**Use Cases - Summary:**

**Waiter**

1. Start shift
2. Open order
3. The waiter chooses a table and provides a number of guests. The system marks the

table as occupied, assigns a number of guests, and creates an empty order assigned

to the waiter that created it.

1. Add item to order

The waiter selects the order and an item. The system adds the item to the order.

1. Remove item from order\*

The waiter selects the order and an item. If the selected item is not part of the already confirmed order, the system removes the item from the order. Otherwise, the system does not remove the item and informs the waiter that a higher authorisation level is required for this action.

1. Confirm order

The waiter selects the order. The system marks all the unconfirmed items in the order as confirmed, updates product stock level, marks the order as confirmed, and sends:

* newly confirmed food items to the kitchen printer
* newly confirmed drink items to the bar printer

1. Take payment

The waiter chooses a table and provides the amount paid. The system deducts the amount from the order balance. If the balance reaches £0, the system marks the order as closed and the table as available.

**Manager**

1. Add table

The manager selects an available table number and sets the maximum capacity. The system adds a new table to the table list.

1. Remove table

The manager selects a table from the table list. The system checks that the table is

not currently linked to any active order. If it isn’t, the system removes the selected

table from the table list.

1. Update product stock level

The manager selects a product and provides a stock level. The system updates the

stock level assigned to the product.

1. Transfer order ownership

The manager selects an order and a waiter. The system reassigns the ownership of the order to the selected waiter.

1. Apply discount

The manager selects an active order and provides the discount value. The system applies the discount to the table.

1. Remove item from order\*

The manager selects an order and an item. The system removes the selected item from the table. If the item has already been confirmed, the system adds the product to the waste list.

1. Close waiter shift

The manager selects a waiter. The system closes the selected waiter’s session (shift), and displays and records the end of shift report. The end of shift report contains:

* total payments taken
* total cash payments taken
* average spend per cover
* time shift started
* time shift closed (now)

**Other:**

1. List tables

When required, the system displays all tables. The displayed information should include:

* whether a table is available (not assigned to any active orders)
* table number
* table capacity (maximum number of covers)

1. List active orders

When required, the system displayed all active orders. The displayed information should include:

* table number
* name and surname of the waiter who owns the order

1. List food items

When required, the system displays a list of all food items. The displayed information should include:

* name
* type (starter, side, main, dessert)

1. List food item details

The user selects a food item. The system displays food item details. The displayed

information should include:

* name
* price
* number of calories
* whether it’s vegan
* whether it’s vegetarian
* whether it contains allergens
* item’s stock level

1. List drink items

When required, the system displays a list of all drink items. The displayed information

should include:

* name
* whether it’s age restricted

1. List drink item details

The user selects a drink item. The system displays drink item details. The displayed

information should include:

* name
* price
* number of calories
* whether it contains allergens
* item’s stock level

1. List active waiters

When required, the system displays a list of all active waiters. The displayed

information should include:

* id
* name and surname
* number of active orders owned
* time shift started

1. List items on order

When required, the system displays a list of all active waiters. The displayed

information should include:

* name
* category (food / drink)
* type - food items only (starter, side, main, dessert)
* whether it’s age restricted - drink items only
* whether it’s validated

**Use Cases - Detailed:**

**Identifier and name** UC1 *Open order*

**Initiator** *Waiter* or *Manager*

**Goal** A new order is opened.

**Precondition** None.

**Postcondition** The new order will have been created, assigned to the

selected table and the user who created it.

**Assumptions** The expected initiator is a waiter or a manager.

**Main success scenario**

1. The waiter/manager selects an available table.
2. The waiter/manager confirms the number of covers with the guests and enters it into the system.
3. The system opens a new order, assigns it to the selected table, sets the number of covers, and sets the user as the owner of the order.
4. The waiter/manager sits the quests at the selected table.

**Extensions:**

1.a. *There are no available tables.* The system informs the user and terminates the use case.

**Identifier and name** UC2 *Add item to order*

**Initiator** *Waiter* or *Manager*

**Goal** An item is added to an order.

**Precondition** The order is open.

**Postcondition** The product will have been added to the selected order. Both

the order and the products will have been marked as

unconfirmed.

**Assumptions** The expected initiator is a waiter or a manager.

**Main success scenario**

1. A guest requests the item to be added to their order.
2. The waiter/manager selects the correct order (if it’s not already selected) and item.
3. The system adds the item to the selected order.
4. The system marks both the new item and the order as unconfirmed.

**Extensions:**

2.a. *Product out of stock*. The selected product is not available. The system informs the

user and terminates the use case

**Identifier and name** UC3a *Remove item from order*

**Initiator** *Waiter*

**Goal** An item is removed from an order.

**Precondition** The item is listed in the order.

**Postcondition** The item will have been removed from the selected order.

**Assumptions** The expected initiator is a waiter.

**Main success scenario**

1. The waiter selects the order and the item to be removed.
2. The system removes the item from the order.

**Extensions:**

1.a. *Product is marked as confirmed.* The system informs the user that this operation

needs to be requested by a manager and terminates the use case.

2.a *No unconfirmed products left on the order.* The system marks the order as

confirmed.

**Identifier and name** UC3b *Remove item from order*

**Initiator** *Manager*

**Goal** An item is removed from an order.

**Precondition** The item is listed in the order.

**Postcondition** The item will have been removed from the selected order.

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager selects the order and the item to be removed.
2. The system removes the item from the order.

**Extensions:**

1.a. *Product marked as confirmed.* The system informs the user that the item has been

confirmed and requested confirmation. If approved, the system adds the item to

the waste list and continues at step 2.

**Identifier and name** UC4 *Confirm order*

**Initiator** *Waiter* or *Manager*

**Goal** An order is confirmed and all of its previously unconfirmed

items are confirmed.

**Precondition** The order is active.

**Postcondition** The order will have been marked as confirmed. All previously

unconfirmed items will have been sent to the bar and the

kitchen for processing.

**Assumptions** The expected initiator is a waiter or a manager.

**Main success scenario**

1. The waiter/manager selects an order.
2. The system sends:
   1. All the drink items marked as unconfirmed to the bar.
   2. All the food items marked as unconfirmed to the kitchen.
3. The system marks all unconfirmed items as confirmed and updates their stock levels.
4. The system marks the order as confirmed.

**Extensions:**

1.a. *Order marked as confirmed.* The system informs the user that the order has been

previously confirmed and terminates the use case

**Identifier and name** UC5 *Take payment*

**Initiator** *Waiter* or *Manager*

**Goal** The payment for the order is accepted.

**Precondition** The order is active.

**Postcondition** The amount paid will have been deducted from the order’s

outstanding balance. If the balance reaches 0, the order will

have been closed.

**Assumptions** The expected initiator is a waiter or a manager.

**Main success scenario**

1. A customer passes a cash payment or a card to a waiter.
2. The waiter registered the amount paid with the system.
3. The system deducts the amount paid from the order’s outstanding balance.

**Extensions:**

3.a. *Order fully paid.* When the balance reaches 0 the system informs the user that the

order has been fully paid for and closes the order.

**Identifier and name** UC6 *Add table*

**Initiator** *Manager*

**Goal** A new table is added.

**Precondition** None.

**Postcondition** The new table will have been created and added to the

available table list.

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager provides the table number.
2. The system creates a new table and adds it to the table list.

**Extensions:**

2.a. *Table number already in use*. The system informs the user, asks them to provide

a different table number, and continues at step 2.

**Identifier and name** UC7 *Remove table*

**Initiator** *Manager*

**Goal** A table is removed.

**Precondition** None.

**Postcondition** The new table will have been created and added to the

available table list.

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager provides a table number.
2. The system identifies a table and removes it from the table list.

**Extensions:**

2.a. *No table identified*. The system informs the user, asks them to provide

a different table number, and continues at step 2.

**Identifier and name** UC7 *Update stock level*

**Initiator** *Manager*

**Goal** A product stock level is updated

**Precondition** The product is listed in the product list. The new stock level is

larger than 0.

**Postcondition** The new stock level will have been assigned to the product

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager selects the product.
2. The manager provides the new stock level.
3. The system updates the stock level for the selected product.

**Extensions:**

None

**Identifier and name** UC8 *Transfer order ownership*

**Initiator** *Manager*

**Goal** A table is transferred to the selected waiter.

**Precondition** Selected order is active. Selected waiter is currently on shift.

**Postcondition** The order ownership will have been transferred to the selected

waiter.

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager selects the order and the waiter.
2. The system transfers the order ownership to the selected waiter.

**Extensions:**

1.a. *Selected waiter is the current owner of the order*. The system informs the user, asks

them to select a different waiter, and continues at step 2.

1.b *One or more payments have already been taken against the order.* The system

informs the user and terminates the use case.

**Identifier and name** UC9 *Apply discount*

**Initiator** *Manager*

**Goal** A discount is applied to the order.

**Precondition** The order is active.

**Postcondition** The order’s outstanding balance is reduced according to the

discount value.

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager selects an order and the nominal discount value in pounds.
2. The system applies the discount to the remaining balance.

**Extensions:**

1.a. *Discount value larger than outstanding balance*. The system informs the user, asks them to select a smaller value, and continues at step 2.

**Identifier and name** UC10 *Close waiter’s session*

**Initiator** *Manager*

**Goal** The waiter’s session is closed.

**Precondition** The waiter has an active session.

**Postcondition** The waiter session will have been closed. The end of session

report will have been created.

**Assumptions** The expected initiator is a manager.

**Main success scenario**

1. The manager selects a waiter.
2. The system checks whether there are any active orders currently assigned to the waiter.
3. The system closes the waiter’s session
4. The system displays and records the waiter’s end of session report, including
   * the total card payments taken
   * the total cash payments taken
   * the average spend per cover
   * the time the shift was started, and the time the shift was ended (which is the current time)

**Extensions:**

2.a. *Waiter has an outstanding open order*. The system informs the user that

the selected waiter has one or more open orders assigned to their account. The

system terminates the use case.

**Identifier and name** UC11 *Add ingredient to beverage item*

**Initiator** *Waiter*

**Goal** An additional ingredient is added to the selected beverage

item.

**Precondition** The selected beverage item has been added to the order.

The selected additional ingredient is in stock.

The selected beverage item has not yet been confirmed.

**Postcondition** The additional ingredient will have been added to the selected

beverage item.

**Assumptions** The expected initiator is a waiter.

**Main success scenario**

1. The waiter selects a beverage item listed on the selected order.
2. The system displays the list of available additional ingredients.
3. The waiter chooses one or more ingredients to be added and saves the selection.
4. The system adds the ingredients to the selected item and updates the item’s:
   * Price
   * ‘Is age restricted?’ status
5. The system updates the stock levels for the selected additional ingredients

**Identifier and name** UC12 *Add ingredient to food item*

**Initiator** *Waiter*

**Goal** An additional ingredient is added to the selected food

item.

**Precondition** The selected food item has been added to the order.

The selected additional ingredient is in stock.

The selected food item has not yet been confirmed.

**Postcondition** The additional ingredient will have been added to the selected

food item.

**Assumptions** The expected initiator is a waiter.

**Main success scenario**

1. The waiter selects a food item listed on the selected order.
2. The system displays the list of available additional ingredients.
3. The waiter chooses one or more ingredients to be added and saves the selection.
4. The system adds the ingredients to the selected food item and updates the item’s:
   * Price
   * ‘Is Vegan?’ status
   * ‘Is Vegetarian?’ status
   * ‘Contains Allergen?’ status
5. The system updates the stock levels for the selected additional ingredients

**Identifier and name** UC13 *Remove ingredient from food item*

**Initiator** *Waiter*

**Goal** An ingredient is removed from the selected food

item.

**Precondition** The selected food item has been added to the order.

The selected additional ingredient is listed against the selected food item

**Postcondition** The additional ingredient will have been removed from the

selected food item.

**Assumptions** The expected initiator is a waiter. Not all ingredients can be

removed from the food item. Ingredients that cannot be

removed will not be listed for the user.

**Main success scenario**

1. The waiter selects a food item listed on the selected order.
2. The system displays the list of removable additional ingredients stored against the selected food item.
3. The waiter chooses one or more ingredients to be removed and saves the selection.
4. The system removes the ingredients to the selected food item and updates the item’s:
   * Price
   * ‘Is Vegan?’ status
   * ‘Is Vegetarian?’ status
   * ‘Contains Allergen?’ status
5. The system updates the stock levels for the selected additional ingredients

**Identifier and name** UC14 List active orders

**Initiator** *Waiter*

**Goal** The list of all active orders owned by the waiter is displayed.

**Precondition** The waiter has an opened session.

**Postcondition** The list of all of the waiter’s active orders will have been

displayed.

**Assumptions** The expected initiator is a waiter.

**Main success scenario**

1. The waiter requests to see her or his full list of active orders.
2. The system displays the list full list of all active orders owned by the waiter.