Restaurant Management System

Requirements Specification

Version 2.0

April 3, 2025

**Table of Contents**

[**RESTORANT MANAGEMENT SYSTEM REQUIREMENTS SPECIFICATION 1**](#_heading=h.19c6y18)

[**VERSION 1.0 1**](#_heading=h.3tbugp1)

[**APRIL 19, 2021 1**](#_heading=h.28h4qwu)

[**1.**](#_heading=h.nmf14n) **EXECUTIVE SUMMARY 3**

[1.1](#_heading=h.37m2jsg) Project Overview 3

[1.2](#_heading=h.1mrcu09) Purpose and Scope of this Specification 3

[**2.**](#_heading=h.46r0co2) **PRODUCT/SERVICE DESCRIPTION 3**

[2.1](#_heading=h.2lwamvv) Product Context 3

[2.2](#_heading=h.111kx3o) User Characteristics 3

[2.3](#_heading=h.3l18frh) Assumptions 3

[2.4](#_heading=h.206ipza) Constraints 3

[2.5](#_heading=h.4k668n3) Dependencies 4

[**3.**](#_heading=h.2zbgiuw) **REQUIREMENTS 4**

[3.1](#_heading=h.1egqt2p) Functional Requirements 5

[3.2](#_heading=h.3ygebqi) Non-Functional Requirements 5

[*3.2.1*](#_heading=h.2dlolyb) *Product Requirements 5*

[**3.2.1.1**](#_heading=h.44sinio) **User Interface Requirements** 6

[**3.2.1.2**](#_heading=h.2jxsxqh) **Usability** 6

[**3.2.1.3**](#_heading=h.z337ya) **Efficiency** 6

[3.2.1.3.1](#_heading=h.3j2qqm3) Performance Requirements 6

[3.2.1.3.2](#_heading=h.1y810tw) Space Requirements 6

[**3.2.1.4**](#_heading=h.4i7ojhp) **Dependability** 6

[**3.2.1.5**](#_heading=h.2xcytpi) **Security** 7

[*3.2.2*](#_heading=h.sqyw64) *Organizational Requirements 7*

[**3.2.2.1**](#_heading=h.3whwml4) **Environmental Requirements** 7

[**3.2.2.2**](#_heading=h.2bn6wsx) **Operational Requirements** 7

[**3.2.2.3**](#_heading=h.qsh70q) **Development Requirements** 7

[*3.2.3*](#_heading=h.3cqmetx) *External Requirements 7*

[**3.2.3.1**](#_heading=h.1pxezwc) **Regulatory Requirements** 7

[**3.2.3.2**](#_heading=h.49x2ik5) **Ethical Requirements** 7

[**3.2.3.3**](#_heading=h.2p2csry) **Legislative Requirements** 7

[3.2.3.3.1](#_heading=h.147n2zr) Accounting Requirements 7

[3.2.3.3.2](#_heading=h.3o7alnk) Security Requirements 7

[3.3](#_heading=h.1rvwp1q) Domain Requirements 7

[**4.**](#_heading=h.4bvk7pj) **USER SCENARIOS/USE CASES4. SOFTWARE DESIGN / DIAGRAMS........................................................................23**

**4.1 REQUIREMENTS ANALYSIS .................................................................................................................. 23**

**4.1.1 User Scenarios....................................................................................................................... 23**

**4.1.1.1 User Scenarios List...................................................................................................................... 23**

**4.1.1.2 User Scenarios Extended............................................................................................................ 24**

**4.1.2 User Cases............................................................................................................................. 32**

**4.2 BEHAVIORAL DIAGRAMS..................................................................................................................... 45**

**4.2.1 Use Case Diagrams ............................................................................................................... 45**

**PMS Documentation**

**June 3, 2018 Page 3**

**4.2.2 Activity Diagrams.................................................................................................................. 50 7**

# 

# **4. User Scenarios/Use Cases**

**4.1.1.1 User Scenarios List**

| **Nr** | **Name** | **Description** |
| --- | --- | --- |
| **US\_01** | User logs in | Users (admin, waiter, customer) log in using username and password. |
| **US\_02** | Change password | Users (admin, waiter, customer) change their password. |
| **US\_03** | Add new customer | Admin adds a new customer to the system. |
| **US\_04** | Edit customer information | Admin or waiter edits existing customer details. |
| **US\_05** | Assign table to customer | Admin assigns a table to a customer who has arrived. |
| **US\_06** | Change table assignment | Admin or waiter changes the table assignment for a customer. |
| **US\_07** | View all customers | Admin views a list of all customers. |
| **US\_08** | Search for a customer | Admin or waiter searches for a specific customer. |
| **US\_09** | View orders by customer | Admin or waiter views a customer’s order history. |
| **US\_10** | Add a new menu item | Admin adds a new item to the restaurant menu. |
| **US\_11** | Update menu item | Admin updates the details of an existing menu item. |
| **US\_12** | Remove menu item | Admin removes a menu item from the system. |
| **US\_13** | Place order for customer | Waiter takes a new order from a customer. |
| **US\_14** | Edit order | Waiter or admin edits a customer's order. |
| **US\_15** | Apply discount to an order | Waiter applies a discount to a customer’s order. |
| **US\_16** | Cancel an order | Waiter or admin cancels a customer's order before delivery. |
| **US\_17** | Record customer feedback | Waiter records feedback from customers about their experience. |
| **US\_18** | Generate end-of-day report | Admin generates a report at the end of the day summarizing the operations. |
| **US\_19** | View daily revenue | Admin views the revenue generated on a specific day. |
| **US\_20** | View popular menu items | Admin views which menu items are the most ordered. |
| **US\_21** | Log out | Admin, waiter, or customer logs out from their account. |

**Table 2. User scenarios list**

**4.1.1.2 Extended User Scenarios**

**1. US\_01 – User logs in**  
a. User chooses their user type: admin, waiter, or customer.  
b. User is redirected to the login page.  
c. User enters their username and password.  
d. User checks the reCaptcha ‘I am not a robot’.  
e. User presses the login button.  
f. If data is correct, the user is redirected to their profile page.  
g. If data is not correct, an error message will be shown, and the user repeats the process from step b.

**2. US\_02 – Change password**  
a. User logs in following the steps in US\_01.  
b. User chooses the “Change Password” menu.  
c. User enters their old password and the new password (twice).  
d. User presses the “Save Changes” button.  
e. If the old password is correct and the new passwords match, a message “Password was changed successfully!” will be shown.  
f. If the old password is incorrect or the new passwords don’t match, the user will be alerted: “Password was not changed. Please try again.”

**3. US\_03 – Add new customer**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Add New Customer”.  
c. Admin fills in the required fields (name, contact, etc.).  
d. Admin double-checks the entered data.  
e. Admin clicks “Add Customer”.  
f. An alert appears: “Are you sure you want to continue?” YES/NO.  
g. If YES, the system validates the data and creates the account. A message “Customer added successfully!” will be shown.  
h. If validation fails, an error message will appear, and the admin will correct the issue.

**4. US\_04 – Edit customer information**  
a. Admin or waiter logs in following the steps in US\_01.  
b. Admin or waiter selects the customer to edit from the list.  
c. Admin or waiter edits the customer information (name, address, etc.).  
d. Admin or waiter double-checks the entered data.  
e. Admin or waiter clicks on “Save Changes”.  
f. The system validates the data and saves it if everything is correct. A message “Customer information updated successfully!” will be shown.  
g. If the data is incorrect, an error message will appear, prompting corrections.

**5. US\_05 – Delete customer**  
a. Admin logs in following the steps in US\_01.  
b. Admin selects the customer to delete from the list.  
c. Admin clicks on “Delete Customer”.  
d. A confirmation dialog appears: “Are you sure you want to delete this customer?” YES/NO.  
e. If YES, the system deletes the customer. A message “Customer deleted successfully!” will be shown.  
f. If NO, no changes occur, and the admin returns to the customer list.

**6. US\_06 – Add new waiter**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Add New Waiter”.  
c. Admin enters the waiter’s details (name, contact info).  
d. Admin clicks “Add Waiter”.  
e. A confirmation dialog appears: “Are you sure you want to add this waiter?” YES/NO.  
f. If YES, the system validates the data and adds the waiter. A message “Waiter added successfully!” will be shown.  
g. If validation fails, an error message will appear, and the admin will correct the issue.

**7. US\_07 – Edit waiter information**  
a. Admin logs in following the steps in US\_01.  
b. Admin selects the waiter to edit from the list.  
c. Admin edits the waiter’s information (name, shift details, etc.).  
d. Admin clicks “Save Changes”.  
e. The system validates the data and saves it. A message “Waiter information updated successfully!” will be shown.  
f. If there is an error in the data, an error message will appear.

**8. US\_08 – Delete waiter**  
a. Admin logs in following the steps in US\_01.  
b. Admin selects the waiter to delete from the list.  
c. Admin clicks on “Delete Waiter”.  
d. A confirmation dialog appears: “Are you sure you want to delete this waiter?” YES/NO.  
e. If YES, the system deletes the waiter. A message “Waiter deleted successfully!” will be shown.  
f. If NO, no changes occur, and the admin returns to the waiter list.

**9. US\_09 – Customer views menu**  
a. Customer logs in following the steps in US\_01.  
b. Customer clicks on “Menu” in the navigation bar.  
c. Customer views the list of available items, including categories, prices, and descriptions.  
d. Customer selects a category or item to view detailed information.  
e. If desired, the customer adds the item to their cart by clicking “Add to Cart”.

**10. US\_10 – Customer places an order**  
a. Customer logs in following the steps in US\_01.  
b. Customer browses the menu and adds items to their cart.  
c. Customer reviews the cart and presses “Place Order”.  
d. Customer chooses the payment method and enters payment details.  
e. Customer presses “Confirm Order”.  
f. The system processes the payment, sends an order confirmation, and notifies the waiter.

**11. US\_11 – Waiter views orders**  
a. Waiter logs in following the steps in US\_01.  
b. Waiter clicks on “Orders” in the navigation bar.  
c. Waiter sees a list of orders placed by customers, including their status (pending, completed).  
d. Waiter selects an order to view detailed information (items ordered, customer details).  
e. Waiter can mark the order as completed once served.

**12. US\_12 – Waiter marks an order as completed**  
a. Waiter logs in following the steps in US\_01.  
b. Waiter selects an order from the orders list.  
c. Waiter clicks “Mark as Completed”.  
d. The system updates the order status and notifies the customer of completion.

**13. US\_13 – Admin views restaurant performance**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Performance” in the navigation bar.  
c. Admin views reports on sales, orders, and customer satisfaction.  
d. Admin can filter reports by time period or specific items/services.

**14. US\_14 – Admin views employee performance**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Employee Performance” in the navigation bar.  
c. Admin selects the employee to review.  
d. Admin views the employee’s sales, feedback, and work hours.  
e. Admin can filter performance by time or other parameters.

**15. US\_15 – Admin views customer reviews**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Customer Reviews” in the navigation bar.  
c. Admin views a list of customer reviews with ratings and feedback.  
d. Admin can respond to reviews or mark them as addressed.

**16. US\_16 – Customer views order status**  
a. Customer logs in following the steps in US\_01.  
b. Customer clicks on “My Orders” in the navigation bar.  
c. Customer views the list of their past orders, including status updates (pending, completed, etc.).  
d. Customer can click on an order to view detailed information.

**17. US\_17 – Admin generates sales report**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Reports” and selects “Sales Report”.  
c. Admin specifies the time range for the report.  
d. Admin clicks “Generate Report”.  
e. The system processes the request and displays the sales data.

**18. US\_18 – Admin updates menu items**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Menu” and selects an item to update.  
c. Admin modifies the item details (price, description, availability).  
d. Admin clicks “Save Changes”.  
e. The system validates and updates the menu.

**19. US\_19 – Customer requests order modifications**  
a. Customer places an order following US\_10.  
b. Customer decides to modify the order (e.g., change items, add items).  
c. Customer contacts the waiter through the system chat or call option.  
d. Waiter processes the modification, updates the order, and notifies the customer of changes.

**20. US\_20 – Admin manages system settings**  
a. Admin logs in following the steps in US\_01.  
b. Admin clicks on “Settings” in the navigation bar.  
c. Admin can update general system settings (working hours, notification preferences, etc.).  
d. Admin saves changes and the system applies the updates.

**21. US\_21 – Waiter views customer history**  
a. Waiter logs in following the steps in US\_01.  
b. Waiter clicks on “Customer History”.  
c. Waiter selects a customer and views their order history.  
d. Waiter can see details of past orders, preferences, and feedback.

**4.1.2 User Cases**

## UC\_01 – User logs in

|  |  |
| --- | --- |
| **UC Name** | **UC\_01 - User logs in** |
| **Summary** | This use case describes the process where a user (admin, waiter, or customer) logs in using their username and password. |
| **Dependency** | None |
| **Actors** | Primary Actor: User (admin, waiter, customer) Secondary Actors: None |
| **Preconditions** | The user must have an existing account in the system. The user is on the login page. |
| **Description of the Main Sequence** | User chooses their user type (admin, waiter, or customer). User is redirected to the login page. User enters their username and password. User completes the reCaptcha verification. User presses the login button. If the credentials are correct, the user is redirected to their profile page. |
| **Description of the Alternative Sequence** | User enters incorrect credentials. The system shows an error message ("Invalid username or password"). User is prompted to retry the login process. |
| **Non-functional Requirements** | Performance: The login process should be completed within 2 seconds. Security: The system should ensure encrypted transmission of username and password using SSL/TLS. Availability: The login page must be available 99.9% of the time. |
| **Postconditions** | If the login is successful, the user is directed to their respective profile page. If login fails, the user remains on the login page and receives an error message. |

## UC\_02 – Change password

|  |  |
| --- | --- |
| **UC Name** | **UC\_02 - Change password** |
| **Summary** | This use case allows a user to change their account password. |
| **Dependency** | UC\_01 |
| **Actors** | Primary Actor: User (admin, waiter, customer) Secondary Actors: None |
| **Preconditions** | User is logged in. User is on the profile or settings page. |
| **Description of the Main Sequence** | User clicks "Change Password". User enters old password. User enters new password and confirms it. System validates input. Password is updated successfully. |
| **Description of the Alternative Sequence** | Old password is incorrect. System shows error. User retries. |
| **Non-functional Requirements** | Security: Password must be hashed. Usability: Password criteria are shown clearly. Responsiveness: Confirmation within 2 seconds. |
| **Postconditions** | Password is updated. User uses new password for future logins. |

## UC\_03 - Add new customer

|  |  |
| --- | --- |
| **UC Name** | **UC\_03 - Add new customer** |
| **Summary** | Admin adds a new customer to the system. |
| **Dependency** | None |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. Admin is on the customer management panel. |
| **Description of the Main Sequence** | Admin clicks "Add Customer". Admin enters required details. System validates and saves information. Confirmation is shown. |
| **Description of the Alternative Sequence** | Required fields are missing. Validation fails. Admin is prompted to correct data. |
| **Non-functional Requirements** | Data Integrity: All inputs must be validated. Responsiveness: Customer is added within 2 seconds. |
| **Postconditions** | New customer is added to the database. |

## UC\_04- Edit customer information

|  |  |
| --- | --- |
| **UC Name** | **UC\_04 - Edit customer information** |
| **Summary** | Admin or waiter edits existing customer details. |
| **Dependency** | UC\_03 |
| **Actors** | Primary Actor: Admin / Waiter Secondary Actors: None |
| **Preconditions** | User is logged in. Customer record exists. |
| **Description of the Main Sequence** | Admin/waiter searches for customer. Selects the customer from list. Edits required information. System validates and updates data. Success message shown. |
| **Description of the Alternative Sequence** | Customer not found. Validation fails. No changes saved. |
| **Non-functional Requirements** | Auditability: Changes should be logged. Validation: Fields must meet data rules. |
| **Postconditions** | Customer record is updated in the system. |

***UC\_05 - Assign table to customer***

|  |  |
| --- | --- |
| **UC Name** | **UC\_05 - Assign table to customer** |
| **Summary** | Admin assigns a table to a customer who has arrived. |
| **Dependency** | UC\_03 |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. Customer is registered and present in the system. |
| **Description of the Main Sequence** | Admin selects customer. Admin clicks "Assign Table". Admin selects available table. System updates table status. Confirmation is shown. |
| **Description of the Alternative Sequence** | No tables available. Admin notified to wait or choose later. |
| **Non-functional Requirements** | Real-time updates on table status. Responsiveness within 2 seconds. |
| **Postconditions** | Table is marked as occupied and linked to customer. |

***UC\_06 - Change table assignment***

|  |  |
| --- | --- |
| **UC Name** | **UC\_06 - Change table assignment** |
| **Summary** | Admin or waiter changes the table assignment for a customer. |
| **Dependency** | UC\_05 |
| **Actors** | Primary Actor: Admin / Waiter Secondary Actors: None |
| **Preconditions** | User is logged in. Customer already has a table assigned. |
| **Description of the Main Sequence** | User selects customer. Clicks “Change Table”. Selects new table. System updates the table record. Confirmation shown. |
| **Description of the Alternative Sequence** | New table unavailable. System shows warning. Admin selects another table. |
| **Non-functional Requirements** | Table update should reflect in real-time. System should prevent conflicts. |
| **Postconditions** | Customer is moved to new table. Old table becomes available. |

***UC\_07 – View all customers***

|  |  |
| --- | --- |
| **UC Name** | **UC\_07 - View all customers** |
| **Summary** | Admin views a list of all customers. |
| **Dependency** | UC\_03 |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. |
| **Description of the Main Sequence** | Admin clicks “View Customers”. System retrieves customer data. List is displayed with search and sort options. |
| **Description of the Alternative Sequence** | No customers found. System shows empty list or message. |
| **Non-functional Requirements** | Must support pagination for large data. Search must be responsive. |
| **Postconditions** | Admin has access to full customer list. |

***UC\_08 – Search for a customer***

|  |  |
| --- | --- |
| **UC Name** | **UC\_08 - Search for a customer** |
| **Summary** | Admin or waiter searches for a specific customer. |
| **Dependency** | UC\_07 |
| **Actors** | Primary Actor: Admin / Waiter Secondary Actors: None |
| **Preconditions** | User is logged in. Customers exist in the database. |
| **Description of the Main Sequence** | User enters search criteria. System filters customer list. Matching results shown instantly. |
| **Description of the Alternative Sequence** | No match found. System displays “No results”. |
| **Non-functional Requirements** | Search must be case-insensitive. Partial match support. Search within 1 second. |
| **Postconditions** | Matching customer record is displayed. |

***UC\_09 – View orders by customer***

|  |  |
| --- | --- |
| **UC Name** | **UC\_09 - View orders by customer** |
| **Summary** | Admin or waiter views a customer’s order history. |
| **Dependency** | UC\_03 |
| **Actors** | Primary Actor: Admin / Waiter Secondary Actors: None |
| **Preconditions** | User is logged in. Customer exists and has placed orders. |
| **Description of the Main Sequence** | User searches for a customer. Selects “View Orders”. System displays order history with date and status. |
| **Description of the Alternative Sequence** | No orders found. System shows message “No orders available”. |
| **Non-functional Requirements** | Search must be responsive. Results must load under 2 seconds. |
| **Postconditions** | Order list is shown to the user. |

## UC\_10 - Add a new menu item

|  |  |
| --- | --- |
| **UC Name** | **UC\_10 - Add a new menu item** |
| **Summary** | Admin adds a new item to the restaurant menu. |
| **Dependency** | None |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. |
| **Description of the Main Sequence** | Admin navigates to menu management. Clicks “Add New Item”. Fills in item name, category, price, and description. System validates input and saves item. Success message appears. |
| **Description of the Alternative Sequence** | Fields are incomplete or invalid. Admin corrects and retries. |
| **Non-functional Requirements** | Fields must be validated. Item added within 2 seconds. |
| **Postconditions** | New item is added to the menu and visible to customers. |

## UC\_11 - Update menu item

|  |  |
| --- | --- |
| **UC Name** | **UC\_11 - Update menu item** |
| **Summary** | Admin updates the details of an existing menu item. |
| **Dependency** | UC\_10 |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. Item exists in the menu. |
| **Description of the Main Sequence** | Admin searches for item. Clicks “Edit”. Updates desired fields. Saves changes. Confirmation appears. |
| **Description of the Alternative Sequence** | Item not found. System shows error. Admin retries. |
| **Non-functional Requirements** | All fields must support edit. System confirms change within 2 seconds. |
| **Postconditions** | Item details are updated. |

## UC\_12 - Remove menu item

|  |  |
| --- | --- |
| **UC Name** | **UC\_12 - Remove menu item** |
| **Summary** | Admin removes a menu item from the system. |
| **Dependency** | UC\_10 |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. Item exists. |
| **Description of the Main Sequence** | Admin searches for item. Clicks “Delete”. Confirms deletion. System removes item. Confirmation appears. |
| **Description of the Alternative Sequence** | Admin cancels deletion. No changes are made. |
| **Non-functional Requirements** | Deletion is confirmed instantly. Item removed from list and database. |
| **Postconditions** | Item is deleted and no longer visible in menu. |

## UC\_13 - Place order for customer

|  |  |
| --- | --- |
| **UC Name** | **UC\_13 - Place order for customer** |
| **Summary** | Waiter takes a new order from a customer. |
| **Dependency** | UC\_05 |
| **Actors** | Primary Actor: Waiter Secondary Actors: None |
| **Preconditions** | Waiter is logged in. Customer is seated at a table. |
| **Description of the Main Sequence** | Waiter opens active table list. Selects the table/customer. Adds menu items to the order. System stores order in progress. Order sent to kitchen. |
| **Description of the Alternative Sequence** | Menu item unavailable. Waiter informed to remove or replace item. |
| **Non-functional Requirements** | Orders must sync instantly with kitchen. System must auto-save drafts. |
| **Postconditions** | Order is placed and visible in kitchen display. |

## UC\_14 - Edit order

|  |  |
| --- | --- |
| **UC Name** | **UC\_14 - Edit order** |
| **Summary** | Waiter or admin edits a customer's order. |
| **Dependency** | UC\_13 |
| **Actors** | Primary Actor: Waiter / Admin Secondary Actors: None |
| **Preconditions** | Order has already been placed. User is logged in. |
| **Description of the Main Sequence** | User opens current order. Clicks "Edit". Adds/removes menu items. Confirms changes. System updates the order. |
| **Description of the Alternative Sequence** | Item not available. System prevents changes. User chooses alternative. |
| **Non-functional Requirements** | Order changes must reflect in real time. Kitchen must be notified immediately. |
| **Postconditions** | Order is updated and re-sent to kitchen. |

## UC\_15 - Apply discount to an order

|  |  |
| --- | --- |
| **UC Name** | **UC\_15 - Apply discount to an order** |
| **Summary** | Waiter applies a discount to a customer's order. |
| **Dependency** | UC\_13 |
| **Actors** | Primary Actor: Waiter Secondary Actors: Admin (approval if needed) |
| **Preconditions** | Order is placed and unpaid. Waiter is logged in. |
| **Description of the Main Sequence** | Waiter selects order. Clicks “Apply Discount”. Enters discount value. Confirms discount. System recalculates total. |
| **Description of the Alternative Sequence** | Discount exceeds limit. Admin approval requested. If denied, waiter tries again. |
| **Non-functional Requirements** | Discount logic must follow business rules. Discount must be reflected in final bill. |
| **Postconditions** | Order total is updated with discount applied. |

## UC\_16 - Cancel an order

|  |  |
| --- | --- |
| **UC Name** | **UC\_16 - Cancel an order** |
| **Summary** | Waiter or admin cancels a customer's order before delivery. |
| **Dependency** | UC\_13 |
| **Actors** | Primary Actor: Waiter / Admin Secondary Actors: None |
| **Preconditions** | Order must be pending or in process. User is logged in. |
| **Description of the Main Sequence** | User opens the order. Clicks “Cancel”. Confirms cancellation. System removes order from active list. Notification sent to kitchen. |
| **Description of the Alternative Sequence** | Order already prepared. System prevents cancellation. Refund procedure triggered. |
| **Non-functional Requirements** | System must log cancellation reason. Audit trail enabled. |
| **Postconditions** | Order is marked as cancelled and no longer processed. |

## UC\_17 - Record customer feedback

|  |  |
| --- | --- |
| **UC Name** | **UC\_17 - Record customer feedback** |
| **Summary** | Waiter records feedback from customers about their experience. |
| **Dependency** | UC\_13 |
| **Actors** | Primary Actor: Waiter Secondary Actors: None |
| **Preconditions** | Waiter is logged in. Customer has completed an order. |
| **Description of the Main Sequence** | Waiter accesses the customer record. Clicks “Add Feedback”. Inputs customer comments. Saves feedback. System stores it under customer's profile. |
| **Description of the Alternative Sequence** | Feedback is incomplete. Waiter receives warning. Waiter corrects and saves again. |
| **Non-functional Requirements** | Data should be timestamped. Feedback must be editable. |
| **Postconditions** | Feedback is linked to the customer and stored in system. |

## UC\_18 - Generate end-of-day report

|  |  |
| --- | --- |
| **UC Name** | **UC\_18 - Generate end-of-day report** |
| **Summary** | Admin generates a report summarizing the day’s operations. |
| **Dependency** | Multiple UC dependencies (orders, revenue, feedback, etc.) |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. System contains daily data. |
| **Description of the Main Sequence** | Admin opens Reports section. Selects “End-of-Day Report”. Chooses date. System gathers and summarizes data. Report is displayed/downloadable. |
| **Description of the Alternative Sequence** | No data for the date. System shows message “No report available”. |
| **Non-functional Requirements** | Report must generate within 5 seconds. Supports export to PDF. |
| **Postconditions** | Report is generated and accessible to admin. |

## UC\_19 - View daily revenue

|  |  |
| --- | --- |
| **UC Name** | **UC\_19 - View daily revenue** |
| **Summary** | Admin views the revenue generated on a specific day. |
| **Dependency** | UC\_18 |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. System contains revenue data. |
| **Description of the Main Sequence** | Admin goes to Reports. Selects “View Revenue”. Enters date. Revenue data is shown. |
| **Description of the Alternative Sequence** | No revenue recorded. System shows message. |
| **Non-functional Requirements** | Visual summary via chart. Accurate and real-time display. |
| **Postconditions** | Admin views financial summary for the selected day. |

## UC\_20 - View popular menu items

|  |  |
| --- | --- |
| **UC Name** | UC\_20 - View popular menu items |
| **Summary** | Admin views which menu items are the most ordered. |
| **Dependency** | UC\_13 |
| **Actors** | Primary Actor: Admin Secondary Actors: None |
| **Preconditions** | Admin is logged in. System tracks order frequency. |
| **Description of the Main Sequence** | Admin opens statistics. Selects “Popular Items”. System displays items ordered most. Data includes quantities and timespan. |
| **Description of the Alternative Sequence** | No data available. System shows empty chart. |
| **Non-functional Requirements** | Must show top items within 2 seconds. Data should auto-update daily. |
| **Postconditions** | Admin can analyze menu item performance. |

## UC\_21 - Log out

|  |  |
| --- | --- |
| **UC Name** | **UC\_21 - Log out** |
| **Summary** | Admin, waiter, or customer logs out from their account. |
| **Dependency** | UC\_01 |
| **Actors** | Primary Actor: User (admin, waiter, customer) Secondary Actors: None |
| **Preconditions** | User is logged in. |
| **Description of the Main Sequence** | User clicks on “Logout”. System terminates session. Redirects to home/login page. |
| **Description of the Alternative Sequence** | Session expires automatically. System logs user out. |
| **Non-functional Requirements** | Logout must clear session data immediately. Session timeout after inactivity. |
| **Postconditions** | User session is closed and access is revoked. |