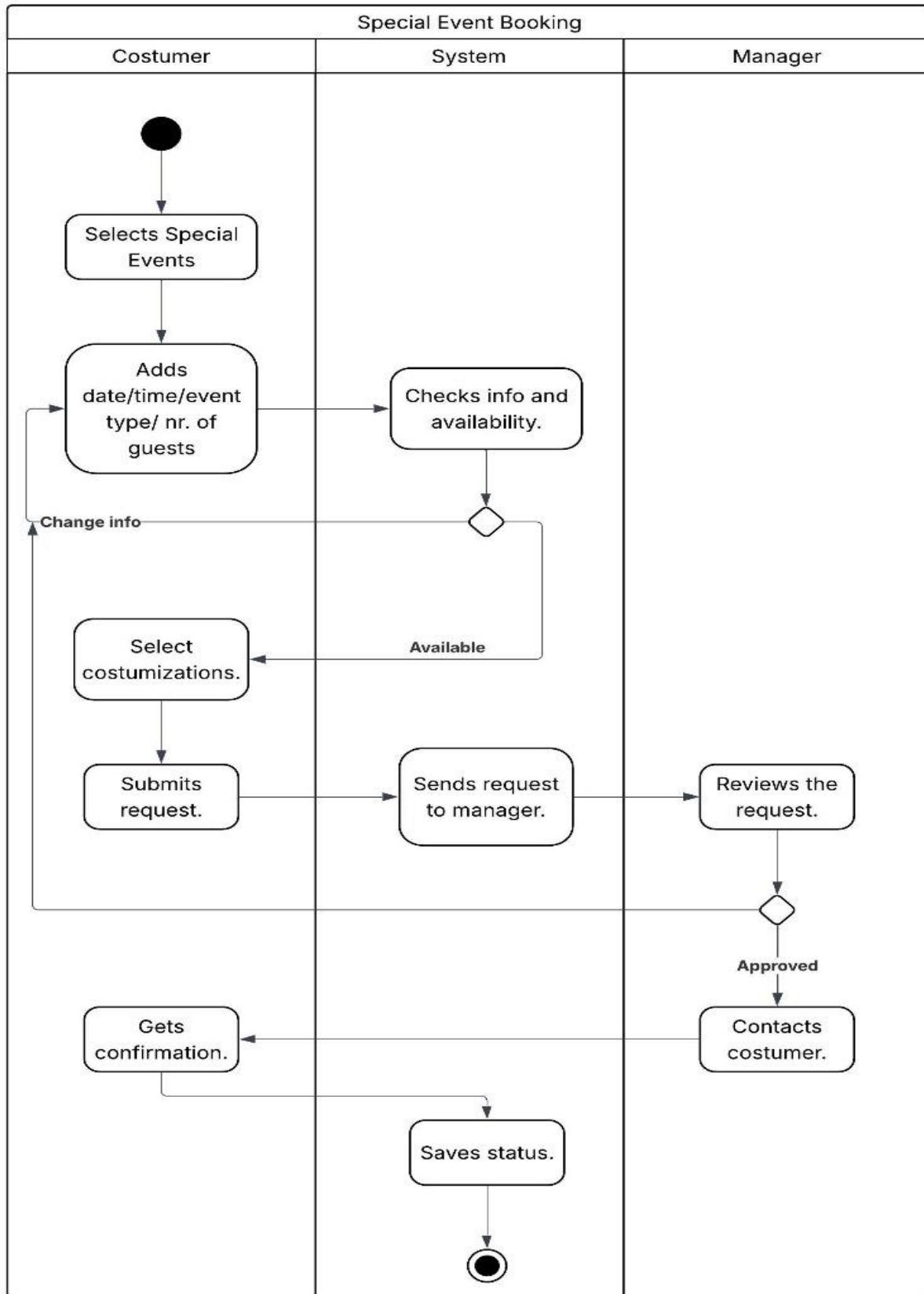
AC_28: Play Area Access Registration – Keisi Loci



Restaurant Management System Requirements Specification
AC_24: Training Session Scheduling – Keisi Loci

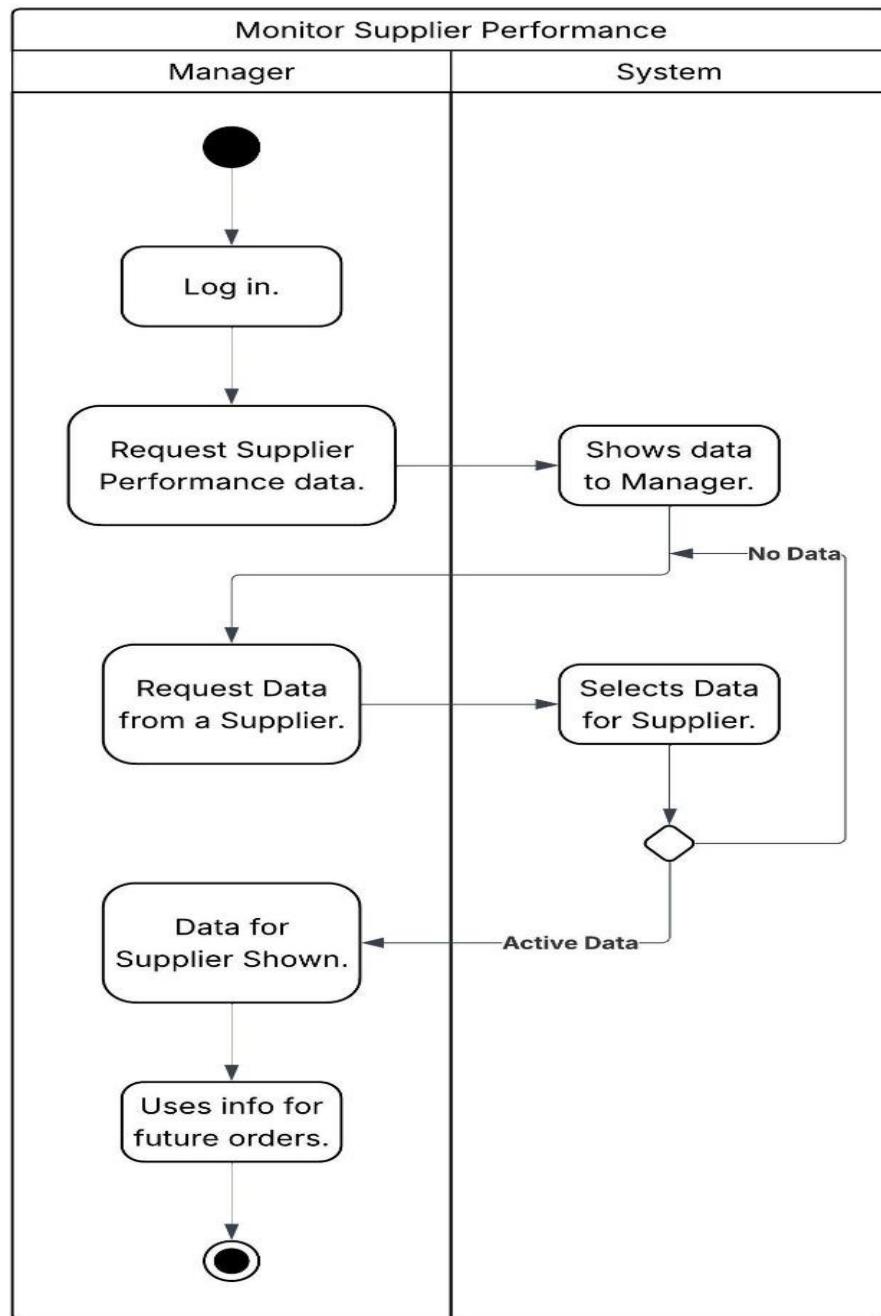


Restaurant Management System Requirements Specification
 AC_30: Manage Dining Time Limits Warning – Arlis Arapi



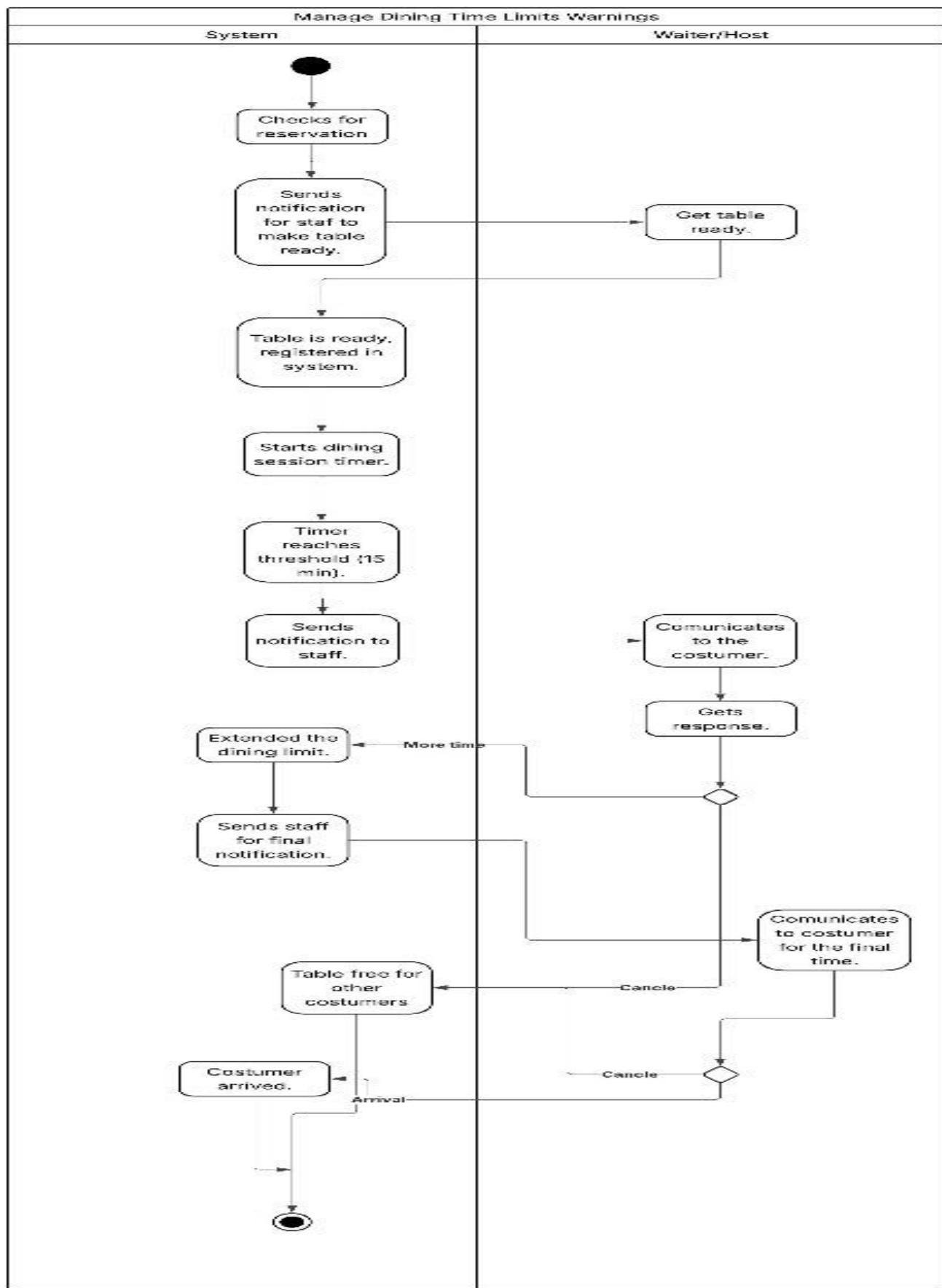
Restaurant Management System Requirements Specification

AC_17: Monitor Supplier Performance – Arlis Arapi



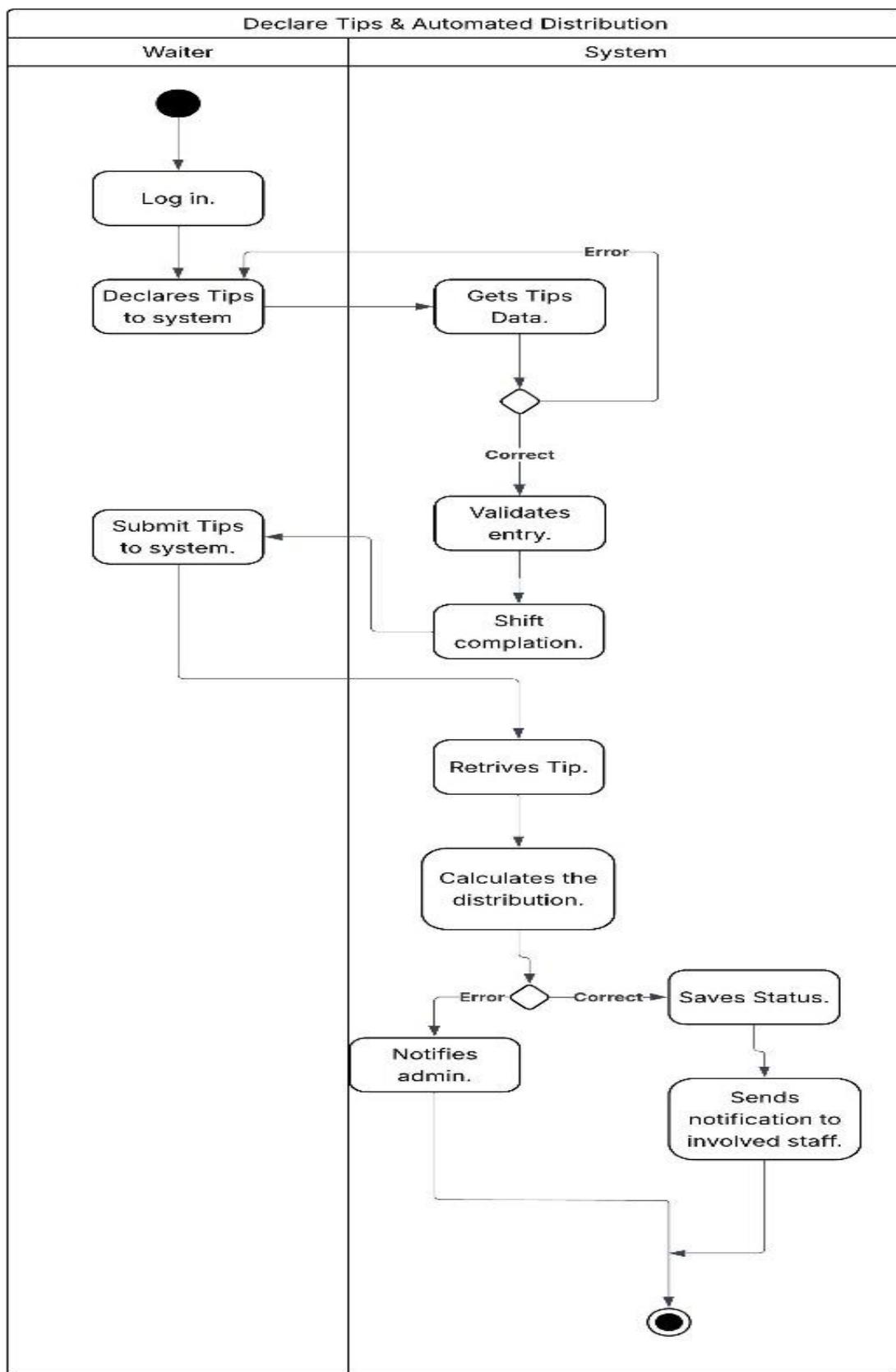
Restaurant Management System Requirements Specification

AC_19: Special Event Booking – Arlis Arapi



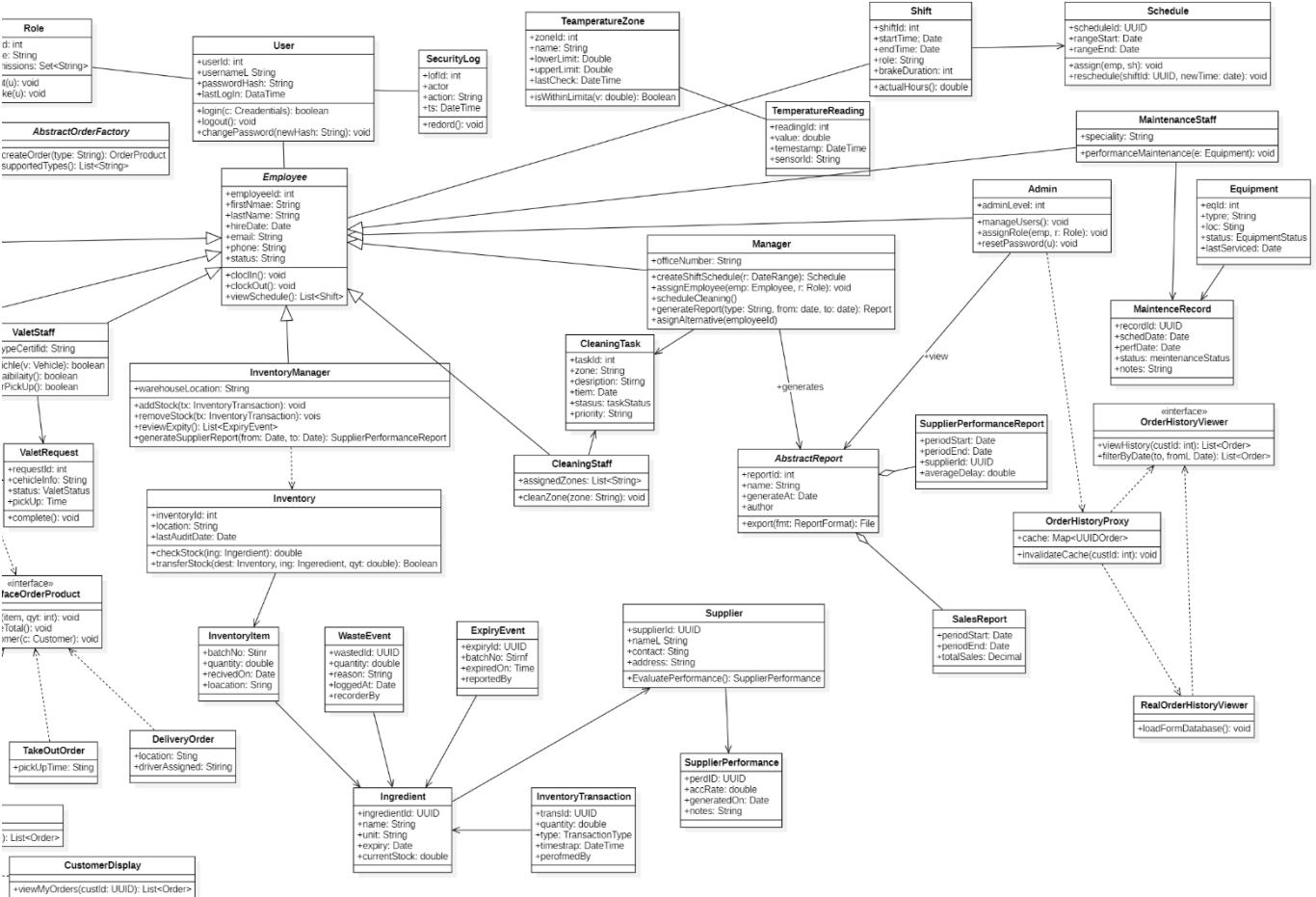
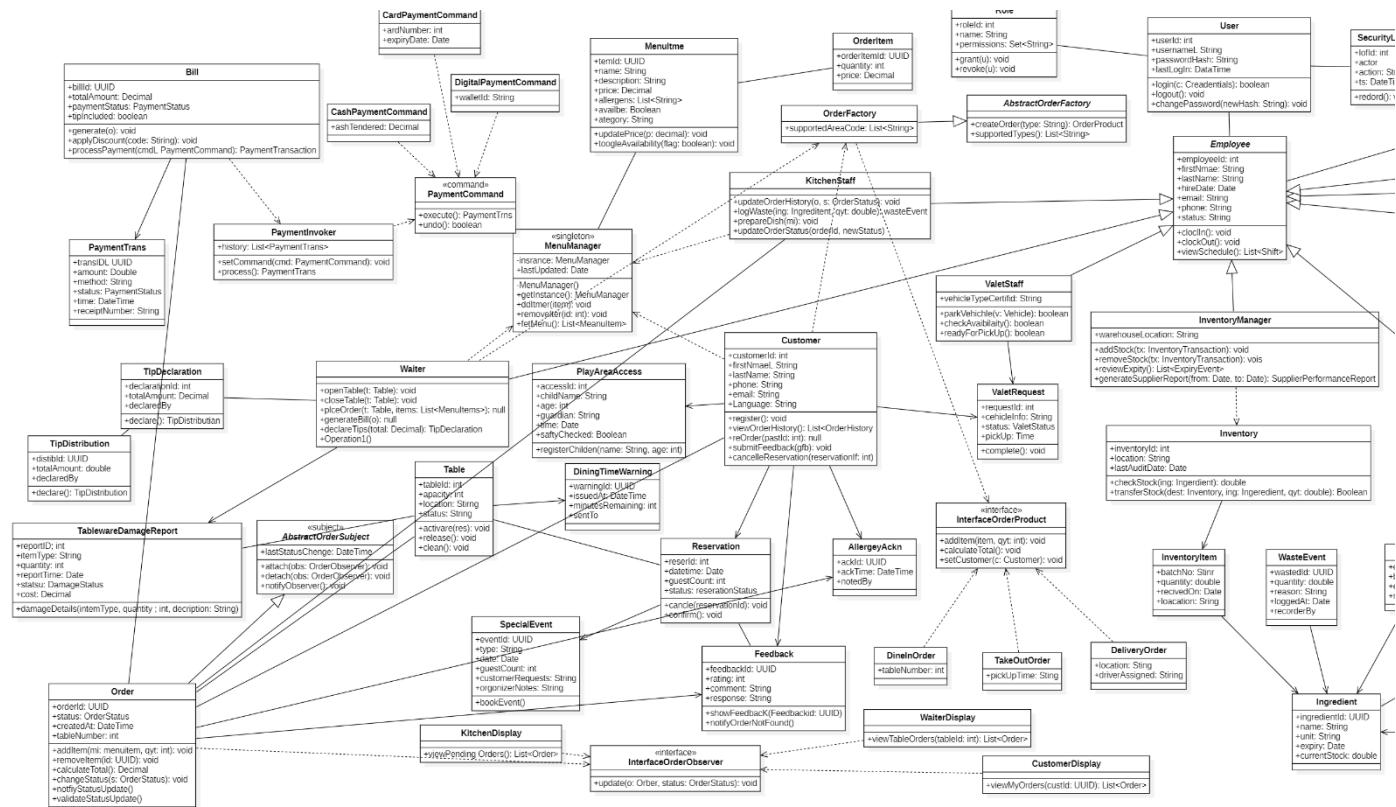
Restaurant Management System Requirements Specification

AC_18: Declare Tips & Automated Distribution – Arlis Arapi

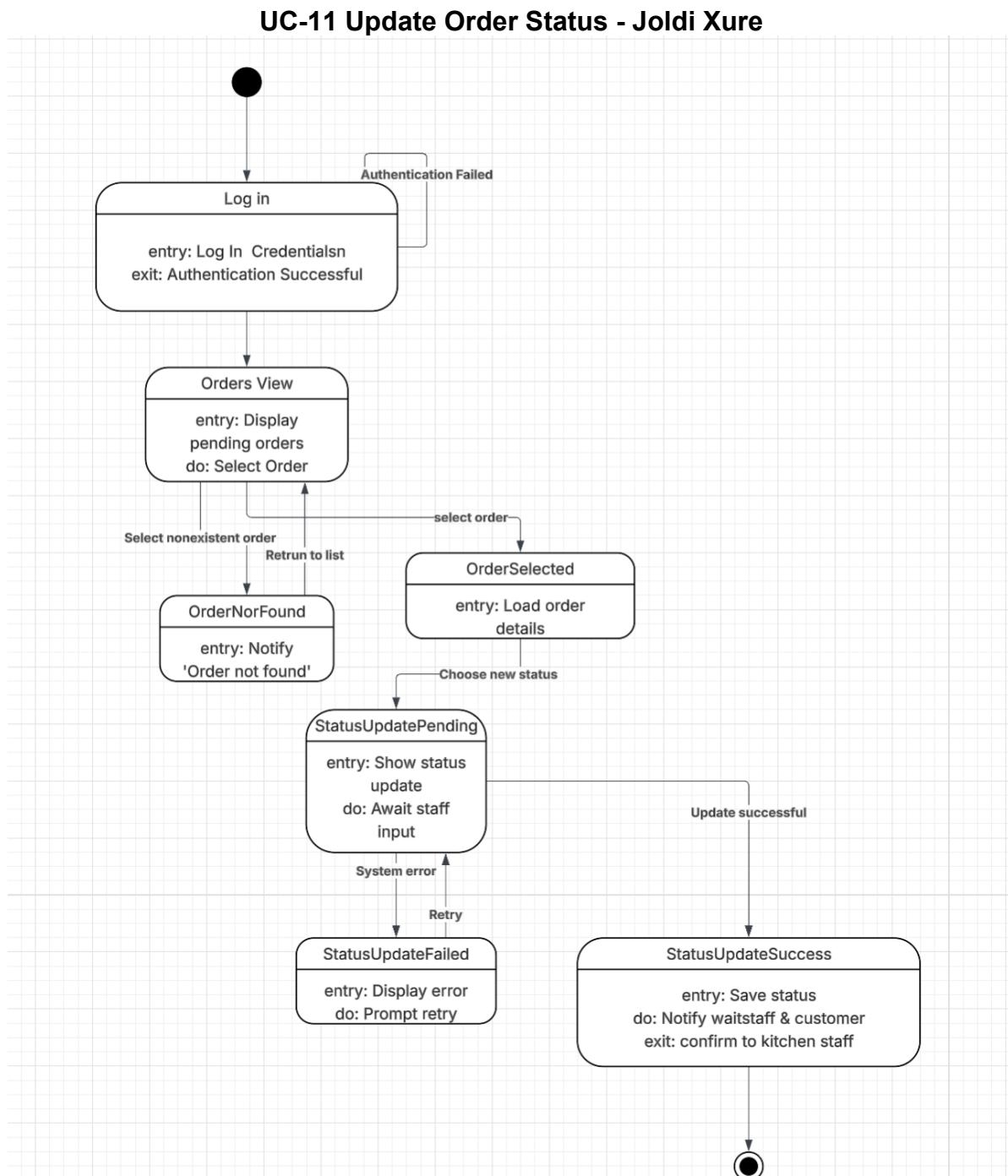


Restaurant Management System Requirements Specification

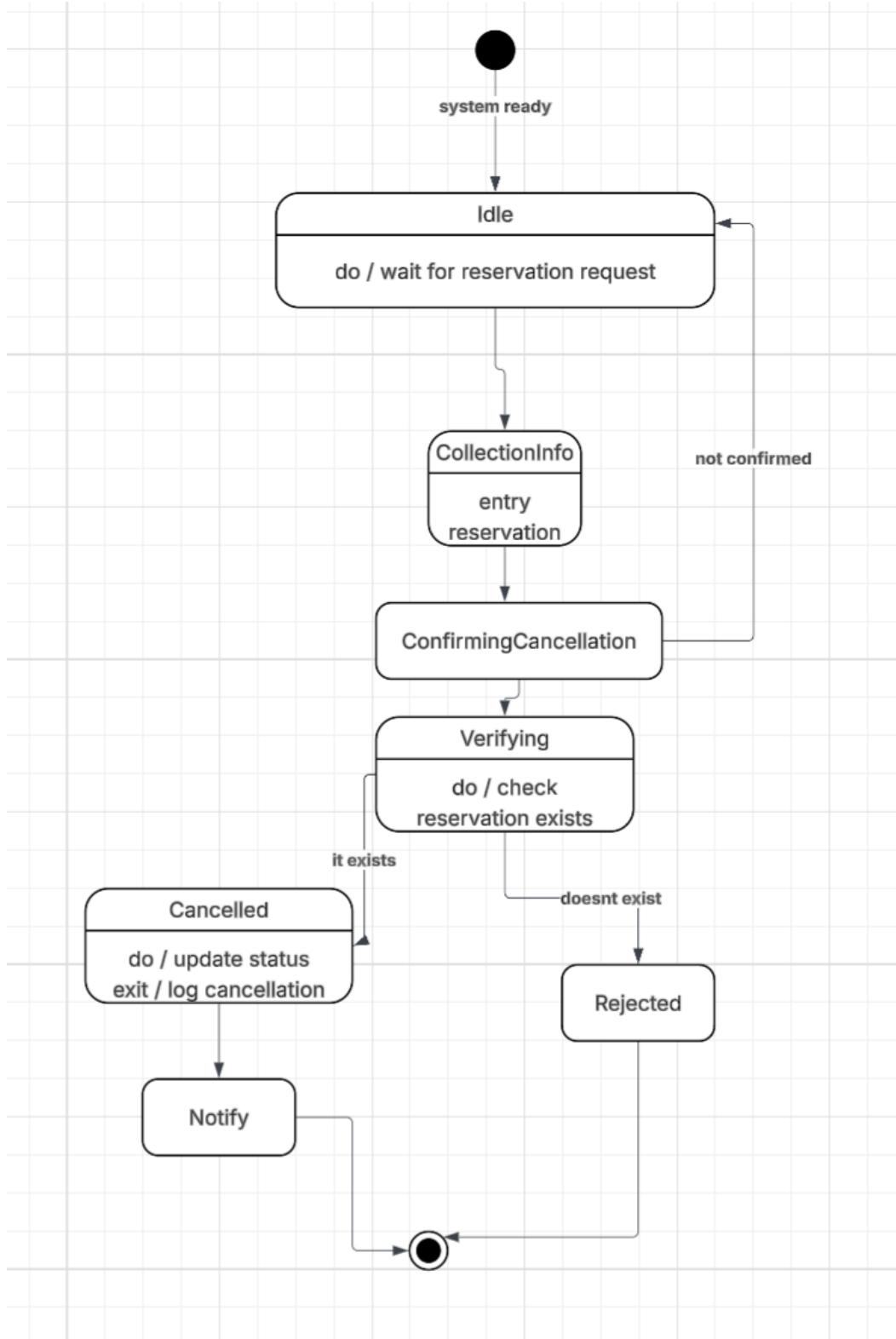
5.4 Class Diagram



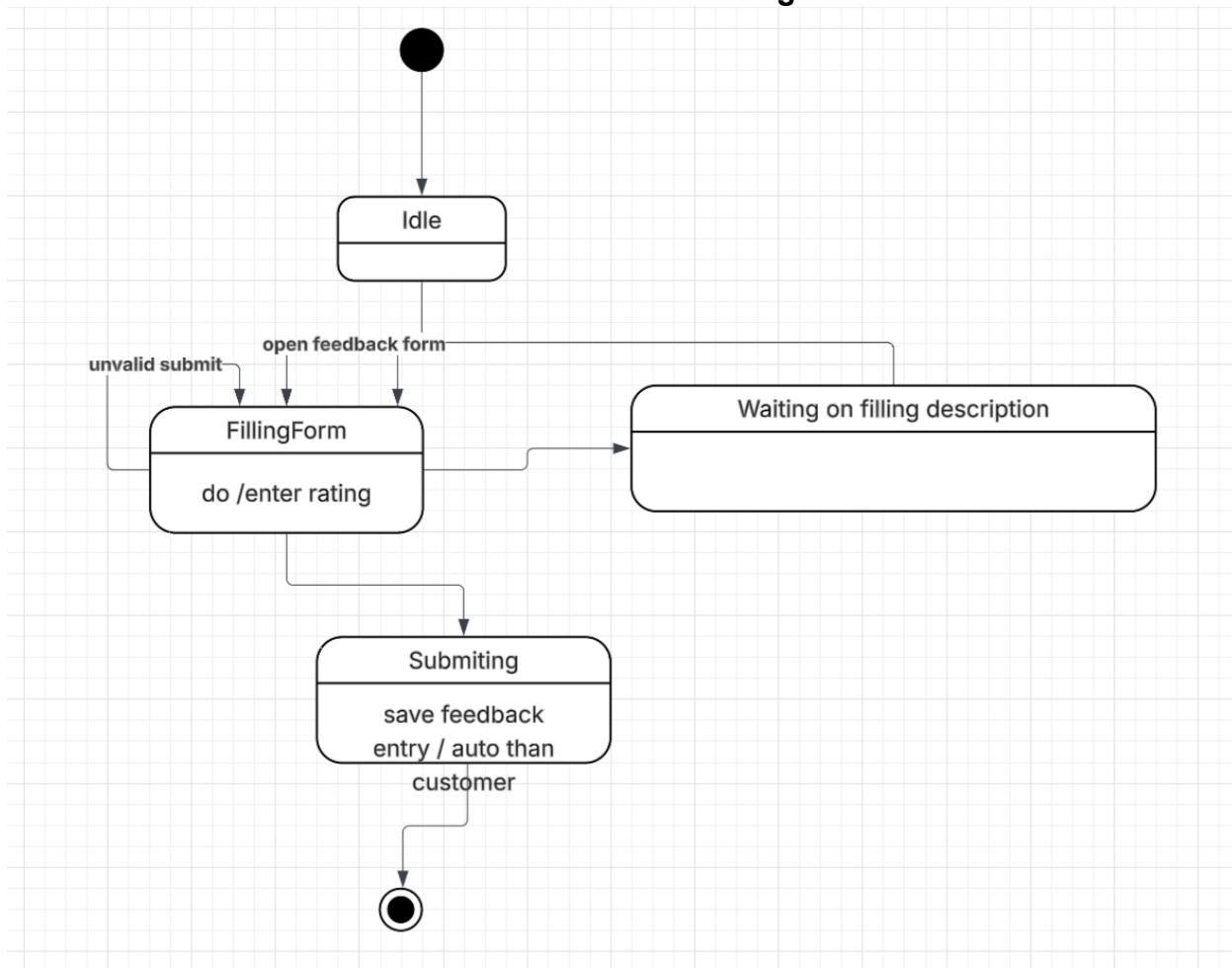
5.5 State Diagram



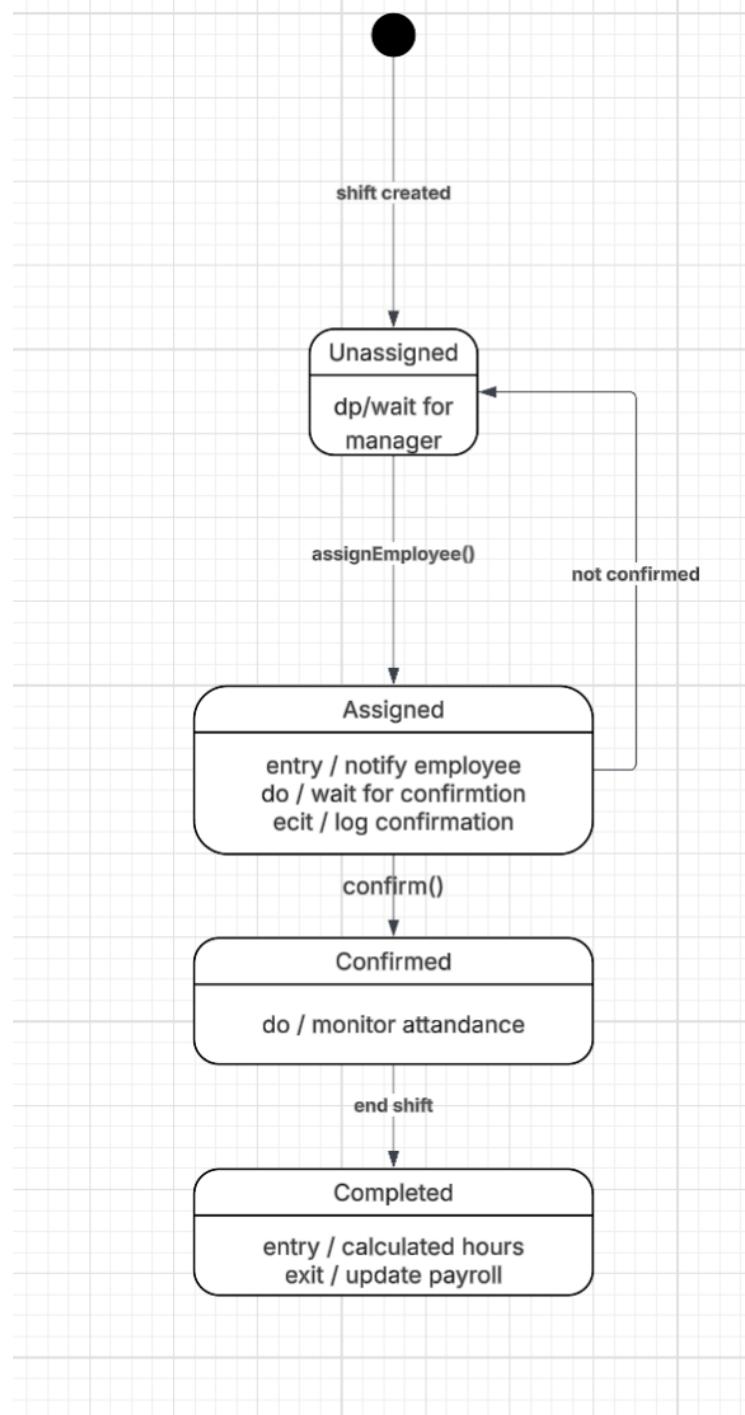
UC-10 Table Cancellation - Joldi Xure



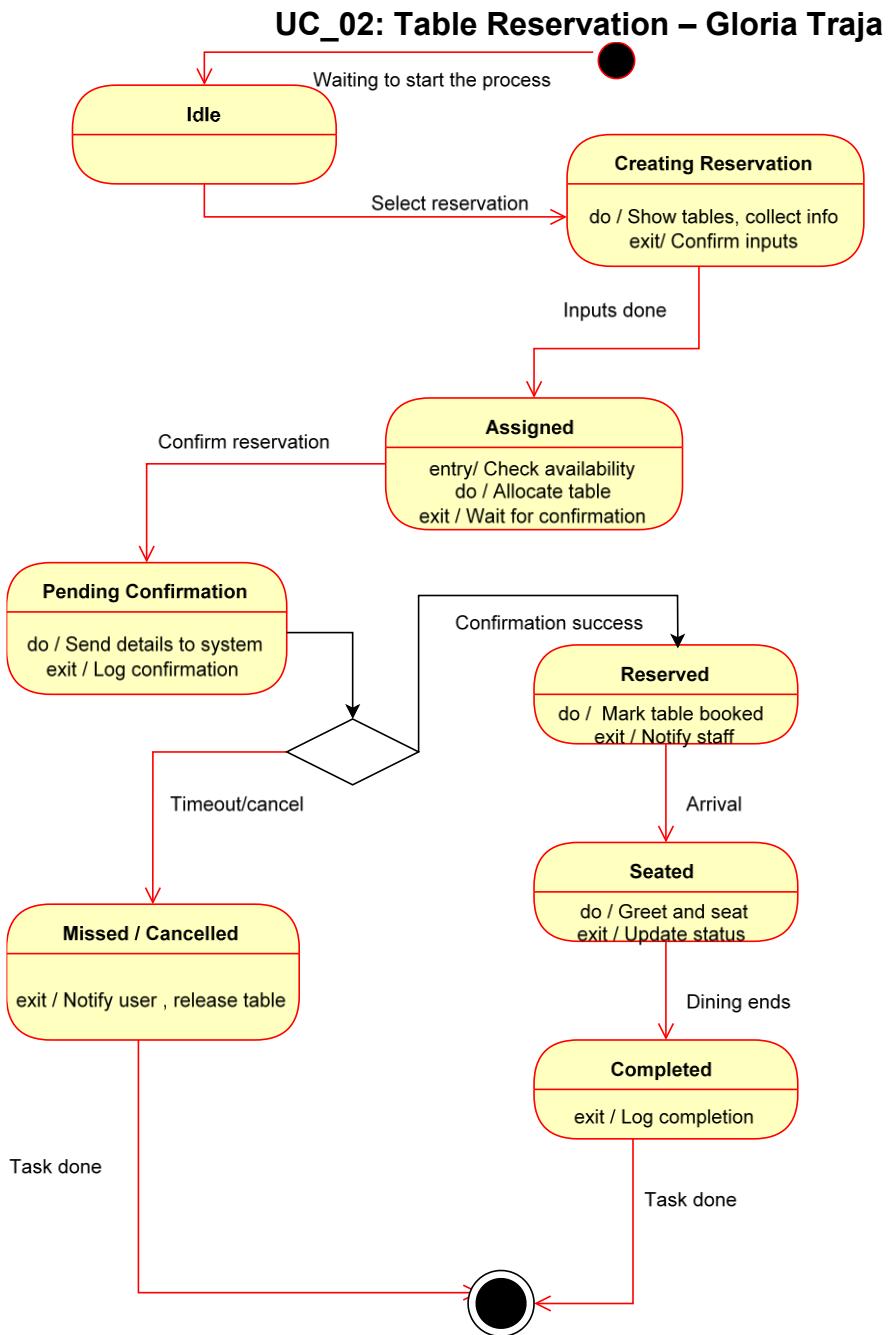
UC-12 Customer Feedback & Ratings - Joldi Xure



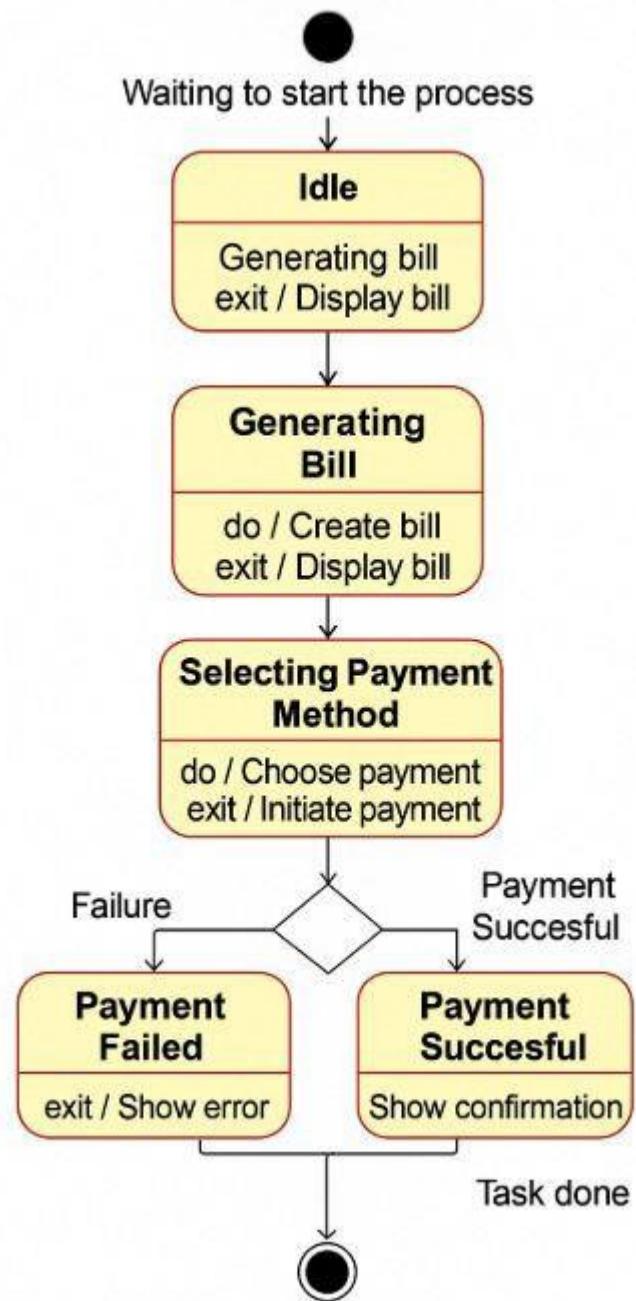
UC-13 Shift Scheduling for Staff - Joldi Xure



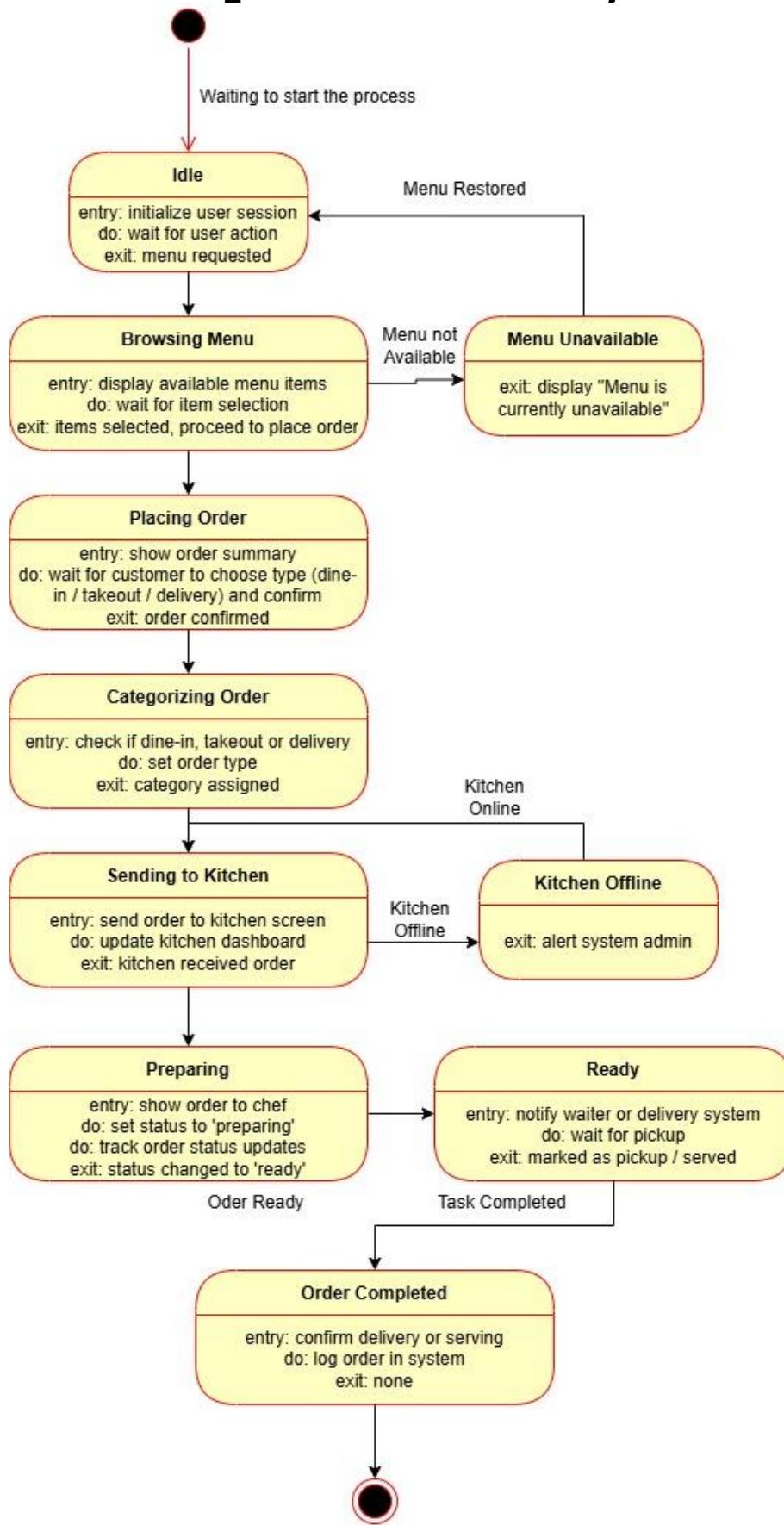
Restaurant Management System Requirements Specification



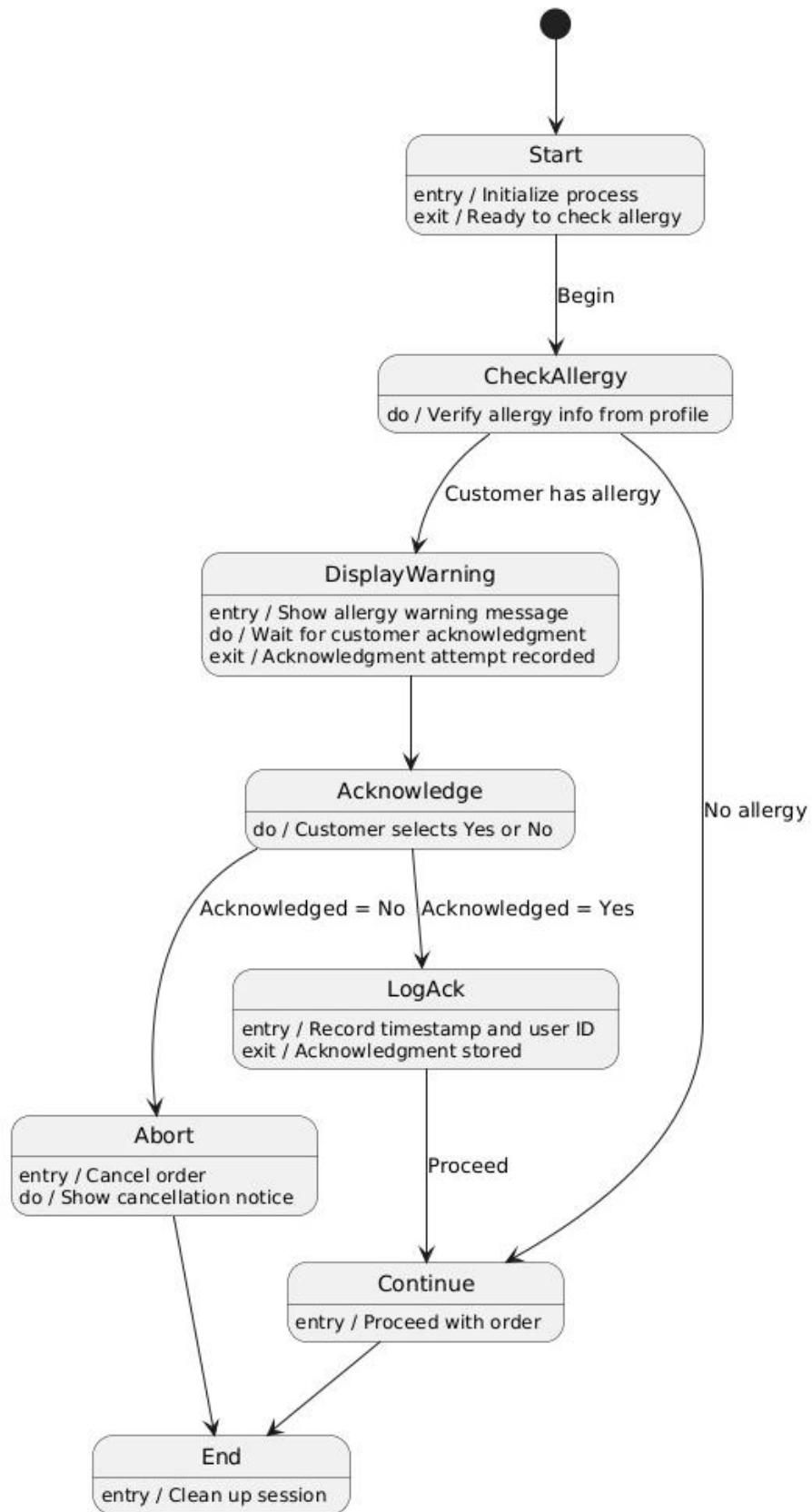
Restaurant Management System Requirements Specification
UC_03: Bill Generation & Payment Processing – Gloria Traja



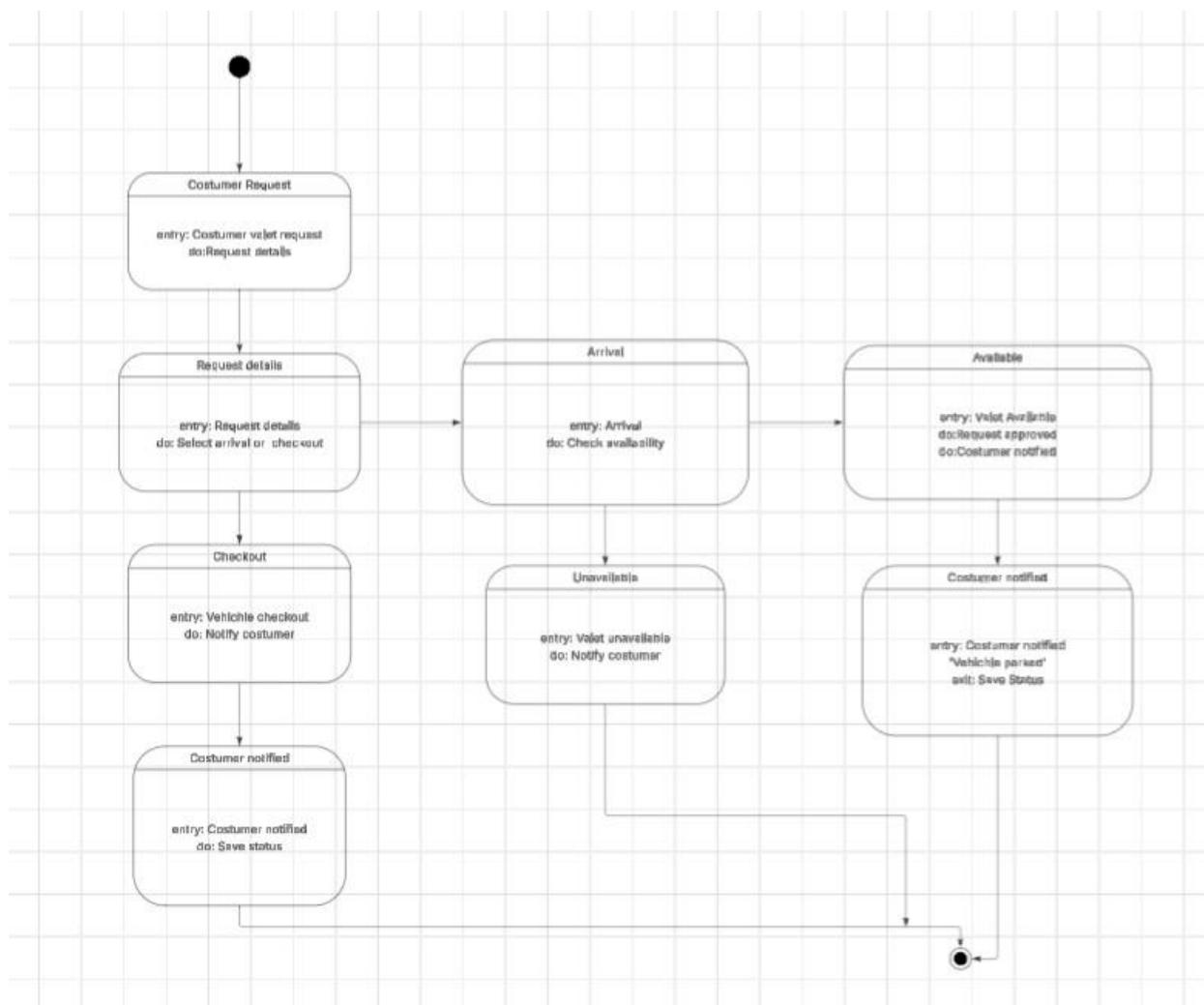
UC_06: Place Order – Alisa Tozaj



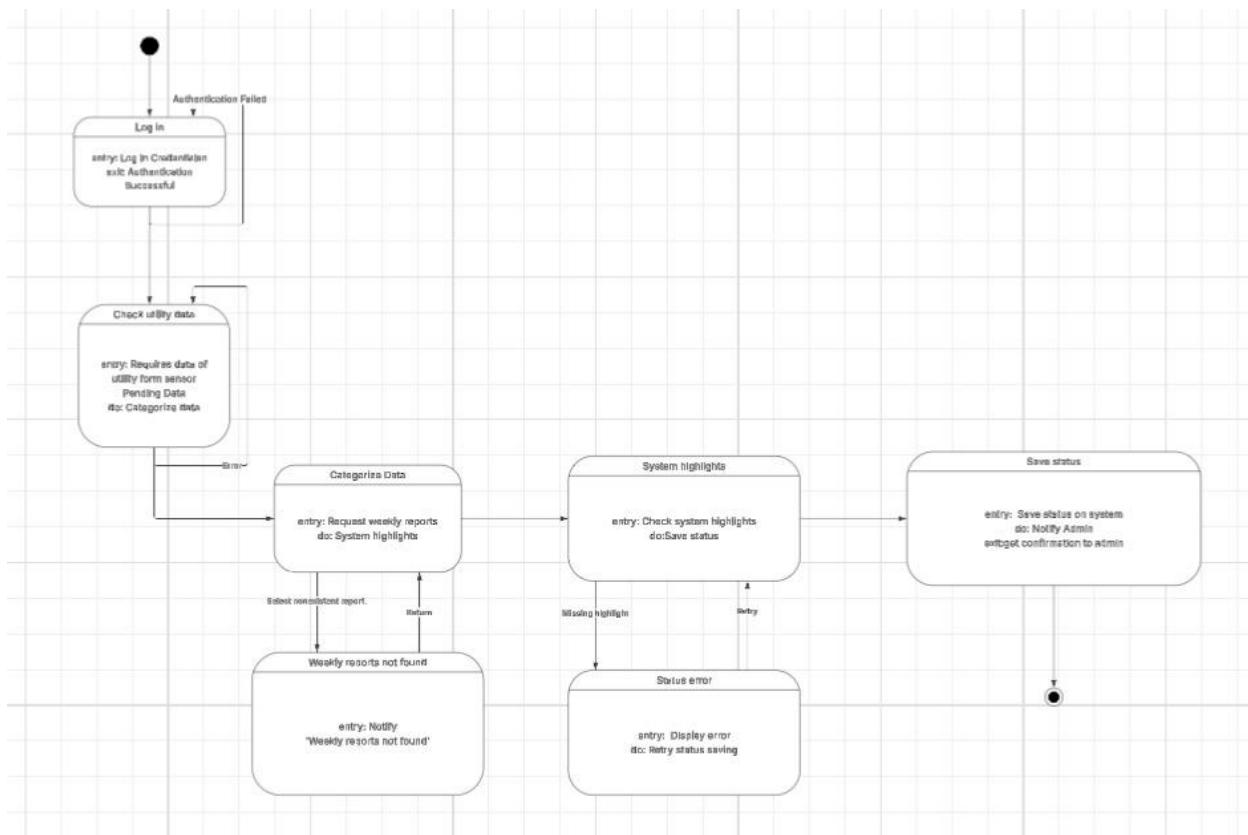
UC_29: Customer Allergy Warning – Ester Qershori



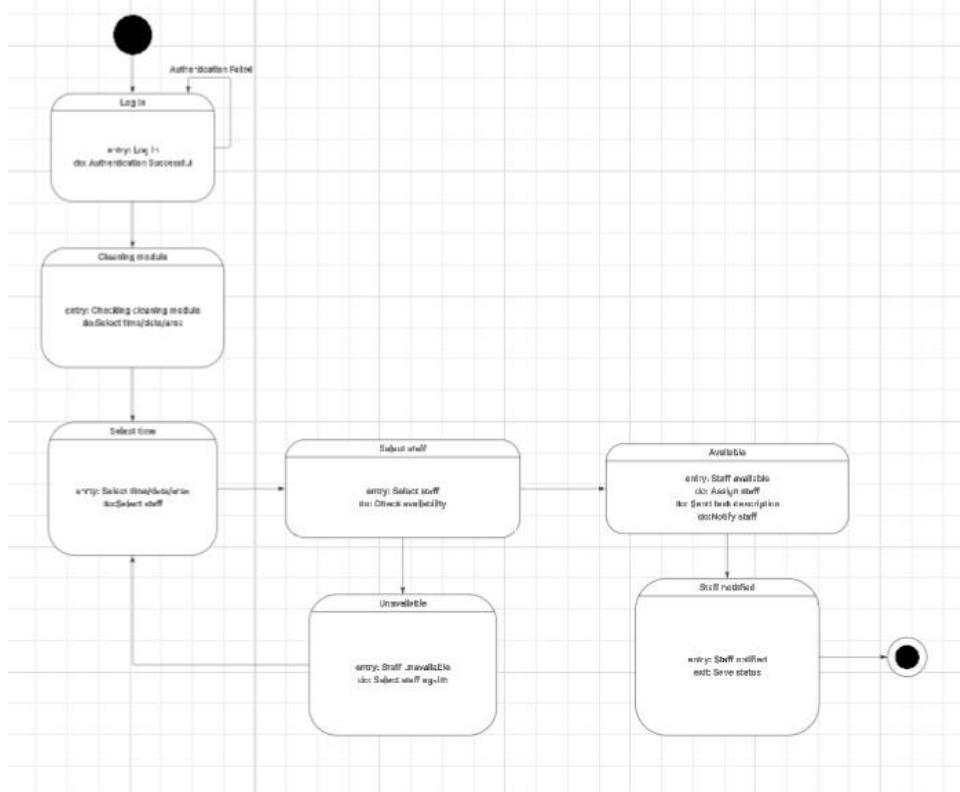
UC_20 - Valet Parking Request Management – Erdi Perhati



Restaurant Management System Requirements Specification
UC21 - Utility Consumption Tracking – Erdi Perhati

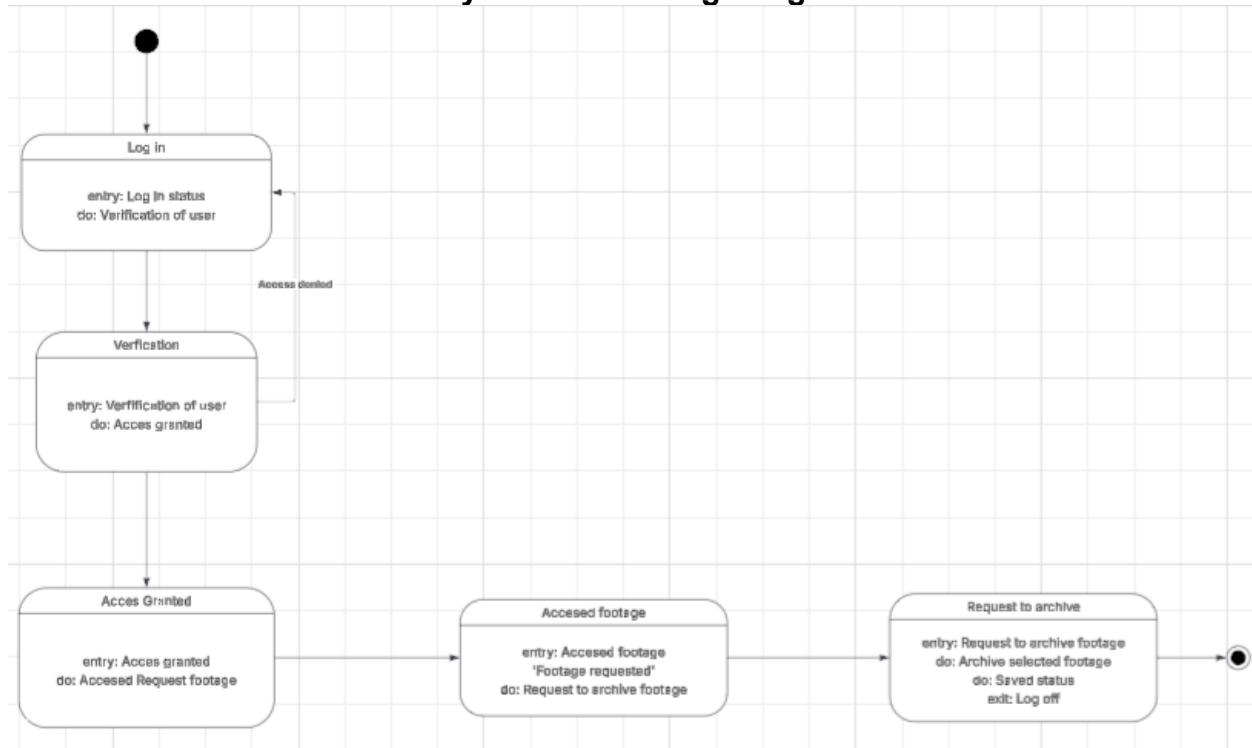


UC22 - Cleaning Schedule Management – Erdi Perhati

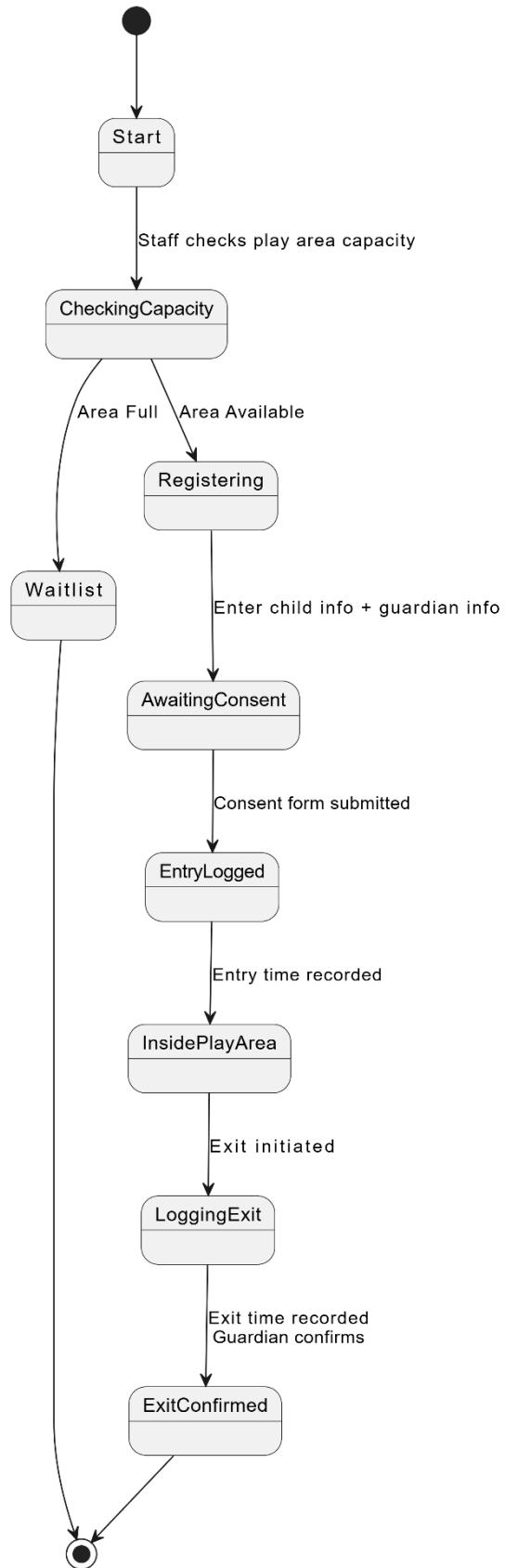


Restaurant Management System Requirements Specification

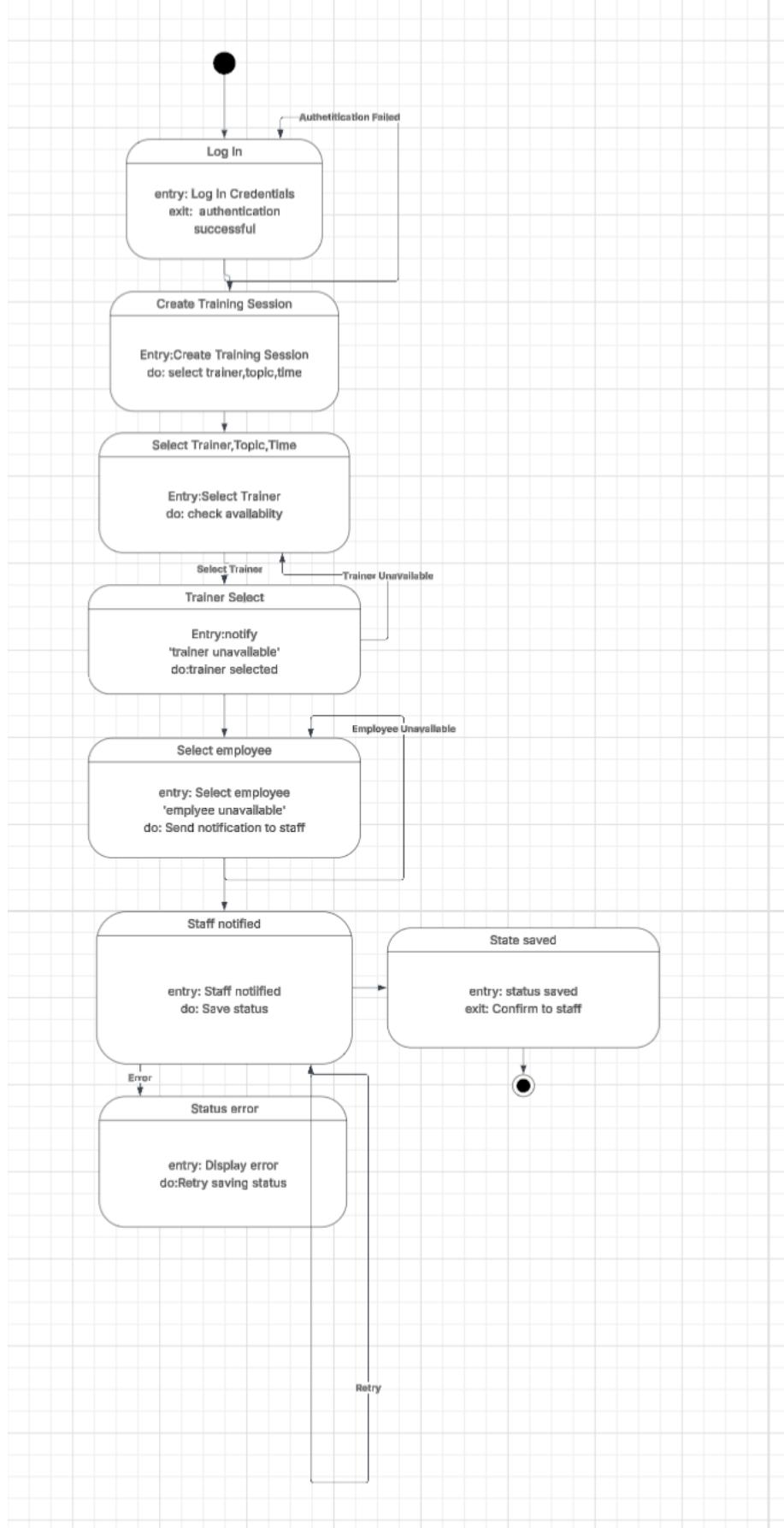
UC23 - Security Camera Footage Log – Erdi Perhati



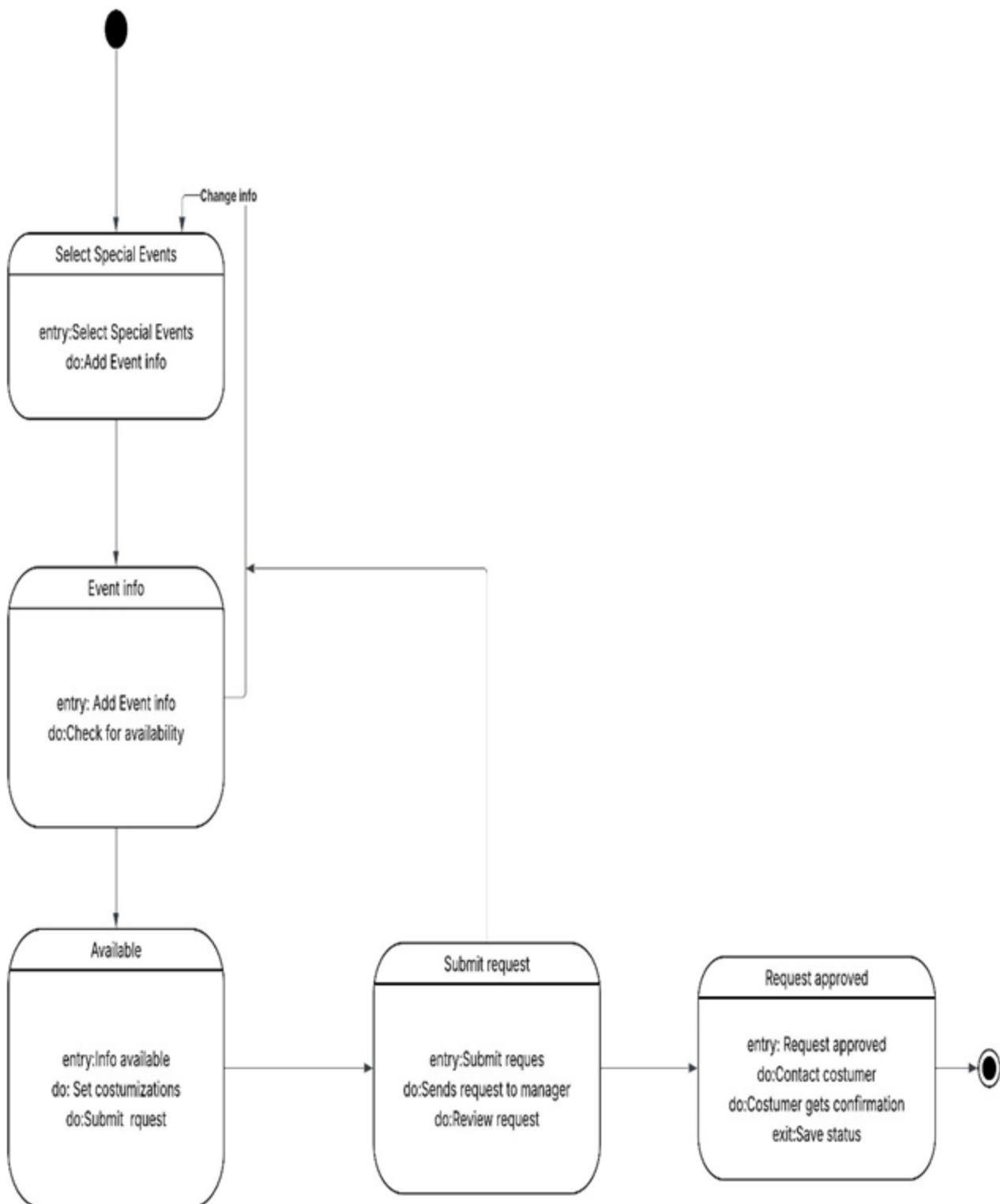
Restaurant Management System Requirements Specification
UC28 – Play Area Access Registration – Keisi Loci



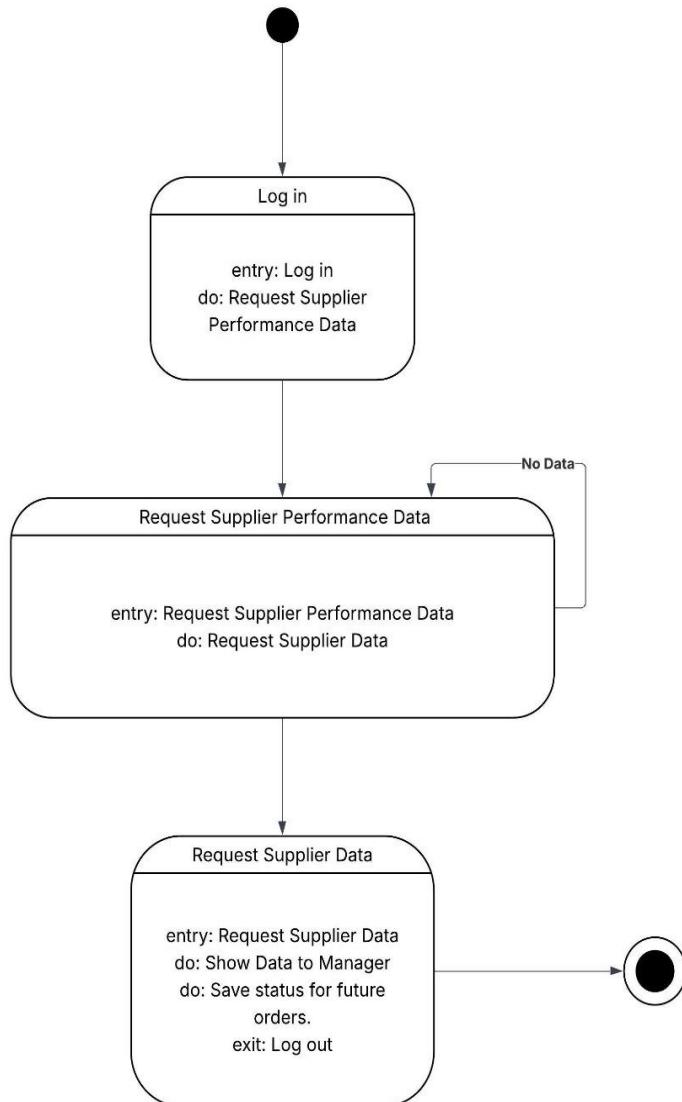
Restaurant Management System Requirements Specification
UC24 – Training Session Scheduling – Keisi Loci



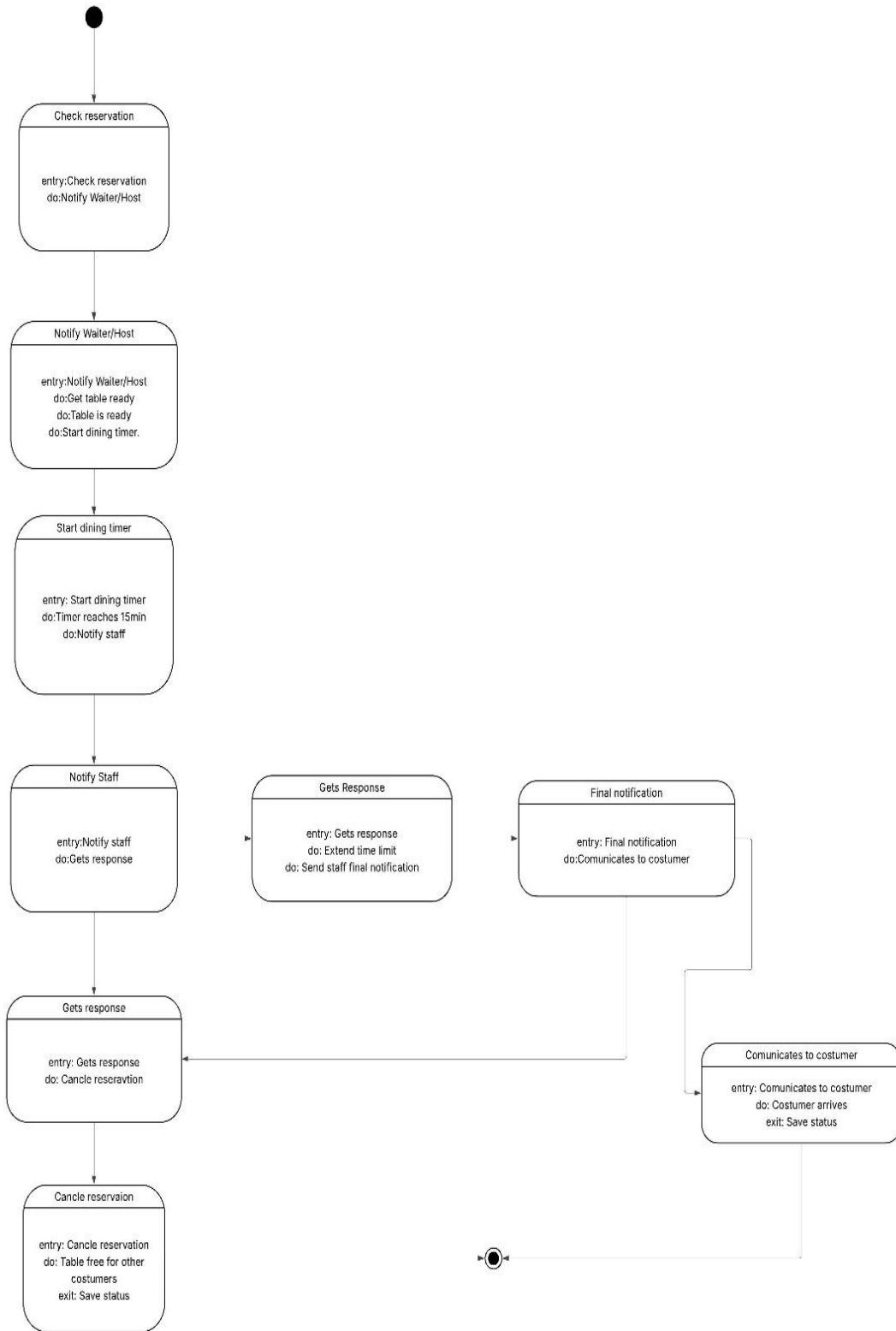
UC30 – Manage Dining Time Limits Warnings – Arlis Arapi



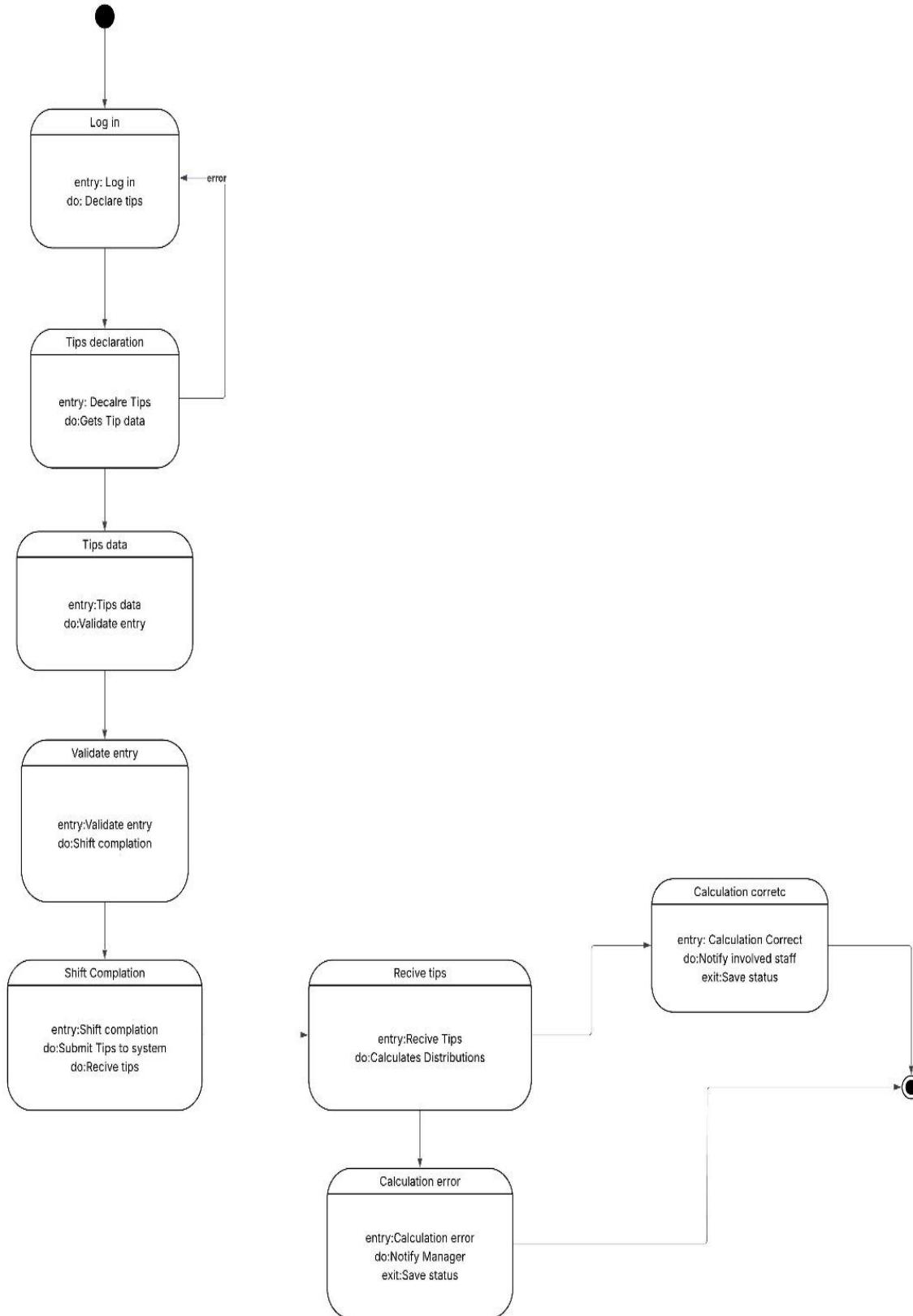
Restaurant Management System Requirements Specification
UC17 – Monitor Supplier Performance – Arlis Arapi



Restaurant Management System Requirements Specification
UC19 – Special Event Booking – Arlis Arapi

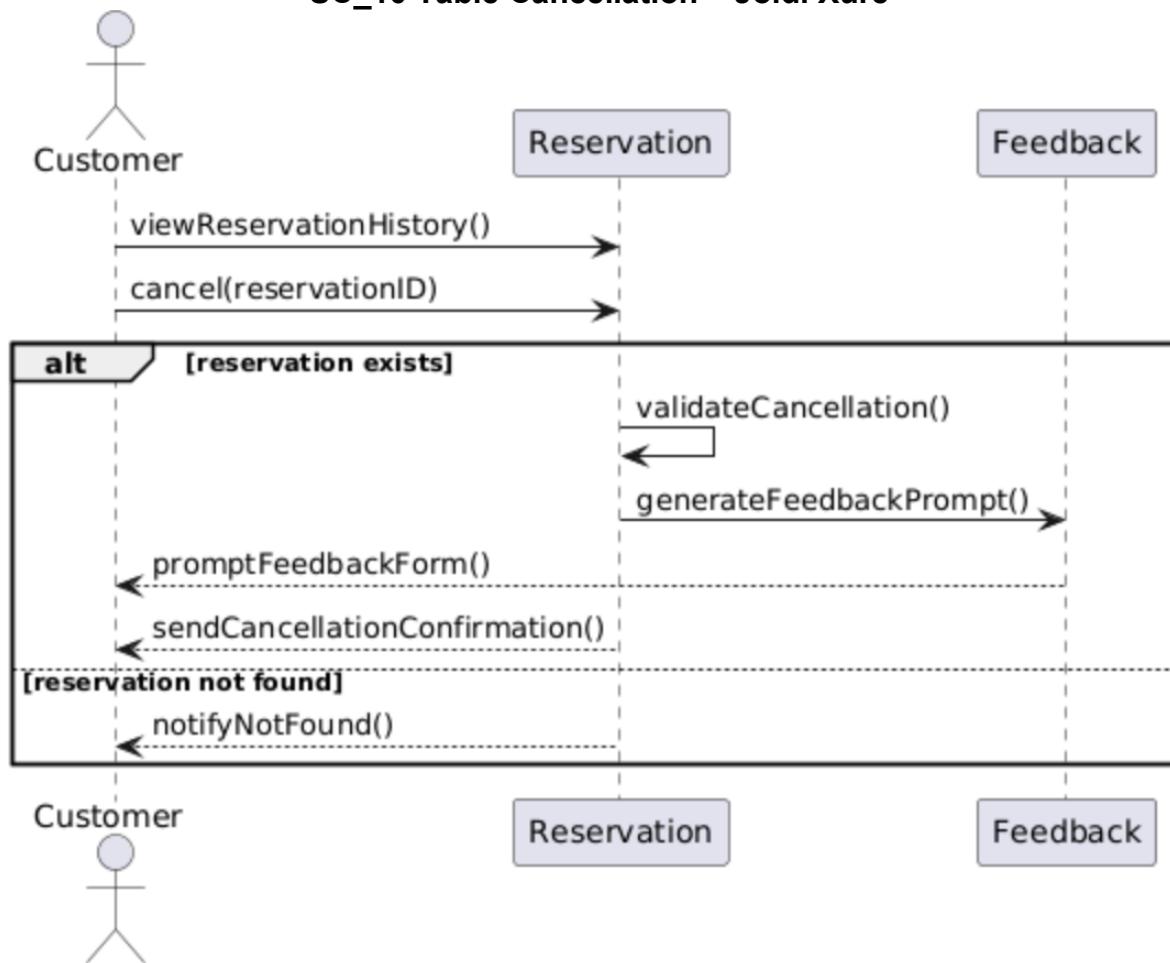


Restaurant Management System Requirements Specification
UC18 – Declare Tips & Automated Distribution – Arlis Arapi

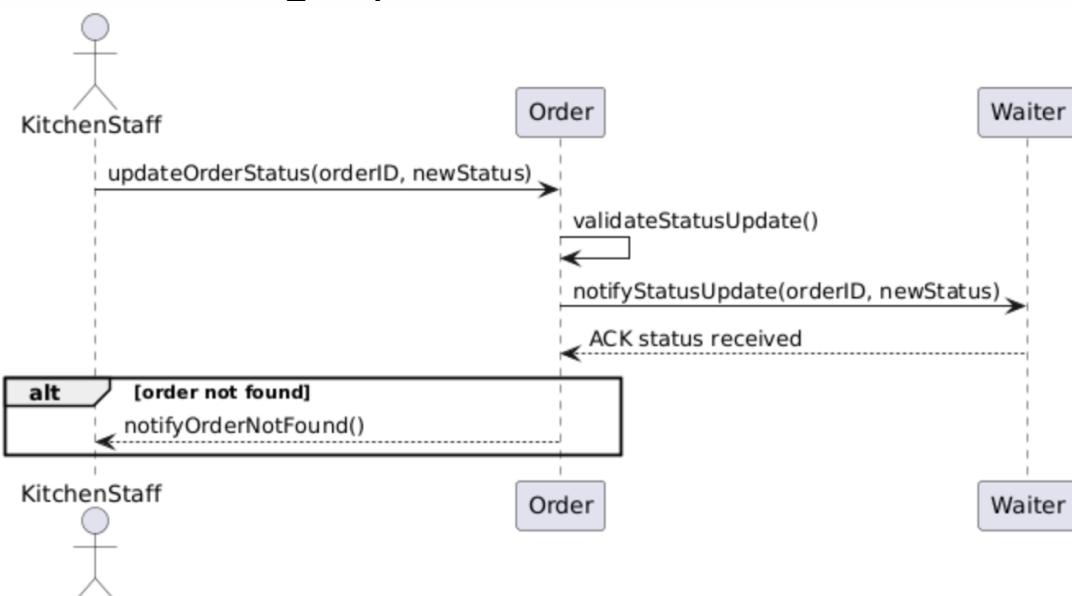


5.6 Sequence Diagram

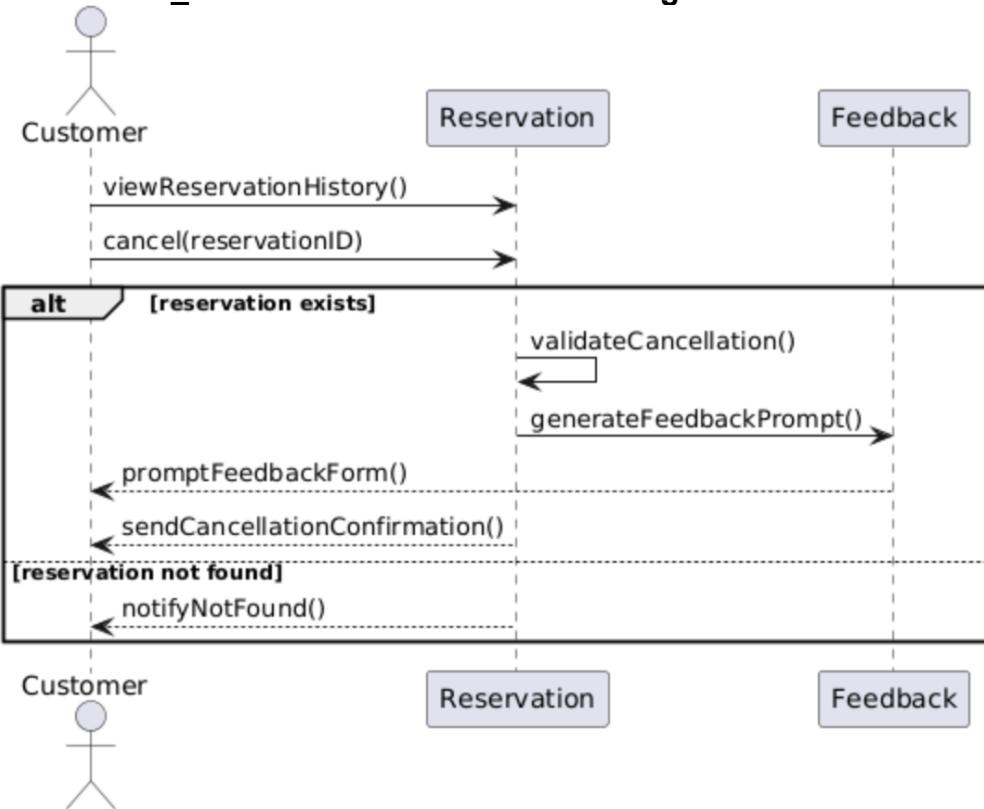
SC_10 Table Cancellation – Joldi Xure



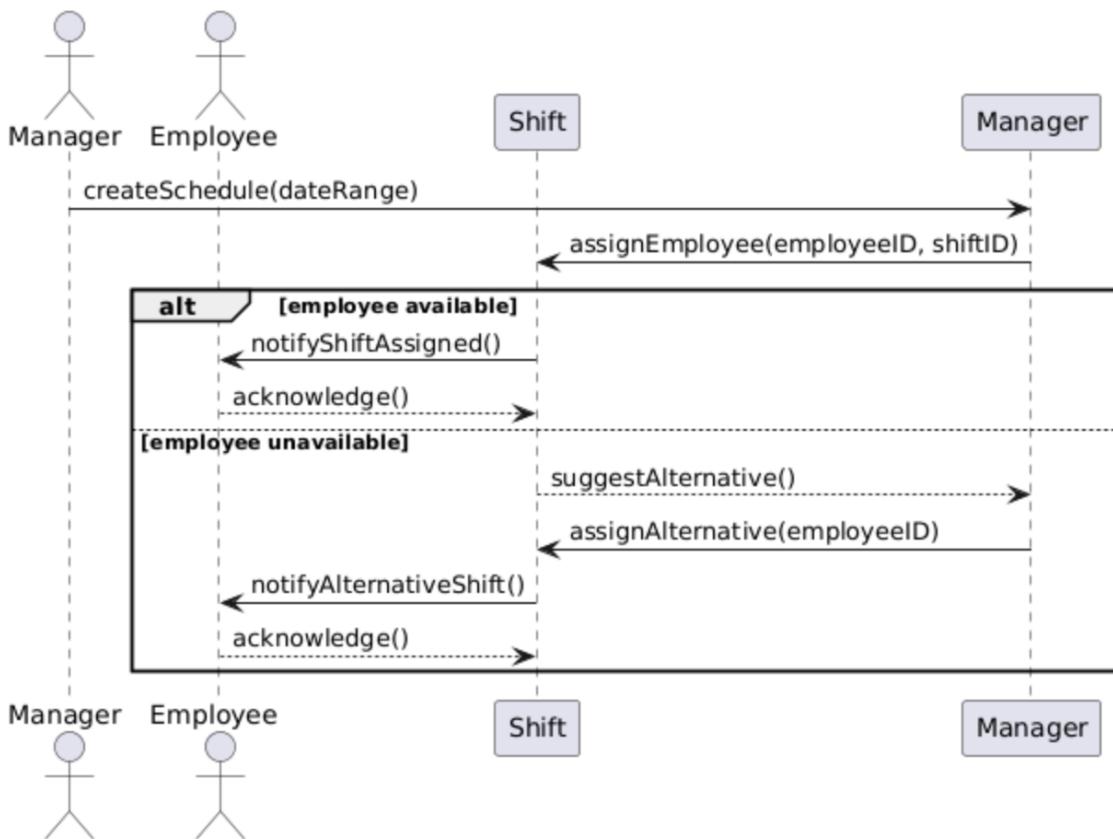
SC_11 Update Order Status – Joldi Xure



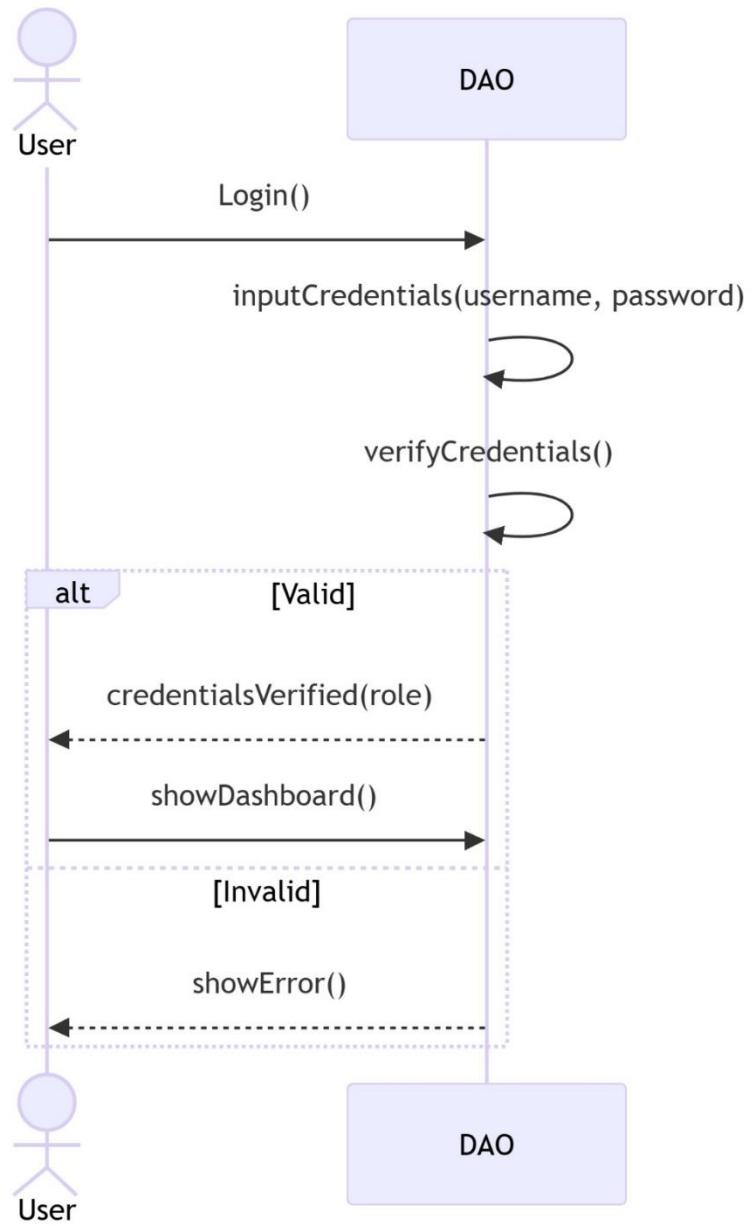
Restaurant Management System Requirements Specification
SC_12 Customer Feedback & Ratings- Joldi Xure



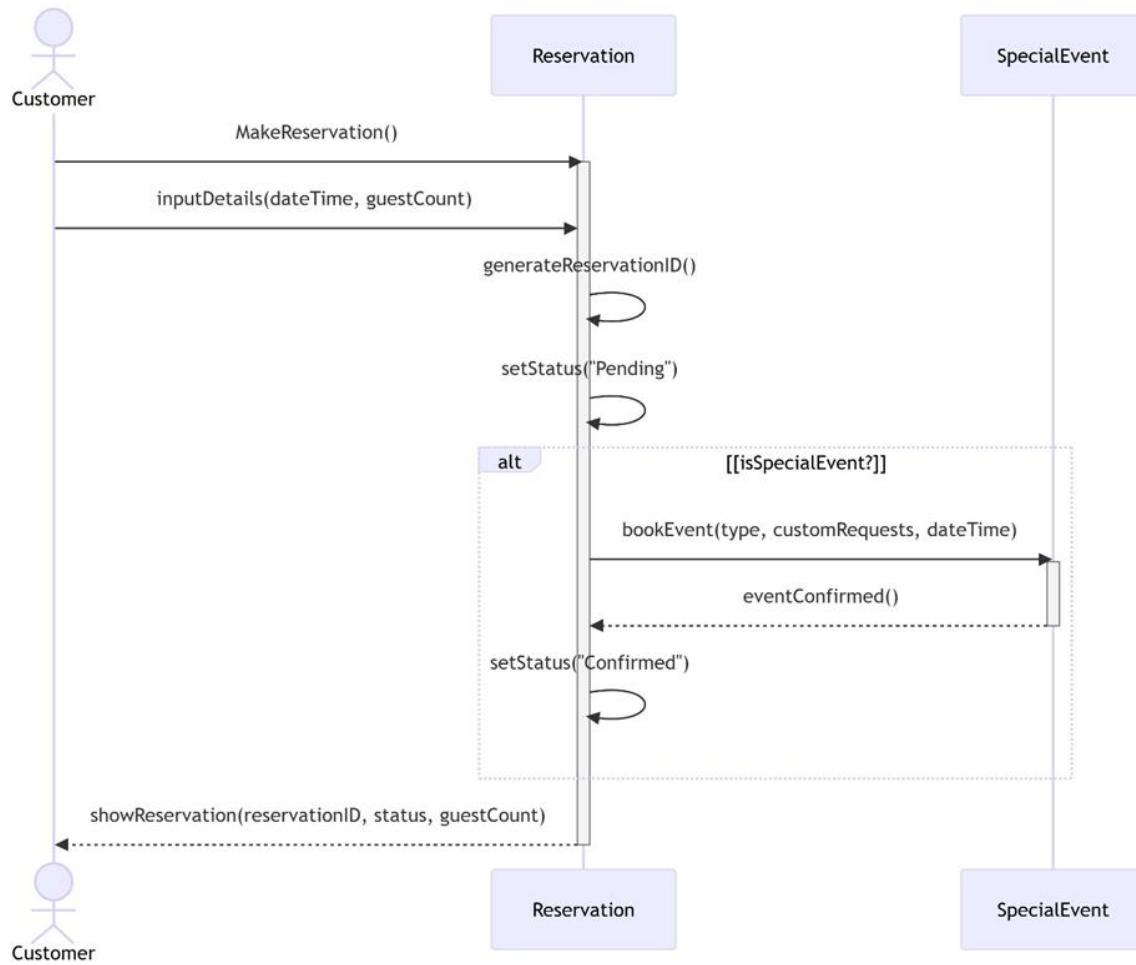
SC_13 Shift Scheduling for Staff – Joldi Xure



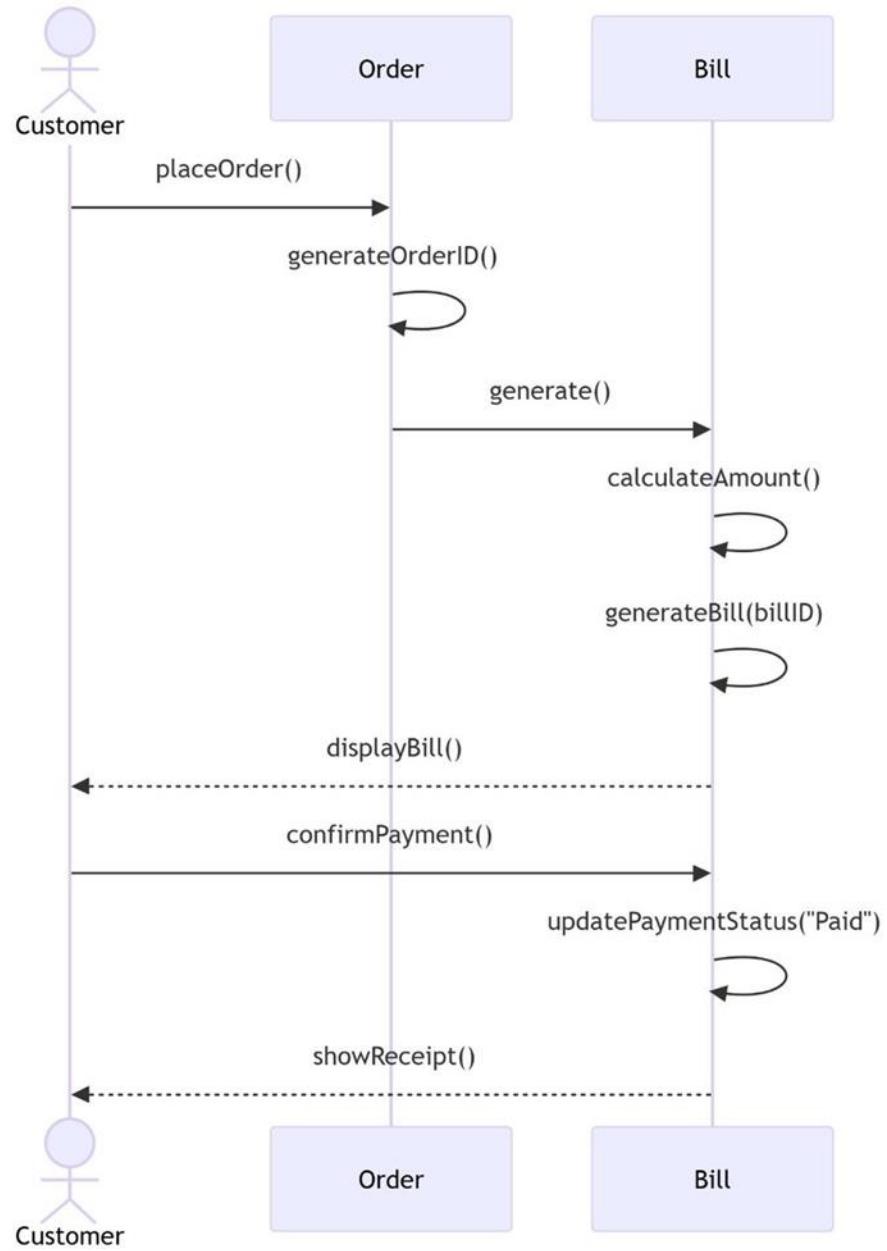
SC_01: Login User – Gloria Traja



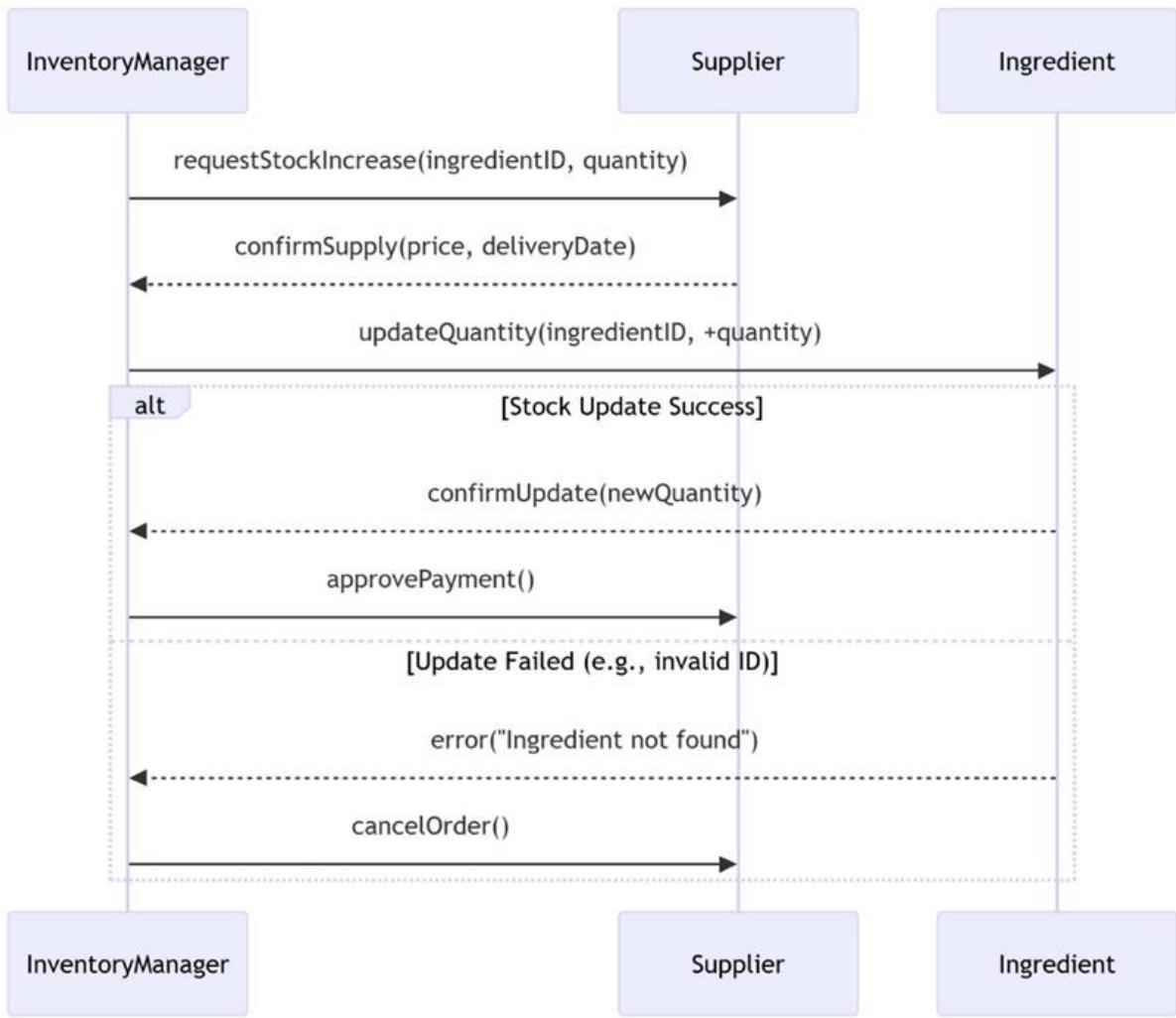
SC_02: Table Reservation – Gloria Traja



SC_03: Bill Generation & Payment Processing – Gloria Traja

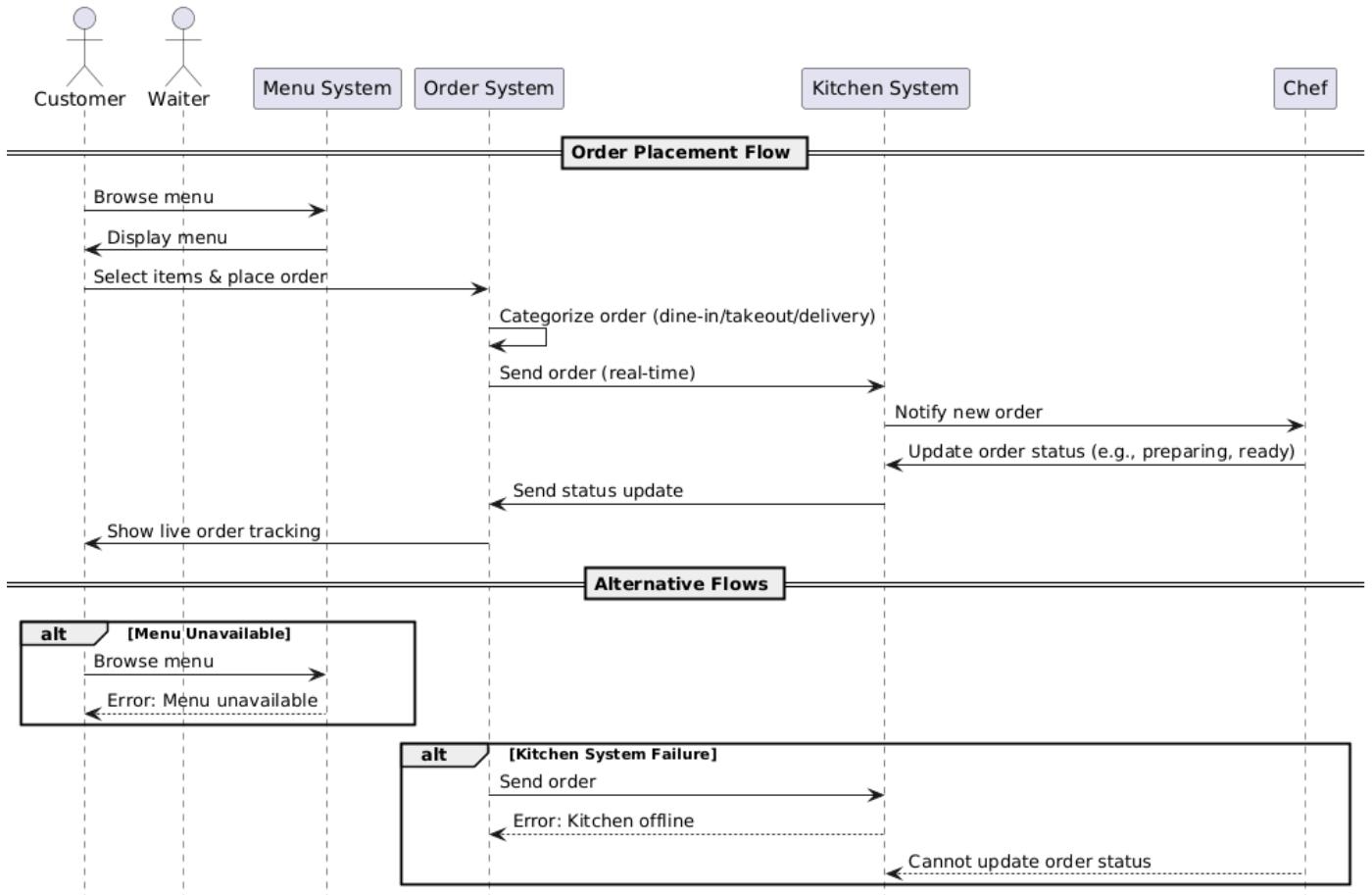


Restaurant Management System Requirements Specification
SC_04: Ingredient Stock Management Increase – Gloria Traja

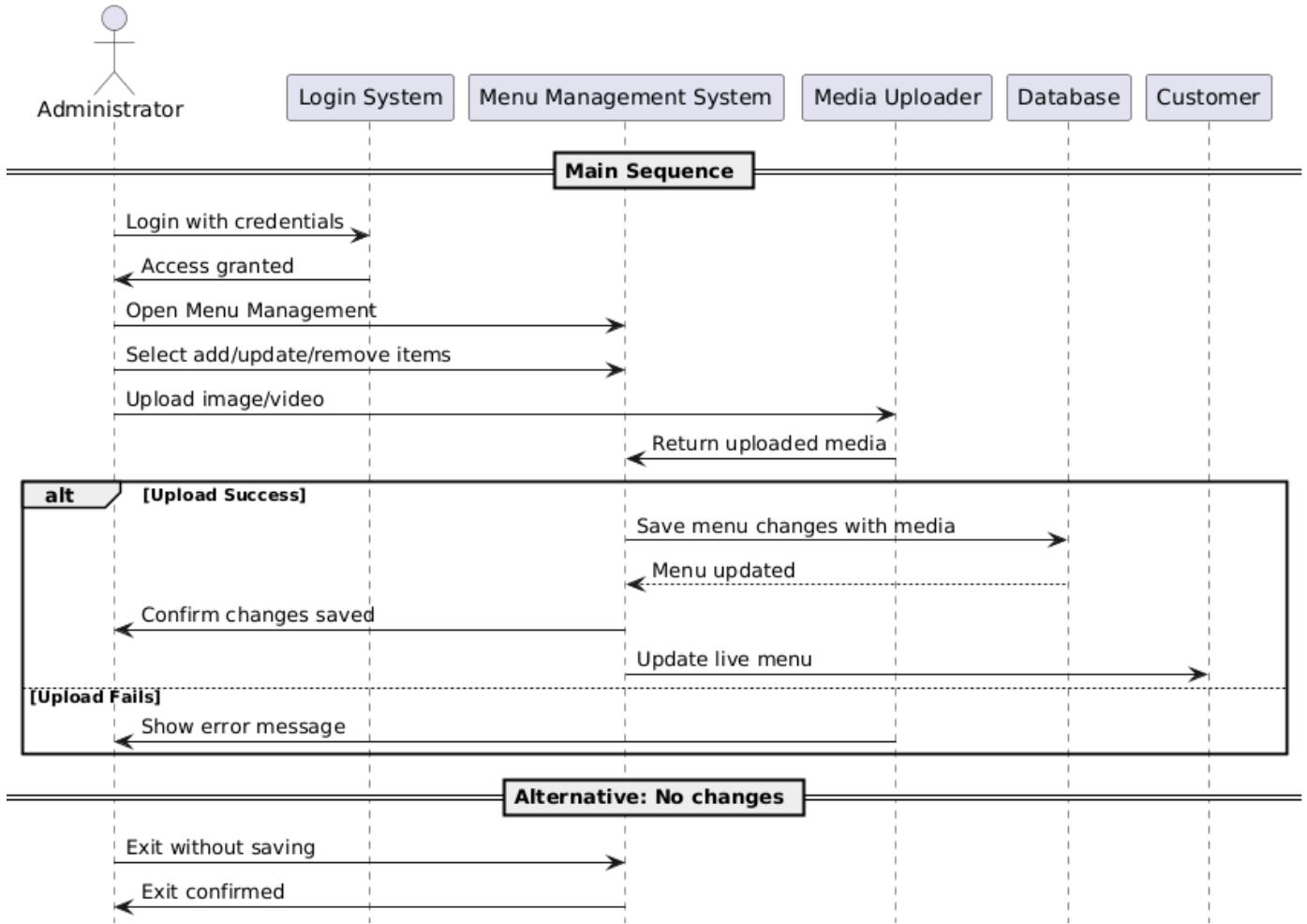


Restaurant Management System Requirements Specification

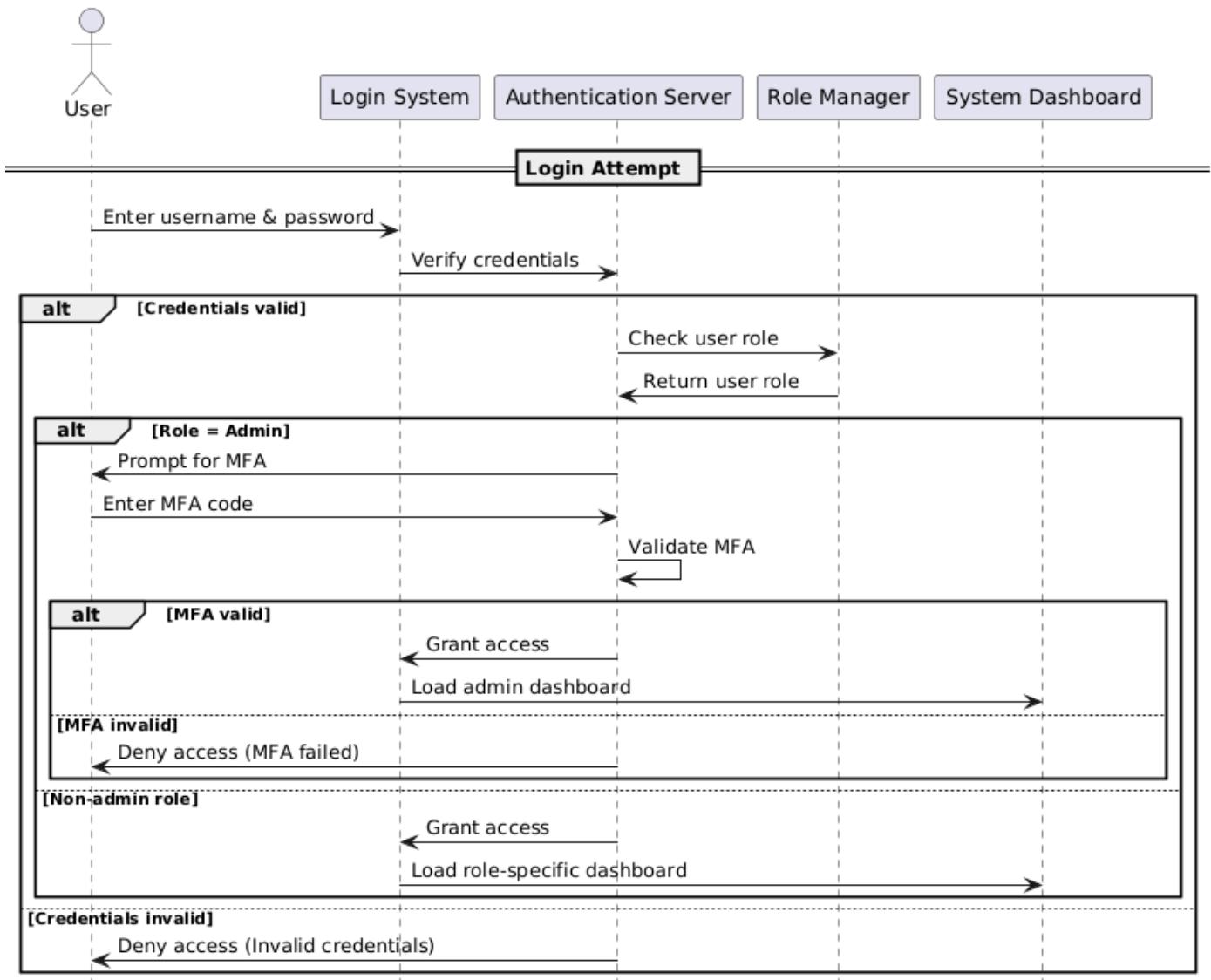
SC_06: Place Order – Alisa Tozaj



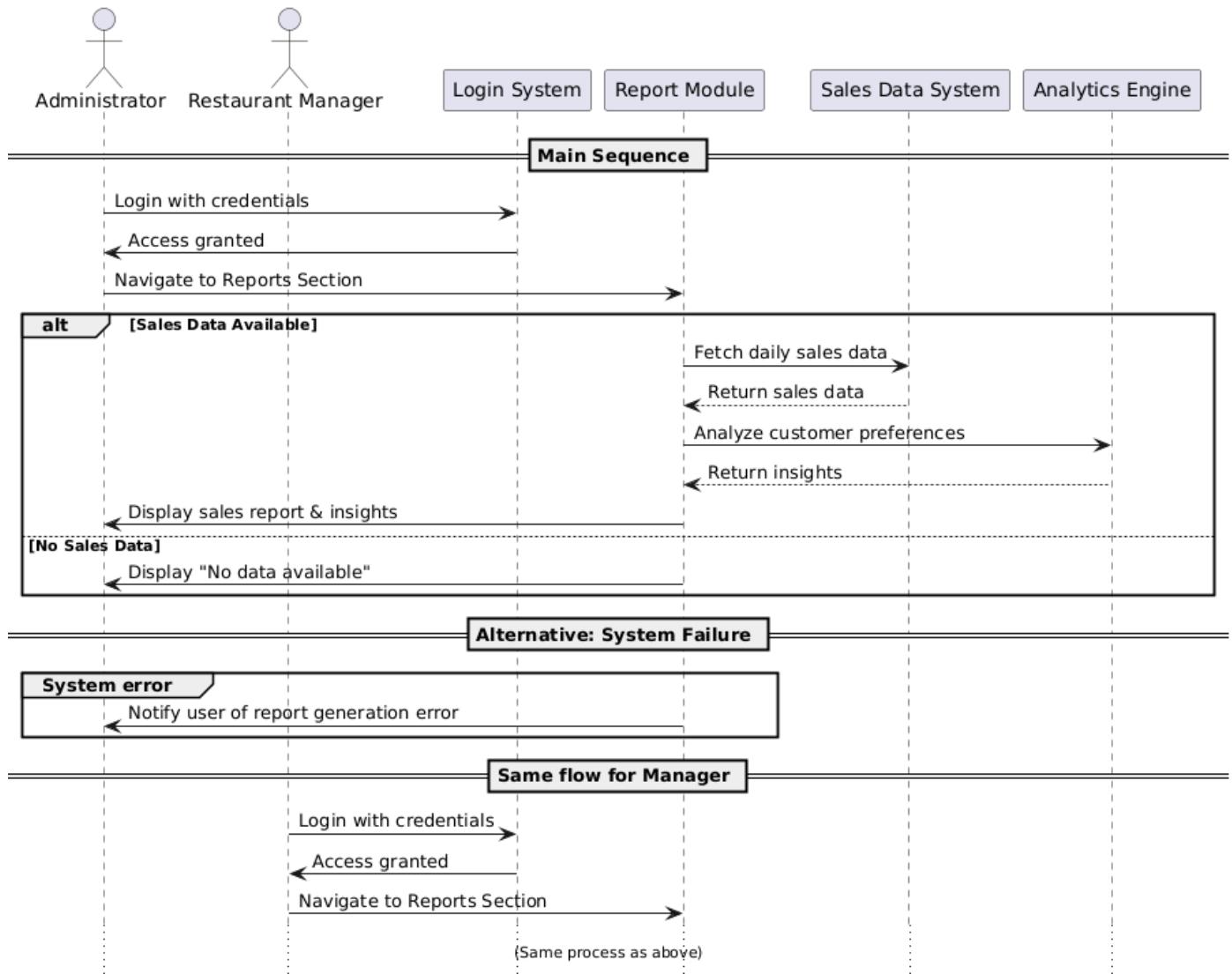
Restaurant Management System Requirements Specification
SC_07: Menu Management – Alisa Tozaj



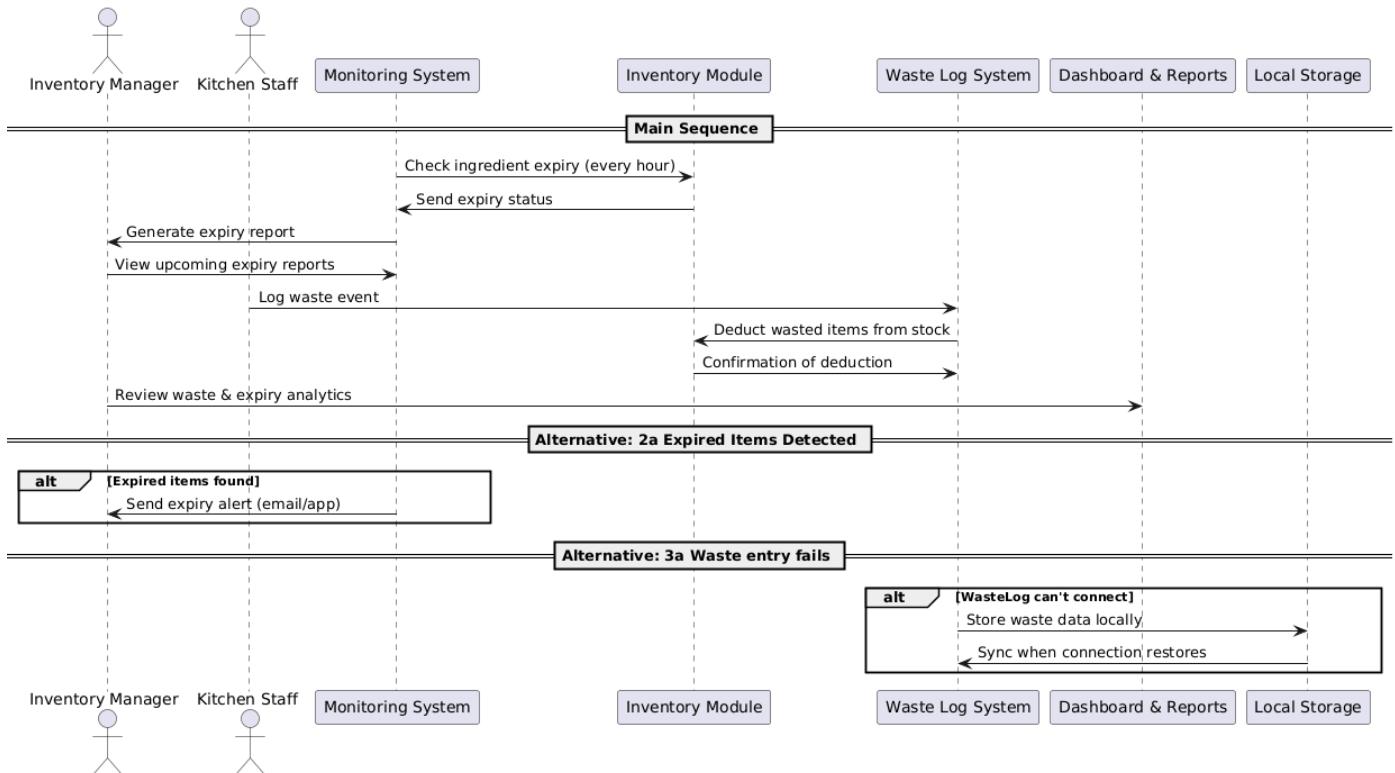
SC_08: User Access & Security – Alisa Tozaj



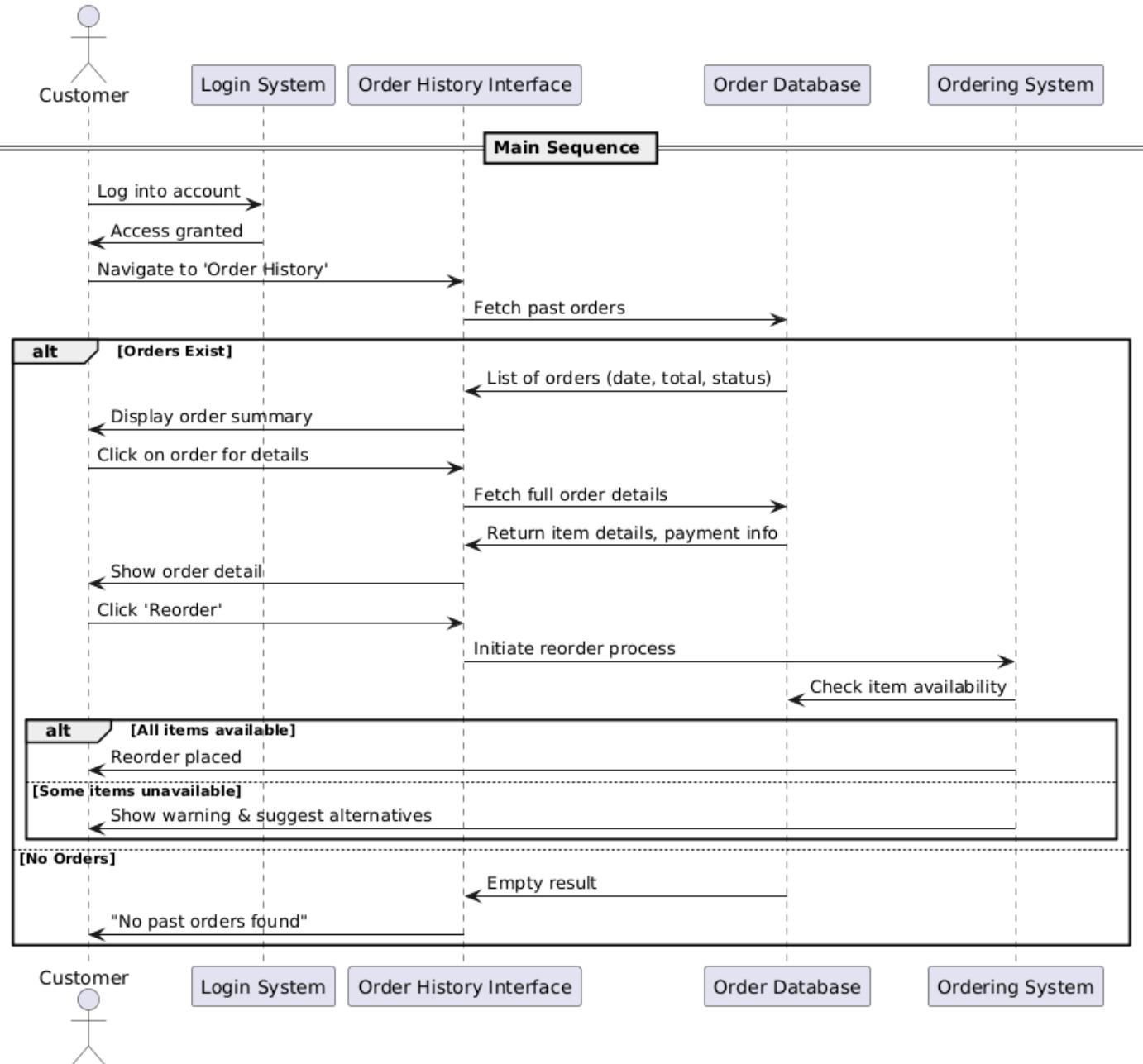
SC_09: Sales Reports – Alisa Tozaj



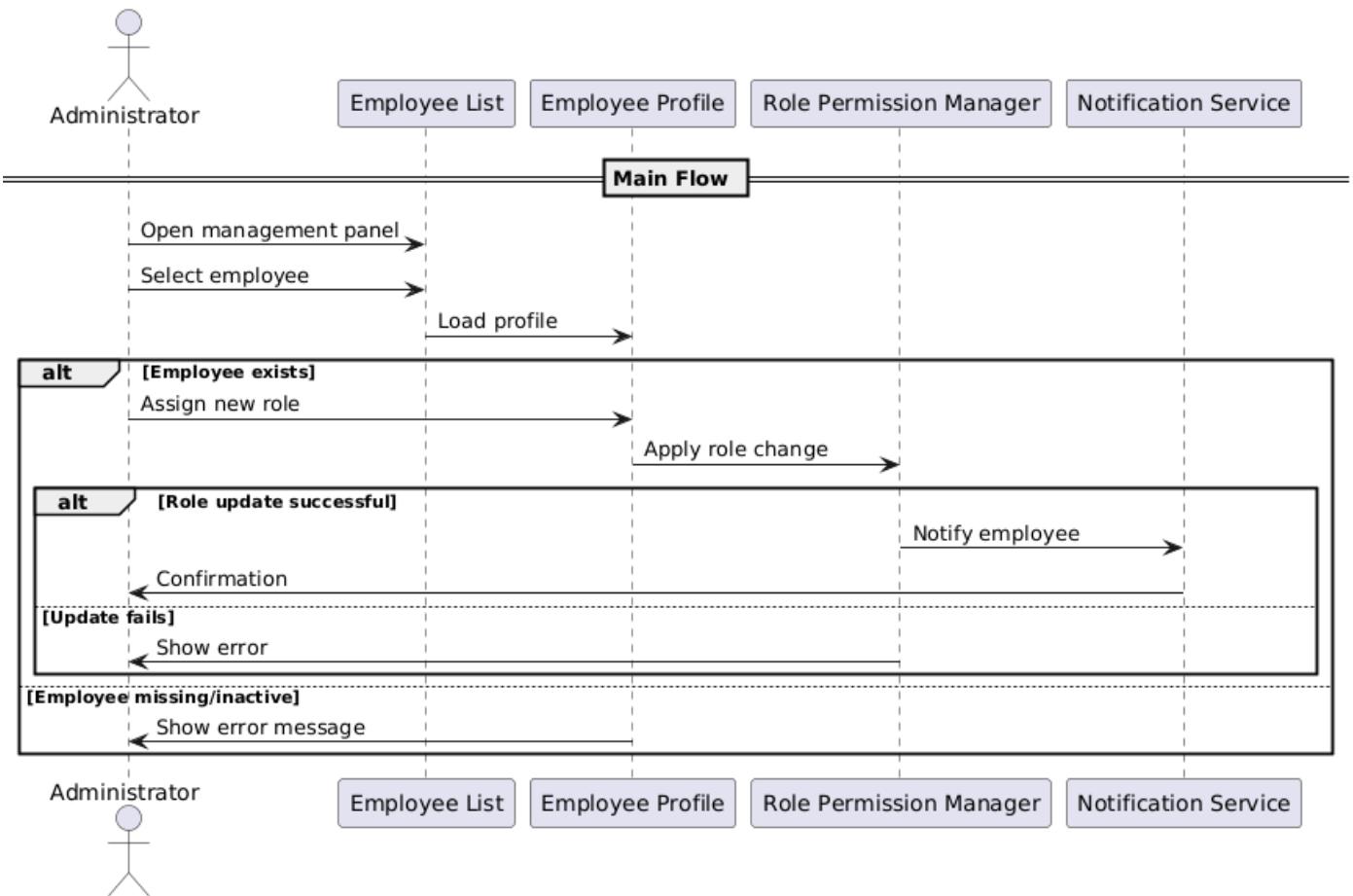
Restaurant Management System Requirements Specification
SC_14: Waste & Expiry Tracking – Ester Qershori



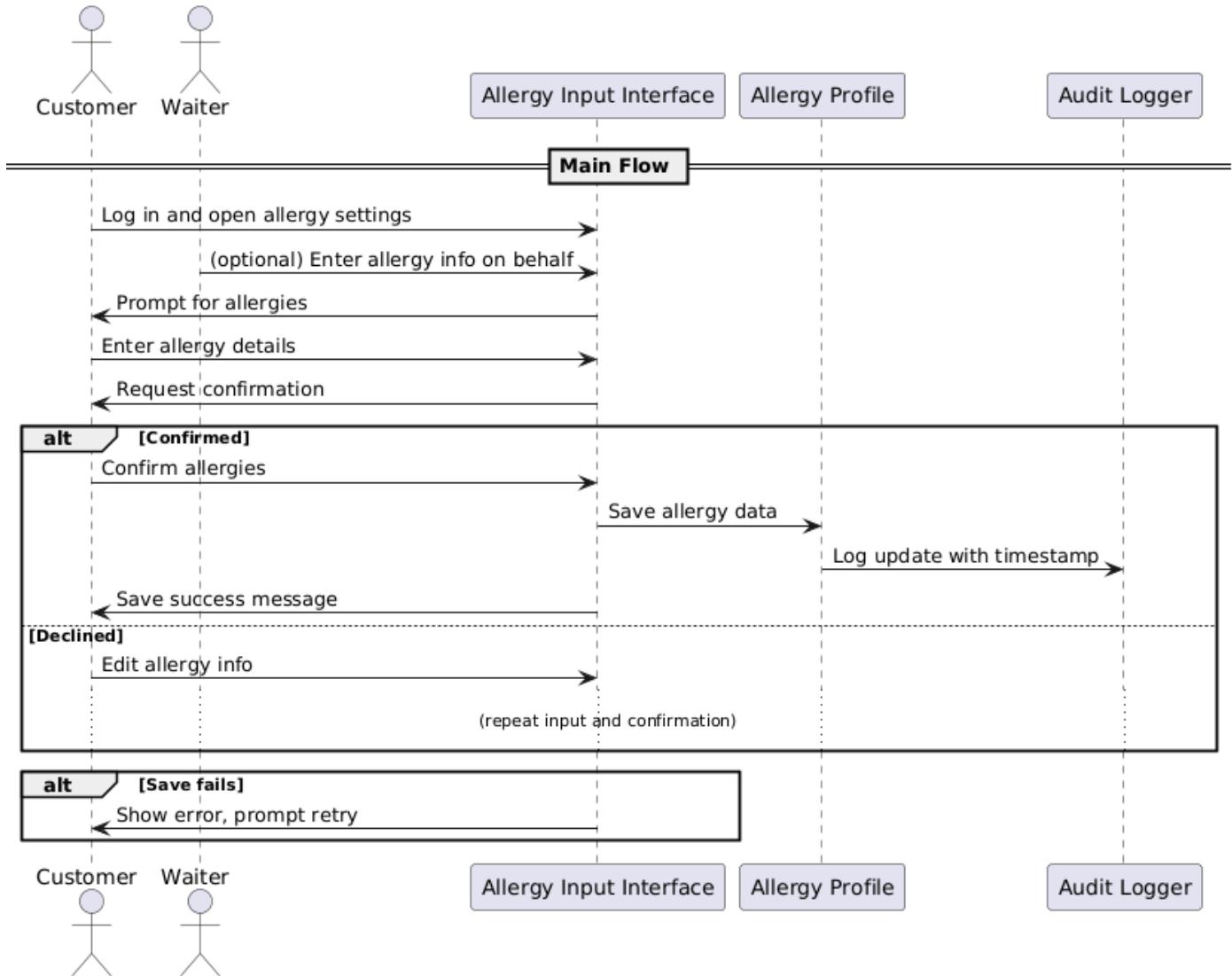
Restaurant Management System Requirements Specification
SC_15: Customer Order History Viewer – Ester Qershori



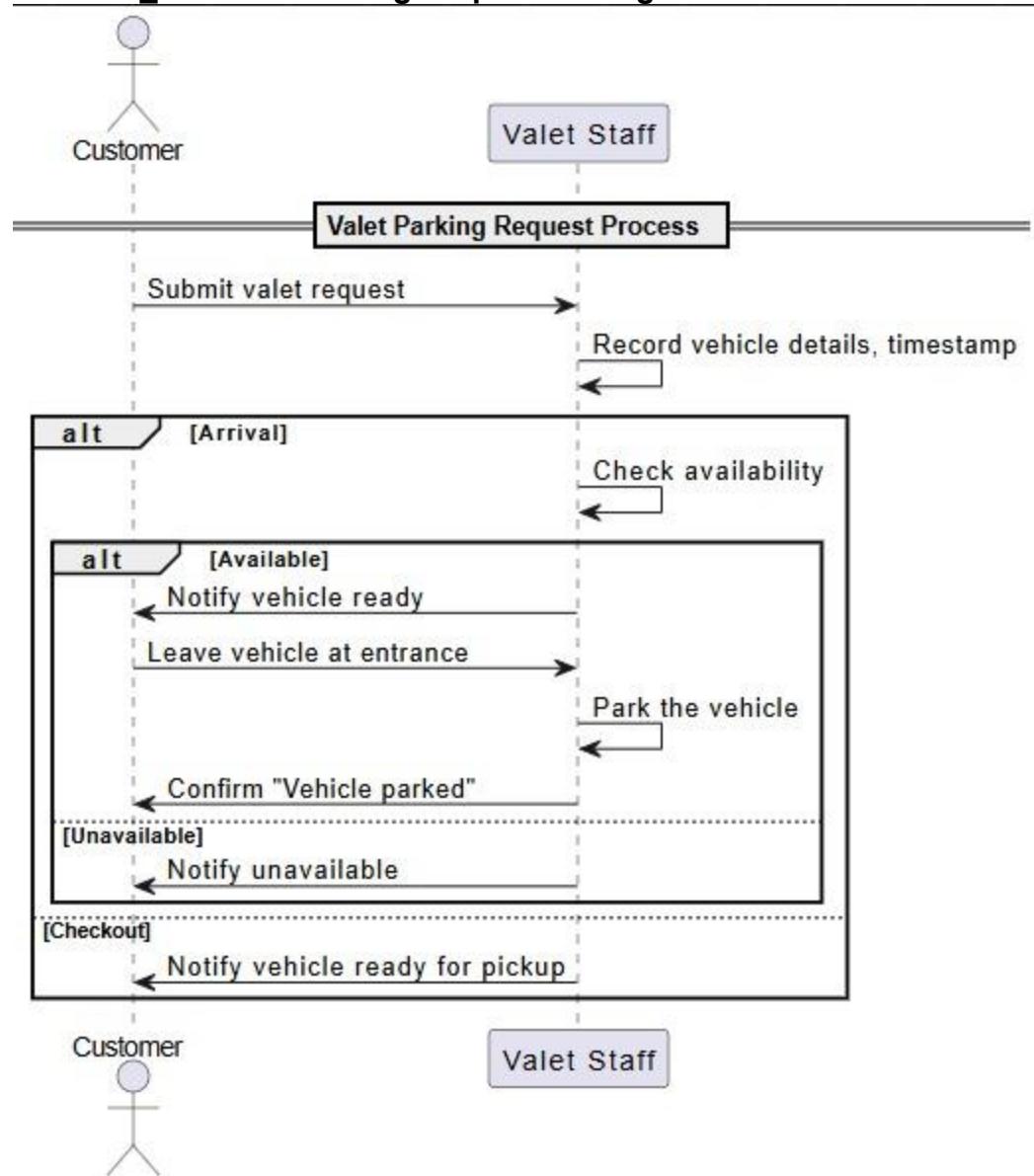
Restaurant Management System Requirements Specification
SC_16: Employee Role Assignment – Ester Qershori



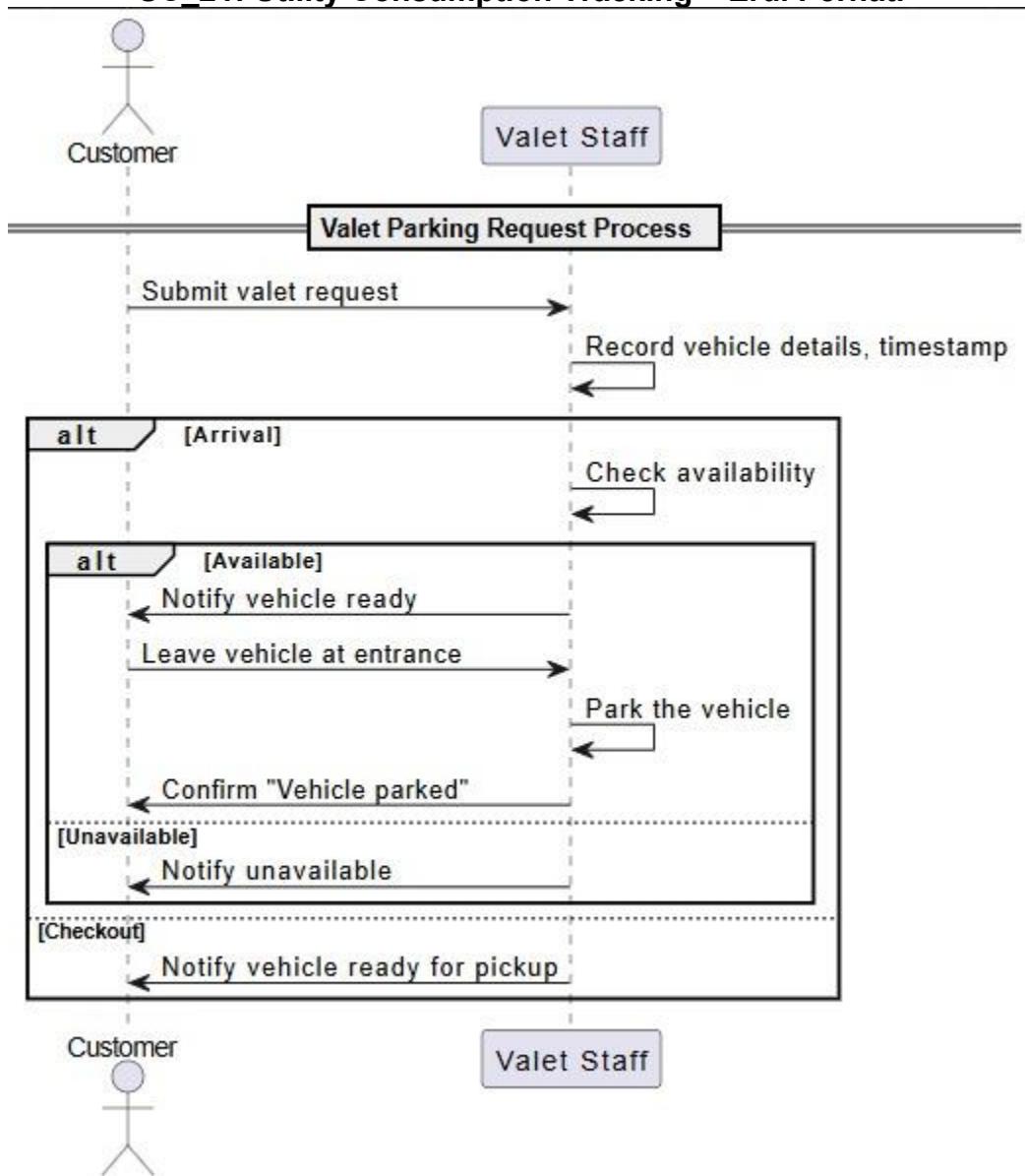
Restaurant Management System Requirements Specification
SC_29: Record Customer Allergy Tracking – Ester Qershori



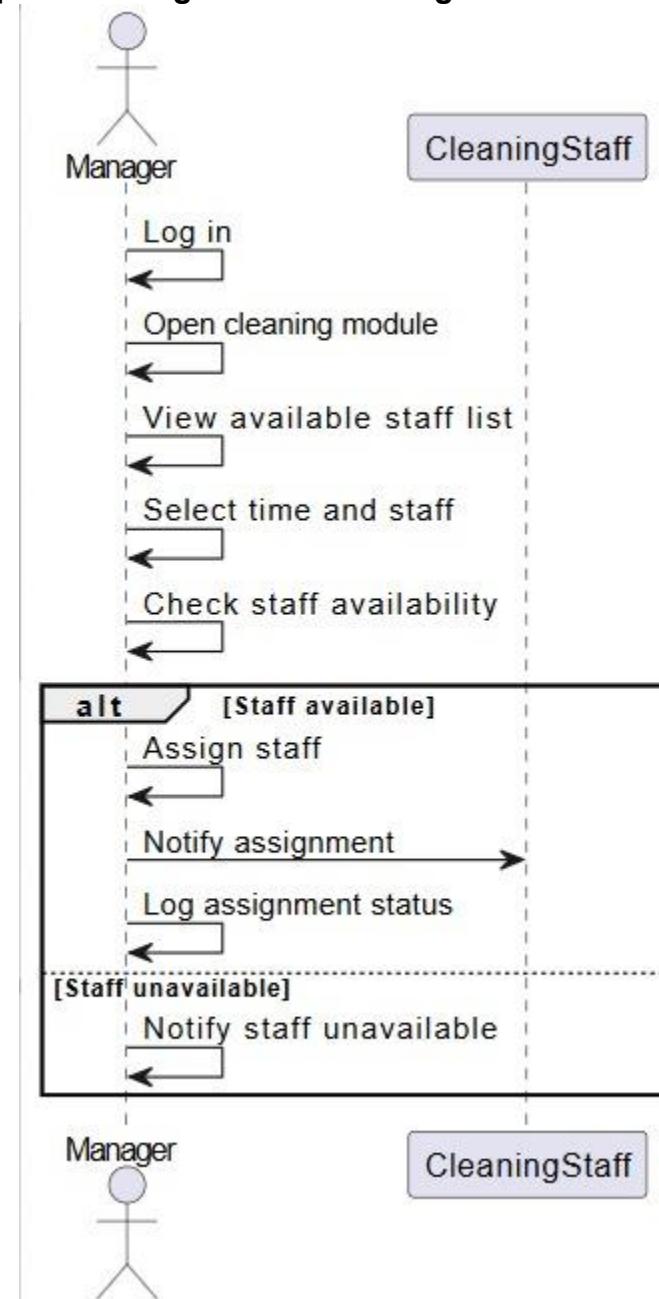
SC_20: Valet Parking Request Management – Erdi Perhati



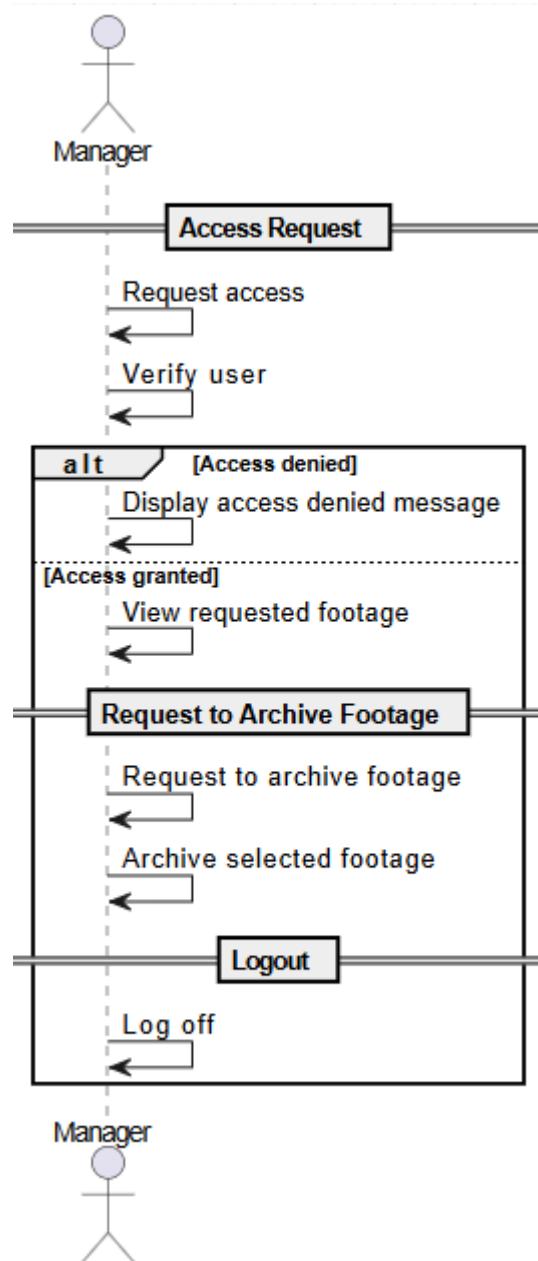
Restaurant Management System Requirements Specification
SC_21: Utility Consumption Tracking – Erdi Perhati



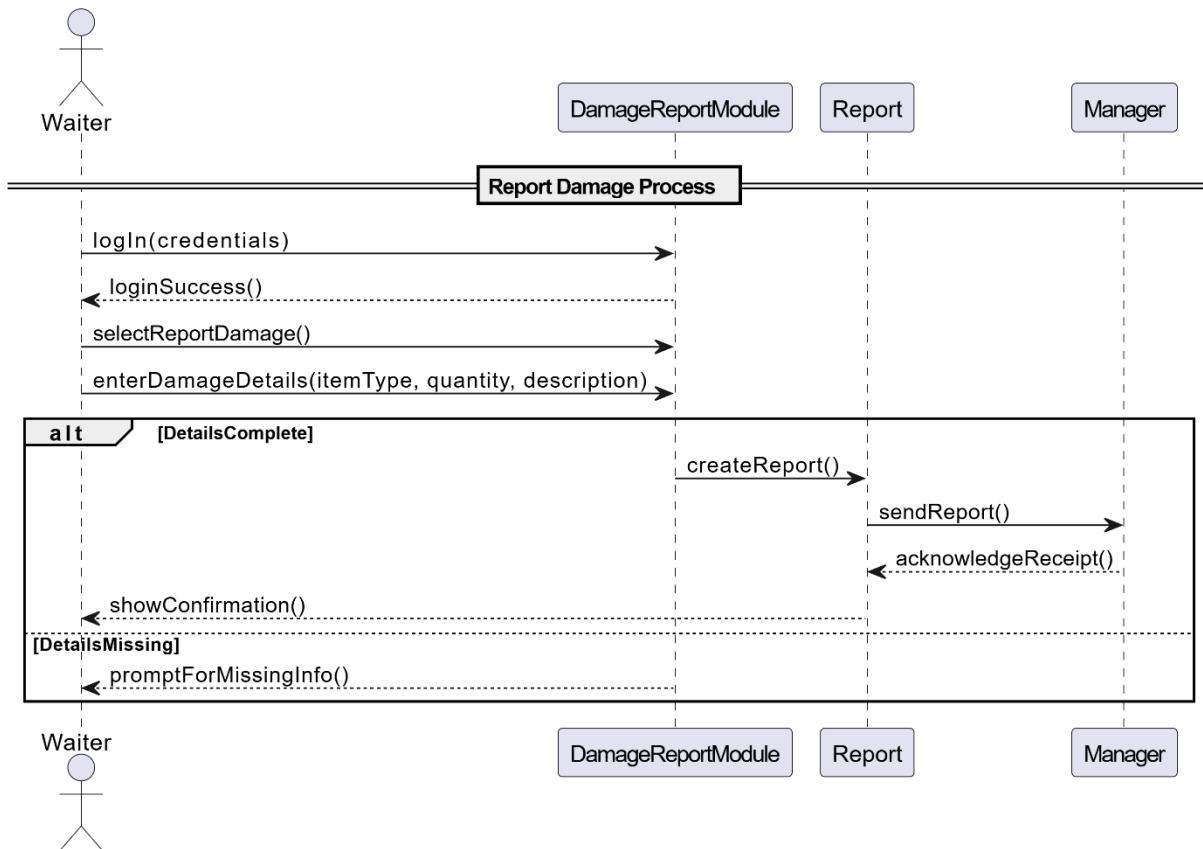
SC_22: Cleaning Schedule Management – Erdi Perhati



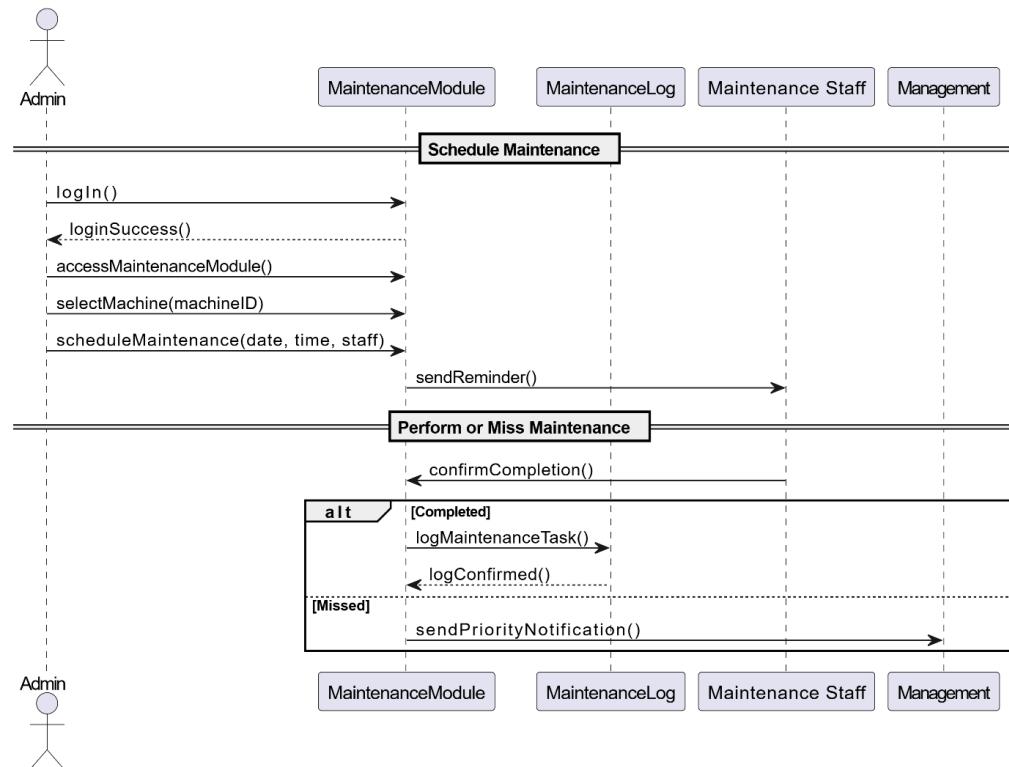
SC_23: Security Camera Footage Log – Erdi Perhati



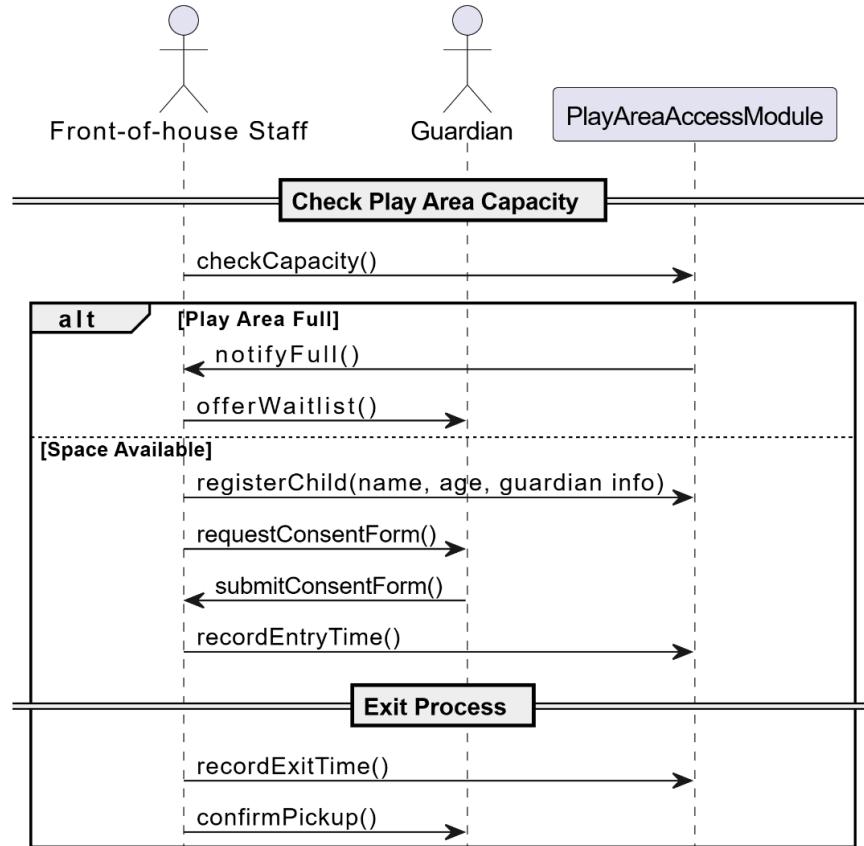
SC_25: Tableware Damage Reporting – Keisi Loci



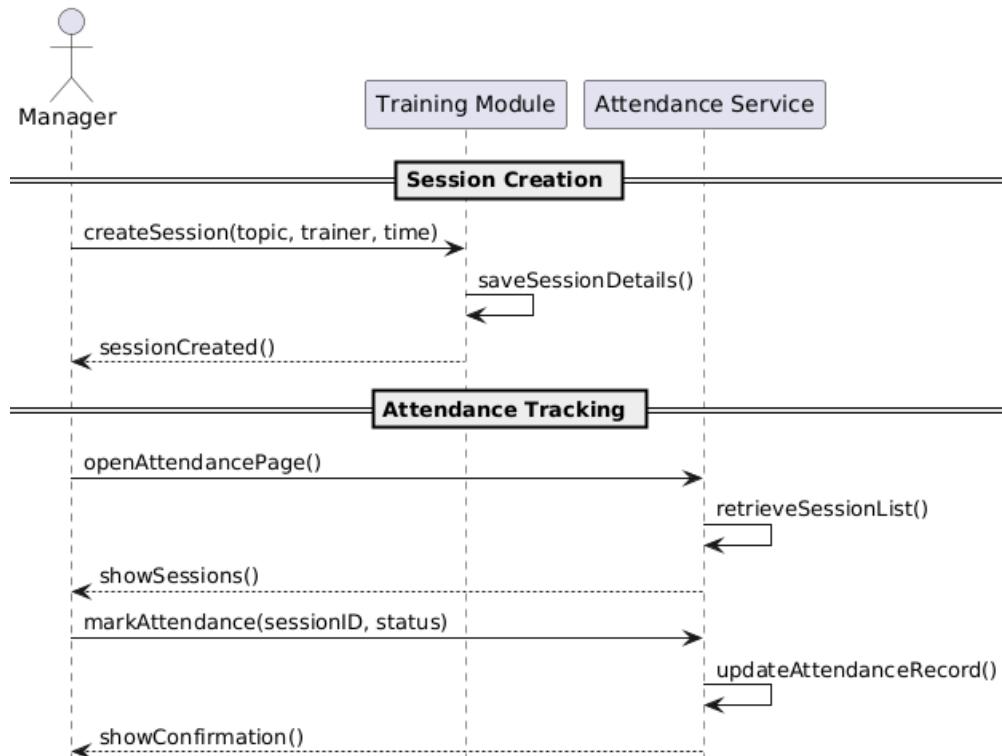
SC_27: Dishwashing Machine Maintenance Scheduling – Keisi Loci



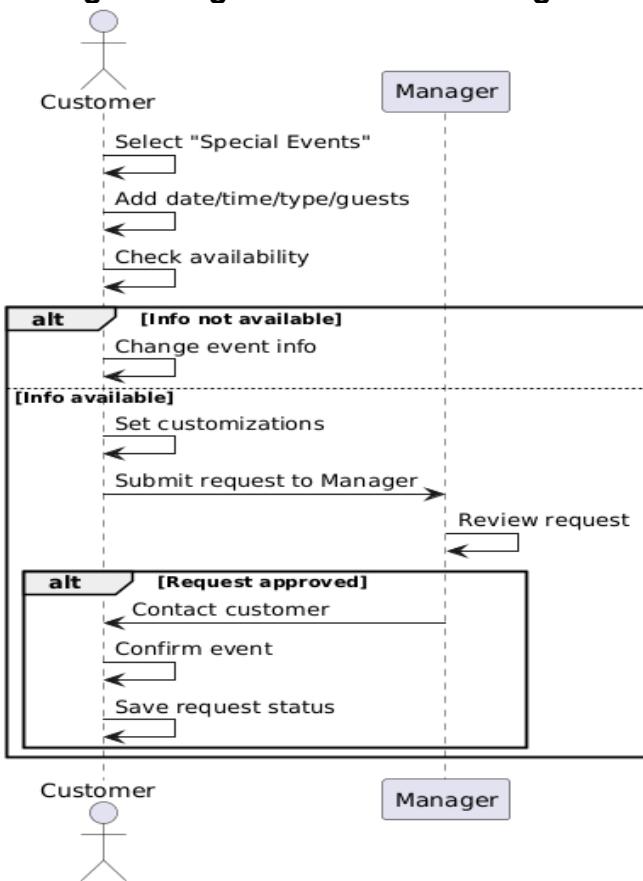
Restaurant Management System Requirements Specification
SC_28: Play Area Access Registration – Keisi Loci



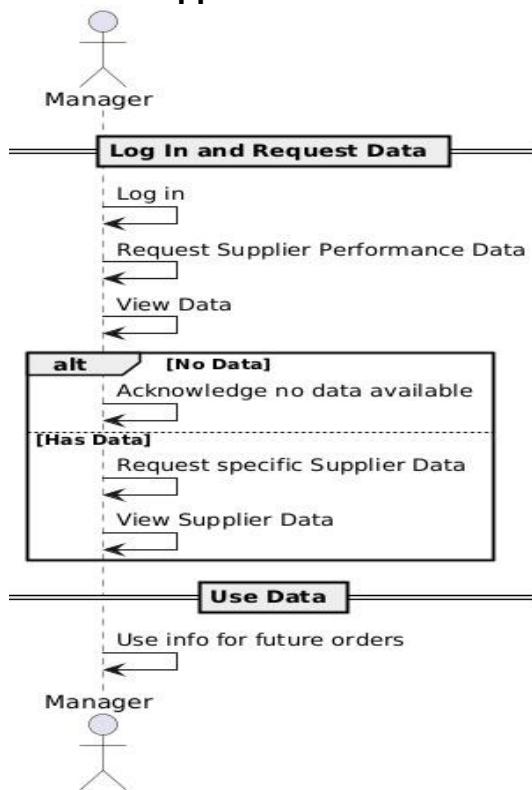
SC_24: Training Session Scheduling – Keisi Loci



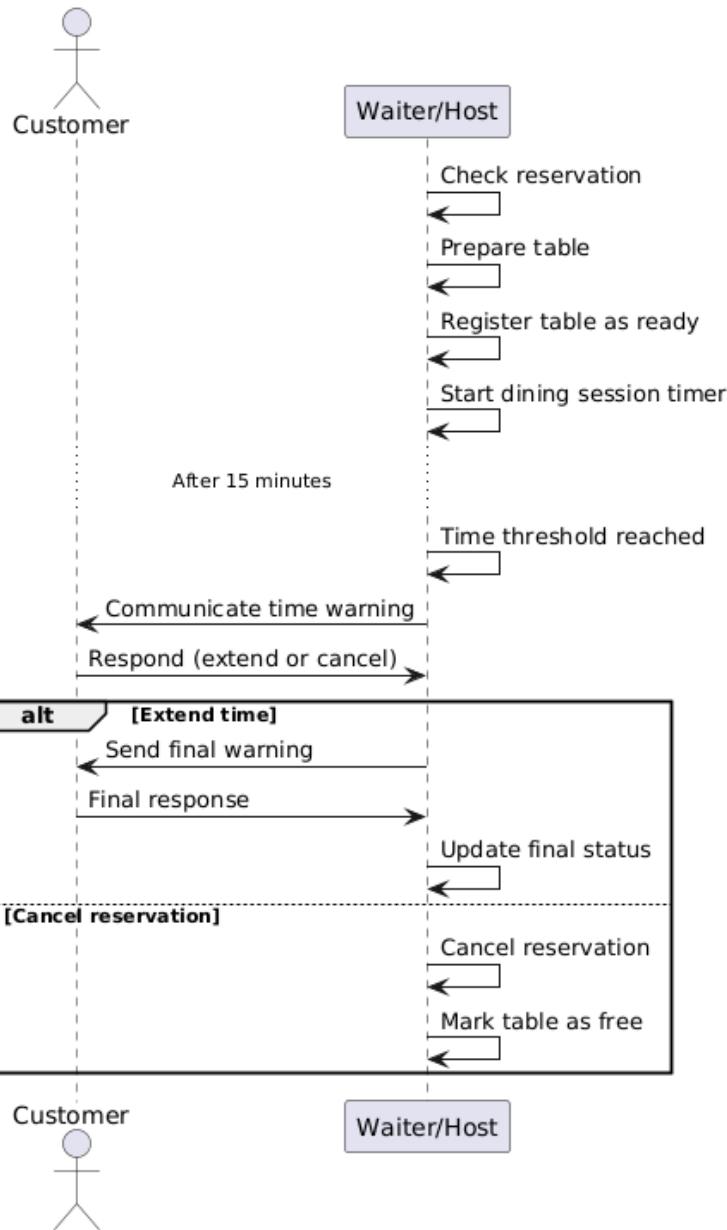
Restaurant Management System Requirements Specification
SC_30: Manage Dining Time Limits Warnings – Arlis Arapi



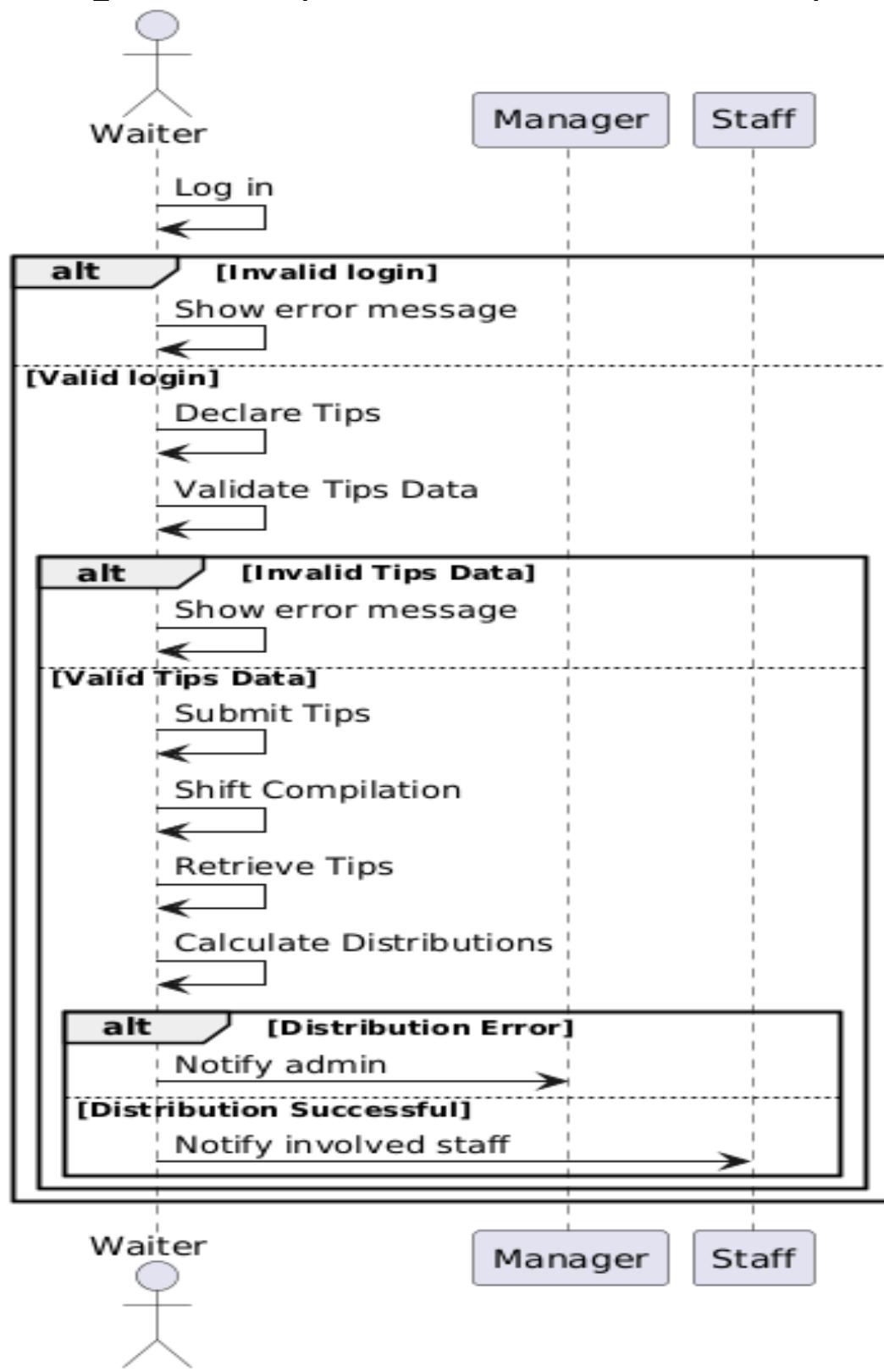
SC_17: Monitor Supplier Performance – Arlis Arapi



Restaurant Management System Requirements Specification
SC_19: Special Event Booking Table – Arlis Arapi

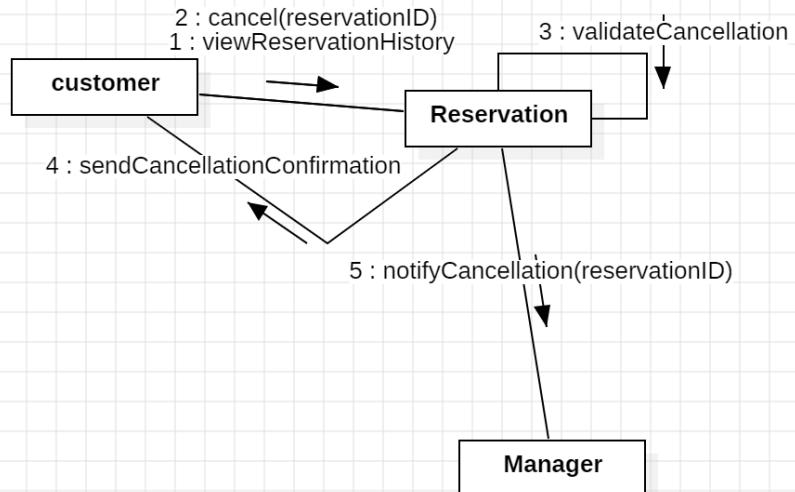


Restaurant Management System Requirements Specification
SC_18: Declared Tips & Automated Distribution – Arlis Arapi

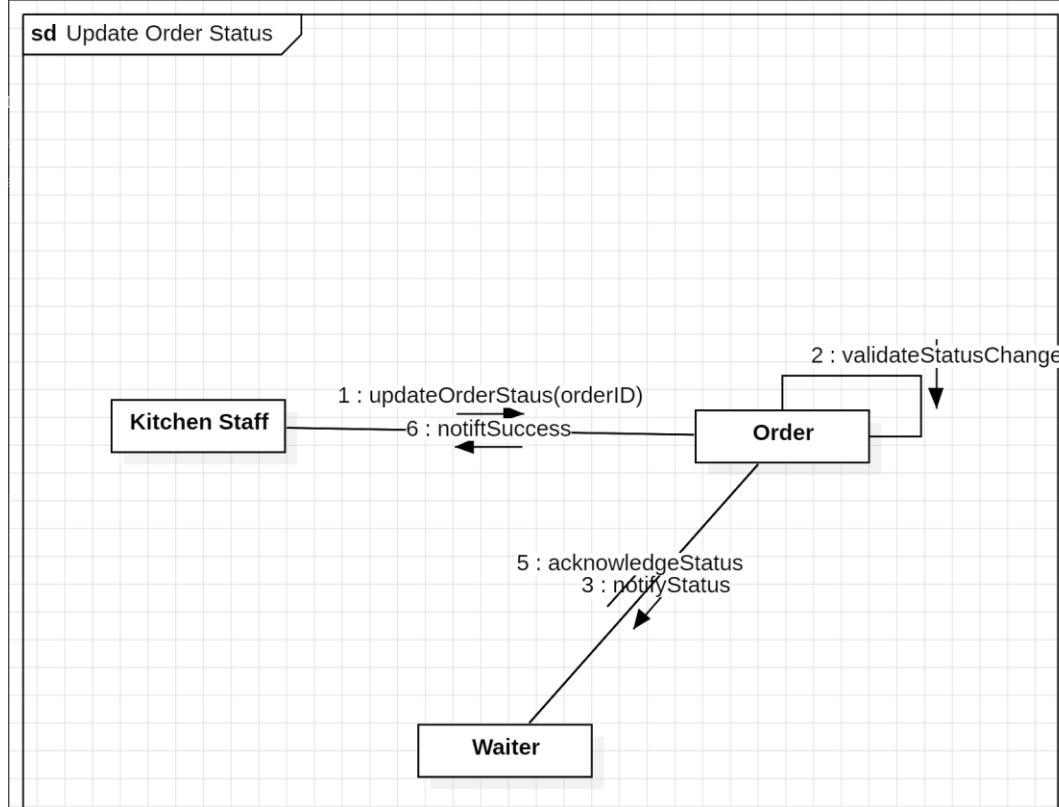


5.7 Collaboration Diagram

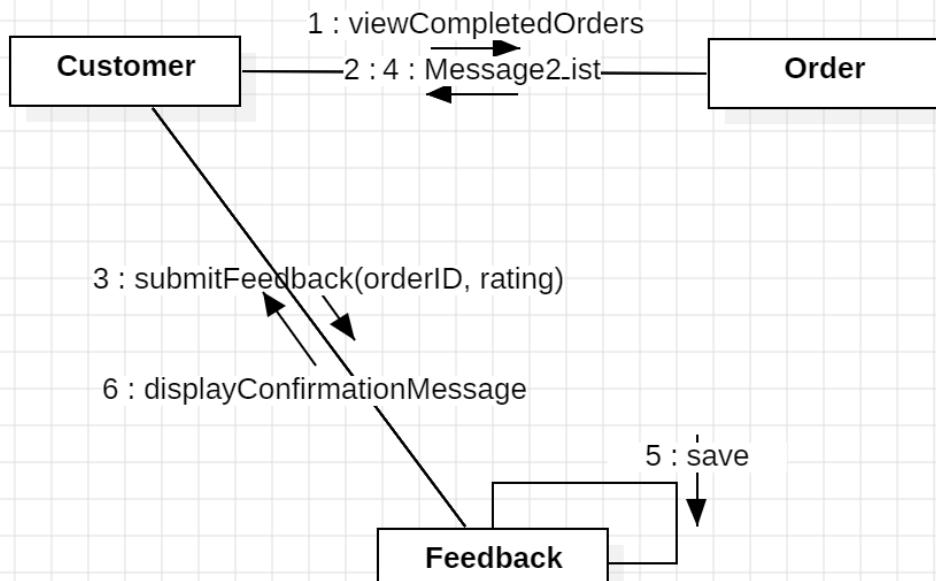
CC-10 Table Cancellation – Joldi Xure



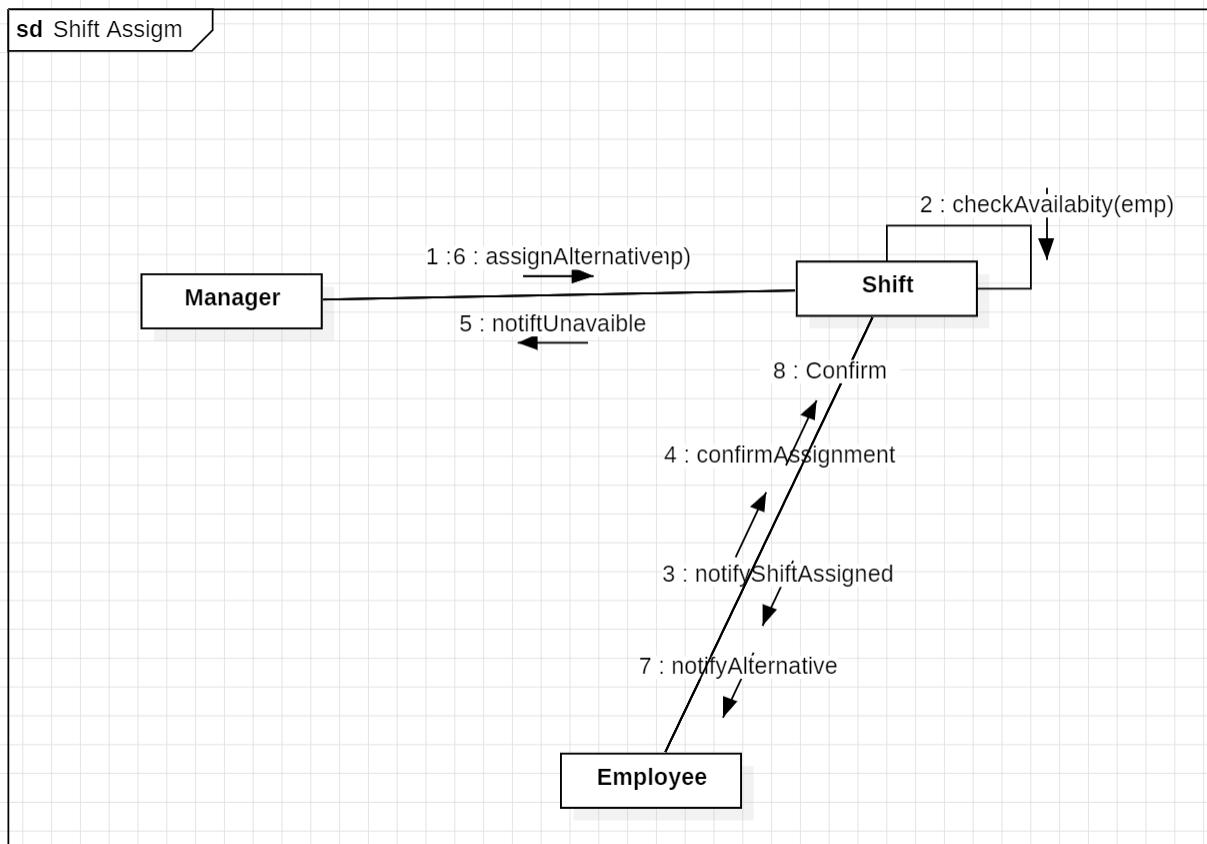
CC -11 Update Order Status – Joldi Xure



CC -12 Customer Feedback & Ratings – Joldi Xure



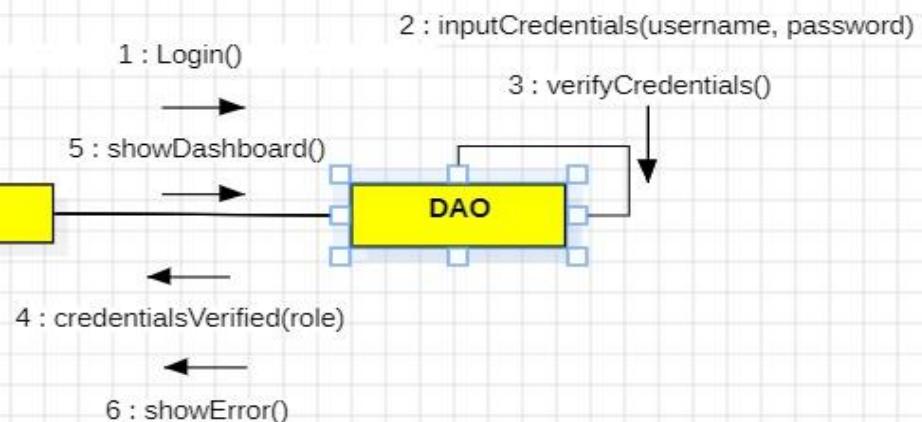
CC -13 Shift Scheduling for Staff – Joldi Xure



Restaurant Management System Requirements Specification

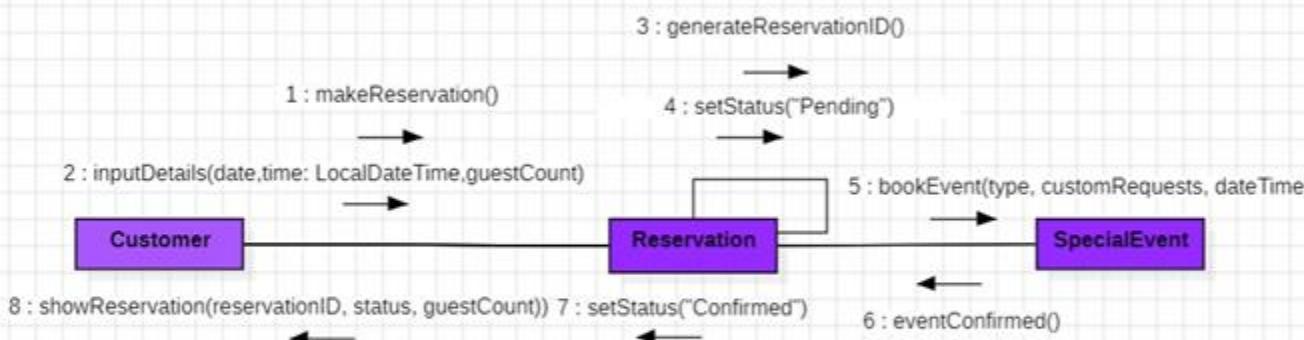
CC_01: Login User – Gloria Traja

sd CommunicationDiagram Login User



CC_02: Table Reservation – Gloria Traja

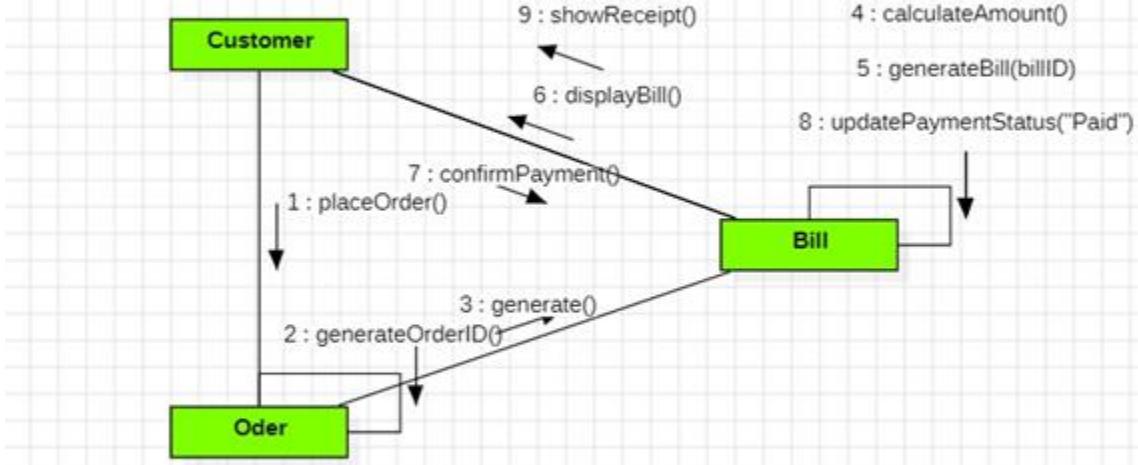
sd Communication Diagram Table Reservation



Restaurant Management System Requirements Specification

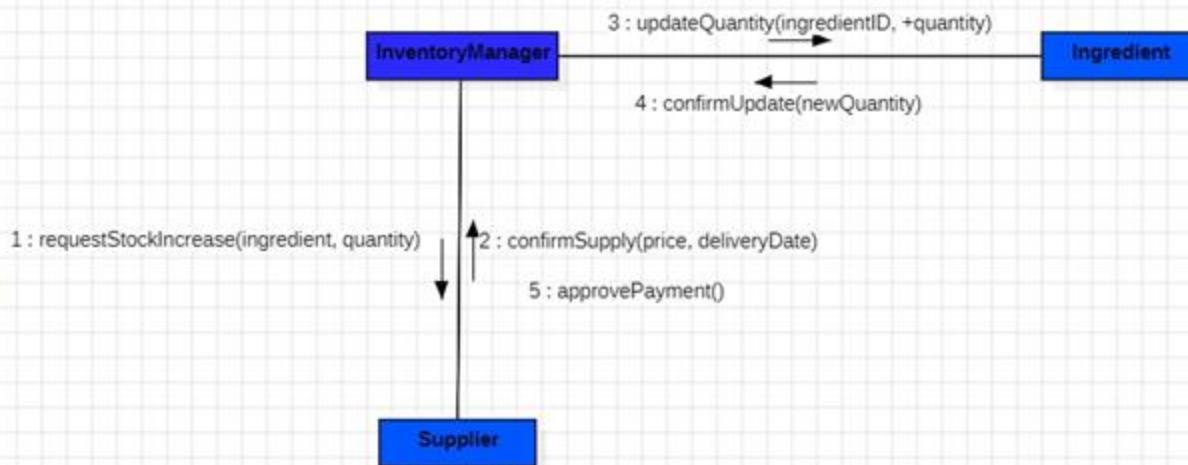
CC_03: Bill Generation & Payment Process – Gloria Traja

sd CommunicationDiagram Bill Generation and Payment Process



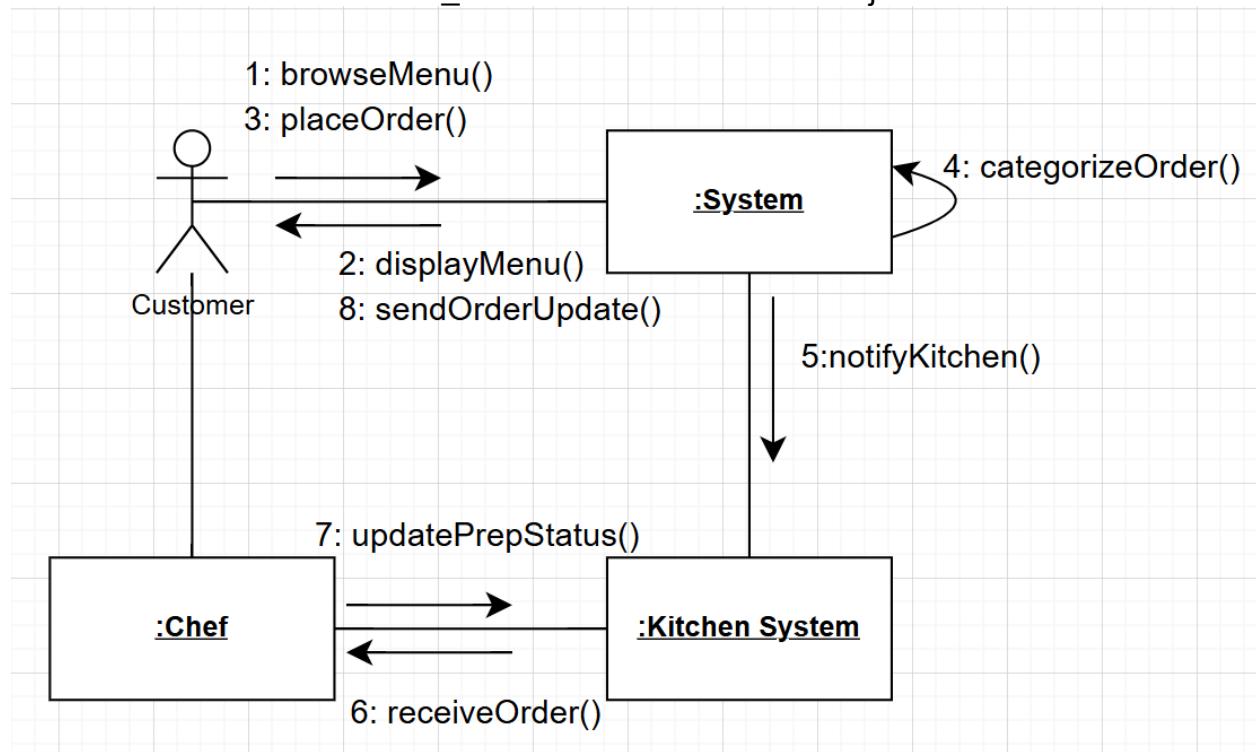
CC_04: Ingredient Stock Management Increase – Gloria Traja

sd CommunicationDiagram Ingredient Stock Management Increase

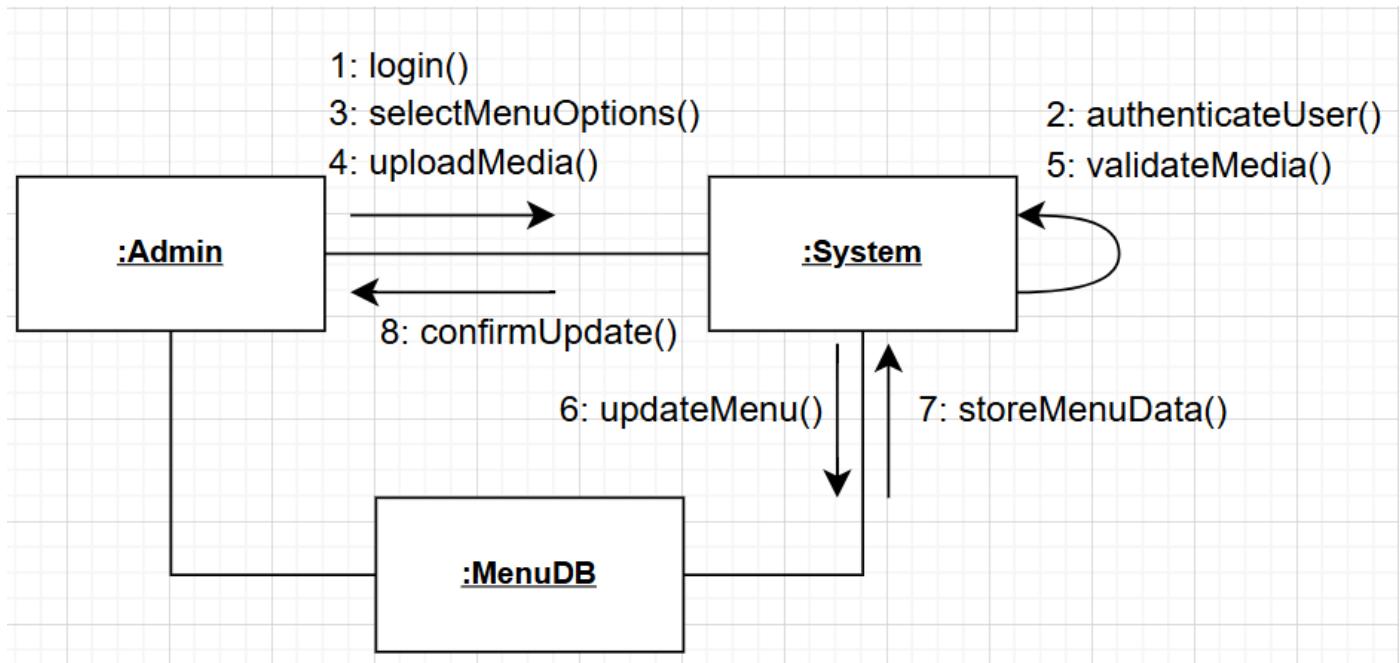


Restaurant Management System Requirements Specification

CC_06: Place Order – Alisa Tozaj

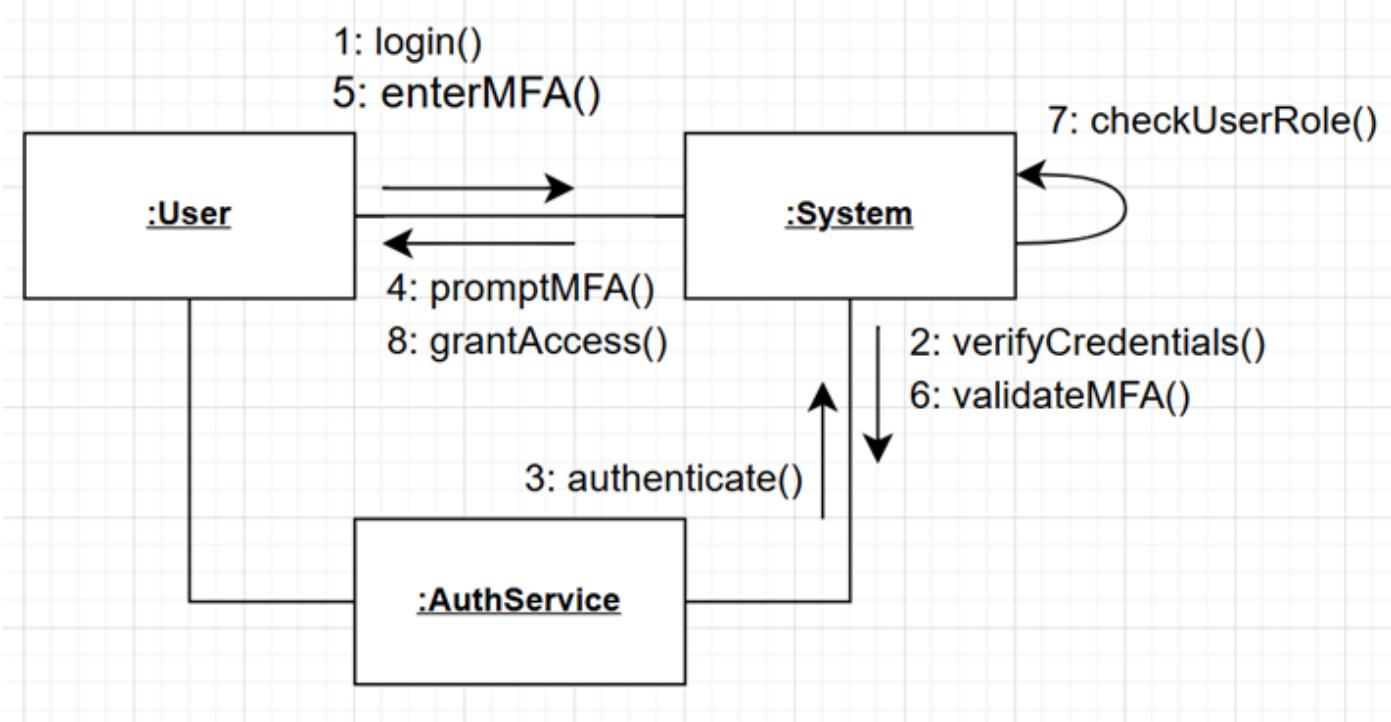


CC_07: Menu Management – Alisa Tozaj

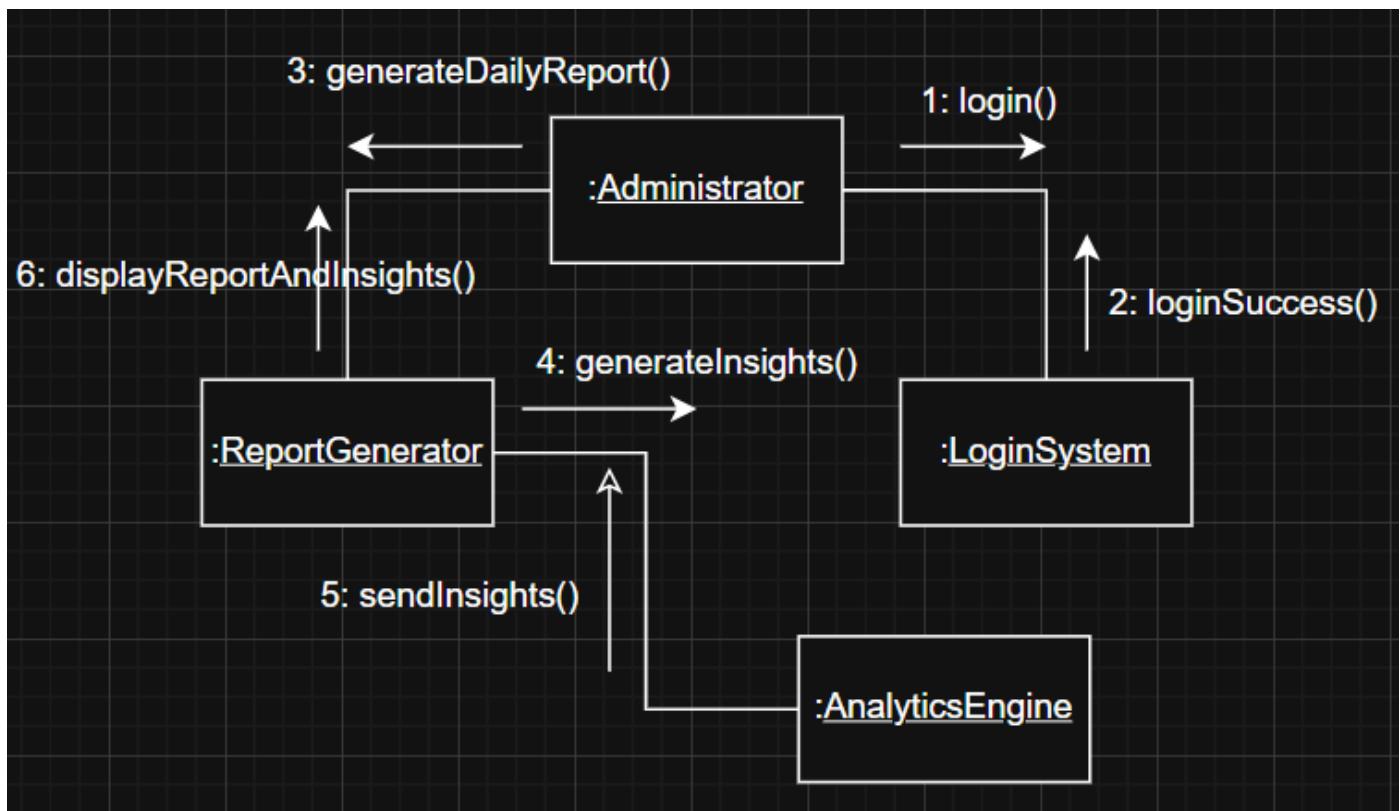


Restaurant Management System Requirements Specification

CC_08: User Access & Security – Alisa Tozaj

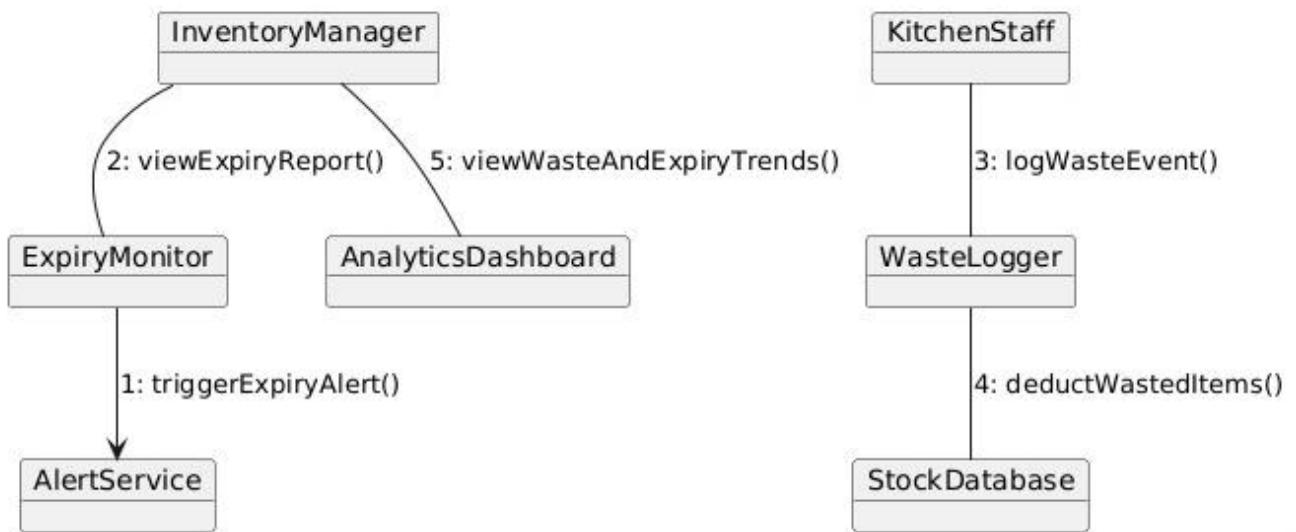


CC_09: Sales & Reports – Alisa Tozaj

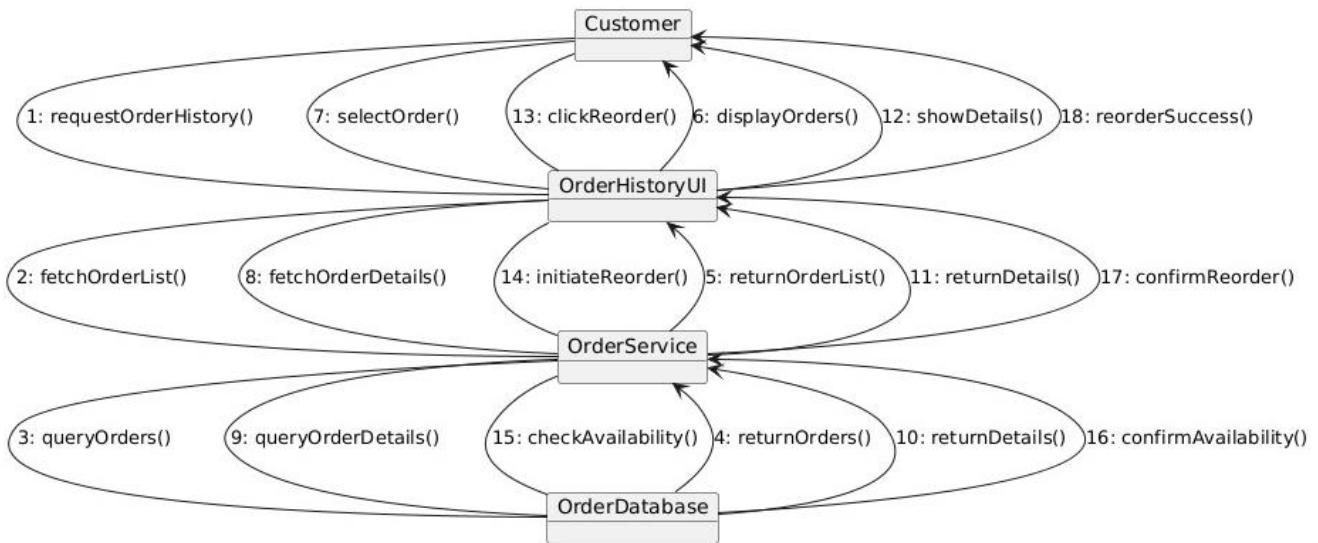


Restaurant Management System Requirements Specification

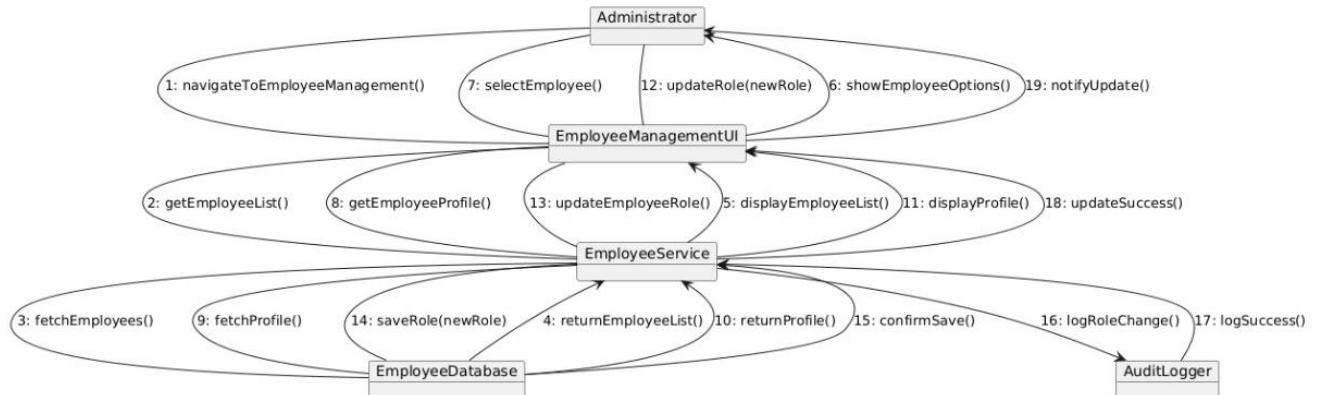
CC_14: Waste & Expiry Tracking – Ester Qershori



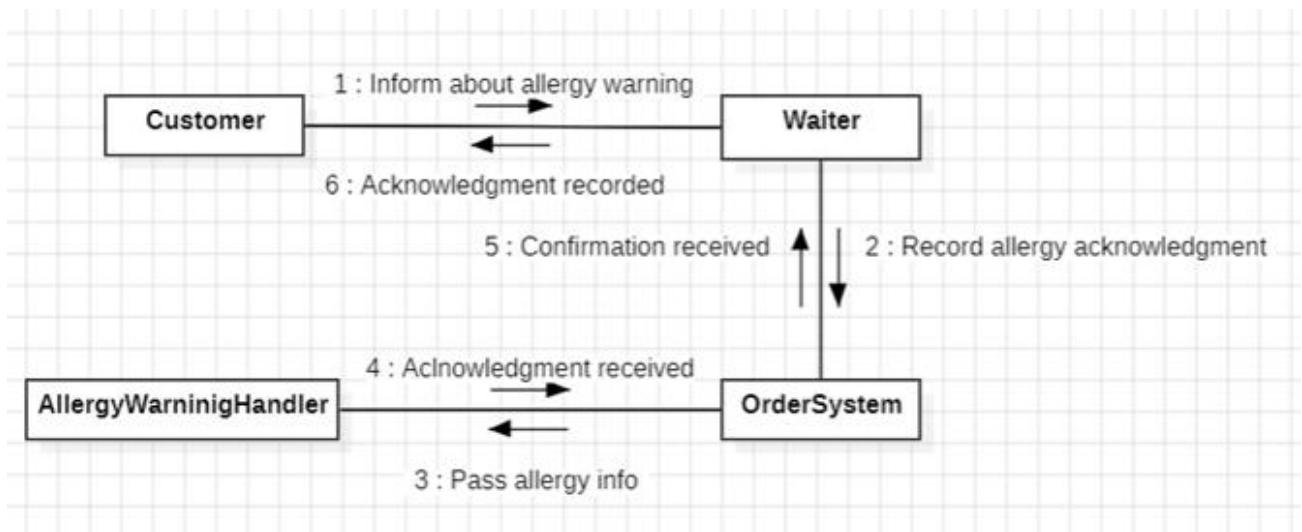
CC_15: Customer Order History Viewer – Ester Qershori



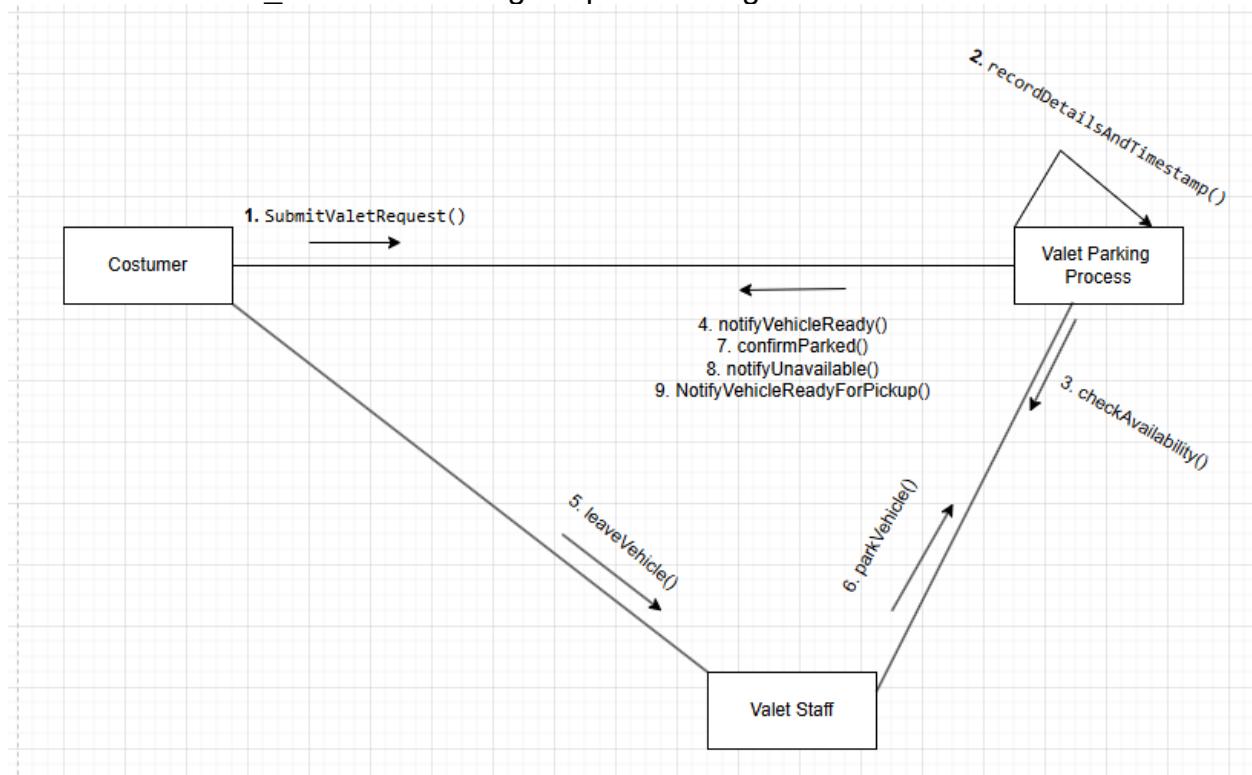
CC_16: Employee Role Assignment – Ester Qershori



Restaurant Management System Requirements Specification
 CC_29: Record Customer Allergy Tracking – Ester Qershori

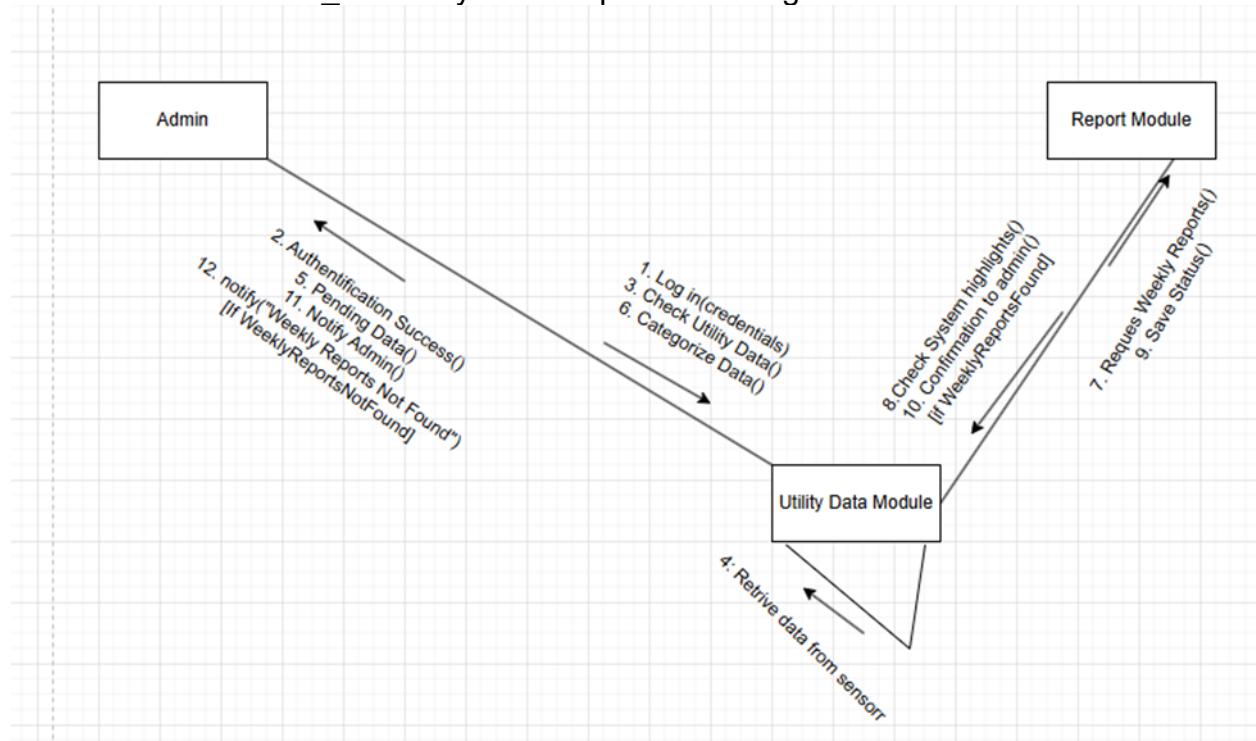


CC_20: Valet Parking Request Management - Erdi Perhati

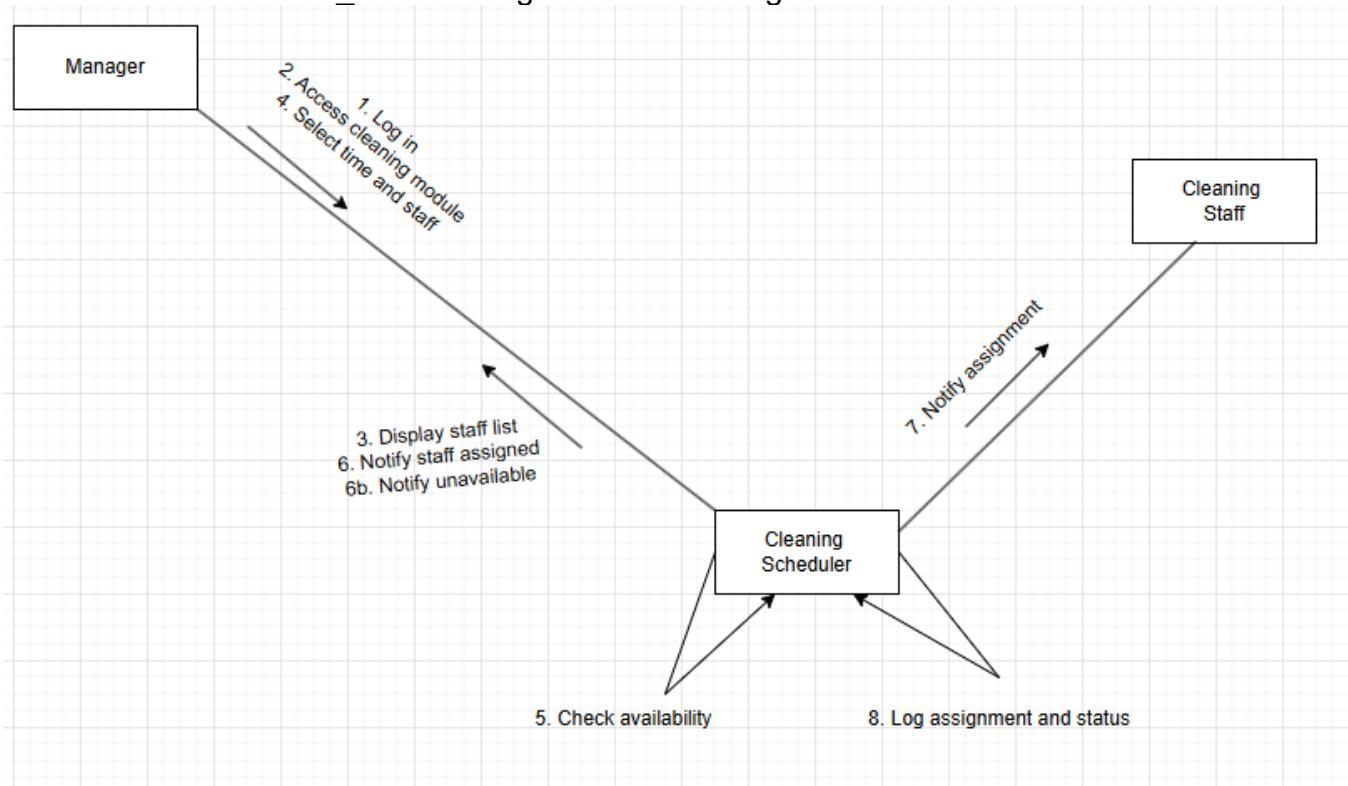


Restaurant Management System Requirements Specification

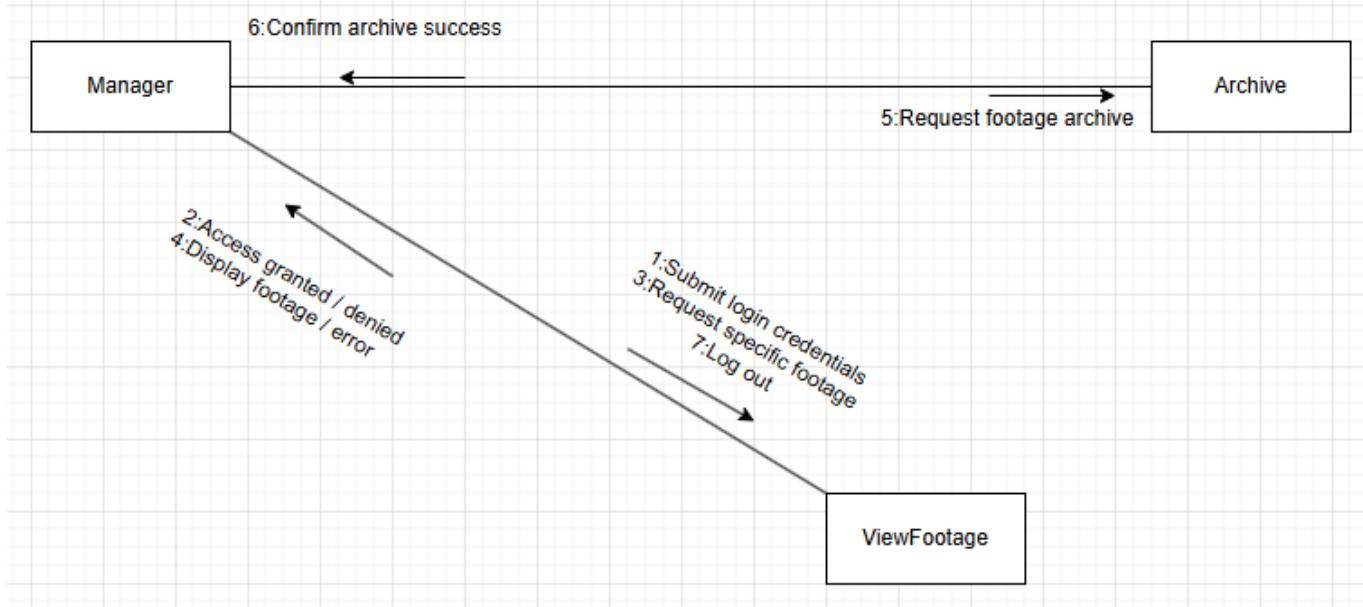
CC_21: Utility Consumption Tracking - Erdi Perhati



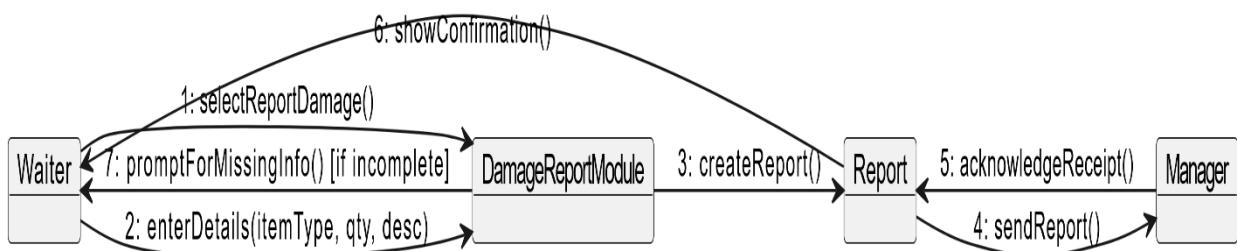
CC_22: Cleaning Schedule Management - Erdi Perhati



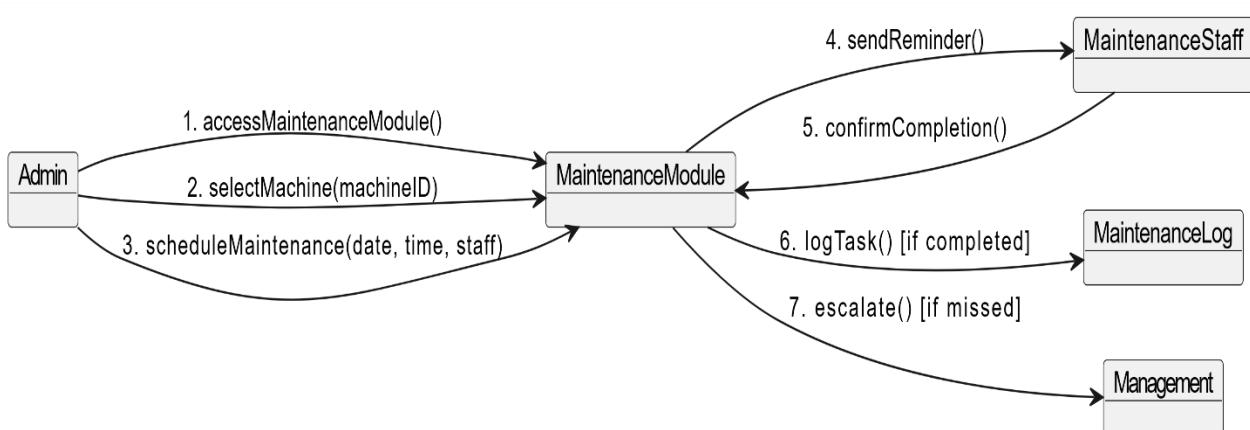
CC_23: Security Camera Footage Log - Erdi Perhati



CC_25: Tableware Damage Reporting – Keisi Loci

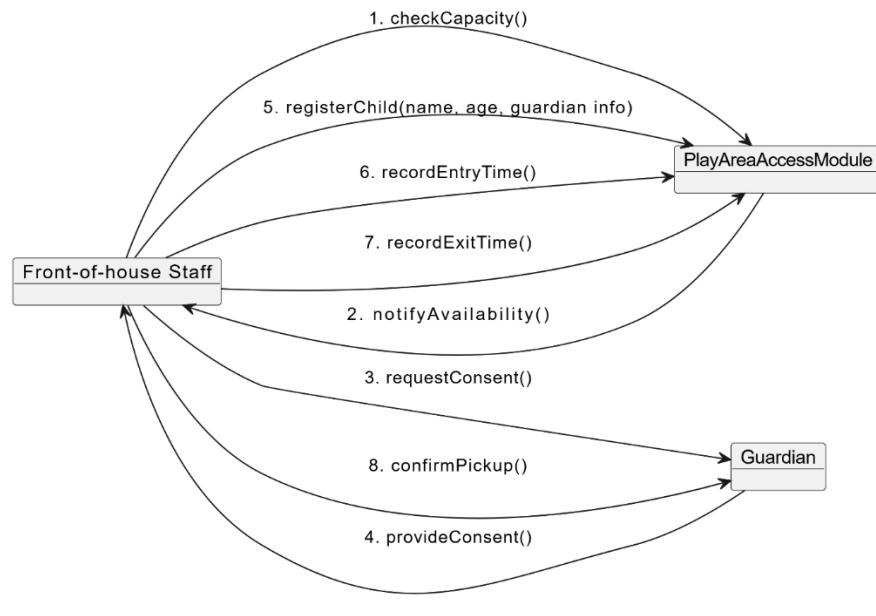


CC_27: Dishwashing Machine Maintenance Scheduling – Keisi Loci

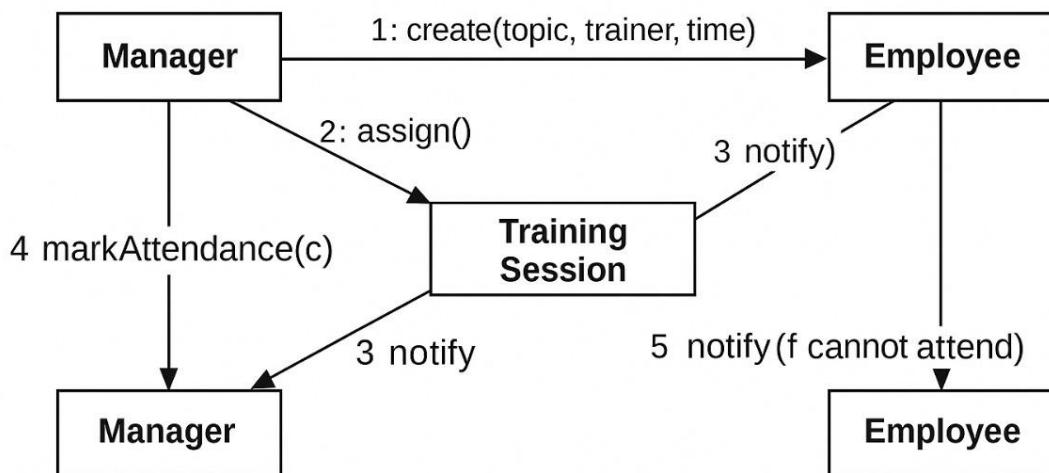


Restaurant Management System Requirements Specification

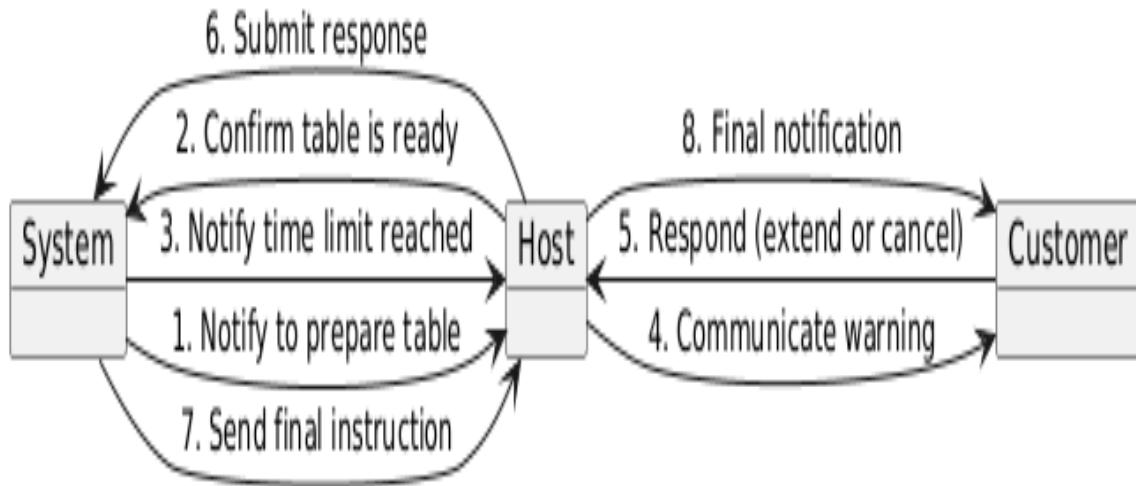
CC_28: Play Area Access Registration – Keisi Loci



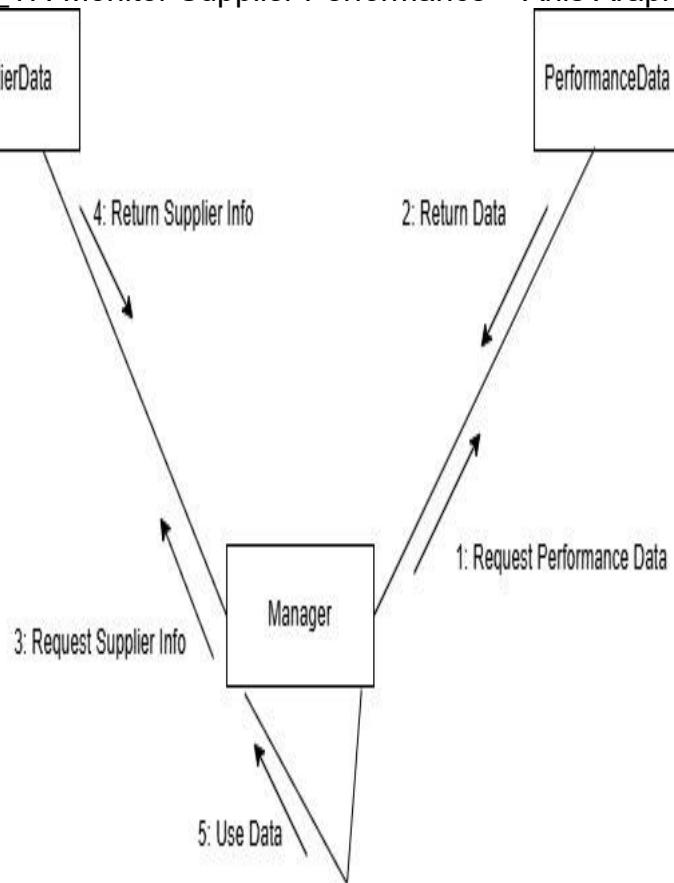
CC_24: Training Session Scheduling – Keisi Loci



Restaurant Management System Requirements Specification
CC_30: Manage Dining Time Limits Warnings – Arlis Arapi

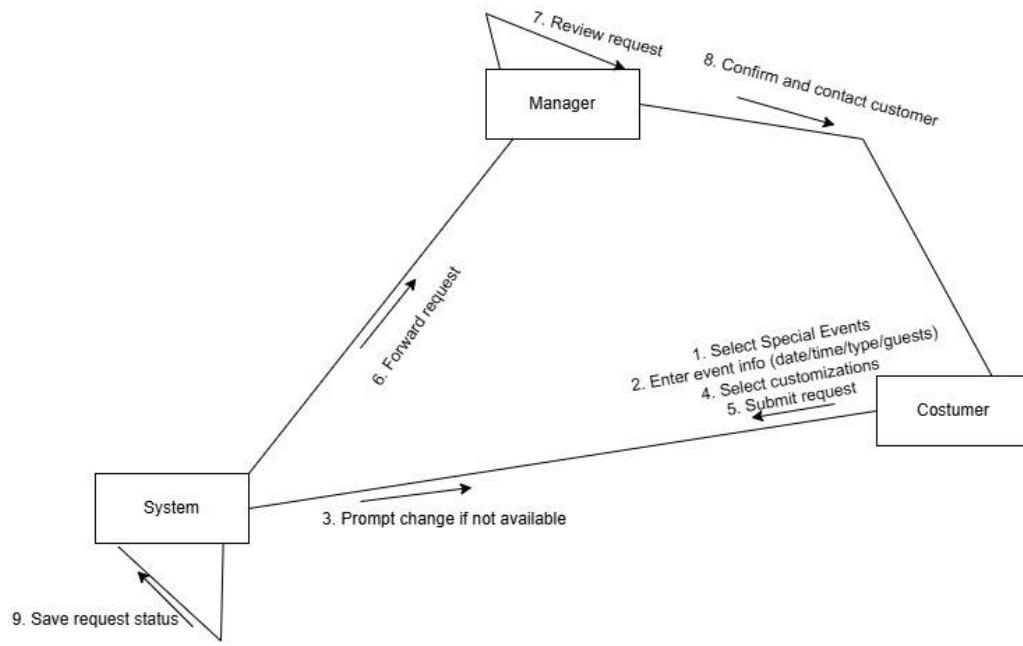


CC_17: Monitor Supplier Performance – Arlis Arapi

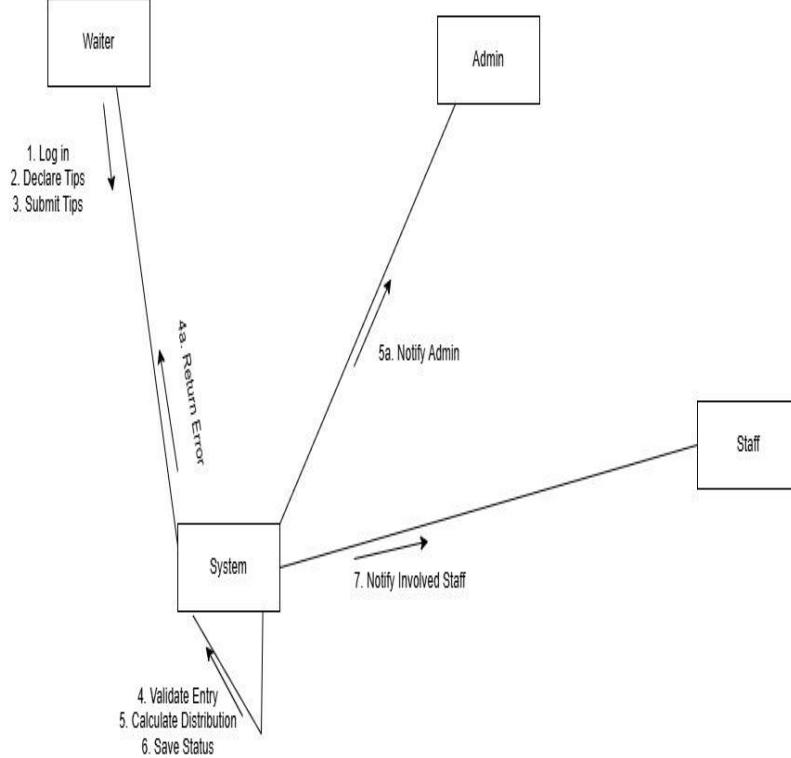


Restaurant Management System Requirements Specification

CC_19: Special Event Booking – Arlis Arapi



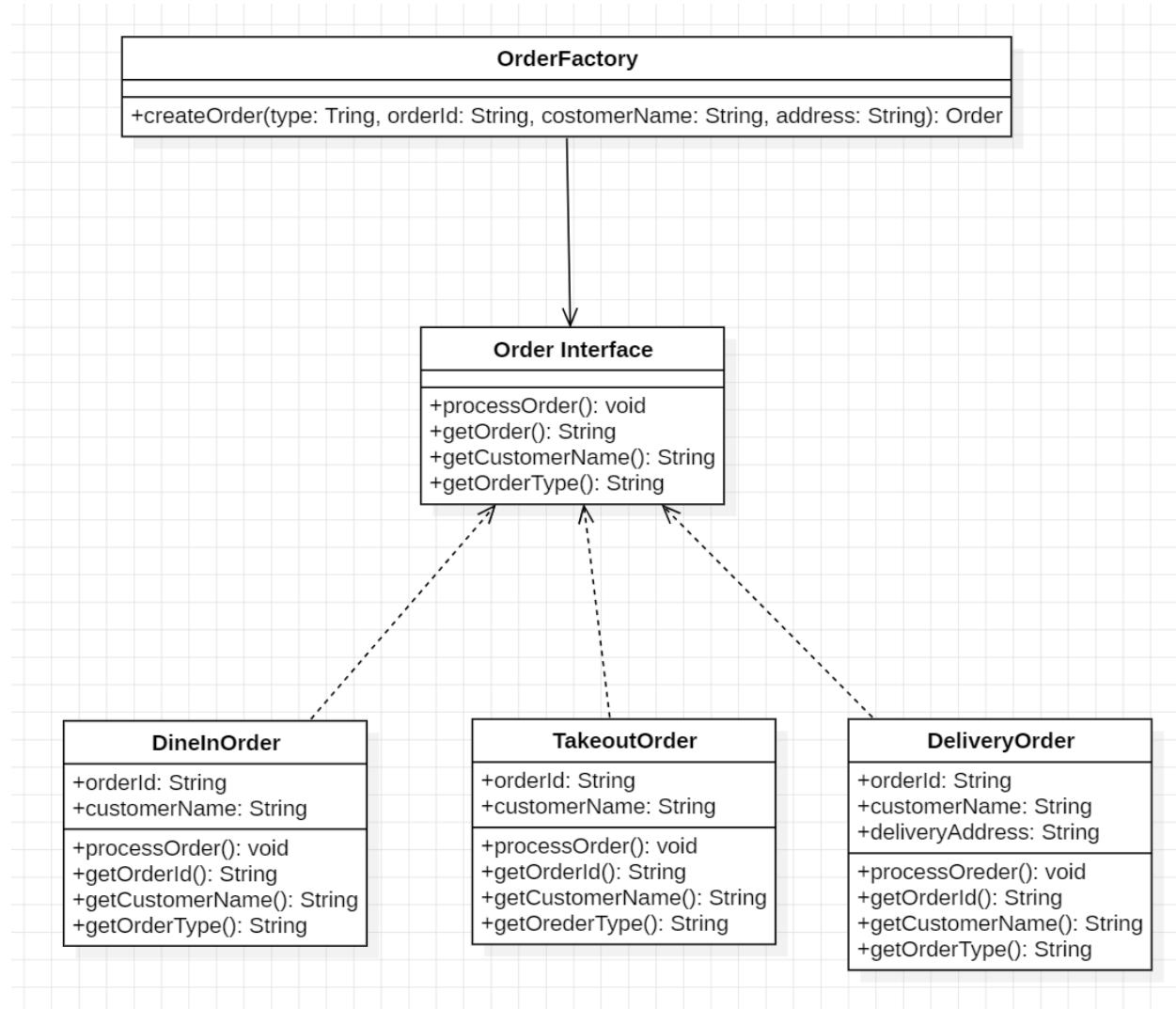
CC_23: Declared Tips & Automated Distribution – Arlis Arapi



6. Design Patterns

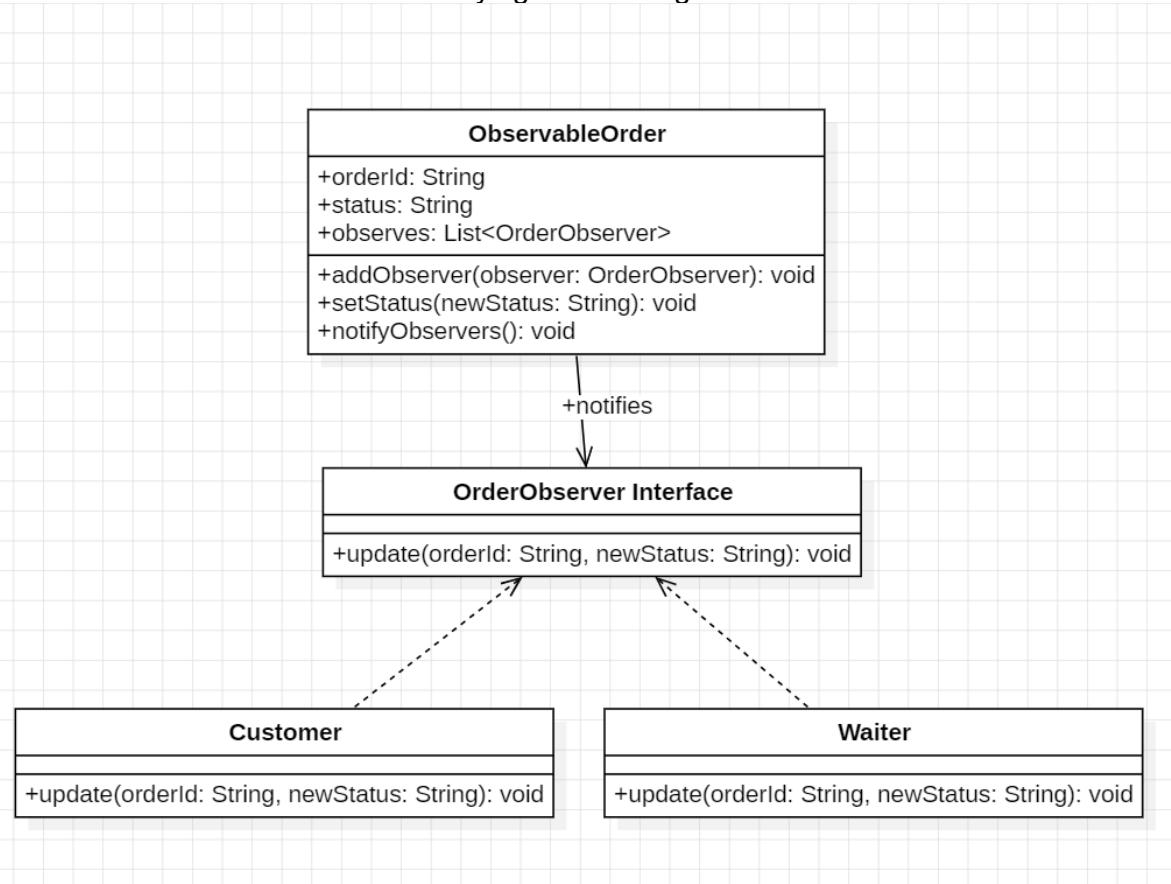
Factory Method Pattern – Joldi Xure

The Factory Method pattern is employed to abstract the process of object creation, allowing the system to instantiate specific order types (such as DineInOrder, TakeoutOrder, or DeliveryOrder) without coupling the core application logic to concrete class implementations. In the context of a restaurant management system (RMS), this approach promotes extensibility and maintainability by centralizing the instantiation logic within a dedicated factory class. This ensures that new order types can be integrated with minimal impact on existing code, in line with the Open/Closed Principle of object-oriented design. It also supports consistent instantiation behavior and reduces duplication, making the system more scalable and easier to evolve over time.



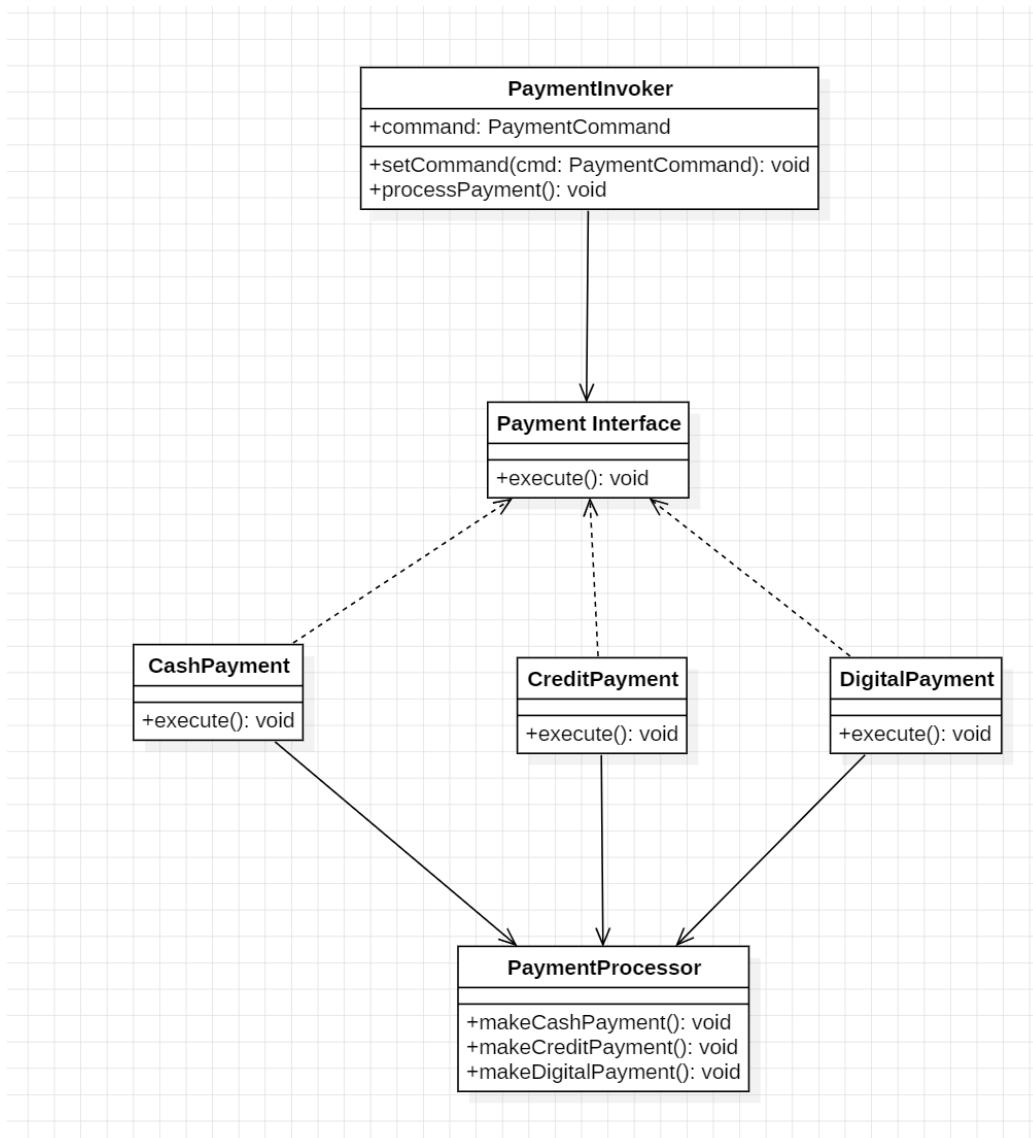
Observer Pattern – Joldi Xure

The Observer pattern is implemented to establish a one-to-many dependency between the order state and its listeners, such that when the status of an order changes (e.g., from "Preparing" to "Ready"), all subscribed entities—such as customers, waitstaff dashboards, or display systems—are automatically notified. This design is critical in a restaurant management system where timely communication and real-time updates are essential for operational efficiency. By decoupling the order management logic from the various user interfaces, the pattern enhances system modularity and scalability, enabling seamless integration of new notification channels or devices without modifying the core logic.



Command Pattern – Joldi Xure

The Command pattern is utilized to encapsulate payment-related actions (e.g., cash, credit card, or digital payment) as standalone command objects that implement a unified interface. This abstraction allows the restaurant management system to decouple the initiation of a payment operation from its execution, providing flexibility in how and when payments are processed. It supports extensible and maintainable transaction handling by isolating payment logic from user interface components, and enables advanced capabilities such as queuing, logging, and future implementation of undo or audit functionality. This design not only ensures clear separation of concerns but also facilitates compliance, traceability, and robust transaction management in high-volume environments.



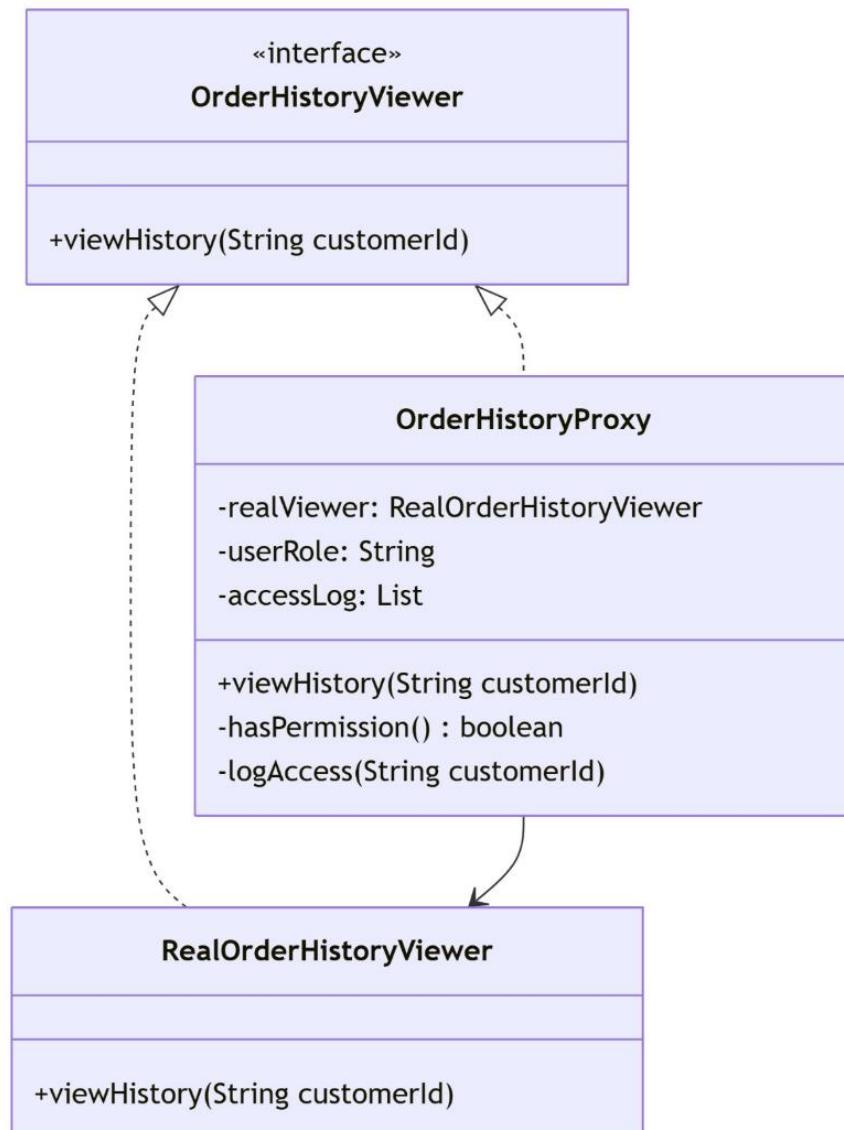
Gloria Traja - Proxy Pattern

Customer Order History Viewer

In this design, the Proxy Pattern is implemented to control access to sensitive customer order history data. The OrderHistoryViewer interface defines the viewHistory(customerId) method, and the OrderHistoryProxy acts as an intermediary between the client and the actual RealOrderHistoryViewer. The proxy checks for user role permissions before granting access and maintains a log of all view requests to support auditing and accountability. This design encapsulates access control and usage tracking while still allowing authorized users full functionality through the real object.

The Proxy Pattern here provides:

- Security, by limiting access based on user roles (e.g., only managers can view history),
- Monitoring, by logging access attempts,
- Separation of concerns, by offloading permission logic from the core data viewer.



Singleton Pattern – Menu Management

In this design, the Singleton Pattern is implemented through the MenuManager class to ensure that only one instance of the menu system exists throughout the entire application. The constructor is made private, preventing external instantiation, and a static method getInstance() provides a global access point. This design guarantees that all components—such as waiters, kitchen staff, and online ordering systems—interact with the same centralized menu, maintaining consistency in menu items. Additionally, MenuManager encapsulates methods like addItem(), removeItem(), updateItem(), and getMenu() to support controlled and synchronized modifications. By using the Singleton Pattern here, the system:

- Avoids duplication of menu data,
- Maintains a single source of truth,
- Simplifies coordination of real-time updates across the restaurant.

