Acknowledgment

I would like to express my deepest appreciation to all those who aide me to complete this report. A special gratitude is giving to my final year project manager, Mr. BAYREM TRIKI, whose contribution in stimulating suggestions and encouragement, helped me to coordinate this project especially in writing this report. And a special thanks goes to my supervisor Mr. ANIS BEN BETTAIEB, who guided me throughout the development of **"Cake it up !"** application.

Furthermore I would also like to acknowledge with much appreciation the crucial role of the school EPI, who gave the permission to use all required equipment and the necessary materials to complete this project. Finally, we have to appreciate the guidance will be given by other supervisor, especially in my project exposition that will surely improve my presentation skills thanks to their comment and advices.

Table of Contents

List of Figures

List of Tables

General introduction

Bakeries are a popular type of foodservice establishment, and they allow you to express your culinary creativity while also serving a unique market, the problem is you can spend all day and night in the kitchen creating the next best cake, but if no one knows about it, it doesn't matter.

In order to succeed in the bakery business, some marketing and advertising can make a big difference.

Depending on all of that, our application named **"Cake it up !"** have to fill in the needs of both parts the business owner and the client. It needs an easy way for the owner to promote their business and easier way for the client to find a place that matches him.

This report contains three chapters; the first chapter will be taking over the project specification, presenting the problematics and their solutions. The second chapter will be concerned about conception, different diagrams; class and sequence. Then we will proceed in the realization of the application as a third chapter which in it we will talk about the used technologies, architecture and the implementation. Last but not least, the general conclusion which will give a general and brief idea on what we did throughout this report.

Chapter 1

Preliminary Study

Introduction

This chapter is dedicated to list the different obstacles that a business owner and a client face, which will guide us to give details on the proposed solution followed by the specification of the project. We will also introduce both the project life cycle and the used methodology. We will also give a detailed description of the use case diagram and the system sequence diagram. Finally we will end this section by a conclusion.

1.1 Analysis of the existing

We are in an era where technology leads everything, there is an application for most of the people's needs, however most of these softwares lack some details that are not yet fulfilled.

This section will start by listing some of the applications or ways people use to get information about pastries, then criticize every one of them to show what is missing.

1.1.1 Social media

The mostly common way is using the social media such as **"FACEBOOK"** and **"INSTAGRAM"** where you can find a verity of choices and number of opinions that can guide the user to select the right place.

We can also use it to figure out the cost of the meal or get direction as shown in Figure 1.1.

The social media also give you the ability to communicate with the manager of the place in private to either pass an order or to ask for more details regarding a specific product.

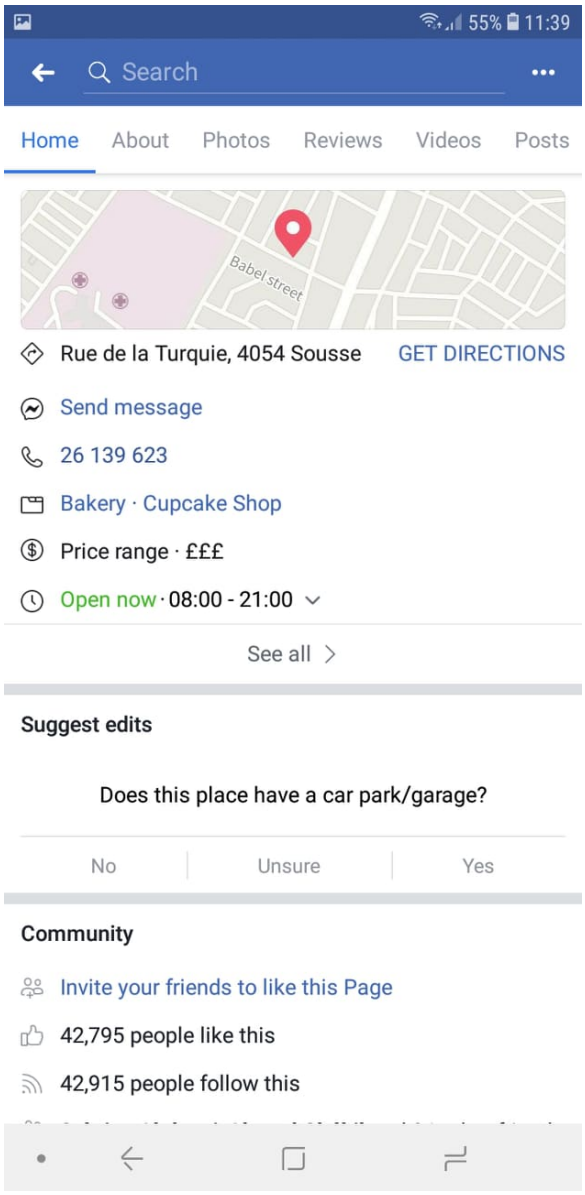

Figure 1.1: Application: FACEBOOK[1]

[1]Source: https://www.facebook.com/

1.1.2 Google maps

"Google maps" has been popular for quite a while now, it can be used to figure out the nearby pastries, reviewing shop and get directions as showing in Figure 1.2.

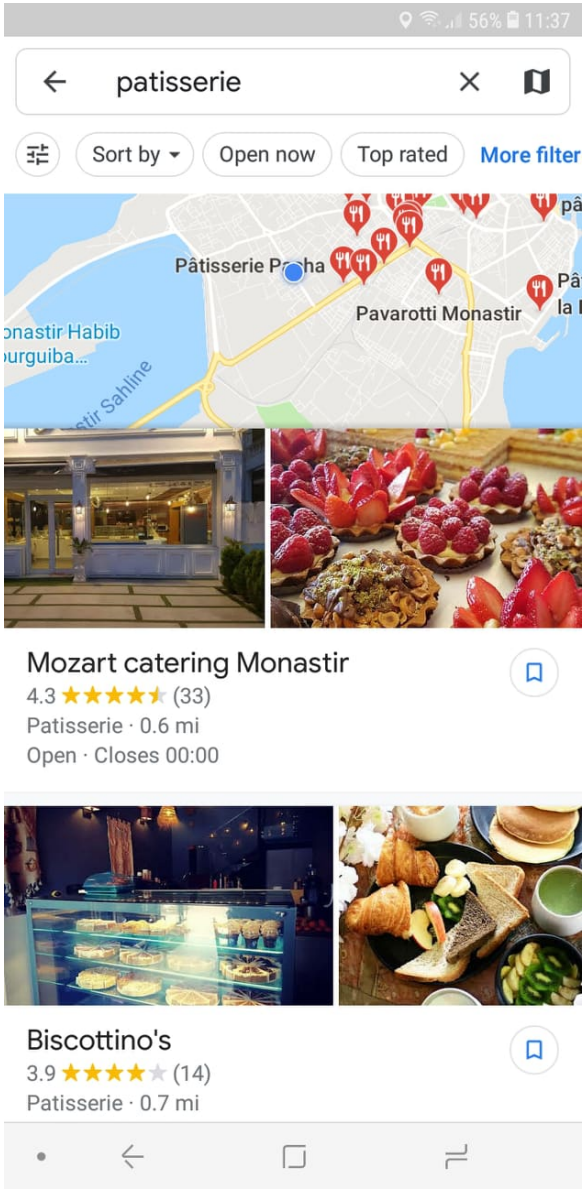

Figure 1.2: Google maps[2]

[2]Source: https://www.google.com/maps

1.1.3 Bakery days

"Bakery days" gives you the ability to design your cake as presented in Figure 1.3:

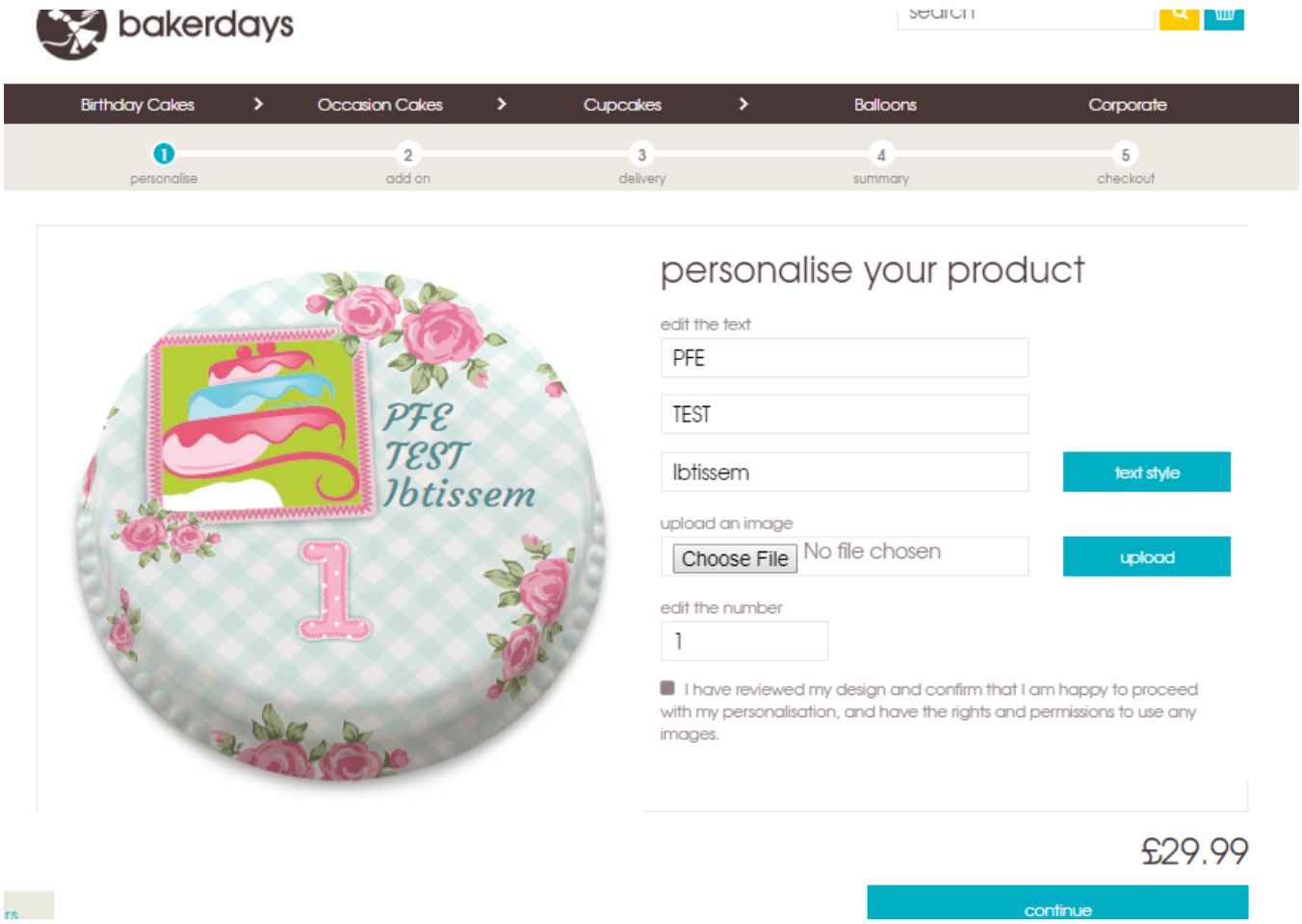

Figure 1.3: Web site: Bakery days[3]

1.1.4 YELP

"Yelp" has tremendous power in the pastries industry, and having a strong backing of positive Yelp reviews is like having a flock of golden geese reviews from Yelp can do wonders for any business.

Advantages of **"Yelp"**:

- The application provide two parts, the business owner's section presented in Figure 1.4 and the visitor in Figure 1.5.
- The user can add his own business by adding as many details as possible.

[3]Source: https://www.bakerdays.com/

- He can respond to the reviews.
- Manage his shop.
- A visitor can rate and review a shop.

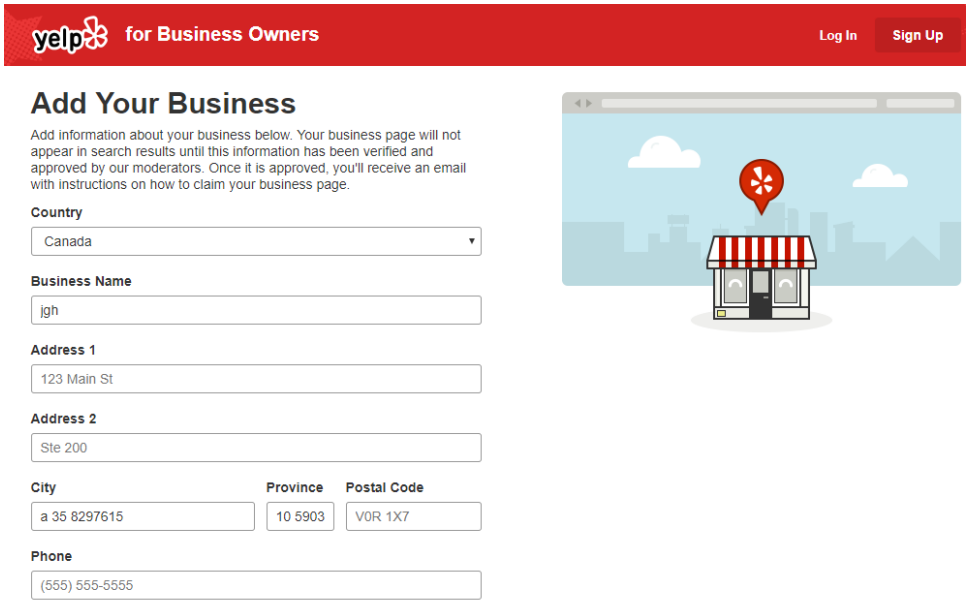

Figure 1.4: Web application: Yelp (business owner)[4]

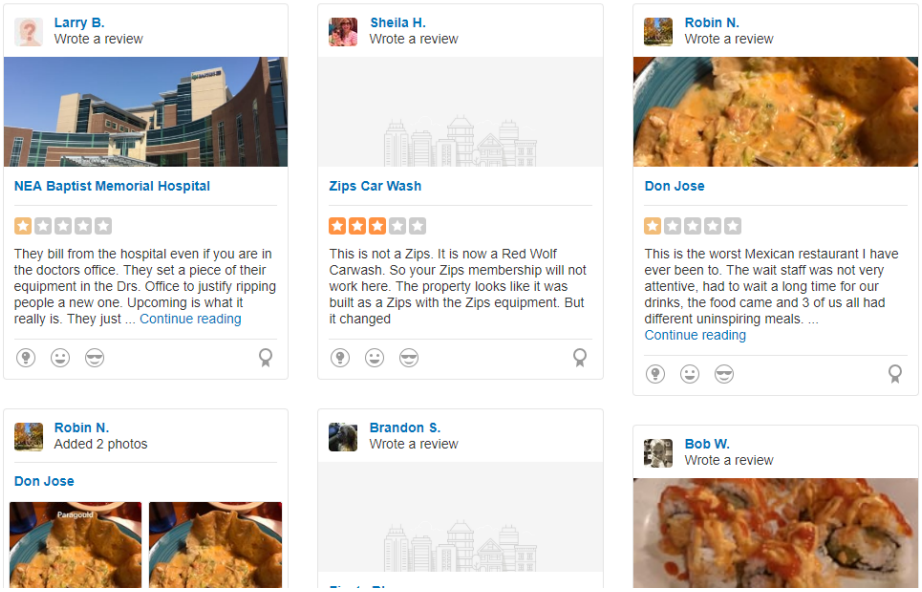

Figure 1.5: Web application: Yelp (user)[5]

[4]Source: https://biz.yelp.ca
[5]Source: https://www.yelp.com

1.1.5 Criticism of the existing

As shown in Table 1.1, there are number of disadvantages that can not be ignored.

Table 1.1: List of disadvantages of the existing.

Existing	Disadvantages
Social media	Time consuming. Risk of being wrongly informed about the place. No information about the prices. No security in passing orders.
Google maps	Places are added by anyone not the owner. The suggestion are only based on the location of the user. No on-line orders. Not enough information about the products.
Bakery days	The personal design includes only the image on the cake or the writing. The site is for one specific bakery. Only available as a web site. Covers only the United Kingdom.
YELP	The site is designed only for reviews. Can not pass an order. Too many types of business. Not available worldwide.

1.2 Solution and requirements

This section will give a small presentation of the solution regarding previous criticism followed by the requirements for this solution.

1.2.1 Proposed Solution

As a solution for the previous issues, I provided a web/mobile application named **"Cake it up !"**.

In one hand, the mobile part will be dedicated to the clients everywhere. It provides an easier way to view all the pastries nearby or far away and get access to all their product's information (price, description, ...). In addition to that he can pass orders from his phone. He also has the capability to rate and give his own opinion about the bakeries or even block/report a shop. Finally, he can create his own cake design in 3D (forms, layers, colors, perfumes, ...).

On the other hand, the web application is accessible for bakeries owners, so they can contribute by adding their place to the application with detailed description of their products (prices, ingredients, ...). The owner can also manage the orders of his clients or report any suspicious ones. The application gives also a daily statistic of the shop and its performance.

1.2.2 Requirements

The purpose of the requirements also called specifications is to give a clear picture of the application, in terms of the capability required. It will also identify constraints on the software solution, that are important in guiding decision making throughout the development process.

Describing what the software system does and how it does so effectively usually means describing it from more than one viewpoint. Each viewpoint will convey some information about the system that other viewpoints omit. We distinguish two viewpoints Functional and non-Functional.

1.2.2.1 Functional requirements

The Functional requirements are any requirement which specifies what the system should do. The application should be able to response to all the user's needs:

- Web
  - Web master
    - Connexion as a super Admin of the application
    - Manage reports
    - View statistics
    - Activate/deactivate pastries
    - Block clients
    - Manage account
    - Receive notifications
  - Bakery manager
    - Create an account
    - Connexion
    - Manage shop
    - Manage complains
    - Manage orders
    - View statistics
    - Manage products
    - Manage events
    - Receive notifications
- Mobile
  - Visitor
    - Inscription
    - Connexion
    - Manage cart

    - Consult pastries and products
  - Customer
    - Connexion
    - Manage account
    - Manage orders
    - Create complains
    - Manage cart
    - Add views
    - Consult pastries and products
    - Design a 3D cake
    - Report a Bakery
    - Access to history of orders
    - Receive notifications
    - Rate a pastry shop

1.2.2.2 Non-Functional requirements

Non-Functional requirements are any requirement which specifies how the system performs a certain function. They generally specify the system's quality attributes or characteristics.

- Security: every account is secured by a encrypted password
- Performance: quick response
- Reliability: fault tolerance
- Maintainability: easy to fix and evolve
- Availability: the application can be used anytime and anywhere

1.3 Used methodology

As a methodology we decided to use one the waterfall models which is the V-model. The V-Model is a unique, linear development methodology used during a Software Development

Life Cycle (SDLC).

The V-Model depicted in Figure 1.6 offers a finer grained view of the various steps and interactions pertaining to the development process and can be regarded as the most commonly used work-flow that applies at system or software level.

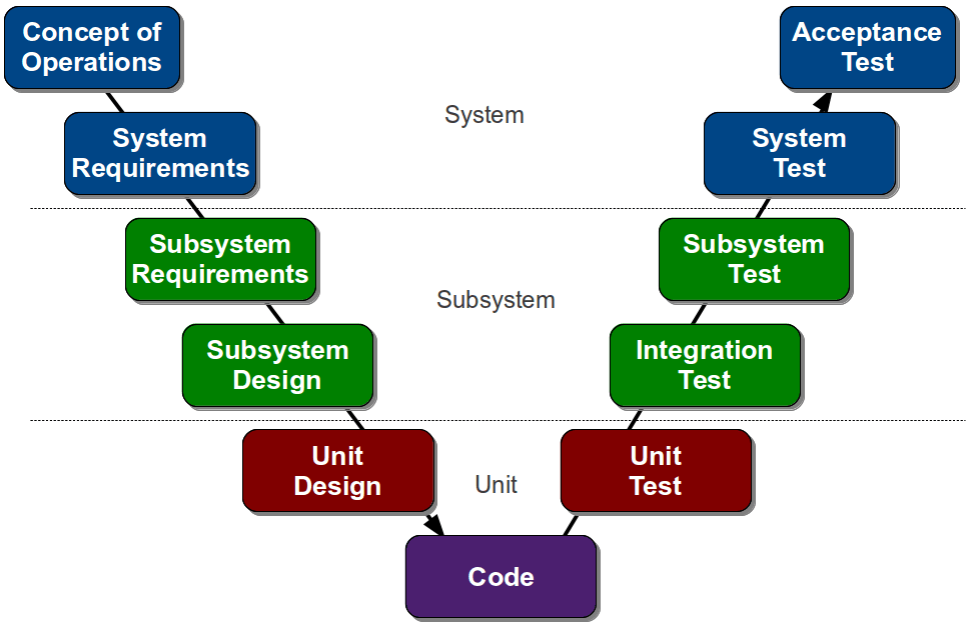

Figure 1.6: Methodology: V-Model[6]

1.4 Use case diagram

The purpose of a Use Case Diagram (UCD)[7] in Unified Modeling Language (UML)[8] is to demonstrate the different ways that a user might interact with a system.
A use case helps represent:

- Scenarios in which your system or application interacts with people, organizations, or external systems
- Goals that your system or application helps those entities (known as actors) achieve

[6]Source: GitHub.

[7]A use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system

[8]The OMG's Unified Modeling Language™ (UML®) helps you specify, visualize, and document models of software systems, including their structure and design, in a way that meets all of these requirements

- The scope of your system

1.4.1 Actors

"Cake it up !" includes two main actors and one secondary actor. They are divided into two categories web and mobile as shown in Table 1.2.

Table 1.2: List of actors.

	Actor	**Role**
Web	Pastry manager	He is the owner of the pastry, he is responsible for the managing of his own bakery.
	Administrator	This is a secondary actor, he has access to all the data of both web and mobile applications.
Mobile	Customer/Visitor	He has only access to the mobile application.

1.4.2 Modeling Language

For the modeling language, we chose UML [1] short for Unified Modeling Language.

UML is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.

1.4.3 Use case: web

This section will give a detailed description of the general and detailed use cases that exists in the web application.

1.4.3.1 General user case

Figure 1.7 shows the general use case of the web application which will be explained in Table 1.3.

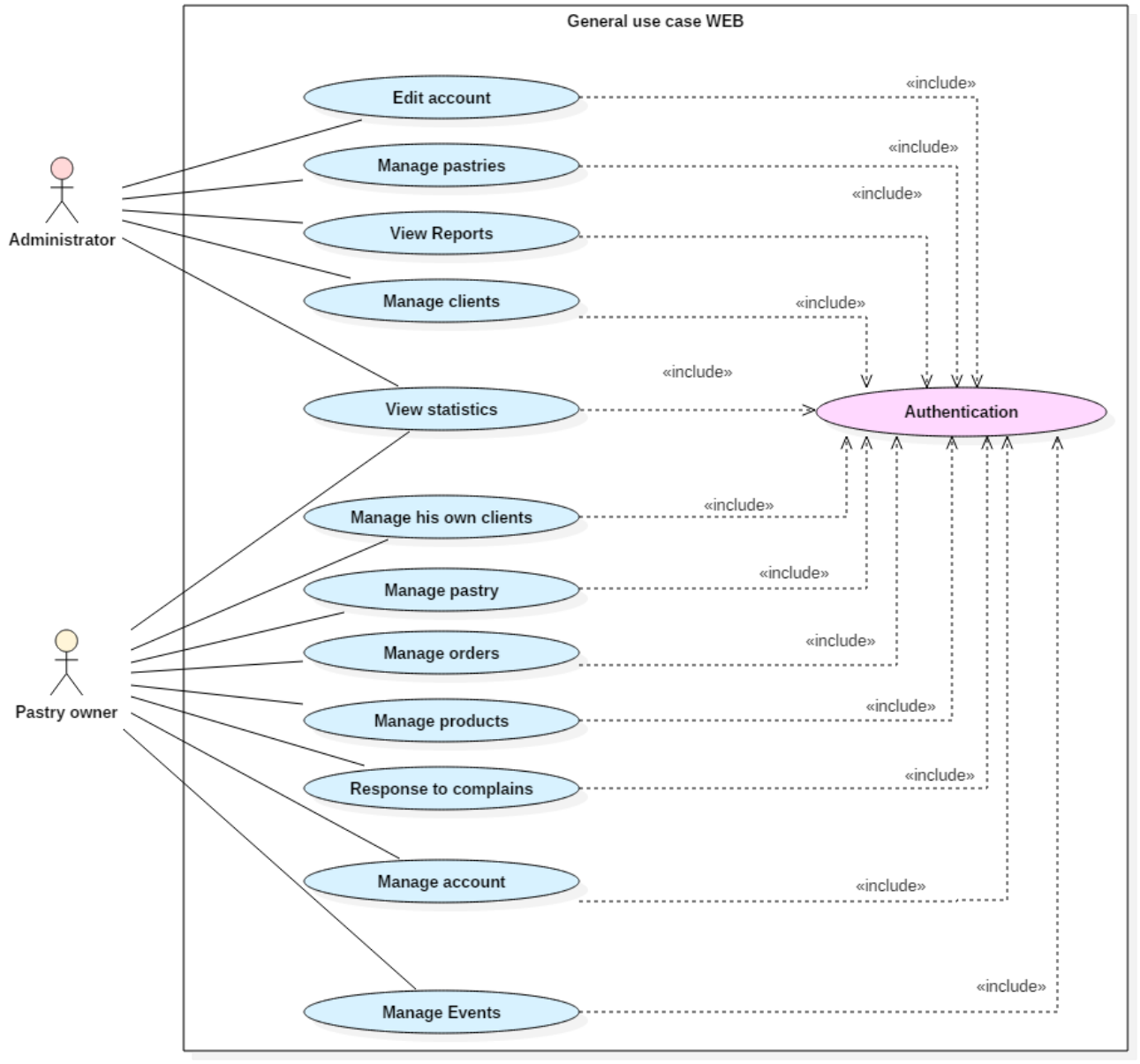

Figure 1.7: General use case: Web

Table 1.3: General use case web explanation

	Use case	**Explanation**
Administrator	Edit account	The administrator has access to his own private account where he can edit his personal information,
	Manage pastries	He can also view the list of all the pastries and manage them but with limits,
	View Reports	Viewing the reports is also an option, a report can be send from a client or a pastry,
	Manage clients	Managing the clients is one of the Administrator's tasks.
Patsy owner	Manage his own clients	The second user which is the pastry's owner have access to the list of his clients (the ones who made at least one order with them),
	Manage pastry	He can view the information of his pastry and edit or deactivate it,
	Manage orders	All the orders made by a client can be seen by the pastry's owner, and he has multiple actions he can use on an order(accept, refuse, cancel, ...),
	Manage products	Every pastry has a list of products that are created and managed by the pastry's owner,
	Response to complains	The complains made by the clients are only visible to this user, he has the ability to response to these complain by a message,
	Manage account	The pastry's owner have access to his own information (email, password)
	Manage events	The pastry's owner have a secondary option which is handling his private calendar

1.4.3.2 Administrator's detailed use case diagrams

This section will cover some of the detailed use case diagrams for the Administrator.

Use case: Manage Pastries

Figure 1.8 shows the detailed use case diagram for the general use case "Manage pastries". To manage the list of pastries means the Administrator has to ability to block, activate or deactivate a pastry and to see the details a specific pastry.

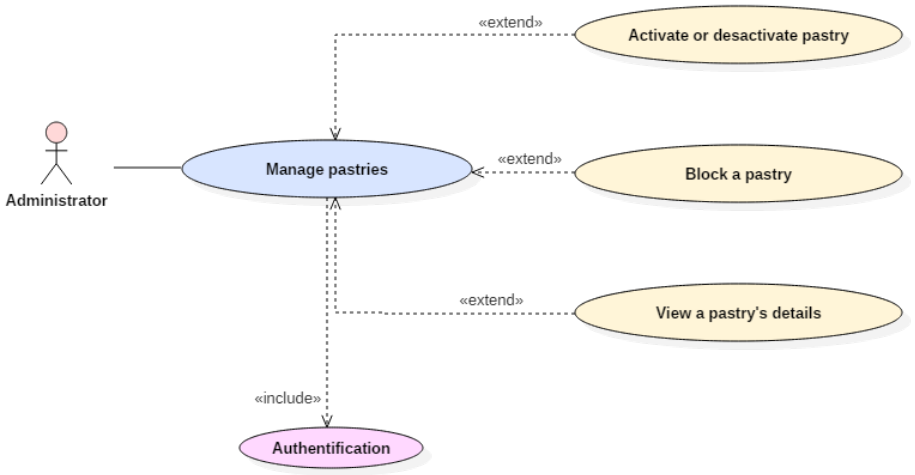

Figure 1.8: Detailed use case: Manage pastries

Use case: Manage Clients

The administrator can also manage the list of all the clients, he is able to block a client or view the last one's details as showing in Firgure 1.9.

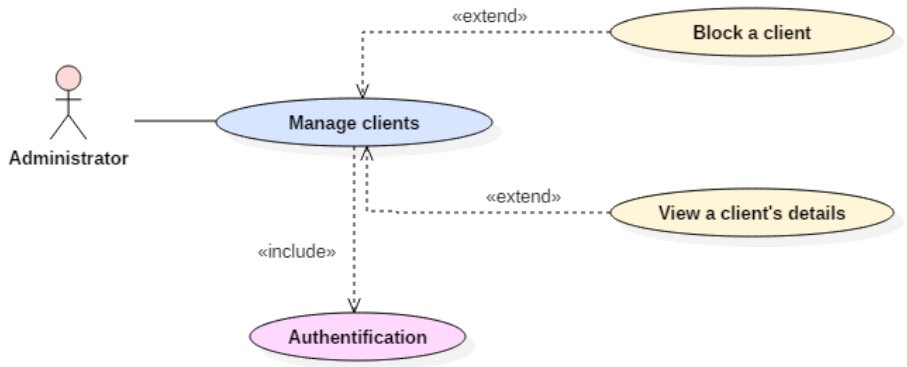

Figure 1.9: Detailed use case: Manage clients

1.4.3.3 Pastry manager's detailed use case diagrams

The pastry's manger or the pastry's owner has multiple tasks as showing in the general use case diagram, and each use case is divided into a number of other use cases.

Use case: Manage his own Clients

After the pastry's owner authenticate, just like the Administrator, a pastry's owner can manage list of clients, the only difference is the clients list for the pastry's owner has only the clients which they passed an order with the pastry. He can report a client or view the client's information like the Figure 1.10 shows.

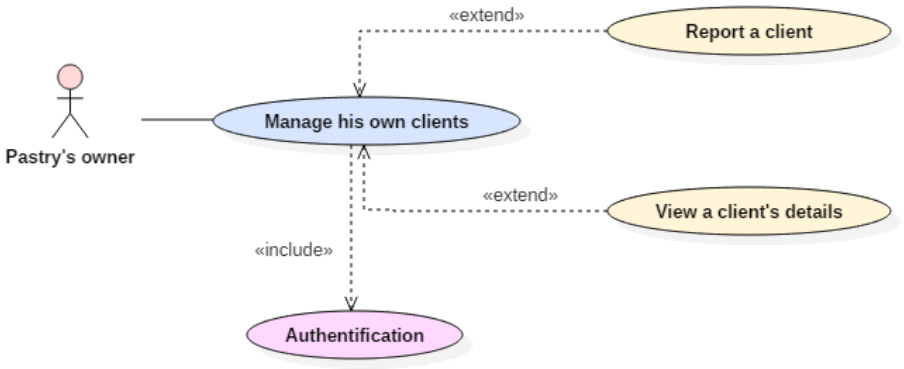

Figure 1.10: Detailed use case: Manage his own clients

Use case: Manage pastry

Figure 1.11 gives a detailed use case diagram of the use case 'Manage pastry', showing that a pastry's owner can desactivate his pastry which mean it will no longer be visible in the mobile application, he can consult all the details considering his pastry such as name, description, schedule, ..., from the same action he can view the list af reviews written about his pastry.

Editing a pastry also has two use cases, the first one is editing all the personal information (name, description, location, ...), and the second is to change the color of the template showing on the mobile application for his pastry. None of the previous actions can be done without authentication.

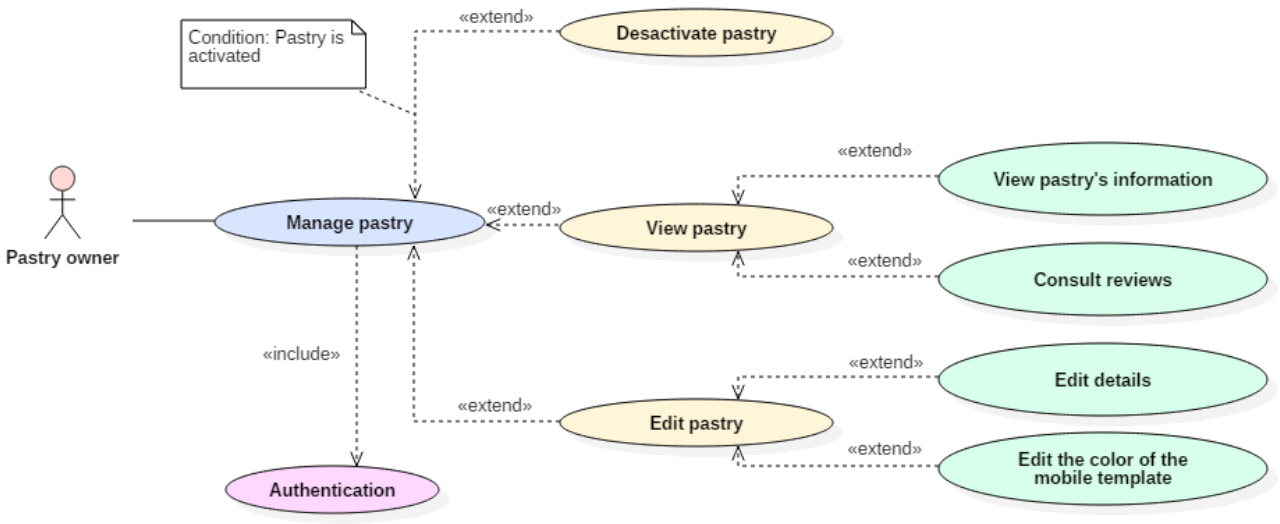

Figure 1.11: Detailed use case: Manage his own clients

Use case: Manage products

Managing the list of products includes, viewing, editing and deleting the product as showing in Figure 1.12

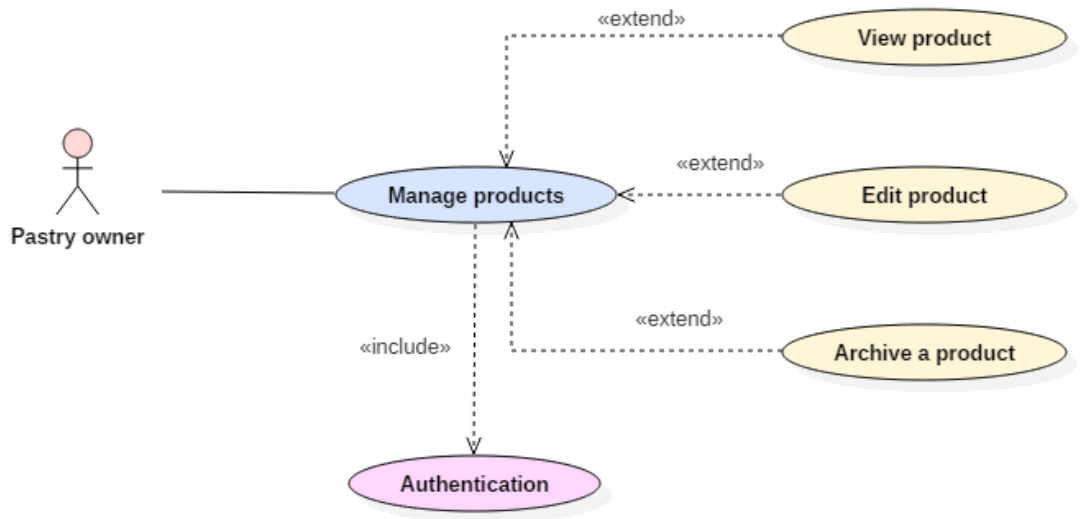

Figure 1.12: Detailed use case: Manage products

1.4.4 Use case: Mobile

1.4.4.1 General use case

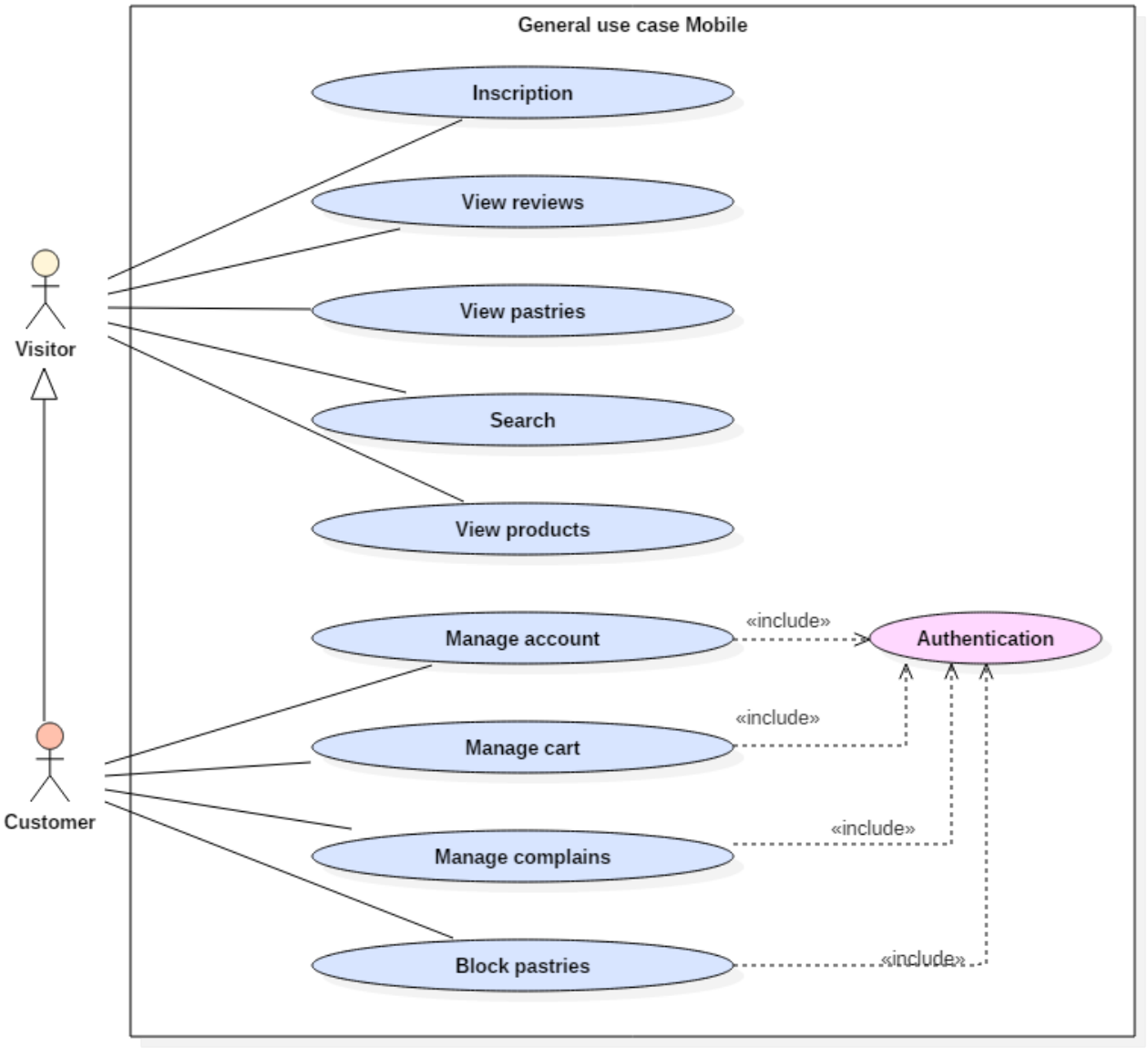

Figure 1.13: General use case: Mobile

1.4.4.2 Detailed use case

Conclusion

Chapter 2

Conception

Introduction

In this second chapter we will dedicate it to the conception part, in which we will include our class diagram with a detailed explanation. Then, we will be adding different sequence diagrams with more explanation.

2.1 Architecture

2.1.1 Mobile

2.1.2 Web

2.2 Class diagram

Class diagrams are one of the most useful types of diagrams in UML as they clearly map out the structure of a particular system by modeling its classes, attributes, operations, and relationships between objects.

Table 2.1: Class diagram explanation

Entities	Definition
User	The user can be the place owner, place manager or the foodies
Place manager	The person who manges the place.
Place owner	The person who owns the place.
Foodie	The person who will be attending the place.
Place	Places are added by the owner.
Reservation	The foodie can book a dish in a place.
Menu	The menu is composed from dishes.
Dish	The food that a foodie will come for.
Ingredient	A dish has ingredients.
Likes/Dislikes	The foodie can like or dislike a place.
Comments	The foodie can comment on a place.
Rating	The foodie can give the place a rating.
Offer	A place can make an offer.
Location	A place has a location.

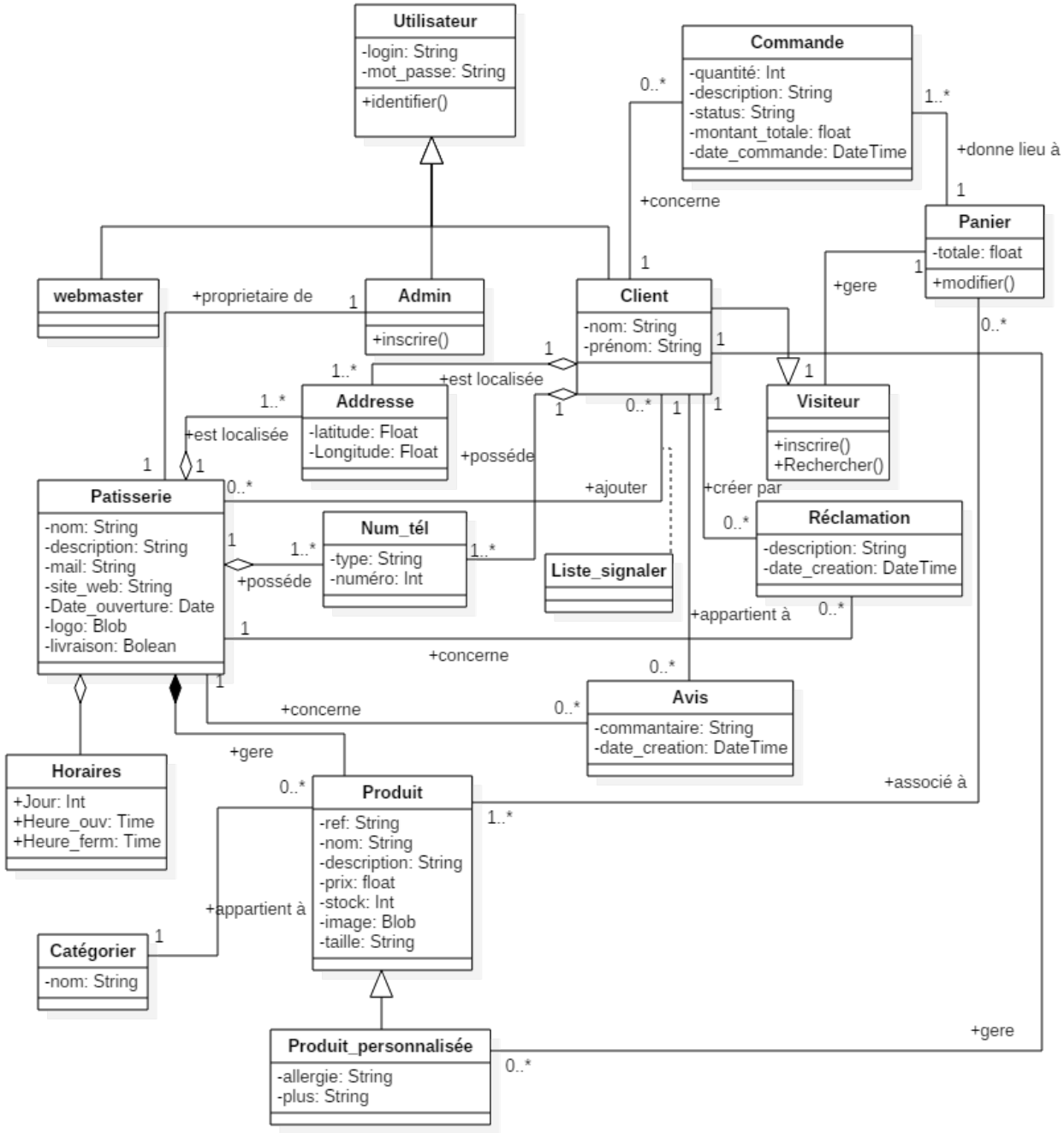

Figure 2.1: Class Diagram

2.3 Interaction sequence diagram

A sequence diagram (SD) is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

2.3.1 Web

2.3.2 Mobile

Conclusion

Resuming this chapter, we ended up having a clearer vision about our objectives, now we pass on to the next chapter in which we will explain more about implementation.

Chapter 3

Realization

Introduction

3.1 Choice of the architecture

3.2 Used technologies and frameworks

3.3 Used software

3.4 Implementation

3.4.1 Web

3.4.1.1 Shared interface

3.4.1.2 Administrator's interface

3.4.1.3 Bakery manager's interface

3.4.2 Mobile

Conclusion

To finalize this chapter, we ended having a clearer vision about the different functionalities, which will guide us to end up this project report with a conclusion summing up what we

did, finalizing it with our perspectives.

General conclusion

Bibliography

[1] M. Fowler, *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional, 2004.

[2] A. Kumar, *Sencha MVC architecture*. Packt Publishing Ltd, 2012.

[3] L. Red Hat Middleware, "Hibernate - persistance relationnelle en java standard,"

[4] C. Scarioni, *Pro Spring Security*. APPRESS.

[5] J. Spurlock, *Bootstrap: Responsive Web Development*. " O'Reilly Media, Inc.", 2013.

[6] J. Ducket, "Javascript and jquery: Interactive front-end web development. 1 uppl," 2014.

[7] B. Phillips and B. Hardy, "Android programming: The big nerd ranch guide, big nerd ranch," 2013.

Appendix