People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University of August 20, 1955 Skikda

Faculty of Sciences

Department of Computer Science

Master thesis

For obtaining the diploma of Master degree in Computer Science

Option: Computer Systems

Subject

The improvement of the interactivity of the online appointment application Mouaidy www.rdv-doctor.net

Prepared and Presented by:

- Berdoudi Lina Malak Ellah.
- Boudebza Nour El-Houda.

Supervised by:

Dr. Boulehouache Soufiane.

Session: 2021 / 2022

Dedication

This simple dissertation thesis is dedicated to:

My dear father Mohamed Mustapha "Rahimahu Allah".
To my mother, the dearest one in the world, the candle that alwaysguided me on the right path, the light of my life, who did everything to my success and happiness, for her encouraging words, supportingmeand believing in me, and for her prayers throughout my studies - my God protect her-.

To my sisters Maissa and Asma for their help, courage and motivation throughout my university career and for being a part of my life.

To my brothers Ramzi and Oussama.

To my friend Nour for the friendship we shared, for every single moment and foreverything.

To my best friends Oudjdane and Samah who have contributed to this work from nearand far for their encouragement.

Lina Malak Ellah

Dedication

This simple dissertation thesis is dedicated to:

My parents, thank you, mom, thank you, dad. Three little words to which so much can be added.

Thank you, mom, thank you dad for always being there listening, encouraging, and most of all believing in me, for there love, sacrifice and education, thanking them for me being where I am; A million words would be too short to express the love I have for you both. I can never thank them enough.

To my brother, for all he has done for me, for his help and motivation throughout my university career and for being a part of my life.

To my best friend Lina, thank you for your help, thank you for your generous gesture and unselfishness is a proof of your beautiful friendship.

To all my friends who have contributed to this work from near and far for their encouragement.

Nour El-Houda

Acknowledgments

First of all, we would like to thank ALLAH, the almighty for all the courage and the patience he gave us for the completion of this work.

Our sincere gratitude goes to our thesis supervisor

Dr. Boulehouache Soufiane, a doctor at the University of August 20, 1955 - Skikda, for having assumed the role of supervisor of this master thesis and for having guided our work, for the advice he always gave, for his scientific rigor throughout the project, for his patience, and for having always encouraged us.

We would like also to thank all those who helped us directly or indirectly to carry out this work.

Abstract

Today, people prefer to accomplish services online by a few clicks from the comfort of their places. However, when it comes to booking the next visit to the doctor, it can become a real challenge. Especially during the peak time of COVID-19. As there are not really potentials technologies of Web tools, most of the available services schedule the appointments in old-fashioned ways, and even if some of them use online booking tools, after trying it, you would prefer to make a phone call rather than using them, especially for the semi-literate users. So there is a need to develop a simple, effective, and easy to learn User Interface (UI) of the online appointment system for a productive doctor-patient consultation.

This work concerns the improvement of the user experience of an online appointment web application, where all categories of patients especially semi-literate can choose their doctors who are located anywhere in Algeria, and book their appointments. The main goal of this work is to improve the interaction and effectiveness of the online appointments application Mouaidy, for all users, through user-centric design.

Keywords: Online Appointment System, Online Schedule Design, User Interface Design (UID), Front End.

Résumé

Aujourd'hui, les gens préfèrent accomplir des services en ligne en quelques clics dans le confort de leur lieu. Cependant, lorsqu'il s'agit de réserver la prochaine visite chez le médecin, cela peut devenir un véritable défi. Surtout pendant la période de pointe de COVID-19. Comme il n'existe pas vraiment des potentielles technologies des outils du Web, la plupart des services disponibles planifient les rendez-vous à l'ancienne, et même si certains d'entre eux utilisent des outils de réservation en ligne, après l'avoir essayé, vous préférerez passer un appel téléphonique plutôt que de les utiliser, surtout les utilisateurs semi-alphabètes. Il est donc nécessaire de développer une Interface Utilisateur (UI) simple, efficace et facile à apprendre du système de rendez-vous en ligne pour une consultation médecin-patient productive.

Ce travail concerne l'amélioration de l'expérience utilisateur d'une application web de rendez-vous en ligne où toutes les catégories de patients notamment semi-alphabètes peuvent choisir leurs médecins qui se trouvent n'importe où en Algérie, et prendre leurs rendez-vous. L'objectif principal de ce travail est d'améliorer l'interaction et l'efficacité de l'application de rendez-vous en ligne Mouaidy, pour tous les utilisateurs, grâce à une conception centrée sur l'utilisateur.

Mots-clés : Système de rendez-vous en ligne, Conception d'horaires en ligne, Conception D'interface Utilisateur (UID), Front End.

مـــلخص

اليوم، يفضل الناس إنجاز الخدمات عبر الإنترنت ببضع نقرات من أماكنهم المريحة. ومع ذلك، عندما يتعلق الأمر بحجز الزيارة التالية للطبيب، فقد يصبح ذلك تحديًا حقيقيًا. خاصة خلال وقت الذروة لـ COVID-19. نظرًا لأنه لا توجد بالفعل إمكانيات تكنولوجية لأدوات الويب، معظم الخدمات المتاحة تقوم بجدولة المواعيد بالطرق القديمة، وحتى إذا كان بعضها يستخدم أدوات الحجز عبر الإنترنت، بعد تجربتها، تفضل إجراء مكالمة هاتفية بدلاً من استخدامها، خاصة بالنسبة للمستخدمين شبه الأميين. لذلك هناك حاجة لتطوير واجهة مستخدم (UI) بسيطة وفعالة وسهلة التعلم لنظام المواعيد عبر الإنترنت للحصول على استشارة مثمرة بين الطبيب والمريض.

يتعلق هذا العمل بتحسين تجربة المستخدم لتطبيق الويب للمواعيد عبر الإنترنت، حيث يمكن لجميع فئات المرضى وخاصة شبه الأميين اختيار أطبائهم الموجودين في أي مكان في الجزائر وحجز مواعيدهم. الهدف الرئيسي من هذا العمل هو تحسين تفاعل وفعالية تطبيق المواعيد عبر الإنترنت موعدي لجميع المستخدمين، من خلال تصميم يركز على المستخدم.

الكلمات المفتاحية :نظام المواعيد عبر الإنترنت، تصميم جدول الطبيب عبر الإنترنت، الواجهة الأمامية ، تصميم واجهة المستخدم.

.

Contents

List of Figures

Chapter I

Chapter II

Chapter III

Chapter IV

List of Tables

General Introduction

1. Introduction

In this modern world, technology has become a replacement of some manually works and a necessary tool for solving problem, as it arises, everything has been easy for us. That is why in medical field "Online Appointment System" is very significant nowadays. Doctors are life savers; they should not be too far from those who need healthcare. Doctors should make services accessible and available to their patients, as more and more people is having the need on medical attention.

Before the last few years, doctor appointments were usually taken on the phone calls or by visiting the hospitals in person. This process needed the involvement of individuals so the ability to take appointment was restricted to the availability of schedulers, phone lines or the physical presence of a person. With the growth of time, everybody demanded timeless and efficient medical care delivery because manual appointments (that requires the physical presence of both individuals) and long waiting lines have formed an irritating situation for the healthcare institutions especially during COVID-19. So, it created a need for such an integrated health care system that could deliver seamless care to patients. The emergence of online appointment system offered timeless and efficient access to health care services. Therefore, for hospitals and other medical societies, online appointment booking has a great importance and a subject of interest (Koole, 2007) [1].

Booking appointment online has become a new trend in the past few years and is considered as one of the key processes in the healthcare industry. Bailey (1952) considered scheduling system as a trade-off or a compromise between a doctor and patient's waiting times [2]. Patients who get late for the appointments or who fails to come becomes the reason for the underutilization of a doctor's time. Idle time and underutilization of doctor's time are also resulted by gaps in the appointment times (Bailey, 1954) [3]. Different researchers agreed that main patient dissatisfaction is caused by long waiting times. Cayirli (2004) defined access time as the time between patients' request for the appointment and the time he is checked up [4]. According to Veral, waiting time is the time between consultation and the scheduled time while neglecting the early arrival of a patient (Veral, 2006) [5]. Different researchers defined waiting /access time in different ways. A well-designed appointment system supposed to improve patients' satisfaction by reducing cost and time of clinics and hospitals especially in the busy lives we are leading today.

With the growing population need for more efficient ways to access a medical treatment is also growing. Through an online appointment system, a user gets access to the doctor's online webpage and can make an appointment with online software. Patient can also provide additional medical history in advance, giving adequate time to the doctor to prepare the necessary information for consultation. In this way, online appointment systems are helping doctors and patients. Nowadays, there are many kinds of online appointment tools available in the markets which are easy to set up and not too much expensive. Online appointment system offers value-added services and lots of benefits to the doctors and patients. It makes the patient appreciated by eliminating the hassle of long waiting times. Online appointment systems are also getting popular because of its low-cost availability, that why the application web of this system, has to be simple and easy to use especially for those semi-literate users.

2. Problem statement

When patients or even doctors use most of online appointment applications, which schedule the appointments in old fashioned ways such as the Mouaidy application, they would prefer to make a phone call rather than using them especially for the semiliterate users.

3. Objectives

The goal of this work is to facilitate, improve the interaction and the effectiveness of the online appointments application Mouaidy to be effectively used in Algeria. This application provides patients an easy way of booking online doctor’s appointment. As well as, it provides doctors or their assistants to manage their schedule and patient’s appointments.

This work proposes the use of HTML, CSS, Bootstrap, JQuery and JavaScript at the client side while PHP and MySQL on the server side.

4. Manuscript structure

Besides this introduction and the general conclusion, this dissertation is divided into four main chapters.

- In the first chapter, we aim to present a set of basic concepts of Online Appointment System. We introduce definitions, role, advantages and limitations of Online

Appointment System. Following this, we will present the existing system in Algeria, and given examples of online appointment platforms.

- The second chapter is dedicated to present the Front End techniques, which we will use to improve the interaction of our web application.
- In the third chapter, we will present the design or the conception of our web application with: the different types of UML diagrams and the Website UX flowcharts for the main actors of our application.
- In the Fourth and last chapter, we will present our database and the implementation environment, following this we will use the lessons learned from the techniques of Front End, to improve the interaction and will be devoted to the implementation of the proposed online appointment application.

Chapter I: Online Appointment Systems

1.1 Introduction

The need for healthcare services is growing with the increase in population and the number of patients who seek health care at hospitals, medical facilities, holistic groups, and physicians practice has improved significantly. And according to the development that the web and phone applications knows in last year's most of patients now prefer to use online appointment systems to have the best healthcare services. The online appointment system, present to patients the capability of access to the doctor's online webpage and book their appointments and also doctors can organize their schedule and do a lot of services. This system offers value-added services and lots of benefits to the doctors and patients.

1.2 Online appointment system

1.2.1 Definition

An online appointment system is a web-based system which is made up of independent components or web pages, working together for a common purpose. Such systems are available on the internet for the users to accomplish some purpose.

James (1999) defined internet as a system which provides an opportunity to millions of people to get access to a large amount of stored data and get connected to each other despite distances. Increased rate of missed appointments and patient dissatisfaction pushed to recognize the need for better-quality care services [6]. With the advancement of information technology and the sensitive situation of medical care, online appointment system emerged as an important output for efficient and timely delivery of health care services. Earlier, appointment requests were used to make through the phone call, email or fax but with the advancement of technology and internet, the trend changed towards the proper online appointment systems.

According to Gruca (2004), a patient can make an appointment for a doctor either by going directly to the doctor's clinic or making a manual appointment through the phone call or sending the email. With the emergence of the internet, health care centers can connect and communicate with their patients efficiently. Therefore, many healthcare centers have started using online appointment systems. Such systems ensure efficiency and effectiveness.

1.2.2 Role

Online Appointment Systems saves time as well as generates income too that facilitates consultations, different operations and builds a good relationship between doctors and their patients [7].

According to this important role in current users life studies show that more than 5 billion people worldwide are expected to maximize the usage of web and mobile applications this year. Statistical data from 2019 show constant growth of usage of online appointment system and the trend is no different in 2020 and 2021 (or even more due to novel coronavirus outbreak).

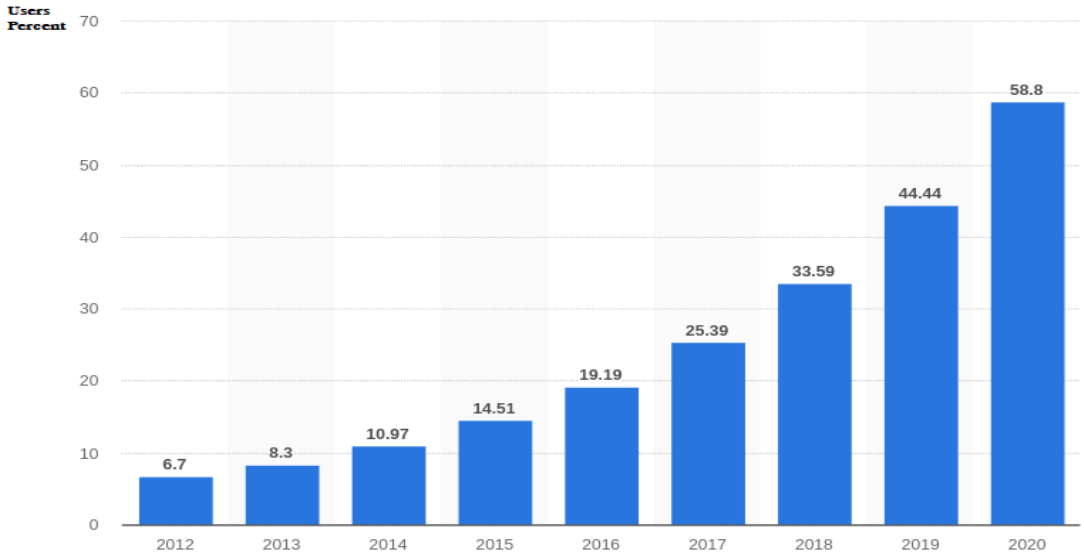


Figure 1: Growth of using online appointment systems during few last years.

1.2.3 Advantages and Limitations

Compared to traditional appointment methods, the current online appointment scheduling system has unique advantages and limitations.

a. Advantages

- **Patient Centeredness**

Patient-centeredness is one of the six quality aims proposed by the Institute of Medicine to improve health care quality [8]. Web-based medical scheduling as a medical self-service offers a more patient-centered means to make appointments [9]. Most Web based appointment systems are interfaced with a calendar-like list. Patients can browse and select the most convenient appointment time from the available time slots [10]. In contrast, patients are only given very limited options of available time slots in traditional appointment systems. Besides time slots, some of the Web-based systems allow patients to filter physicians by physicians' attributes such as education background, experience, gender, and reviews from other patients [11].

Another convenience from improved patient access is that patients can fill out registration forms, get prescreened and review practice policies online before they show up and this can smooth workflow and reduce misunderstandings.

In the self-servicing Web-based appointments, patients' own descriptions of the reason for visit are often more detailed and illuminating [12]. Sometimes, patients might be uncomfortable or unable to vocalize certain symptoms (eg, sexual health problems) to the scheduler over the phone or in person, and they may make an untrue statement [13]. They tend to be more candid when they schedule online by themselves [14]

- **Reduced No-Show Rates**

No-show is a significant cause of wasted clinical resources [15]. The patient-centered design in Web-based appointments has the potential to decrease no-show rates [11, 16]. The reasons for the reduction of no-shows after implementing Web-based scheduling have not been systematically studied in the literature, but it could be attributed to the improved access in Web-based scheduling that allows patients to easily verify, cancel, and reschedule their appointments [16]. A possible reason is that patients feel more responsible for their appointments when they make appointments by themselves [17].

- **Reduced Waiting Time**

Waiting is an indicator of service quality and a source of dissatisfaction that affects health care outcomes and patient retention [18, 19]. Long waiting time may make patients seek care from other providers and thus this can potentially cause a loss in revenue.

The most cited benefit of real-time scheduling is after-hour access [10, 20, 21]. Real-time scheduling requires minimal intervention of schedulers and thus can help reduce the waiting time caused by human factors. The available time slots are transparent to patients through the Web interface. Patients are free to claim available appointment slots anytime and anywhere [14, 22]. The support of same-day or soon appointments by some real-time systems can help further shorten the time between when the appointment is requested and when the medical service is fulfilled. Although there is a concern that the ability to book in advance for chronic conditions might be diminished by same-day appointments due to the limited number of appointment slots [23], same-day appointments could produce positive outcomes as long as the provider can find a balance in his or her capacity. For providers, it is possible to reuse the time slots released due to late cancellations. These allotted time slots will be otherwise wasted if traditional appointment methods are used because of the longer turnaround time.

b. Limitations

- **Internet Connection**

To book an appointment online, you definitely need internet connection; be that on your phone or using a computer. In the case of no internet, you will be unable to do this. Of course, in today's day and age, this doesn't seem like a major issue, since we have access to the internet almost at all times. However, if the connection is absent for some reason, you will run into this complication. This could be a bit more problematic if you lose access to the internet, for example, you won't be able to see your bookings and cancellations. Here, we suggest that when you choose an internet provider, make sure that it is a reliable one that will give you the least amount of connection issues [24].

- **High Demand**

When you provide the option for online appointment scheduling, you open your doors to new users (patients or doctors). The easiness of booking an appointment online makes your service accessible to almost anyone witch can create a large influx of demand for your online available services. The result of this increase in demand is a very large volume of bookings that need to be handled and may force you to expand your resources, upgrade your existing means, etc.

- **Uncertainty**

An online appointment scheduling system can create uncertainty and confusion for users. The extent of uncertainty definitely depends on the type of services you provide. Here is an example. If you are a doctor and you allow your patients to book appointments via an online scheduling platform, you will have given them the option to choose which service they want from you. However, some patients may not be able to identify exactly what they need from you, which could force them to book the wrong appointment type.

One way to avoid this is by providing a thorough enough explanation of exactly what you offer.

1.3 Existing System

In Algeria, there is a huge trend of private medical clinics and hospitals. Doctors run their own private clinics and consult patients during the evening or any time of the day depending on their availability. Some are popular and known to all while some are known by few people. This situation proves to be a challenge for new practitioners as they are known to very few people even if they have a good academic background. On the other side, patients also face difficulty in finding and choosing a nearby doctor. Unscheduled appointments, long waiting lines and

keeping medical files in physical form are also the common problems faced by the patients of Algeria.

1.3.1 Examples of Online Appointment Platforms

a. DZDOC

DZDOC is a multi-service medical platform created by Khidma Tech, a computer engineering Service Company that benefit from extensive know-how in the field of information technology.

In May 2015, DZDOC launched the first online appointment booking service in Algeria to offer Algerian doctors and patients a simple, efficient and comfortable solution for managing their medical appointments [25].

Our objective for our patients is to improve access to medical care for Algerian citizens throughout the national territory by setting up computer tools and online services that are free and accessible to the general public to preserve their health and that of his relatives. And also provide Algerian doctors and health professionals with a complete service that includes all the tools and computer technologies they need for better management of their daily work.

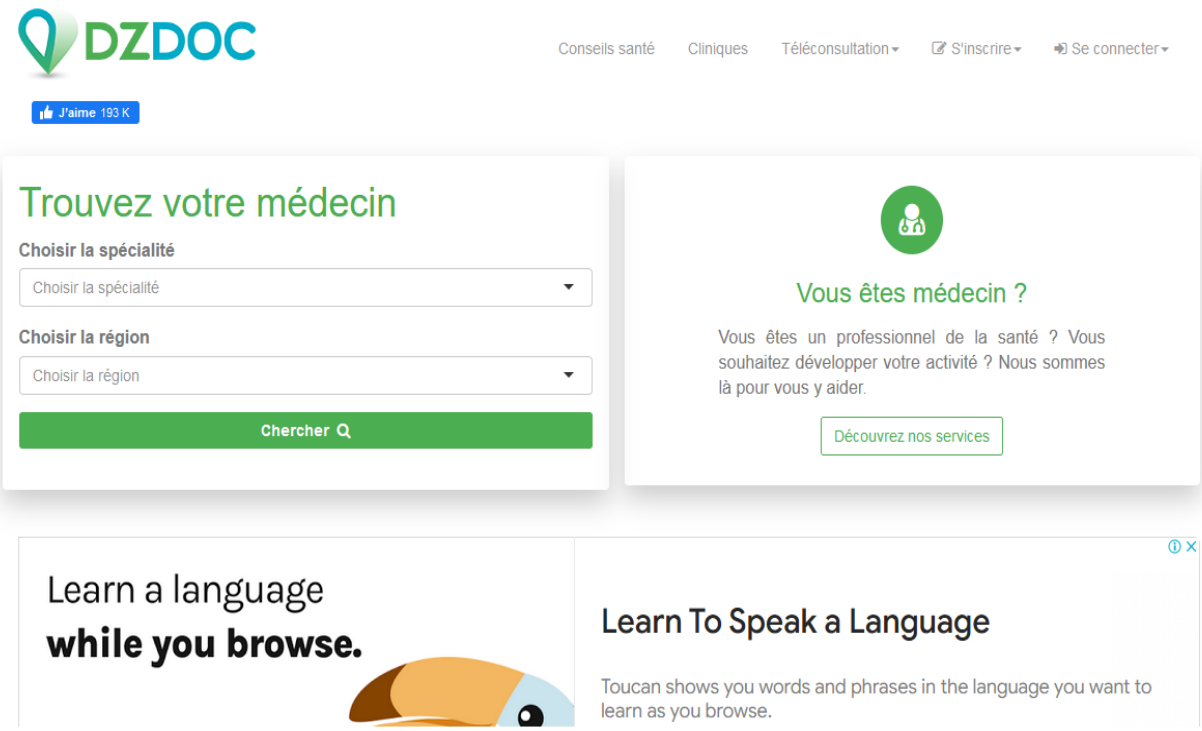


Figure 3: Home page of the online appointment platform DZDOC.

b. SihhaTech

SihhaTech is a Start-up of the innovative group AFIND (Algerian Foundation for Innovation and Development) and leader in the emerging market of online appointment booking in multi-channel and multi-terminal mode [26].

SihhaTech is a Web 2.0 service for making appointments online with a doctor or dentist through its portal https://sihhatech.com. The database is organized by Wilaya and/or medical specialty and includes the name and address of the health facility, as well as the exact location to access it. For patients, a simple click on one of the neighboring providers listed at the end of the chosen search criteria, the platform displays all the details: address, telephone number, opening days and hours and geolocation on a Google map. Since the doctor's agenda is linked to the SihhaTech database, the time slots already allocated are automatically taken into consideration. Once the appointment is fixed, the platform sends a reminder SMS to the patient 24 hours before the deadline. Practitioners, who subscribe to the platform benefit, at the same time, from better visibility of their range of services, while improving the proper management and updating of their appointments.

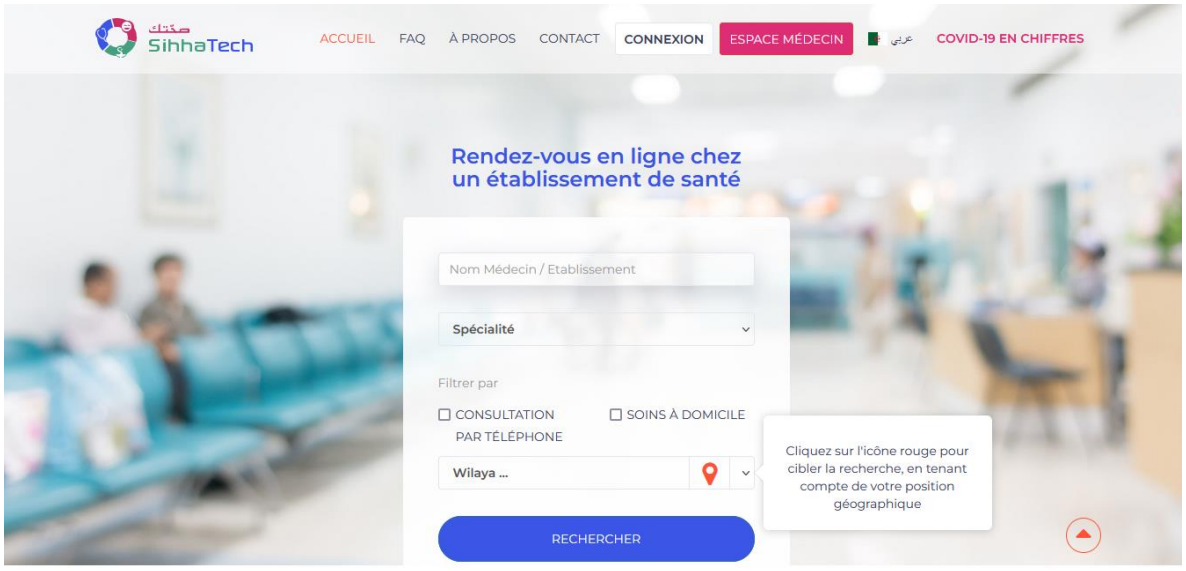


Figure 4: Home page of the online appointment platform SihhaTeck.

c. Bessiha

Beesiha is an online medical appointment service, founded by a group of young Algerian engineers and entrepreneurs, accompanied by health professionals. Completely free, the appointment service is available 24 hours a day, 7 days a week with the mobile application or the

website. For doctors, Beesiha offers an online agenda connected to their patients' appointments, in order to better manage consultations and their waiting room.

Our main objective is to improve the health experience, while making it more accessible for all users [27].

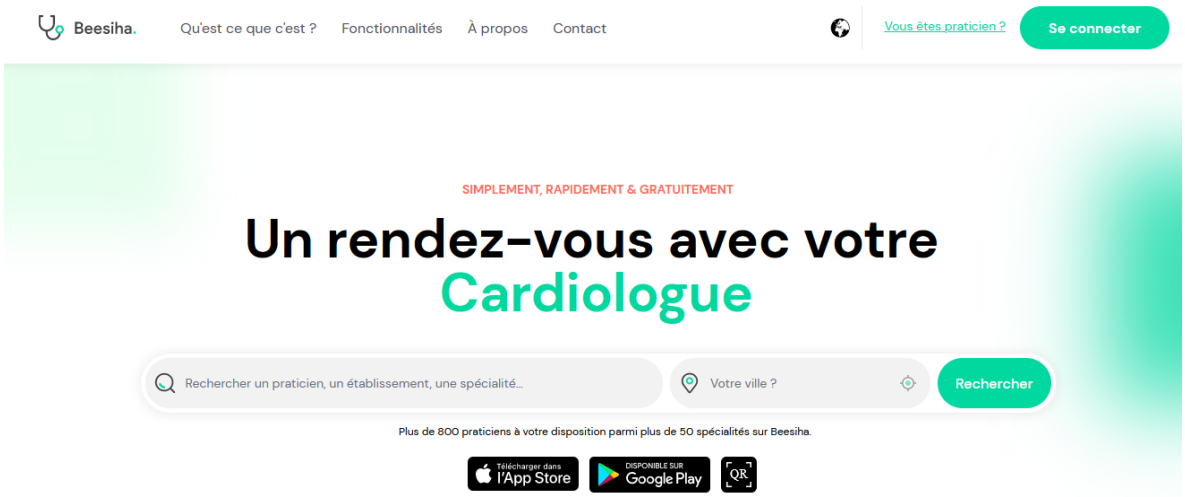


Figure 5: Home page of the online appointment platform Beesiha.

d. Mouaidy

Mouaidy is an online appointment web application, created by Mr. Boulehouache Soufiane, a doctor at the University of August 20, 1955 - Skikda, accompanied by health professionals for making appointments online with a doctor through its portal http://www.rdv-doctor.net/.

For doctors, Mouaidy offers an online agenda where they could manage their schedules and connected to their patients' appointments. It also allow to patients searching doctors by their name or their specialties and booking their appointments.

Strong points:

- It saves doctors and patients time.
- It can help doctors to see more patients.
- Patients can reach their doctors.
- Less paperwork.
- Totally free.

Weak points:

However, if we compare Mouaidy with the other online appointment platforms used in Algeria, it could have some of weak points like:

- Difficult to use due to the complicated interface.
- Need more time for performing user's tasks.
- Without guidance.
- Old fashion design and absence of attractive colors.

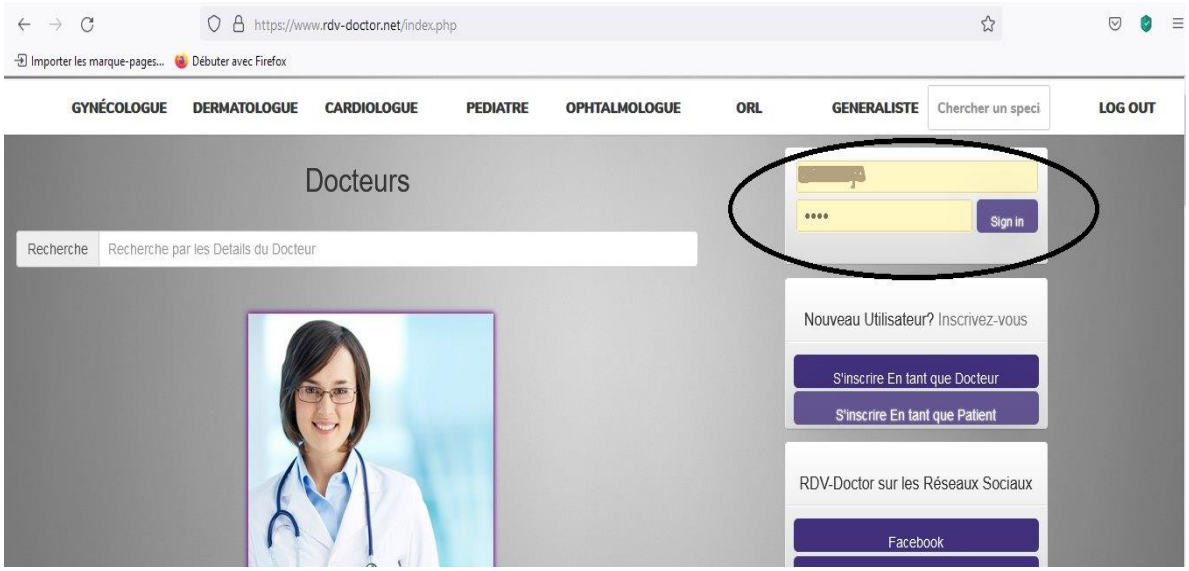


Figure 6: Home page of the online appointment application Mouaidy.

1.5 Conclusion

In this chapter, we presented the Online Appointment System, its definition, role, advantages and limitations and focusing on the existing system in Algeria. Then we gave examples of online appointment platforms.

We have chosen to improve the interactivity of the Mouaidy application because of its accessible source code.

The following chapter is dedicated to present the Front End techniques that we will use to improve the interaction and effectiveness of our web application.

Chapter II: Font End Techniques

2.1 Introduction

With the fast progress of technology from the birth of the internet, businesses become need to have a strong online presence in order to succeed, and having a good website or app is fundamental for this to happen.

When planning to build a new software product like a website, app, or even an internal tool for your own company the first thing you think about it is the front end of the system because is the part that will communicate with the users and reflects them the main goal of the software product [28, 29]. Here is where the Front End Design and Development process takes action and plays a vital role.

Which appealing user interface designs and developments can ensure better results, help you generate more traffic, and keep your visitors engaged and retained.

In this chapter, we aim to present a set of basic concepts of the Front End. We introduce the definition, role of the Front End and also the popular tools that used by designers and developers to create it. Following this, we present the stages of the most process used in the creation of front end and his advantages, limitations and also his various principals with number of example about this principals.

2.2 The Front End

2.2.1 Definition

The Front End of an application typically refers to the layer that represents the User Interface (UI). It is everything the user can see or interact with it on software products as fonts, colors, menus, buttons, tables, and many more, which is responsible for the total look and feel of a good user experience [30].

2.2.2 Role of Front End part

- **Communicates business intent**

  The Front End of the app should reflect the main goal of the company. The design, graphics, and content should work together to communicate with the users by providing the answer to their questions and giving them a good experience in the app that encourages them to use it [29].

- **Improves performance**

  If your loading speed of a web site is low, you can easily lose customers to other. The Front End maintenance requires site performances so that it could load faster.

- **Optimizes navigation**

The Front End helps the users to find what they are looking, for faster and easier on the site, by optimized navigation that makes the users see exactly what is the next step they need to do and how they do it. That happens with a clean, properly structured, and carefully planned interface.

2.2.3 Front End tools

a. Front End design tools

- Invision.
- Framer.
- Figma.
- Sketch.
- Adobe XD.

b. Front End development tools

- HTML (Hyper Text Markup Language).
- CSS (Cascading Styles Sheets).
- Javascript.
- JQUERY.
- BOOTSTRAP.

2.3 User Centered design (UCD)

2.3.1 Definition

The term of the User Centered Design was first introduced in 1977 by computer scientist Rob Kling. It gained further attention and popularity as it was featured in Don Norman's books User Centered System Design: New Perspectives on Human-Computer Interaction (1986) and The Design of Everyday Things (1988) [31].

User Centered Design (UCD) is a collection of processes that focus on putting users at the center of product design and development. This type of design hands-on approach that essentially comes under design thinking methodology has changed the way of thinking about humans and computers by putting users' needs, objectives, abilities and feedback on the front while planning, designing, and developing the Front End of any software product in the aim to make it more effective, efficient and safe and highly accessible for them [32, 33].

Its processes and strategies vary depending on the type of design system; however, they are usually based on a combination of research and user experience (UX) design activities. [34]

2.3.2 User Centered Design steps

Creating a Front End for a mobile or web application or any software product is not just only about coding, is a time- and money-consuming process, which takes efforts and human resources and goes through a few different stages, including the research about the users' needs and requirements to the creation of design that based about the (rough outlines of the user flow), and finally the development of front end and test it if good or not [35].

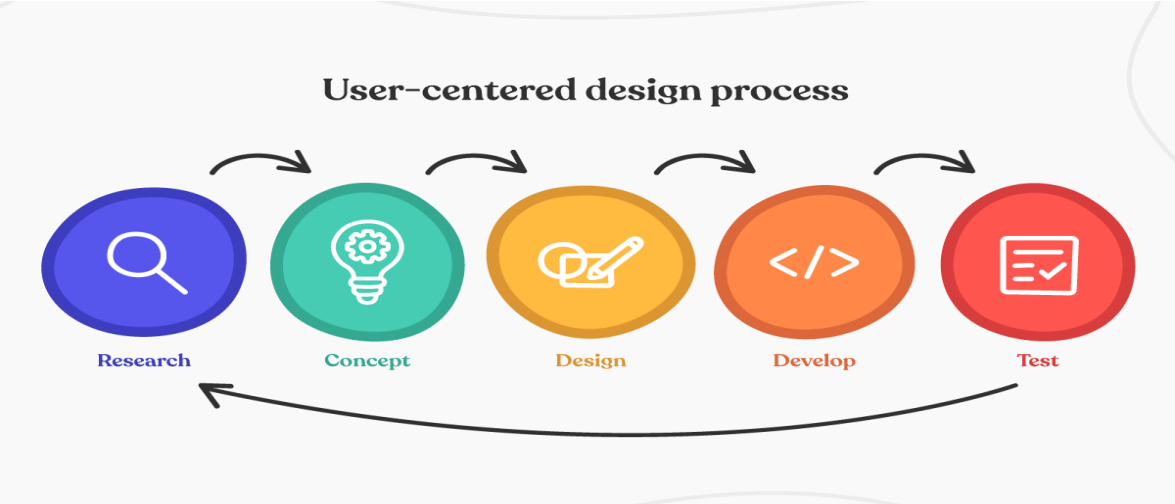


Figure 1: User Centered Design process steps.

a. Research: Understand the context of its use

The user research is the first and the main part in the process of User Centered Design because the information that collects and analyzed in this phase is critical to understand the problem and empathize with the user in the purpose to ensure that the front end of the system is usable and meets user needs [36, 37].

During this stage, we start with collect data by describing the expectations of the project and find out the type of user who will be using the system then understanding the context in which they may use it.

In other words, we must find the full answer to these principal questions:

- What the goal of the project is.
- Who the target audience is.
- How they will use the product.

This phase may entail ethnographic studies, user/buyer interviews, competitive analysis, and empathy maps for answer these questions.

b. Concept: Specify user requirements

This phase usually includes:

- **Creating user personas:** Personas are fictional characters that represent the different user types for your product. As you design your product, you can reference these personas as realistic representations of your target audience.
- **Creating Scenarios:** A scenario can be thought of as a sequence of events that occur with the product's user. The 'persona' created after market research should be used as the main character of this story. Scenarios should be designed to learn about the different outcomes of an event: there can be the best-case scenario, where everything works out ideally for the main character, the worst-case scenario, where nothing works out, and an average-case scenario, which is a regular day in the life of the character, where nothing special occurs.
- **Creating Use-Cases:** A use case describes the interaction between the persona and the rest of the world. It represents an event that may consist of intricate interactions between the character and the world. It is depicted by a series of simple steps for the character to achieve a goal in a cause and effect pattern.

c. Design

Once we have collected all the information about the system, and become the users' wants, needs, and expectations from a system are clear, we are ready now to start design the final look of the system user interface. This phase usually includes various activities them:

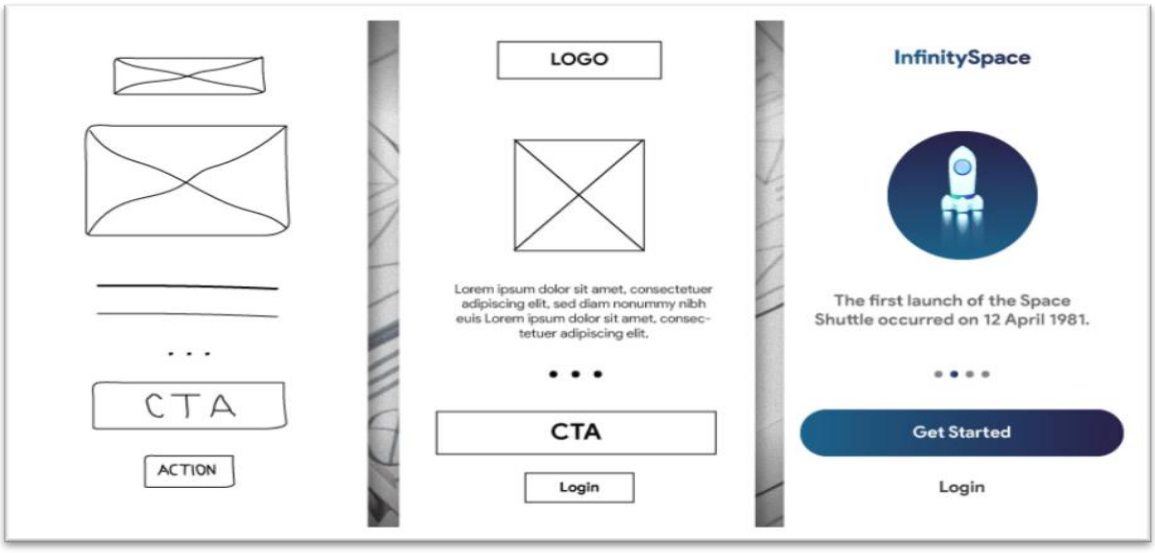


Figure 2: Sketch, wireframe, prototype of Infinity Space application.

- **Sketching:** It is the easiest and fastest way to visualize our ideas. You can do this by drawing by hand on a piece of paper, on a whiteboard, or in a digital tool. It is very useful during brainstorming sessions because it can help the team visualize a broad range of design solutions before deciding which one to go with [38].
- **Creating wireframes.** A wireframe is a tool that helps designers visualize the basic structure of a future page, including the key elements and how they fit together. Wire framing acts as the backbone of the product, and designers often use them as a foundation for mockups and prototypes.
- **Creating prototypes.** While wireframes are mostly about structure and visual hierarchy (the look), prototypes are about the actual interaction experience (the look and feel). A prototype is like a simulation of the product and may be low-fidelity (clickable wireframes) to high-fidelity (coded prototypes).

d. Development

At the development stage, the front-end developer takes the design layout of the user interface and converts them into CSS, JavaScript (JS), or HTML code. All the requirements from the previous phases are transformed into the actual system. Developers define the most relevant front-end frameworks, tools, and the best development practices to ensure that the product is developed in the most efficient way. They purchase and install the respective software and hardware and create the actual code on the basis of given specifications [38].

e. Validation (Testing)

Validation is an essential step in the design process because it helps teams understand whether their design works for their users. Usually, the validation phase starts after the high-fidelity design is ready, since testing with high-fidelity designs provides more valuable feedback from end-users). During a series of user testing sessions, the team validates the product with both stakeholders and end-users [39, 40].

The validation phase of the UX process may include the following activities:

- **Testing sessions.** User testing sessions with people who represent your target audience are very important. There are many different formats to try, including moderated/unmoderated usability testing, focus groups, beta testing, and A/B testing.

- **Surveys.** Surveys are a great tool for capturing both quantitative and qualitative information from real-world users. UX designers can add open-ended questions like "What part of the product you dislike?" to get user opinions on specific features.
- **Analytics.** Quantitative data (clicks, navigation time, search queries, etc.) from an analytics tool can be very helpful to uncover how users interact with your product.

2.3.3 User Centered Design methods

Now that we have explored some key principles behind User Centered Design, let is explore some of the research methods that any user-centric designer should familiarize themselves with. With understanding the user being a fundamental part of User Centered Design, knowing which research method to use and how to employ it effectively is crucial [41, 42].

✓ **Focus Groups**

Running a focus group involves inviting a group of your intended users to collectively share their thoughts and opinions about a product, a user path or simply particular issues you will be solving with your product.

Some characteristics of focus groups:

- A good way to get multiple perspectives at once.
- Great for defining product use cases.
- Requires an experienced moderator.
- Data is largely qualitative.
- Generally a small sample size.
- Relatively low costs, especially when done remotely.

✓ **Questionnaires & surveys**

Well-designed questionnaires and surveys can help obtain a large amount of statistical data regarding specific challenges or needs your users face.

Some characteristics of questionnaires and surveys:

- Feedback is generally brief and simple.
- Care needs to be taken to design effective questions that are unbiased.
- Data can be both qualitative and quantitative.
- Allows for a larger sample size.
- Relatively low costs of respondents are sourced from your user base.

✓ **Interviews**

Interviews are especially effective in the early stages of the design process. The open format allows you to dig for detailed insights that may be overlooked in other research methods.

Some characteristics of user interviews:

- Good for gathering in-depth information regarding individual needs and behaviors.
- Requires an experienced interviewer and detailed analysis of answers.
- Data is mostly qualitative.
- Small sample size.
- Time consuming and therefore high opportunity cost.

✓ **Usability Testing**

In usability testing, users interact with the product directly while a moderator takes notes and records feedback. This used to be done mainly in a live environment, but can also be done asymmetrically with certain tools. It is an excellent for of ethnographic research and a good way to uncover bugs and other issues.

Some characteristics of usability testing:

- Used to generate feedback on designs and user interaction.
- Requires at least a developed prototype to test.
- Data can be qualitative and quantitative.
- Small to medium sample size.
- High cost when done live, cheaper when done with tools like Hotjar.

✓ **Card Sorting**

Card sorting is a UX method used primarily to test and design the overall architecture of a website or application. It generally involves asking users to organize content, pages and topics into categories that make sense to them and possibly helping you label these categories. The key is to gain unique insight into how users think about the features and content on your website or app so that you can build a user-friendly architecture.

Some characteristics of card sorting:

- Important for making decisions on architecture.
- Data is quantitative.
- Usually a medium sample size.

- Relatively time-consuming.

✓ **Participatory Design**

Participatory design sessions are a great way to involve users in the generation of different product prototypes directly. However, they are best used when combined with other design methods, not as standalone process.

- Used to involve all shareholders (employees, users, designers, partners, etc.) in the design process.
- It can generate valuable insights but can be complicated to carry out effectively.
- Data is qualitative.
- Usually a small sample size.
- Cost and time varies.

There is no single research method that is best suited for User Centered Design. It is important to have a firm grasp of all the different methods out there so that you can adapt your research to your design process and easily access the information you need to be successful.

2.3.4 Notable Examples of User Centered Design

Due to bad usability, roughly 70% of online businesses fail. In fact, 88% of people indicate that they would not return to a site following a bad user experience. For this reason, many companies have begun implementing UCD into their designing processes [44].
Below are a variety of examples that have demonstrated great use of the key principles of UCD.

- **Duolingo**

  This language learning app takes simplification to a new level. Its simplistic interface allows users to learn a language easily through the process of completing tasks.
  Duolingo turns language learning into a game by letting users advance to new categories after completing a section. This process helps keep the user engaged while meeting their goal of learning a new language.

Figure 3: Duolingo allows you to learn language through play.

- **Apple**

The design process of a website and mobile app should be focused on making information quickly accessible. There is no better example of this principle than Apple's web pages.

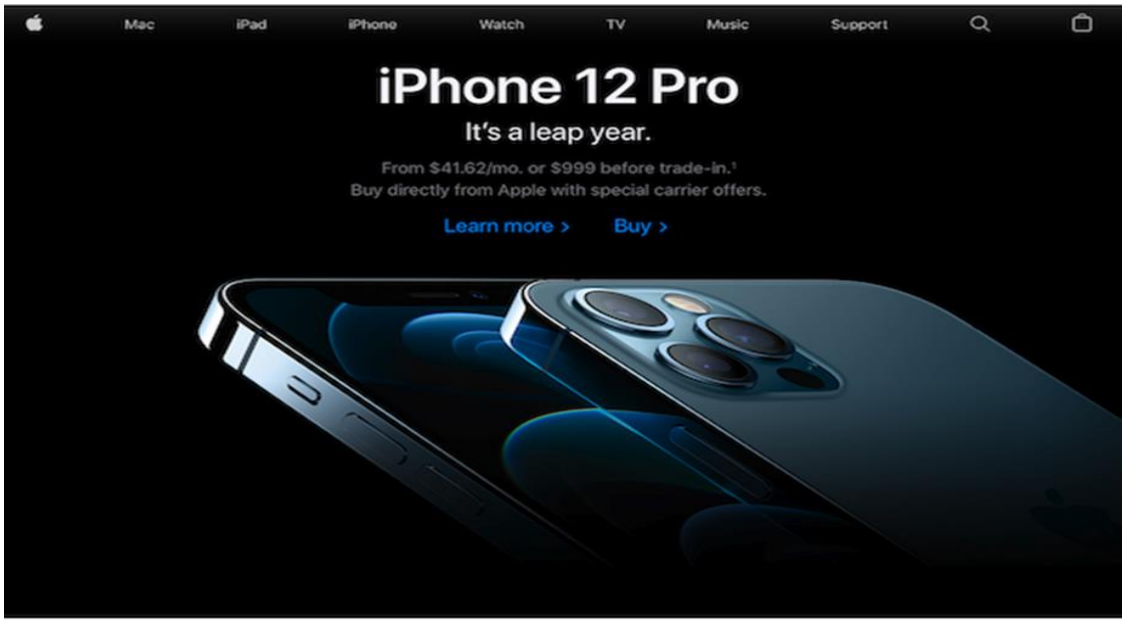


Figure 4: Apple website.

- **Rover**

This company knows how to value the trust of its users- especially when it comes to their pets. Rover allows users to book a sitter and review online photo updates during their absence.

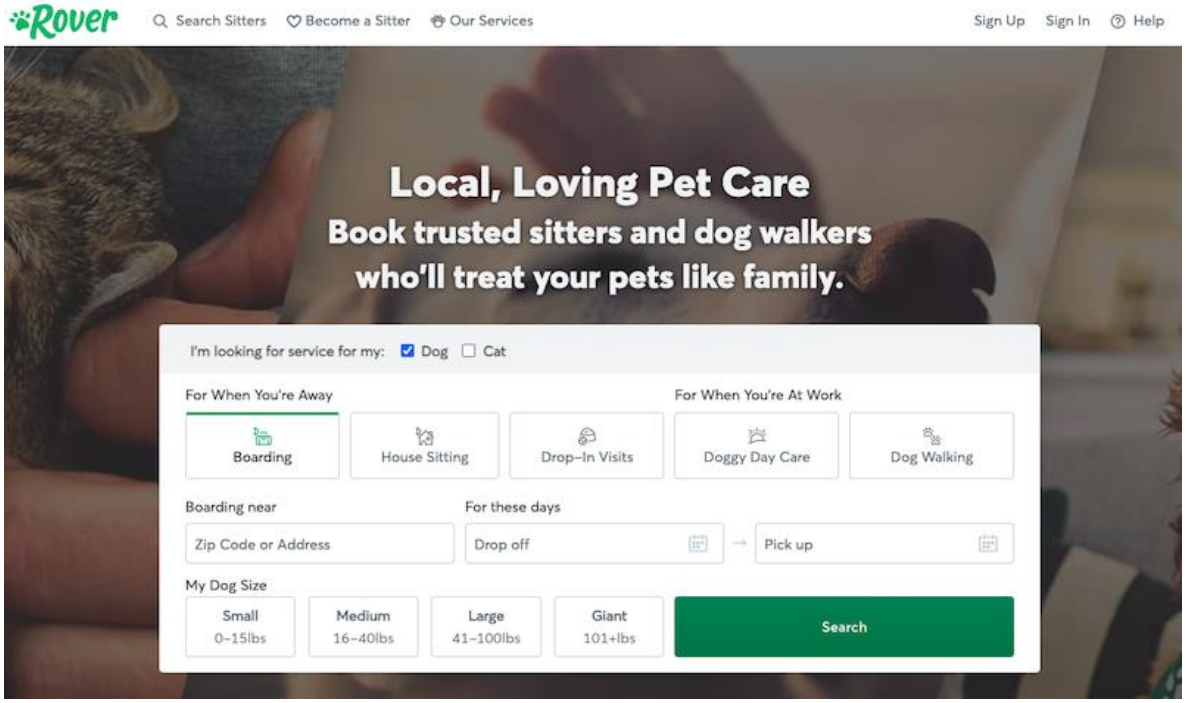


Figure 5: Rover website.

This added detail helps them appeal to their users by showing them that they understand how hard it is to leave your pet with someone else. Through this UCD, users are more inclined to view Rover as a trusting site and one they would like to invest in.

2.3.5 Benefits of User Centered Design

The illustrious designer Frank Chimero, who has worked with companies like Nike, famously said "People ignore design that ignores people".

Figure 6: Benefits of UCD.

- **Increase sales**: UCD is one such method where you can involve your potential user at every phase of the design process which helps you to give a better understanding of requirements and expectations. Once you achieve those requirements and fulfill expectations you will deliver a better product which ultimately results in more and more sales.
- **Reduce chances of human error**: As the title suggests, when you involve your user in the iterative design process it will automatically help you reduce human error and allow you to create a trouble-free product.
- **Leads to more inclusive products**: It helps you to leave all the biases out of the product and focus on all different types of users whether in terms of age, culture, occupations and so on.
- **Stronger sense of empathy**: Because the design team is in closer contact with users, they develop not only a deeper understanding but also a stronger sense of empathy towards them.
- **User centered design help designers**: You spend less time prototyping when you know what targeted users want from design. UCD helps designers to understand user's pain points and bespoke requirements just by testing and measuring key demographics. Also, by getting instant feedback.

2.4 Front End Design principles

The user interface is a critical part of any software product. When it is done well, users do not even notice it. When it is done poorly, users can not get past it to efficiently use a product. To increase the chances of success when creating user interfaces, there is stack of Interface design principles represent high-level concepts that is used to guide software design and developments.

2.4.1 Place Users at the Center

"Obsess over customers: when given the choice between obsessing over competitors or customers, always obsess over customers. Start with customers and work backward." – Jeff Bezos

Your user's goals are your goals, so learn them. Restate them, repeat them. Then, learn about your user's skills and experience, and what they need. Find out what interfaces they like and sit down and watch how they *use* them. Do not get carried away trying to keep up with the competition by mimicking trendy design styles or adding new features. By focusing on your user first, you will be able to create an interface that lets them achieve their goals [38].

2.4.2 Visual hierarchy

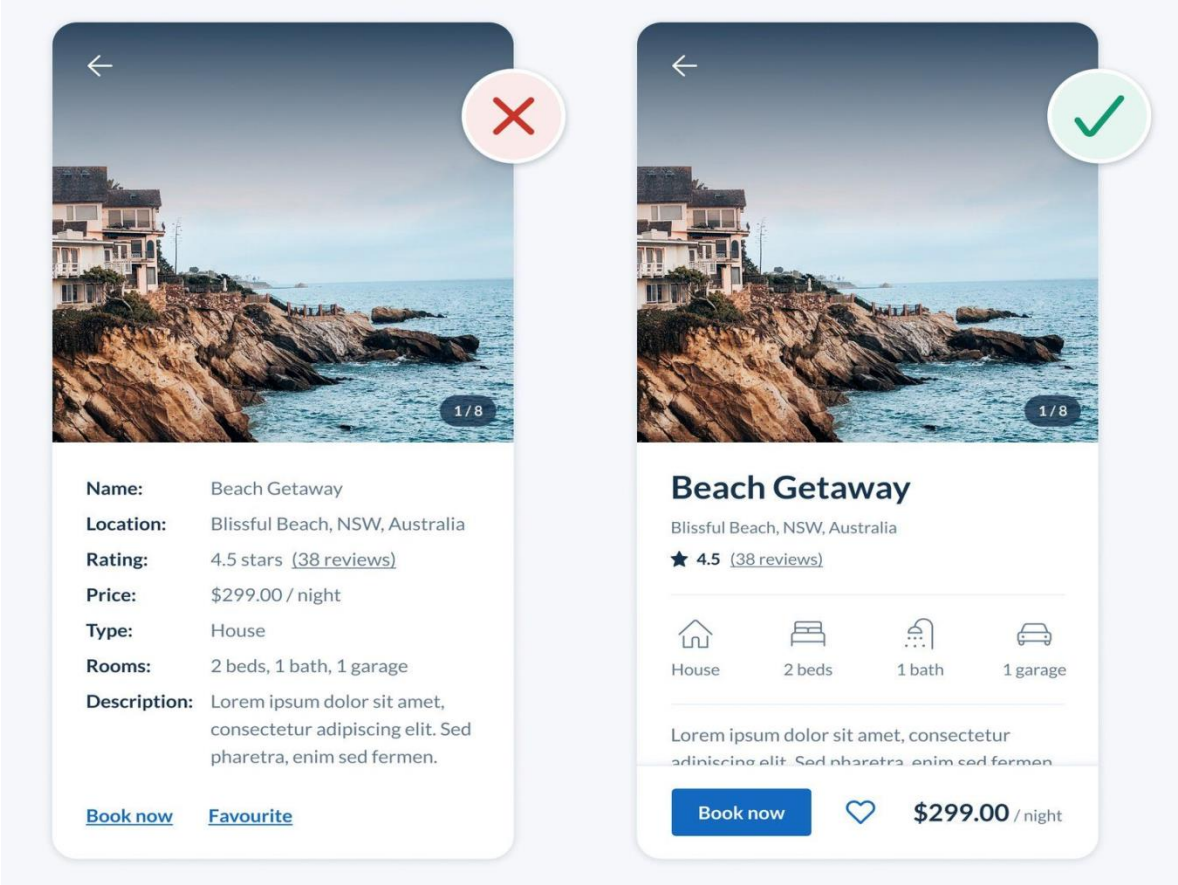


Figure 7: Good visual hierarchy Front end vs Bad visual hierarchy Front end.

Visual hierarchy is a method of organizing user interface elements in order of importance. In other words, the idea of this principle is to have a well-organized user interface; in ways that explains the level of importance of each element or content and guides users to take the desired action [45].

The average time people spend watching any design is 8 seconds, so that is why visual hierarchy is important, we need to have in mind this time when creating a design and structure correctly all elements depending on their importance to give the viewer the main information.
To ensure that users see the most important information first, then the next, and so on, you must by playing with a various elements such as color, size, font, and position...etc.

- **Size:** Size and scale one of the important tools, particularly in the case of text. It is human tendency to identify that bigger text is more important. Bigger elements are more noticeable that is why our eyes automatically catch the bigger text, images or illustrations.

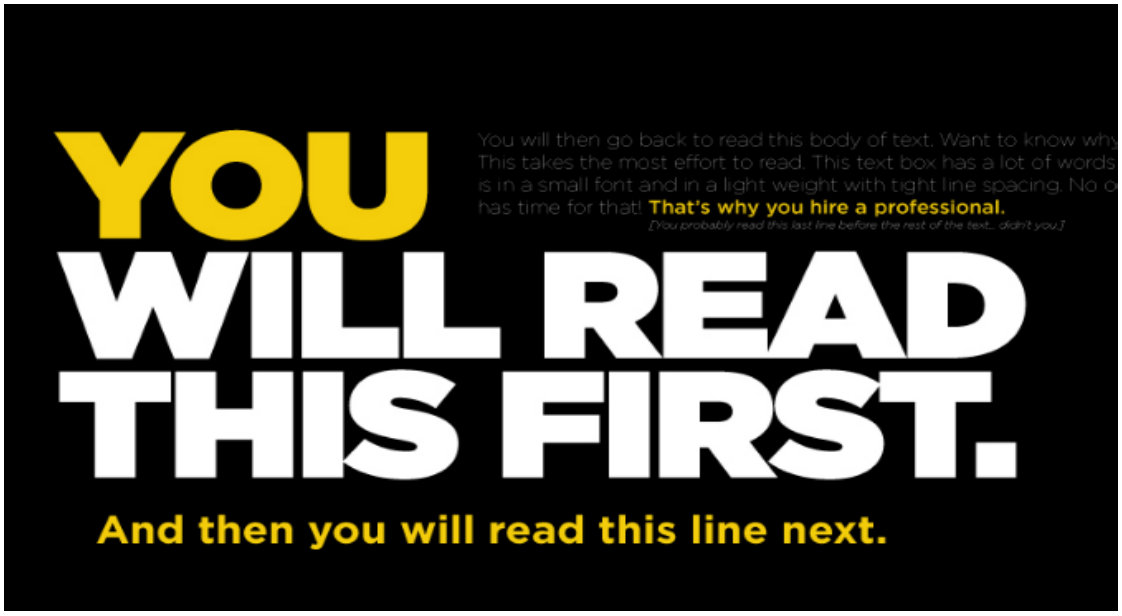


Figure 8: Creation of visual hierarchy with size.

- **Typographic Hierarchy:** People are not interested to read every single word you have to say. Typographic hierarchy allows you to organize your content in a way that helps readers find the information that is most pertinent to their needs.
  We can do this with Play around with different sizes, weights and styles of fonts to establish visual hierarchy. This is exactly what happens here.

Figure 9: Using different sized fonts to create visual hierarchy.

- **Color:** The first and one of the most necessary elements of establishing visual hierarchy is color. Bright colors stand out the most and can be used in muted color schemes to direct users to take a certain action.

✓ **Leveraging the Tenets of Color Psychology**

Colors trigger a response in the viewers. So, by using the right UI colors, you will stimulate a response in the viewers. So, using color psychology means you can modify the viewer's response and drive their behavior towards your desired objective.

For example: Blue's color associated with both water and sky, giving it a calming and peaceful effect. Blue represents stability, trust, peace, harmony, and calm. It soothes your mind. It linked to consciousness and intellect [46, 47].

Blue can make your brand appear reliable and trustworthy. Blue induces a calming effect on the body; making us feel safe, secure, strengthen the concept of trust.

For that we can see lot of Brand use the blue color as Facebook, PayPal for helping customers associate their brands with quality, reliable, and safe products.

Figure 10: Facebook and PayPal logo Brand.

✓ **Use Denotive Colors in Your UI**

You can use colors to denote a meaning to the elements of the interface. For example, a UI color like red is used to share the state of change in the interface and depict the not allowed action.

Figure 11: Denotive colors in User Interface.

Similarly, the green color can be used to say that the desired action is complete. In this, the UI designers also need to maintain the color consistency for dark and light modes. For example, it is not good to have different colors to show the Success and Failure state in light and dark interface mode.

✓ **Apply 60-30-10 color Rule to distribute the color**

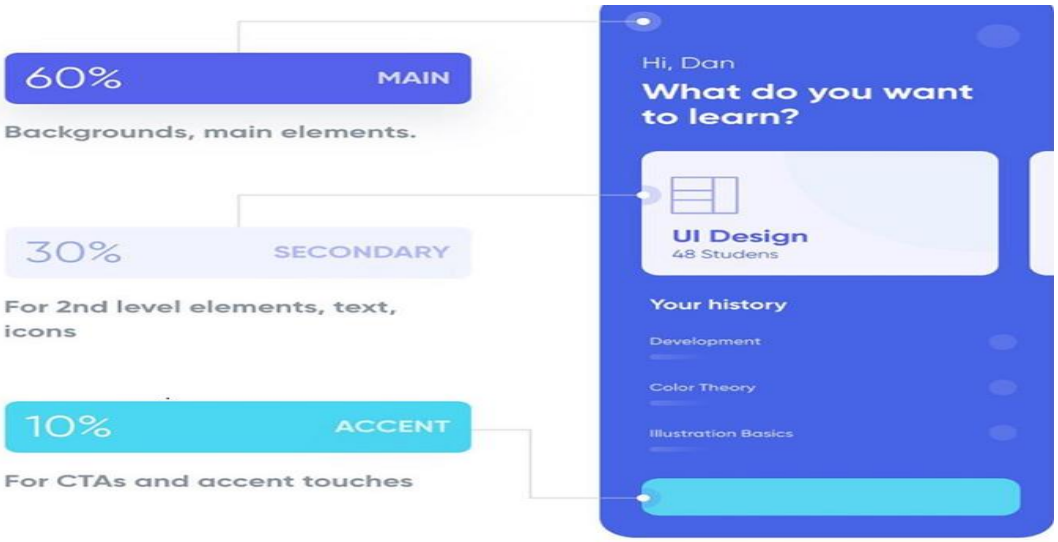


Figure 12: 60-30-10 color Rule to distribute colors in User Interface.

The 60–30–10 rule is a necessary addition to every sort of UI design and one of the most important things to cover in UI color schemes. This rule implies that you need to use 60% color composition of the dominant color, 30% for the secondary color, and 10% for the accent color.

It is useful as it helps create a balance and provides a unique comfort to the eyes for moving between different areas within the application or website.

- **Placement:** Users scan pages in a particular pattern. There are page areas that will get more attention than others. You can place important content in those high attention areas. Above the fold and along the top and left-hand sides is where the user's eyes fixate on the most.
  For example, the users are habitual to see the global menu or the logo of a website on the top. Not finding the global menu on top can lead to confusion and bad user experience.

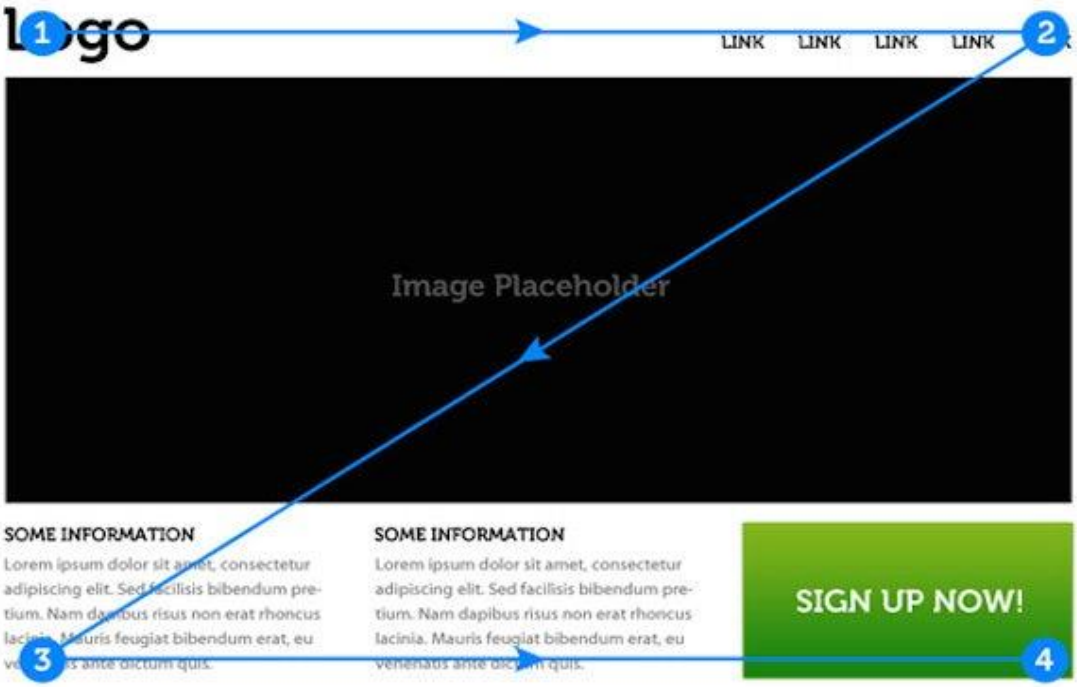


Figure 13: Website home page with good placement of logo and global menu.

- **Whitespace**: The space around your content can be used to draw the eye to specific parts of the content. For instance, white spaces break up content into easily digestible parts and allow the eye to zero in on various parts of the page.

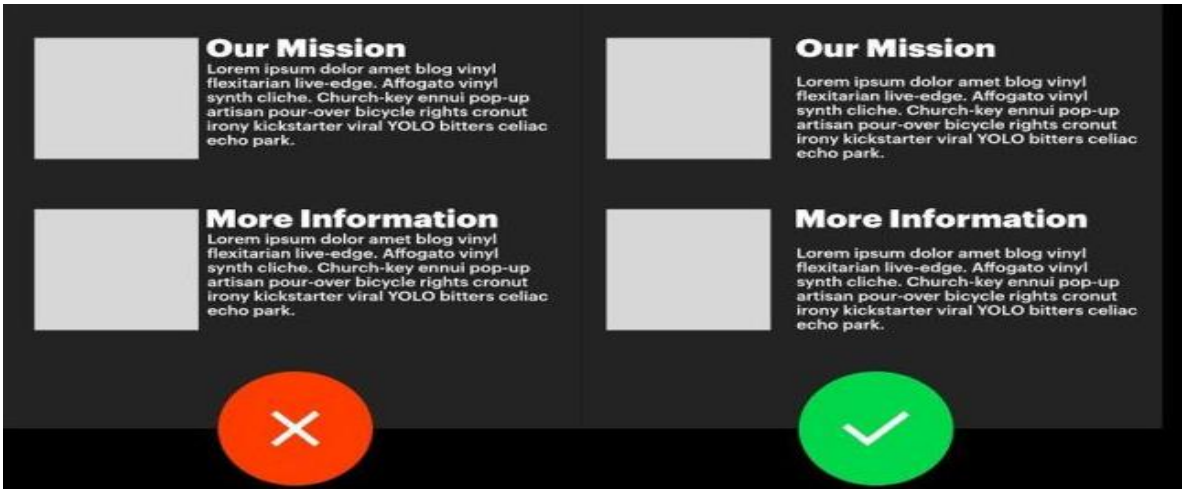


Figure 14: User interface with Whitespace vs user interface without Whitespace.

- **Contrast:** It is another important principle which helps us design better interfaces. It is not only about light and dark color opposites or small and big sizes: it is more than that. Contrast help us organize our content in a better way, also it helps users to focus on certain elements. Apart from emphasizing elements it also creates a visual balance and interest in design. However, too much contrast can be confusing for user [48].

  So designers should understand the priority of each element on the screen and size them accordingly like the title of a blog would require more attention than body text. Similarly, in interface design, a Call To Action (CTA) might need special importance than others.

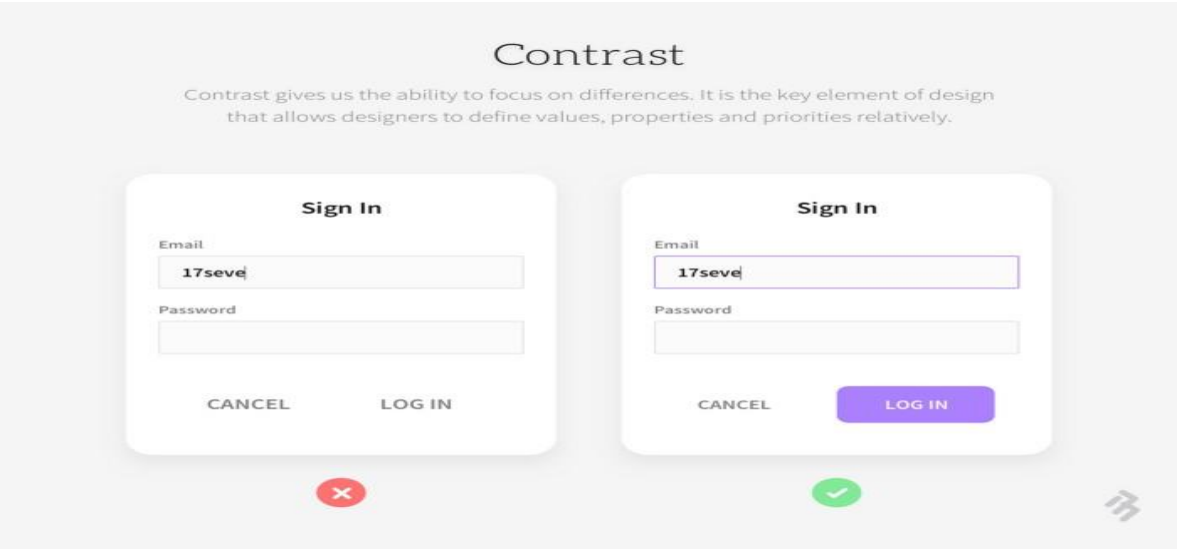


Figure 15: CTA button with contrast vs CTA button without contrast.

2.4.3 Place users in control of the interface

This design principle deals with the fact that users want to control digital products, whether to complete tasks or customize things to suit their needs.

It is true that design teams try to give users as much control as possible. This is manifested in designs that allow users to undo things, change settings, customize the appearance of the user interface, and create shortcuts, and so on. Users love being able to tailor the product to truly suit their needs, creating a completely unique experience [49].

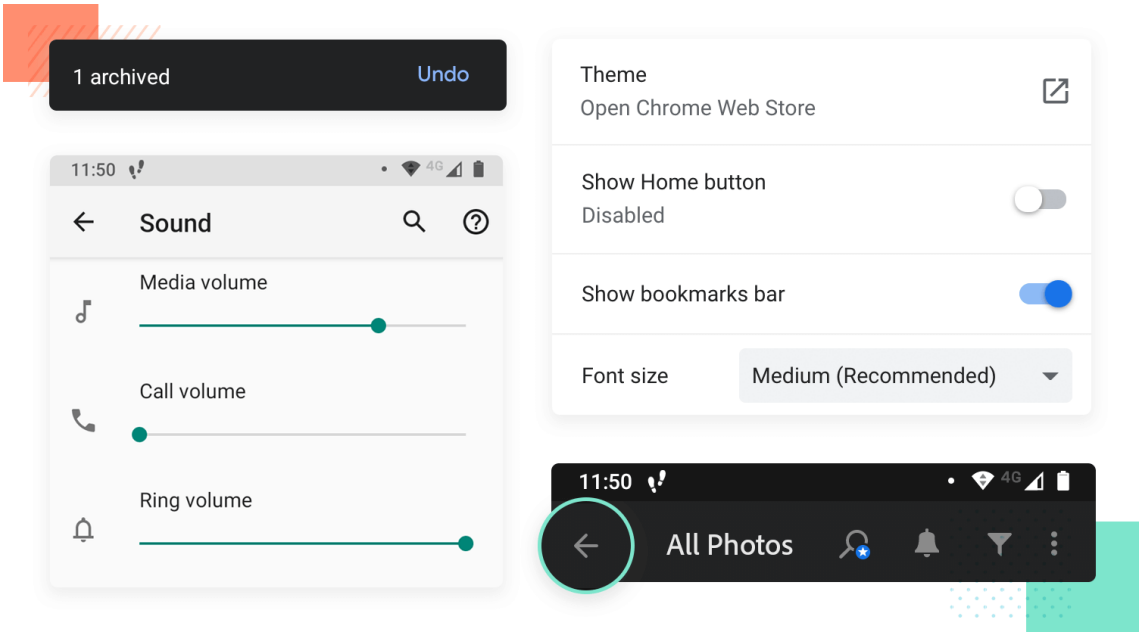


Figure 16: Mobile setting.

2.4.4 Make actions reversible – be forgiving

No matter how clear your design is people will make mistakes. For that the interface should be tolerant and have the able to handle the human mistakes. If a mistake is performed, the user should always be able to revert back without any losses for that Undo and redo options should be provided wherever necessary.

As the input field no one likes to start over because he/she put in the wrong birth date format whenever you are creating forms, allow users to click the Back or Previous button and go back to the page they were last on. Do not take them all the way back to the homepage or the start [44, 45].

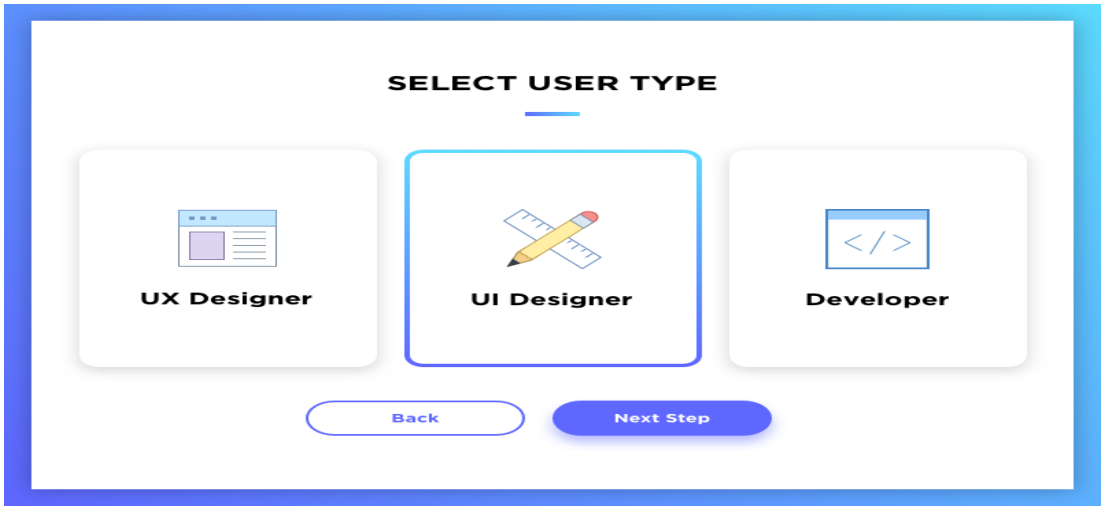


Figure 17: Multi-Form step with Back button.

Another good example for be forgiving on the front end development is when a user forgets his password; we will not block his account straightaway. We will provide hints and recovery methods.

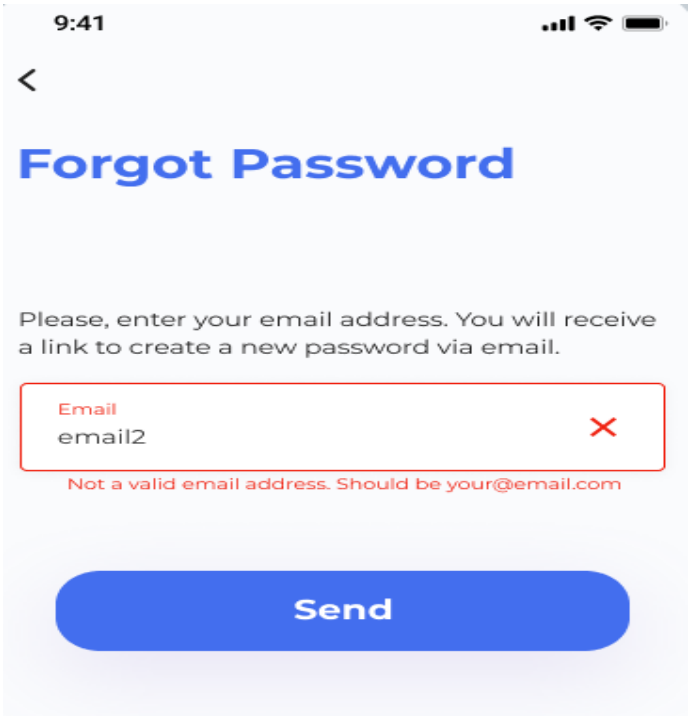


Figure 18: Create another password in the case of the users lost their password.

Also 'Undo' function can be extremely helpful when users choose system function by mistake. In this case, the undo function serves as an 'emergency exit,' allowing users to leave the unwanted state.

Like here, the 'Cancel' action after deletion gives the user the option to undo an action in case it was performed by mistake.

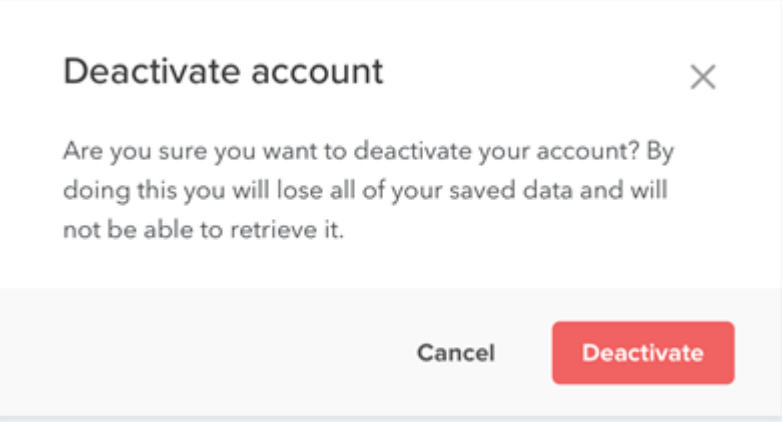


Figure 19: Confirmation message about deactivate account in the case of the users choose it by mistake.

Also, Gmail's notification message one of good example of such emergency exits with an undo option when users accidentally delete an email.

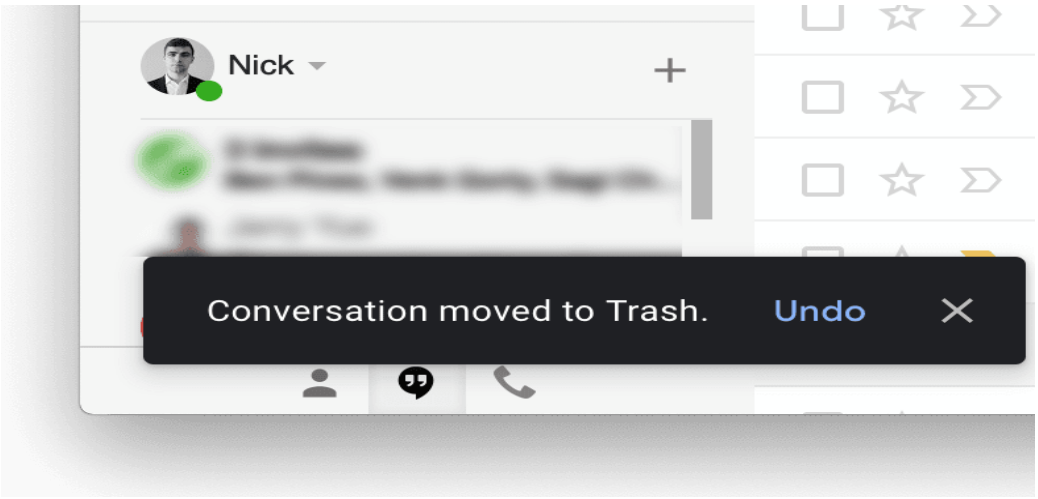


Figure 20: Gmail's notification message.

2.4.5 Error Prevention

Good user interfaces not only tolerate a user when an error has occurred but they are also actively designed to prevent errors from happening in the first place.

You can achieve that with provide hints and guidance so that the user can achieve tasks without any mistakes.

For example, while setting a password, a good UI system prompts about the length of password. It also tells what symbols and characters to include and how to set a strong password.

Figure 21: Error message informs the user what symbols and characters to include and how to set a strong password.

One common way of promoting errorless use is to hide actions and options that are invalid. If an operator tries to dispatch an unload command on a vehicle, for that to work, the vehicle must have a load. If all options are available, and an operator clicks “Unload” they are going to get frustrated when the system tells them this failed. Instead, the system should recognize that the action is not possible, and hide it from the selection.

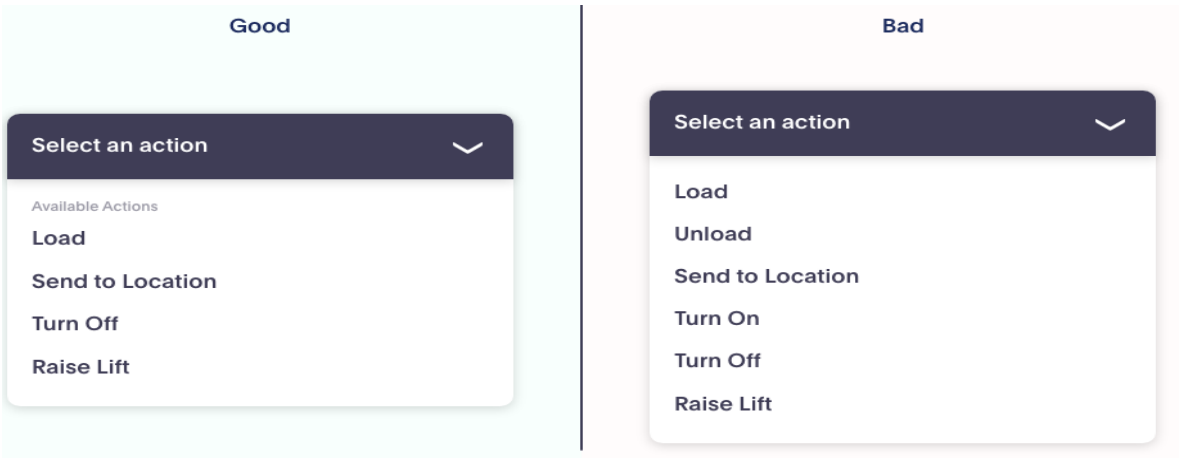


Figure 22: Prevent errors with hide actions from the selection.

2.4.6 Provide informative feedback – be acknowledging

Your interface should at all times speak to the user, when his/her actions are both right and wrong or misunderstood. Always inform your users of actions, changes in state and errors, or exceptions that occur. Due to this principle, the users can avoid performing actions repeatedly for surety [50].

Here are several ways to achieve that:

✓ **Every user action in the interface should give a reaction**.

For example, when users interact with an interactive object (such as a button), it is essential to provide some indication that an action has been acknowledged. We can achieve that by using button states, in each state we change the form and the color of the button to make more effective visual communication with the users [47].

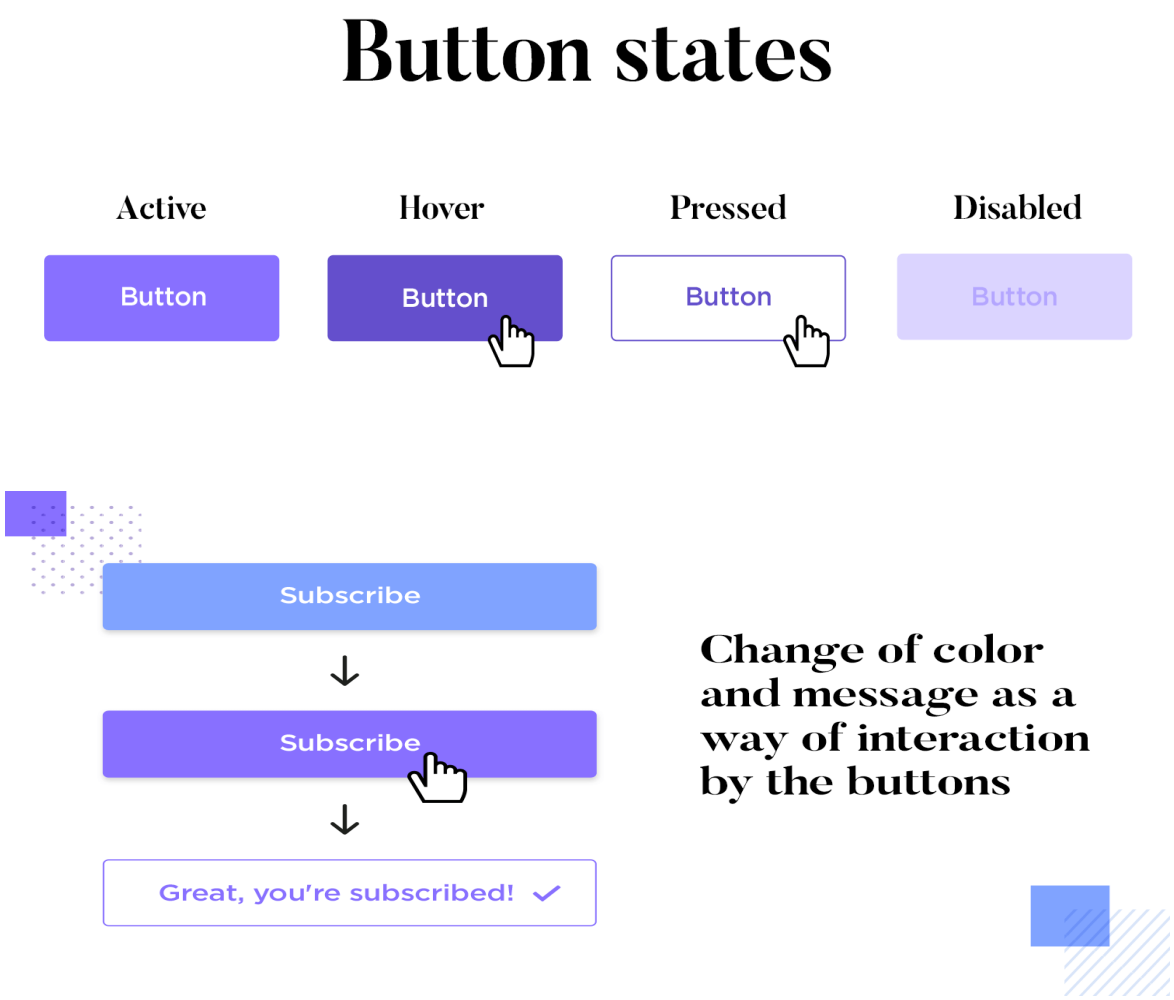


Figure 23: Change the form and the color of the button as way of interaction with the users.

✓ ***Use notification message***

From the good ways, also to make the users know that something is happening as a consequence of their actions or inform them if the activity done is successful or not is notification message.

For example, you have sent a message for newsletter from a website. If you do not receive a success notification from them in any form, you might feel confused [48]. You might want to register again to make sure the action has taken place successfully.

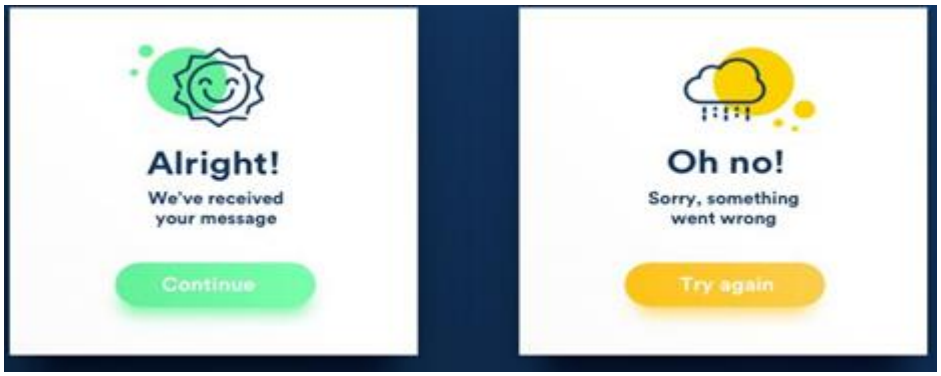


Figure 24: Notification message.

✓ **Use Communicate errors**

It is important to provide a clear and understandable reason for an error (where possible), and give instructions so that the error can be remedied. Leaning on technical language, or database error terms will not be beneficial to the user. Color is also important as it can help indicate the severity of an error or warning.

Figure 25: Not valid email address error message.

2.4.7 Show the visibility of system status

Visibility of system status is essential when users initiate an action that takes some time for a computer to complete. As the users need to know that there is a reaction coming from the computer, they need also to know where they stand on the process pathway (it just open or it completed or it cancelled) and how much is still to go [44].

For example, if there is a loading screen, uploading or downloading file, the percentage of work done should be displayed. So that, the user knows that the work is in progress and do not start panicking.

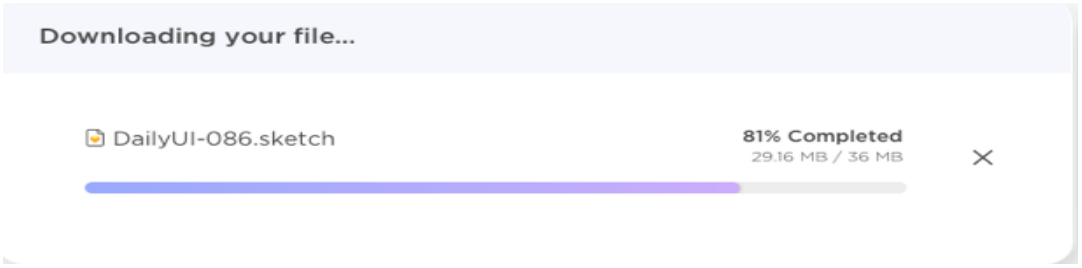


Figure 26: Progress bar of downloading file.

2.4.8 Strive for Simplicity and Clarity Above All

"Good design is obvious. Great design is transparent."

Minimalism and clarity one of the things that should be at the front of the minds of designers and developers when they want to create something unique because in the rush of creativity to that it can be easy for them to unintentionally clutter their interfaces and even the product itself.

The best interface designs are invisible; when the users visit your website, you want them to walk the particular way and do the specific steps. If they have to solve a labyrinth to find what they want, they will be more likely to leave.

The front end should be very straightforward, and easy to understand for simplifying the interaction with the app for the user.

Here are several ways to achieve that:

✓ Avoid cluttering the interface

With unnecessary or irrelevant information, the user interfaces that have too many features can lose their focus and dilute their allure by creating a negative experience for users. It is important that design teams show restraint and prioritize the truly crucial bits.

Whenever you are thinking about adding a new feature or element to your interface, ask the question, “Does the user really need this?” or” Are you adding things because you like or want them?

Figure 27: Bad users interface because it has too much element.

In the other hand we have Google page is simple with minimal design that prevents cluttering and maintains.

Google maintain clarity by designing for the user’s main goals is the research. The developers don’t confuse the users with extraneous information that are irrelevant to those goals. For that become one of the most usable browsers in the world.

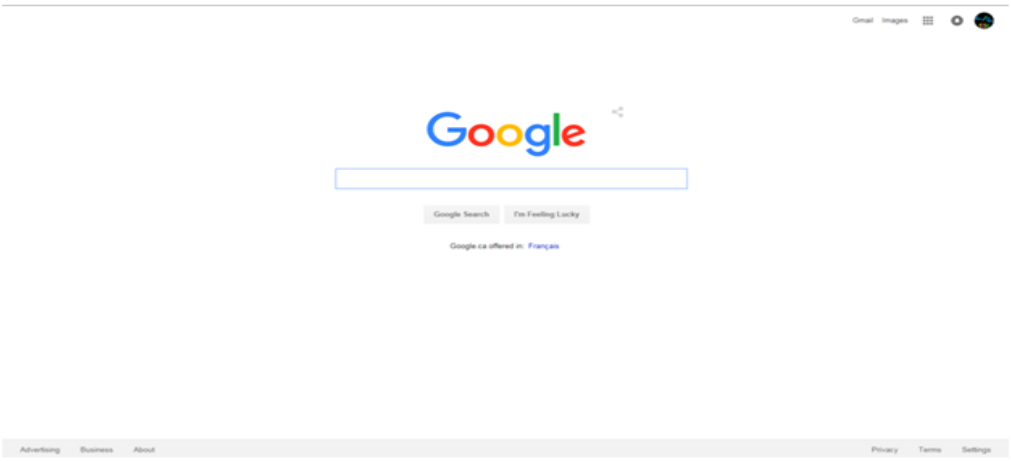


Figure 28: Google page.

✓ **If a task is complex and difficult, break it into simpler steps**

Complexity in interface design should be avoided as much as possible. Therefore, this will reduce the cognitive load on the user.

For example, the form is one of the principal causes of the user anxiety and frustration on any app; they will likely never be filled out. There are a few psychological factors at play here, but essentially, users just do not want to sift through a giant field block. For that, you can break up this complex activity into smaller steps or process for the purpose of making the experience seem simple and more manageable.

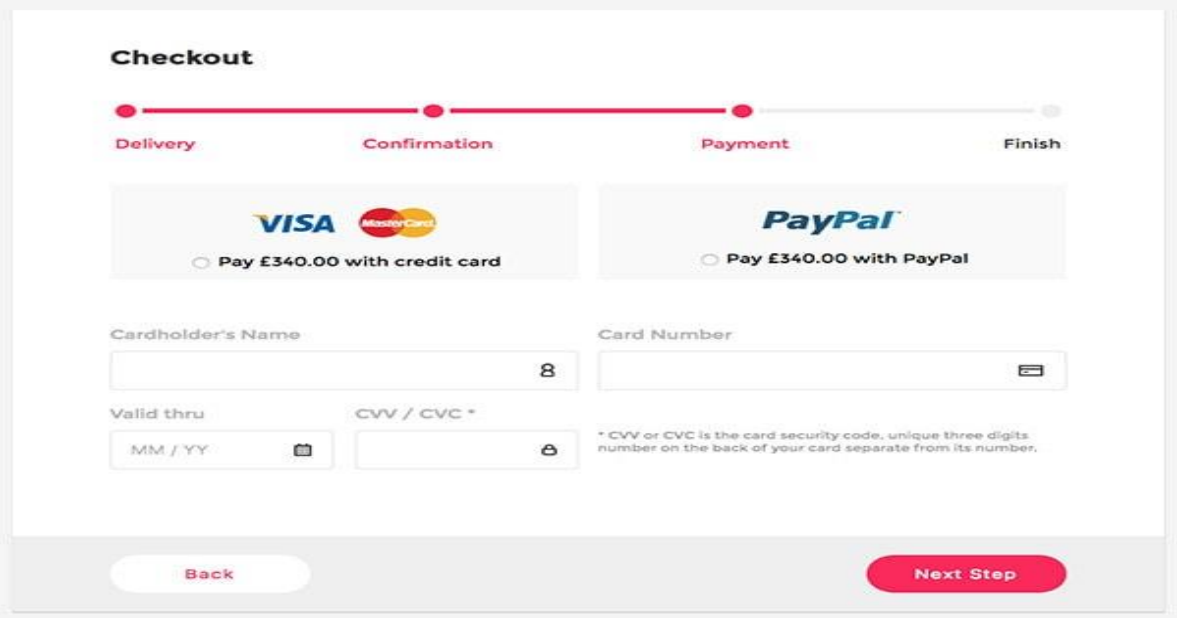


Figure 29: Checkout form with mutli-step to break the complex of activity.

✓ **Provide clear and concise labels for actions and keep your messaging simple**

For example:

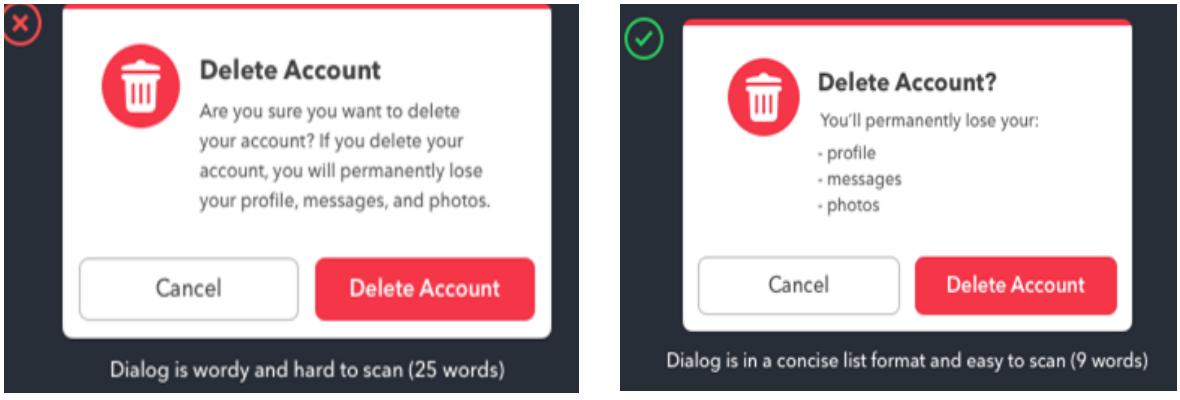


Figure 30: Hard dialog to scan vs Clear and simple dialog to scan.

2.4.9 Familiarity and Consistency

✓ **Stay away from jargon and system-oriented terms**

It is important to use language that is easy to read and understand. The system should speak the user's language, with words, phrases, and concepts familiar to the user, rather than jargon or system-oriented terms.

For example, imagine if designers changed the labels of the familiar to the user. As 'Find' instead of 'Search'? Or if the label for the home page became 'Main' for the main page instead of 'Home'?

This will create confusion and stops users from completing their task, and if users are uncertain, they will not feel comfortable completing a transaction. This is not what you want when users are shopping on your site.

Figure 31: High familiarity vs low familiarity naming for the user.

✓ **Use real-world metaphors**

Despite the fact that most of us are now extremely familiar with digital products, it is still a good idea to use real-world metaphors. Some designers feel very strongly that these metaphors improve the general usability of the product, because they are so easy to understand even at first glance [48].

There are many examples of these metaphors in elements that users will know and recognize. The silliest one, perhaps, is the garbage bin icon. It immediately tells us that anything placed in there will be removed from sight, possibly eliminated forever. There is power in that kind of familiarity. It is the reason why buttons look like real physical buttons or why toggle switches look like actual switches. Arguably, all icons are real-life metaphor.

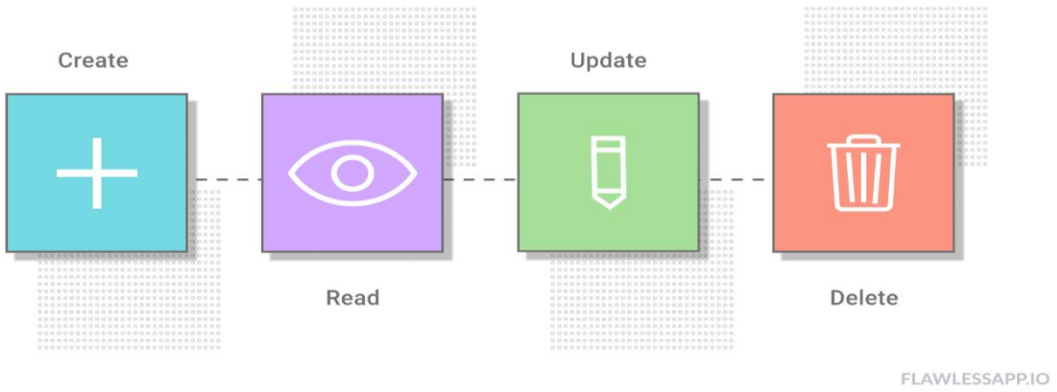


Figure 32: Real world metaphors icons.

2.4.10 Make it easy to explore and use

This can sound a bit broad and general, but it is a fundamental principle in UI that connects to many important concepts. A product that is easy to use is more likely to offer a high standard of usability, enjoy a short learning curve and be effective in helping users achieve tasks [44].
We can achieve that with:

✓ **Design the navigation system effectively**

This will help the user to navigate through the interface smoothly. Visual cues should be in place that set priorities and guide a user into the desired direction. A user will likely ask 'where am I?' or 'how can I get where I want to go'. Knowing what users expect and organizing the content accordingly is key to good information architecture and UX design.
The user should at all times know his location in the interface and should not feel lost. In this case, you can use breadcrumbs to show the user journey. This will allow him to easily navigate without any confusion.

Figure 33: Location-based breadcrumb links are particularly useful as supplemental navigation.

2.5 Conclusion

In this chapter, we presented the Front End techniques, its definition, role and benefits. Following this we presented the User Centered Design process, steps and methods. Then we gave the Front End Design principles.

The following chapter is dedicated to present the analyze and design of our application.

Chapter III: Analyze and Design

3.1 Introduction

For several decades, the computer world has always dreamed of a process that can guarantee the efficient development of quality software, valid whatever the size and complexity of the project, and presenting good practices adapted to the method in question, especially since , nowadays, the requested software is more and more imposing and demanding than before. Modeling is a very important step in the development of computer projects, it is a process aimed at specifying the needs and requirements of the actors, the system and the overall architecture of the work in the form of diagrams. The goal of this chapter is to define some important general points on the method and tools highlighting the realization of our project. We will begin by presenting the unified modeling language UML (Unified Modeling Language), define the generic approach of the software development process that accompanies it, and finally we will present the main basic concepts of web services.

3.2 Requirement Specification and Analysis

The first step in engineering a software system is to understand what the system should do this is referred to as the specifications of the requirements for the software system or, alternatively, as Requirements Specifications or Software Requirements Specifications (SRS) [51]. Once the requirements have been specified, very often the next step is to analyze them to update, modify, and prioritize the requirements so that the software development team develops a better understanding of the system to be developed. The requirements specification and analysis is a twostep process.

3.2.1 Requirement Specification

Requirements specification is the first phase of an application life cycle. The purpose of this step is to describe the application to be developed. System requirements can be categorized as either functional requirements or nonfunctional requirements [52].

Figure 1: Functional requirements Vs Nonfunctional requirements.

a) Functional requirements

Functional requirements can be most simply defined as: something the system must do. If the system does not meet a functional requirement it will fail. This is because it will not be able to achieve something it must do to operate properly. The functional requirement concept can also be understood through reviewing the system in terms of inputs and outputs. Functional requirements specify what the system must do in response to different inputs and what it must output [53, 54]. In this part we detail the functionalities, which the system must provide to the different actors, which are:

- Management of doctors.
- Management of patients.
- Management of assistants.
- Management of specialties.
- Management of Wilayas.
- Management of appointments.
- Management of schedules.
- Management of communication between users.

b) Nonfunctional requirements

Nonfunctional requirements in software engineering can be explained as: Requirements that describe how the system works. Nonfunctional requirements are focused on how the system goes about delivering a specific function. At first glance they might be seen as less important than functional requirements but both have a part to play in a good system. In short, nonfunctional requirements are all about system usability. If nonfunctional requirements are not met, users may become frustrated with how the system works and go elsewhere [54].

And for our application this are nonfunctional requirements:

- The code must be clear to allow future evaluations or improvements.
- The platform should provide quick access to information, and should update in real time.
- The platform offers a friendly and easy to use interface.
- The platform must guarantee the confidentiality, integrity and consistency of the data.

3.2.2 Requirement Analysis

Requirements analysis involves frequent communication with system users to determine specific feature expectations, resolution of conflict or ambiguity in requirements as demanded by the various users or groups of users, avoidance of feature creep and documentation of all aspects of the project development process from start to finish. Requirements analysis is a team effort that demands a combination of hardware, software and human factors engineering expertise as well as skills in dealing with people [55].

a. Identification of Actors

An actor is the idealization of a role played by an external person, process or thing that interacts with a system. Our application contains four main actors:

1. **The administrator:** The administrator of our online appointment system is responsible of managing access rights and deleting actors and also managing the specialties and Wilayas for each doctor.
2. **The doctor:** The role of our doctor is to manage his schedule and his assistants.
3. **The assistant:** The essential role of the assistant is to manage patient's appointments and can also managing doctor's schedule.
4. **The patient:** The patient can definitely search his doctor and book his appointments.

b. Identification of Use Cases

A use case defines a way of using the system and describes its functional requirements. Each use case contains one or more scenarios that define how the system should interact with users (called actors) to achieve a specific job goal or function. A use case actor can be a human or another system external to the one we are trying to define. It makes it possible to adopt the language of the end user or the domain expert. Each use case is represented in a use case diagram [56].

❖ **The administrator:**

- Authentication.
- Manage Specialties (add, delete, edit, search).
- Manage Wilayas (add, delete, edit, search).
- Manage Doctors (validate, delete, search).
- Manage Patients (add, delete, edit, search).

- Manage Users messages (send, replay).

❖ **The doctor:**

- Authentication.
- Sign Up.
- Manage Schedule.
- Manage Assistant.
- Add CV.
- Contact administrator.
- Manage Profile.

❖ **The assistant:**

- Authentication.
- Manage Schedule (add, delete, edit).
- Manage Appointments (add, delete, edit).
- Consult appointments list (validate, cancel).
- Contact administrator.
- Manage Profile.

❖ **The patient:**

- Authentication.
- Sign Up.
- Search doctor (search with Wilaya, specialty, and name of doctor).
- Manage Appointments (add, delete, edit).
- Contact administrator.
- Manage Profile.

3.3 The modeling language UML

3.3.1 Definition

UML, short for Unified Modeling Language, 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 [57, 58]. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software [59, 60].

3.3.2 Types of UML Diagrams

The current UML standards call for 13 different types of diagrams: class, activity, object, use case, sequence, package, state, component, communication, composite structure, interaction overview, timing, and deployment. These diagrams are organized into two distinct groups [61]:

a. Structural UML diagrams

- Class diagram.
- Package diagram.
- Object diagram.
- Component diagram.
- Composite structure diagram.
- Deployment diagram.

b. Behavioral UML diagrams

- Activity diagram.
- Sequence diagram.
- Use case diagram.
- State diagram.
- Communication diagram.
- Interaction overview diagram.

- Timing diagram.

3.3.3 Use case Diagram

The use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e. use case diagram). A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior [62]. A use case diagram is usually simple. It does not show the detail of the use cases:

- It only summarizes some of the relationships between use cases, actors, and systems.
- It does not show the order in which steps are performed to achieve the goals of each use case.

❖ **The global use case diagram**

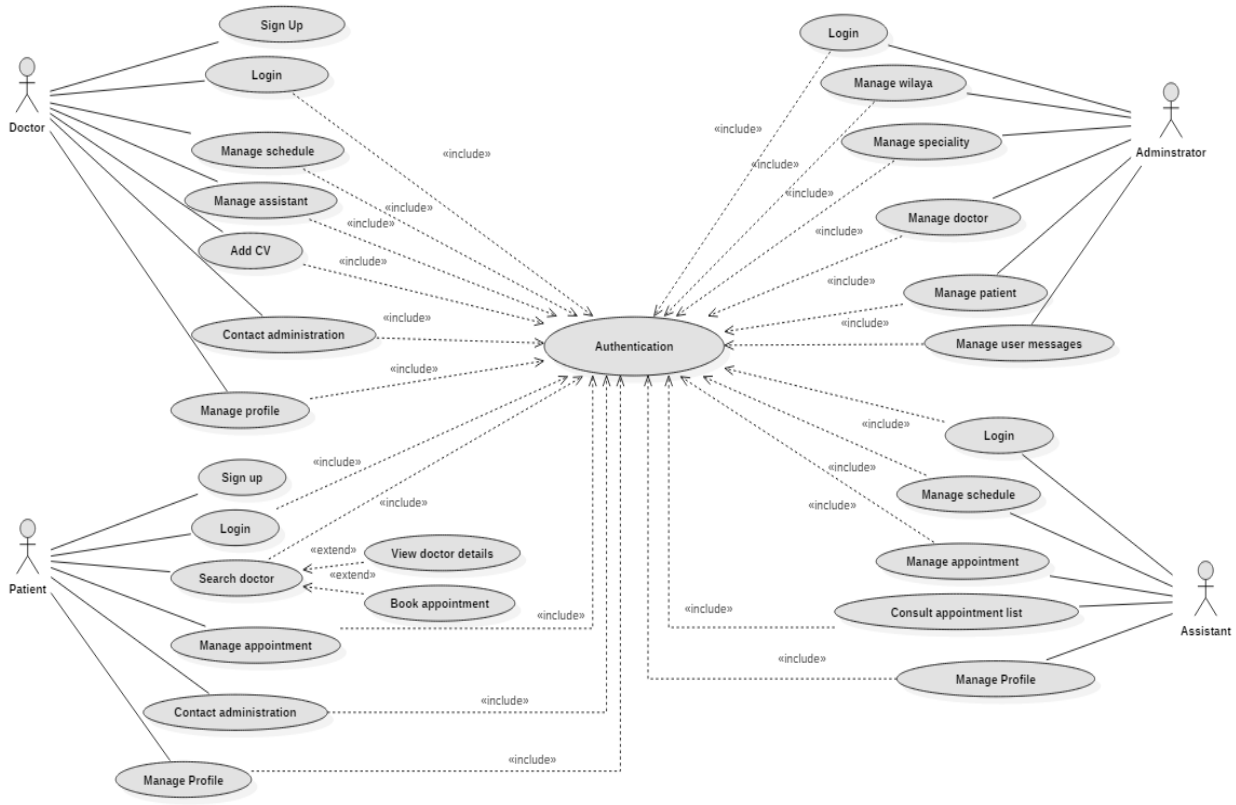


Figure 2: Global use case diagram.

❖ Administrator use case diagram

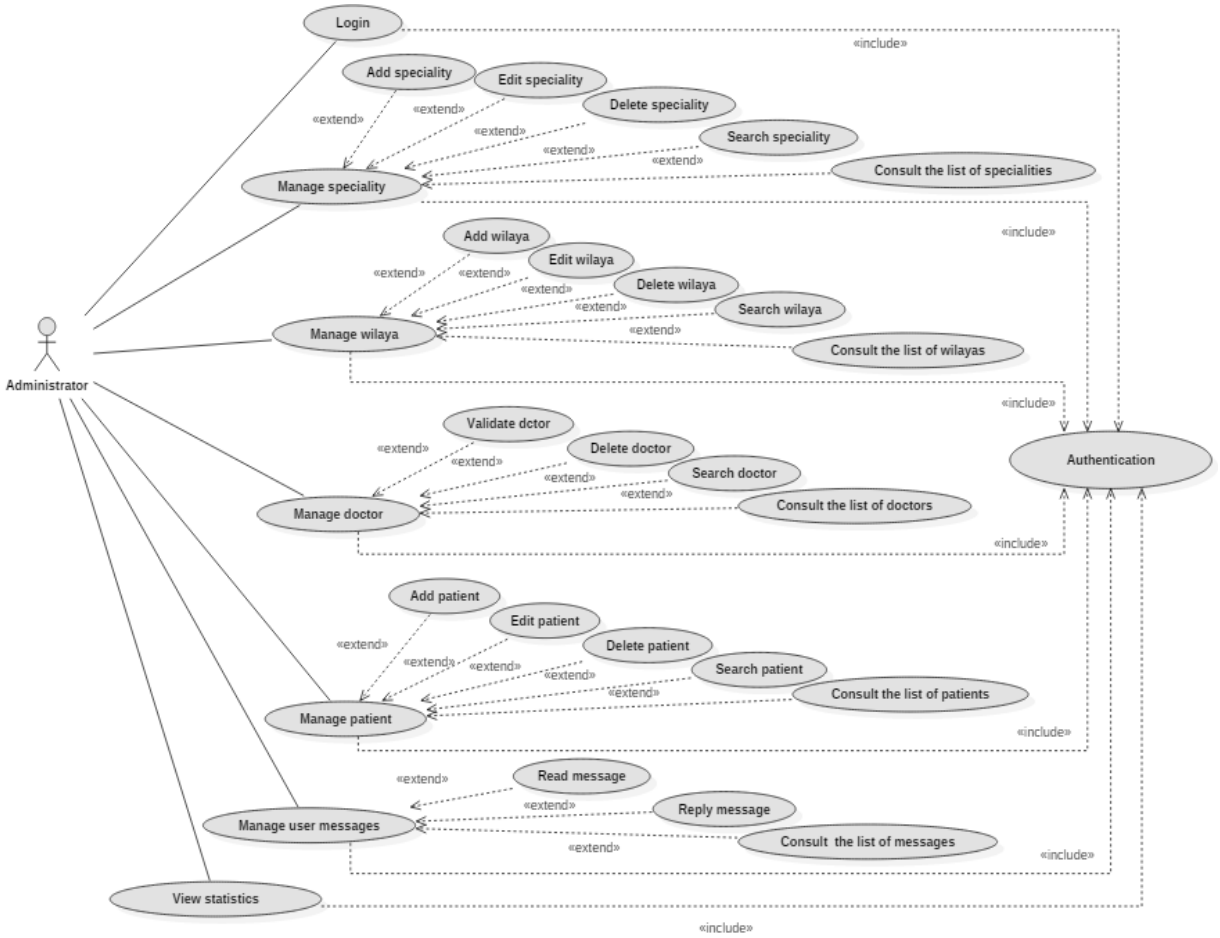


Figure 3: Administrator use case diagram.

❖ Doctor use case diagram

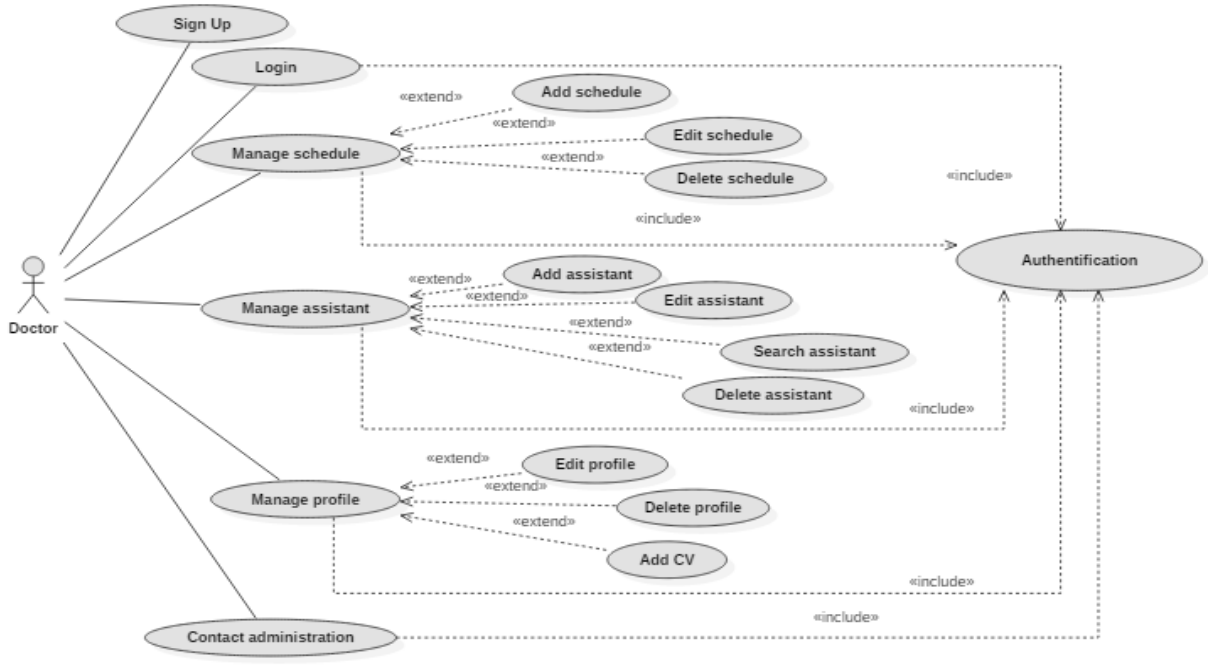


Figure 4: Doctor use case diagram.

❖ Assistant use case diagram

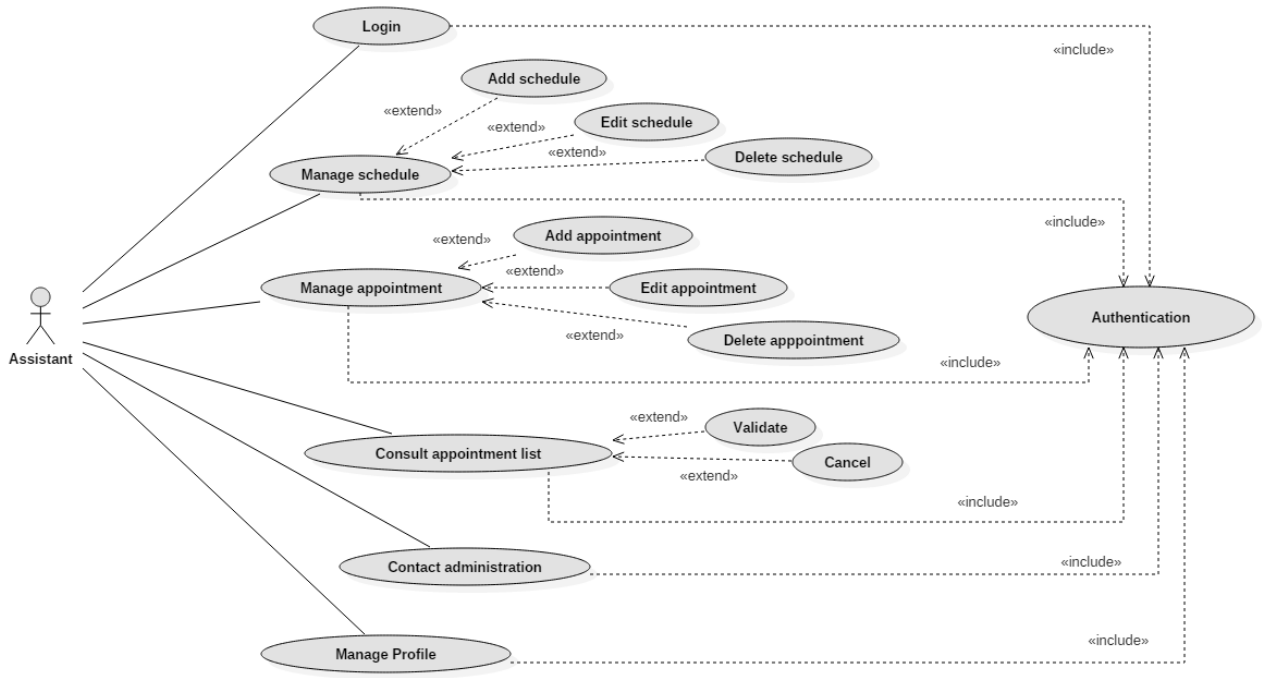


Figure 5: Assistant use case diagram.

❖ Patient use case diagram

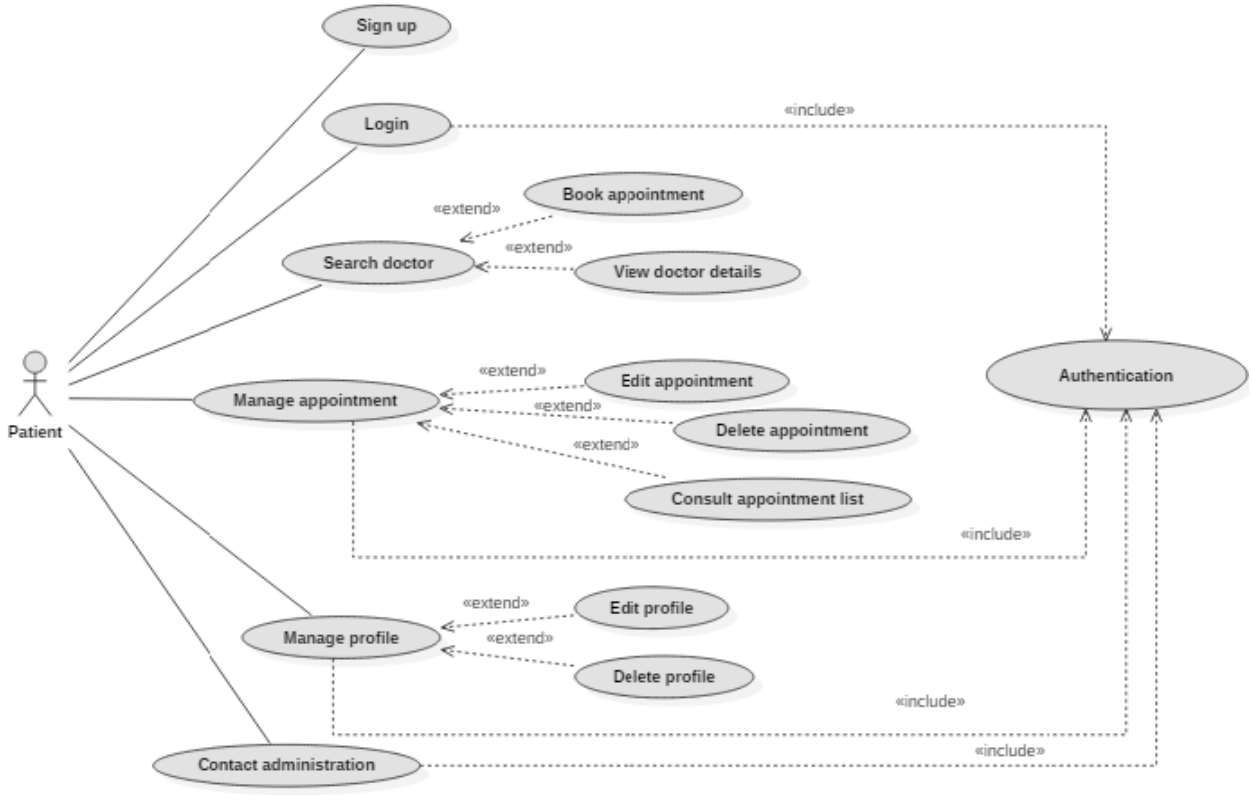


Figure 6: Patient use case diagram.

3.3.4 Class Diagram

The class diagram is one of the UML classes and illustrates the system objects, as well as the static relationships that may exist between them. Furthermore, the class diagram illustrates the properties and operations of class objects, as well as the restrictions that apply to the connection [65, 66].The objects of the class diagram are the following:

1) Class Diagram: The boxes within the diagram represent classes. Each class is divided into three sections. The upper section is for the class name. The middle section is for the class attributes. The lower section is for the class operation.

2) Properties: Essentially, each property is a single concept, which represents a class feature. The properties are represented in two different approaches, namely attributes and associations.

3) Attributes: describe a property in a text form in the class box.

4) Associations: simply, the association is the other approach to notating a property. About the same information that appears in the attributes can be shown in the association.

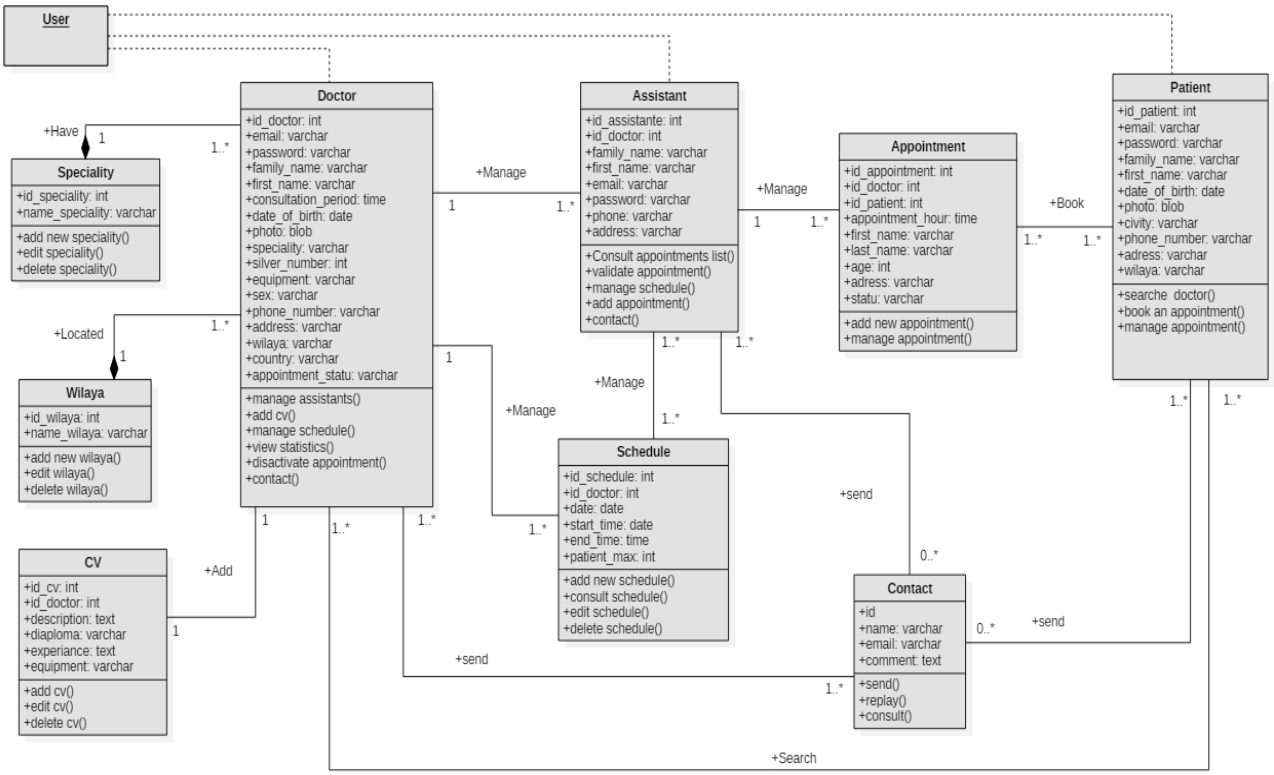


Figure 7: Class diagram.

3.3.5 Sequence Diagram

A sequence diagram models the behavioral aspects of a model. A sequence diagram also called as event diagram projects the way objects interact with each other and exchange the messages among each other using parallel vertical lines called as lifelines and horizontal arrows shows the messages exchange between the objects also tells about when to send what messages[63, 64, 66].

- **Authentication**

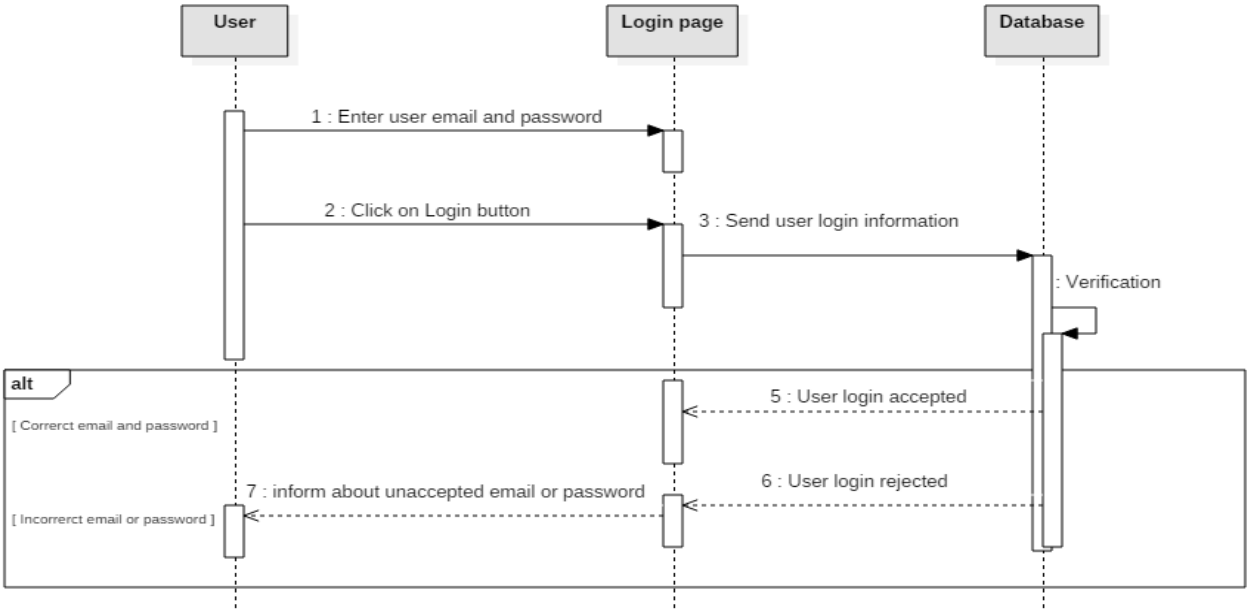


Figure 8: Sequence diagram «Authentication».

- **Sign Up**

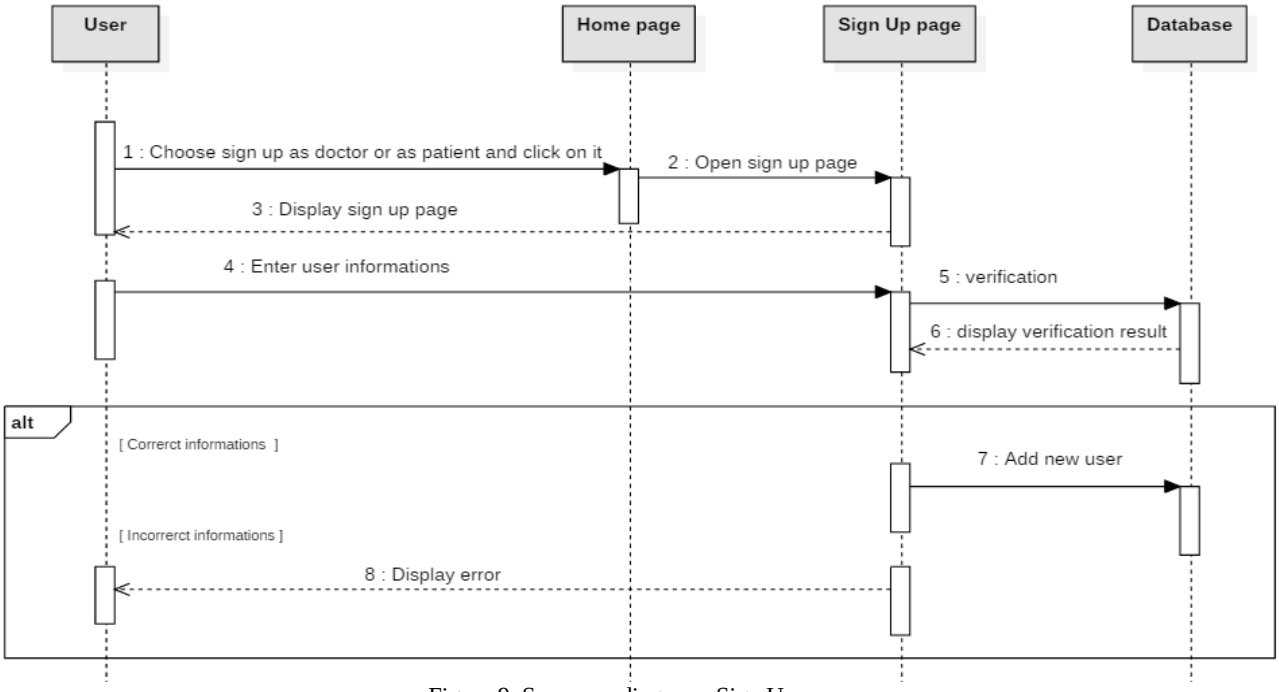


Figure 9: Sequence diagram «Sign Up».

- **Search doctor**

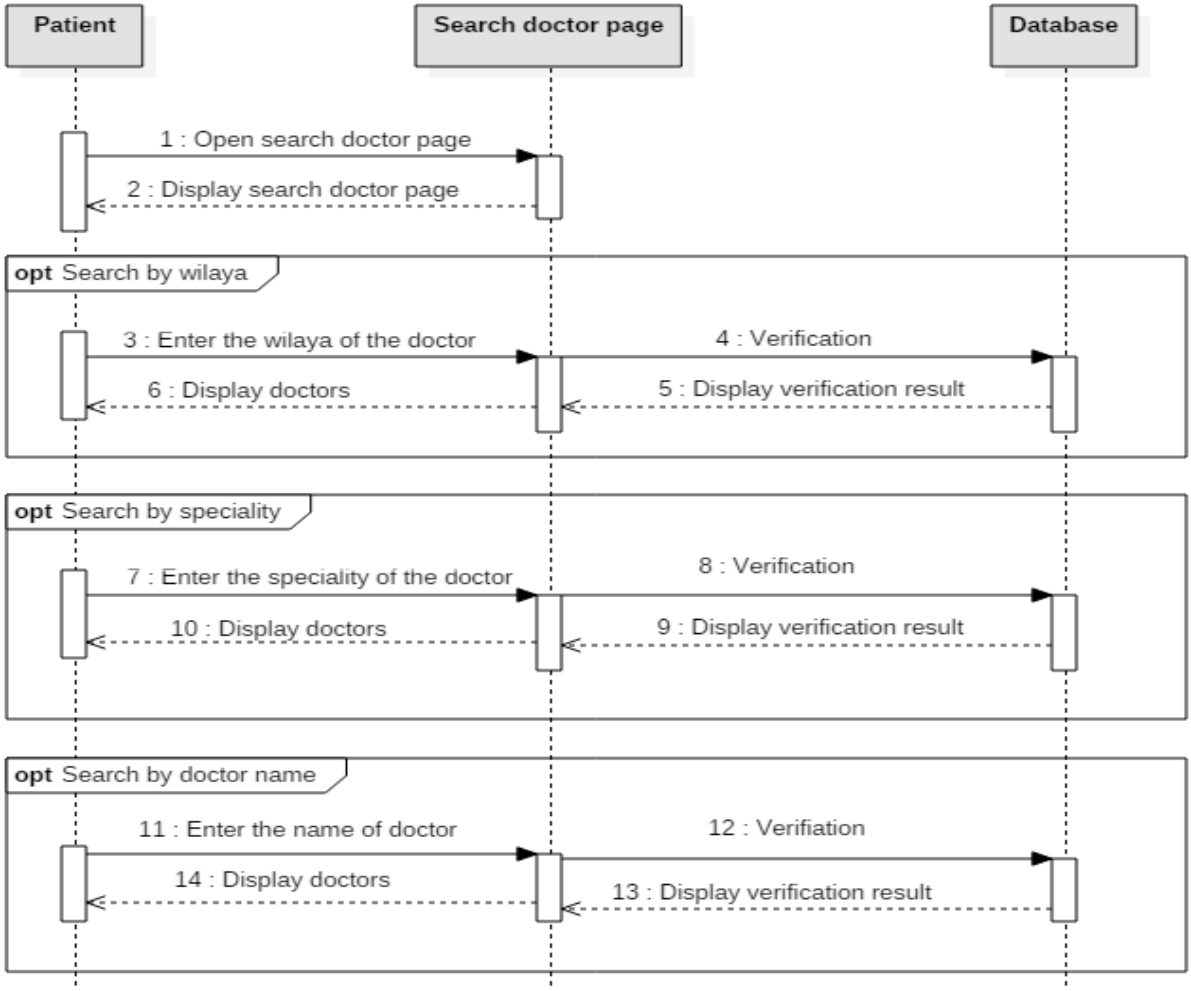


Figure 10: Sequence diagram «Search doctor».

- **Book appointment**

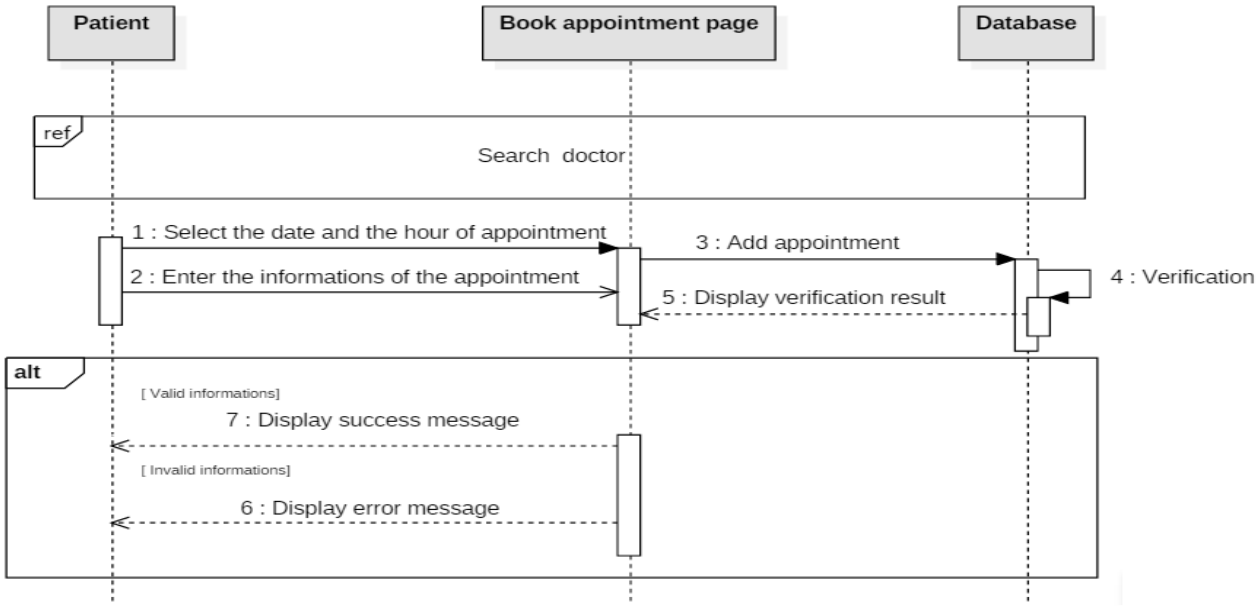


Figure 11: Sequence diagram «Book appointment».

- **Manage appointment**

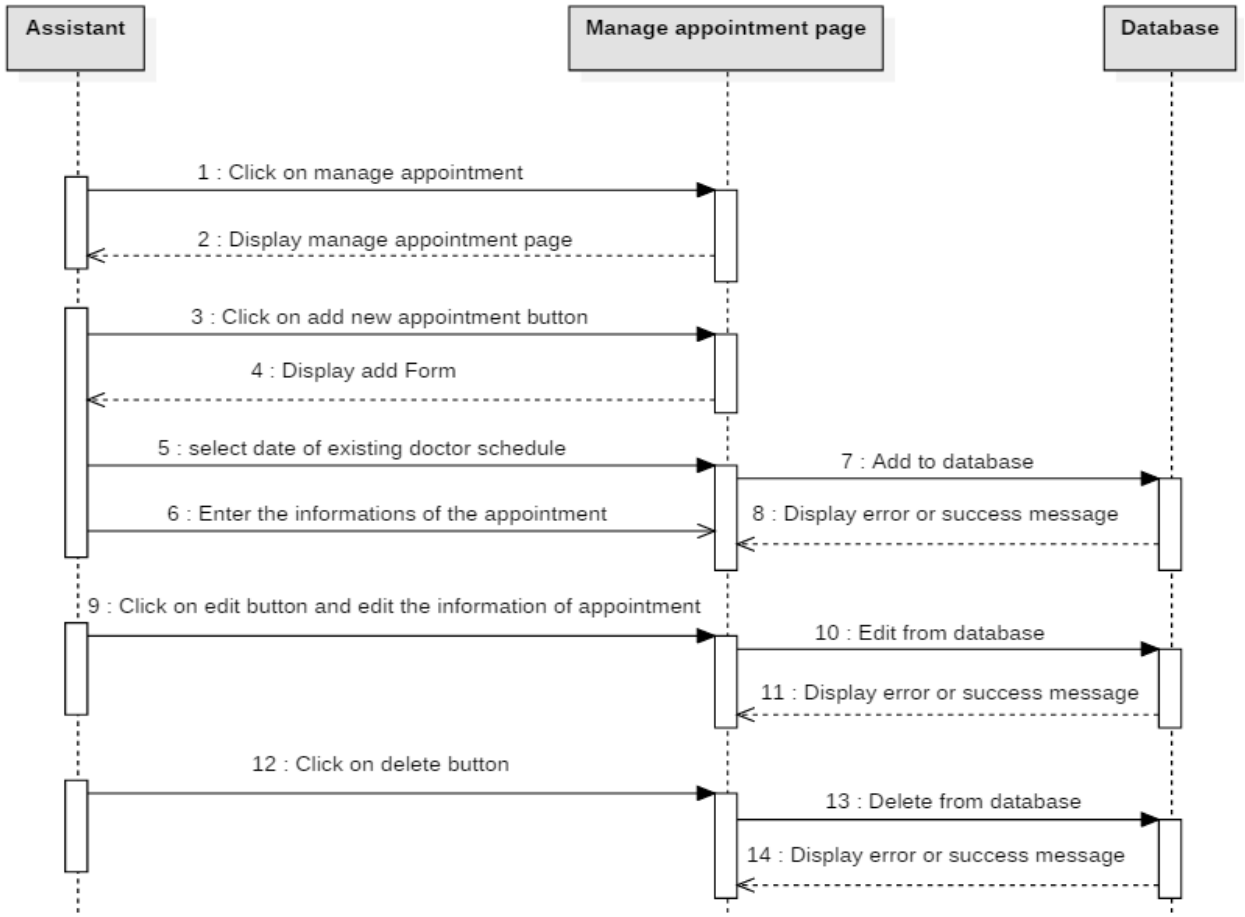


Figure 12: Sequence diagram «Manage appointment».

- **Validate appointment**

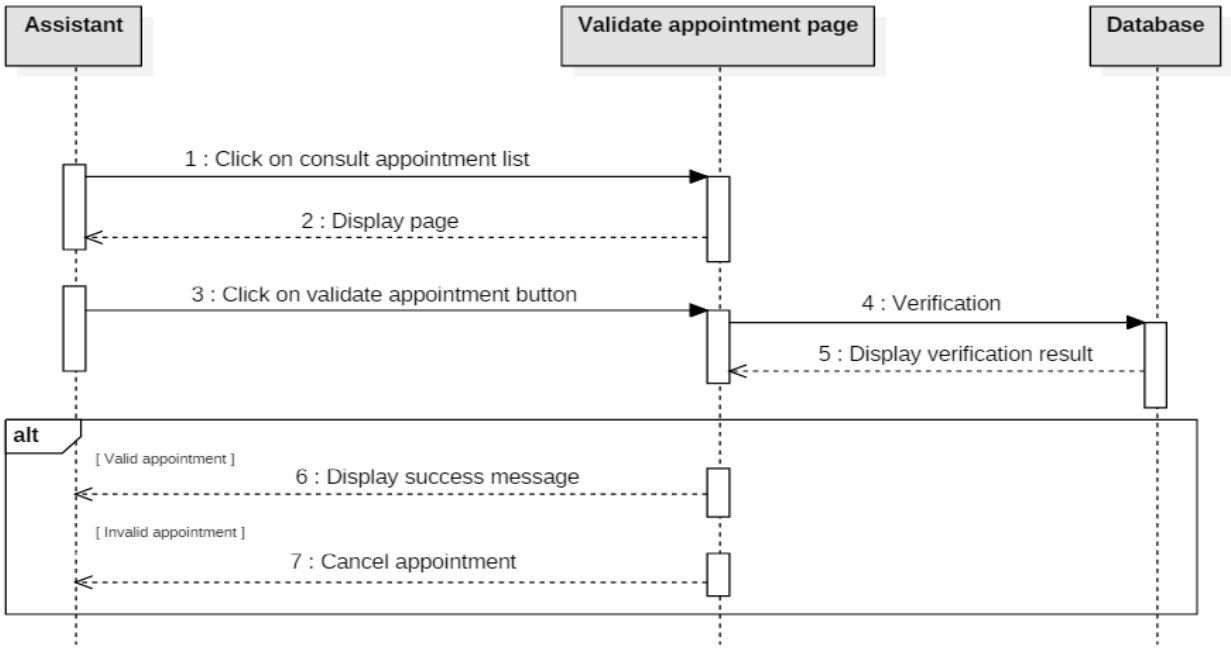


Figure 13: Sequence diagram «Validate appointment».

- **Validate doctor**

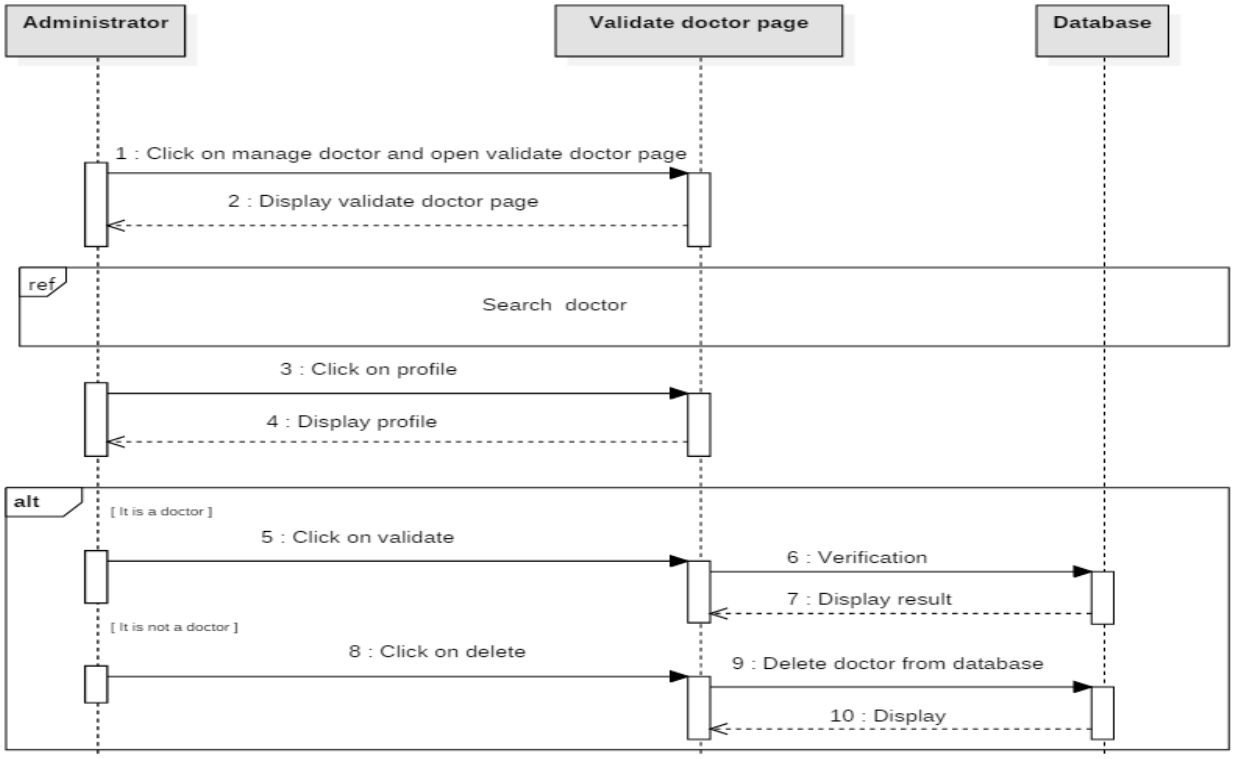


Figure 14: Sequence diagram «Validate doctor».

- **Manage schedule**

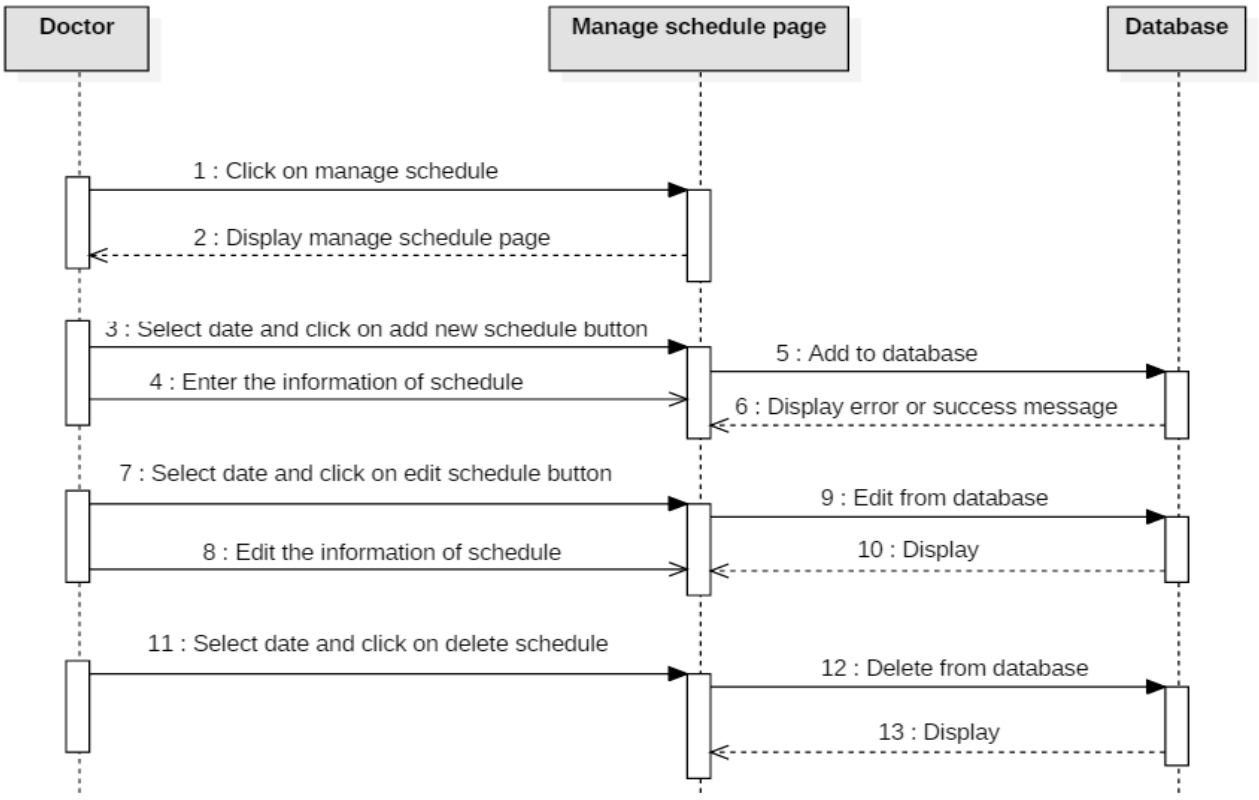


Figure 15: Sequence diagram «Manage schedule».

3.4 Modeling with Website UX Flowcharts

3.4.1 Definition

Flowcharts are diagrams of user flows and tasks in processes. Designers use these versatile tools to visualize the interactions in designs and present easy-to-understand maps of designs to stakeholders. They connect labeled, standardized symbols with lines to show everything users might do in interactive contexts.

3.4.2 Map out New Designs and Evaluate Existing Ones using Flowcharts

In user experience (UX) design, designers use flowcharts mainly to plot how users move through an interface, such as an app, to achieve their goals. They describe the relationships between pages/screens and show all interactive possibilities, the starting points, actions required, moments of decision and endpoints as users encounter and use interfaces [67].

3.4.3 How do I design a UX website flowchart?

To design your UX website flowchart, you will want to take all that wonderful user research along with your flowchart outline and build it out using one of several design tools made specifically for creating flowcharts and diagrams. There are a few options out there, so make sure you choose one that will be the easiest to implement with your website design software and is also accessible by all of your cross-functional team members [68].

Once you have selected the design tool you are going to use to create your UX flowchart, begin by taking all those steps and tasks you documented in your outline and start to connect them. This process may feel similar to when you are using a wireframe tool to build out the wireframes for your prototype, and that is good because these user flows will help guide that process, as well.

You will want to make sure that you create an easy-to-understand legend for the shapes and colors you use so that anyone on your team can quickly pick up the diagram and understand it. For example, you might use squares to represent each screen the user will access throughout the flow, whereas you might use diamonds to represent when a user must make a decision. No matter how you decide to design your flowchart, make sure you test it out with different team members to ensure it is as accessible as possible by others.

3.4.4 Website UX Flowcharts

❖ **Website UX Flowchart of the Administrator**

Figure 16: Website UX Flowchart of the Administrator.

❖ Website UX Flowchart of the Doctor

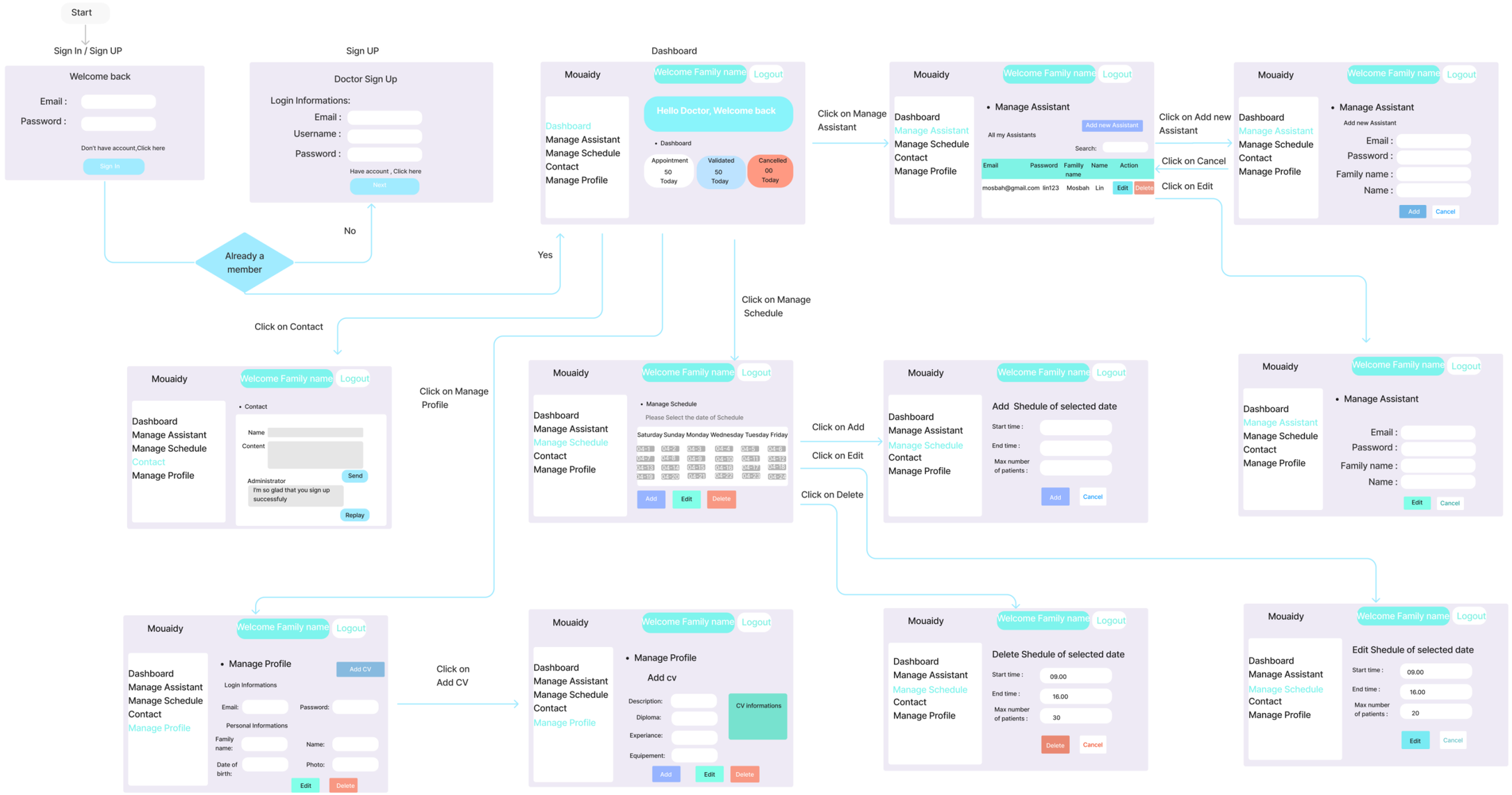


Figure 17: Website UX Flowchart of the Doctor.

❖ Website UX Flowchart of the Assistant

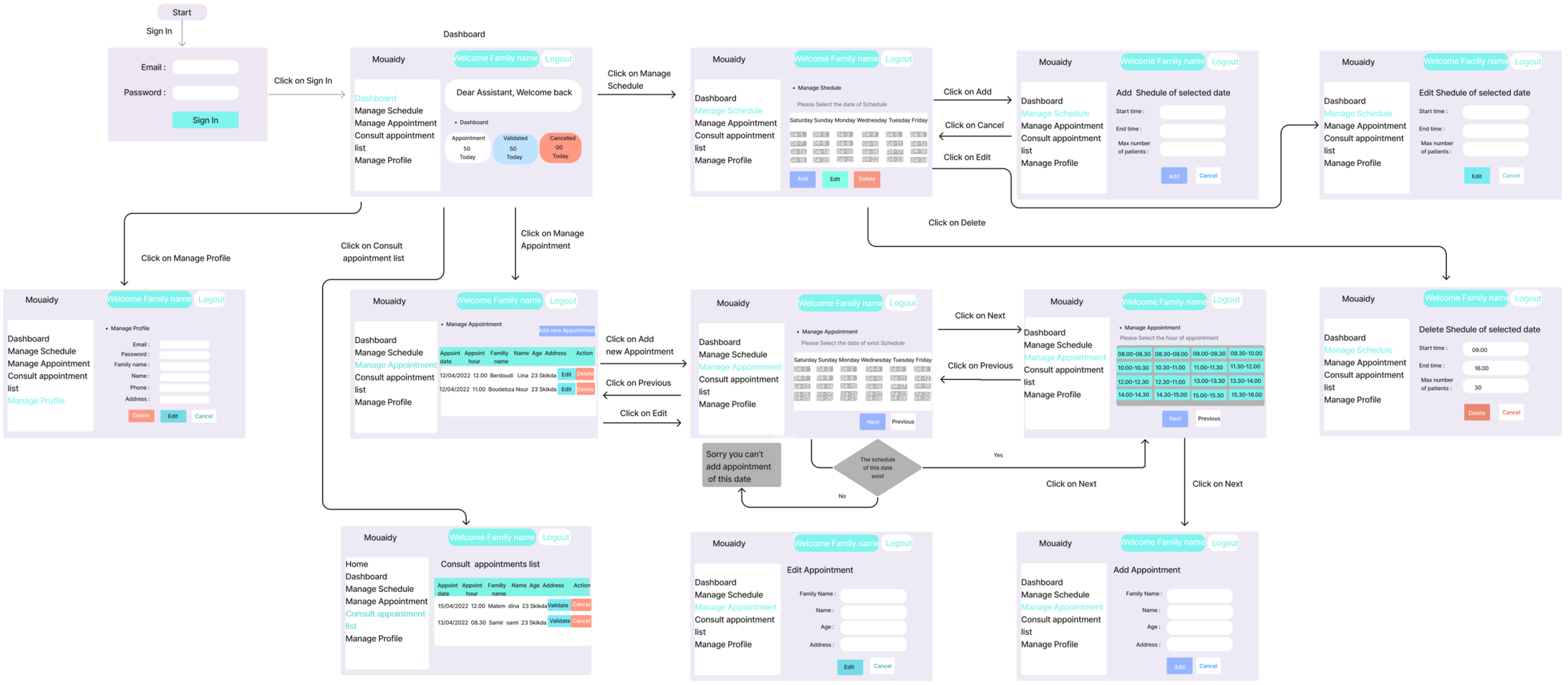


Figure 18: Website UX Flowchart of the Assistant.

❖ The Website UX Flowchart of the Patient

Figure 19: Website UX Flowchart of the Patient.

3.5 Design Tools

3.5.1 Star UML

StarUML is an open source software modeling tool that supports the UML (Unified Modeling Language) framework for system and software modeling. It is based on UML version 1.4, provides eleven different types of diagram and it accepts UML 2.0 notation. It actively supports the MDA (Model Driven Architecture) approach by supporting the UML profile concept, allowing and generating code for multiple languages [69].

3.5.2 Figma

Figma is a vector graphics editor and prototyping tool which is often used for UI design (free, online UI tool that lives in the browser) and primarily web-based, with additional offline features enabled by desktop applications for macOS and Windows. It is currently the industry's leading interface design tool, with robust features that support teams working at every stage of the design process. Like Google Docs, Figma allows many designers to work together on a single document at the same time.

It includes features like real-time collaboration for team and communication with the client, visual versioning, easy design preview, working in the browser and may others [70].

3.6 Conclusion

In this chapter, we presented the design or the conception phase of our project which contains: the use case diagrams, sequence diagrams that helped us to describe in a detailed way the functioning of the system in order to facilitate the implementation, and also the class diagram which illustrates in a global way the structure of the elements that constitute the database associated with our application. Then we used the Website UX Flowchart to plot how users move through an interface.

After having developed the design of our application, we approach now in next chapter to expose the phase of implementation.

Chapter IV: Implementation

4.1 Introduction

In this chapter we will present the architecture of the web application, implementation environment, technologies and programming languages that we used, as well as, some of screenshots of our web application.

4.2 Web Application Architecture

Briefly, the web application architecture is a “skeleton” or layout that displays the interactions between application components, middleware systems, user interfaces, and databases. This kind of interaction allows a number of applications to work together simultaneously [71].

Once a user opens a webpage, the server sends specific data to the browser as a response to the user’s request. To be precise, a web client (or user agent) may request web resources or more commonly-known web documents (HTML, JSON, PDF, and so on) through a web server. Then, with these minimal manipulations, the requested information appears. After that, the interaction between a user and a website starts.

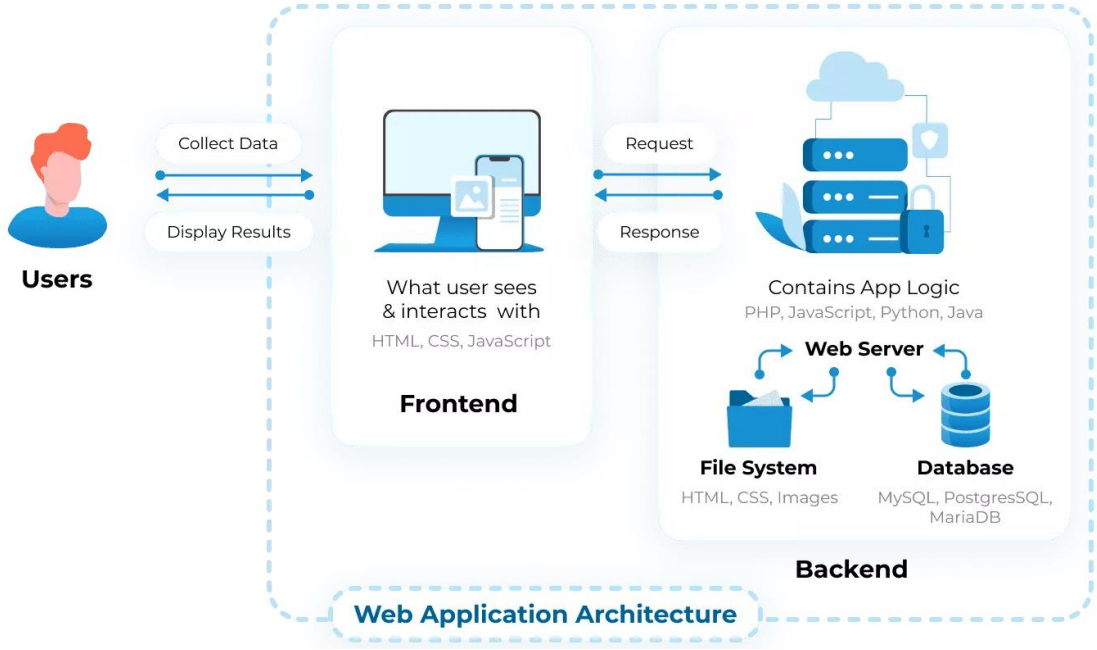


Figure 1: Web Application Architecture.

4.3 Development environment

The implementation of the application requires the use of appropriate hardware and software tools. Thus, we used the following resources:

4.3.1 Hardwares:

Materials	PC DELL	PC ACER
Processeur	Intel(R) Celeron ® CPUN3060 @ 1,60GHz 1,60GHz	Intel(R) Core ™ i3 CPUM 370 @ 2,40GHz 2,40GHz
Memoire RAM	4.00 Go	3.00 Go
Systeme	Windows 10 Professionnel, 64 bits	Windows 10 Professionnel, 64 bits

Table 1: Hardwares used.

4.3.2 Softwares

- **Visual Studio Code**

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as PHP, HTML, CSS, C++, C#, Java, Python, Go) and runtimes (such as .NET and Unity) [72].

- **WampServer**

WampServer is a Web development platform on Windows that allows you to create dynamic Web applications with Apache2, PHP, MySQL and MariaDB. WampServer automatically installs everything you need to intuitively develop Web applications. You will be able to tune your server without even touching its setting files. Best of all, WampServer is available for free (under GPML license) in both 32 and 64 bit versions. Wampserver is not compatible with Windows XP, SP3, or Windows Server 2003 [73].

4.3.3 Programming Languages

4.3.3.1 HTML

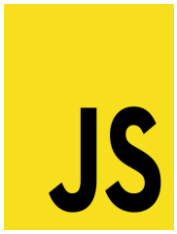
HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages [74]. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01.

4.3.3.2 CSS

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable [75]. CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs and variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

4.3.3.3 JavaScript



JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity [76].

4.3.3.4 JQuery

JQuery is a fast and concise JavaScript Library created by John Resig in 2006 with a nice motto: Write less, do more [77].JQuery simplifies HTML document traversing, event handling,

animating, and Ajax interactions for rapid web development. JQuery is a JavaScript toolkit designed to simplify various tasks by writing less code.

4.3.3.5 Bootstrap

Bootstrap is a free front-end framework for faster and easier web development. It also includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins, and gives you the ability to easily create responsive designs [78].

4.3.3.6 PHP

PHP is a popular general-purpose scripting language that is especially suited to web development. It is also fast, flexible and pragmatic. PHP powers everything from your blog to the most popular websites in the world [79].

4.3.3.7 MYSQL

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications [80].

It is developed, marketed, and supported by MySQL AB, a Swedish company, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is My Ess Que Ell. However, you can pronounce it in your way. Many small and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

4.4 Application Screenshots

4.4.1 Tables of the database used

We have chosen appointment_system as a name of our database, which contains 10 tables:

Table	Lines	Type	Interlacement	Size
admin	4	InnoDB	utf8mb4_general_ci	16 Kio
appointment	11	InnoDB	utf8mb4_general_ci	16 Kio
assistant	8	InnoDB	utf8mb4_general_ci	16 Kio
contact	4	InnoDB	utf8mb4_general_ci	16 Kio
cv	5	InnoDB	utf8mb4_general_ci	16 Kio
doctors	17	InnoDB	utf8mb4_general_ci	16 Kio
patients	10	InnoDB	utf8mb4_general_ci	16 Kio
schedule	6	InnoDB	utf8mb4_general_ci	16 Kio
specialty	2	InnoDB	utf8mb4_general_ci	16 Kio
wilaya	2	InnoDB	utf8mb4_general_ci	16 Kio

Table 2: Tables of the database used.

- **The table of admin**

Name	Type	Size	Key	Null
id_admin	int	11	primary	No
email	varchar	50		No
username	text	20		No
password	varchar	20		No

Table 3: Table of admin.

- **The table of appointment**

Name	Type	Size	Key	Null
id_appointment	int	30	primary	No
id_doctor	int	50		No
id_patient	int	50		No
apointment_hour	time			No
family_name_patient	text			No

name_patient	text			No
age	int	10		No
phone	int	11		No
schedule	date			No
status	tinyint			No
date_created	date			No

Table 4: Table of appointment.

- **The table of assistant**

Name	Type	Size	Key	Null
id_assistant	int	30	primary	No
id_doctor	int	30		No
email	varchar	50		No
password	varchar	30		No
family_name	text			No
first_name	text			No
phone	varchar	11		No
address	varchar	50		No

Table 5: Table of assistant.

- **The table of contact**

Name	Type	Size	Key	Null
id_contact	int	30	primary	No
name	text			No
email	varchar	50		No
content	varchar	100		No

Table 6: Table of contact.

- **The table of CV**

Name	Type	Size	Key	Null
id_cv	int	30	primary	No
id_doctor	int	30		No

description	text			No
experience	text			
diploma	text			No

Table 7: Table of CV.

- **The table of doctors**

Name	Type	Size	Key	Null
id_doctors	int	30	primary	No
email	varchar	50		No
username	text	20		No
password	varchar	20		No
family_name	text			No
first_name	text			No
date_of_birth	date	50		No
civity	text			No
phone_number	int	11		No
photo	text			No
n_approval	bigint	20		No
speciality	text			No
equipment	text			No
consultation_period	int	20		No
wilaya	text			No
city	text			No
address	varchar	50		No
state	text			No

Table 8: Table of doctors.

- **The table of patients**

Name	Type	Size	Key	Null
id_patients	int	30	primary	No
email	varchar	50		No
password	varchar	30		No

family_name	text			No
first_name	text			No
civity	text			No
phone_number	int	11		No
wilaya	text			No
photo	text			No

Table 9: Table of patient.

- **The table of schedule**

Name	Type	Size	Key	Null
id	int	30	primary	No
id_doctor	int	30		No
time_from	time			No
time_to	time			No
max_patient_number	int	20		No
date	date			No

Table 10: Table of schedule.

- **The table of specialty**

Name	Type	Size	Key	Null
id	int	50	primary	No
name	text			No

Table 11: Table of specialty.

- **The table of Wilaya**

Name	Type	Size	Key	Null
id_wilaya	int	60	primary	No
name	text			No

Table 12: Table of Wilaya.

4.4.2 Application User Interfaces

- **Home page**

This interface represents the home page of our online appointment application, where you can search for doctors, change language, register as doctor or patient and login.

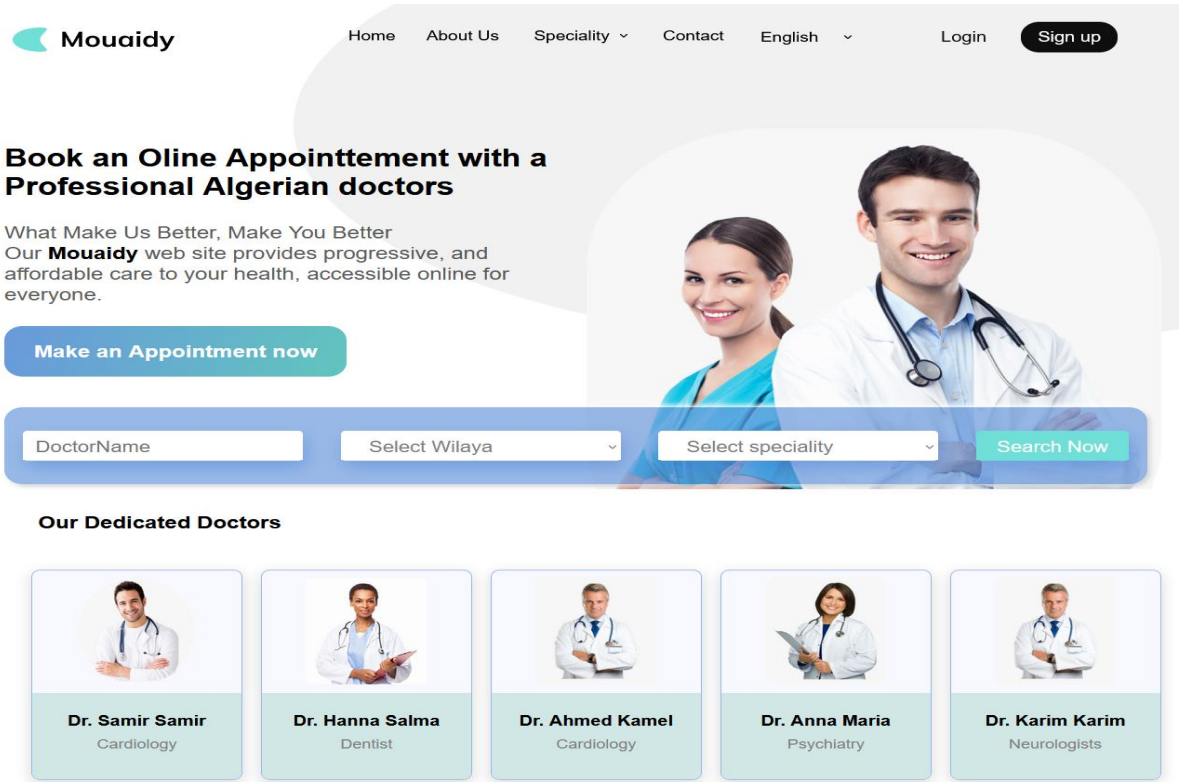


Figure 2: Home page.

- **Login page**

In this page, the users can login to their accounts by entering their email and password.

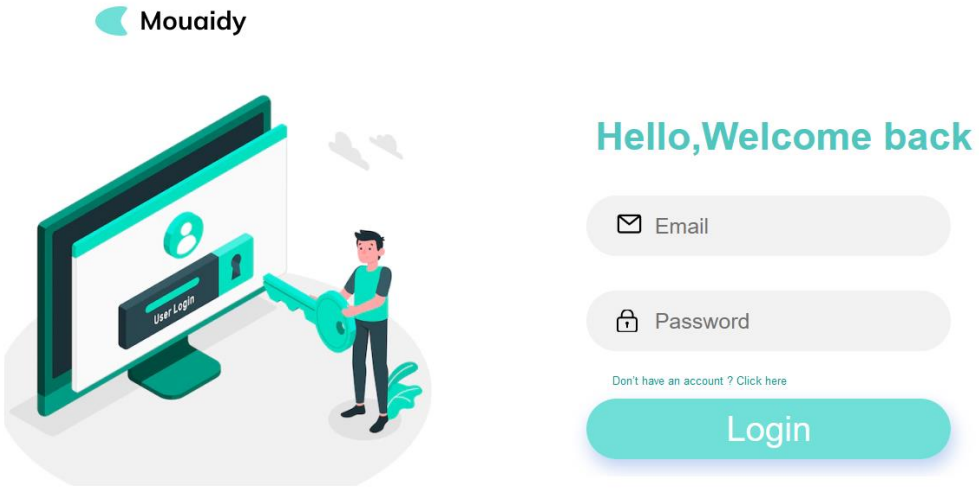


Figure 3: Login page.

➢ **Sign up page (doctor side)**

The following interface shows the sign up form of the doctor who can enter his login information, personal information and professional information.

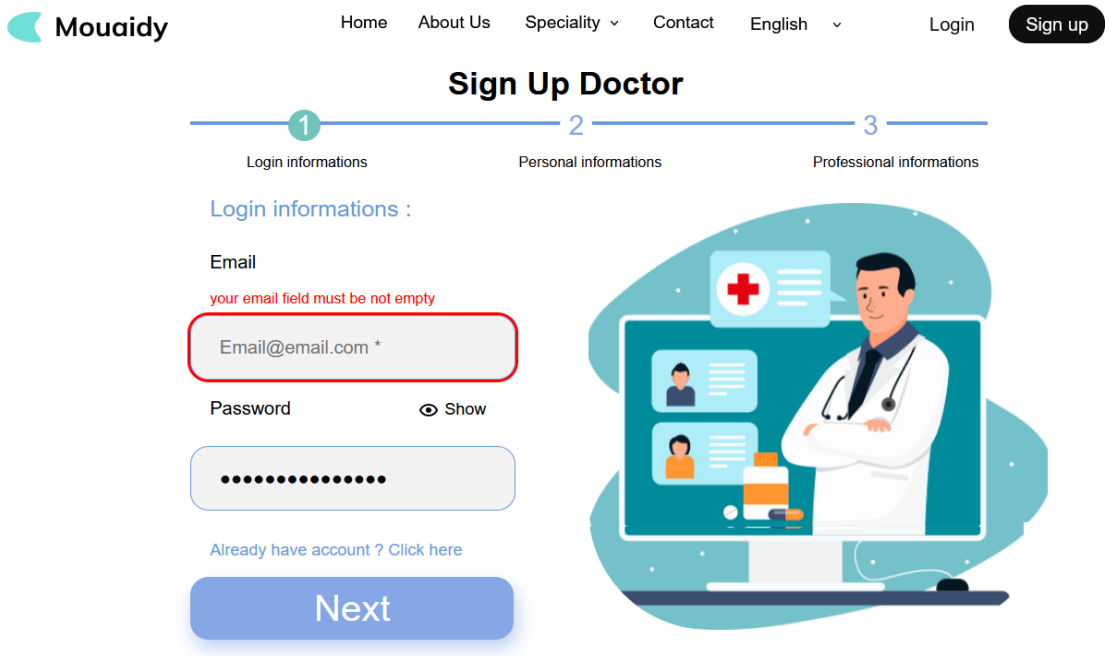


Figure 4: Sign up doctor page.

➢ **Dashboard page**

In this page, the doctor can consult his daily reports of appointments.

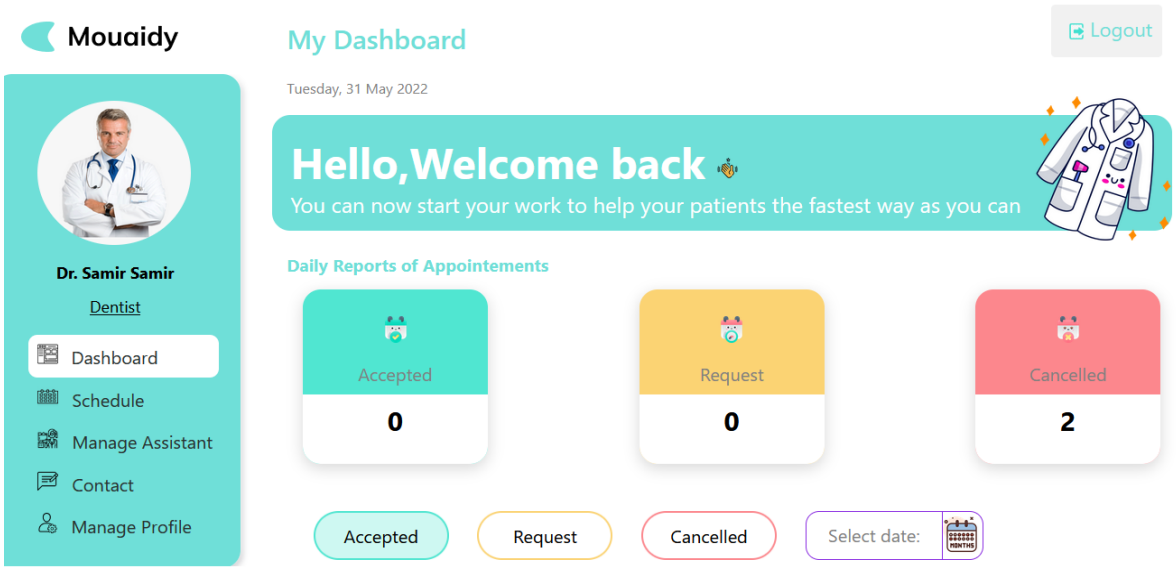


Figure 5: Dashboard page of the doctor.

➢ Manage schedule page

In this interface, the doctor can add, edit, delete schedule after selecting the date of schedule.

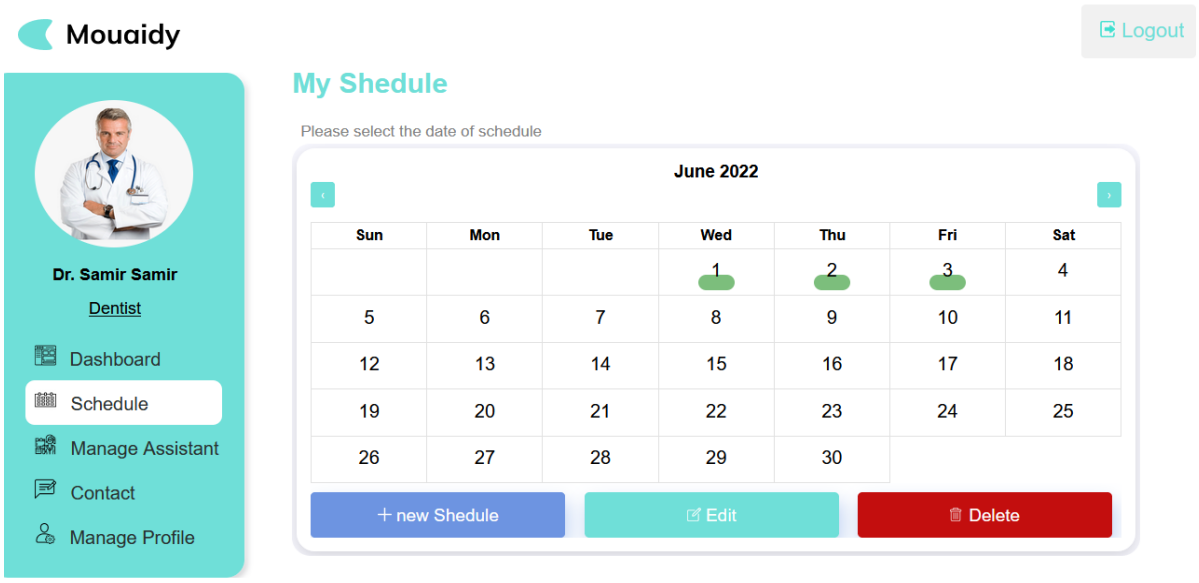


Figure 6: Manage schedule page.

➢ Add new schedule page

The following page shows add new schedule page, where the doctor must enter his start time, end time and the max number of patients.

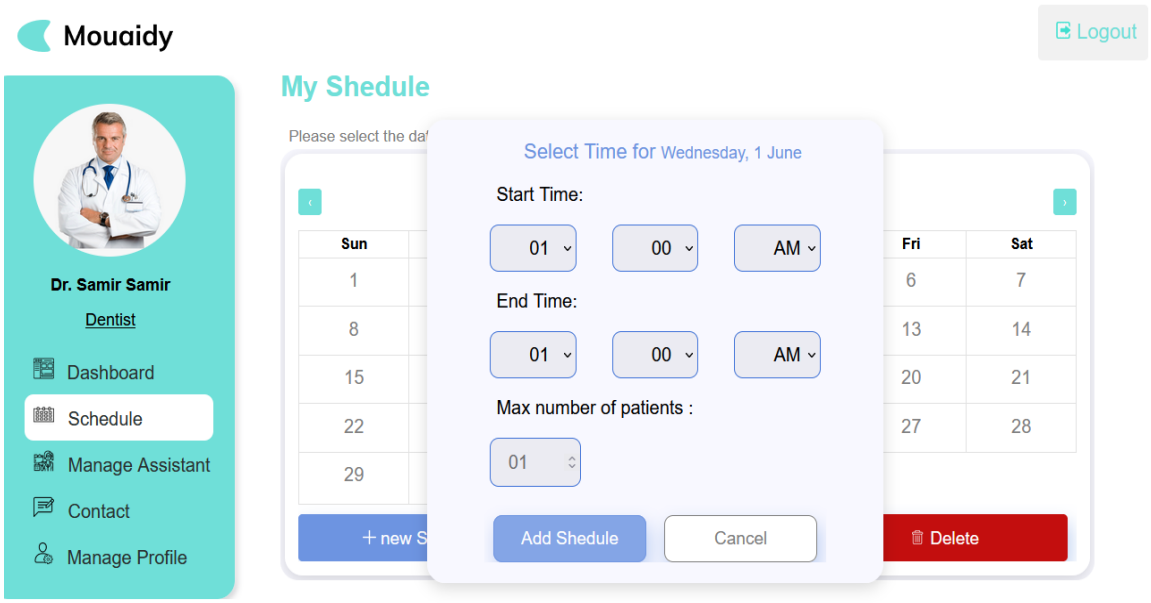


Figure 7: Add new schedule page.

➢ Manage assistant page

In this page, the doctor can consult the list of his assistants, search, edit or delete an existing assistant and add new assistant.

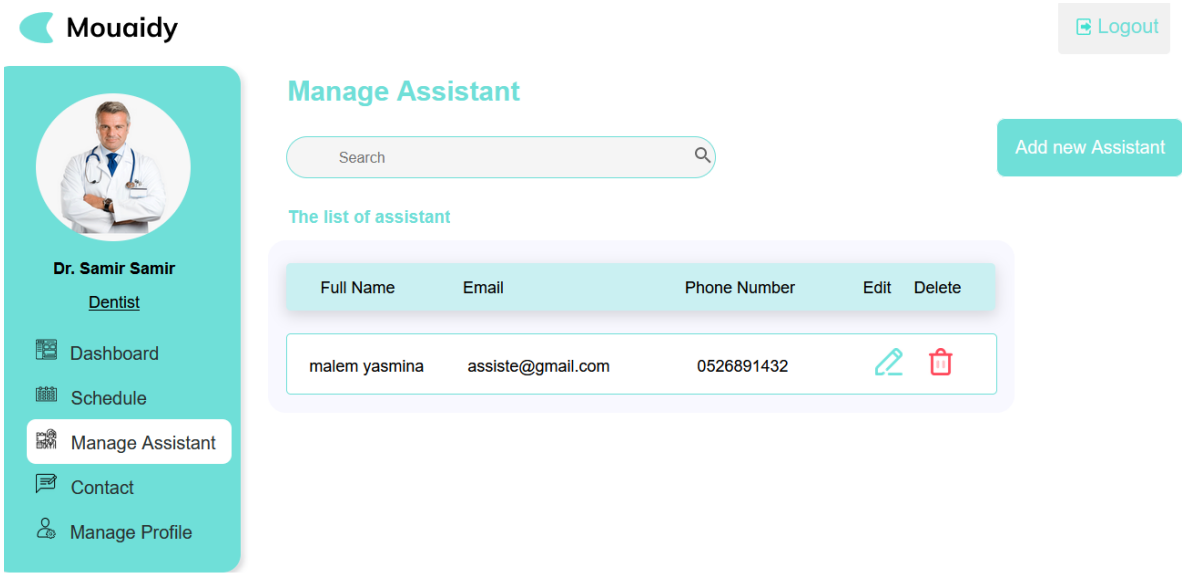


Figure 8: Manage assistant page.

➢ Add new assistant page

The following page will display to the doctor if he clicks on add new assistant button.

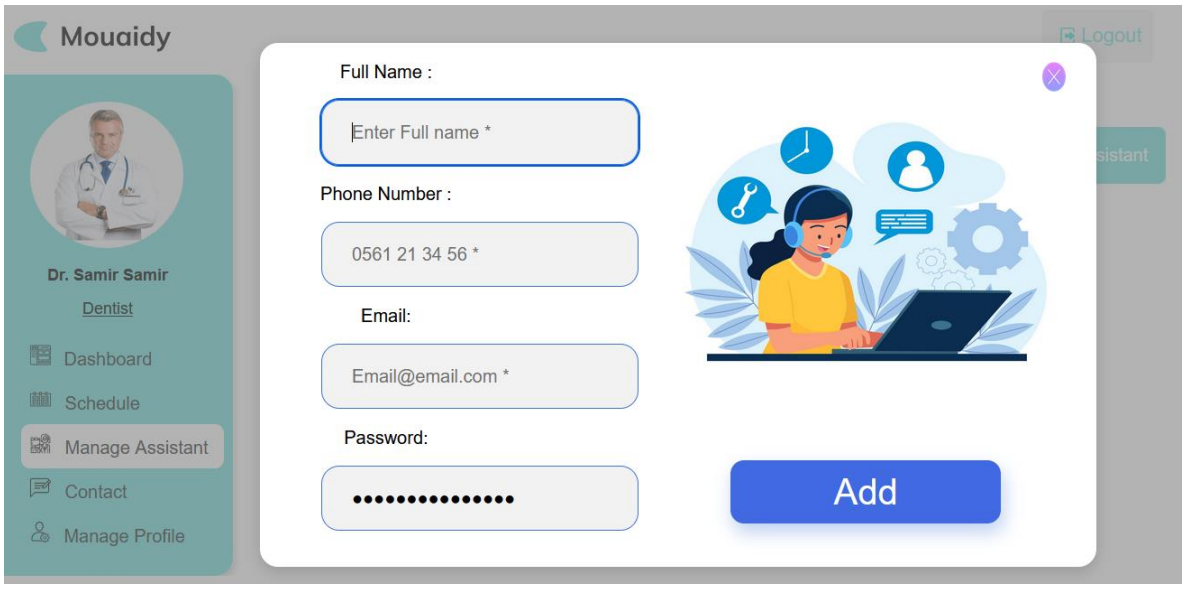


Figure 9: Add new assistant page.

➢ **Sign up page (patient side)**

The next interface shows the sign up form of the patient who can enter his login information and personal information.

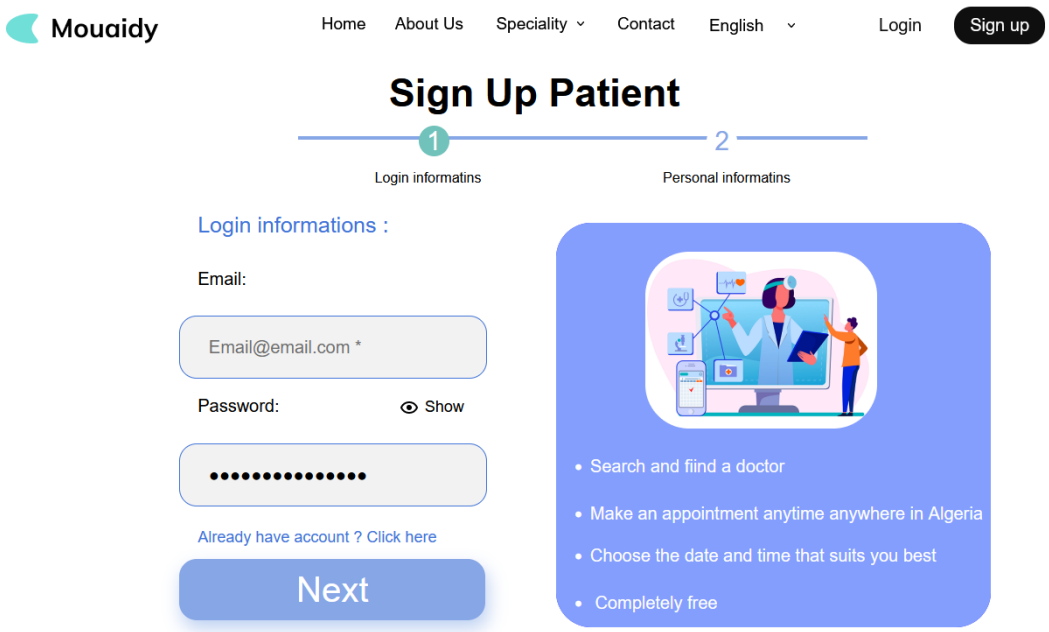


Figure 10: Sign up patient page.

➢ **Search doctor page**

After connecting to his account, here the patient can search doctor by selecting his Wilaya, specialty and entering his name.

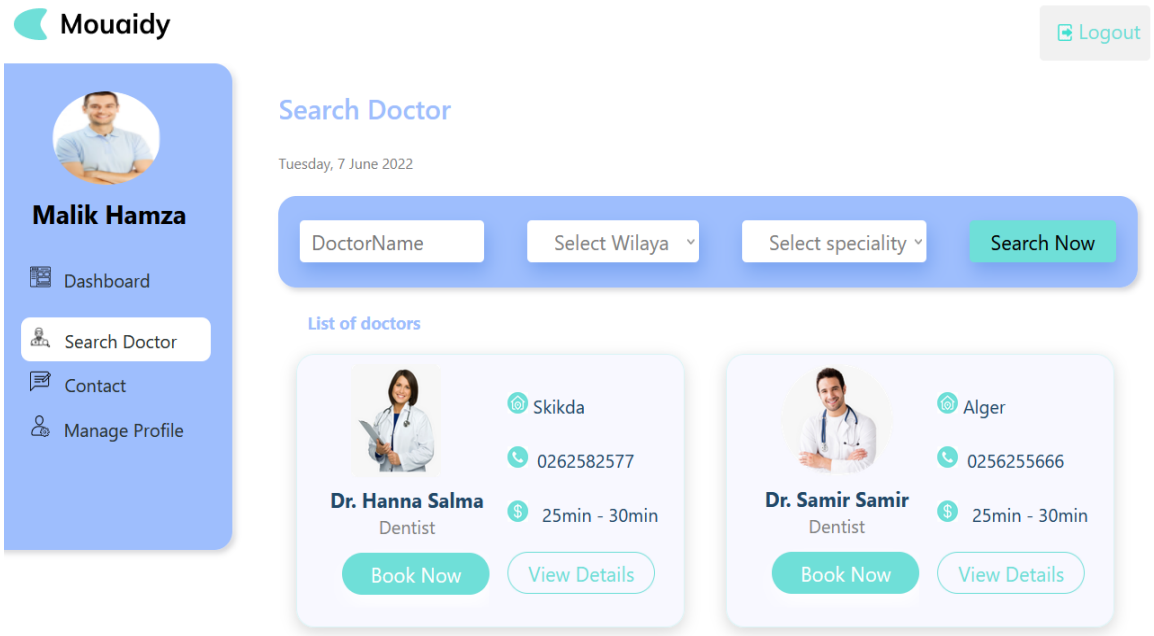


Figure 11: Search doctor page.

➢ Book appointment page

The following interfaces show how the patient connected to our application can book his appointment after finding his doctor.

The figure 12: represent the manage appointment page, here the patient can edit, delete and consult his appointments, he can also make new appointment.

The figure 13, 14, 15: represent the steps to book an appointment; the first step is to select a date that must be scheduled by the doctor or his assistant, the second step is to choose the hour of appointment and the last step is to enter the personal information.

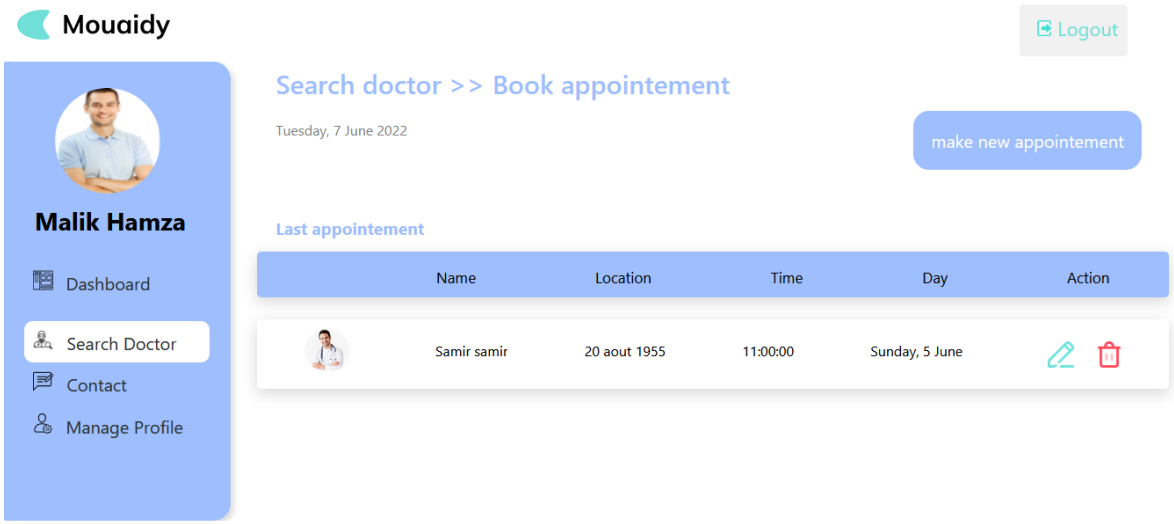


Figure 12: Manage appointment page.

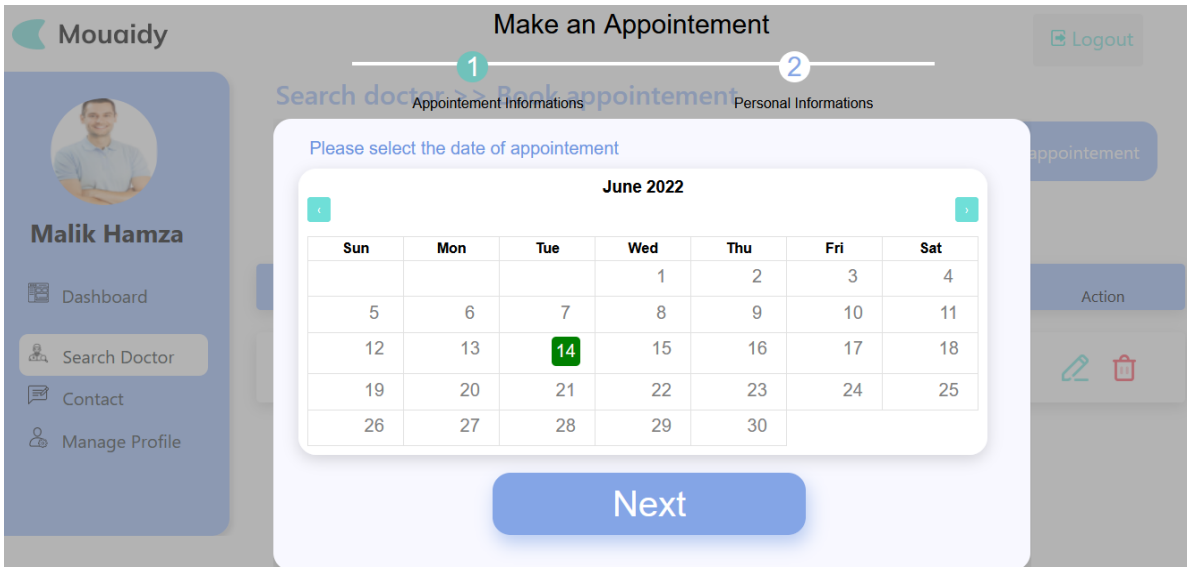


Figure 13: First step to book an appointment.

Figure 14: Second step to book an appointment.

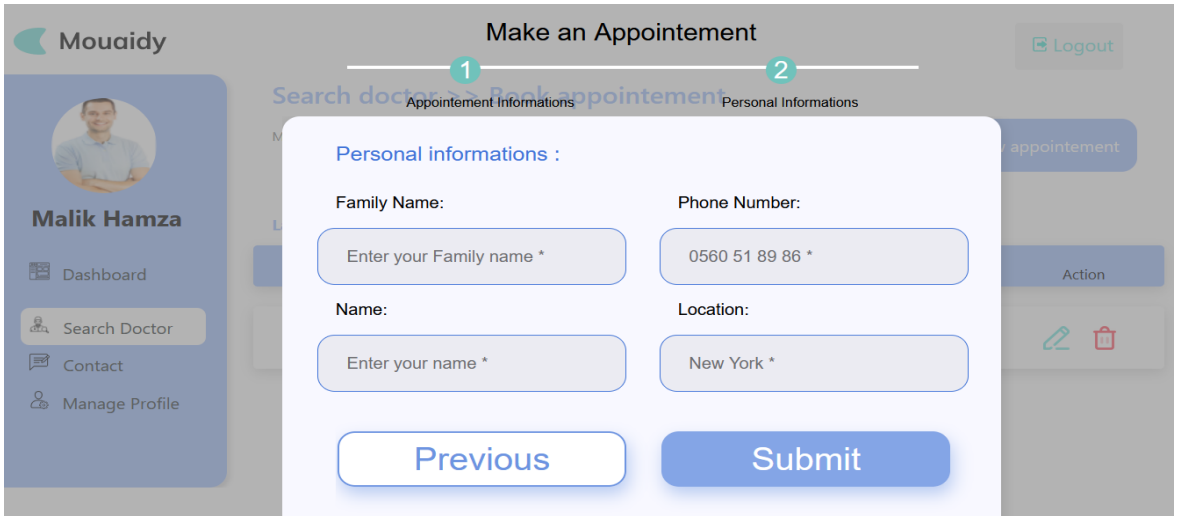


Figure 15: Third step to book an appointment.

➢ Dashboard page (assistant side)

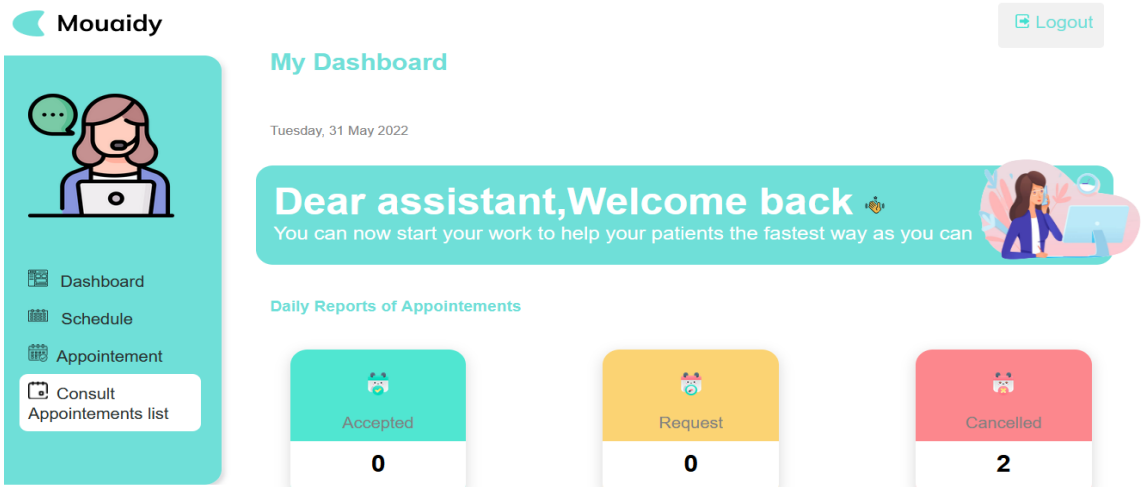


Figure 16: Assistant's dashboard page.

➢ Request appointment page

In this page, the assistant can consult the request appointment list and then decide to accept or consult the appointment.

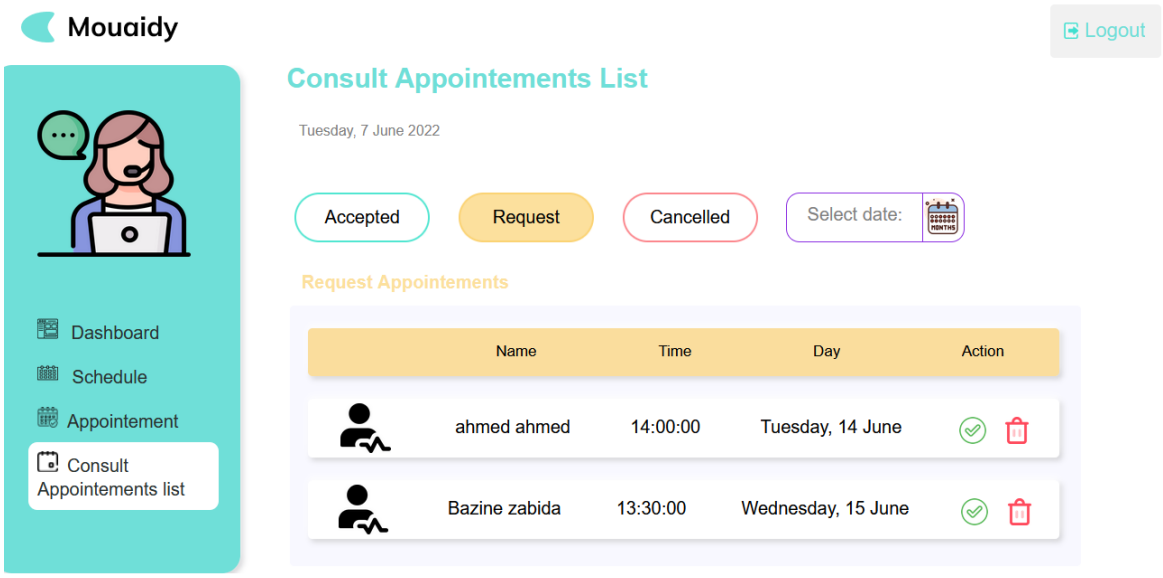


Figure 17: Request appointment page.

➢ Accepted appointment page

In the next page, the assistant can consult the accepted appointment list.

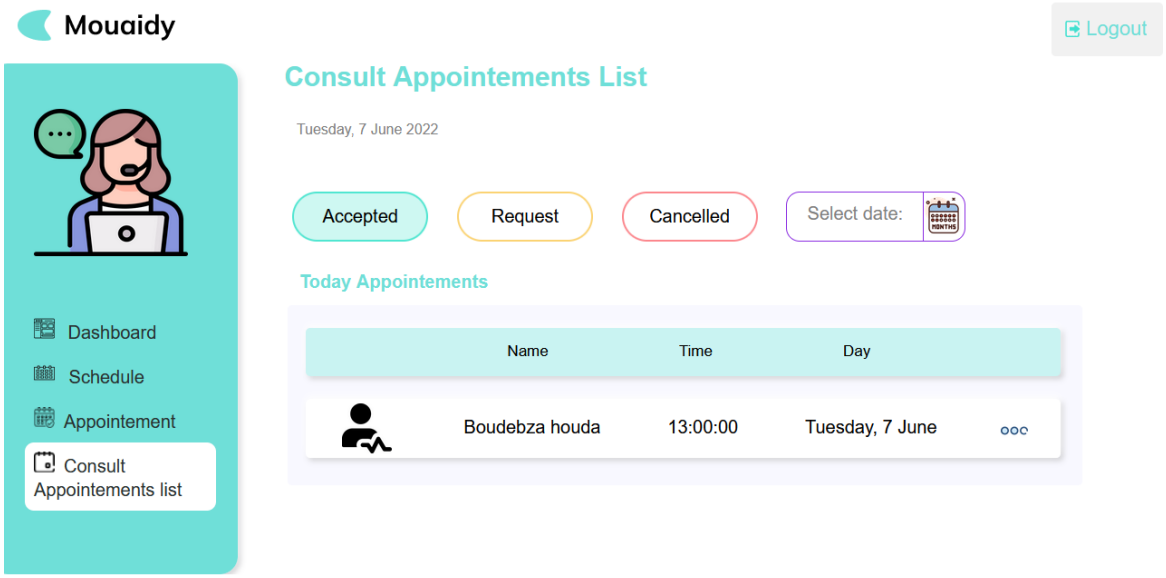


Figure 18: Accepted appointment page.

➢ **Login page (administrator side)**

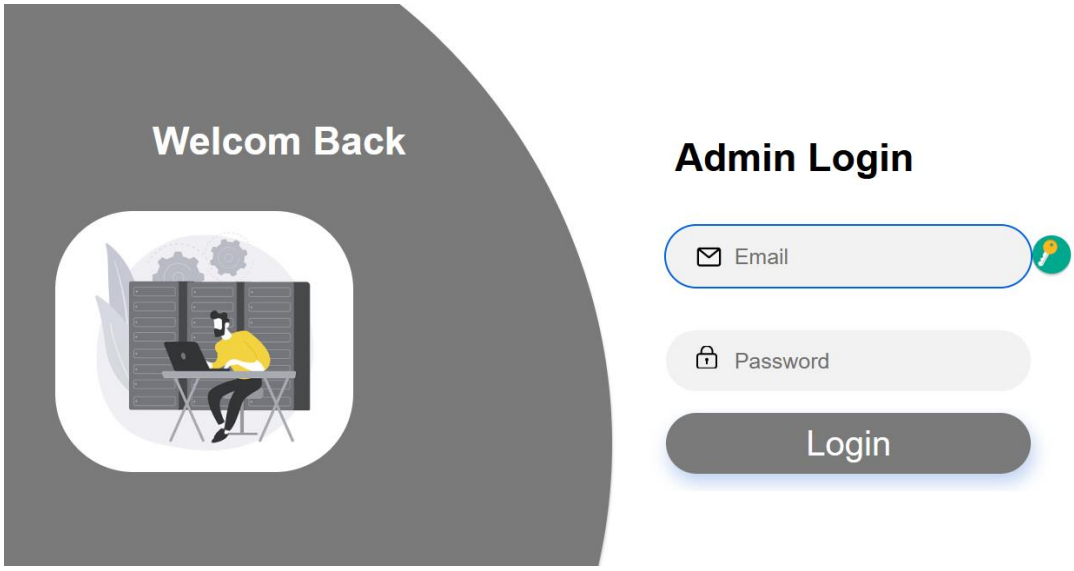


Figure 19: Login page of the administrator.

➢ **Dashboard page**

In this page the administrator can consult his daily reports.

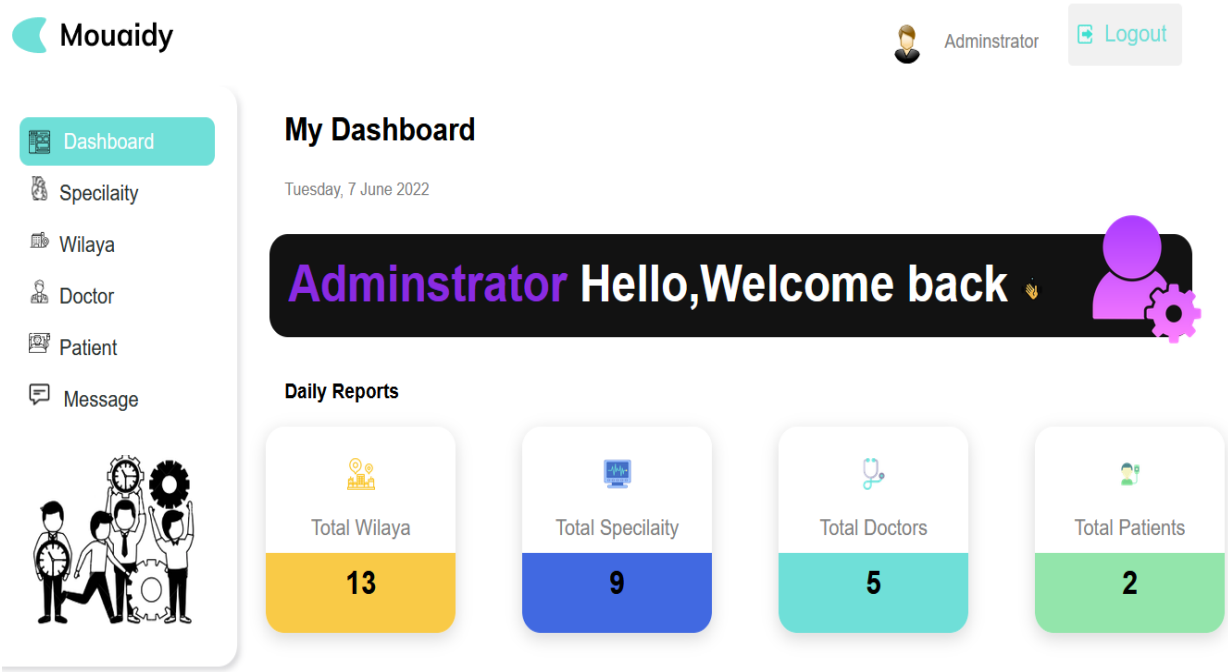


Figure 20: Dashboard page of the administrator.

➢ Manage Specialty page

In this page, the administrator can consult the list of specialties, edit, delete and add new specialty.

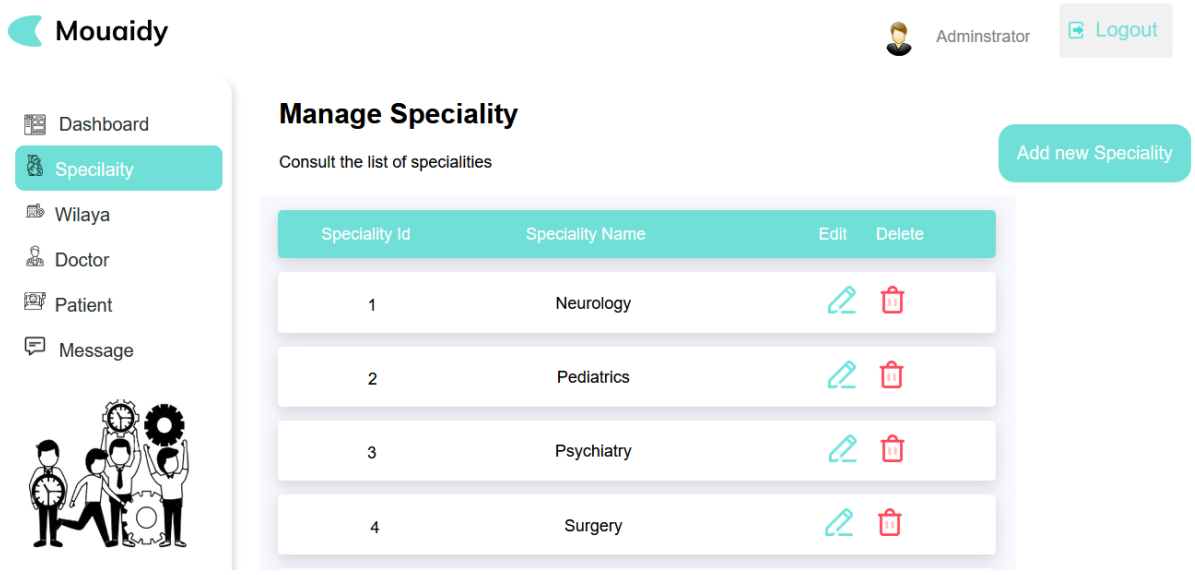


Figure 21: Manage Specialty page.

➢ Add new Specialty page

The following page will display to the administrator, if he clicks on add new specialty button.

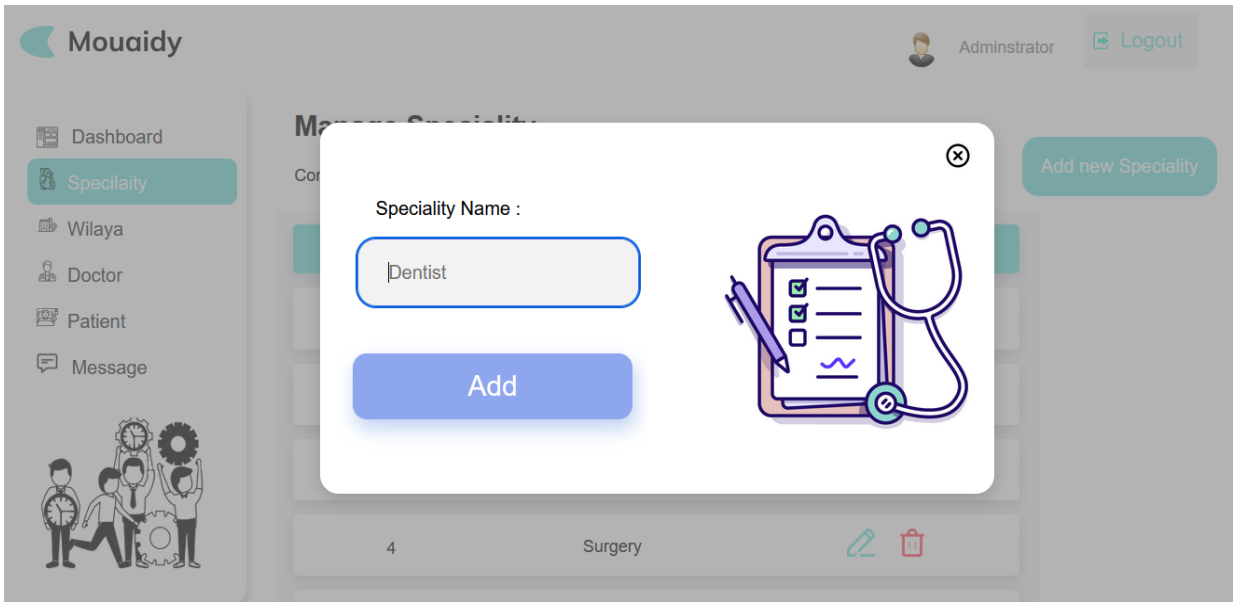


Figure 22: Add new Specialty page.

➢ Manage Wilaya page

In following page, the administrator can consult the list of existing Wilayas, edit, delete and add new Wilaya.

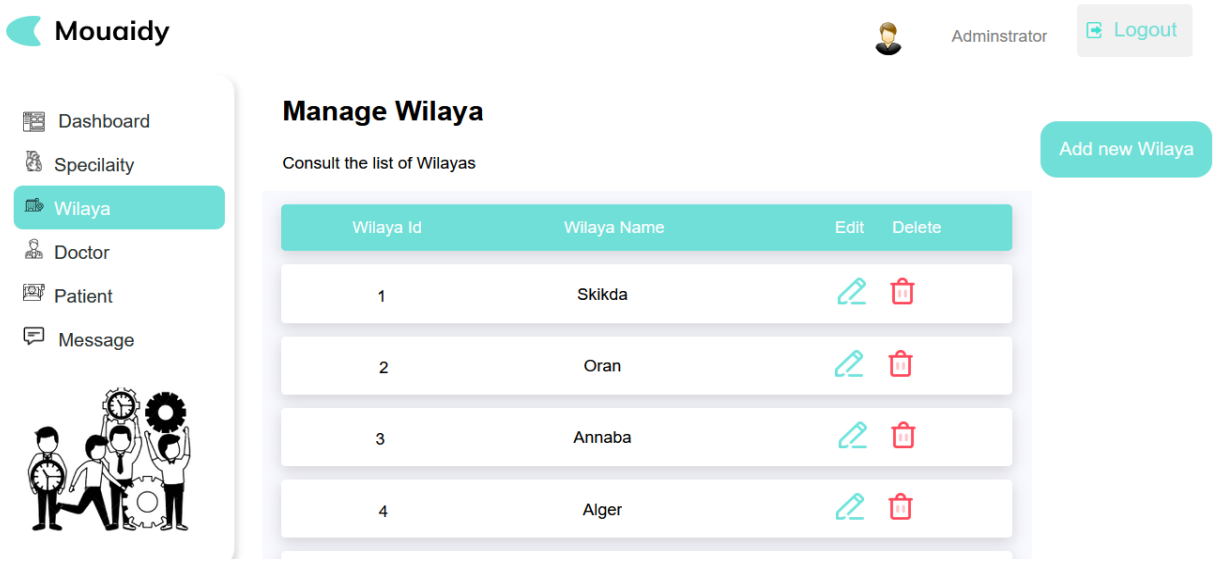


Figure 23: Manage Wilaya page.

➢ Add new Wilaya page

After clicking on Add new Wilaya, here the administrator can enter the name of new Wilaya.

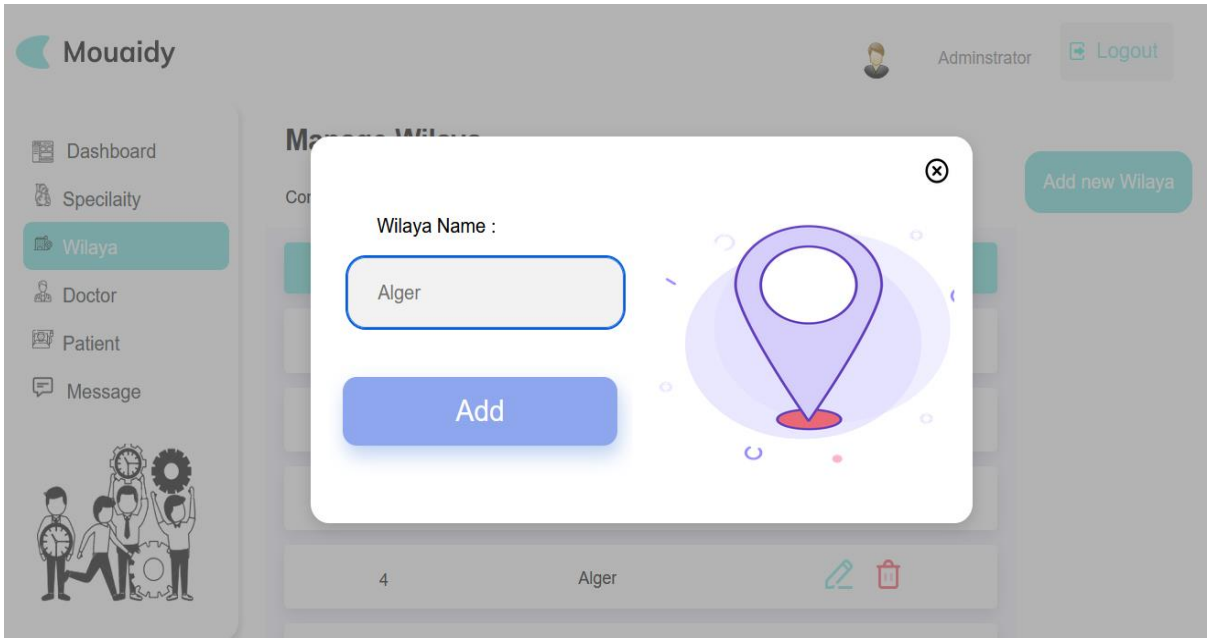


Figure 24: Add new Wilaya page.

➢ Invalidate doctor page

In this page, the administrator invalidates doctors' accounts after verifying their profiles, as they are not real doctors.

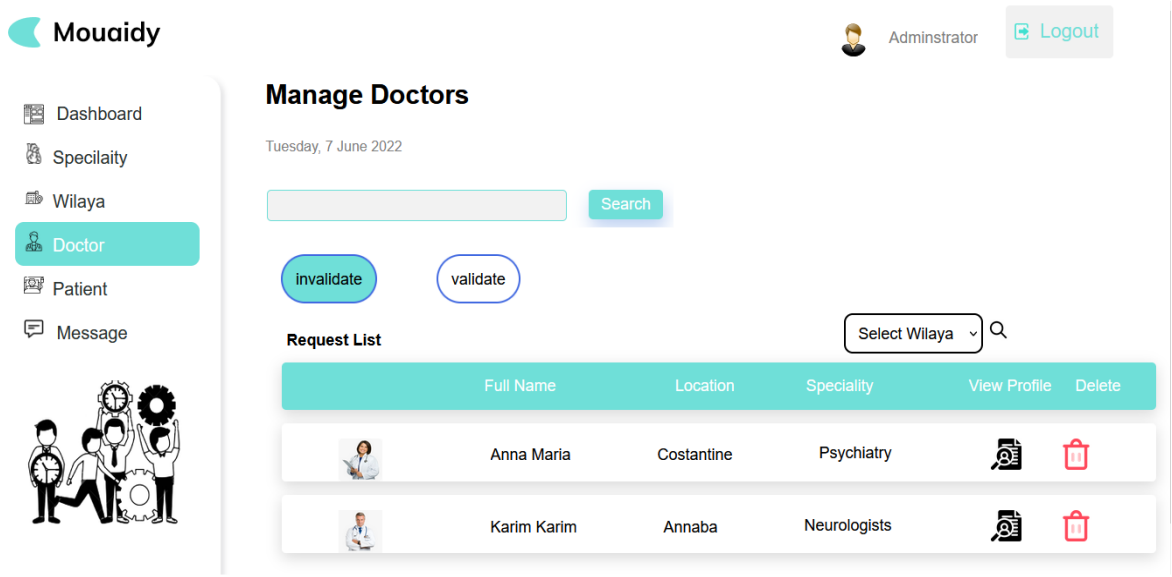


Figure 25: Invalidate doctor page.

➢ Validate doctor page

In following interface, the administrator can validate doctors' accounts, after verifying their profiles and CVs.

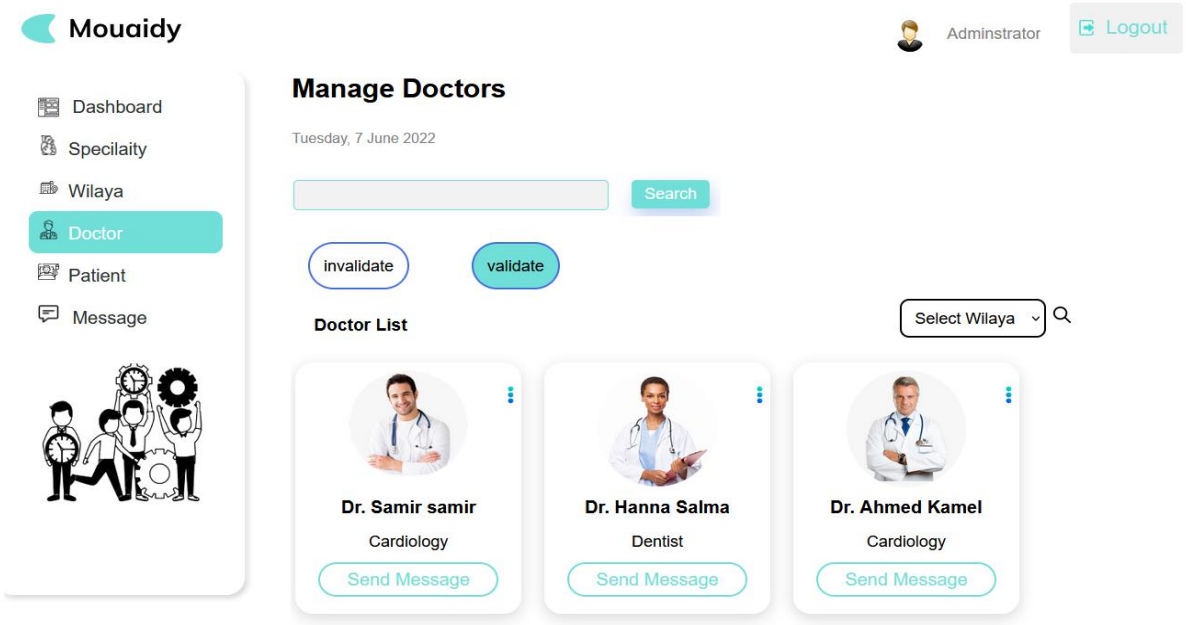


Figure 26: Validate doctor page.

4.5 Conclusion

In this chapter we presented the implementation of our online appointment application and recommendation systems. We have also presented the database and the development environment used with some screenshots explaining the flow and functioning of our work and the results obtained.

General Conclusion

Summary of contributions

In this work, we have addressed the problem of the improvement of the interactivity of the Mouaidy online appointment application by designing and developing an easy, modern and attractive online appointment application. We have proposed several contributions to integrate this knowledge:

- First, we gave a brief overview about the online appointment system, its role, as well as its advantages and limitations. The established after evaluating the Mouaidy application, a set of weak points, which we fixed after in our work.
- Secondly, we familiarized ourselves with Front End techniques that played an important role in the design and Development process. Moreover, appealing user interface designs and developments can ensure better results, help you generate more traffic, and keep your visitors engaged and retained.
- Thirdly, for a clear and explanatory design, we used the UML modeling language and the modern modeling technique witch named the website user experience flowcharts mainly to plot how users move through an interface.
- Lastly the implementation, the web application was developed using different computer software such as Wampserver, JQuery, JavaScript, HTML language, CSS, and the implementation of databases with the MYSQL database manager then the implementation of SQL queries for the manipulation of the data. Then for the interfaces we used Figma and finally the realization of the application under the visual studio programming environment.

In fact, at the end of the realization of this dissertation, we have accumulated an important mass of knowledge as well on the theoretical level as on the practical level, and we estimate that it will be very useful to us in the future, in our further studies.

Perspectives

As further improvements, we could add a lot some or all of these features in the future:

- Online consultations: until now users must come to clinic to be examined. If users show interest in online consultations that don’t require physical examinations, we could add this feature.
- Online payments: until now all the payments are made when a user comes to clinic. If we integrate online consultations or users will show more interest in paying online instead than in cash at clinic, we could integrate this feature. Joomla allows this kind of feature.
- Security system.

List of abbreviations

OAS	Online Appointment System
SRS	Software Requirements Specifications
RS	Requirements Specifications
UX	User Experience
UI	User Interface
UCD	User Centre Design
UID	User Interface Design
CTA	Call To Action
UML	Unified Modeling Language
CV	Curriculum Vitae
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
JS	Java Script
SQL	Structured Query Language

Bibliography

[1] Kaandorp, G. C., & Koole, G. (2007). Optimal outpatient appointment scheduling. Health care management science, 10(3), 217-229.

[2] Bailey, N. T. (1952). A study of queues and appointment systems in hospital out-patient departments, with special reference to waiting-times. Journal of the Royal Statistical Society: Series B (Methodological), 14(2), 185-199.

[3] Bailey, N. T. (1954). Queueing for medical care. Journal of the Royal Statistical Society: Series C (Applied Statistics), 3(3), 137-145.

[4] Cayirli, T., Veral, E., & Rosen, H. (2004). Assessment of patient classification in appointment systems. In 1st Conference of the POMS College of Service Operations.

[5] Cayirli, T., Veral, E., & Rosen, H. (2006). Designing appointment scheduling systems for ambulatory care services. Health care management science, 9(1), 47-58.

[6] Davidson, J. D., & Coward, D. (1999). Java servlet specification (" specification") version: 2.2 final release. Sun Microsystems, 43-46.

[7] Bailey, N. T. (1954). Queueing for medical care. Journal of the Royal Statistical Society: Series C (Applied Statistics), 3(3), 137-145.

[8] Ralston, J. D., Martin, D. P., Anderson, M. L., Fishman, P. A., Conrad, D. A., Larson, E. B., & Grembowski, D. (2009). Group health cooperative's transformation toward patient-centered access. Medical Care Research and Review, 66(6), 703-724.

[9] Denizard-Thompson, N. M., Feiereisel, K. B., Stevens, S. F., Miller, D. P., & Wofford, J. L. (2011). The digital divide at an urban community health center: implications for quality improvement and health care access. Journal of community health, 36(3), 456-460.

[10] Dexter, F., Xiao, Y., Dow, A. J., Strader, M. M., Ho, D., & Wachtel, R. E. (2007). Coordination of appointments for anesthesia care outside of operating rooms using an enterprise-wide scheduling system. Anesthesia & Analgesia, 105(6), 1701-1710.

[11] Siddiqui, Z., & Rashid, R. (2013). Cancellations and patient access to physicians: ZocDoc and the evolution of e-medicine. Dermatology online journal, 19(4).

[12] Zhao, P., Yoo, I., Lavoie, J., Lavoie, B. J., & Simoes, E. (2017). Web-based medical appointment systems: a systematic review. Journal of medical Internet research, 19(4), e6747.

[13] Jones, R., Menon-Johansson, A., Waters, A. M., & Sullivan, A. K. (2010). eTriage–a novel, web-based triage and booking service: enabling timely access to sexual health clinics. International journal of STD & AIDS, 21(1), 30-33.

[14] Lowes, R. (2006). Let patients book their own appointments?. Medical Economics, 83(11), 27-28.

[15] Parmar, V., Large, A., Madden, C., & Das, V. (2009). The online outpatient booking system'Choose and Book'improves attendance rates at an audiology clinic: a comparative audit. Journal of Innovation in Health Informatics, 17(3), 183-186.

[16] Walters, B. A., & Danis, K. (2003). Patient Online at Dartmouth-Hitchcock–Interactive Patient Care Web Site. In AMIA Annual Symposium Proceedings (Vol. 2003, p. 1044). American Medical Informatics Association.

[17] Riddell, S. (2012). Technology: the race to get the NHS online. The Health service journal, 122(6319), 32.

[18] Prakash, B. (2010). Patient satisfaction. Journal of cutaneous and aesthetic surgery, 3(3), 151.

[19] Camacho, F., Anderson, R., Safrit, A., Jones, A. S., & Hoffmann, P. (2006). The relationship between patient's perceived waiting time and office-based practice satisfaction. North Carolina medical journal, 67(6), 409-413.

[20] Zhao, P., Yoo, I., Lavoie, J., Lavoie, B. J., & Simoes, E. (2017). Web-based medical appointment systems: a systematic review. Journal of medical Internet research, 19(4), e6747.

[21] Zhang, X., Yu, P., & Yan, J. (2014, January). Patients' adoption of the e-appointment scheduling service: A case study in primary healthcare. In HIC (pp. 176-181).

[22] Craig, A. (2007). Self-service appointments. Advance for nurse practitioners, 15(10), 24-25.

[23] Sampson, F., Pickin, M., O'Cathain, A., Goodall, S., & Salisbury, C. (2008). Impact of same-day appointments on patient satisfaction with general practice appointment systems. British Journal of General Practice, 58(554), 641-643.

[36] Smaradottir, B. F. (2016, December). The steps of user-centered design in health information technology development: recommendations from a PhD research study. In 2016 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 116-121). IEEE.

[37] Williams, A. (2009, October). User-centered design, activity-centered design, and goal-directed design: a review of three methods for designing web applications. In Proceedings of the 27th ACM international conference on Design of communication (pp. 1-8).

[41] Bevan, N. (2009, August). Criteria for Selecting Methods in User Centred Design. In I-USED.

[42] Goodman-Deane, J., Langdon, P., & Clarkson, J. (2010). Key influences on the user-centred design process. Journal of Engineering Design, 21(2-3), 345-373.

[57] Hennicker, R., & Koch, N. (2001). Systematic Design of Web Applications with UML. Teoksessa: Siau K. & Halpin T. (toim.) Unified Modeling Language: Systems Analysis, Design and Development Issues.

[58] Pilone, D., & Pitman, N. (2005). UML 2.0 in a Nutshell. " O'Reilly Media, Inc.".

[59] Travassos, G. H., Shull, F., & Carver, J. (2002). Working with UML: A software design process based on inspections for the unified modeling language. In Advances in Computers (Vol. 54, pp. 35-98). Elsevier.

[63] Klaib, A. (2005). Data Models and the Dataflow Algebra within WinCASE (Doctoral dissertation, MSc Dissertation 2004/2005, Sheffield University: Sheffield University).

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

[65] Podeswa, H. (2009). UML for the IT Business Analyst. Course Technology Press.

[66] Starr, L. (2001). Executable UML: how to build class models. Prentice Hall PTR.

Webography

[24] "Limitations / Drawbacks of Online Appointment Scheduling" (Accessed Apr. 07, 2022). https://www.flexbooker.com/blog/3-drawbacks-online-appointment-scheduling/

[25] "DZDOC" Accessed: Apr 08, 2022. Online available on: https://dzdoc.com/apropos.php

[26] "SIHHATECK" Accessed: Apr 08, 2022. Online available on: https://sihhatech.com/a-propos

[27] "BEESIHA" Accessed: Apr 08, 2022. Online available on: https://www.beesiha.com/fr/about-us

[28] "User interface" Accessed: Apr 09, 2022. [Online] available on: https://www.hexacta.com/how-ui-design-benefits-your-business/

[29] "Front-end-development" Accessed: Apr 11, 2022. [Online] available https://scand.com/company/blog/front-end-development-steps/

[30] "Front-end" Accessed: Apr 11, 2022. [Online] available https://www.frontend-gmbh.de/blog/frontend-und-backend/

[31] "User-centered-design" Accessed: Apr 15, 2022. [Online] available on: https://www.wowmakers.com/blog/beginners-guide-to-user-centered-design/

[32] "Human centered design" Accessed: Apr 15, 2022. [Online] available on: https://xd.adobe.com/ideas/principles/human-computer-interaction/user-centered-design/

[33] "user-centered-design process" Accessed: Apr 22, 2022. [Online] available on: https://www.cuelogic.com/blog/user-centered-design

[34] "User-centered-design stage" Accessed: Apr 22, 2022. [Online] available on: https://www.chaione.com/blog/5-stages-user-centric-design-thinking

[35] "User-interface-design-development" Accessed: May 10, 2022. [Online] available on: https://bitsens.com/blog/guide-on-user-interface-design-development.html

[38] "user-centered-design principals" Accessed: May 13, 2022. [Online] available on: https://opensenselabs.com/blog/articles/user-centered-design-approach-core-principles-methods

[39] "User experience process" Accessed: May 13, 2022. [Online] available on: https://xd.adobe.com/ideas/guides/ux-design-process-steps/

[40] "Step of user interface development" Accessed: May 25, 2022. [Online] available on: https://scand.com/company/blog/front-end-development-steps/

[43] "Ucd-usability" Accessed: May 25, 2022. [Online] available on: https://blogs.uoregon.edu/uxuo/2013/09/20/ucd-ux-usability-so-whats-the-difference/

[44] "User Control" Accessed: May 25, 2022. [Online] available on: https://xd.adobe.com/ideas/process/ui-design/4-golden-rules-ui-design/

[45] "Usability" Accessed: May 27, 2022. [Online] available on: https://www.justinmind.com/ui-design/principles

[46] "Web development" Accessed: June 05, 2022. [Online] available on: https://www.ionos.com/digitalguide/websites/web-development/user-centered-design/

[47] "Design thinking" Accessed: June 05, 2022. [Online] available on: https://measuringu.com/ucd-and-design-thinking/

[48] "Design process" Accessed: June 06, 2022. [Online] available on: https://www.digitalstockport.info/placing-users-at-the-heart-of-the-councils-design-process/

[49] "User experience process" Accessed: June 06, 2022. [Online] available on: https://www.oreilly.com/library/view/ux-for-the/9781787128477/389ca09f-0284-4916-9a47-d95174acf622.xhtml

[50] "User centered design concept" Accessed: June 06, 2022. [Online] available on: https://www.enlighten.co.nz/Blog/2020/03/Mapping-Your-User-Centred-Design-Process

[51] "Requirements Specification and Analysis" Accessed: June 07, 2022. Online available on: https://techwhirl.com/writing-software-requirements-specifications/

[52] "Requirements Specification" Accessed: June 09, 2022. Online available on: https://www.techtarget.com/searchsoftwarequality/definition/requirements-analysis

[53] "Function vs nonfunctional" Accessed: June 10, 2022. Online available on: https://www.javatpoint.com/functional-vs-non-functional-requirements

[54] "Functional and nonfunctional requirements" Accessed: June 10, 2022. Online available on: https://enkonix.com/blog/functional-requirements-vs-non-functional/#what-are

[55] "Requirements Analysis" Accessed: June 10, 2022. Online available on: https://www.techtarget.com/searchsoftwarequality/definition/requirements-analysis

[56] "Use Cases" Accessed: June 10, 2022. Online available on: https://www.techtarget.com/searchsoftwarequality/definition/use-case

[60] "What is UML" Accessed: June 13, 2022. Online available on: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/

[61] “Type of UML diagrams” Accessed: June 13, 2022. Online available on: https://creately.com/blog/diagrams/uml-diagram-types-examples/

[62] “Use Case diagram” Accessed: June 13, 2022. Online available on: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

[67] “Website UX flowcharts” Accessed: June 14, 2022. Online available on: https://www.interaction-design.org/literature/topics/flowcharts

[68] “Design a UX website flowchart” Accessed: June 14, 2022. Online available on: https://xd.adobe.com/ideas/process/prototyping/creating-ux-flowcharts/

[69] “Star UML” Accessed: June 14, 2022. Online available on: https://www.methodsandtools.com/tools/staruml.php

[70] “Figma” Accessed: June 15, 2022. Online available on: https://alternatives.app/software/Figma

[71] “Web Application Architecture” Accessed: June 15, 2022. Online available on: https://litslink.com/blog/web-application-architecture

[72] “Visual studio” Accessed: June 18, 2022. Online available on: https://code.visualstudio.com/docs

[73] “WampServer” Accessed: June 18, 2022. Online available on: https://sourceforge.net/projects/wampserver/

[74] “Html” Accessed: June 19, 2022. Online available on https://www.tutorialspoint.com/html/index.htm

[75] “Css” Accessed: June 19, 2022. Online available on: https://www.tutorialspoint.com/css/what_is_css.htm

[76] “Js” Accessed: June 19, 2022. Online available on: https://www.javatpoint.com/javascript-tutorial

[77] “JQuery” Accessed: June 19, 2022. Online available on: https://www.devopsschool.com/blog/what-is-jquery-a-brief-look/

[78] “Bootstrap” Accessed: June 22, 2022. Online available on: https://www.w3schools.com/bootstrap/bootstrap_get_started.asp

[79] “Php” Accessed: June 22, 2022. Online available on: https://www.php.net/

[80] “MySQL” Accessed: June 22, 2022. Online available on: https://www.javatpoint.com/mysql-tutorial