Acknowledgments

I would like to express my special thanks of gratitude to my teacher and tutor Mrs. FATMA SBIAA, that helped, supported, and encouraged me to do this wonderful project.

Also, I would like to thank all my teachers for sharing their knowledge, experience, and wisdom with me throughout my studying period.

Finally, I highly thank the members of the jury and whom I have the honor to present this project to them

Table of contents

Table of Figures

GENERAL INTRODUCTION

1. NTRODUCTION

1.1 Invention

Invention is the act of creating new technology. It involves a new scientific or technical idea, and the means of its embodiment or accomplishment. To be patentable, an invention must be novel and have utility.

1.2. Innovation

Innovation may be used synonymously with "invention" or may refer to discovering a new way in which to use or apply existing technology. Everett Rogers thought of innovation as an idea, behaviour, or product that appears new to its potential adopter. There are five main attributes of innovative technology: Relative Advantage, Compatibility, Complexity, Trialability, and Observability.

- **Relative advantage** means the product or behaviour is perceived as being better than the alternatives by the person adopting the innovation. Better can mean a lot of different things. It can be a device that can peel a potato faster so it saves time or a seat belt that offers the advantage of greater safety.
- **Compatibility** refers to how the innovation aligns with the adopter's lifestyle.
- **Complexity** is how easy or difficult innovation is to understand. The easier an innovation is to understand and use, the more likely it is to be adopted. Complex innovations face an additional challenge to mainstream adoption.
- **Trialability** refers to the process of testing the innovation to see if, or how well, it works. Extensive testing usually occurs before an innovation is adopted or taken to market.
- **Observability** involves seeing the product or behaviour in action. It can demonstrate how it can be used. It is easier to get potential adopters to simply observe an expensive product like a car than it is to get all of them in one for a test drive. Also, the more people around you that you see using a product, the more likely you feel like buying that product too.

1.3. Diffusion

Diffusion pertains to the spread of technology throughout a society or industry. It is the process by which a new idea, product, or behaviour is accepted by the market. Technology diffusion means the spread of usage/application of new technology from its current user to others.

The diffusion of innovation theory, introduced by Everett Rogers, explains how different groups of people adopt innovation in different ways, in order to best suit their own needs or desires.

2. THE HOST OF THE IDEA

Now days the usage of mobile apps and websites increased a lot, those techs offer efficient online functionality's, despite that there are vital and important institutions still adapt classic strategies.

For That reason I am happy to introduce you to my solution(Name of the app) that can help patient have a better experience trying to get an appointment in the outpatient consultation of the local Hospitals.

The current solutions embraces an old methodology , based on manual registration, the patient have to wait his turn so he can begin his registration Procedure in the registration office , this method is highly efficient when the number of patients is relatively small, however that's rarely the situation and the number of patients is round 500 daily, sadly that causes an extreme bottle neck effect when it comes to registering Hundreds of patients everyday.

To solve this problem I plan to present a solution that consists of a Mobile app that provides services to patients and helps them have a better experience

When it comes to trying to have a medical appointments and provide them with an AI guide to help them predict there illness before taking an appointment.

This report presents all steps and descriptions of the project and it is composed of 3 main Chapters :

- First chapter presents the general frame of the project and the ecosystem around it ,also our study of the needs and constraints of the Patient

- The second chapter presents the conceptual study of the project
  With the help of UML tools and technology

- The third chapter present the graphical interfaces used and established in the project and explanation of the basic functionality of the UX design

FIRST CHAPTER:

THE STUDY OF THE PROJECT

Chapter 1:Introduction

First of all we must understand the Dimensions of this project and its use case in the real world situations, With that we will present in the first place the public healthcare Institution that is hosting the idea of the project, after that we will introduce a better understanding for the concepts supported by the projects and its usage in the modern local environment without forgetting to mention the functional and non functional needs of the project to better understand the consumers needs and constraints facing the solution.

General Context of the project

The public healthcare fields faces some critical problems especially when it comes to organizing and administering patients ,

That leads us to the host of this project (Farhat Hachad Hospital) that is a teaching Hospital ,specifically the outpatient consultation that welcome every day an average of 500 patients, coming from every state of Tunisia.

The problem reside in managing the flow of patient each morning,

That create a bottle neck effect when it comes to assigning medical appointments to patients .

For that I am glad to introduce the project <<Developing a Smart app to manage ,guide ,help to patient and staff have a better experience

We have to understand the ecosystem around the patient environment to better maximize the efficacity of the project

2.Presentation of the Institution

Figure 1:FH hospital

Created in 1938, the Farhat Hached Hospital in Sousse is a category "A" public health establishment with legal personality and financial autonomy. Due to its history and the multiplicity of its medical services, the hospital is considered an important medical center in the region providing highly specialized therapeutic services.

2.1 The environment of the project

The patients targeted for this project are the outpatient consultation visitors, after investigating and visiting the principal of the consultation provided me with these information's:

. The consultation is starting from 7am ,still the gathering is huge even from 6am , until 12am

.The staff is not sufficient enough

2.2 Current procedure to have an appointment:

2.2.1 Manual classic method

A.1. Make an appointment:

You are required to present yourself (or your loved one) to the outpatient consultations of the referral specialty * provided with a letter from your attending physician.

A.2. On the day of the consultation, you will follow these steps:

Go to the registration desk with the following documents:

Your meeting card

Your national identity card (for adults and holders of care books)

Your valid care card

A PEC decision for long-term illnesses from the CNAM if you are a beneficiary.

Once all these documents have been provided, the registration agent will register you, issue you a care number and send you to the cashier (unless you are exempt from payment).

At the checkout, you pay for your consultation and the cashier has given you a receipt

2.2.2 Via SMS

there is a service via SMS that provides some minor functionality, but it is rarely used now days

2.3. Critique of the existence

We have to mention that the current solutions are practical, however, we seek to improve the situation and make it innovative

.the absence of a mobile or web application that solves the mentioned issues

.the procedure of having an appointment have a main basic manual solution

.the patient is clueless when trying to find his illness and that can cause difficulties while communicating with the hospital staff

.the patient must be present to have an appointment

2.4 Solution

We Have taken into consideration the inconvenience of the existent solutions and that helped us to better present and understand our solution

In fact, the project that I seek to make is a mobile application that uses a machine learning model to predict a patient illness, providing him with online access(with secured Authentication) reservation and generating a QR code when accessing the hospital so he can be added to the database, plus seeing and updating his data……

3. Specification of needs

This phase is by far the most important phase because with a great and solid understanding of the system around the project, we can make a better realization of the project

For that, we can present some functional and non-functional needs of the project

3.1. The Actors

Patient : he is the main actor of the project, and we need to focus the app functionality around him,

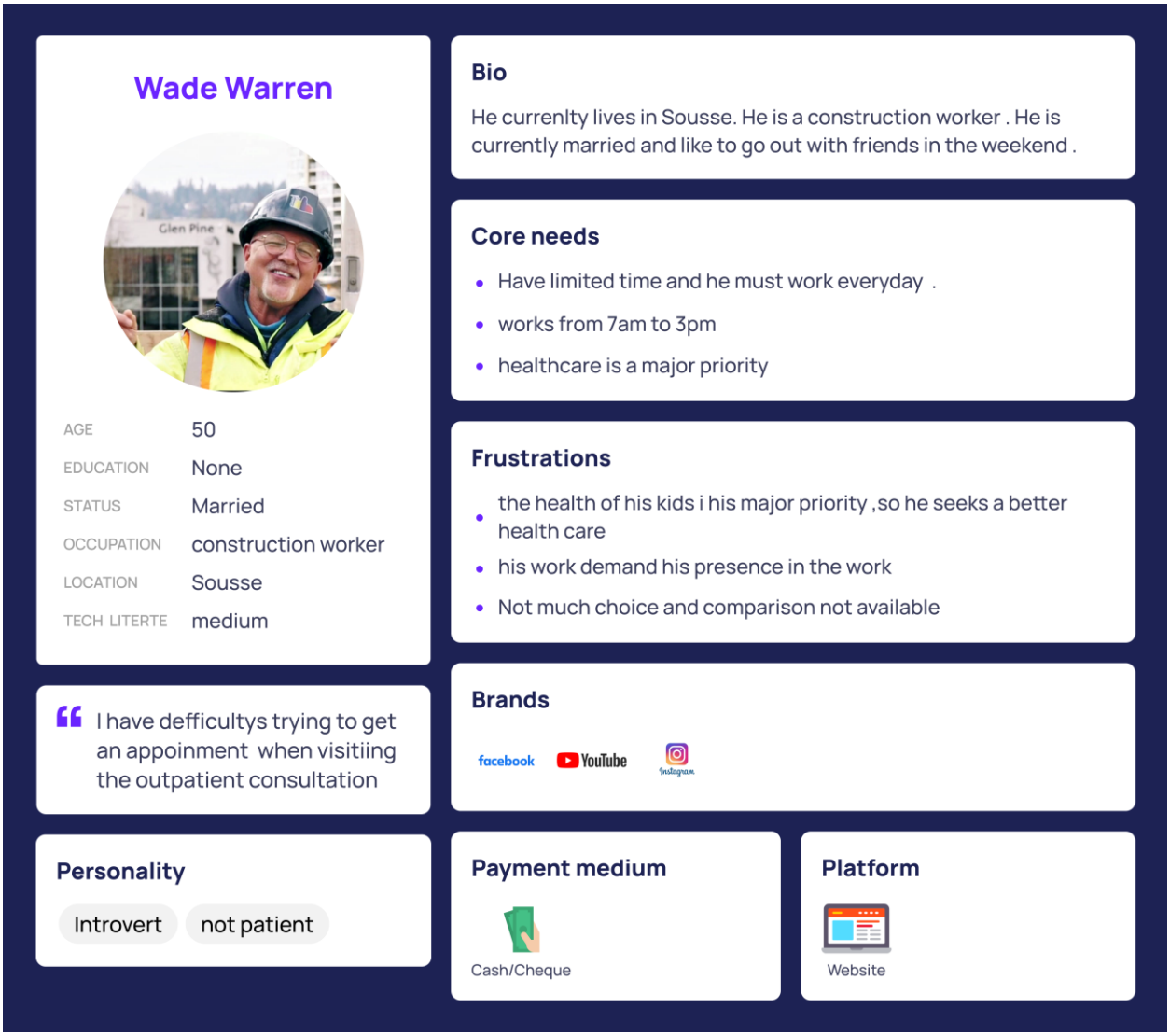


Figure 2:patient persona

There are 3 main personas of the patient (old, young, kid), 70% of patients have smartphones, and 60% of them can use the phone properly

The other 30% are mostly old people and kids, they don’t have a phone or they can not use it properly

Admin:

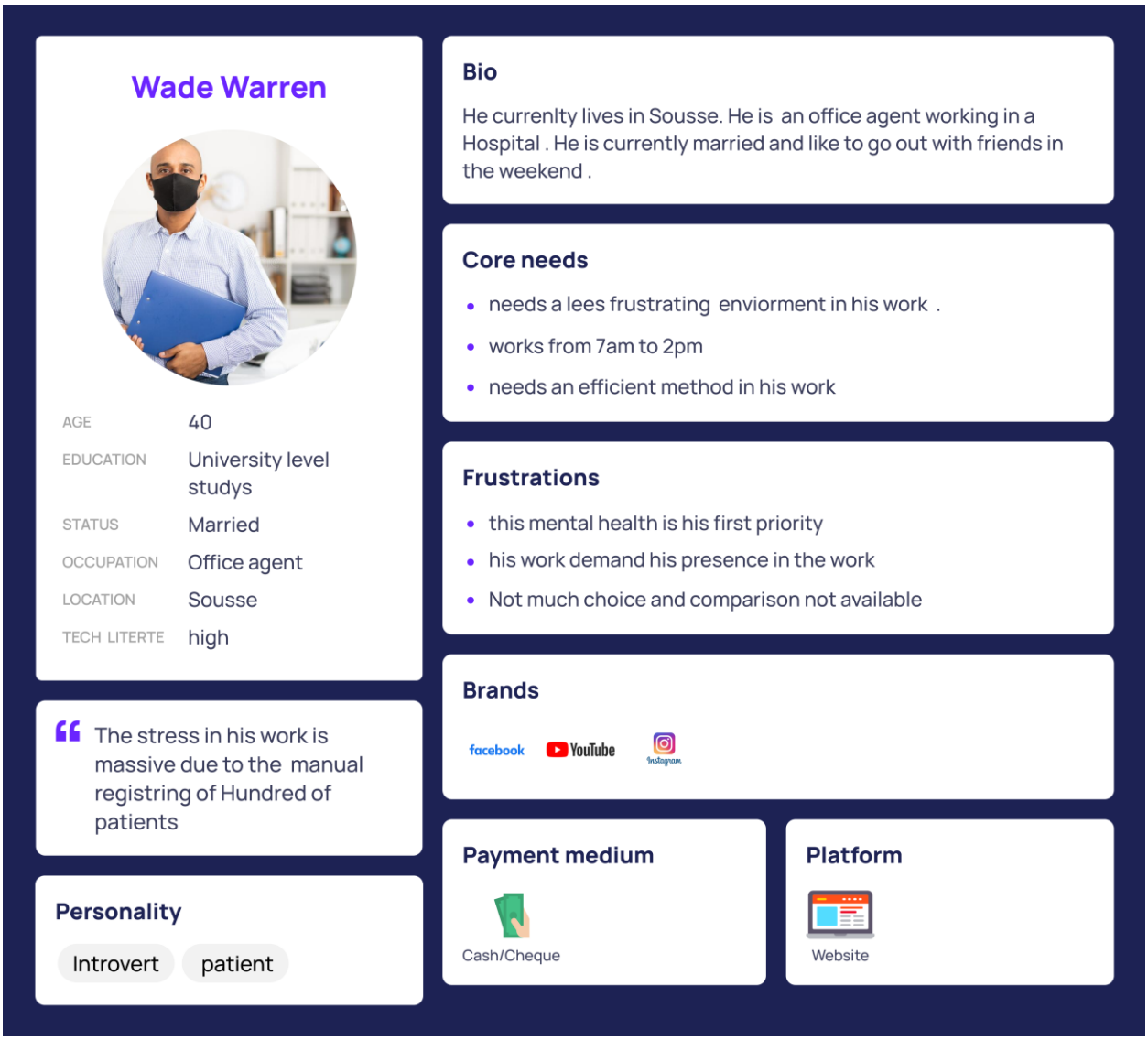


Figure 3:admin persona

The payment agent is the admin responsible for registering the patient and he has access to the database, he receives the payment from the patient after providing him with an appointment.

3.2 Functional needs

The application must cover the certain needs of its users, here we provide some details :

3.2.1 Patient

The patient wanting to have an appointment has the possibility to do this actions :

- Registering(Signup): the patient can signup with his email, CIN,….
- Predicting his illness: the patient can predict his illness via a machine learning model
- Managing the profile: he can update and read his profile information
- Generate a QR code: the patient can generate a QR code to have a pre-appointment scheduling
- Access his appointments: accessing the list of his appointments is possible for the patient
- Track his Payments: the patient has the ability to see his previous Payments

3.2.2 Admin

The admin is the agent responsible for managing the patients and Completing the procedure manually, he can do this functionality:

- Login
- Check Patients That have accessed the institution: he can track patients, that have done online registration and their information
- Adding an appointment to a given patient: the admin can assign, an appointment to a patient
- Check Reservation via QR: the agent can access the state of the patient (if he did respect his appointment or not)

4.3 Non functional needs:

These are the functionality that allows us to provide a reliable product that can respect the preferences and usage of the user ,and can adapt when facing certain dysfunctionality and risks :

. Performance: Our application must be performant, fast and have a better user experience also respect the Patient's needs.

. Reliability: The application must be tolerant of bugs and errors that can happen

.Availability: Our application will be available for Android and IOS users

.Security: The application provide a highly secured system powered by Firebase Authentication

.Ergonomics: we seek to make a beautiful UI design that helps the user have a better experience.

4. Tools used in The application :

For the sake of having a performant application ,we need a specific pack of tools that can help us make a better product :

4.1.Flutter

Figure 4:flutter logo

Flutter is an open-source UI software development kit created by Google. It is used to develop cross-platform applications for Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web from a single codebase. First described in 2015, Flutter was released in May 2017.

4.1.1Flutter advantages

1. Same UI and Business Logic in All Platforms
2. Reduced Code Development Time
3. Increased Time-to-Market Speed
4. Similar to Native App Performance
5. Custom, Animated UI of Any Complexity Available
6. Own Rendering Engine
7. Simple Platform-Specific Logic Implementation
8. The Potential Ability to Go Beyond Mobile

4.2 Firebase

Figure 5:firebase logo

Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development

4.2.1 Firebase Advantages

1. Reliable & Extensive Databases. ...
2. Fast & Safe Hosting. ...
3. Provides A Free Start to Newbies. ...
4. Google Analytics. ...
5. Firebase Cloud Messaging for Cross-Platform. ...
6. Free Multi-Platform Firebase Authentication. ...
7. Firebase Testing Services to Improve App Quality. ...
8. Increment in Revenues with App Indexing API.

4.3.Methodology used in application :

In software development, the V-model represents a development process that may be considered an extension of the waterfall model.

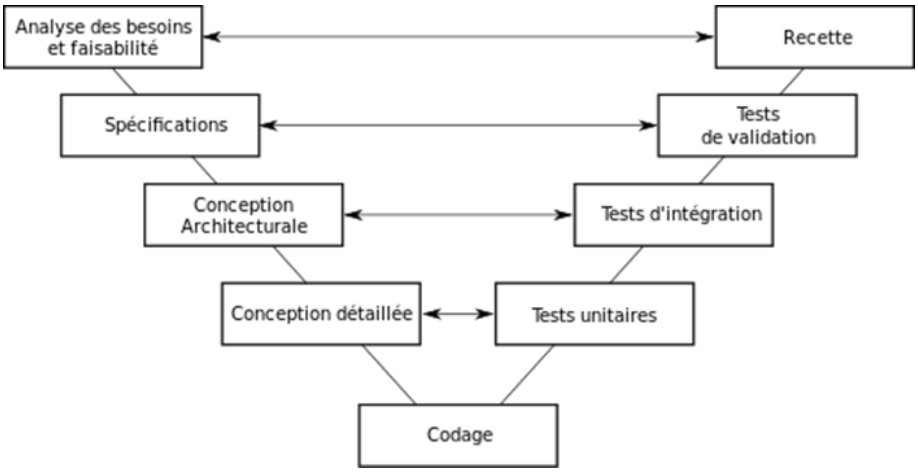


Figure 6:V methodology

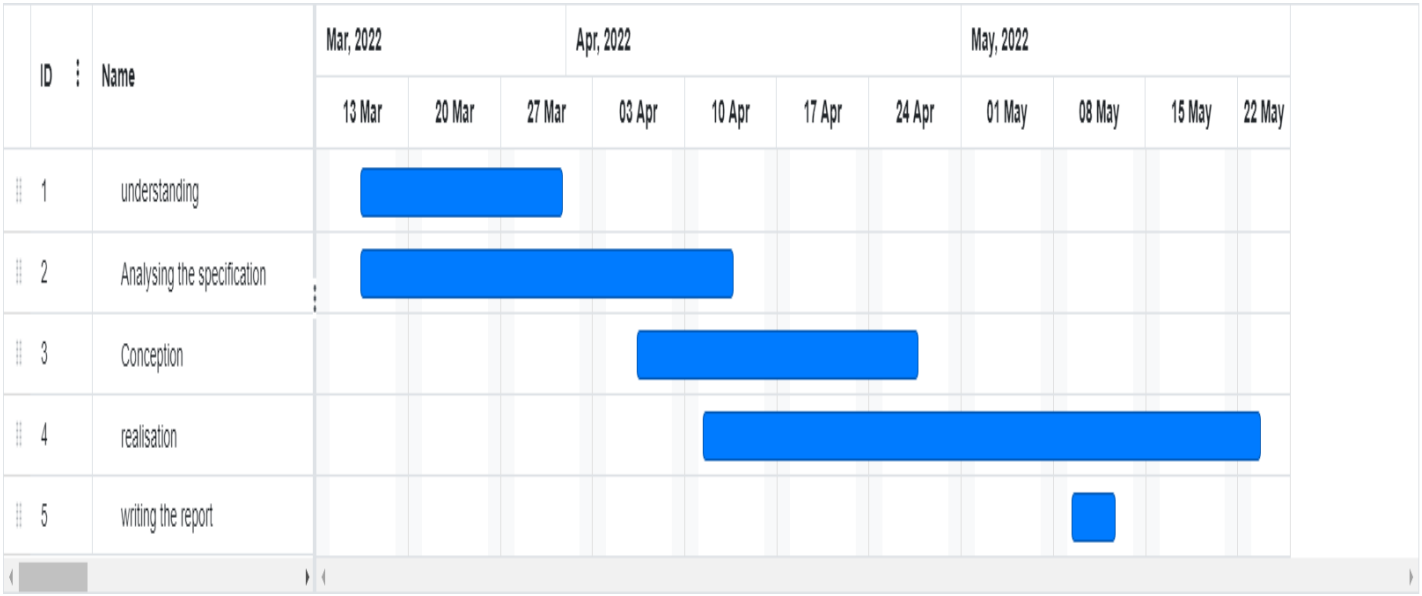
Advantages of the V-Model:

- Optimization of communication between participants through firmly-defined terms and responsibilities

- Minimization of risks and better plannability through firmly-prescribed roles, structures, and results

- Improvement of product quality through integrated quality assurance measures
- Cost savings through transparent reappraisal of the entire product life cycle

6. Chronology



Conclusion:

This chapter is dedicated to describing and defining the general context of this project ,also the presentation of the specifications and needs of the application, this previous concepts helps us understand the problem and leads us to better find a solution suited for the patient.

In the next chapter we will discuss the various conceptual concepts linked to the project so we can better understand the application .

CHAPTER 2:

Concept Study

Introduction

In the previous section we have mentioned and explained the Functionality of the solution and its different actors .

In this phase we will present in the first place the design patterns used in this project and why we did use them ,also I will provide the Use cases Diagrams, Class diagrams and Sequence Diagrams

1.Design Patterns

In software engineering, a software design pattern is a general, reusable solution to a commonly occurring problem within a given context in software design. It is not a finished design that can be transformed directly into source or machine code.

1.1.Flutter design patterns

Flutter developers use various design patterns to write clean and maintainable codebases. They often write reusable widget implementations in individual Dart files, separate the main app screens into different files, and decompose large and isolated widgets into private methods/classes.

But we can also use generic software design patterns in our Flutter apps to improve the quality of the codebase. For example, design concepts like the MVC (Model–view–controller) architecture, repository pattern, service model pattern, and Data Transfer Object (DTO) can also help us to write manageable codebases. The repository pattern in particular motivates us to decouple data access logic (database layer interface) from the business logic via an additional abstraction layer.

Almost all Flutter apps use data sources for data persistence and retrieval. The repository pattern is helpful for all Flutter developers in learning how to organize their code in a better, more manageable way.

1.2.Repository Pattern

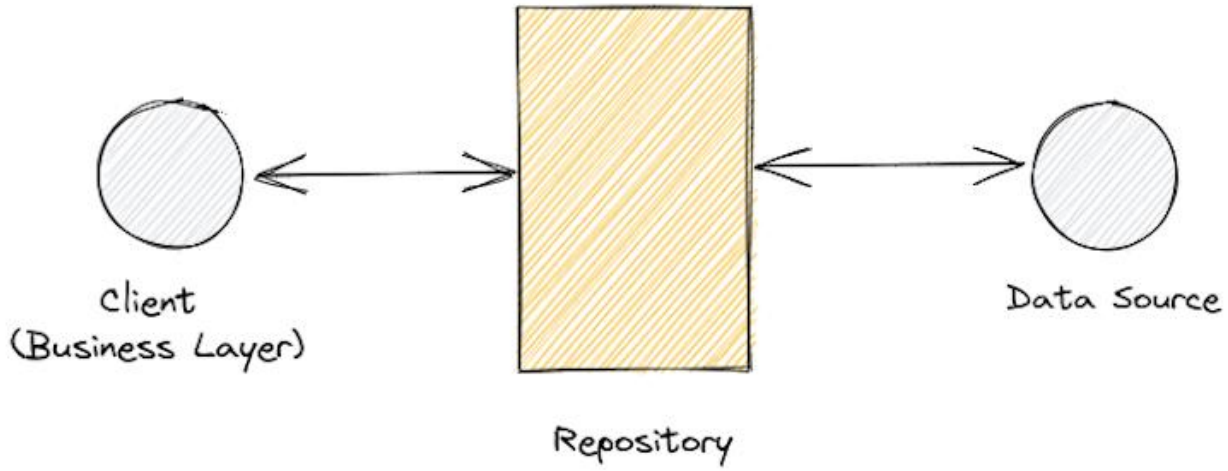


Figure 7:Repository design pattern

you can see in the above diagram, the generic repository pattern consists of three inter-connected components:

1. Client — refers to a component that initiates the data request, like a controller or service
2. Repository — provides data in a domain-friendly format via a specific API, and doesn't let clients directly access data from the source
3. Data source — provides data records according to a data-layer-specific format; the data source can be a RESTful API, SQLite connection, or MongoDB connection

The repository pattern offers us the following key benefits:

- It gives a way to access data from a centralized location to prevent data-access-related code repetition
- The codebase becomes more unit-testable because the data layer gets decoupled from the business logic
- We can easily switch data sources without doing time-consuming code changes

1.3MVC design Pattern

The Model View Controller (MVC) design pattern specifies that an application consist of a data model, presentation information, and control information. The pattern requires that each of these be separated into different objects. MVC is more of an architectural pattern, but not for complete application.

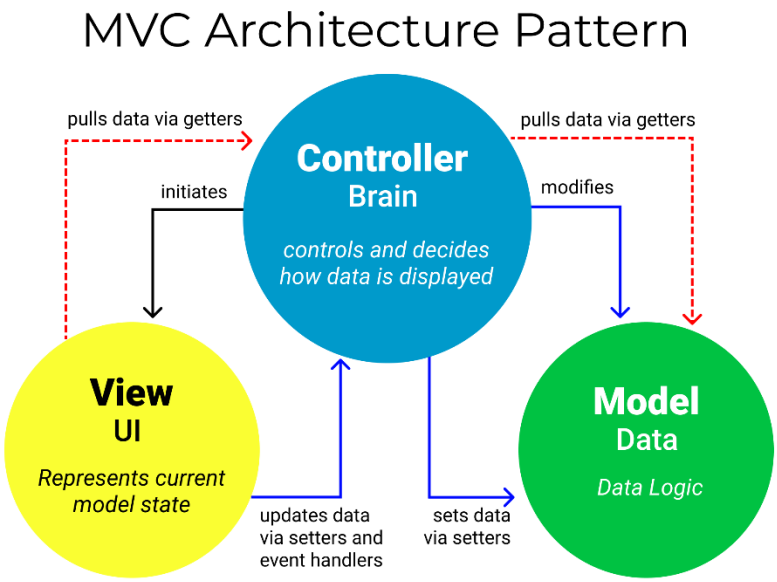


Figure 8:MVC design pattern

2.Use case Diagrams

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

2.1 General use case diagram

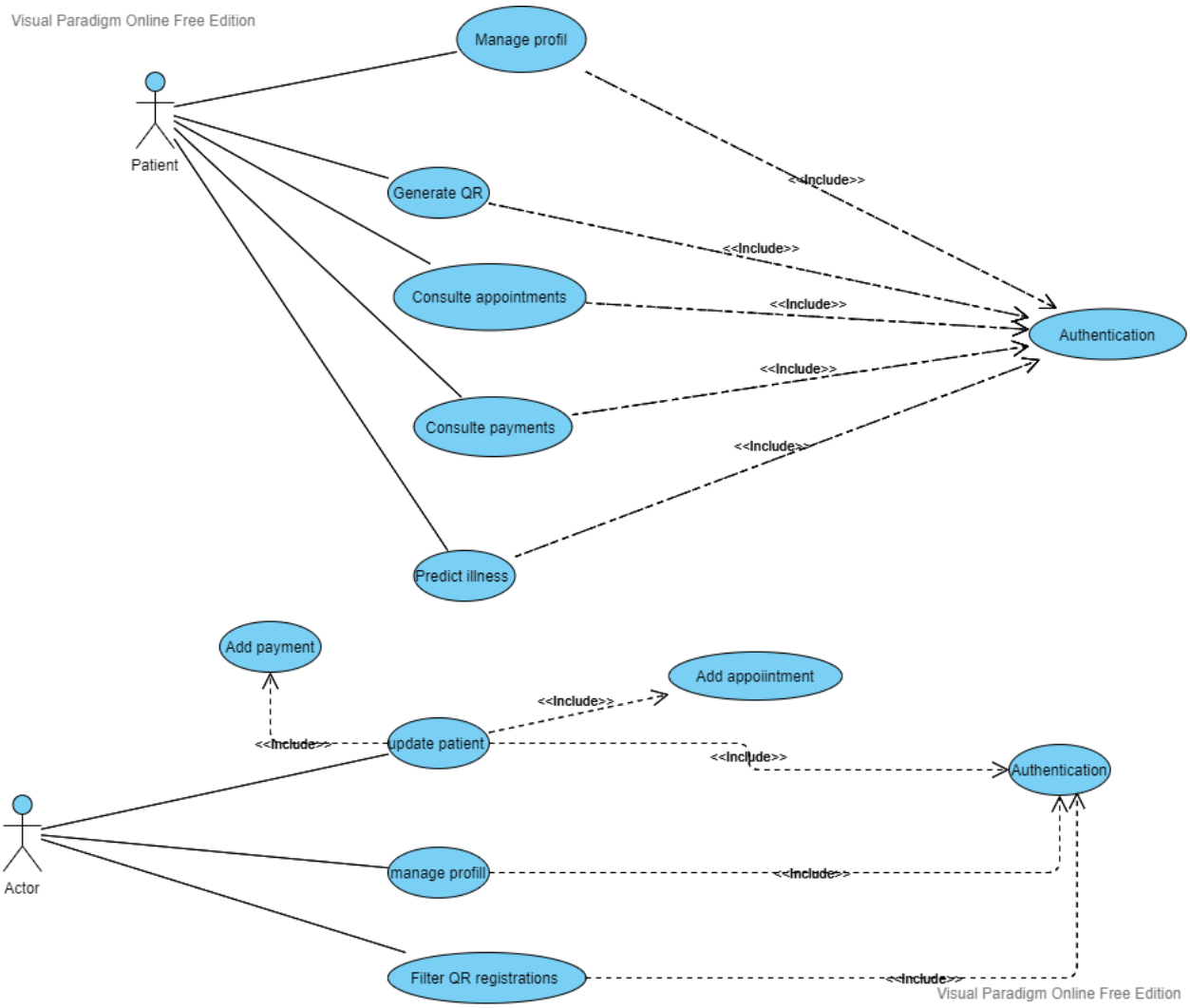


Figure 9:Generall use case diagram

2.1.1. Patient use case diagrams

2.1.1.1. Profile consulting use case diagram

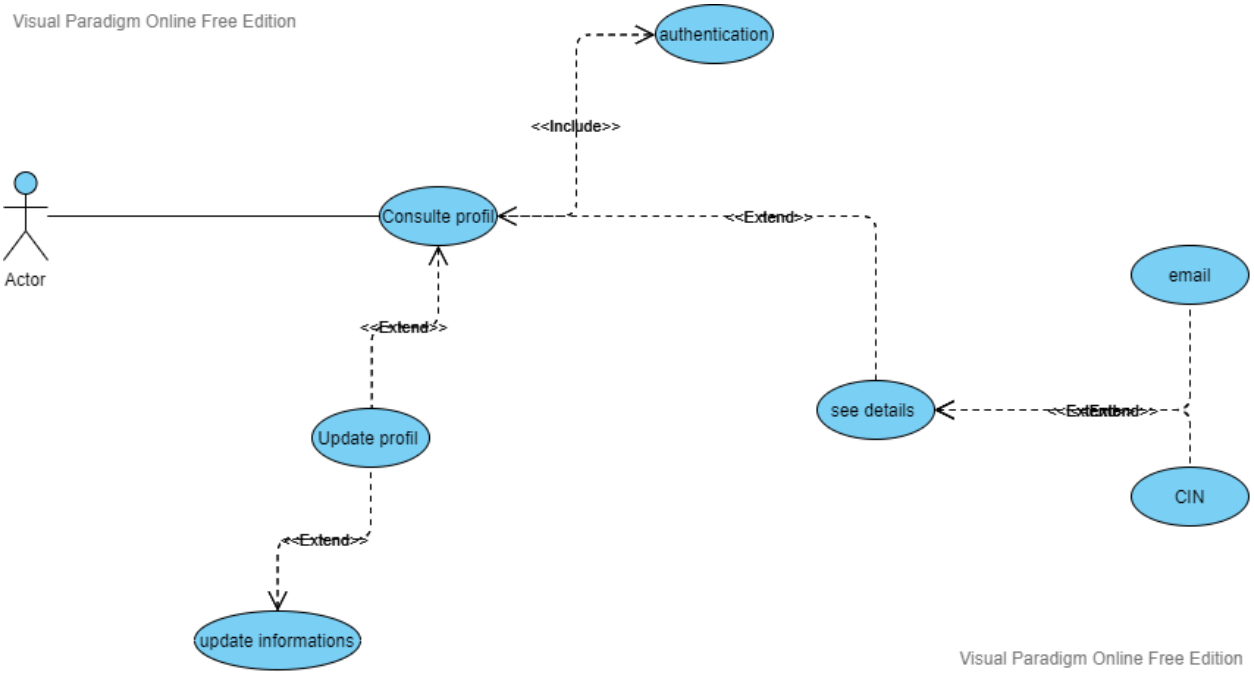


Figure 10: patient consulting profile use case diagram

Use case	Consultee profile
Main actor	Patient
Pre conditions	• Patient signed in
Post Conditions	• Profile modified
Nominal scenario	1. The patient ask for personal information's 2. The system show the interface. 3. The patient access his info . 4. The patient modify or consultee his profile 5. The patient exit
Alternative situations	**A1** : if the patient is not authenticated : a. The system block the actions and logout **A2** : if the connexion with the database is not set : a. exit the operations and wait for the connexion to resset

2.1.1.2. Payments consulting use case diagram

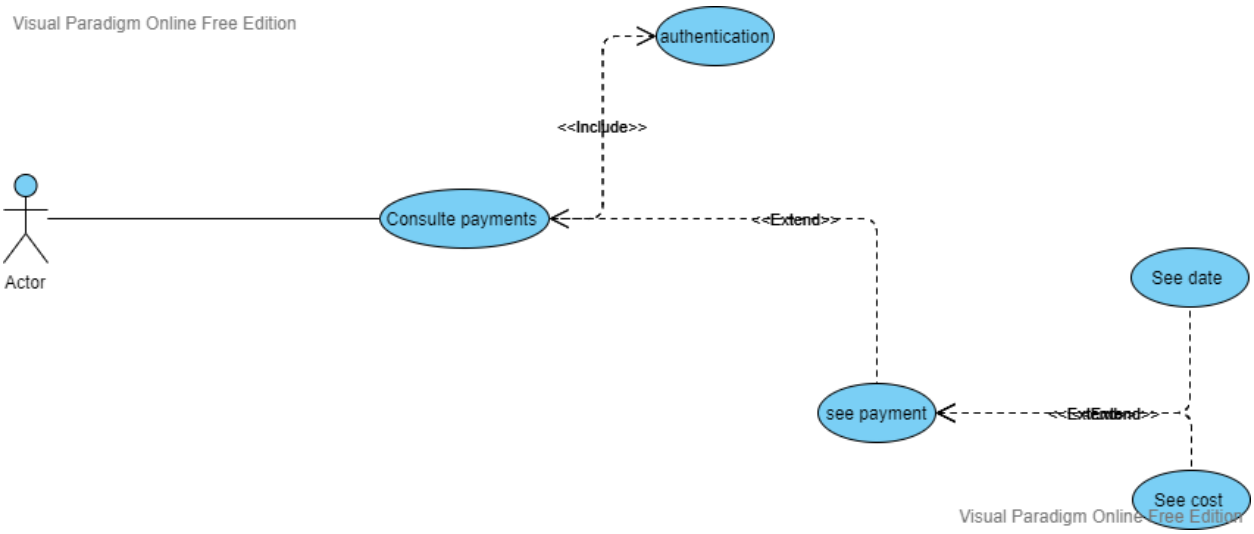


Figure 11: patient consulting payments use case diagram

Use case	Consulting payments
Main actor	Patient
Pre conditions	• Patient signed in
Post Conditions	• Payments viewed
Nominal scenario	6. The patient ask for the payment interface 7. The system show the interface. 8. The patient access the payments . 9. The patient consultee his payments 10. The patient exit
Alternative situations	**A1** : if the patient is not authenticated : b. The system block the actions and logout **A2** : if the connexion with the database is not set : a. exit the operations and wait for the connexion to resset

2.1.1.3. Predict illness use case diagram

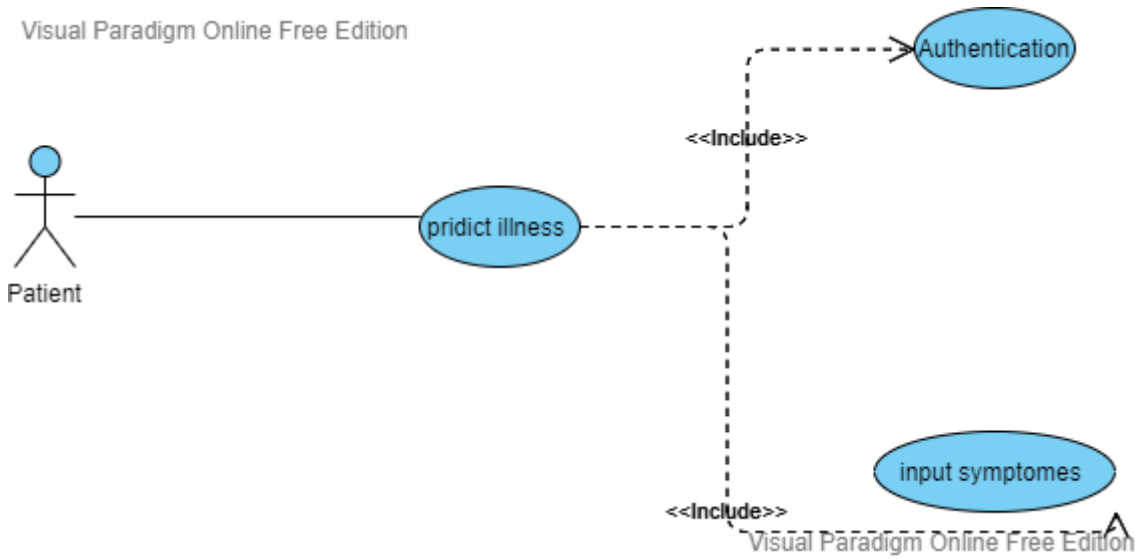


Figure 12: patient illness prediction use case diagram

Use case	Predict illness
Main actor	Patient
Pre conditions	• Patient signed in
Post Conditions	• Illness predicted
Nominal scenario	11. The patient ask for the predict interface 12. The system show the interface. 13. The patient input his symptoms . 14. The patient consultee the prediction 15. The patient exit
Alternative situations	**A1** : if the patient is not authenticated : c. The system block the actions and logout **A2** : if the connexion with the database is not set : a. exit the operations and wait for the connexion to resset

2.1.1.4. Appointments consulting use case diagram

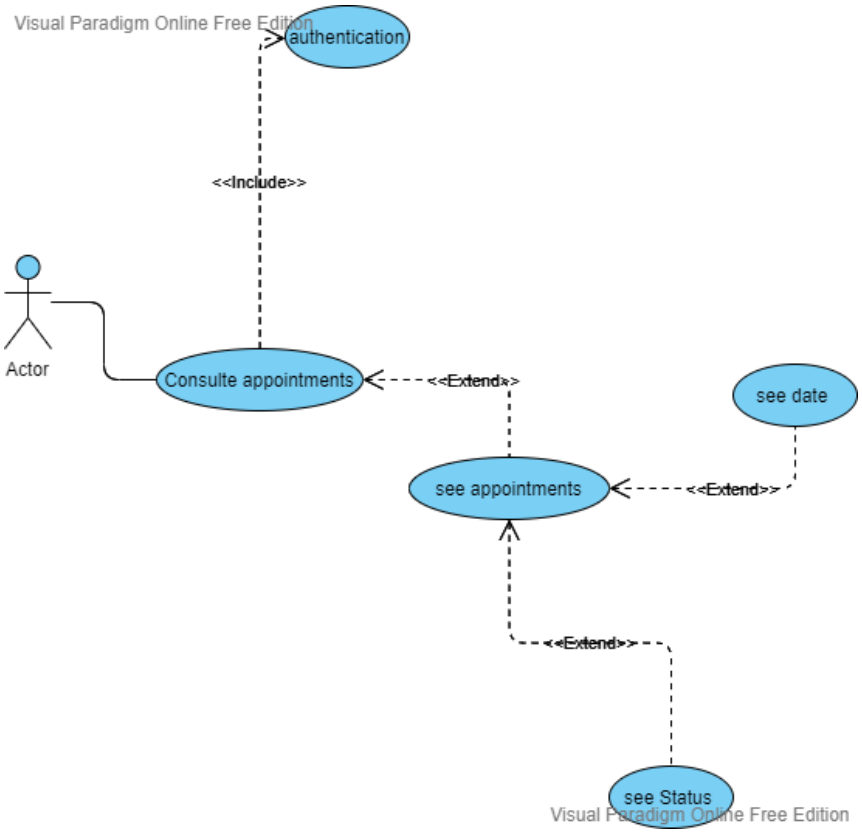


Figure 13:patient consulting appointments use case diagram

Use case	Consulting appointments
Main actor	Patient
Pre conditions	• Patient signed in
Post Conditions	• Appointments viewed
Nominal scenario	16. The patient ask for the appointments interface 17. The system show the interface. 18. The patient access the appointments . 19. The patient consultee his appointments 20. The patient exit
Alternative situations	**A1** : if the patient is not authenticated : d. The system block the actions and logout **A2** : if the connexion with the database is not set : a. exit the operations and wait for the connexion to resset

2.1.1.5. Generating QR use case diagram

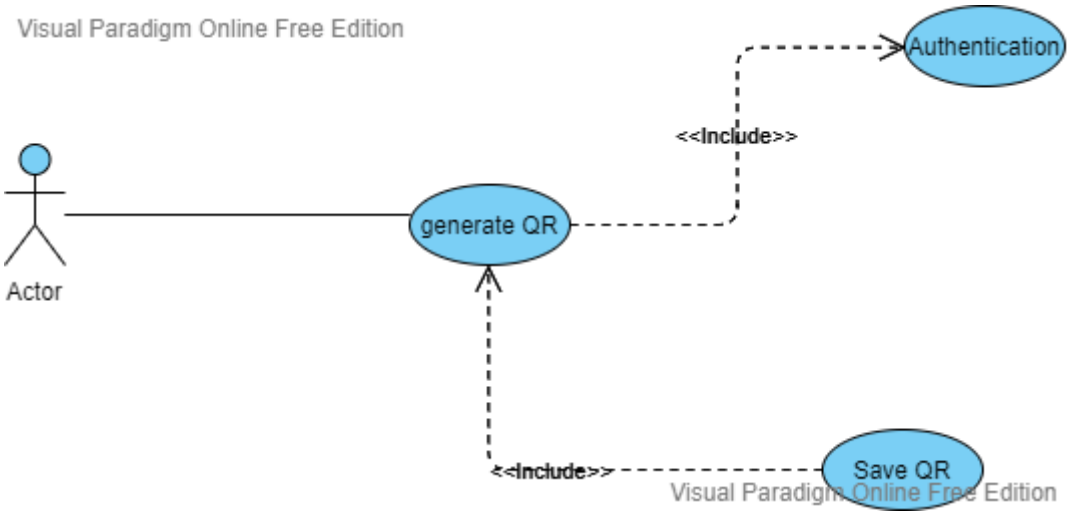


Figure 14:patient QR generating use case diagram

Use case	Generate QR
Main actor	Patient
Pre conditions	• Patient signed in
Post Conditions	• QR gene
Nominal scenario	21. The patient ask for the QR interface 22. The system show the interface. 23. The patient access the QR . 24. The patient Generate a QR code. 25. The QR is added to the database 26. The patient exit
Alternative situations	**A1** : if the patient is not authenticated : e. The system block the actions and logout **A2** : if the connexion with the database is not set : a. exit the operations and wait for the connexion to resset

2.1.2. Admin use case diagram

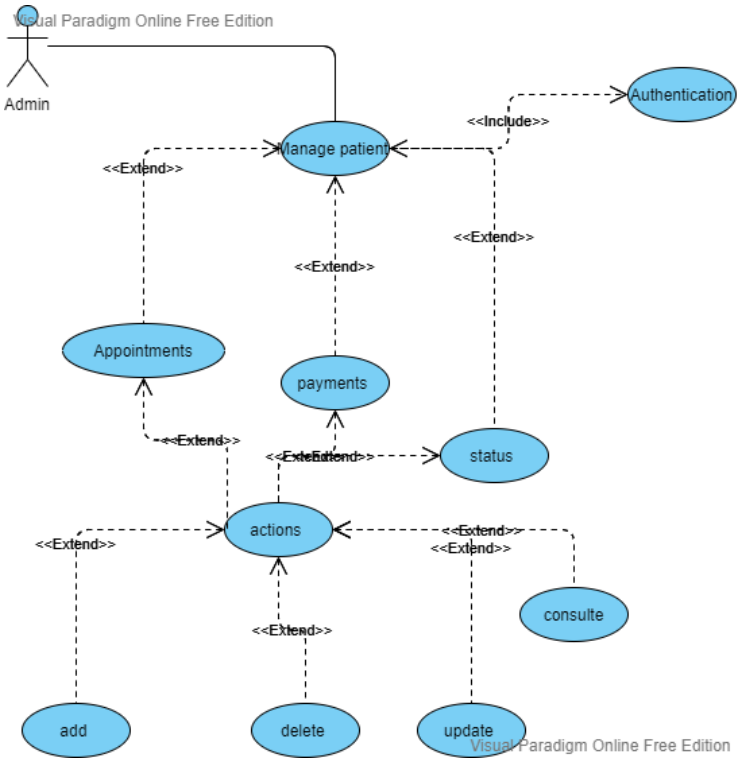


Figure 15: admin use case diagram

Use case	Manage patients
Main actor	Admin
Pre conditions	• Admin signed in
Post Conditions	• Patient managed
Nominal scenario	27. The admin ask for the interface 28. The system show the interface. 29. The admin the patient/appointments/payments details . 30. The admin modify the patient status . 31. The modification is added to the database 32. The admin exit
Alternative situations	**A1** : if the patient is not authenticated : f. The system block the actions and logout **A2** : if the connexion with the database is not set : a. exit the operations and wait for the connexion to resset

3.Class diagram:

In software engineering, a class diagram in the Unified Modelling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships

3.1.General Class Diagram:

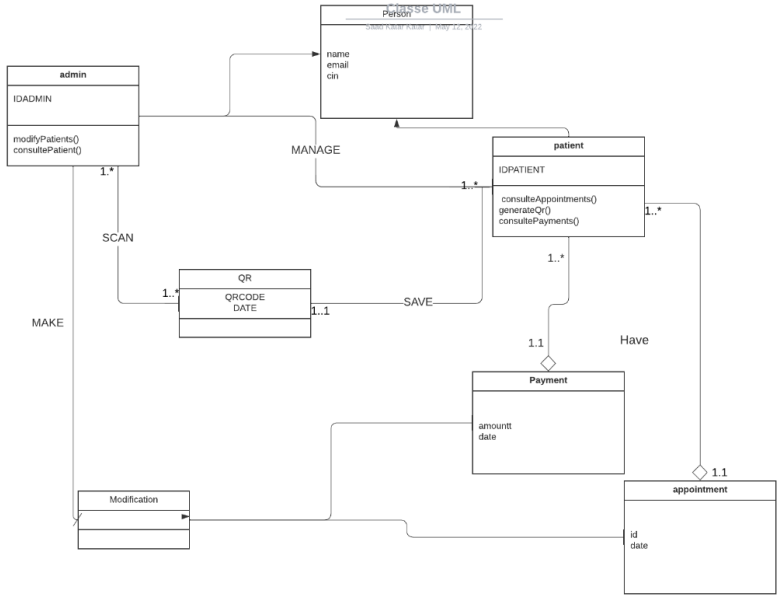


Figure 16:General Class diagram

CONCLUSION

In this chapter we did define design patterns used in this project,

also we applied conception methods and tools such as use case diagram and class diagram to better understand and prepare for the realisation of the application.

In the next chapter we can present the graphical Ui design used in the application.

CHAPTER 3:

GRAPHICAL INTERFACES

UI Design:

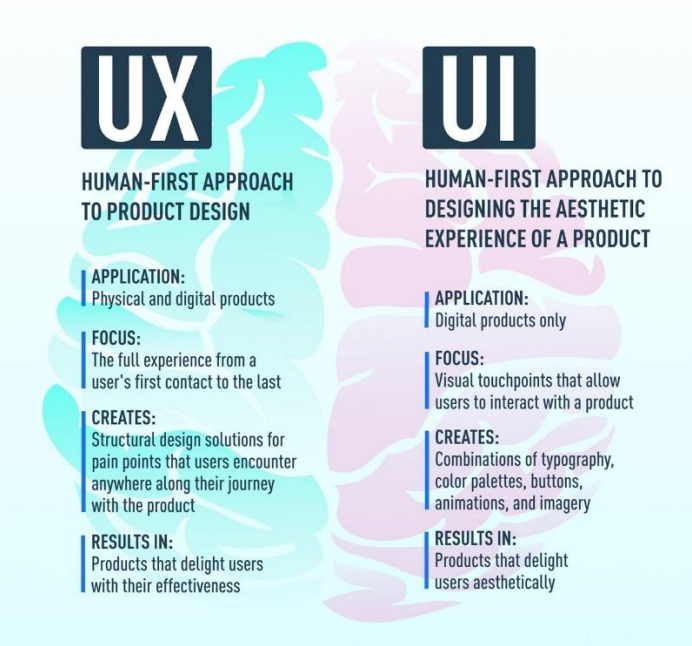


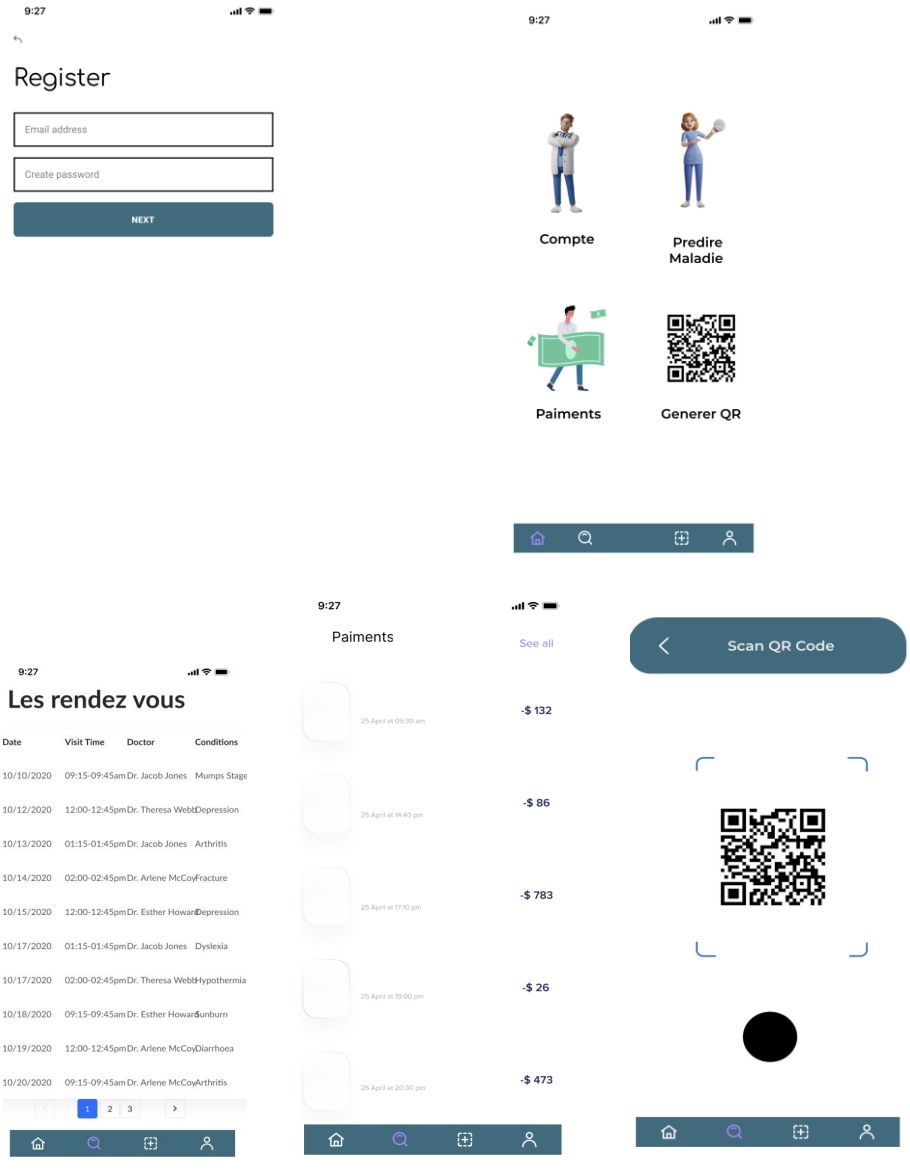
Figure 17:UI/UX deference

User interface **(UI)** design or user interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing usability and the user experience. In computer or software design, user interface (UI) design is the process of building interfaces that are aesthetically pleasing. Designers aim to build interfaces that are easy and pleasant to use. UI design refers to graphical user interfaces and other forms of interface design. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals

UX Design

Essentially, UX applies to anything that can be experienced—be it a website, a coffee machine, or a visit to the supermarket. The “user experience” part refers to the interaction between the user and a product or service. User experience *design*, then, considers all the different elements that shape this experience.

Graphical Interfaces Used in the design:



1.1 Main Home Interface

This interface is responsible for the access to the totality of the functionality describing by the application.

The goal of this interface is to allow the patient to have a better experience with a simple access to the other interfaces and the goal is to void complex UX design and not to overwhelm the patient with unnecessary functionality .

1.2 QR generator :

This interface helps the patient to generate a QR code that will be added to the database so it can be scanned later on via a QR scanning machine in the get of the outpatient consultation

1.3. Appointments interface:

This interface is responsible for displaying the appointment information and details for the patient ,that helps him a better understanding of his healthcare status

1.4. Payments interface :

The patient can track his payments thanks to the payment interface, the admin must update the patient appointments and payments

Summary :

The end of study project presented in this report helped me a lot to obtain new skills and trained me practicing projects solving real-life situations , I learned a lot throughout this project especially how I need to better manage my time .

I hope that the public healthcare institutions develop new methods and strategies to better serve patients , that’s why I feel proud being a part of this development.