OC Pizza

Order management application

Functional design

Spencer Forrest

TABLE OF CONTENTS

1 - VERSIONS

Author	Date	Description	Version
Spencer Forrest	19/06/2019	Document creation	1.0

2 - INTRODUCTION

2.1 - Document purpose

The present document makes up the functional design file for the order management application. This document will present the needs of the client and describe the solution which answers those needs.

2.2 - Client needs

2.2.1 - Context

"OC Pizza" is a young and expanding group that specializes in pizza delivery and take-out. It already has five stores and plans to open at least three more by the end of the year. They desire to set up an information technology system to be launched in all of the pizzerias.

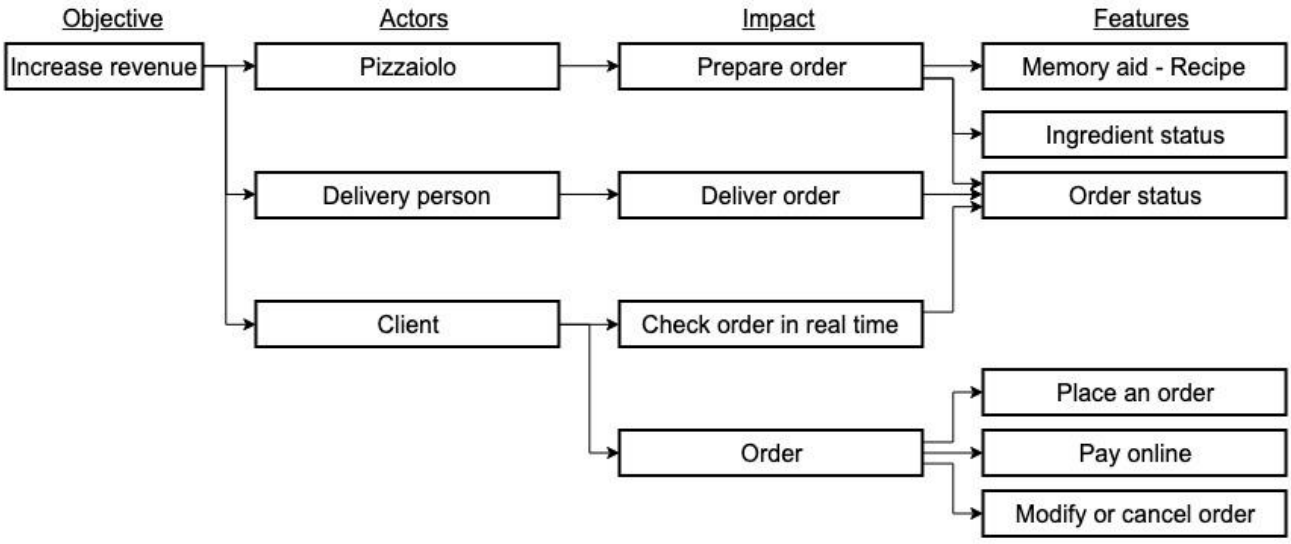
2.2.2 - Stakes and Objectives

“OC Pizza” stakes and objectives are:

- To increase revenue
- To become more efficient in the handling of the orders

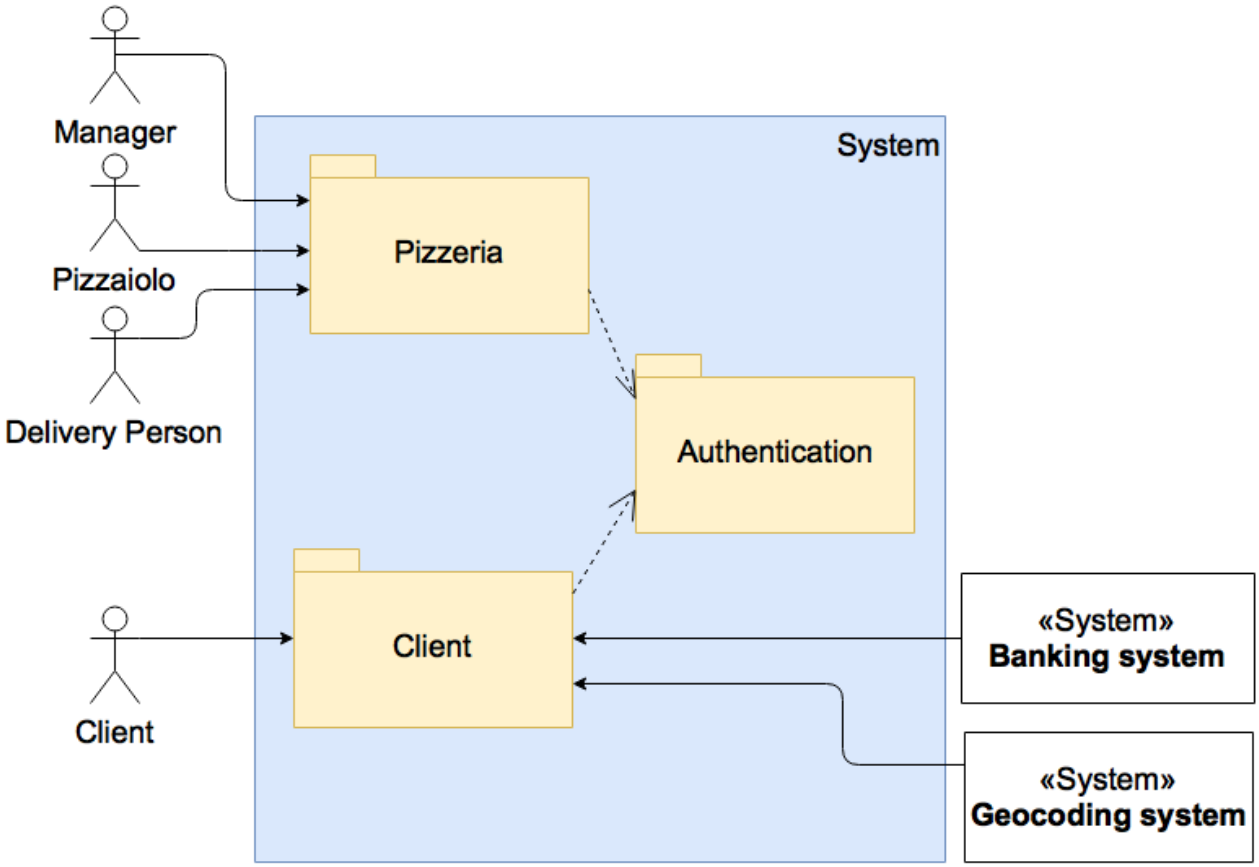
3 - GENERAL DESCRIPTION OF SOLUTION

3.1 - Impact Mapping



4 - FUNCTIONAL AREA

4.1 - Package Diagram



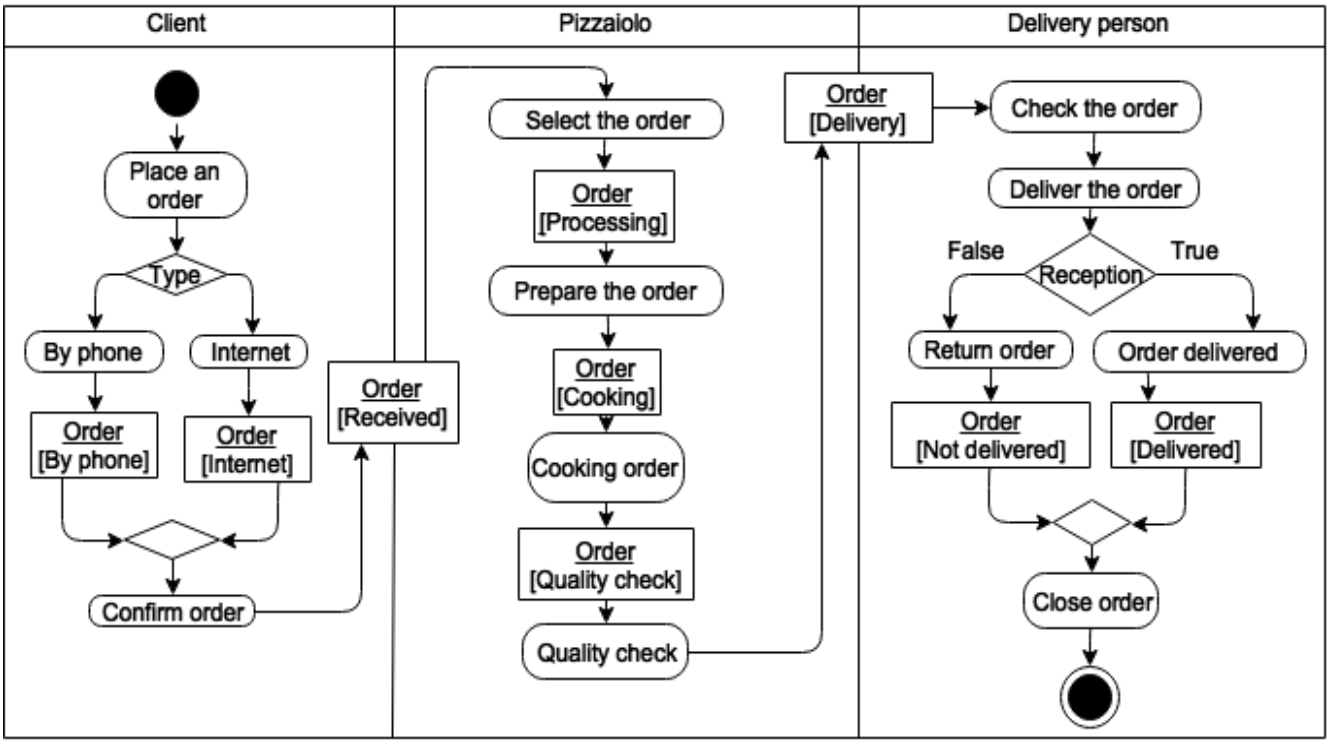
The system will be divided into 3 packages:

- Authentication package.
- Client package where the client, geocoding and banking system will interact with the system
- Pizzeria where the employees and the manager will interact with the system.

5 - WORKFLOW

5.1 - State machine diagram

This diagram presents the workflow of an order.



5.2 - Order life cycle

5.2.1 - Client

Action	Order status
Order online	Internet
Order by phone	By phone
Order confirmed successfully	Received

5.2.2 - Pizzaiolo

Action	Order status
Prepare the pizza	Processing
Cook the pizza	Cooking
Check quality and package up the pizza	Quality check

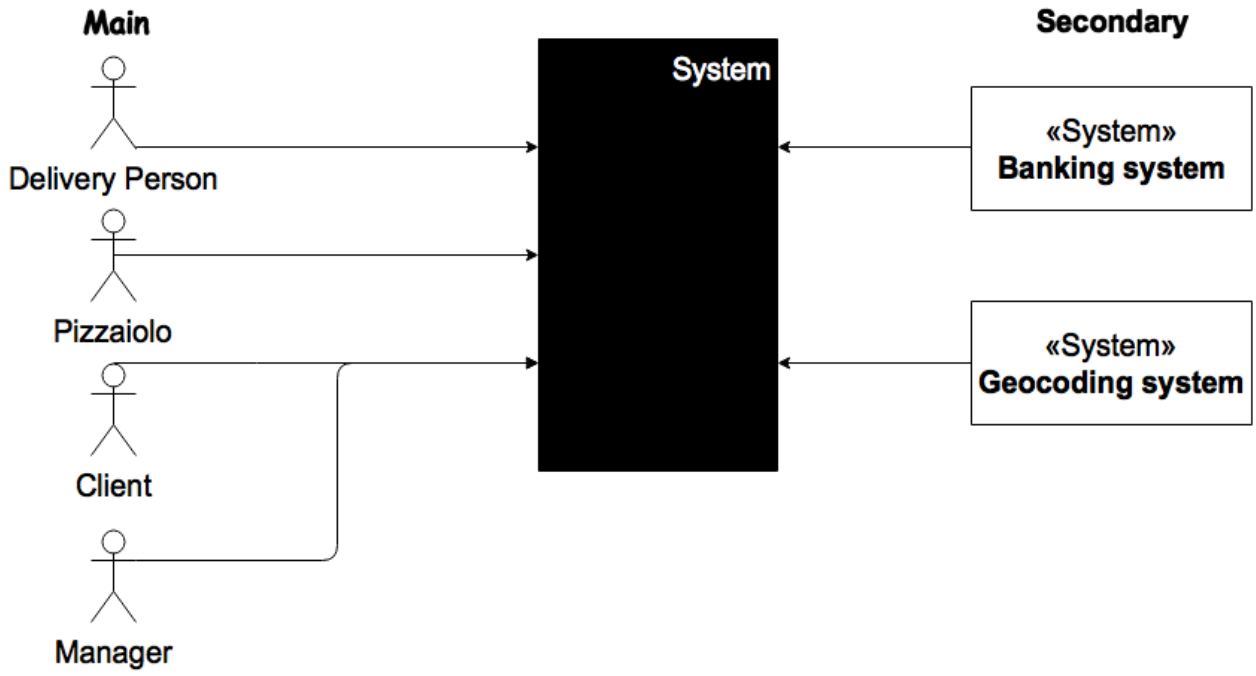
5.2.3 - Delivery person

Action	Order status
Check the order and the delivery address	Delivery
Order successfully delivered	Delivered
Order unsuccessfully delivered	Not delivered

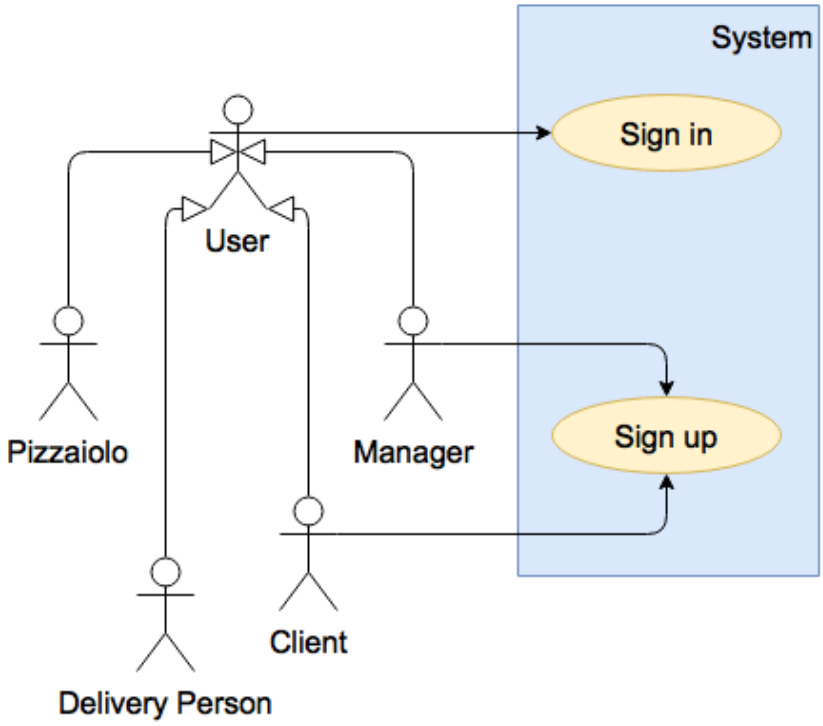
6 - WEB APPLICATION

The web application will manage the order taken by the client and keep track of its status until it is delivered to the client. It will also facilitate the payment by proposing an online solution.

6.1 - Actors



6.2 - Use cases



6.2.1 - Authentication

Employees must sign in:

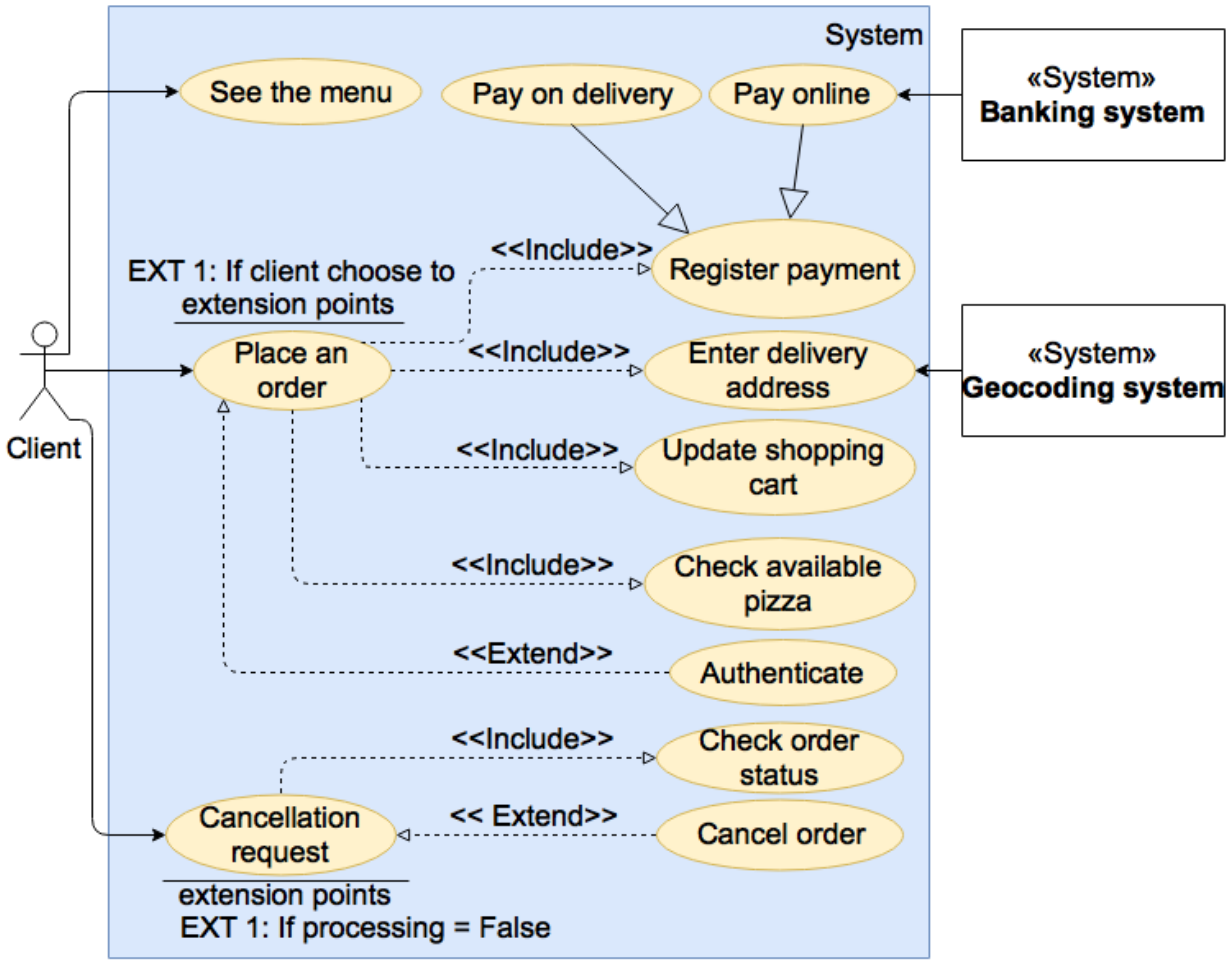
- Only the manager can sign up for a new employee.

Client may sign in:

- New client can sign up.

Actors	Authentication
Manager	Mandatory
Pizzaiolo	Mandatory
Delivery person	Mandatory
Client	Optional

6.2.2 - Client diagram



6.2.3 - Client descriptions

Case n°10

Name: Place an order (« Client » package)

Actor(s): Client
Description: A client should be able to order any available pizza
Author: Spencer Forrest
Date(s): 11/08/2017

Preconditions: none
Start: *The user* asked for the page “Place an order”

DESCRIPTION

Main scenario:
1. *The user* authenticates if he desires to with the use case “Authenticate”
2. *The user* enters his city and his zip code
3. **The system** calls the “Check available pizza” use case
4. **The system** calls the “Update shopping cart” use case
5. **The system** calls the “Enter delivery address” use case
6. **The system** calls the “Register payment” use case
7. **The system** displays the receipt and the actual order status

Alternative scenarios:
1.a *The user* decides to cancel his order
2.a *The user* decides to cancel his order
2.b **The system** stops the client’s order if he is outside of the delivery zone
3.a *The user* decides to cancel his order
4.a *The user* decides to cancel his order
5.a *The user* decides to cancel his order
6.a *The user* decides to cancel his order
6.b The payment did not go through
7.a *The user* decides to close the receipt and the order status page

End:
Main scenario:
On step 1, 2.a, 3, 4, 5, 6 or 7, if *the user* chooses to

Special scenario:

On step 2.b, due to **the system**

Postconditions:
See the “Register payment” use case

COMPLEMENTS

Ergonomics:
A navigation bar highlights the actual use case in use

Case n°11

Name: Authenticate (« Client » package)

Actor(s): Client
Description: A client should be able to authenticate if he desires to in order to facilitate the input of his address information
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: none
Start: *The user* asked the “Authenticate page”

DESCRIPTION

Main scenario:
1. **The system** calls the package “Authentication”

Case n°12

Name: Check available pizza (« Client » package)

Actor(s): Client
Description: The client checks which pizzas are still available to order
Author: Spencer Forrest
Date(s): 11/08/2017

Preconditions: The client can authenticate (« Authenticate » use case)
Start: *The user* asked for the « Check the Pizza » page

DESCRIPTION

Main scenario:
1. **The system** displays the list of pizzas that can still be ordered
2. *The user* chooses a pizza
3. **The system** displays a description with options for the dough and size
4. *The user* adds a pizza in his shopping cart with his chosen options
5. *The user* stops checking the pizza
6. **The system** display the list again (step 1)

Alternative scenarios:
2.a *The user* stops checking the available pizzas
2.b *The user* cancels the order
4.a *The user* cancels the order
5.a *The user* cancels the order

End:
Main scenario: steps 2, 4 or 5, if user chooses to

Postconditions:
See “Update shopping cart” use case

COMPLEMENTS

Ergonomics:
Display pizza in a group of 15

Expected performance:
The list of pizzas should be displayed in less than 5 seconds

Unsolved problem(s):
Should we have different groups of pizza?

Case n°13

Name: Update shopping cart (« Client » package)

Actor(s): Client
Description: The client can add, update, or delete items from the cart.
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be authenticated (« Authentication » use case).
Start: User requested for the « See my cart » page.

DESCRIPTION

Main scenario:
1. **The system** displays a page with items in the cart
2. *The user* can remove or add an item from the cart.
3. *The user* can modify the quantity of an item
4. *The user* can modify the options for an item

Alternative scenarios:
2.a *The user* decides to go back to the previous page.
2.b *The user* decides to cancel the order.
3.a *The user* decides to go back to the previous page.
3.b *The user* decides to cancel the order.
4.a *The user* decides to go back to the previous page.
4.b *The user* decides to cancel the order.

End:
Main scenario: On steps 2, 3 or 4, if user chooses to.

Postconditions:
Cart saved for the session only.

COMPLEMENTS

Ergonomics:
All the items in the cart should be displayed on one page.

Expected performance:
Displaying the page should take less than 5 seconds.

Case n°14

Name: Enter delivery address (« Client » package)

Actor(s): Client
Description: A client has to enter required information for delivery
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate ») so required pieces of information are already filled in
Start: *The user* asked the “Delivery address” page

DESCRIPTION

Main scenario:
1. **The system** displays an empty form if *the user* did not sign in
2. *The user* fills out the form
3. *The user* confirms that the pieces of information are correct
4. **The system** calls an external **geocoding system** to check the address
5. **The system** highlights in red the wrong pieces of information or the fields needed to be filled in
6. *The user* makes any correction needed and confirm

Alternative scenarios:
2.a *The user* decides to go back to the previous page
2.b *The user* decides to cancel his order
3.a *The user* decides to go back to the previous page
3.b *The user* decides to cancel his order
6.a *The user* decides to go back to the previous page
6.b *The user* decides to cancel his order

End:
Main scenario: On steps 2, 3 or 6, if user chooses to.

Postconditions:
Information saved for the session only

COMPLEMENTS

Ergonomics:
The form should fit on one page

Expected performance
The page should be displayed in less than 5 seconds
The external geocoding system should respond in less than 10 seconds

Unsolved problem(s):
Should we use an external geocoding system to check the validity of the address?

Case n°15

Name: Register payment (« Client » package)

Actor(s): Client
Description: The client should be able to register which type of payment he will use
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate »)
Start: *The user* asked for the « Payment » page

DESCRIPTION

Main scenario:
1. **The system** displays a page with the different payment options available
2. *The user* selects one of the options
3. **The system** calls “Pay on delivery” according to the user choice
4. **The system** calls “Pay online” according to the user choice

Alternative scenarios:
2.a *The user* decides to go back to the previous page
2.b *The user* decides to cancel the order

End:
Main scenario: On steps 2, if user chooses to

Postconditions:
See “Pay on delivery” and “Pay online” use case

COMPLEMENTS

Ergonomics:
Displays both payment options in the center as 2 buttons

Expected performance:
The pizzas should be displayed in less than 5 seconds

Case n°16

Name: Pay on delivery (« Client » package)

Actor(s): Client
Description: The client should be able to pay on the delivery
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate »)
Start: *The user* requested the “Pay on delivery” option

DESCRIPTION

Main scenario:
1. **The system** displays:
   - Sum up of the order
   - The client adress
   - The amount due upon delivery

2. *The user* confirms

Alternative scenarios:
2.a *The user* decides to cancel the order
2.b *The user* decides to change his payment option

End:
Main scenario: On steps 2, if user chooses to

Postconditions:
The order is saved in the database
Summary of the order sent to the user by email

COMPLEMENTS

Expected performance:
The page should be displayed in less than 5 seconds

Case n°17

Name: Pay online (« Client » package)

Actor(s): Client
Description: The client must be able to pay online
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate »)
Start: *The user* requested the "Pay online" option

DESCRIPTION

Main scenario:
1. **The system** displays a form
2. *The user* fills out the form
3. **The system** calls an external **banking system**
4. **The system** displays a confirmation of payment

Alternative scenarios:
2.a *The user* decides to cancel the order
2.b *The user* decides to change his payment method
4.a **The system** could not confirm the payment

End:
Main scenario:
On step 2, if user chooses to

Special scenario:
On step 4 if **banking system** transaction fails

Postconditions:
Main scenario:
Order saved into the database
Receipt sent to the client by email

COMPLEMENTS

Expected performance:
The form should be displayed in less than 5 seconds
The banking system should respond in less than 10 seconds

Case n°18

Name: See the menu (package « Client »)

Actor(s): Client
Description: Client must be able to see the menu
Author: Spencer Forrest
Date(s): 14/12/2017

Preconditions: None
Start: *The user* asked for the « See the menu » page

DESCRIPTION

Main scenario:
1. **The system** displays a page with a list of pizzas offered by the pizzeria
2. *The user* chooses a pizza
3. **The system** displays a description of the pizza with options for the dough and size
4. *The user* leaves the pizza description page
5. **The system** displays a list of pizzas (step 1)

Alternative scenarios:
2.a *The user* leaves the menu page
4.a *The user* leaves the menu page

End:
Main scenario: On steps 2 or 4, if user chooses to

Postconditions: None

COMPLEMENTS

Ergonomics:
Display pizza in a group of 15

Expected performance:
The pizzas should be displayed in less than 5 seconds

Unsolved problem(s):
Should we have different groups of pizza?

OC Pizza

Case n°19

Name: Cancellation request (« Client » package)

Actor(s): Client
Description: The client should be able to cancel his order
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate »).
Otherwise, he uses his phone number to identify the order
Start: *The user* asked for the « Check order » page

DESCRIPTION

Main scenario:

1. **The system** calls “Check order status” use case
2.**The system** calls “Cancel order” if the status is “Received”
3. **The system** calls back “Check order status” use case (go back to step 1)

COMPLEMENTS

Ergonomics:

The order status should be displayed on a single page

Expected performance:

The pages should be displayed in less than 5 seconds
The order status is updated in real time

Case n°20

Name: Check order status (« Client » package)

Actor(s): Client
Description: The client should be able to check the order status in real time
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate »).
Otherwise he uses his phone number to track his order.
Start: *The user* requested the « Order status » page

DESCRIPTION

Main scenario:

1. **The system** displays a bar which indicates the order status in real time
2. **The system** displays a button “Cancel order” if the status is “Received”
3. *The user* can cancel the order if the status is “Received”

Alternative scenarios:

3.a *The user* decides to quit tracking his order

End:

Main scenario: On step 3, if *the user* chooses to

Postconditions:

The order status is saved in real time in the database.
The user receives an email when the order is ready to be delivered.

COMPLEMENTS

Ergonomics:

The page should display a progress bar to follow the order status.

Expected performance:

Display the page in less than 10 seconds.
Update of the progress bar in real time.

Case n°21

Name: Cancel order (« Client » package)

Actor(s): Client
Description: The client can cancel the order
Author: Spencer Forrest
Date(s): 19/06/2017

Preconditions: The client might be signed in (Use case « Authenticate »). Otherwise, he uses his phone number to identify the order. The status of the order should be “Received”.

Start: *The user* asked for the « Cancel order » page

DESCRIPTION

Main scenario:
1. **The system** displays a page with the sum up of the order to cancel.
2. *The user* presses the button “Confirm.”
3. **The system** displays the main page of the web site.

Alternative scenarios:
2.a *The user* doesn’t want to cancel the order anymore.

End:
Main scenario: On step 2, if *the user* chooses to

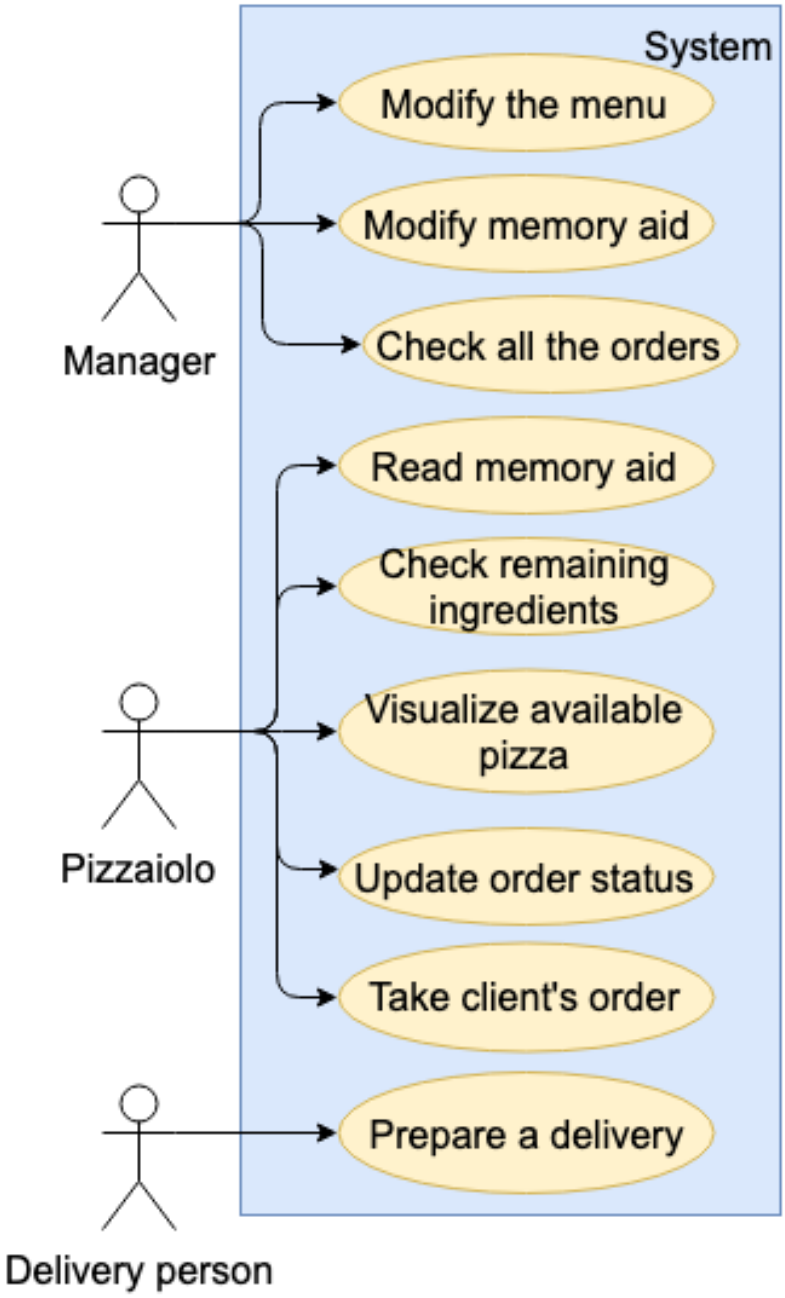
Postconditions:
Update order status in the database.
Send cancellation email to the client.

COMPLEMENTS

Ergonomics:
Sum up of the order should only be displayed on one page.

Expected performance:
The page should be displayed in less than 5 seconds.

6.2.4 - Pizzeria diagram



6.2.5 - Pizzeria descriptions

Case n°1

Name: Read memory aid (« Pizzeria » package)

Actor(s): Pizzaiolo
Description: Pizzaiolo should be able to read the memory aid
Author: Spencer Forrest
Date(s): 09/09/2017

Preconditions: *The user* must be signed in as pizzaiolo
(“Sign in” use case – “Authentication” package)
Start: *The user* requested the “Read memory aid" page

DESCRIPTION

Main scenario:

1. **The system** displays a page with the list of pizzas.
2. *The user* selects a pizza.
3. **The system** displays the ingredients and the instructions to prepare the pizza.
4. *The user* can leave the instructions page.
5. **The system** goes back to the list of pizzas (back to step 1).

Alternative scenarios:

2.a *The user* decides to quit reading the memory aid
4.a *The user* decides to quit reading the memory aid

End:

Main scenario: On steps 2 or 4, if user chooses to.

COMPLEMENTS

Ergonomics:

Display pizza in a group of 15

Expected performance:

The pizza list and the instructions must be displayed in less than 10 seconds

Unsolved problem(s):

If the pizzeria proposes more than 30 different pizzas, should we group them by category?

Case n°2

Name: Check remaining ingredients (« pizzeria » package)

Actor(s): Pizzaiolo
Description: Pizzaiolo should be able to check the quantity of ingredient left
Author: Spencer Forrest
Date(s): 11/08/2017

Preconditions: *The user* must be signed in as pizzaiolo
(“Sign in” use case – “Authentication” package)
Start: *The user* requested the “Remaining ingredients" page

DESCRIPTION

Main scenario:

1. **The system** displays a page with the quantity of each ingredient still available
2. *The user* goes back to the main menu

End:

Main scenario: On steps 2, if user chooses to

COMPLEMENTS

Ergonomics:

Display the ingredients by group of 15. However, *the user* may be able to ask to show 30 or 60 ingredients per page

Expected performance:

The page should be displayed in less than 10 seconds

Unsolved problem(s):

Should we group the ingredients by sections?

Case n°3

Name: Visualize available pizza («Pizzeria» package)

Actor(s): Pizzaiolo
Description: Pizzaiolo can check which pizza he is still able to make with the ingredients left.
Author: Spencer Forrest
Date(s): 11/08/2017

Preconditions: *The user* must be signed in as pizzaiolo
("Sign in" use case – "Authentication" package)
Start: *The user* requested the "Available pizza" page

DESCRIPTION

Main scenario:

1. **The system** displays a list of pizza which can still be prepared
2. *The user* goes back to the main menu

End:

Main scenario: On step 2, if user chooses to

COMPLEMENTS

Ergonomics:

Display the pizza by group of 15

Expected performance:

Displaying the list of pizza should take less than 5 seconds

Case n°4

Name: Update order status («Pizzeria » package)

Actor(s): Pizzaiolo
Description: The pizzaiolo should be able to update the order status
Author: Spencer Forrest
Date(s): 11/09/2017

Preconditions: *The user* must be signed in as pizzaiolo
(“Sign in” use case – “Authentication” package)
Start: *The user* requested the “Update order status" page

DESCRIPTION

Main scenario:

1.**The system** displays a list of incomplete orders on a page
2. *The user* selects an order
3.**The system** displays the order details with its status
4. *The user* selects the “Processing” status if at least one item is being prepared
5. *The user* selects the “Cooking” status if at least one item is being cooked
6. *The user* selects the “Quality check” status if at least one item is being checked
7. *The user* goes back to the list of orders (Go back to step 1)

Alternative scenarios:

2.a *The user* decides to quit updating the order status
4.a *The user* decides to quit updating the order status
4.b *The user* decides to go back to the order list
5.a *The user* decides to quit updating the order status
5.b *The user* decides to go back to the order list
6.a *The user* decides to quit updating the order status
7.a *The user* decides to quit updating the order status

End:

Main scenario: On steps 2, 4, 5, 6 or 7, if user chooses to

Postconditions:

Order status saved in database with date and time

COMPLEMENTS

Ergonomics:

Display the uncompleted orders with their actual status
Display all the order items on one page

Expected performance:

The page should be displayed in less than 10 seconds
Displaying and modifying the status should take less than 5 seconds

Unsolved problem(s):

What happen if the order does not pass the quality check?

Case n°5

Name: Prepare a delivery (« Pizzeria » package)

Actor(s): Delivery person
Description: Delivery person should be able to prepare a delivery
Author: Spencer Forrest
Date(s): 11/09/2017

Preconditions: *The user* must be signed in as delivery person
(“Sign in” use case – “Authentication” package)
Start: *The user* requested the “Prepare a delivery" page

DESCRIPTION

Main scenario:

1. **The system** displays a list of orders ready for delivery
2. *The user* selects one of the orders
3. **The system** displays the client information (Name, address and phone)
4. **The system** displays the order detail (items)
5. *The user* can select the status “Delivery”
6. *The user* can select the status “Delivered” or “Not delivered”
8. **The system** goes back to the list of orders (back to step 1)

Alternative scenarios:

2.a *The user* decides to leave the delivery system
5.a *The user* decides to leave the delivery system
5.b *The user* decides to select another order
6.a *The user* decides to leave the delivery system
6.b *The user* decides to select another order

End:

Main scenario: steps 2, 5 or 6, if user chooses to

Postconditions:

Order status saved in database in real time

COMPLEMENTS

Ergonomics:

Display the order list on one page with their actual status
Display all the items of an order on one page

Expected performance:

Displaying the order list should take less than 10 seconds
Displaying and saving the order status should take less than 5 seconds

Case n°6

Name: Modify the menu (« Pizzeria » package)

Actor(s): Manager
Description: The manager should be able to update the menu
Author: Spencer Forrest
Date(s): 11/06/2017

Preconditions: *The user* must be signed in as manager
(“Sign in” use case – “Authentication” package)
Start: *The user* requested the “Modify the menu" page

DESCRIPTION

Main scenario:

1. **The system** displays a list of pizza on a page
2. *The user* selects a pizza
3. **The system** displays a description and picture of the pizza
4. *The user* modify the description or the picture
5. *The user* confirms the modifications
8. **The system** goes back to the list of pizza (back to step 1)

Alternative scenarios:

2.a *The user* decides to leave the modification menu
4.a *The user* decides to leave the modification menu
4.b *The user* decides to stop modifying the pizza
5.a *The user* decides to leave the modification menu
5.b *The user* decides to stop modifying the pizza

End:

Main scenario: On steps 2, 4 or 5, if user chooses to

Postconditions:

Modifications saved in the database

COMPLEMENTS

Ergonomics:

Display the pizza by group of 10

Expected performance:

The page should be displayed in less than 10 seconds.
Displaying and modifying the instructions and picture should take less than 5 seconds.

Unsolved problem(s):

Does the pizzeria only sell pizza?

Case n°7

Name: Modify memory aid (« Pizzeria » package)

Actor(s): Manager
Description: The manager should be able to update or delete the memory aid
Author: Spencer Forrest
Date(s): 11/06/2017

Preconditions: *The user* must be signed in as manager
("Sign in" use case – "Authentication" package)
Start: *The user* requested the "Modify memory aid" page

DESCRIPTION

Main scenario:

1. **The system** displays a list of pizza on a page
2. *The user* selects a pizza
3. **The system** displays the ingredients needed and the instructions
4. *The user* modifies the ingredients and instructions to prepare the pizza
5. *The user* confirms the modification
8. **The system** goes back to the list of pizza (Go back to step 1)

Alternatives scenario:

2.a *The user* decides to leave the memory aid
4.a *The user* decides to leave modifying the memory aid
4.b *The user* decides to stop modifying the pizza
5.a *The user* decides to leave modifying the memory aid
5.b *The user* decides to stop modifying the pizza

End:

Main scenario: On steps 2, 4 or 5, if user chooses to

Postconditions:

Modifications saved in the database

COMPLEMENTS

Ergonomics:

Display the ingredients by group of 10

Expected performance:

The page should be displayed in less than 10 seconds
Displaying and modifying the instructions and ingredients should take less than 5 seconds

Case n°8

Name: Check all the orders (package « Partie pizzeria »)

Actor(s): Manager
Description: Manager should be able to see all the orders, present and past
Author: Spencer Forrest
Date(s): 11/09/2017

Preconditions: *The user* must be signed in as manager
("Sign in" use case – "Authentication" package)
Start: *The user* requested the "See all orders" page

DESCRIPTION

Main scenario:

1. **The system** displays a list of orders with their status
2. *The user* selects an order
3 **The system** displays the order information (Items, prices, timestamp, status)
4. *The user* check the order information
5. **The system** goes back to the list of orders (Go back to step 1)

Alternatives scenario:

2.a *The user* chooses to quit checking orders
4.a *The user* chooses to quit checking orders

End:

Main scenario: On steps 2, 4 or 5, if user chooses to

Postconditions: None

COMPLEMENTS

Ergonomics:

Display the orders by group of 10
The user should be able to sort the orders by price, time and/or status

Expected performance:

Displaying the order list should take less than 5 seconds

Case n°22

Name: Take client's order (« Pizzeria » package)

Actor(s): Pizzaiolo
Description: Pizzaiolo should be able to take a client's order
Author: Spencer Forrest
Date(s): 11/09/2017

Preconditions: *The user* must be signed in as pizzaiolo
("Sign in" use case – "Authentication" package)
Start: *The user* requested the "Client order" page

DESCRIPTION

Main scenario:

1. **The system** calls the "Place an order" use case ("Client" package)
2. **The system** set the status order to "By phone"
2. **The system** goes back to the main menu

Alternative scenarios:

1.a *The user* decides to cancel his order

End:

Main scenario: On step 1, if user chooses to

Postconditions:

Order and its status saved in the database with a timestamp
Send an email of the order details to the client

COMPLEMENTS

Ergonomics / Expected performance:

See the “Place an order” use case (“Client” package)

6.3 - General management rules

6.3.1 - Client

Use case	Rules
“Enter delivery address”	Form automatically filled out if client signed in
	Geocoding system checks if the address exists
“Check available pizza”	Available pizza depends on the ingredients left
“Register payment”	Only available if shopping cart is not empty
	If registering payment is a success, the order is a success
“Cancellation request”	Only if the order status is “Received”

6.3.2 - Pizzeria

Use case	Rules
“Check remaining ingredients”	Automatically updated for each pizza prepared
	Can be updated manually to add or remove ingredients from the stock
“Visualize available pizza”	Pizza still available depends on remaining ingredients
“Modify memory aid”	Update the type and quantity of ingredient needed for a pizza
	Linked with “Check remaining ingredients” et “Visualize available pizza”
“Update order status”	Update the order status: “Received”, “Processing”, “Cooking” or “Quality check”
“Prepare a delivery”	Update the order status: “Delivery”, “Delivered” or “Not delivered”