1. Add a new product

Use Case: Add a new product  
Triger/Goal: To enter details of a new product  
Actors: Operator

Main Flow:

1. Operator enters a product id
2. System validates that the id is unique
3. Operator enters a short description
4. System saves the new product’s details

Extensions:

2a. Id is not new  
 1. System notifies Operator and cancels use case

1. Remove a product – provided there are no standing orders for this product

Use Case: Remove a product – provided there are no standing orders  
Trigger/Goal: To remove a product provided there are no standing orders for this product  
Actors: Operator

Main Flow:

1. Operator enters id of product to be deleted
2. System checks if product id exists
3. System checks if any standing orders involve the product
4. System removes the product

Extensions:

2a. Id does not exist  
 1. System notifies Operator and cancels use case  
 3a. product is involved in a standing order  
 1. System notifies operator and cancels use case

1. Add a new customer

Use case: Add a new customer  
Trigger/Goal: to add a new customer with at least one delivery address and one standing order  
Actors: Operator

Main Flow:

1. Operator enters customer id
2. System checks if customer id is unique
3. Operator enters name
4. Operator enters address
5. System checks if address is unique
6. Operator enters name of contact person
7. Operator enters phone number of contact person
8. System assigns unique address id
9. System saves address and contact person
   1. Repeat steps 4 – 10
10. Operator enters product id
11. System checks product id exists
12. Operator enters price
13. Operator enters quantity required each day
14. Operator enters start date
15. Operator enters end date
16. System checks end date is after start date
17. System checks start date and sets the status of the standing order
18. System saves standing order
    1. Repeat steps 11 – 19
19. System saves Customer

Extension:

2a. customer id is not unique

1. Appropriate message is displayed, and previous step is recalled

5a. address is not unique

1. Appropriate message is displayed, and previous step is recalled

12a. product id does not exist

1. Appropriate message is displayed, and previous step is recalled

17a. end date is less the start date

1. Appropriate message is displayed, and previous step is recalled
2. Add a delivery address to an existing customer

Use Case: Add a delivery address to an existing customer  
Trigger/Goal: To add a new delivery address to an existing customer  
Actors: Operator

Main Flow:

1. Operator enters customer id
2. System checks if customer id exists
3. Operator enters address
4. System checks if address is unique
5. Operator enters name of contact person
6. Operator enters phone number of contact person
7. System saves the new delivery address assigning it a unique id

Extensions:

2a. Customer id doesn’t exist  
 1. System notifies Operator and cancels use case  
 4a. Address is not unique  
 1. System notifies Operator and cancels use case

1. Remove a delivery address provided no deliveries have been made to this address

Use Case: Remove a delivery address  
Triggers/Goal: to remove a delivery address provided no deliveries have been made to that address  
Actors: Operator

Main Flow:

1. Operator enters a customer id
2. System checks if id exists
3. Operator enters an address
4. System checks if address exists
5. System checks if address has had any deliveries
6. System removes address

Extensions:

2a. customer id doesn’t exist  
 1. System notifies Operator and cancels use case  
 4a. address doesn’t exist  
 1. System notifies Operator and cancels use case  
 5a. deliveries have been made to the address  
 1. System notifies Operator and cancels use case

1. Add a standing order to an existing customer

Use Case: Add a standing order to an existing customer  
Triggers/Goal: To add a standing order to an existing customer  
Actors: Operator

Main Flow:

1. Operator enters customer id
2. System validates customer id
3. Operator enters delivery address
4. System validates delivery address
5. Operator enters product id
6. System validates product id
7. Operator enters agreed price
8. Operator enters quantity required for:  
   - Sunday  
   - Monday  
   - Tuesday  
   - Wednesday  
   - Thursday  
   - Friday  
   - Saturday
9. Operator enters start date
10. System checks start date is not before current date
11. Operator enters termination date
12. System checks termination date is after start date
13. System creates new standing order with unique id
14. System checks start date and sets the status to “active”
15. System saves standing order

Extension:

2a. customer id doesn’t exist  
 1. System notifies Operator and cancels use case  
 4a. delivery address doesn’t exist  
 1. System notifies Operator and cancels use case  
 6a. product id doesn’t exist  
 1. System notifies Operator and cancels use case  
 10a. start date is less than current date  
 1. System notifies Operator and cancels use case  
 12a. termination date is before start date  
 1. System notifies Operator and cancels use case  
 14a. start date is later then current date  
 1. System sets status to “closed”

1. Close a standing order

Use Case: Close a standing order  
Triggers/Goal: to close a standing order  
Actors: Operator

Main Flow:

1. Operator enters customer ID
2. System checks customer ID
3. Operator enters standing order id
4. System checks if standing order id exists
5. System deletes standing order

Extension:

2a. customer id doesn’t exist  
 1. Appropriate message is displayed, and previous step is recalled

4a. standing order id doesn’t exist

1. Appropriate message is displayed, and previous step is recalled
2. List standing orders that need to be filled for a date

Use Case: List standing orders that need to be filled for a date  
Triggers/Goals: list all standing orders that need to be filled on a date ordered by customer name and within that order id  
Actors: Operator

Main Flow:

1. Operator enters date
2. System retrieves all standing orders for that date
3. System sorts collection by customer name
4. System sorts standing orders within each customer name by order id

Extension:

1. Add a delivery

Use Case: Add a delivery  
Triggers/Goals: add a delivery to the system  
Actors: Operator

Main Flow:

1. Operator enters delivery day
2. Operator enters delivery address
3. System validates address exists
4. Operator enters delivery number
5. System retrieves all standing orders for day
6. System saves the delivery

Extension:

3a. delivery address doesn’t exist  
 1. Appropriate message is displayed, and previous step is recalled

1. Generate a delivery docket

Use Case: Generate a delivery docket  
Triggers/Goal: generate a delivery docket  
Actors: Operator

Main Flow:

1. Operator enters delivery date and delivery number
2. System retrieves delivery
3. System prompts Operator asking if all orders were filled
4. Operator enters “yes”
5. System generates delivery docket with delivery date, day of the week, standing order id and quantity

Extension:

2a. delivery does not exist  
 1. Appropriate message is displayed, and previous step is recalled

4a. operator enters “no”  
 1. Operator enters standing order id  
 2. System adds a \* to the standing order