

REPUBLIC OF CAMEROON

Peace-Work-Fatherland

REPUBLIQUE DU CAMEROON

Paix-Travail- Patrie



**AFRICAN INSTITUTE OF COMPUTER
SCIENCES - CAMEROON PAUL BIYA
TECHNOLOGICAL CENTER OF
EXCELLENCE**

P.O. Box: 13719 Yaoundé

Tel: +237.242.729.957 Fax: 22729958

E-mail: contact@iaicameroun.com

**INSTITUT AFRICAIN DE
L'INFORMATIQUE – CAMEROUN
CENTRE TECHNOLOGIQUE DE
L'EXCELLENCE PAUL BIYA**

B.P: 13719 Yaoundé

Tel: +237.242.729.957 Fax: 22729958

Courriel: contact@iaicameroun.com

PERSONALIZED PROJECT REPORT

DEVELOPMENT OF A PATIENTS' CONSULTATION AND FOLLOW-UP PLATFORM FOR CAMEROON GOVERNMENT HOSPITALS

Case Study: ODZA District Hospital

Option: Software Engineering

Written By:

KAMENI SEPDEU ANGE CHRIS

Level III student at AICS Cameroon

Supervised By:

Mrs. ONGUENE Vanessa

Academic Year 2023 - 2024



DEDICATION

This work is dedicated to the SEPNDEU's family for their numerous encouragement and support towards my academic success.



ACKNOWLEDGEMENT

Drafting this document would have not been possible without the contribution of some people who took upon themselves to see this work being accomplished. Our gratitude goes to the following people:

- ✚ The Resident Representative of IAI-Cameroon, Mr. Armand Claude ABANDA, for his support, words of encouragement and the different advices on how to approach situations.
- ✚ Our supervisor Mrs. ONGUENE Vanessa for her moral support, effort and advices throughout the academic year.
- ✚ To our academic teachers Mr. AGBOR Anderson AND Mr. MESSIO for their advices and assistance in realizing this document.
- ✚ To all my teachers of AICS CAMEROON for the knowledge they impacted me with for me to be where I am today.
- ✚ For my brothers and sister for their love and support.
- ✚ The countless contributors of open-source programming community, for their great help in learning basic skills and detecting and solving bugs.
- ✚ To all my classmates for their collaborative work throughout the academic year.



Content

DEDICATION.....	i
ACKNOWLEDGEMENT	ii
Content	iii
LIST OF FIGURES.....	v
LIST OF TABLES.....	vii
GLOSSARY	viii
Abstract	ix
Résumé	x
General Introduction	1
PART I.....	2
EXISTING SYSTEM.....	2
PART II.....	12
SPECIFICATION BOOK	12
PART III	29
ANALYSIS BOOK	29
PART IV	65
CONCEPTION PHASE	65
PART V.....	85
REALIZATION PHASE	85
PART VI	96
TEST OF FUNCTIONALITIES	96
PART VII.....	103
INSTALLATION GUIDE AND USER GUIDE	103



DEVELOPMENT OF A PATIENTS' CONSULTATION AND FOLLOW-UP PLATFORM FOR CAMEROON GOVERNMENT HOSPITALS



GENERAL CONCLUSION.....	cxviii
BIBLIOGRAPHY	cxviii
WEBOGRAPHY	xiv
TABLE OF CONTENT.....	xv

LIST OF FIGURES

Figure 1: Doctor Survey Question 1	6
Figure 2: Doctor Survey Question 2	7
Figure 3: Doctor Survey Question 3	7
Figure 4: Patient Survey question 1	15
Figure 5: Patient Survey Question 2	15
Figure 6: Patient Survey Question 3	16
Figure 7: Patient Survey Question 4	16
Figure 8: Gantt project planning	22
Figure 9: UML 2.5 diagrams overview (source: https://creately.com/blog/diagrams/uml-diagram-types-examples/)	33
Figure 10: 2TUP diagram (source: https://www.mysciencework.com/omniscience/pervasive-mobile-healthcare-system-based-on-cloud-computing/).	36
Figure 11: Use Case Diagram formalism	39
Figure 12: General Use Case Diagram.....	43
Figure 13 consult appointment use case diagram.....	44
Figure 14: consultation use case diagram	45
Figure 15: Communication Diagram formalism	49
Figure 16: Authenticate Communication Diagram	51
Figure 17 Book appointment communication diagram.....	52
Figure 18: Formalism of sequence diagram	53
Figure 19: Authenticate Sequence Diagram.....	56
Figure 20 Book appointment sequence diagram	57
Figure 21: Formalism of activity diagram.....	58
Figure 22 Authenticate activity diagram	61
Figure 23: Consultation activity diagram.....	62
Figure 24 Book appointment activity diagram.....	63
Figure 25: Hardware diagram of the system	68
Figure 26: n-tier Architecture (Source: https://www.pinterest.es/pin/752241943987565648/).....	69



Figure 27: The MVC architecture (Source: https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/)	70
Figure 28: Formalism of Class Diagram	71
Figure 29: System class diagram.....	74
Figure 30: Formalism of state machine diagram.....	76
Figure 33 Account state machine diagram	79
Figure 32 Consultation state machine diagram	79
Figure 33 Appointment state machine diagram.....	80
Figure 34: Formalism of package diagram	81
Figure 35 iKare package diagram	83
Figure 36: Formalism of deployment diagram.....	88
Figure 37 iKare deployment diagram.....	90
Figure 38: Formalism of component diagram (Source: https://www.pinterest.com/pin/551128073157994549/)	91
Figure 39 iKare mobile component diagram.....	93
Figure 40 iKare mobile component diagram.....	94
Figure 41: Admin Modules tests	100
Figure 42: Professional Modules tests	101
Figure 43 mongodb logo	106
Figure 44 Downloading MongoDB community version.....	106
Figure 45 Launching the installation wizard.....	107
Figure 46 Clicking on next.....	107
Figure 47: Agreement License	108
Figure 48 Select complete setup.....	108
Figure 49: Run service network user.....	109
Figure 50: Selecting MongoDB compass and clicking next	109
Figure 51: Click on install to start installation	110
Figure 52: Click on finish to complete installation	110
Figure 53: iKare login Screen	111
Figure 54: iKare register screen	112
Figure 55: iKare patient's welcome screen.....	113
Figure 56: Book appointment screen	113
Figure 57: Patients' consultation screen	114

LIST OF TABLES

Table 1: Limitations, Consequence and proposed solution.....	9
Table 2: Chronogram of activities	21
Table 3: Software Resources (source : Mercurial 2022)	23
Table 4: Hardware resources (source: https://www.scribd.com/document/561202966/mercuriale-2022)	24
Table 5: Human Resources (source : Mercurial 2022).....	25
Table 6: Global Estimation.....	25
Table 7: List of participants.....	27
Table 8: Differences between UML and MERISE	34
Table 9: Use Case Diagram Component	40
Table 10: Actors of our System	42
Table 11: Authenticate Textual Description	46
Table 12: Book appointment textual description.....	47
Table 13: Video Consultation textual description	48
Table 14: Elements of a communication diagram	50
Table 15: Elements of sequence diagram	54
Table 16: Elements of activity diagram.....	59
Table 17: Elements of class diagram.....	72
Table 18: Elements of state machine diagram.....	77
Table 19: Elements of package diagram.....	82
Table 20: Elements of deployment diagram.....	89
Table 21: Elements of component diagram.....	92








GLOSSARY

- + 2TUP: Two Track Unified Process.
- + AICS: African Institute of Computer Sciences.
- + APK: Android Package Kit.
- + DBMS : Database Management System
- + GUI: Graphical User Interface
- + IDE: Integrated Development environment
- + IPA: iOS App Store Package.
- + JSON: JavaScript Object Notation
- + MVC: Model View Controller.
- + UML: Unified Modelling Language.

Abstract

Health is one of the most valuable assets that a human can possess. The recent arousal of a pandemic in 2020 has made the world in general and our country in particular discover their vulnerability regarding health. Also, the inequality in the distribution of resources across areas in the national territory in terms of distances (transport costs involved), number of doctors (low doctor to patient ratio) has made it difficult for patients in certain enclaved areas to receive the most effective consultation and follow-ups they need. This carelessness can often lead to more serious cases of illnesses or even death in some cases. For this reason, we decided to develop an application to facilitate the consultation and follow-up of patients. This application is known as **iKare**. **iKare** is a mobile and web application that enables registered patients to take in their vitals with the help of an IOT device, book for appointments for consultations through video calls, receive regular forms after been prescribed drugs for follow-ups, search for nearby pharmacies, chat with health professionals on a health topic and manage their electronic health records just to name a few. The health professionals on the other hand will conduct consultations through video calls, prescribe drugs at the end of every consultation, verify all forms answered by patients. This will help the effective consultation and diagnosis of patients. In order to accomplish this project, we used UML (Unified Modelling Language) as our modelling language coupled with 2TUP (2 Track Unified Process) in order to form a method.

Keywords:

-  Health
-  Pandemic
-  Consultation
-  iKare
-  Follow-up
-  Diagnosis
-  IOT device



Résumé

La santé est un atout précieux pour les individus. La pandémie de 2020 a révélé la vulnérabilité du monde, y compris notre pays, en matière de santé. De plus, l'inégalité dans la répartition des ressources et des médecins rend difficile l'accès aux soins dans certaines zones isolées. Pour remédier à cela, nous avons développé l'application web et mobile iKare. Cette application permet aux patients enregistrés de surveiller leurs paramètres de santé à l'aide d'un appareil IOT, de prendre des rendez-vous en ligne pour des consultations vidéo, de recevoir des formulaires de suivi, de trouver des pharmacies à proximité et de discuter avec des professionnels de la santé. Les professionnels de la santé utilisent également l'application pour mener des consultations vidéo, prescrire des médicaments et vérifier les formulaires remplis par les patients. Pour mener à bien ce projet, nous avons utilisé UML et 2TUP comme méthodologie.

Mots-clés

- ✚ Santé
- ✚ Pandémie
- ✚ Consultation
- ✚ iKare
- ✚ Suivi
- ✚ Appareil IOT



General Introduction

Technology is advancing at an incredible rate, spanning almost every sector from business to agriculture and other sectors. Due to the huge technological advancement, we now live in a fast-paced society where everything is gradually being digitalized making life easier. As a developing country, most businesses in Cameroon strive to make advantage of the digital economy to grow their businesses thereby boosting the economy as well. To achieve such a goal enterprise, need qualified personnel having skills in computer sciences and related fields. It is in this regard that institutions like AICS Cameroon are the place to be nowadays in order to acquire such skills. Level III students at AICS are required to produce at the end of the year, a personalized project. As the name implicates, it is a project that is or might be of moral or psychological importance to us. We then decided to work on the theme “**DEVELOPMENT OF A PATIENTS' CONSULTATION AND FOLLOW-UP PLATFORM FOR CAMEROON GOVERNMENT HOSPITALS**”. Based on our theme, we needed to make thorough research in order to provide an acceptable solution. We divided this report into seven (7) parts which are as follows:

1. **Existing System:** Here, we shall present the already present system in place, that is the one used for consultation and follow-up purposes.
2. **Specification Book:** In this book, we specify the needs of the user taking into considerations the time and cost of the project.
3. **Analysis Document:** Here, we shall present the analysis method chosen together with the presentation of all the diagrams used for the analysis of this project.
4. **Conception phase:** This presents the generic and detailed conception of the project to bring out real world constituents.
5. **The Realisation phase:** This phase will permit us to visualize the implementation process of the solution.
6. **Test of functionalities:** In this phase, we shall present to you the different functionalities or modules of our application and how they work.
7. **The User Guide:** This elaborates on all conditions necessary to use the application and how to use it.



PART I

EXISTING SYSTEM

Preamble

The existing system is a document that provides a view of the system currently in place, that is how it carries out its different activities, also it provides a deep understanding of this system associated to the various limitations, the problems that result from these and the solution we propose.

Content

INTRODUCTION

- I. PRESENTATION OF THE THEME
- II. STUDY OF THE EXISTING SYSTEM
- III. CRITICISMS OF THE EXISTING SYSTEM
- IV. PROBLEMATICS
- V. PROPOSED SOLUTION



INTRODUCTION

As Engineers our role in the society is problem-solving which is targeted at making life easier for everyone and in order to bring forth a solution, we ought to ask ourselves some questions to know why the problem exist in the first place and how it is being dealt with at the time. The existing system refers to the system put in place to carry out the work done in the field on which our theme is based. Understanding this system is a great step in solving the problems that we might identify. It's of great importance that we take into consideration this system before proposing a suitable solution that will ameliorate or replace the current one.



I. PRESENTATION OF THE PROJECT THEME

Our theme is “DEVELOPMENT OF A PATIENTS' CONSULTATION AND FOLLOW-UP PLATFORM FOR CAMEROON GOVERNMENT HOSPITALS”. This application (iKare) will enable a distance consultation of patients by doctors. If we take the case of Corona Virus in 2020, we recall that it is an air borne disease, so easily transmitted in crowded areas. In our case a hospital is not only crowded but also a container of so many diseases especially during a pandemic. The idea is not to eliminate visits to a doctor by a patient which in my opinion is impossible, we want to limit such visits to limit the spread of diseases, transport costs and all other issues. Consultation will be done through video calls, chat messages for less serious cases, patients can also book for appointments with doctors for consultation either online or physically. At the end, depending on the doctor's diagnosis the patient will receive a prescription of drugs he/she needs to buy. The application is also equipped with a pharmacy search section, where a list of pharmacies closer to the patient is displayed, the patient filters the list by drug name, opened pharmacies.

In case lab tests are required, the patient can book an appointment for lab tests at his desired time. Also, while taking the drugs, the patient is been followed-up by a series of forms he fills concerning, the progress of his medication and how is feeling. If the current health situation is not good after 2/3 of the medications taken, the doctor is alerted.

iKare does not only allow consultations, scheduling doctors program, searching pharmacies, booking appointments for lab tests but also patients follow-up. All these to the patients' advantage.

II. DESCRIPTION OF THE EXISTING SYSTEM

We notice more and more an overheating of health centers because of the exponential increase of patients in Africa in general and in Cameroon in particular. As a result, health professionals are faced with a busy and often poorly organized schedule. Also, patients are constantly encountering problems during their medical follow-up. This is because they have to wait in long lines to be seen or even not be able to see a doctor. Even if an appointment has been made, it is possible that it is cancelled and the patient is not aware of it.

From a survey carried out from the 28th of July to 01st August 2023, the following results were obtained:

In which health institution do you work ?

12 réponses

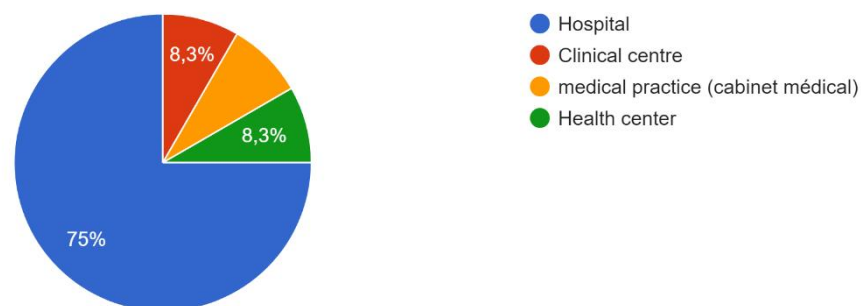


Figure 1: Doctor Survey Question 1

- 75% of the doctors work in a hospital while the 25% of the ones left are redistributed equally in a clinical center, health center and medical practice.

How do I get seen by a doctor ?

12 réponses

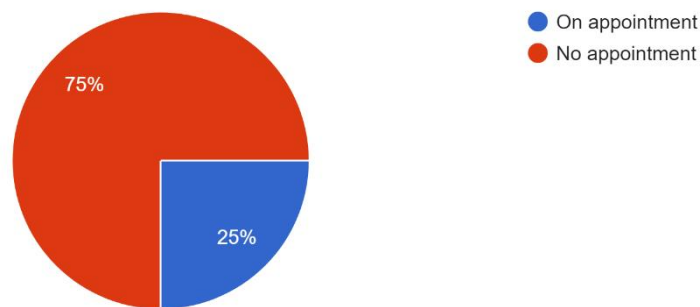


Figure 2: Doctor Survey Question 2

- 75% of doctors don't attend by appointments.

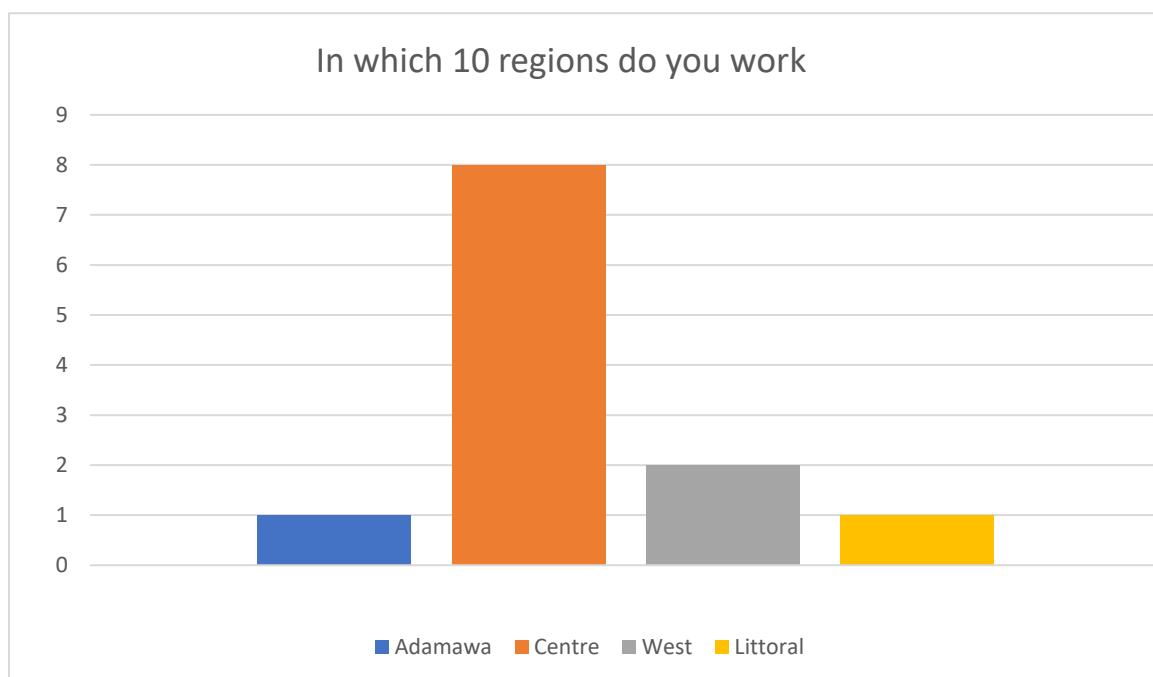


Figure 3: Doctor Survey Question 3

66.4% of the doctors work in Yaounde.

From the above results, we came out with the following observations

- ❖ Most health institutions consult without making appointments. A patient arriving at the institution usually encounters long queues. The patient usually waits for quite a long



time before his parameters like body temperature, blood pressure, weight are collected by nurses after he/she is done with the consultation fee.

- ❖ During the various exchanges between the patient and the doctor, certain questions are asked by the specialist in order to be able to identify the history and the illness from which the patient is suffering.
- ❖ At the end of the consultation, the doctor diagnoses and based on this diagnosis issues prescription to the patient.
- ❖ The patient buys the drugs at the health institute's pharmacy or if not available moves around looking for another where he can purchase the drug.
- ❖ If there are any tests to be carried out, the patient pays for the tests and goes to the laboratory where he gives a sample of his blood, faeces, urine or any other organic matter to the lab technician for analysis. The duration of the tests and results vary according to the specific tests involved. It usually varies from hours to days or weeks.
- ❖ Some follow ups are done at the level of the health institution (inpatient follow-ups) while others are done through calls (outpatient follow-ups) or doctors can schedule an appointment with the patient to try to see the evolution of the prescribed treatment, possibly present the results of a prescribed examination in order to find the anomaly.
- ❖ In case of emergency the patients are first taken care of before any identification according to the severity of the disease.

III. LIMITATIONS OF THE EXISTING SYSTEM

At the end of our collection of information concerning the medical consultation process in hospitals, we find that this system has weaknesses that should not be overlooked despite their warm welcome. However, it should be noted that some observations have been made within the clinic and it would be wise for us to suggest solutions to the various problems noted.

Table 1: Limitations, Consequence and proposed solution

LIMITATIONS	CONSEQUENCE	PROPOSE SOLUTION
A queue of sick people such as dizziness or a headache, the patient is forced to line up and wait for long hours to get a doctor's advice	<ul style="list-style-type: none"> - Patients' care is a tedious task. - Purchase and consumption of drugs offered by street vendors. 	<ul style="list-style-type: none"> - System proposes available doctors for remote consultation. - Set up spaces for "Online doctors".
Huge transport costs for people living in remote areas without a nearby hospital.	Calls to neighborhood doctors	Reduce these expenses by setting up an online medical consultation, no matter where the patient is.
Little or no communication between the doctor and the patient, the latter who will be forced to travel whenever the disease progresses.	Self-medication resulting in death of the patient in most cases.	Facilitate the dialogue between the doctor and his patient by setting up a video call module.
Drugs not always found in the nearby health institute's pharmacy or nearby hospitals (pharmacies).	Unnecessary moving around town in search of the drugs which can delay time of taking the drugs and can cause death in severe cases.	Developing a platform that facilitates search of pharmacies depending on location, drugs available, day shifts.
Appointments and schedules still done manually	Patient is not well taken care of and might lead to confusions and forgetfulness.	Developing a platform that manages appointments and schedules to avoid these confusions

Transport costs spent on outpatient follow-ups	Patient does not arrive on time and is not well taken care of.	Enhance follow-up through daily surveys on the platform and alerting a doctor in case of unexpected change or behavior.
--	--	---

IV. PROBLEMATIC

Since dialogue is the basis of a medical consultation, it is more than necessary that it be done in a simpler and more explicit way as possible because human lives are at stake. Through these various observations, we ask ourselves the question **HOW CAN WE FACILITATE ONLINE COMMUNICATION BETWEEN PATIENTS AND DOCTORS, FOLLOW-UPS OF PATIENTS AND ACCESS TO PRESCRIBED DRUGS FROM PHARMACIES?** We will try to offer a solution through our application.

V. PROPOSED SOLUTION

After our study and criticism of the existing situation, we propose to design a Web and a mobile application that will allow the citizen (Health professionals and Patients) of Cameroon to:

Health Professionals

- ✚ Have Electronic Agenda to manage availability and Appointment.
- ✚ Chat with patients.
- ✚ Consult through video calls.
- ✚ Set patients prescription online.

Patients

- ✚ Localize all nearby health institutes.
- ✚ View Health Specialist availability from different health institutes.
- ✚ Chat with health professionals.
- ✚ Receive notification after appointment reschedule.
- ✚ View medical prescription.
- ✚ Localize nearby, opened pharmacies.

CONCLUSION

Having reached the end of this part, we studied the existing system by questioning different health professionals in different institutes through the survey that enabled us to come out with the limitations of the system that leads to the consequences and we proposed solutions to the various limitations. Finally we proposed our solution which is a mobile and web application. This step is necessary because we need to understand how the system put in place currently works before we can facilitate or ameliorate the processes been carried out in this system.



PART II

SPECIFICATION BOOK

Preamble

The Specification book is a document which is been established by the customer needing a particular product and the producer (engineer) who is to create the product, which is to be respected to its fullness during the realization of the product. From this, we obtain the major importance of the specification book.

Content

INTRODUCTION

- I. CONTEXT AND JUSTIFICATION
- II. OBJECTIVES OF THE PROJECT
- III. EXPRESSION OF NEEDS
- IV. ESTIMATED COST OF THE PROJECT
- V. PROJECT PLANNING
- VI. LIST OF PARTICIPAANTS AND DELIVERABLES.

CONCLUSION



INTRODUCTION

The specification book of our report helps us provide details about our theme to improve our understanding of it and increase the likelihood of it succeeding. To delimitate the scope of our project, we will specify the context of our theme. From the context, we will list the problems we have decided to address throughout the project. After presenting our solution, we will talk about the objective we have set for the project. Also, we will explore the needs to which our system will respond both at the functional and non-functional level. We will then look at the estimated financial requirements for our project and establish a plan we will follow to complete our project on time. From here, we will discuss what is expected of us by the end of the project under the project deliverables.

I. CONTEXT AND JUSTIFICATION

A. Context

Cameroon is a home to closely 28,703,875 million inhabitants and in situations where a pandemic is declared, the feeling of anxiety and uncertainty can overwhelm people, and health system can struggle to cope with an uncontrolled and exponential demand. There are approximately 1.1 physicians and 7.8 nurses and midwives per 10,000 populations (WHO AHWO Cameroon Fact Sheet, 2010).

A survey carried out from July 29th to July 31st, 2023, revealed the following results.

When going to the hospital for consultation, do you encounter long queues ?

48 réponses

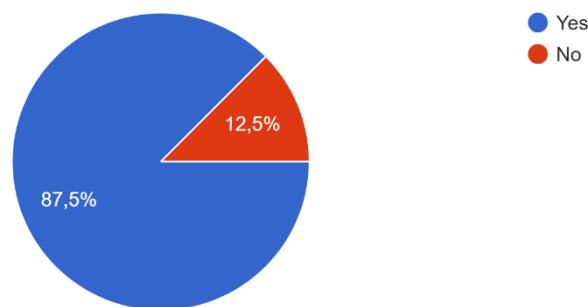


Figure 4: Patient Survey question 1

We can see from the above that, of the 48 people who answered to the questions from the survey, 87,5% encounter queues when going to the hospital. Also,

After been prescribed drugs, which problem do you often encounter, getting the drugs ?

48 réponses

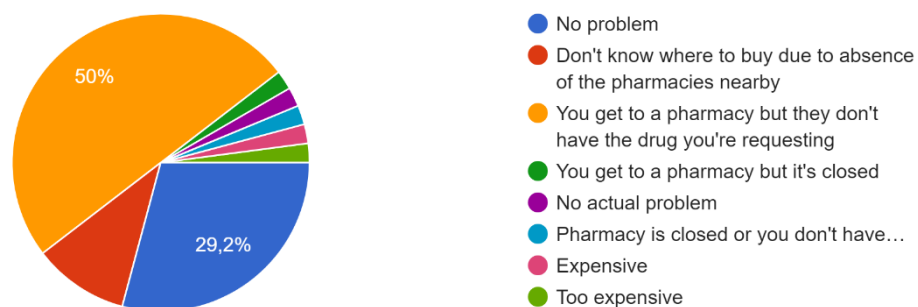


Figure 5: Patient Survey Question 2

It's clear that only, 29.2% of those interviewed don't have a problem getting a prescribed drug, the rest are distributed between locating the pharmacy that's opened at that moment, locating the pharmacies nearby, getting the drug from the opened pharmacy, or knowing ahead of time the price of the drug you are looking for ensuring you have enough money on you.

Do you often forget to take medications?
48 réponses

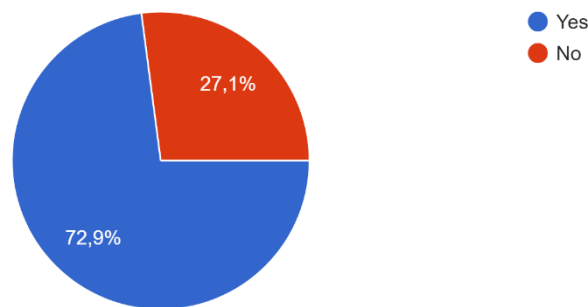


Figure 6: Patient Survey Question 3

We can also see that 72.9% of those who answered often forget to take medications, they might have many things in mind and can't even see the time pass.

Do you wish to be followed-up, may be each time you're on a medication ?
48 réponses

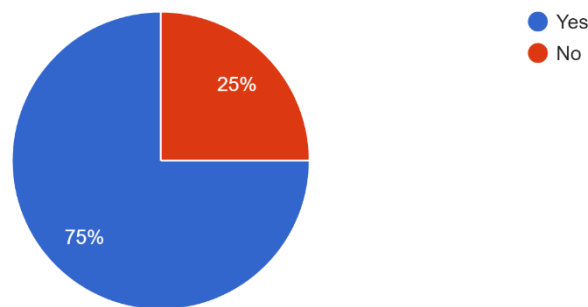


Figure 7: Patient Survey Question 4

We can also notice that 75% of those who answered require follow-up each time they are prescribed a drug or each time they are on medication.

B. Justification

The health sector is experiencing a growing number of patients. As a result, with hospitals and clinics overheated, the management of patients and their medical follow-up is a major problem for patients and doctors. Insofar as it is for the most part impossible to know the availability of health professionals and to make an appointment or to be consulted on time. Even after consultation, patients do not know how to continue especially when the hospital's pharmacy does not have the required drugs, they are looking for. They might be moving around the whole town looking for a specific drug because either the pharmacies they find don't have that specific drug, they are closed, or the price of the drug is beyond their expectations.

For this reason, iKare will allow the patient(s) to be consulted remotely, without having to follow a queue, or by taking appointments depending on your availability. The notion of distance here means that the doctor and the patient are not in the same place during the consultation; and this will limit the risk of contamination, save time, and have centralized management of patient data. Also, iKare will favor searching of pharmacies by filter, that is according to those nearby, opened, having the required drugs and raising awareness concerning the price of the medication. Patients will be able to book appointments for lab tests.

II. OBJECTIVES OF THE PROJECT

A. GENERAL OBJECTIVE

The main objective of this project is to enable follow-ups of patients through remote consultations.

B. SPECIFIC OBJECTIVES

- ❖ The patients will be able to log into the platform, book for an appointment for video consultations.
- ❖ Be consulted, diagnosed and issued prescriptions according to the diagnosis carried out by the doctor.
- ❖ Book for tests in a laboratory if required, the tests are sent directly to the laboratory from the doctor.
- ❖ Receive a list of all pharmacies and applying a filter by location, drug searched and pharmacy type (day / night).
- ❖ To be followed up, during the illness to recovery period, through regular surveys on how they feel.
- ❖ Discuss with doctors through chat messages for less serious cases.

III. EXPRESSION OF NEEDS

1. Functional Needs

In this context, functional requirements describe what the system or application should do. The modules are as follows;

- ❖ The administrator (admin) should be able to:
 - **Account Creation:** The health Institute Admin is responsible for creating the health professionals' accounts and lab administrators' accounts for a particular health institute.
 - **Account Management:** Here users can manage their account by modifying some of their information.
 - **User Authentication:** Allow the users to register or login into the application while also providing a secure mechanism to recover their account in case they forgot their password or otherwise. In addition, the user account must be confirmed to ensure identity proof.
 - **Manage Electronic Record Book:** Here the patients can upload recent medical booklets, these records are kept secured on the platform, each time medical personnel wishes to access it, he/she requires permission from the patient. Also, the patient's data are encrypted using sophisticated algorithms.
 - **Localize pharmacies:** Here patients will get a view of all pharmacies nearby, opened by that time and which have the required drugs they are looking for.
 - **Manage Appointment:** Here doctors have an agenda where they can validate, cancel, re-schedule an appointment, view their appointment date and even set a reminder.
 - **Follow-up:** Allow the users to answer a series of questions each time they take a specific drug, this allows the production of a dataset and notifies the doctor in charge when the patient situation is abnormal.
 - **Exchange between Patient and Health Professional:** It provides an interface where the doctor can have online consultation with patients who make an appointment. Allows health specialists to create meeting and add more participants, also a translation module will enable native speakers communicate with medical professionals, removing language barriers.

2. Non-Functional Needs

They specify the quality attribute of a software system. They judge the software system of application based on Performance, Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to its success. Failing to meet non-functional requirements can result in a system that fails to satisfy user need.

Performance, Scalability and Security

Performance defines how fast a software system, or its piece responds to certain user's action under certain workload. In most cases, this metric explains how much a user must wait before the target operation happens (the page renders, a transaction is processed, etc.) given the overall number of users at the moment. But it's not always like that. Performance requirement may describe background processes invisible to users. Our goal will be to provide our users with the best performance as it affects the overall user experience.

Scalability accesses the highest workloads under which the system will still meet the performance requirements. In this project we will mainly leverage the power of cloud storage and third-party API's.

- ✚ The application should have a friendly user interface (UI) and should be easy to use.
- ✚ The code should be clear to facilitate future development and improvement.
- ✚ The web application should be resizable when opened on any device (android phone, computer, tablet web browsers)

Security is also one of the most important aspects of any system, especially the ones dealing with sensitive user's information.

- ✚ The application should provide a strong security mechanism to reassure user's that they can trust it for their sensitive information.
- ✚ Some of the security principles include reassuring confidentiality, accountability, Integrity, Authentication.

IV. PROJECT PLANNING

a. Chronogram of activities

Table 2: Chronogram of activities

PHASE	OBJECTIVE	OUTPUT	DURATION	PERIOD
EXISTING SYSTEM	Study of the existing system	Existing System	5 days	4 th March to 8 th March
SPECIFICATION BOOK	Specification of the user needs	Specification Book	5 days	11 th March to 14 th March
ANALYSIS	Capture of needs Use case and textual description Modelling	Analysis Book	15 days	18 th March to 5 th April
CONCEPTION	Preliminary conception and Detailed conception	Conception book	15 days	8 th April to 26 th April
REALIZATION	Implementation Unitary test Integration Test Development, Deployment, Component diagrams	Realization book	30 days	29 th April to 7 th June
TEST OF FUNCTIONALITIES	Testing of the software and debugging	Test of functionalities	10 days	10 th June to 21 st June
INSTALLATION AND USER GUIDE	Documenting software	User Guide	1 w	21 st June to 28 th June

b. Gantt Project planning

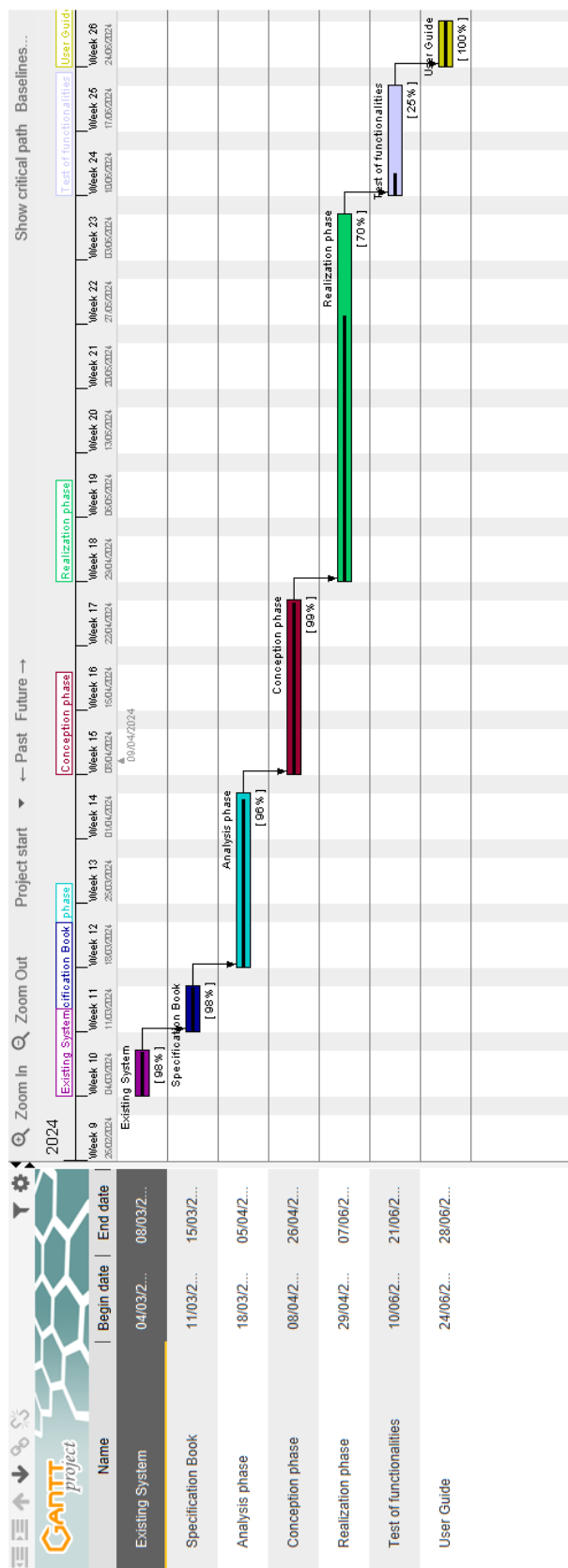


Figure 8: Gantt project planning

V. ESTIMATED COST OF THE PROJECT

a. Software Resources

Table 3: Software Resources (source : Mercurial 2022)

RESOURCES	DESIGNATION	USAGE	QUANTITY	UNIT COST (FCFA)
Formatting	Microsoft Office 365	Writing of the report	1	47 998
Cloud Storage	Google Cloud	Saving our report	1	Freemium
Web browser	Google Chrome	View web pages	1	Freeware
Code Editor	Visual Studio Code	For writing the code of the application	1	Freeware
Project planning	Gantt Project	For building a Gantt chart	1	Freemium
Illustrating tool	Ichogram	For building a geographical location of the enterprise	1	Freemium
UML Analysis	Sybase PowerAMC	For drawing UML diagrams	1	Freemium
Testing Mobile App	Expo Go	For testing the mobile application	1	Freemium
TOTAL 1			8	47 998

b. Hardware Resources

Table 4: Hardware resources (source: <https://www.scribd.com/document/561202966/mercuriale-2022>)

RESOURCES	HARDWARE	Quantity	UNIT COST (FCFA)
Computer	<i>LAPTOP HP ELITEBOOK 8540P CORE I5, DISQUE DUR 500 GO, RAM 4 GO, ECRAN LARGE 15" HD</i>	1	402 500
Printer	Printers	1	546 250
Network	Local network installation	1	300 000
Removable Disk	Removable Disk	1	11 555
Smart phone	Smart phone	2	150 000
TOTAL 2		6	1 410 305

c. Human resources

Table 5: Human Resources (source : Mercurial 2022)

ROLE	NUMBER OF DAYS	QUANTITY	COST PER DAY	TOTAL PRICE (FCFA)
Project Manager	120	01	30 000	3 600 000
Analyst	45	01	25 000	1 125 000
UI/UX Designer	30	01	20 000	600 000
Programmer	45	01	15 000	675 000
Tester	14	02	25 000	700 000
Margin error	/	/	/	1,475,000
TOTAL 3				8 175 000

d. Global Estimation

Table 6: Global Estimation

TOTAL 1(FCFA)	TOTAL 2(FCFA)	TOTAL 3(FCFA)	OVERALL TOTAL (FCFA)
47 998	1 410 305	8 175 000	9 633 303
NINE MILLION SIX HUNDRED AND THIRTY THREE THOUSAND THREE HUNDRED AND THREE			

VI. CONSTRAINTS

1. Technical constraint

For the development of our system, we have sufficiently robust tools to guarantee a minimum of security, extensibility and excellent scalability. Moreover, the programming phase will have to follow all the technical standards for a better performance in a reduced execution time, this is why the choice of the development technologies is crucial.

2. Time Constraint

The project will be realized in 14 weeks starting from the beginning date coupled with many other school projects.

3. Cost constraint

The realization of our project will require expenditures in human resources, material and software a total cost of 9 633 303 FCFA.

VII. LIST OF PARTICIPANTS AND DELIVERABLES

a. LIST OF PARTICIPANTS

Table 7: List of participants

NAME	FUNCTION	ROLE
Mrs. EBENYE Vanessa	Follows up student	Supervisor
KAMENI SEPDEU Ange Chris	AICS Student	Student at AICS

b. DELIVERABLES

1. In project management, any component materializing the result of a realization service is called a deliverable. In the case of our project, the deliverables are: A report composed of the following document
 - ❖ The application.
 - ❖ The user guide.
 - ❖ The powerpoint.



CONCLUSION

Reaching the end of this part. The list of objectives to be achieved has been enumerated and made clear. The specification book permitted us to present the different actors associated with the project as well as the requirement and the provisional planning needed for the achievement of our project. We will move directly to the next part which is the analysis phase. In the analysis phase, we will model our system with a modelling language and a unified process, do a comparative study of uml and merise.



PART III ANALYSIS BOOK

Preamble

The main objective of the analysis phase is to capture the user's need, the delimitation of the field of study and to have a clear understanding of the system in study. To achieve this, we will use UML (Unified Modelling Language) with the 2TUP (2 Track Unified Process) as method applied to UML to analyze the system. We will start by doing a comparison between UML and MERISE, of various unified processes and lastly, we will present the modelling of the solution we propose (Our software).

Content

INTRODUCTION

I. METHODOLOGY

1. COMPARATIVE STUDY OF UML AND MERISE.
2. COMPARATIVE STUDY OF UNIFIED PROCESSES.

II. CHOICE OF THE ANALYSIS METHOD

III. MODELLING OF THE PROPOSED SOLUTION

CONCLUSION



INTRODUCTION

System development can be thought of as having two major components: System analysis and system design which both help in understanding the details of the existing system or the system to be designed. The analysis and design of information systems has most of the time vocation to allow the creation of databases, which must represent as closely as possible the reality of the field studied thus requiring the use of a design method. This is why our choice will be directed on the UML method as it offers much to developers seeking a user-centered approach and / or a wide scope in design. This part of the report consists of the comparative study of UML and MERISE, unified processes and finally the various diagrams that meet the functional need requirements.

I. METHODOLOGY

1. COMPARATIVE STUDY OF UML AND MERISE





a. MERISE

MERISE stands for “Méthode d’Etude et de Réalisation Informatique pour des Systèmes d’Entreprise”. Although it is prescriptive to some extent, MERISE permits the participation of end users and senior management as well as data processing professionals in its decision cycle. MERISE is a method for designing, developing and carrying out IT projects. The goal of this method is to achieve the design of an information system. The MERISE method is based on the separation of data and processing to be carried out in several conceptual and physical models. The essentials of the approach lie in its three cycles: the decision cycle, the life cycle and the abstraction cycle, which cover data and process elements equally. The separation of data and processing ensures longevity in model. Indeed, the arrangement of data does not have to be often overhauled, while treatments are more frequently.

b. UML

UML (Unified Modelling Language) is a standard notation for the modelling of real world objects as a first step in developing an object-oriented design methodology. Its notation is derived from and unifies the notations of three object-oriented design and analysis methodologies: Grady Booch's methodology for describing a set of objects and their relationships, James Rumbaugh's Object-Modelling Technique (OMT), Ivar Jacobson's approach which includes a use case methodology. Other ideas also contributed to UML, which was the result of a work effort by Booch, Rumbaugh, Jacobson, and others to combine their ideas, working under the sponsorship of Rational Software. UML captures information about the static and dynamic view of a system. UML 2.5 comprises of 14 diagrams which represent the different views of a system. The 14 diagrams can be subdivided into two, Static or structural and Dynamic diagrams. These diagrams include;

i. STATIC OR STRUCTURAL DIAGRAMS

-  Class diagram;
-  Object diagram;
-  Component diagram;
-  Deployment diagram;

- ✚ Composite Structure diagram;
- ✚ Package diagram;
- ✚ Profile Diagram;

ii. BEHAVIOURAL OR DYNAMIC DIAGRAMS

- ✚ Use case diagrams;
- ✚ Activity diagram;
- ✚ State machine diagram;
- ✚ Sequence diagram;
- ✚ Communication diagram;
- ✚ Global Interaction diagram;
- ✚ Timing Diagram;

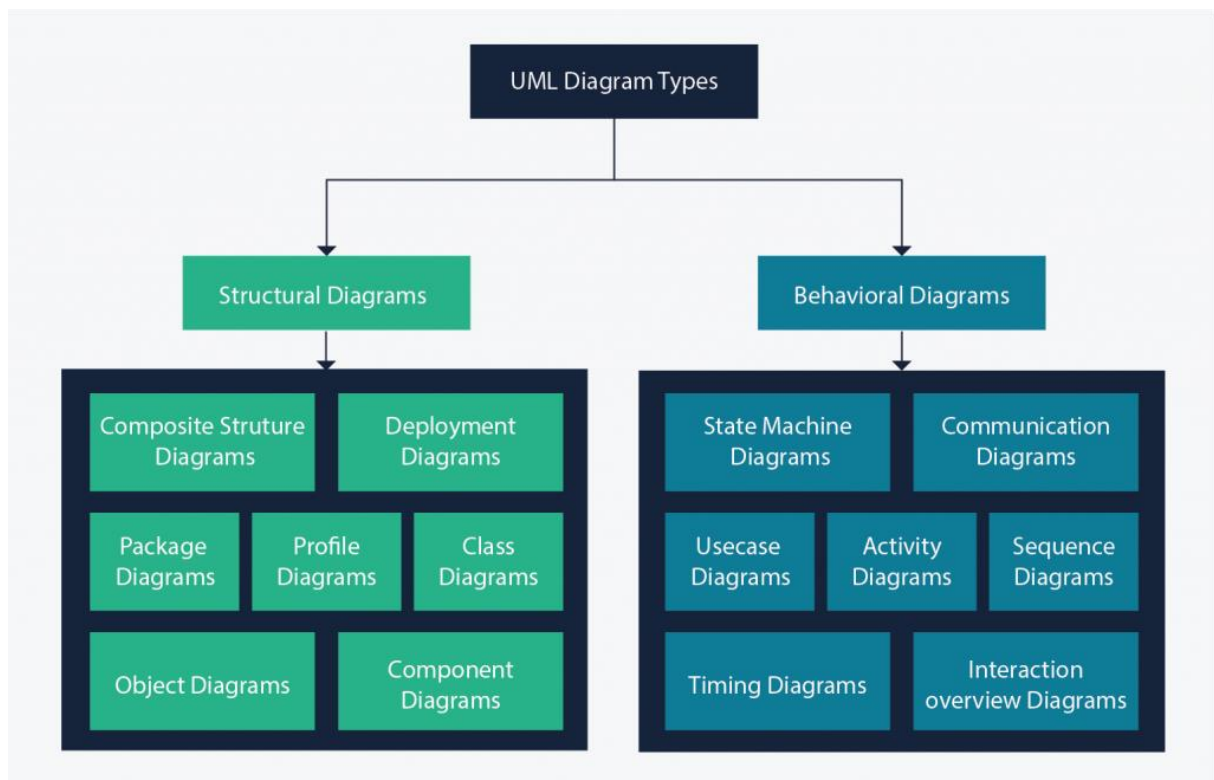


Figure 9: UML 2.5 diagrams overview (source: <https://creately.com/blog/diagrams/uml-diagram-types-examples/>)

On very important notice is that UML is not a method but a modelling language. As such to give it an approach we need to associate UML to a Unified Process (UP) in other to give our conception a methodology to follow. There exist several Unified Processes, but our modelling approach will be the 2TUP (Two-track unified process) which we will use in the course of our project.

Table 8: Differences between UML and MERISE

MERISE	UML
It stands for Méthode d'Étude et de Réalisation Informatique pour les Systèmes d'Entreprises	Unified Modeling Language
MERISE is a systemic method of analysis and design of information systems. That is, it uses a systems approach.	UML is however not a method but rather an object modeling language to which it is necessary to associate an approach to make it a method. This is the case with the 2TUP method; RUP and XP.
MERISE proposes to consider the real system from two points of view: - A static view (data) - A dynamic view (treatments). That is, with the MERISE method, we have a separate study of the data and the treatments.	UML offers a different approach from that of MERISE in that it combines data and processing. Because with UML, centralizing the data of a type and the associated processing makes it possible to limit the maintenance points in the code and facilitates access to information in the event of software development. In addition, UML describes the dynamics of the information system as a set of operations attached to the objects of the system.
Rational	Object

2. COMPARATIVE STUDY OF UNIFIED PROCESSES

a. A Unified Process

A Unified Process is a process of development of software constructed on UML; it is iterative, incremental, centered on architecture, driven by use cases and requirements.

Iteration is distinct sequence of activities with a basic plan and evaluation criterion that produces an internal or external output. Either the content of an iteration is improved, or the evolution of the system is evaluated by users.

An increment is the difference between two released products at the end of two iterations. Each iteration that the group is capable of integrating the technical environment in order to develop a final product and give users the possibility of having tangible results.

Centered on architecture the different models derived during the establishment of system must be reliable and coherent.

Driven by use case and requirements enables the clear definition of a users' needs and priorities respectively thereby minimizing the risk of project failure.

b. The Two Track Unified Process (2 TUP)

2TUP is a unified process which is built on UML and has as objective to bring solution to constraints of functional and technical changes imposed on information systems by strengthening controls on development capacities. It proposes a Y-shaped development life cycle that separates the functional aspect from the technical aspects, and the merging of these two forms the implementation aspect. 2TUP distinguishes therefore two branches: the functional and technical branches, the combination of the result of these two branches forms the third: the realization branch – where we realize our system. The diagram below illustrates the branches of 2TUP.

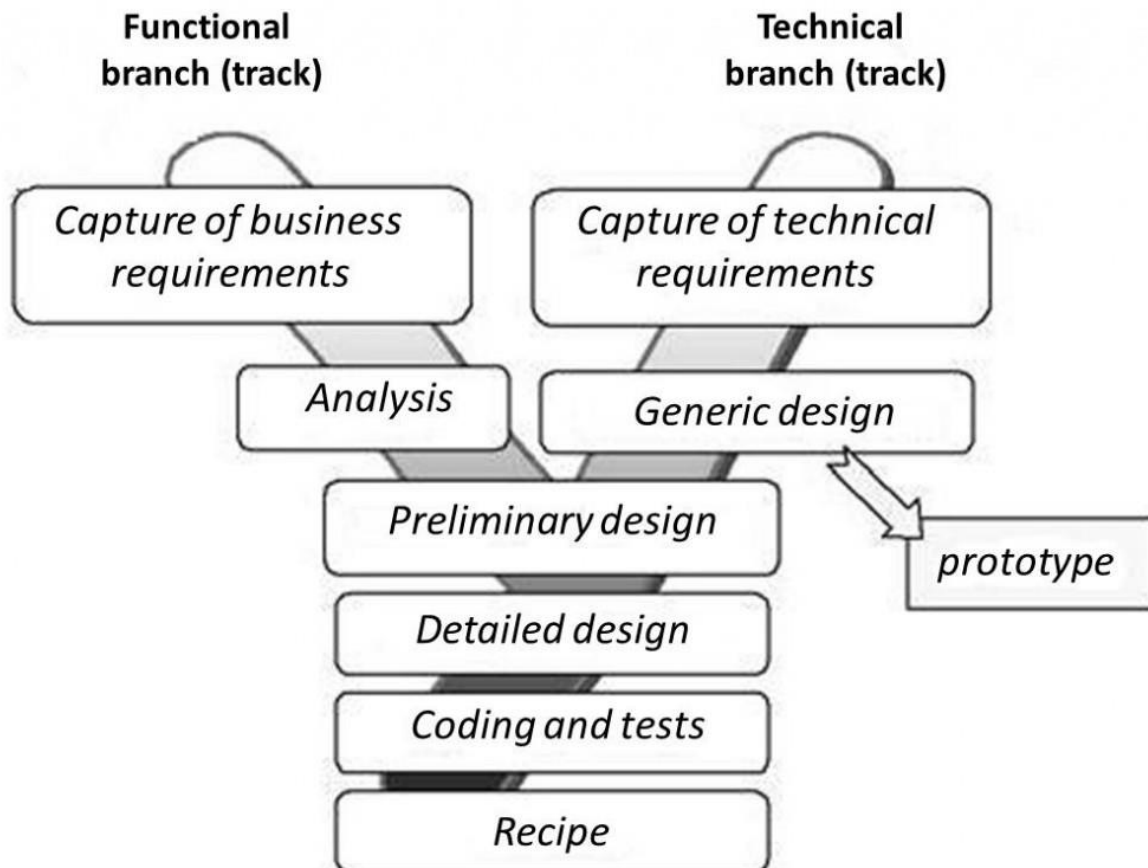


Figure 10: 2TUP diagram (source: <https://www.mysciencework.com/omniscience/pervasive-mobile-healthcare-system-based-on-cloud-computing>).

a. The left branch (functional branch)

It captures the functional needs of a system. This ensures the production of software that meets the needs/requirements of the user. The analysis here consists of studying precisely the functional specification in order to obtain an idea of what the system is going to realize, and its result does not depend on any technology.

b. The right branch (Technical branch)

The technical branch enumerates the technical needs and proposes a generic design validated by a prototype. The technical needs include constraints and choices related to the conception of the system, the tools and equipment as well as the integration constraint with the existing system condition.

c. The middle branch (Realization or Implementation branch)

In this branch, we study the preliminary conception, detailed conception, and documentation of the system. The realization branch supports the following:
Preliminary conception: This is the most sensitive step of 2TUP as it is the confluence of the functional and technical branch. It is completed when the deployment model, the operating model, the logical model, interphases and the software configuration model are defined. We have the following diagrams:

- ✚ Component Diagram ;
- ✚ Deployment Diagram ;
- ✚ Package Diagram ;
- ✚ Composite Structure Diagram ;

Detailed conception: This is the detailed design of each feature of the system. We have the following diagrams:

- ✚ Class;
- ✚ Object;
- ✚ Sequence;
- ✚ Timing Diagram;

Coding and testing: This is the phase where we program the designed features and test the coded features.

The recipe: Also known as the deliverables is the validation phase of the functions of the developed system.

II. CHOICE OF THE ANALYSIS METHOD

The reason why we chose UML modelling language and the software development process 2TUP instead of many others that exist, include:

UML is the current standard for programming in an object-oriented language. For this reason, it is widely understood and well known making it easy for a new programmer to join the project and be productive from the very first day.

UML diagrams allow teams to virtualize how a project is or will be working, and they can be used in any field, not just software engineering. The diagrams will allow teams to virtualize together how a system, or a process will work or did work. It can provide new ideas for how teams have to collaborate to achieve the goal of the workflow process.

2TUP is centered around the creation and maintenance of a model, rather than the production of mountain documents.

2TUP is user oriented as it permits the development of software that responds to the needs of the users through the study of the user needs.

2TUP is iterative and incremental, hence it enables the project team to produce refined amelioration if necessary and easily integrate it in the already existing system.

2TUP by permitting the project team identify and test the key functionalities of the system limits the risk related to building the system.

III. MODELLING OF THE PROPOSED SOLUTION

A. Capture of Functional Needs

The first step of the left (functional) branch of Two Track Unified Process (2TUP) is the capture of the functional needs. At this step, we capture the intended behavior of the system that maybe express as services, tasks or functions the system is required to perform.

1. USE CASE DIAGRAM

a) Definition

Use case diagram shows the functionalities of a system, their interdependencies and how they relate with actors of the system. A use case is a specification of behaviour. The main objectives of the use case diagram are:

- ❖ Provide a high-level view of the system.
- ❖ Identify the functions of the system.

Use case diagrams are completed with a textual description of each use case that is intended to define the use case in greater details.

b) Formalism

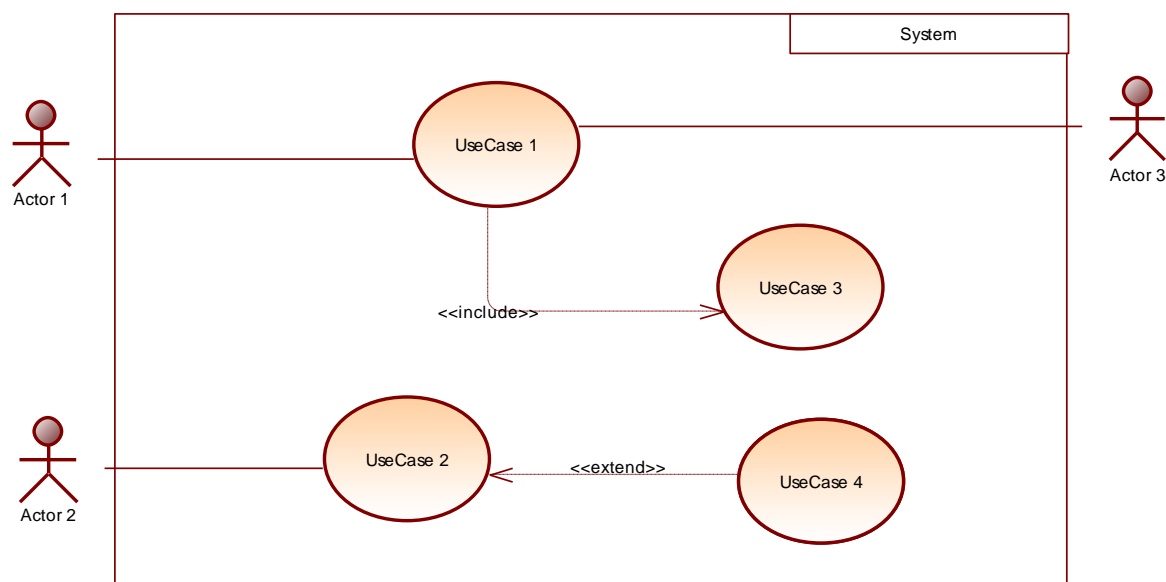

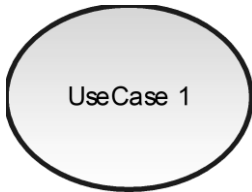

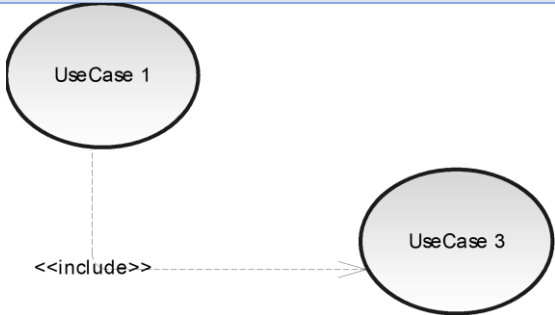
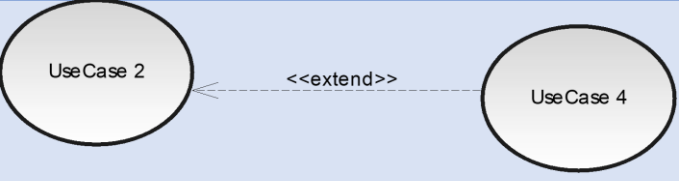
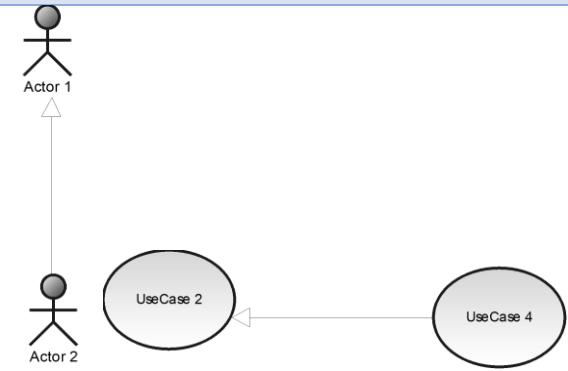
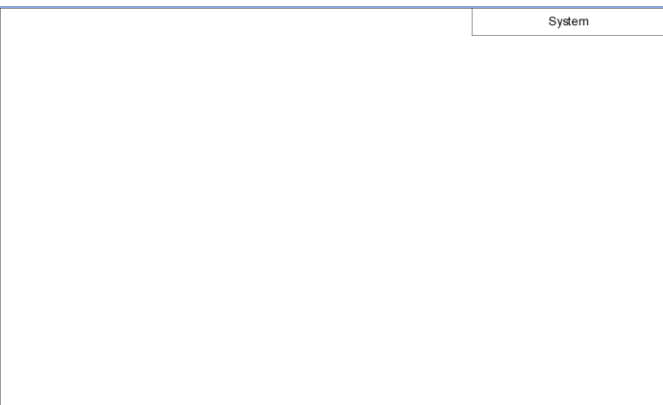


Figure 11: Use Case Diagram formalism

Table 9: Use Case Diagram Component

Elements	Notation	Description
Actors	 <p>Actor 1</p>	Represents an entity that directly interacts with the system. The actor is what performs the different possible actions of the system
Use case	 <p>UseCase 1</p>	A use case represents a functionality of the system. It is an action that can be performed by an actor.
Association	 <p>Actor 1</p> <p>UseCase 1</p>	it indicates that an actor takes part in a use Case.
Include	 <p>UseCase 1</p> <p>UseCase 3</p> <p><<include>></p>	An inclusion denotes that an included action must be performed before the including action can be performed.

Extend		An extension denotes that an extending action may be performed while an extended action is being performed.
Generalization		This shows that an actor or a use case is a kind of another abstract or concrete actors can be defined and later specialized using generalization relationship.
System		It is a container of use cases which interact with external actors

c) The Actors of our System

Table 10: Actors of our System

Actor	Role
Administrator	The administrator is responsible of managing the accounts of patients and health professionals (validating, deleting accounts) and general functioning of the system.
Super Administrator	He also manages the whole system and all the users (including the administrator)
Patient	The user that receives services from the health professionals and the apps
Health Professional (Doctor)	These are verified personnel in charge of counselling and providing consultation to the users.
Geolocation API	Aids in the geographical location of the searched and selected pharmacy
Payment API	Provides payment services to the application in terms of MTN mobile money and Orange Money.
SMS API	Provides a service to send SMS to users, notifying them of updates on your account.

d. General Use Case Diagram

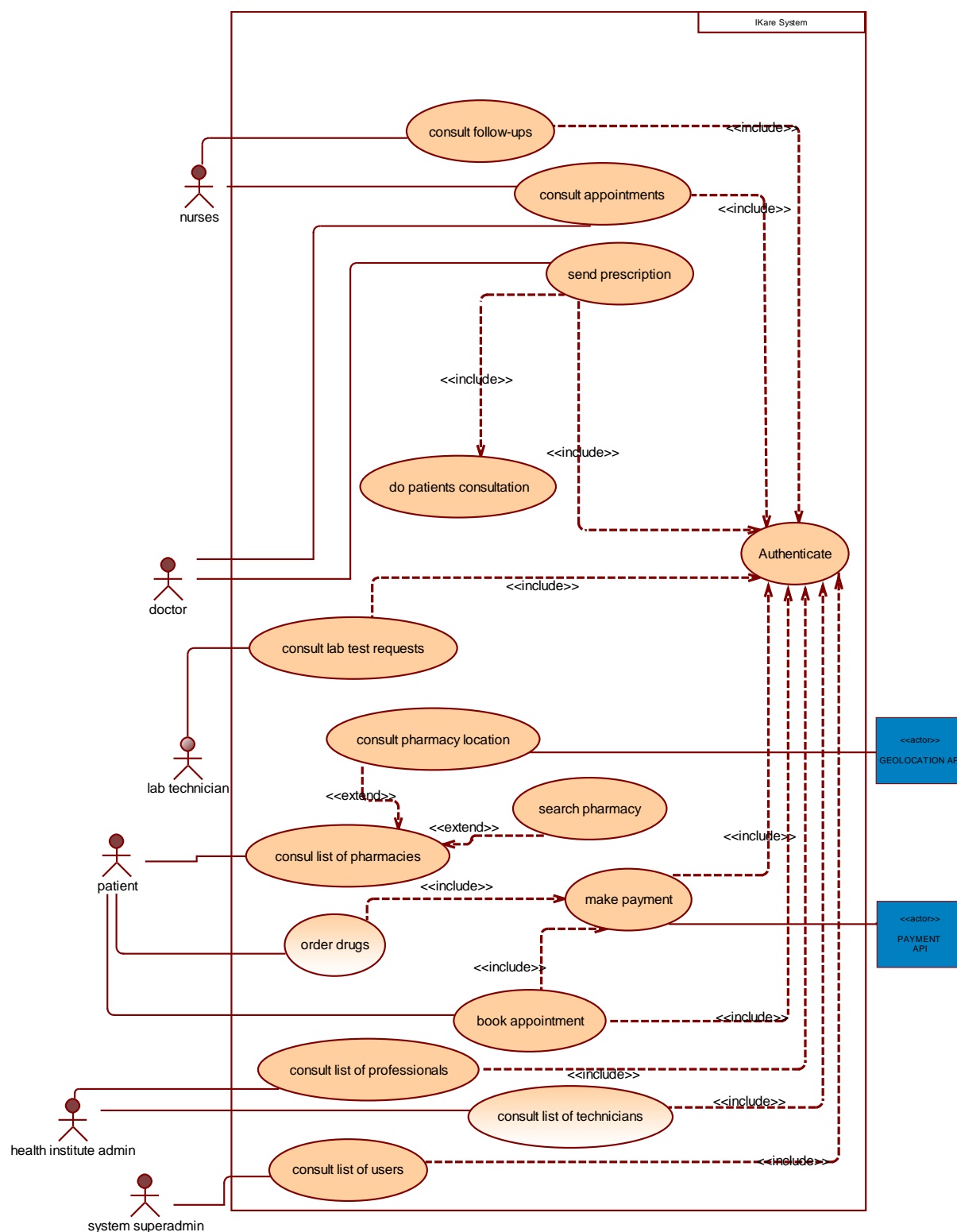


Figure 12: General Use Case Diagram

e. Consult appointment use case diagram

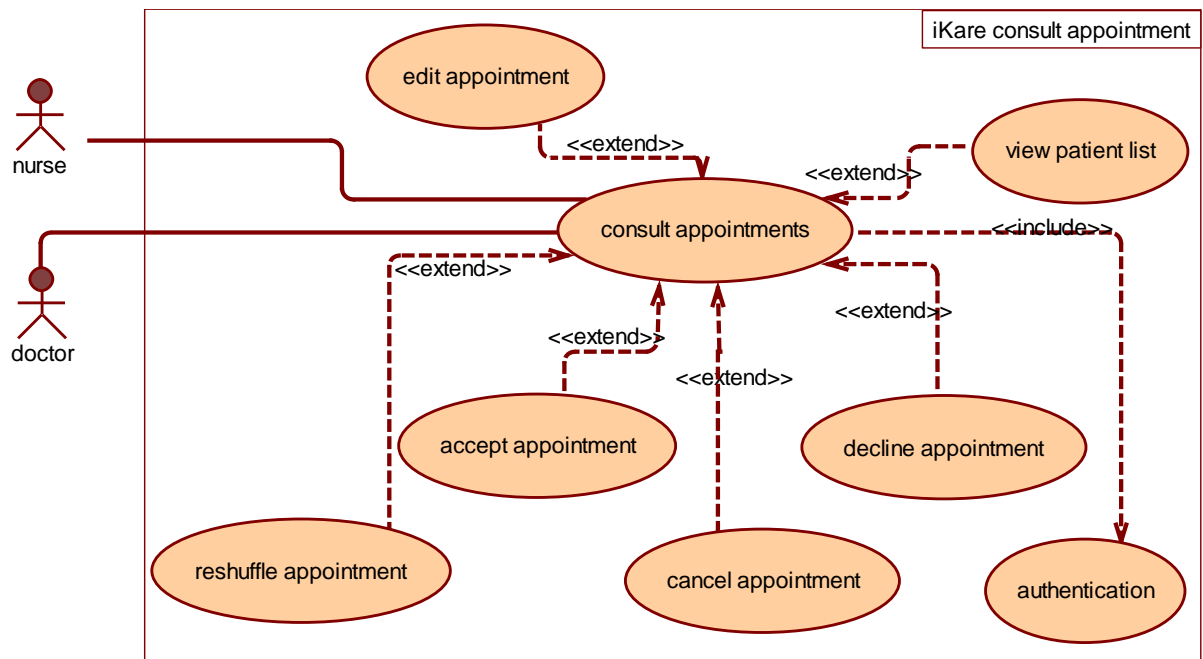


Figure 13 consult appointment use case diagram

f. Consultation use case diagram

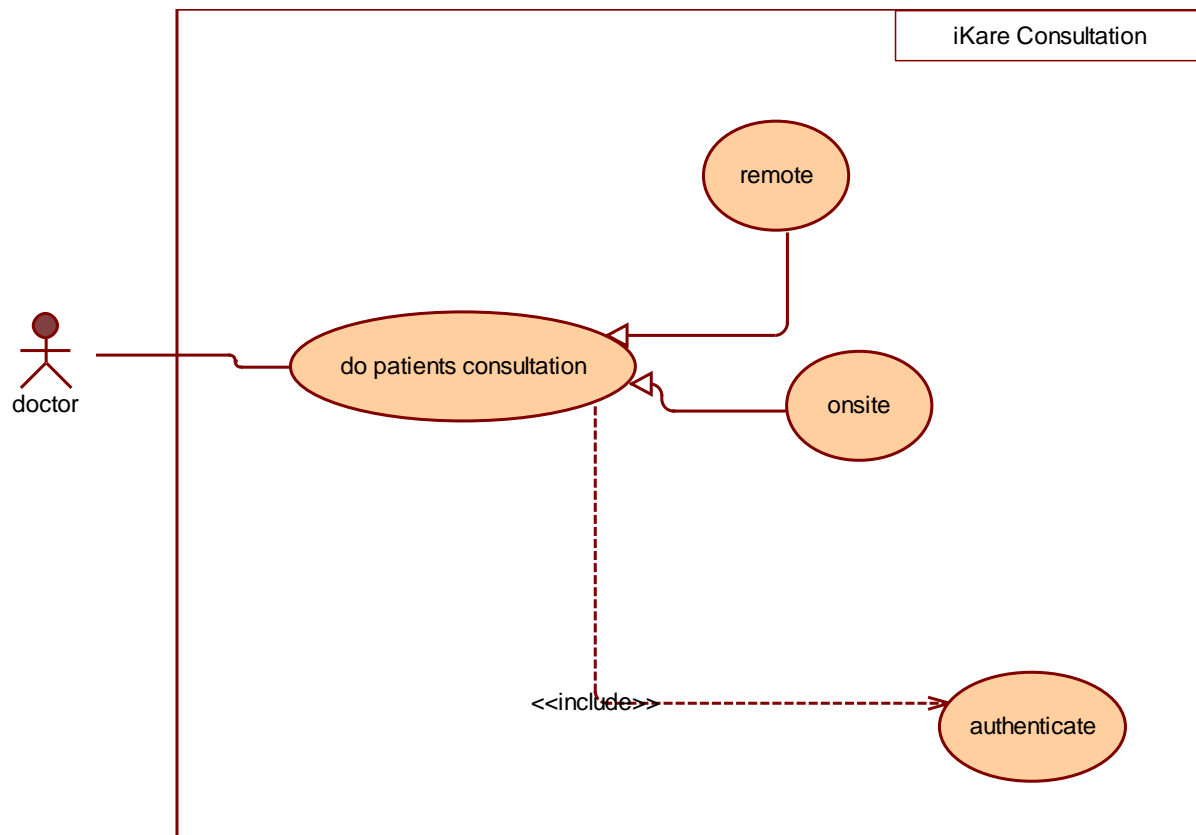


Figure 14: consultation use case diagram

g. Textual description of use cases

Table 11: Authenticate Textual Description

Title	Authenticate
Summary	The user needs to authenticate
Actors	Patient, Health Professional, Lab technician
Precondition (s)	<ol style="list-style-type: none"> 1. The app is launched. 2. The actor has an account on the platform.
Triggers	The user clicks on login button
Nominal Scenario	<ol style="list-style-type: none"> 1. The system displays the login form. 2. The actor fills and submits the form. 3. The system verifies conformity of the form fields 4. The system sends the data to the dbms. 5. The dbms returns result of the query. 6. The system displays a success message to the actor.
Alternative Scenarios	<ol style="list-style-type: none"> 7. At step 4 of the nominal scenario, the user enters mismatched or missing information. 8. The system displays an error message then returns to step 2 of the nominal scenario.
Postcondition of success	The user has access to his/her dashboard
Postcondition of failure	The user does not have access to the platform
Non-functional requirement	Entering the password must not be visible on the screen

Table 12: Book appointment textual description

Book appointment	
Title	
Summary	The user books for a consultation appointment.
Actors	Patient
Precondition (s)	The actor is authenticated.
Triggers	The user clicks on book appointment button.
Nominal Scenario	<ol style="list-style-type: none"> 1. The system displays the list of hospitals. 2. The actor selects a hospital from list. 3. The system displays the doctors from the hospital. 4. The actor selects a doctor. 5. The system displays a form. 6. The actor fills and submits the form. 7. The system displays a success message
Alternative Scenarios	<ol style="list-style-type: none"> 8. At step 4 of the nominal scenario, the user enters mismatched or missing information. 9. The system displays an error message then returns to step 2 of the nominal scenario.
Postcondition of success	The user has booked for an appointment.
Postcondition of failure	The user has not booked for an appointment.

Table 13: Video Consultation textual description

Video Consultation	
Title	Video Consultation
Summary	The user needs to consult through video calls.
Actors	Health Professional
Precondition (s)	The actor has authenticated
Triggers	The user clicks on consult button
Nominal Scenario	<ol style="list-style-type: none"> 1. The system displays a list of upcoming appointments. 2. The actor clicks on appointment card. 3. The system displays the appointment details. 4. The actor clicks on launch call button. 5. The system displays video consultation UI.
Alternative Scenarios	<ol style="list-style-type: none"> 6. At step 5 lack of network connectivity arises 7. The system displays an error message
Postcondition of success	The actor is able to consult.
Postcondition of failure	The actor can't consult.
Non-functional requirement	There must be a good network connectivity.

2. COMMUNICATION DIAGRAM

a. Definition

Communication Diagrams model the interactions between objects in a sequence. They describe both the static structure and the dynamic behavior of a system. It is a simplified version of a Collaboration Diagram introduced in UML 2.0. A communication diagram is more focused on showing the collaboration of objects rather than the time sequence.

b. Formalism

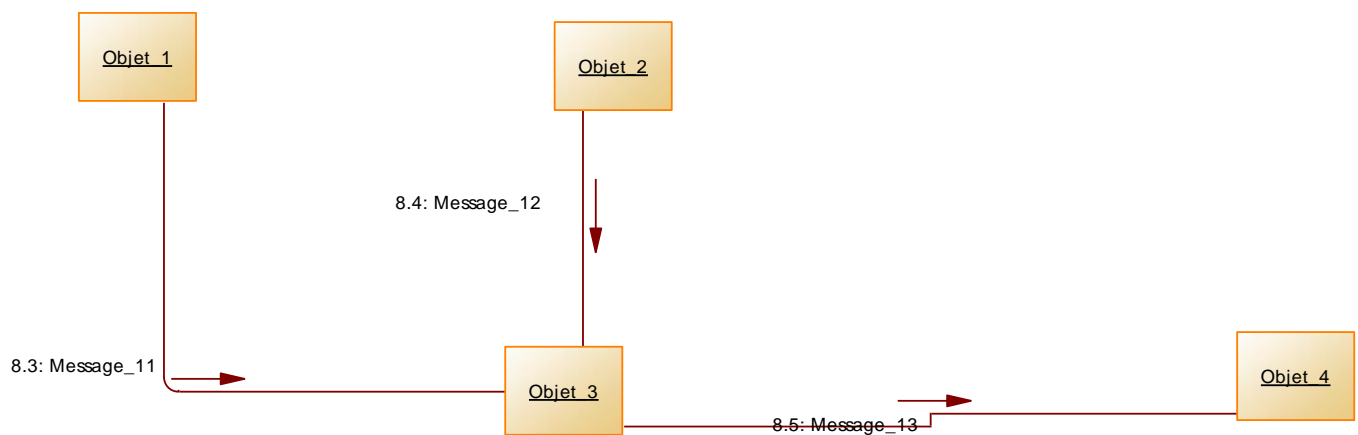
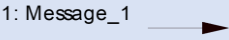





Figure 15: Communication Diagram formalism

c. Components of a Communication Diagram

Table 14: Elements of a communication diagram

ELEMENT	NOTATION	DESCRIPTION
Message		Designs a particular communication between lifelines.
Connectors		It represents the relationships that exist between lifelines
Dependency		A dependency is a relationship that signifies a single or a set of model elements for their specification
Lifeline		An object represents an individual participant in the interaction conversation.

d. Authenticate Communication Diagram

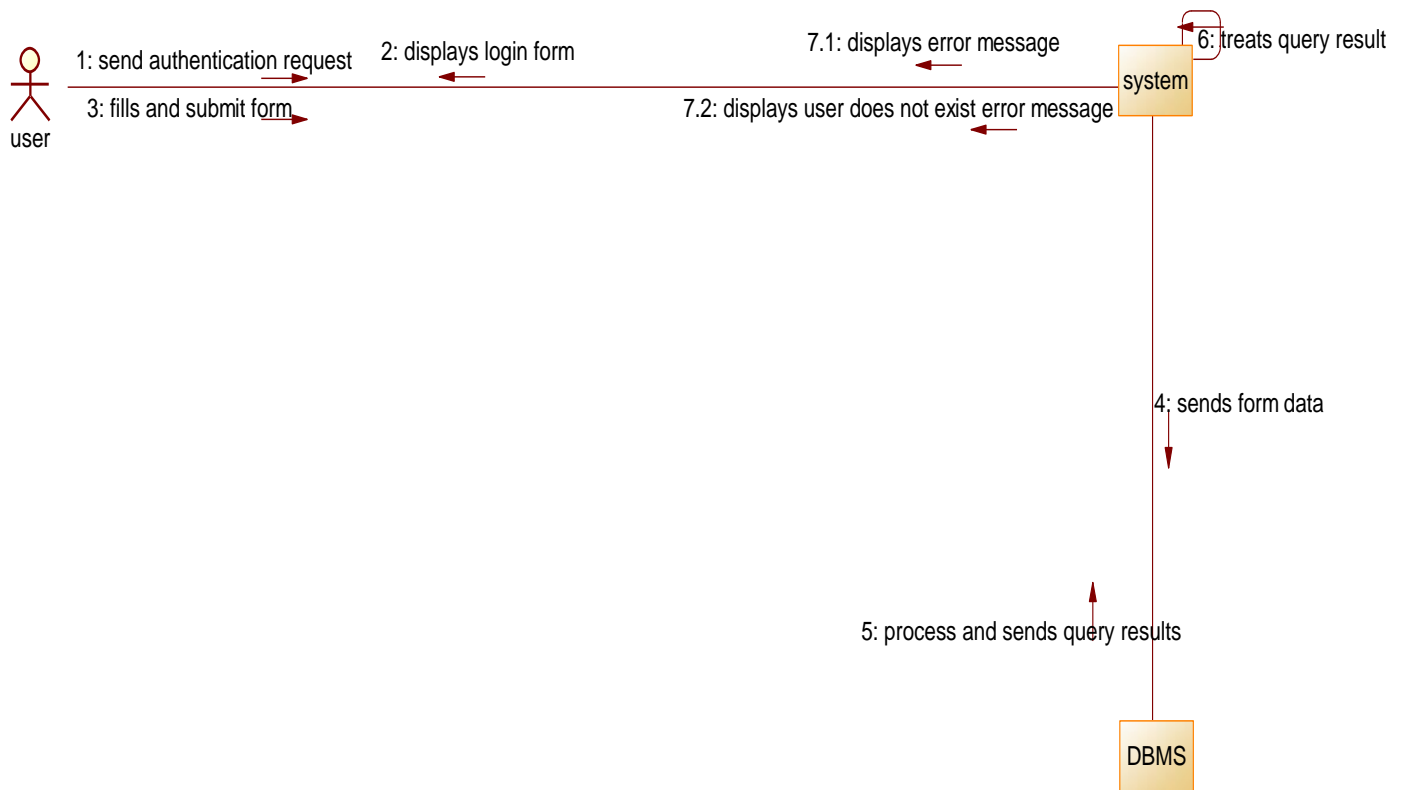


Figure 16: Authenticate Communication Diagram

e. Book appointment communication diagram

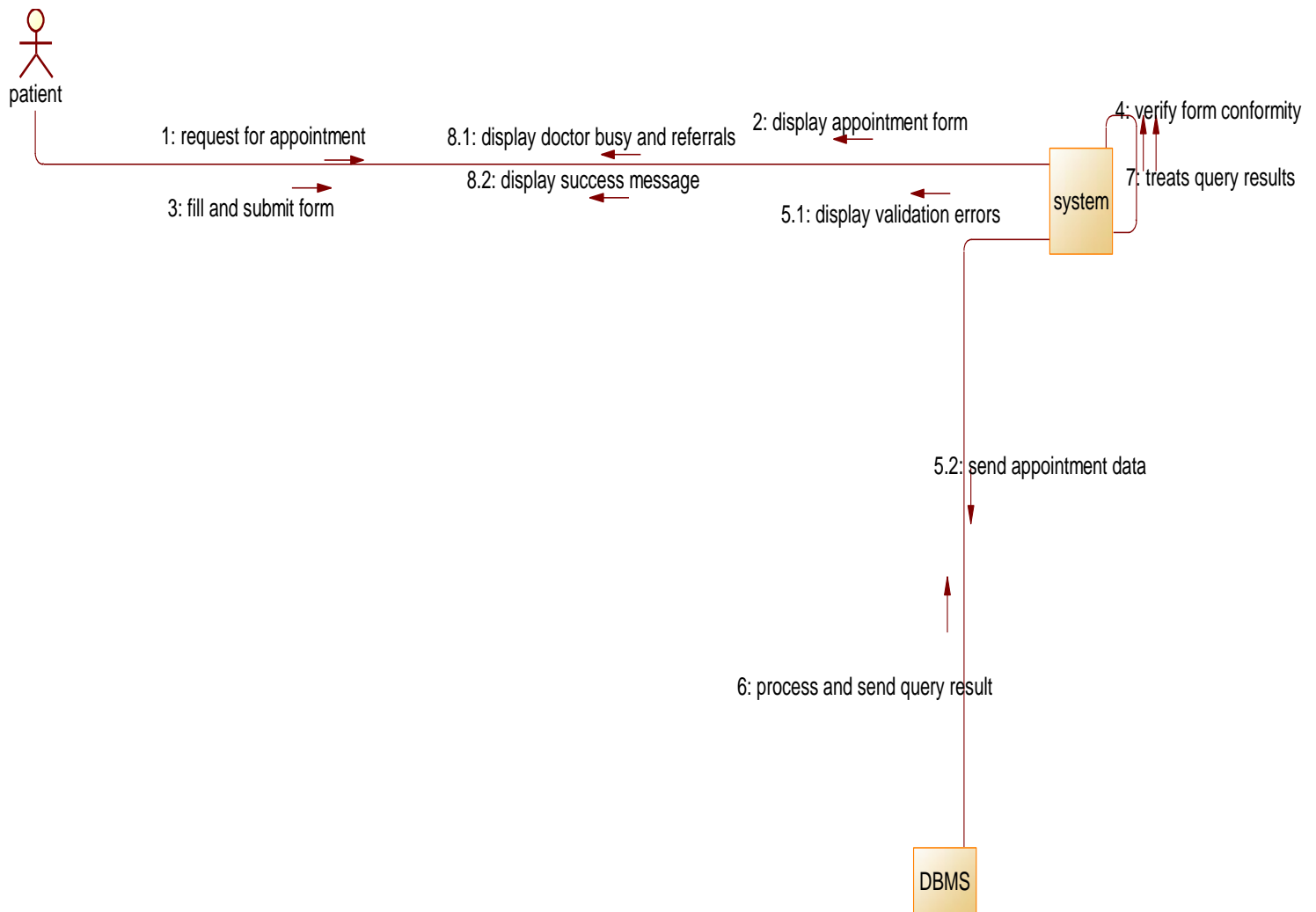


Figure 17 Book appointment communication diagram

3. SEQUENCE DIAGRAM

a. Definition

A Sequence diagram describes interactions among classes in terms of an exchange of messages over time. They are also called event diagrams. A Sequence diagram is a good way to visualize and validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modelling a new system.

b. Formalism

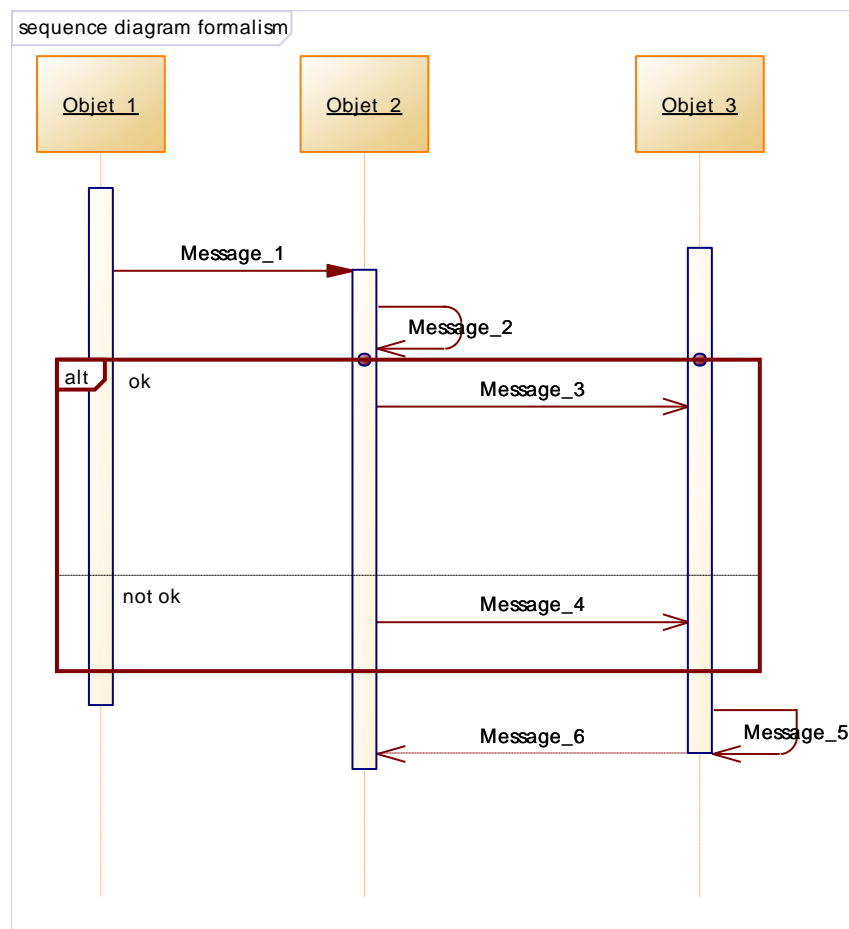

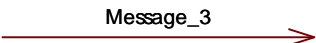
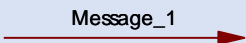

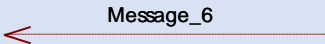



Figure 18: Formalism of sequence diagram

c. Components of Sequence Diagram

Table 15: Elements of sequence diagram

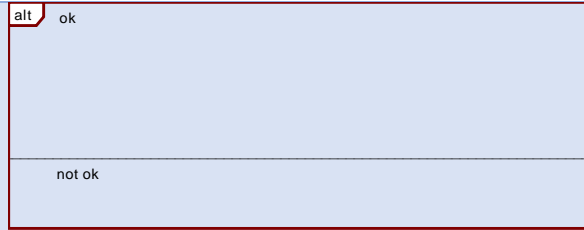
ELEMENT	NOTATION	DESCRIPTION
Lifelines		They represent rows or objects instances that participate in the sequence being modelled.
Asynchronous Message		It is a message that receives an indirect response.
Synchronous Message		It is a message that sends and want response before it continues a process
Self-Message		self-message can represent a recursive call of an operation, or one method calling another method belonging to the same object.
Return Message		It represents the response of a message.
Actor		They send and receive message.



DEVELOPMENT OF A PATIENTS' CONSULTATION AND FOLLOW-UP PLATFORM FOR CAMEROON GOVERNMENT HOSPITALS



Combined Interaction Fragment



An articulation of interaction diagram, defined by an operator and operands.

d. Authenticate sequence diagram

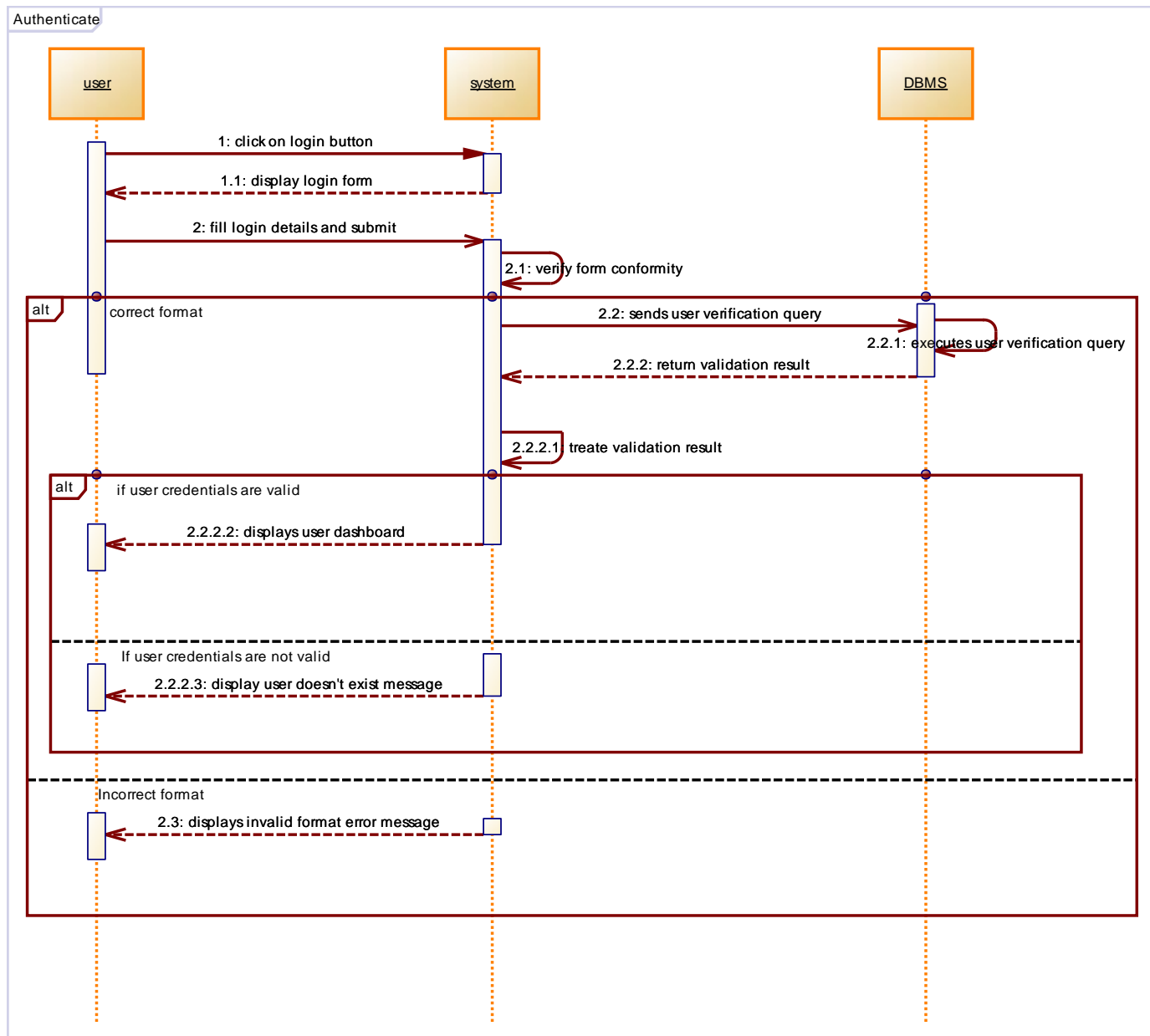


Figure 19: Authenticate Sequence Diagram

e. Book appointment sequence diagram

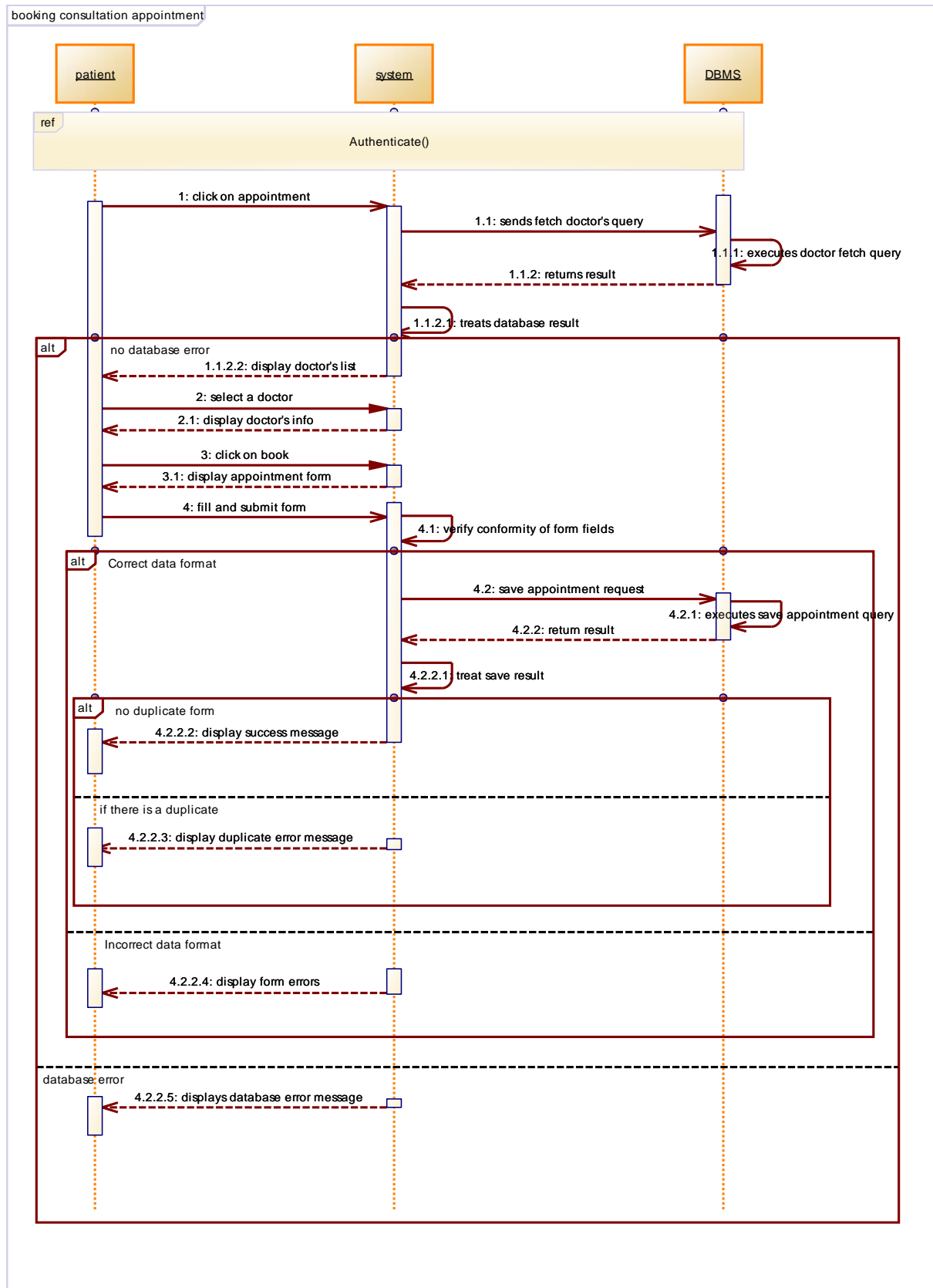


Figure 20 Book appointment sequence diagram

4. ACTIVITY DIAGRAM

a. Definition

An activity diagram is a graphical representation of workflows that show the steps needed in the realization of a process; showing the details from a start point to an end point through all decisions and actions that can possibly be performed. Activity diagrams are intended to model both the computational and organizational process. They flow can be sequential, branched or concurrent. Below is an activity diagram formalism.

b. Formalism

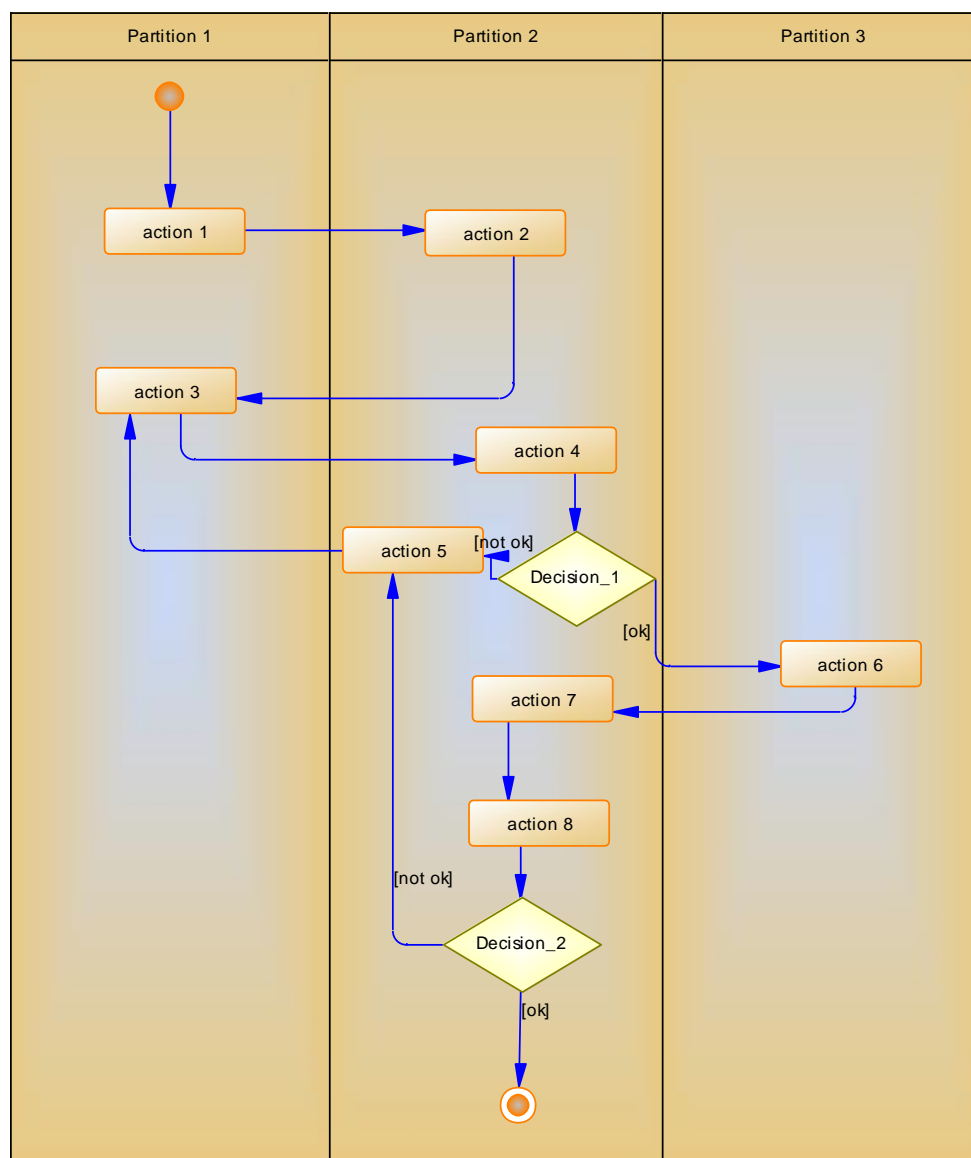





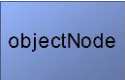
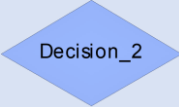
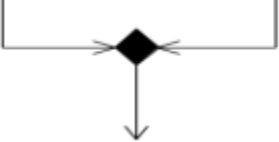
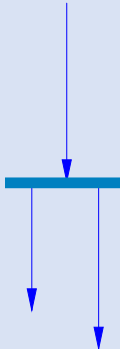
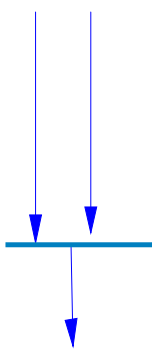
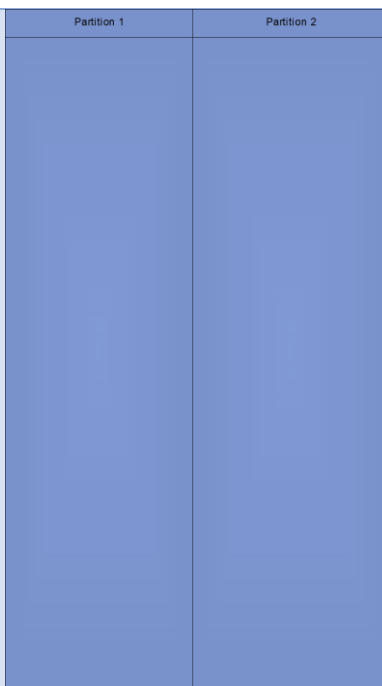


Figure 21: Formalism of activity diagram

c. Elements of activity diagram

Table 16: Elements of activity diagram

Element	Diagrammatic Representation	Description
Activity		Used to represent a set of actions.
Action		Represent a task to be performed.
Activity edge		A directed connection between two activity nodes through which tokens may flow
Initial node		Shows the beginning of an activity or set of actions.
Final node		Stops all controls and object flows in an activity.
Object node		Represents an object connected to a series of object flows.
Decision node		Represents a test condition that splits an incoming activity edge into opposite outgoing activity edges.
Merge node		Reunite different decision paths created using a decision node.

Fork node		Slits behaviour into parallel or concurrent flows of activities (or actions).
Join node		Unites a set of parallel or concurrent flows of activities or actions.
Swimlane and partition	<div> <div>Partition 1</div> <div>Partition 2</div> </div> 	A way of grouping activities performed by the same actor in an activity diagram or to group actions in the same thread.

d. Authenticate activity diagram

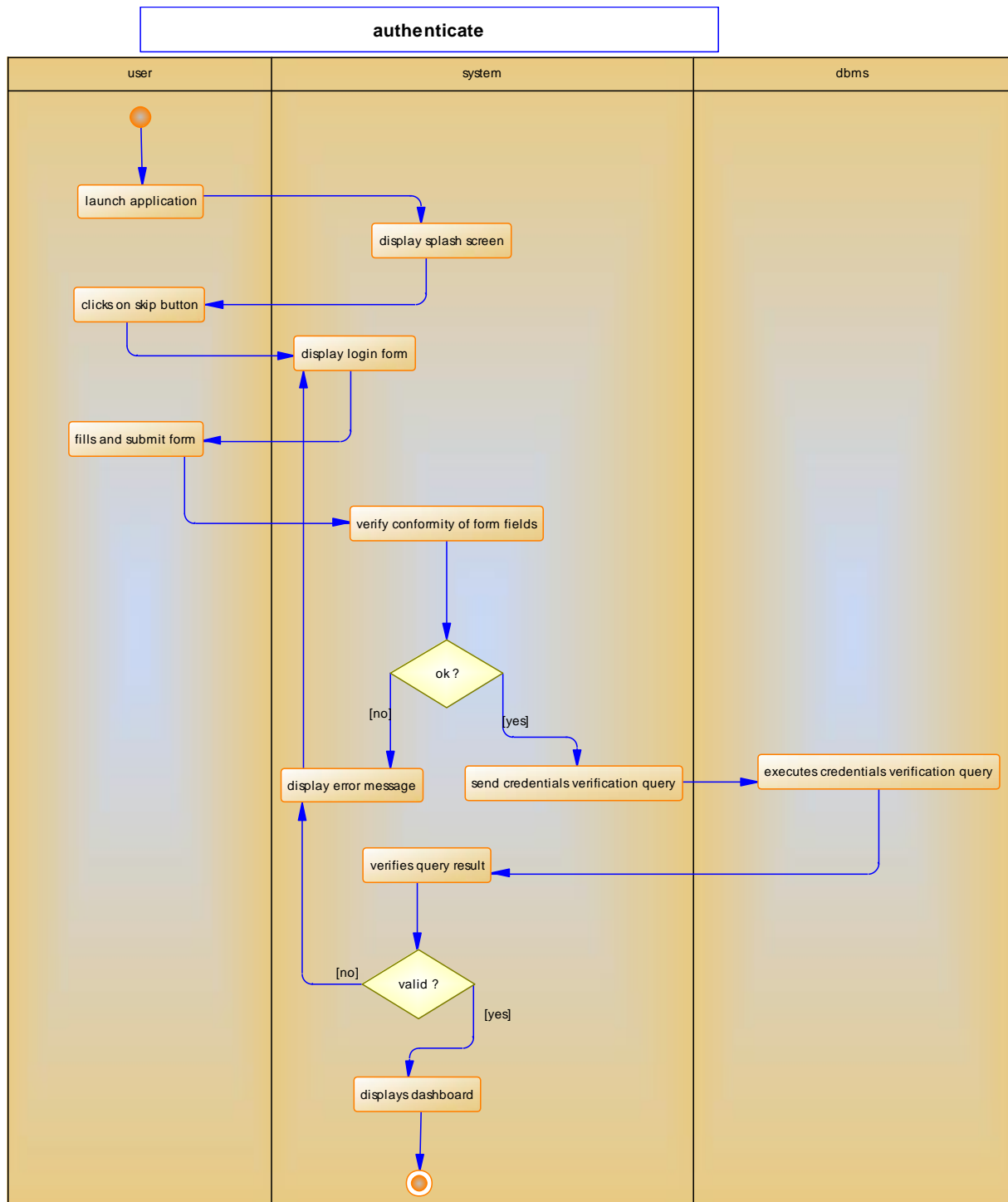


Figure 22 Authenticate activity diagram

e. Consultation activity diagram

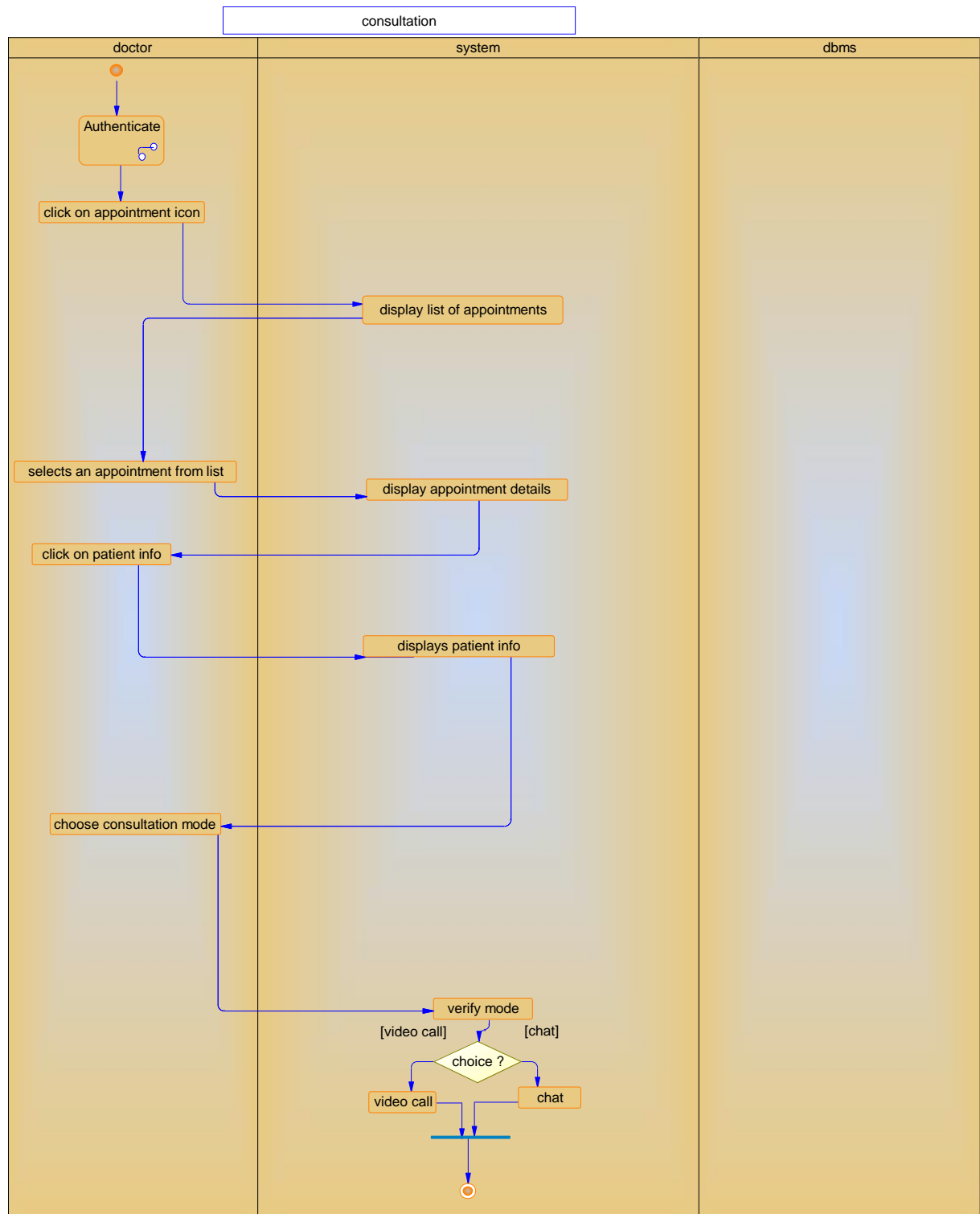


Figure 23: Consultation activity diagram

f. Book appointment activity diagram

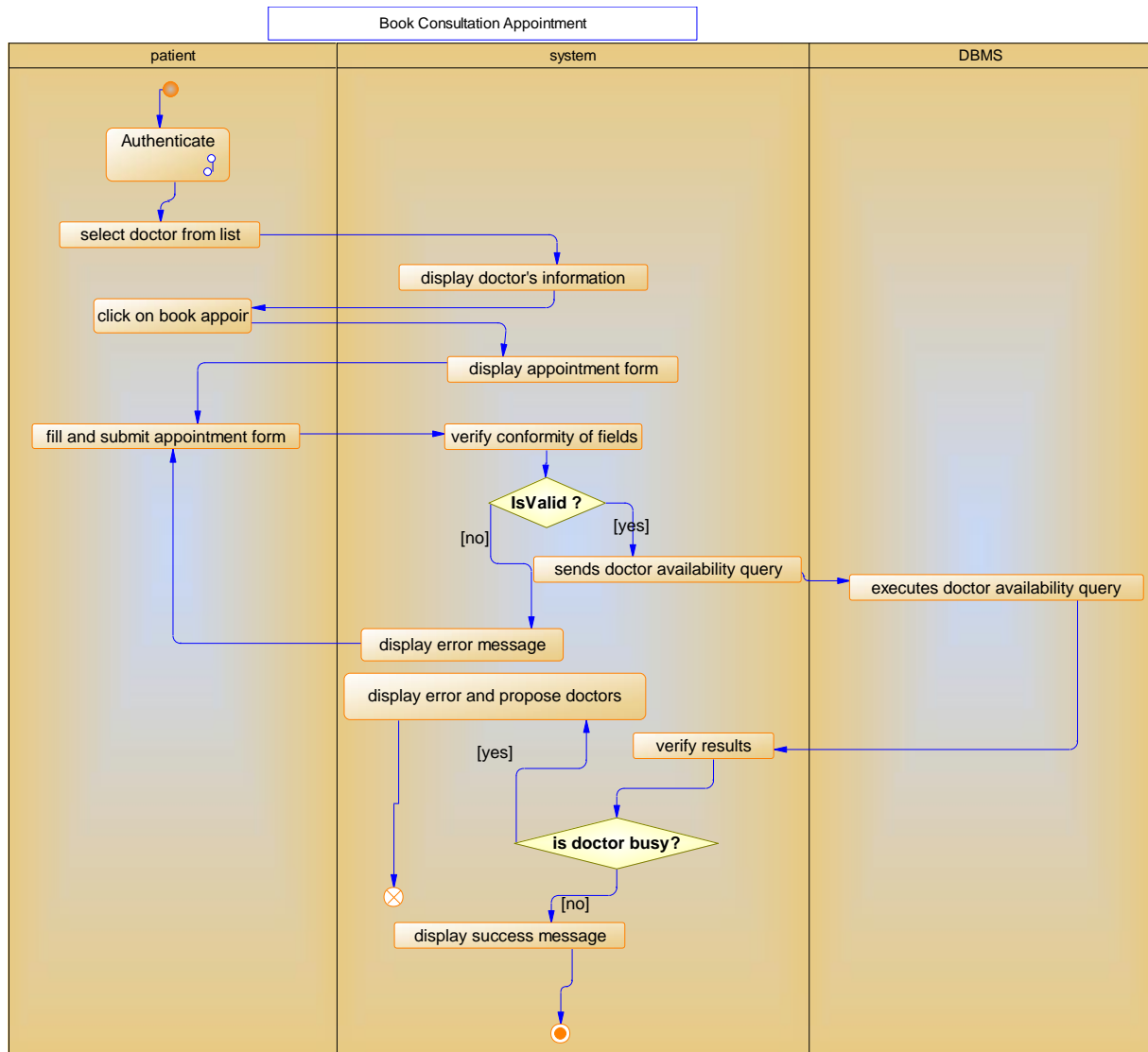


Figure 24 Book appointment activity diagram

CONCLUSION

In the analysis phase, we chose a software development process and modelling language, after which we explained the functional need of our system, we saw the use case diagram which shows the relationship between the actors and use case (the action the actor can perform on the system), we saw the communication diagram which represents the architecture of the system based on object oriented programming, we saw the sequence diagram which represents the flow of messages between elements in the system, and lastly the activity diagram which shows the workflow of our system. We will now move to the conception phase in which we will present the Technical branch of our system together with related diagrams.



PART IV

CONCEPTION PHASE

Preamble

The conception phase will permit us to present in an orderly manner the components necessary for the good functioning of our software and also the architecture used for the proposed solution. It bridges the gap between the analysis phase and the realization phase.

Content

INTRODUCTION

1. TECHNICAL BRANCH

A. GENERIC DESIGN

B. CAPTURE OF TECHNICAL NEEDS

C. RELATED UML DIAGRAMS

i. CLASS DIAGRAM

ii. STATE MACHINE DIAGRAM

iii. PACKAGE DIAGRAM

CONCLUSION

INTRODUCTION

The conceptual phase will describe in detail the necessary specifications, features and operations that will satisfy the functioning requirements of the proposed system as modelled in the analysis phase. This phase is meant to identify and consider essential components (hardware /or software), structure (network capabilities), processes and procedures for the system to accomplish its objectives. We will look at some diagrams such as the class diagram, state machine diagram and package diagram.

A. GENERIC DESIGN

a. Hardware diagram of the system

The hardware diagram simply shows how the system components of our system are deployed; it shows the positioning of each device into its right proportion.

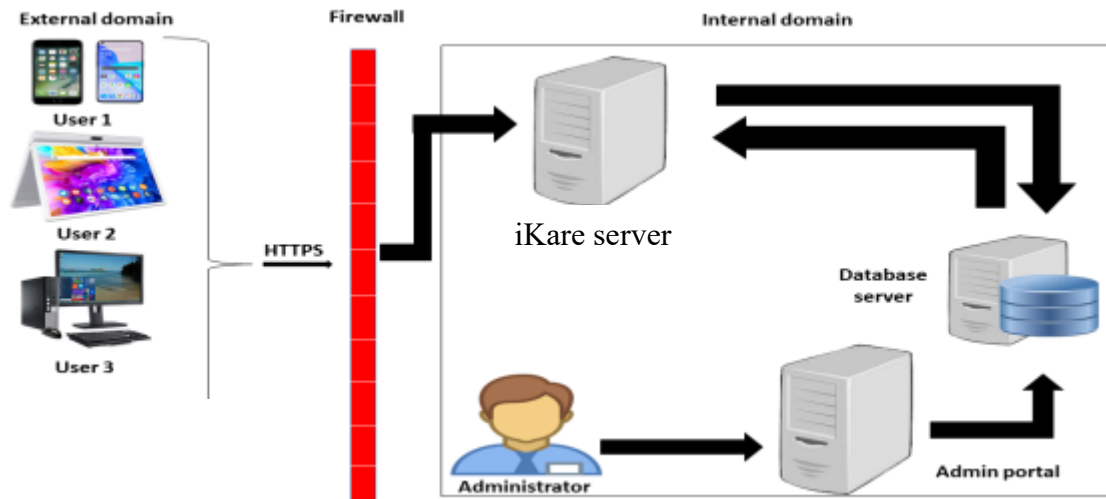


Figure 25: Hardware diagram of the system

b High Level Architecture of the System

The high-level architecture diagram provides an overview of the entire system, identifying the main components that would be developed for the product and their interfaces.

B. CAPTURE OF TECHNICAL NEEDS

a. Physical Architecture

The design of the DBMS depends on its architecture. An n-tier architecture partitions the whole system into related but separated n modules, which can be independently modified, altered, changed or replaced. A large amount of data on web servers, personal computers (pc) and others are linked with networks with the help of basic client or server architecture.

Within the scope of our project, we made use of the n-tier architecture. This architecture separates its tiers from each other based upon the user and the manipulated data in the database. Each layer has a well-defined communication interface, and the evolution of the layer is independent of the other. The n-tier of our system is made up of:

- ✚ The hardware tier, which represents our IOT device to collect vitals and it is connected directly to our mobile phone using a wired connection.
- ✚ The client tier, which is also known as our presentation interphase.
- ✚ Application Tier, which represents our webserver.
- ✚ The data tier, which represents our DBMS server.

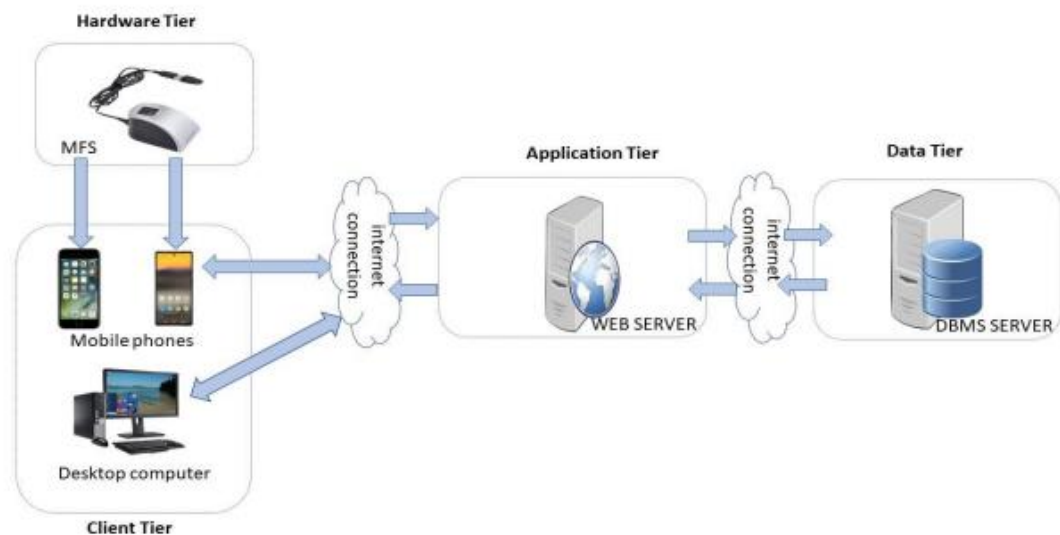


Figure 26: n-tier Architecture (Source: <https://www.pinterest.es/pin/752241943987565648/>)

b. Logical Architecture

Model View controller or MVC as it is popularly called, is a software design pattern for developing application. A model view controller pattern is made up of the following three parts.

- ❖ **Model:** The lowest level of the pattern which is responsible for maintaining data.
- ❖ **View:** This is responsible for displaying all or a portion of data to the user.
- ❖ **Controller:** It handles software codes that controls the interactions between the model and the view.

MVC is popular as it isolates the application logic from the user interface and supports separation of concerns. Here the controller receives all requests for the application then works with the model to prepare data needed by the view. The view then uses the data prepared by the controller to produce a final response. The MVC can be represented as follows :

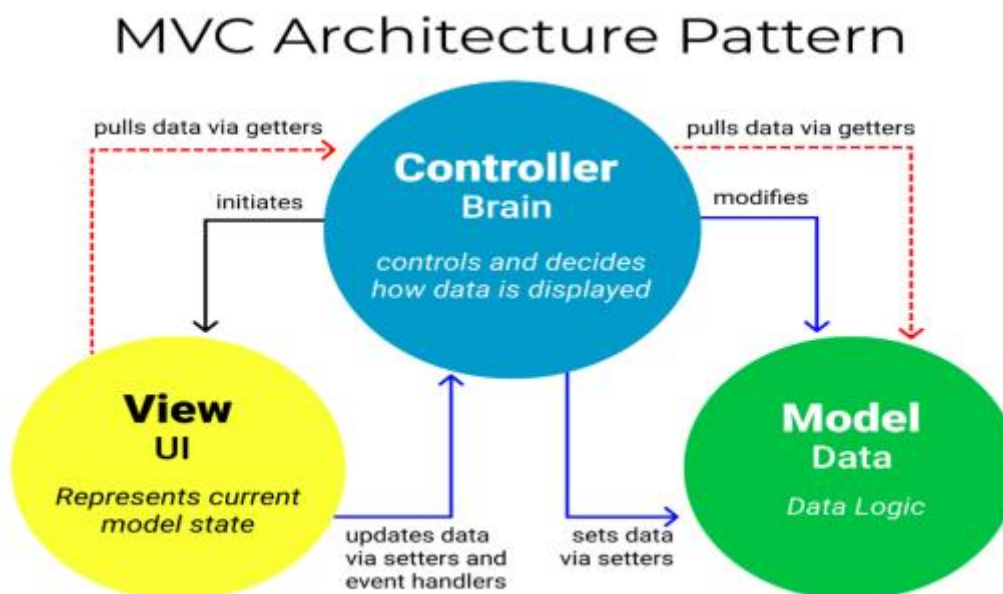


Figure 27: The MVC architecture (Source: <https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/>)

C. RELATED UML DIAGRAMS

1. CLASS DIAGRAM

a. Definition

A class diagram is a static diagram. It represents the static view of an application. class diagram is not only used for visualizing, describing and documenting different aspect of the system but also for constructing executable code of the software application. Class diagram describes the attribute and operation of a class and constraints imposed on the system. Its purpose is to model the static view of an application.

b. Formalism

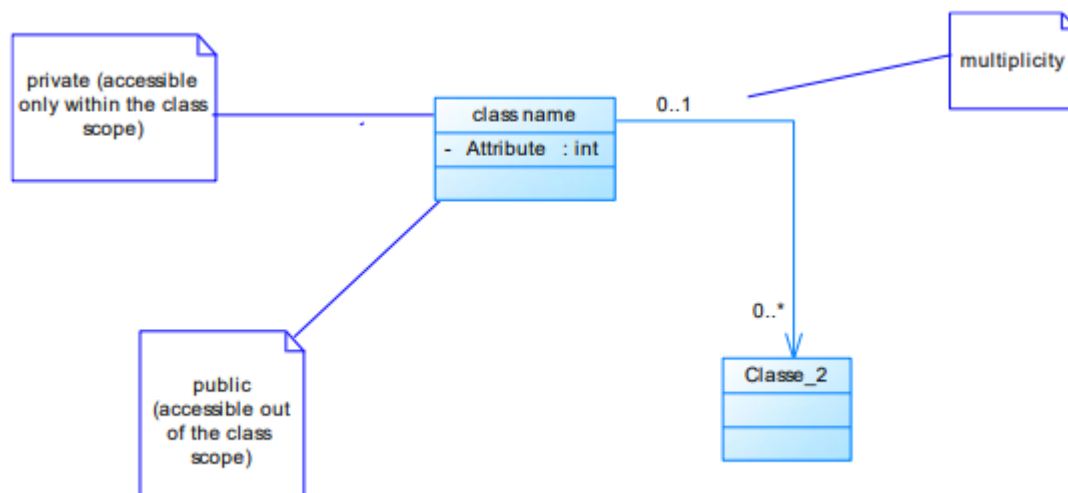
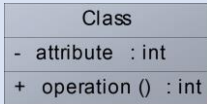



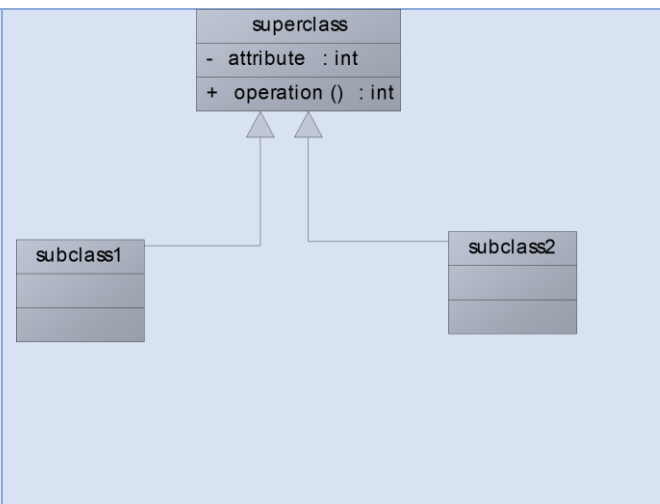
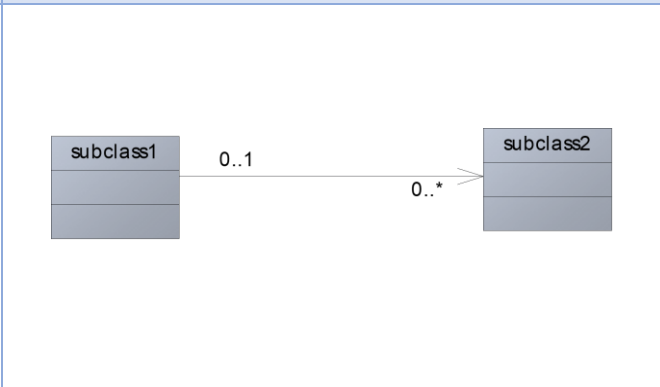


Figure 28: Formalism of Class Diagram

Table 17: Elements of class diagram

Element	Representation	Description
Class		A class is an element that defines the attributes and behaviors that an object can generate
Composition		If a parent of a composite is deleted, usually, all its parts are deleted with it.
Aggregation		If the parent of the aggregate is deleted, usually the children are not deleted.
Dependency		It existed between two classes, if one changes it may cause the change in the order, but the other way around

Generalization		<p>it a relationship between a whole thing (called superclass) and a more specific thing (called subclass)</p>
Association		<p>It is a general type of relationship between elements, it may include cardinality, roles etc.</p>

d. Business Rules

R1: A user manages one or more notifications, and a notification is managed by only one user.

R2: A user owns one role only, and a role is owned by one or more users.

R3: A user manages one or more requests, and a request is managed by one or more lab technicians.

R4: A Super Admin administers one or more hospital user and a hospital user is administered by one super admin.

R5: A hospital contains zero or many users and a user is found in zero or many hospitals.

R6: A user manages zero or many appointments and an appointment is managed by only one user.

R6: A user manages zero or many EHRs and an EHR is managed by only one user.

R8: An appointment leads to zero or many consultations and a consultation follows only one appointment.

R9: A prescription follows one or more consultations, and a consultation leads to one or more prescriptions.

R10: A follow-up form is issued after one or more consultationprescriptions and a consultationprescription leads to one or more follow-up forms.

R11: An EHR contains one or more documents, and a document is found in only one EHR.

2. STATE MACHINE DIAGRAM

a. Definition

A state machine diagram describes the behaviour of a single object in response to a series of events in a system. Also known as the state machine diagram, it models the dynamic flow of control from the state of a particular object within a system.

b. Formalism

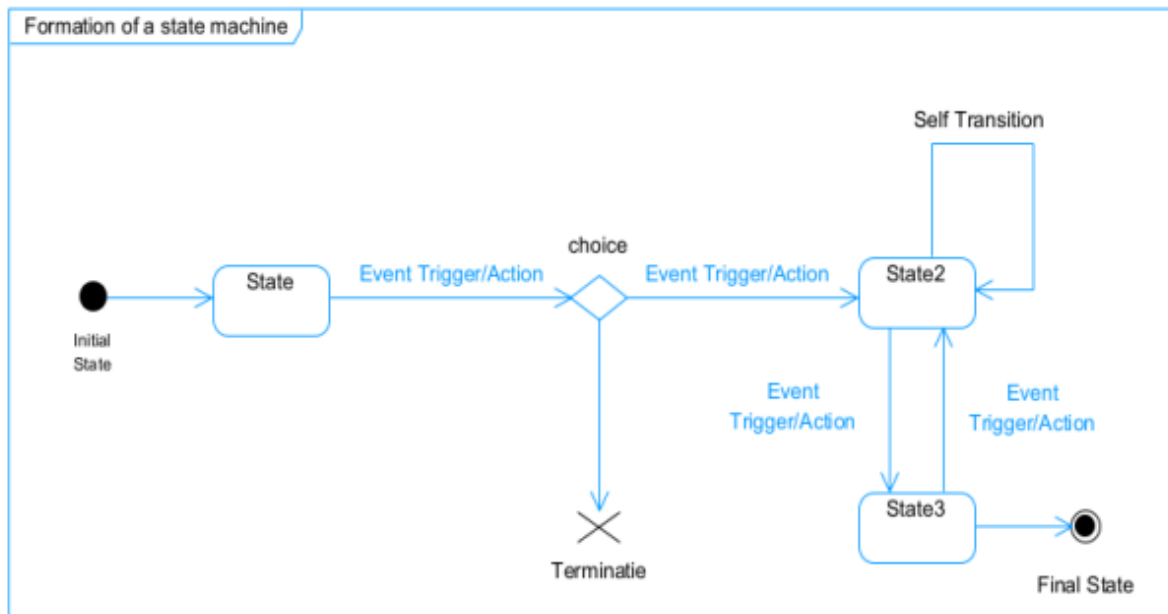



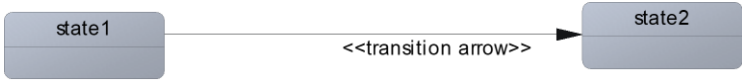


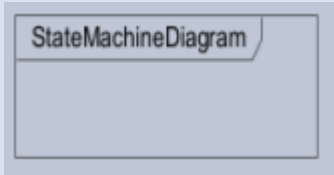


Figure 30: Formalism of state machine diagram

Table 18: Elements of state machine diagram

Element	Representation	Description
State		Models a situation during which a certain invariant condition holds.
First (Initial State)		It represents a default vertex, that is, a source for a single transaction to the default or composite state.
Final State		A state specifying that the enclosing region is complete.
Transition		A direction relation between a source and a target vertex.
Choice pseudo state		A diamond symbol that indicates a dynamic condition with branched potential results
Terminate		Implies that the execution of a

		state by means of it context is terminated.
Diagram overview		A placeholder for the linked states in a state machine diagram.

c. Account state machine diagram

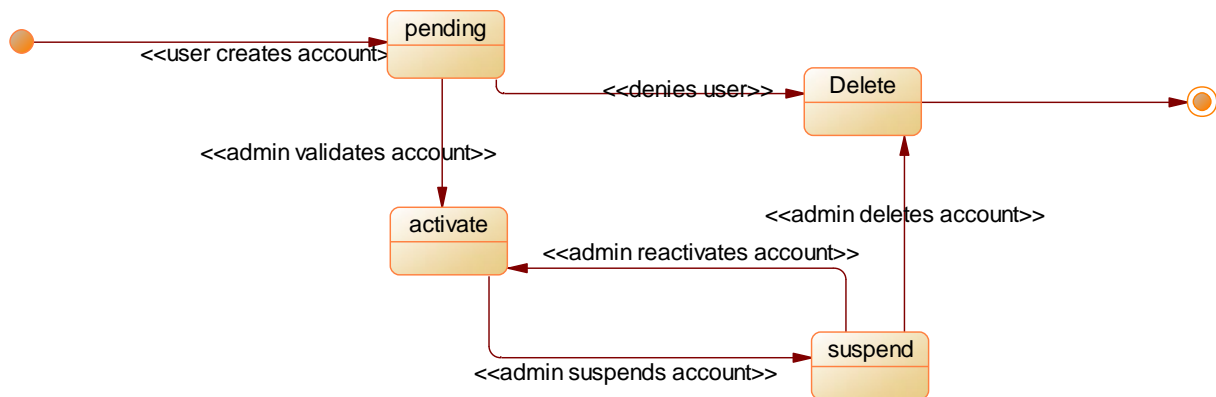


Figure 31 Account state machine diagram

d. Consultation state machine diagram

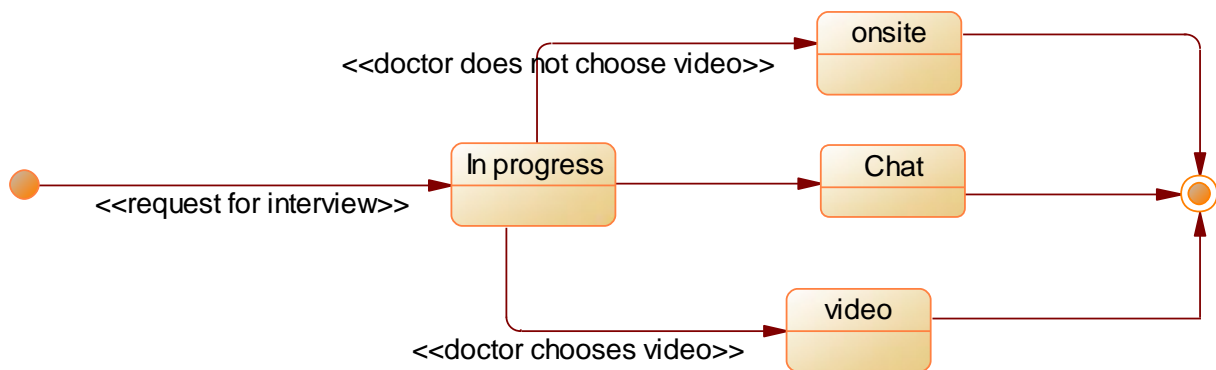


Figure 32 Consultation state machine diagram

e. Appointment state machine diagram

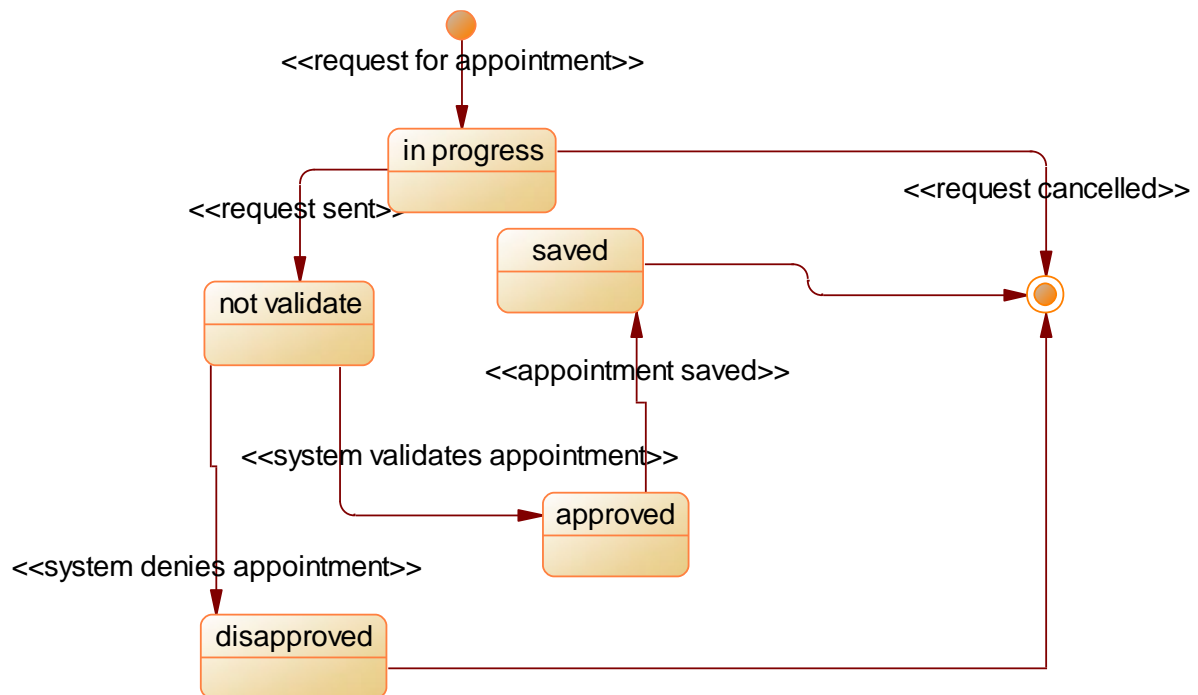


Figure 33 Appointment state machine diagram

3. PACKAGE DIAGRAM

a. Definition

This is a structural diagram used to show the organization and arrangement of various model elements in the form of packages. A package diagram is the grouping of related uml elements such as classes, diagrams or even other packages.

b. Formalism

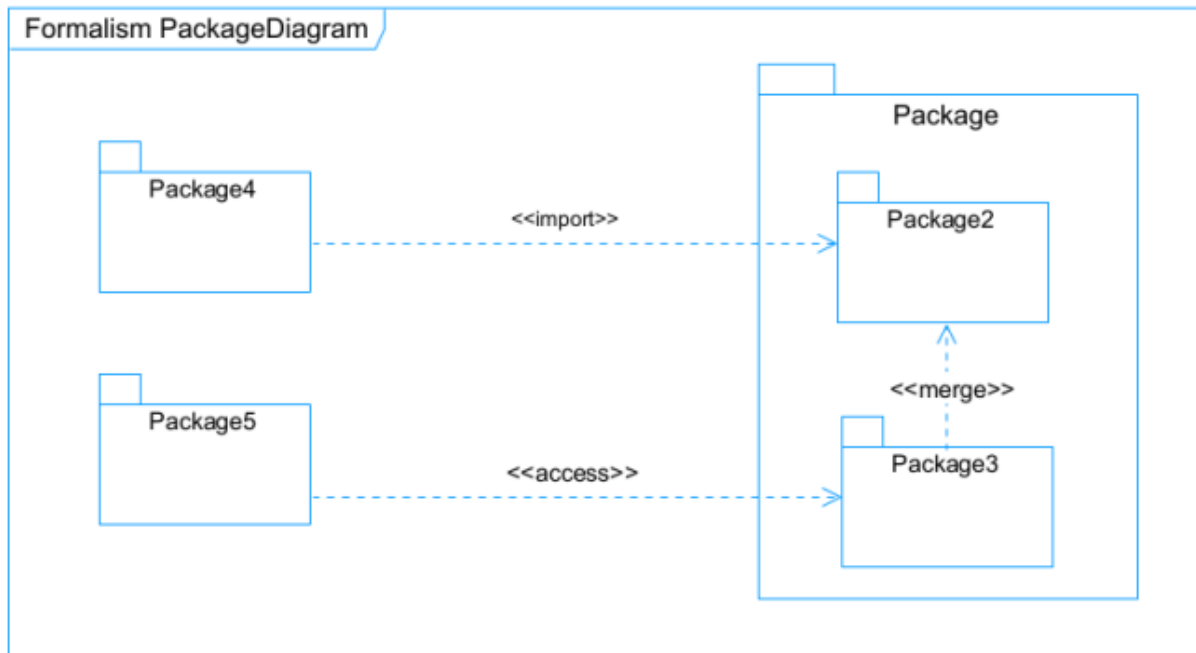
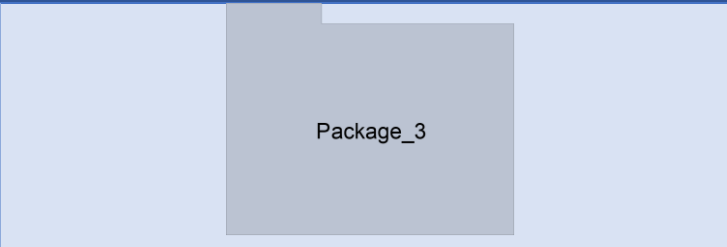
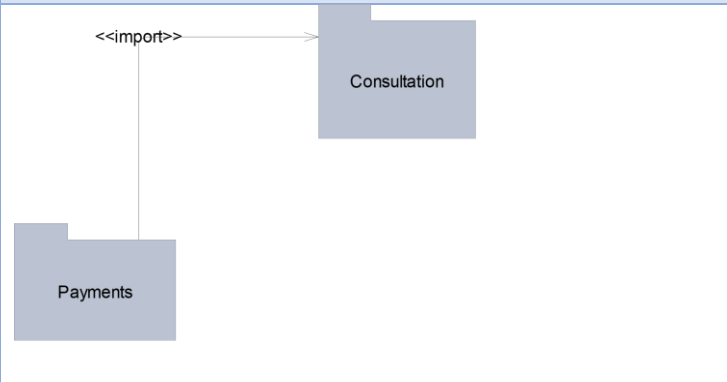
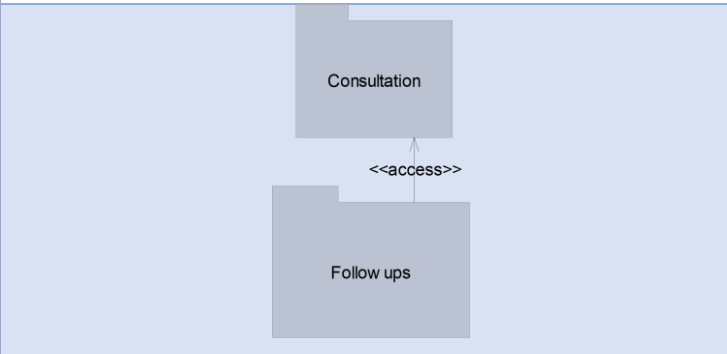
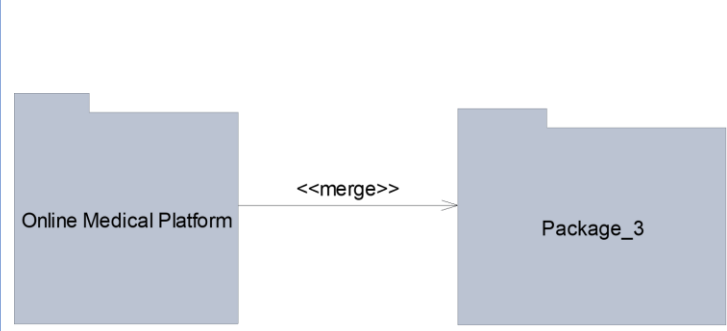


Figure 34: Formalism of package diagram

Table 19: Elements of package diagram

Element	Representation	Description
Package		A package is a namespace use to group related elements;
Package import		A relationship Indicate that, functionality has been imported from one package to another.
Package access		A relationship Indicates that one package requires assistance from the function of another package.
Package merge		It is a relationship which shows that, the functionality of two packages are combines to a single function.

c. iKare package diagram

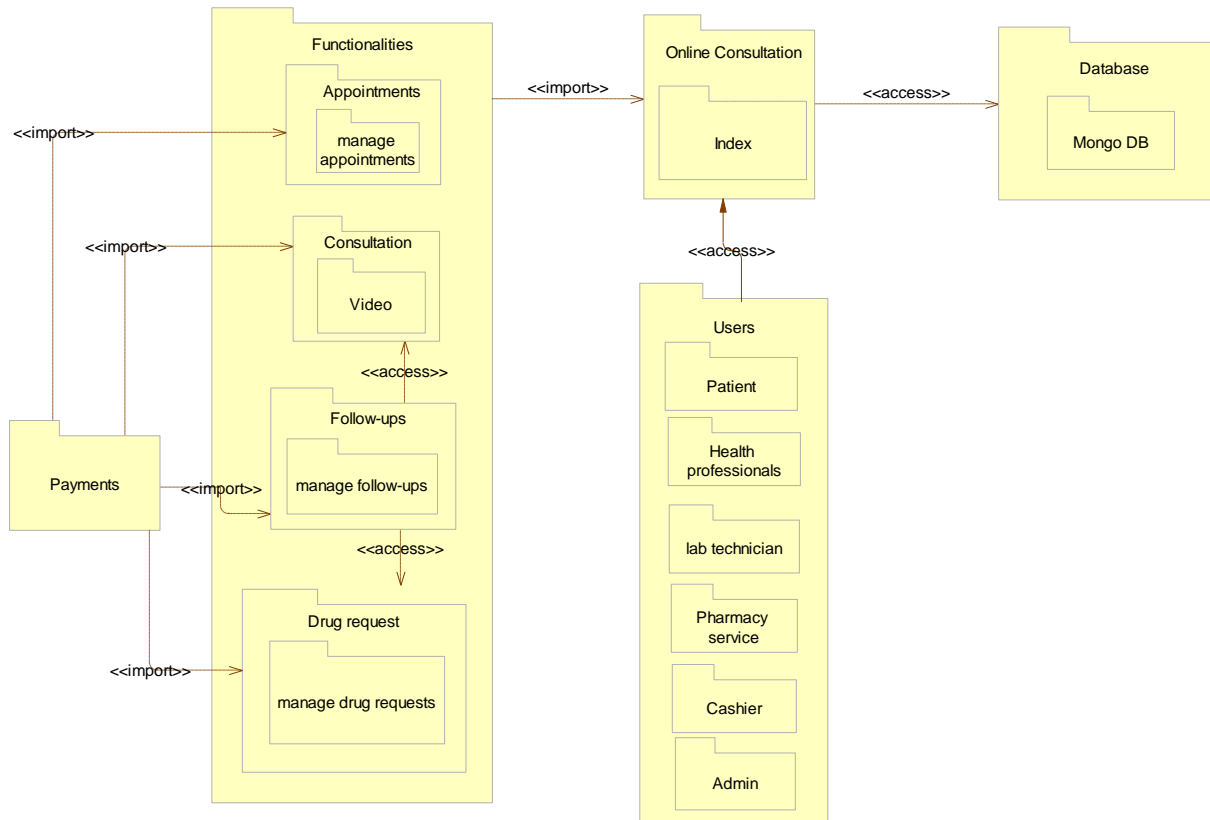


Figure 35 iKare package diagram

CONCLUSION

In the conception phase, we set as objective to plane the different aspect of our system by showing how it will be structure and deployed within existing technical architectures. We began by considering the technical constraints for our system, after which we proceeded to identify the components of our system, how they are grouped together and how they should be deployed on appropriate deployment targets. We finished this phase by looking at interactions between the various aspects and actors of our system. The next phase of our report is the realization phase where we will look at aspects concerning the implementation of our system.



PART V

REALIZATION PHASE

Preamble

In this phase we will to straight forward in the implementation of our solution, we will base ourselves on the analysis and conception phases and also present the component and deployment diagrams.

Content

INTRODUCTION

- I. DEPLOYMENT DIAGRAM
- II. COMPONENT DIAGRAM

CONCLUSION



INTRODUCTION

Here in the realization phase, we will see some diagrams related to the physical aspect of the system like libraries, documents, as well as the physical topology of the components of the system when the software is been deployed.

1. DEPLOYMENT DIAGRAM

a. Definition

Deployment diagram is a structural diagram used to visualize the topology of the physical components of a system, where the software is deployed. They consist of nodes and their relationship. It is related to the component diagram because the components are deployed using the deployment diagram. A deployment diagram consists of nodes. Nodes are nothing but physical hardware used to deploy the application.

b. Formalism

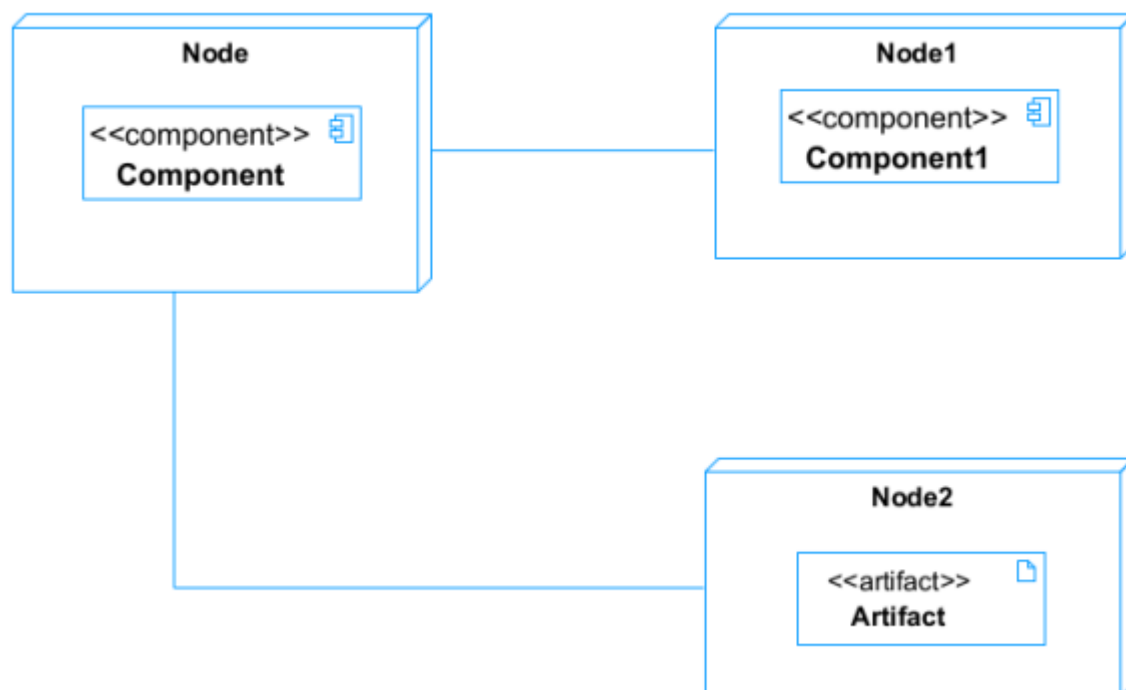
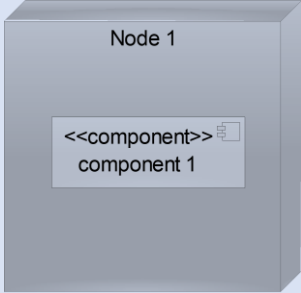
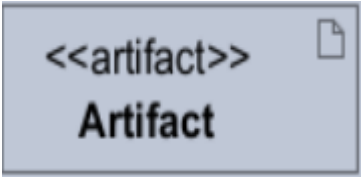
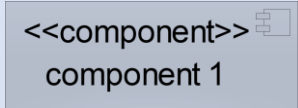



Figure 36: Formalism of deployment diagram

Table 20: Elements of deployment diagram

Element	Representation	Description
Node		It is a hardware used to deploy the application
Artifact		An artifact is a major product, which is produced or used during the development of a software. E.g diagrams, data models, setup scripts
Component		It represents a modular part of a system that encapsulates its content and whose manifestation is replaceable within its environment.
Association		An association helps to connect two nodes together which permits them to communicate together

c. System deployment diagram

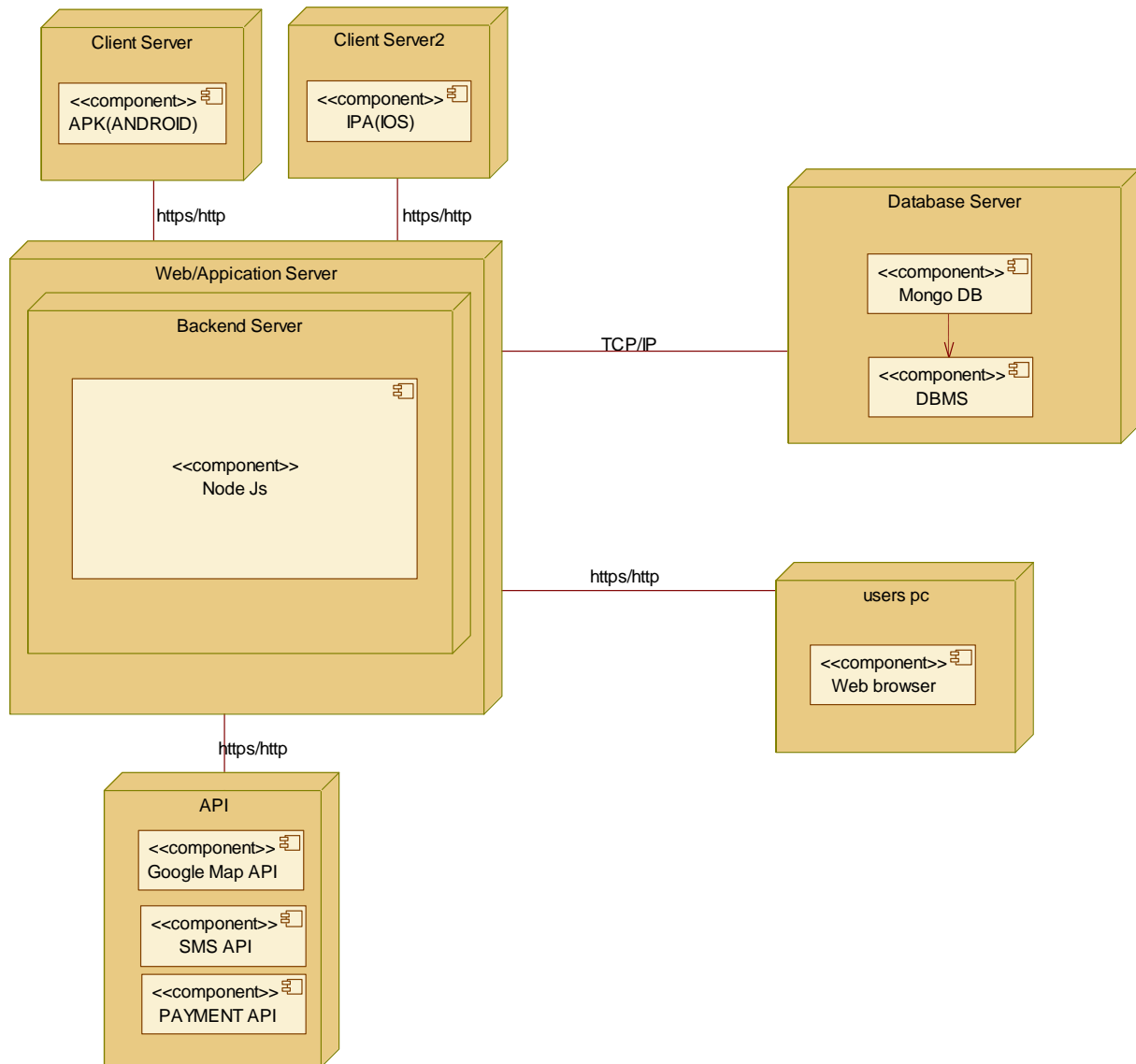
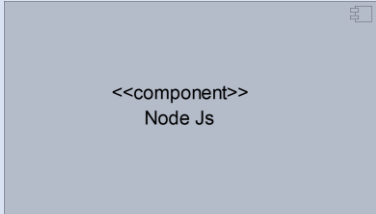

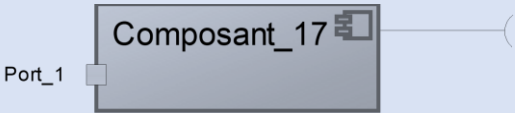




Figure 37 iKare deployment diagram

Table 21: Elements of component diagram

Name	Representation	Description
A Component		A component is an abstract logical unit block of a system. It is represented as a rectangle with a smaller rectangle in the upper right corner which serves as its icon for recognition.
Dependency		Dependency is a directed relationship which is used to show that some components are dependent on others for their correct functioning.
Required Interface		It is a straight line from the component box with an attached half circle representing interfaces where a component requires information in order to perform its own functions.
Provided Interface		It is a straight line from the component box with an attached circle representing interfaces where a component produces information used by required interfaces.
Port		A port (represented by a small square at the end of a

required or provided interface) is used when the components delegate the interfaces to an internal class.

d. System mobile component diagram

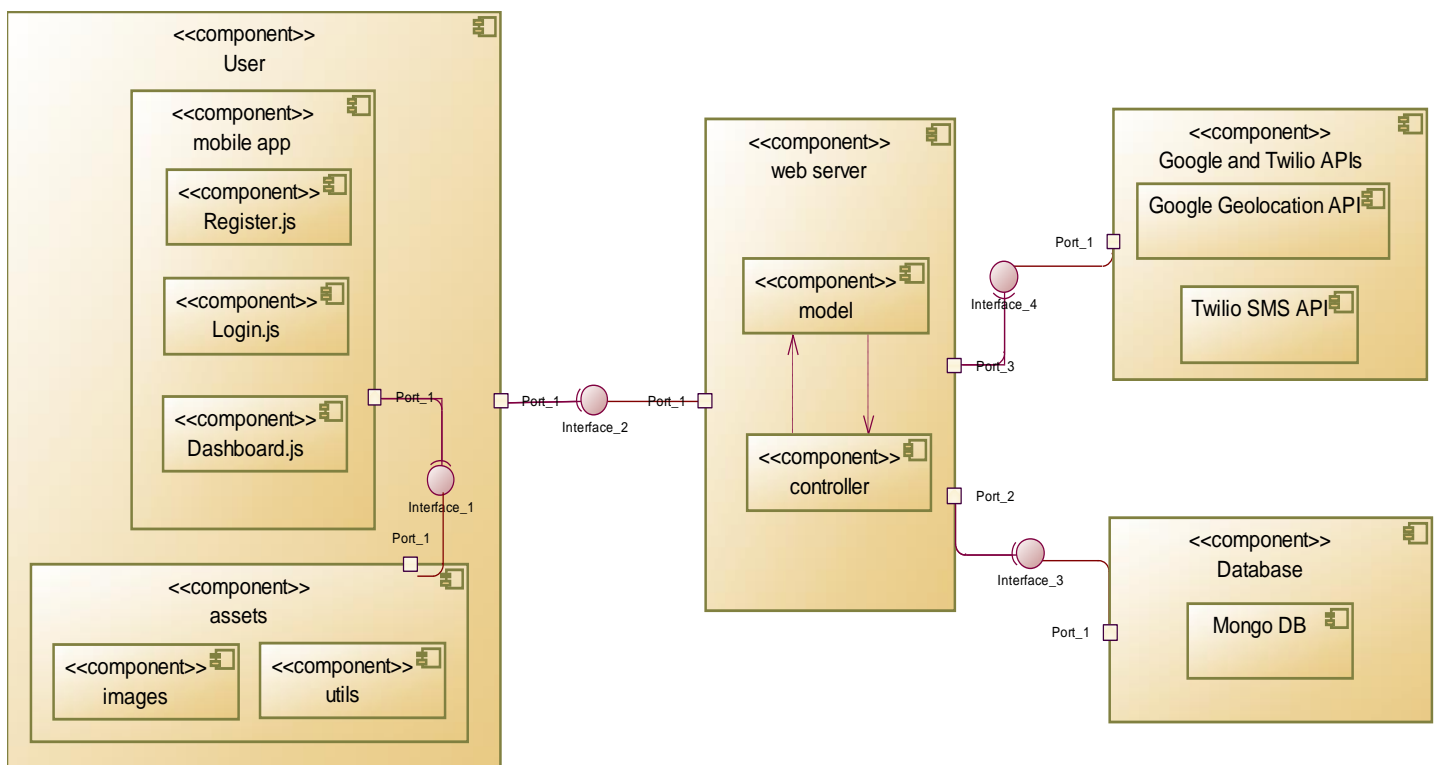


Figure 39 iKare mobile component diagram

e. System web component diagram

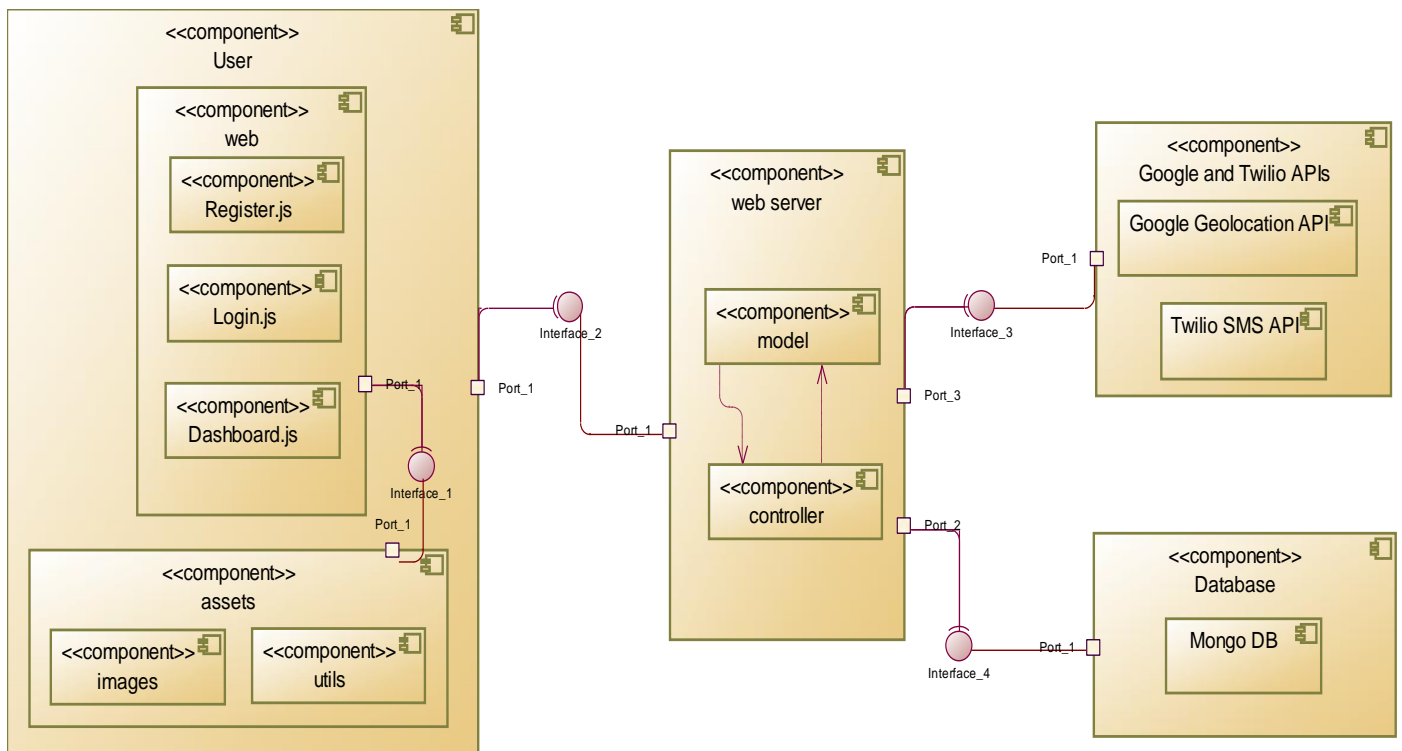


Figure 40 iKare mobile component diagram



CONCLUSION

In our realization phase, we implemented our application. In other to accomplish this phase, we made use of our analysis and conception phase. We also drew the deployment and component diagrams which depict the structure of our system in terms of modules, files, assets, how the different elements interact with each other. We will move to the test of functionalities phase, where we will examine the different modules present in our app and how beneficial they are to its different users.



PART VI

TEST OF FUNCTIONALITIES

Preamble

In this phase, we will present the various functionalities of our application.

Content

INTRODUCTION

1. APPLICATION FUNCTIONALITIES
2. TESTS SHOWCASES

CONCLUSION



INTRODUCTION

The test of functionalities phase helps us to know more about the solution we are building be it web or mobile. It provides the different functionalities or modules found in our application and how they are beneficial to the users. Hence, we are going to explore the different functionalities present in iKare.

1. APPLICATION FUNCTIONALITIES

+ Authentication

This functionality enables the users to have access to their workspace or dashboard. In case he/she does not have an account he/she will register.

+ Consultation

It enables communication between the health professional and the patient through video calls or chat for diagnosis.

+ Book and Manage Appointments

It helps patients book appointments for consultation and health professionals to either edit, cancel, approve the appointments received from patients.

+ Pharmacy Search

At the end of a consultation, the health professional often issues prescriptions to patients, this module helps patients quickly identify the pharmacies nearby, having the drugs they are looking for and opened at the time.

+ Follow-ups

When drugs have been taken, a regular follow-up is needed to make sure that the drugs taken by the patients have the desired effect on them. If the patient is not relieved from his illness, the doctor will be informed.

+ E-Health Record

This module helps in keeping track of patients' past records, to enable effective diagnosis to be carried out by future health professionals.

+ Account and Notification Management

The users are capable of modifying account details by editing their passwords, profile, name, emails, profile pictures. Notifications are received by doctors and patients to remind them of approaching consultations, account creation (Health staff) and validation, also reminding patients to take drugs.

+ Log out

It enables users to close a working session.

2. TESTS SHOWCASES

The following are the tests carried out using mocha, chai and chai-http on our APIs' endpoints.

i) Admin Modules tests

Here we have tested :

- ❖ `/api/admin/getAdmins` routes that enables the super administrator to get the list of all the admins created on the platform.
- ❖ `/api/admin/createAdmin` that enables the super admin to create an admin. If an admin already exists with that email address, it returns an error message. In this our case, there is a technical issue (due to network connectivity) reason why there is one 1 failing.

```
D:\Web_App_Development\End_of_year\iKare_Logic>npm test

> ikare_logic@1.0.0 test
> mocha --timeout 10000

Server listening on port: 5000

ADMIN
Connected to db
/GET admin
  ✓ It should return all admins
/POST admin
SMS sent from +12566702929 to +237696953772, message id SMf5efae10694bb22f96a8cba0dab3df8f
  ✓ It should create an admin (4266ms)
/POST admin
  ✓ It should not create an existing hospital
/POST admin
  1) It should not create an existing admin

3 passing (14s)
1 failing

1) ADMIN
  /POST admin
    It should not create an existing admin:
      Error: Timeout of 10000ms exceeded. For async tests and hooks, ensure "done()" is called; if returning a
      f_year\iKare_Logic\test\test.js)
        at listOnTimeout (node:internal/timers:564:17)
        at process.processTimers (node:internal/timers:507:7)
```

Figure 41: Admin Modules tests

ii) Health Professional Creation and Manage modules

Here we tested

- ❖ /api/admin/getProfessionals endpoint that enables an administrator to get all the health professionals registered in that institute.
- ❖ /api/admin/createProfessional endpoint that enables an administrator to create a professional if that matricule is not yet registered, in the reverse case an error message displays.

```
D:\Web_App Development\End_of_year\iKare_Logic>npm test

> ikare_logic@1.0.0 test
> mocha --timeout 10000

Server listening on port: 5000

PROFESSIONAL
Connected to db
/GET doctor
  ✓ It should return all professional
/POST Health Professional
  RestException [Error]: The 'To' number 696953772 is not a valid phone number.
    at success (D:\Web_App Development\End_of_year\iKare_Logic\node_modules\twilio\lib\base\Version.js:79:23)
    at process.processTicksAndRejections (node:internal/process/task_queues:95:5) {
    status: 400,
    code: 21211,
    moreInfo: 'https://www.twilio.com/docs/errors/21211',
    details: undefined
  }
  ✓ It should create a doctor (8656ms)
/POST Health Professional
  ✓ It should not create a doctor with the same matricule, in the same hospital

3 passing (9s)
```

Figure 42: Professional Modules tests



CONCLUSION

In the test of functionalities phase we explained the different functionalities of our application and the benefits to the various users (patient, health professional, admin, lab technician, super admin). This phase is essential in the understanding of the application. We will move to the last phase which is the installation and user guide where we will elaborate on how to install and use **iKare**.



PART VII INSTALLATION GUIDE AND USER GUIDE

Preamble

The purpose of the user guide is to provide users of our platform with step-by-step instructions on how to install and use the system.

Content

INTRODUCTION

I. INSTALLATION OF THE APPLICATION

II. SHOW CASE

CONCLUSION



INTRODUCTION

This is the final phase of our report. In this phase we will walk through the requirements for our system, the necessary installation process, accessing our system and its features, all this in a step-by-step manner to facilitate the setting up of the platform for the first-time users. The steps of different processes will be accompanied by images. After we will showcase our application by viewing the different screens.

I. INSTALLATION OF THE APPLICATION

A. DATABASE SERVER INSTALLATION

To deploy a software, we need to install its database server, in our case MongoDB will be installed in our system which is Windows.



Figure 43 mongodb logo

STEP 1 : Downloading the MongoDB MSI installer package

We go to <https://www.mongodb.com/try/download/community> in order to download MongoDB. Note that we are downloading the community version since it is free and open source.

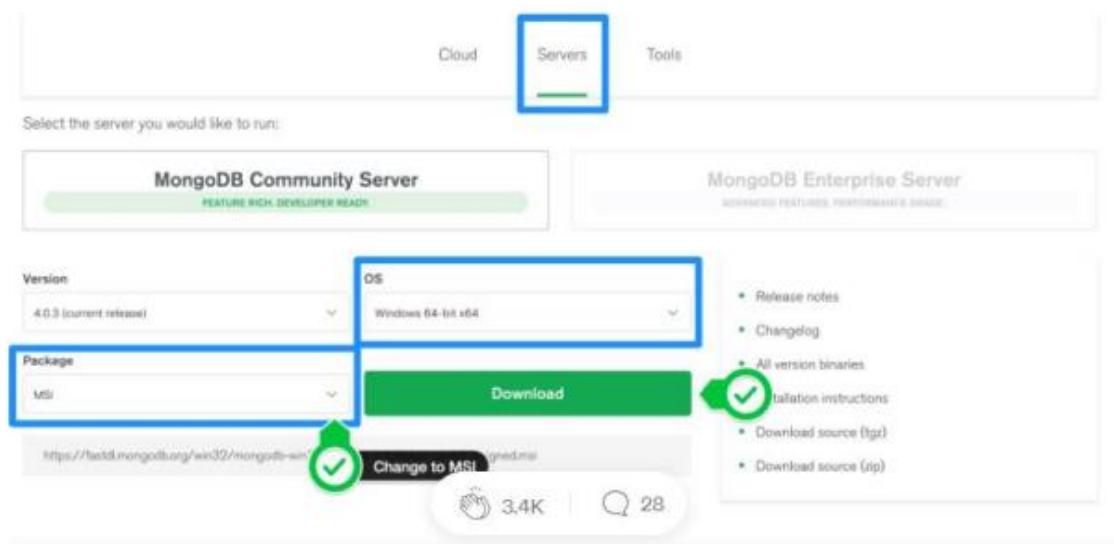


Figure 44 Downloading MongoDB community version

STEP 2: Install MongoDB with the installation Wizard

- A. Make sure you logged in as a user with admin privileges. Then navigate to your downloads folder and double click on the .msi package you just downloaded. This will launch the installation wizard.

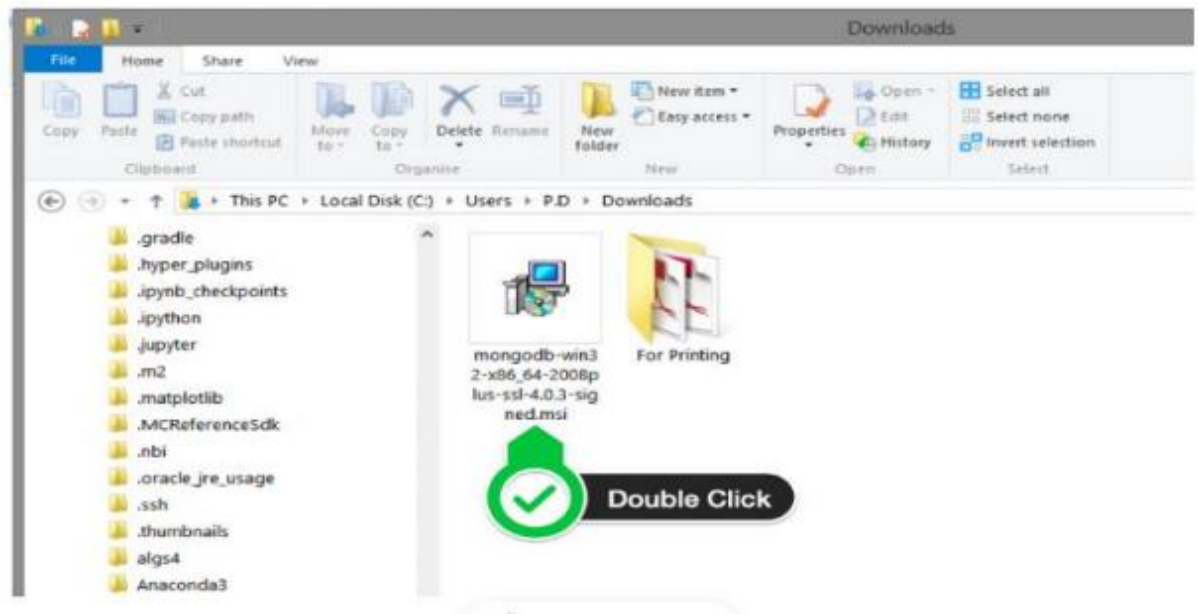


Figure 45 Launching the installation wizard

- B. Click on the next button



Figure 46 Clicking on next

C. Accept the license agreement then click Next

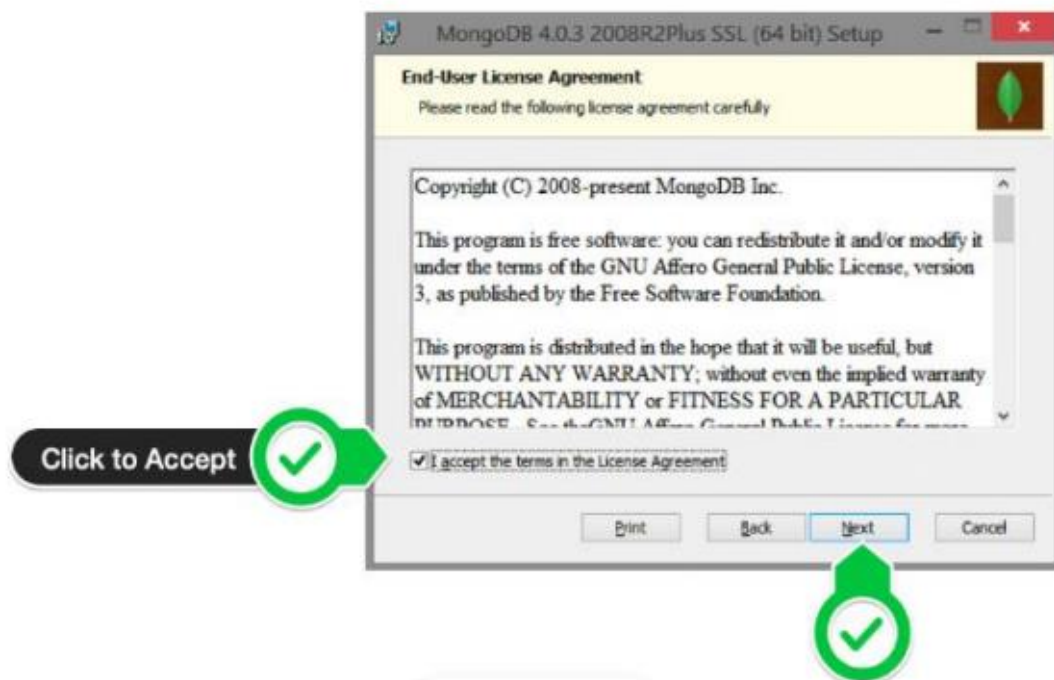


Figure 47: Agreement License

D. Select the Complete setup

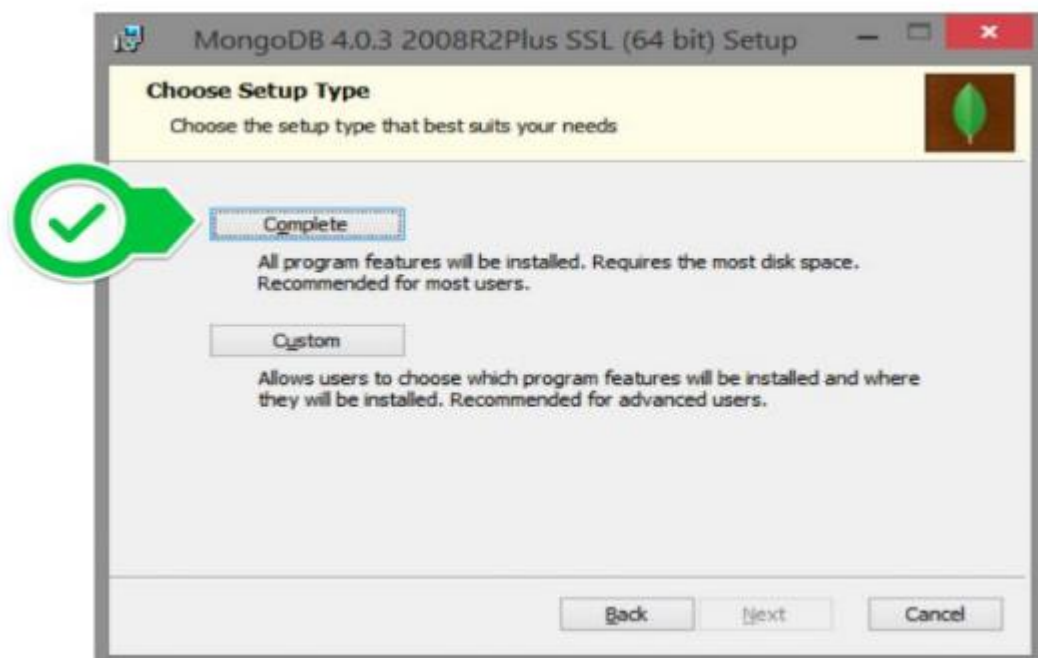


Figure 48 Select complete setup

- E. Select “Run service as Network Service user” and make a note of the data directory, we’ll need this later.

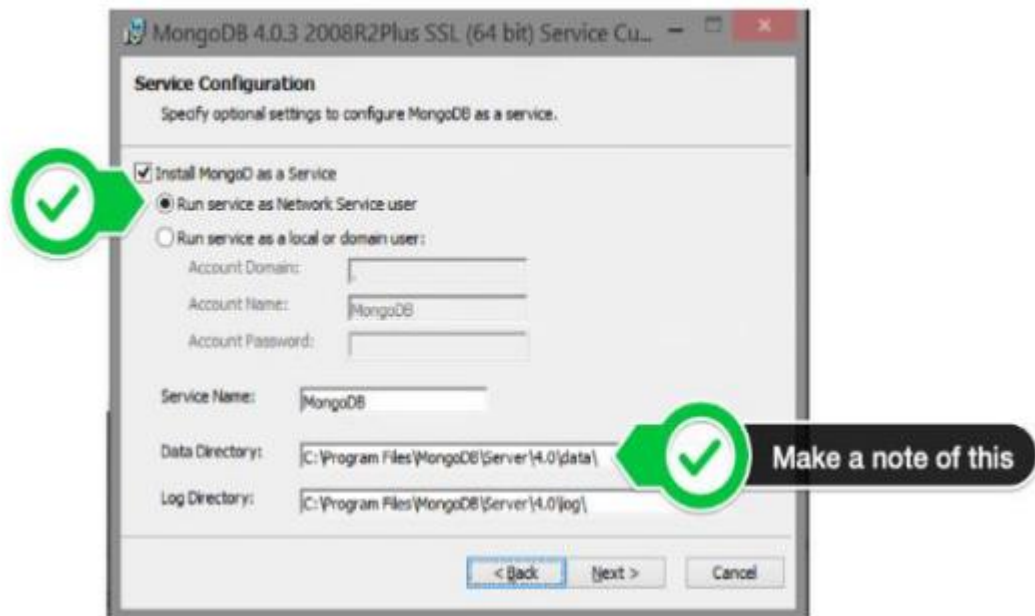


Figure 49: Run service network user

- F. Select mongo DB compass since you don’t have that already

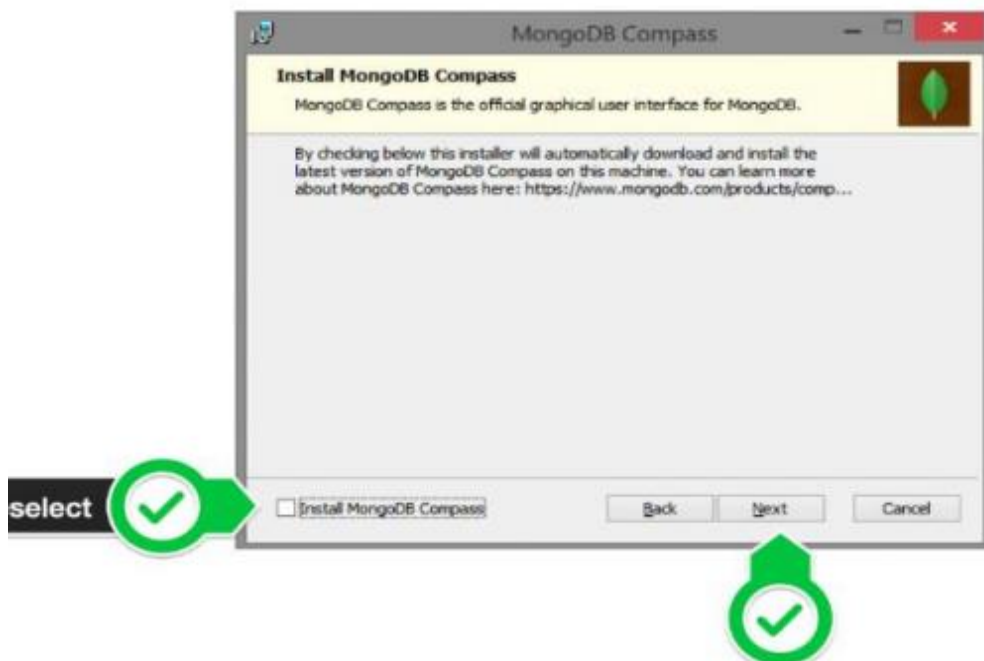


Figure 50: Selecting MongoDB compass and clicking next

G. Click Install to begin installation

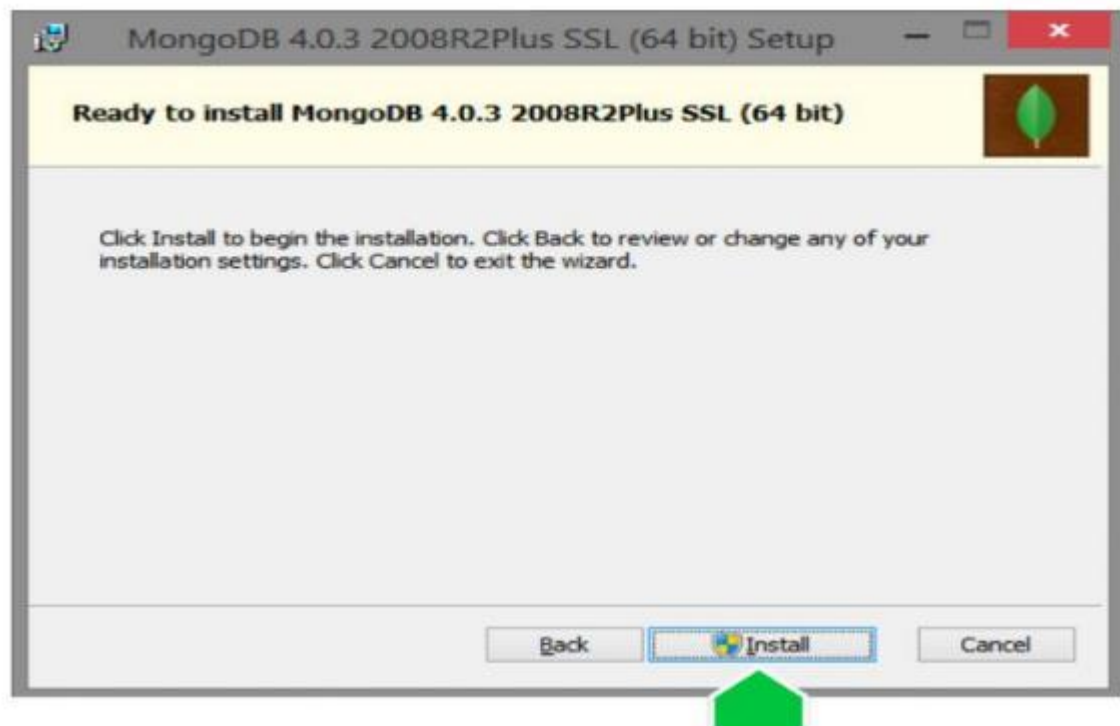


Figure 51: Click on install to start installation

H. Click on finish to complete installation

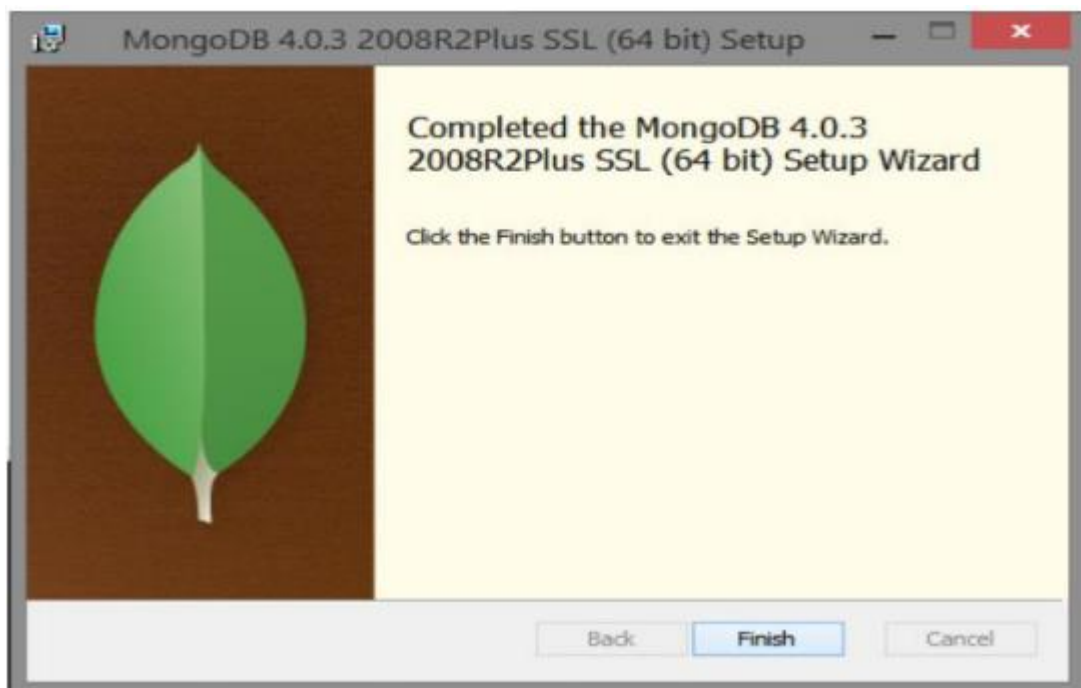


Figure 52: Click on finish to complete installation

B. WEB APPLICATION

To be able to use iKare web application, just connect to the internet and navigate to the website to either create an account or login.

II. SHOWCASES

a. Here we have the login screen

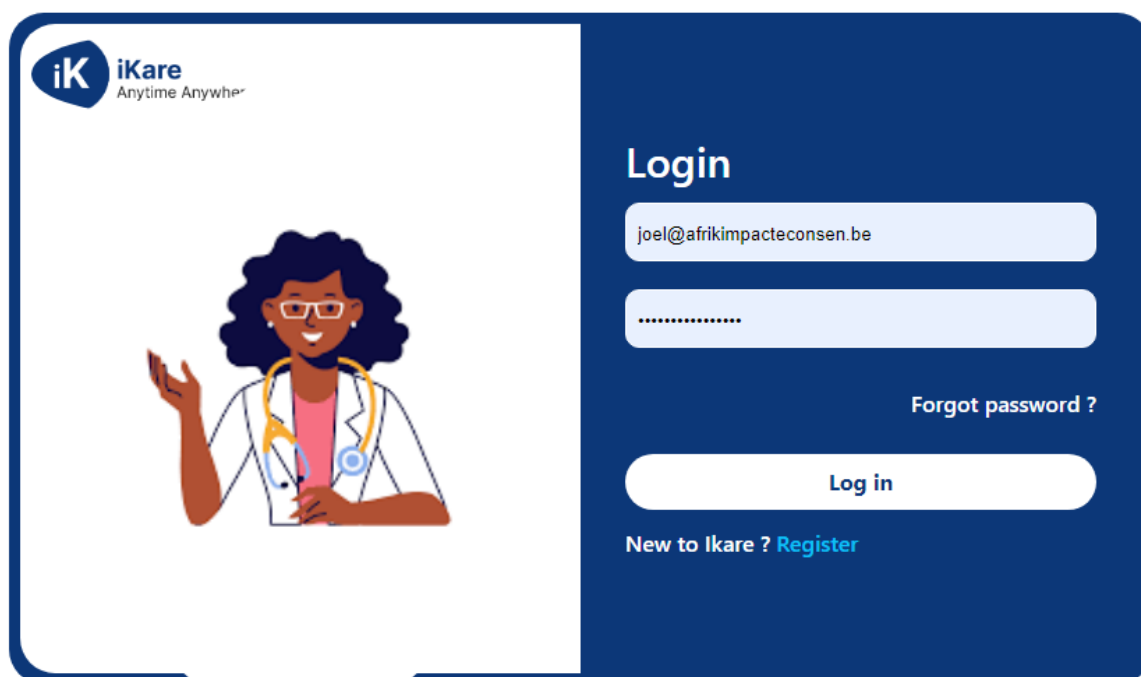
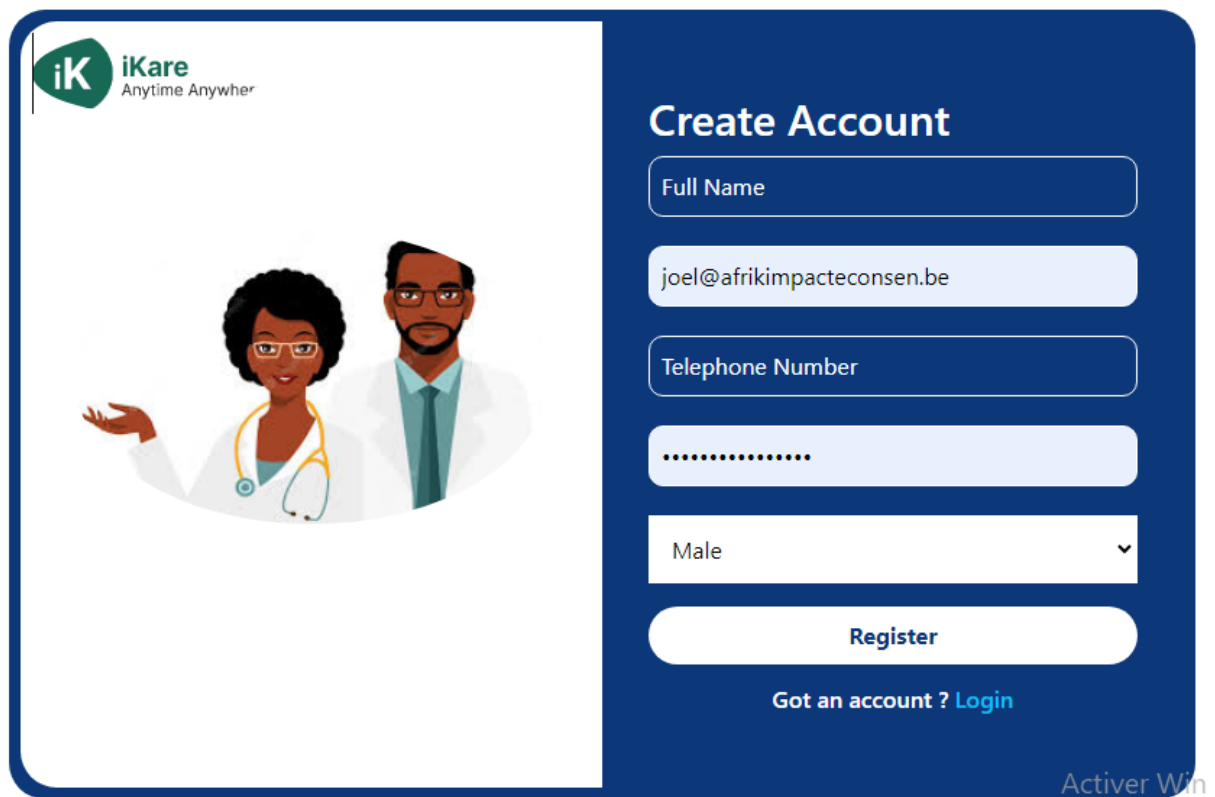


Figure 53: iKare login Screen

b) This is the register screen



The image shows the 'Create Account' screen of the iKare platform. On the left, there is a logo for 'iKare Anytime Anywhere' and an illustration of a female doctor and a male doctor. On the right, there is a registration form with the following fields: 'Full Name', 'Email' (containing 'joel@afrikimpacteconsen.be'), 'Telephone Number', and a gender dropdown menu (currently set to 'Male'). Below the form is a 'Register' button and a link 'Got an account ? Login'. The text 'Activer Win' is visible in the bottom right corner of the screen.

Figure 54: iKare register screen

d) Here we have the patient's welcome screen

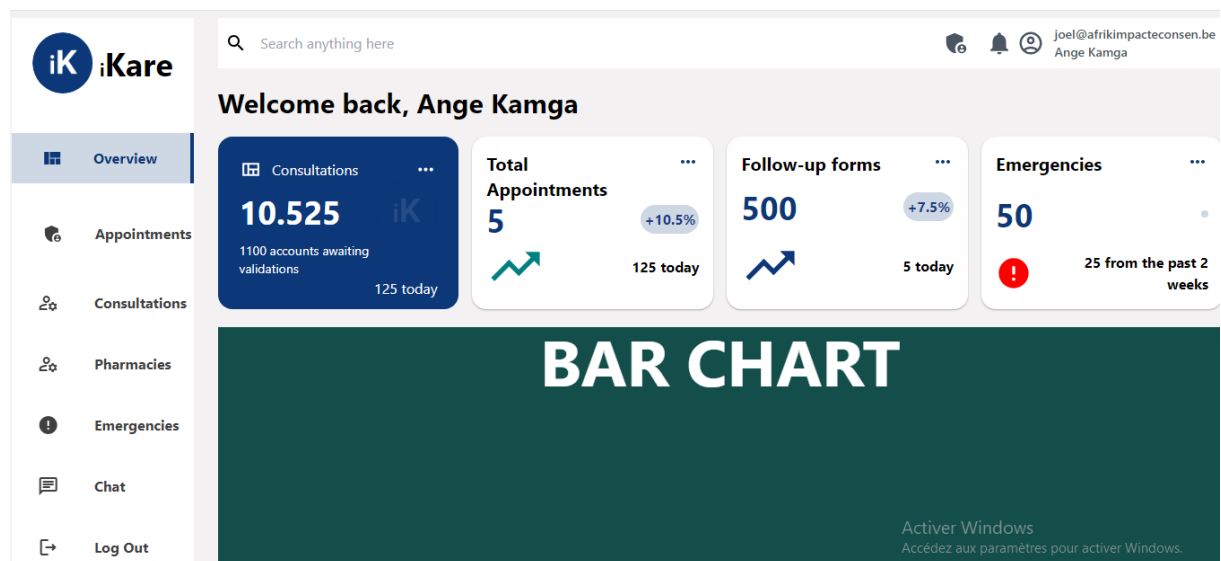


Figure 55: iKare patient's welcome screen

e) Book appointment screen

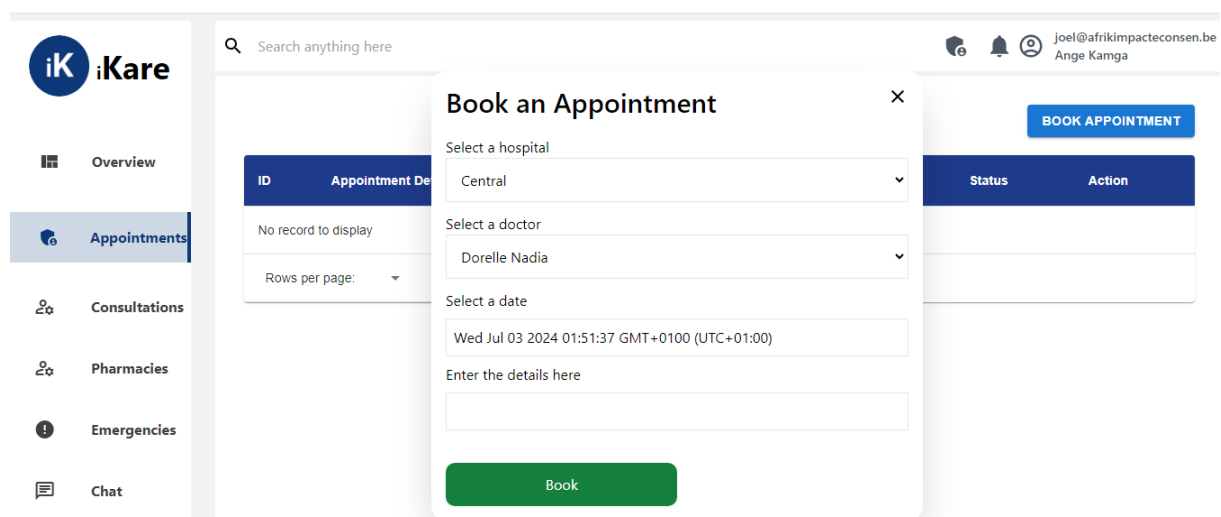
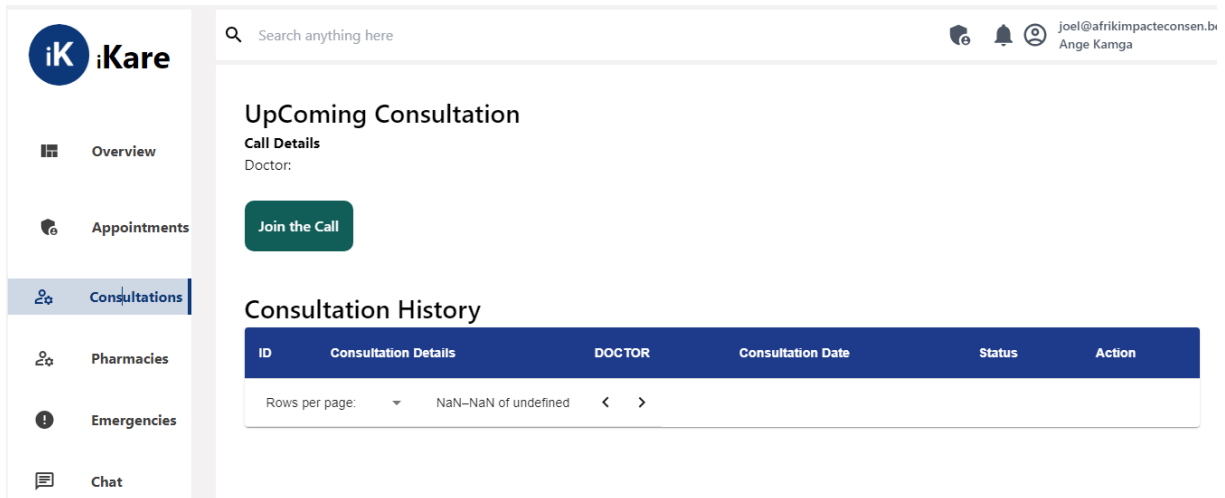


Figure 56: Book appointment screen

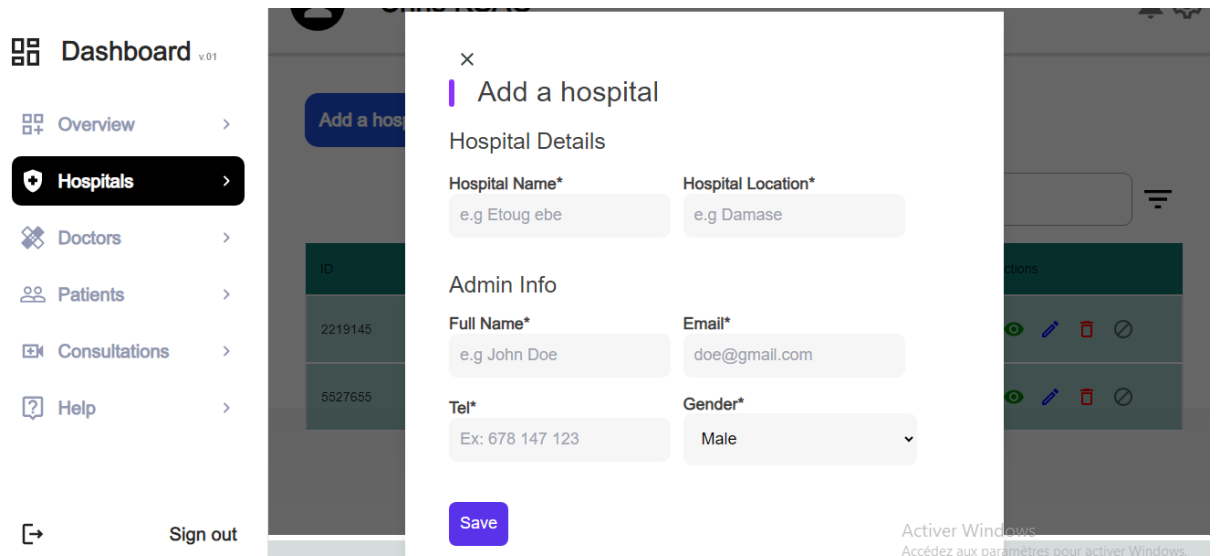
f) Patient's consultation screen



The screenshot shows the iKare patient consultation interface. On the left is a sidebar with navigation options: Overview, Appointments, Consultations (selected), Pharmacies, Emergencies, and Chat. The main content area has a search bar at the top right with the text "Search anything here" and user information "joel@afrikimpacteconsen.bi Ange Kamga". Below the search bar, there's a section for "UpComing Consultation" with a "Join the Call" button. Further down is a "Consultation History" section featuring a table with columns: ID, Consultation Details, DOCTOR, Consultation Date, Status, and Action. Below the table is a pagination control showing "Rows per page:" and "NaN-NaN of undefined" with navigation arrows.

Figure 57: Patients' consultation screen

f. Creation of hospitals and admin page



Add a hospital

Hospital Details

Hospital Name*

Hospital Location*

Admin Info

Full Name*

Email*

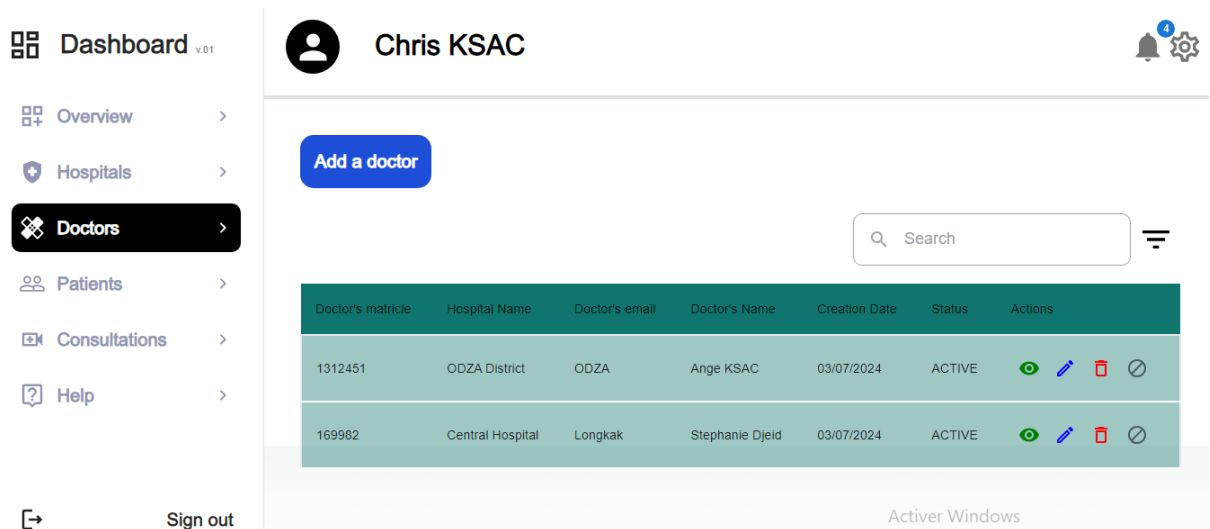
Tel*

Gender*

Save

Figure 58: Admin and hospital Creation Page

g. Add a doctor page



Chris KSAC

Add a doctor

Doctor's matricule	Hospital Name	Doctor's email	Doctor's Name	Creation Date	Status	Actions
1312451	ODZA District	ODZA	Ange KSAC	03/07/2024	ACTIVE	View Edit Delete
169982	Central Hospital	Longkak	Stephanie Djeid	03/07/2024	ACTIVE	View Edit Delete

Figure 59: List of Health Professionals page



CONCLUSION

Having put in place the platform, it was not sufficient for we had to produce a manual that will help its various users. That is why we presented the different tools to be installed and how they are to be installed in order to run this application without any problem and how the users will use this platform once the environment is set up.



PERSPECTIVES

Considering the fact that this project can be improved to a higher extend, as perspectives we can bring some ameliorations such as:

- + Implementing an IOT smart tool for collecting vitals.
- + Using Virtual and Augmented Reality for Health Conferences.
- + Developing an AI model.

GENERAL CONCLUSION

Having come to an end of our project, we learned how to manage stress and time with multiple school projects. For our project we worked on the theme “DEVELOPMENT OF A PATIENTS' CONSULTATION AND FOLLOW-UP PLATFORM FOR CAMEROON GOVERNMENT HOSPITALS”, our greatest desire is to facilitate consultation, follow-up of patients and access to prescribe drugs by doctors. We began by identifying the current system in place for the consultation process, project constraints and requirements in the specification book, then we went forth to analyze our system (iKare) using UML-2TUP methodology. We made use of Sybase Power AMC, a modelling software used to draw our various diagrams. In order to realize our project, we used react-native for mobile front-end, React js for the front-end, NodeJS for the backend and MongoDB as our database. Our system, iKare will enable patients to book appointments for consultations, be consulted by a medical professional, receive notifications or reminders for medications or consultations. We won't end here, updates and important improvements are going to be made in the nearest future in terms of functionality, security to make it more reliable.



BIBLIOGRAPHY

- ✚ Mercurial 2022
- ✚ International Journal of Database Management Systems (IJDMS) Vol.10, No.2, April 2018.
- ✚ Tracking 100 Core Health Indicators in Cameroon in 2019 & SDG Focus-En.



WEBOGRAPHY

- + https://www.tutorialspoint.com/object_oriented_analysis_design/index.htm Object Oriented Analysis & Design Tutorial. July 15th, 2023, 09:00 PM.
- + <https://icograms.com/designer> Designing a geolocalisation diagram. July 31st, 2023 09:00 PM.
- + <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/> What is a Class Diagram? August 21st, 2023, 09:00 PM.
- + <https://formik.org/docs/guides/validation> FORMIK documentation. August 24th 2022 10:30 AM.
- + <https://console.twilio.com/?frameUrl=/console> SMS API information. August 31st, 2023, 5:30PM
- + <https://primereact.org> DataTable. September 18th, 2023, 3:30 PM.
- + <https://www.mongodb.com/docs/mongoose/current/reference/associations/> Associations. September 21st, 2023, 4:30PM
- + <https://redux.js.org/tutorials/essentials/part-5-async-logic> Async Logic and Data Fetching. September 22nd, 2023, 12:30PM.
- + <https://github.com/videosdk-live/webrtc/blob/main/react-native-webrtc-app/client> React Native Video Calling App with react-native-webrtc. September 24th, 2023, 6:30 PM.
- + <https://www.scaler.com/topics/react-native-webrtc/> React Native Video Calling App with react-native-webrtc. September 24th, 2023, 7:30 PM.
- + <https://www.digitalocean.com/community/tutorials/test-a-node-restful-api-with-mocha-and-chai> Test a Node RESTful API with Mocha and Chai. September 25th, 2023, 9:00 PM.



TABLE OF CONTENT

DEDICATION.....	i
ACKNOWLEDGEMENT	ii
Content	iii
LIST OF FIGURES.....	v
LIST OF TABLES.....	vii
GLOSSARY	viii
Abstract	ix
Résumé	x
General Introduction	1
PART I.....	2
EXISTING SYSTEM.....	2
Preamble.....	3
Content	3
INTRODUCTION.....	4
I. PRESENTATION OF THE PROJECT THEME.....	5
II. DESCRIPTION OF THE EXISTING SYSTEM	6
III. LIMITATIONS OF THE EXISTING SYSTEM	9
IV. PROBLEMATIC	10
V. PROPOSED SOLUTION.....	10
CONCLUSION	11
PART II.....	12
SPECIFICATION BOOK	12
Preamble.....	13



Content	13
INTRODUCTION.....	14
I. CONTEXT AND JUSTIFICATION	15
A. Context.....	15
B. Justification.....	17
II. OBJECTIVES OF THE PROJECT	18
A. GENERAL OBJECTIVE	18
B. SPECIFIC OBJECTIVES.....	18
III. EXPRESSION OF NEEDS	19
1. Functional Needs	19
2. Non-Functional Needs	20
Performance, Scalability and Security	20
IV. PROJECT PLANNING	21
a. Chronogram of activities.....	21
b. Gantt Project planning	22
V. ESTIMATED COST OF THE PROJECT	23
a. Software Resources.....	23
b. Hardware Resources	24
c. Human resources.....	25
d. Global Estimation	25
VI. CONSTRAINTS.....	26
VII. LIST OF PARTICIPANTS AND DELIVERABLES	27
a. LIST OF PARTICIPANTS.....	27
b. DELIVERABLES.....	27
CONCLUSION	28
PART III	29
ANALYSIS BOOK	29



Preamble.....	30
Content	30
INTRODUCTION.....	31
I. METHODOLOGY	32
1. COMPARATIVE STUDY OF UML AND MERISE	32
a. MERISE	32
b. UML	32
2. COMPARATIVE STUDY OF UNIFIED PROCESSES	35
a. <i>A Unified Process</i>	35
b. <i>The Two Track Unified Process (2 TUP)</i>	35
II. CHOICE OF THE ANALYSIS METHOD	38
III. MODELLING OF THE PROPOSED SOLUTION	39
A. Capture of Functional Needs.....	39
1. <i>USE CASE DIAGRAM</i>	39
2. <i>COMMUNICATION DIAGRAM</i>	49
3. <i>SEQUENCE DIAGRAM</i>	53
4. <i>ACTIVITY DIAGRAM</i>	58
CONCLUSION	64
PART IV	65
CONCEPTION PHASE	65
TEST OF FUNCTIONALITIES	66
Preamble.....	66
Content	66
INTRODUCTION.....	67
A. GENERIC DESIGN	68
a. <i>Hardware diagram of the system</i>	68
b <i>High Level Architecture of the System</i>	68



B.	CAPTURE OF TECHNICAL NEEDS.....	69
a.	<i>Physical Architecture</i>	69
b.	<i>Logical Architecture</i>	70
C.	RELATED UML DIAGRAMS	71
1.	<i>CLASS DIAGRAM</i>	71
2.	<i>STATE MACHINE DIAGRAM</i>	76
3.	<i>PACKAGE DIAGRAM</i>	81
	CONCLUSION	84
	PART V.....	85
	REALIZATION PHASE	85
	Preamble.....	86
	Content	86
	INTRODUCTION.....	87
	1. <i>DEPLOYMENT DIAGRAM</i>	88
	2. <i>COMPONENT DIAGRAM</i>	91
	CONCLUSION	95
	PART VI	96
	TEST OF FUNCTIONALITIES	96
	Preamble.....	97
	Content	97
	INTRODUCTION.....	98
	1. APPLICATION FUNCTIONALITIES	99
	2. TESTS SHOWCASES	100
	CONCLUSION	102
	PART VII.....	103
	INSTALLATION GUIDE AND USER GUIDE	103
	Preamble.....	104



Content	104
INTRODUCTION.....	105
I. INSTALLATION OF THE APPLICATION	106
<i>A. DATABASE SERVER INSTALLATION</i>	106
<i>B. WEB APPLICATION</i>	111
II. SHOWCASES	111
CONCLUSION	116
PERSPECTIVES.....	117
GENERAL CONCLUSION.....	cxviii
BIBLIOGRAPHY	cxviii
WEBOGRAPHY	xiv
TABLE OF CONTENT.....	xv