Eastern Visayas State University Ormoc City Campus
Bachelor of Science in Information Technology
Engineering Department

ST. PATRICK POLYCLINIC SCHEDULING AND QUEUING MANAGEMENT SYSTEM

A proposal
presented to the faculty of the
Bachelor of Science Information Technology
Engineering Department

In partial fulfillment of the requirements for the
Degree of Bachelor of Science in Information Technology

By:

Gilbert T. Cabangal
Rasalie C. Pedroso
Zherra Stifannie D. Serat

Adviser:
Mr. Wilfred Jude Perante

2021

ST. PATRICK POLYCLINIC SCHEDULING AND QUEUING MANAGEMENT SYSTEM

A proposal

Presented to the Faculty of the

Bachelor of Science in Information Technology,

Eastern Visayas State University

Ormoc City Campus

In Partial Fulfillment of the Requirements

for the Degree of Bachelor of Science in Information Technology

By

Gilbert T. Cabangal

Rasalie C. Pedroso

Zherra Stifannie D. Serat

Adviser

Mr. Wilferd Jude Perante

2021

EXECUTIVE SUMMARY OR ABSTRACT

Making an online appointment is the new trend in our time, particularly in the health care field. Because of the world's current health problem, arranging an appointment online before visiting to the clinic is necessary. It does not only provide an easy way for patients to schedule an appointment (they don't have to wait in line for hours), but it also helps to reduce patient flow. Future patients in St. Patrick Polyclinic can access the St. Patrick Polyclinic Scheduling system through a web browser, which will allow them to make a schedule online by simply providing their personal information. The system features a queue management system which will not only benefit the patients but the entire management as well. Each patient appointment has a unique QR code that can be utilized in queues. Queuing management aids in patient flow management. The St. Patrick Polyclinic scheduling and queuing management system is meant to provide a comfortable and convenient approach to schedule an appointment.

Keywords: Scheduler, Online Appointment, Queuing, Healthcare, Clinic.

ACKNOWLEDGEMENT

This thesis became a reality thanks to the generous support and assistance of many people, and we would like to express our heartfelt gratitude to each and every one of them.

First and foremost, we want to thank our **GOD ALMIGHTY DEAREST JESUS** for the wisdom, strength, peace of mind, and good health that he has bestowed upon us in order to complete this capstone project.

We would want to express our appreciation to our **family** for their encouragement, support, and financial assistance in completing this paper. To our **parents**, whose love, guidance, and eternal inspiration accompany us in everything we do, to our particular inspirations, most notably to my son **JON REIN**, who encourages me to do my best, to work hard in every stage of completing this project.

We would like to express our gratitude and thanks to our adviser, **MR. WILFERD JUDE PERANTE** for imparting his knowledge and expertise in this study.

We are grateful to the EVSU Ormoc Information Technology Faculty for their advice and constant supervision, as well as for supplying necessary project information and help in completing this endeavor.

Distinguished members of the panel headed by **ENGR. JOSEPH JAYMEL S. MORPOS together with ENGR. EDWARD B. BERTULFO** and **MR. ALEXANDER B. HIPE** for the approval of our work.

Our thanks and appreciation also go to the people who have willingly helped us out with their abilities.

TABLE OF CONTENTS

Chapter 3: Technical Background

Chapter 4: Methodology

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In this day and age, where technology contributes a lot of progress in various sectors of industry and everything we do needs to be done efficiently and effectively, ICT has been providing a lot of project systems that have been approved to make our daily lives easier, particularly in the healthcare industry.

Healthcare is one of the world's fastest-growing industries. Medical appointments were normally made over the phone and by physically visiting clinics since the procedure needs a participation of people. Scheduling of appointments was limited to the availability of schedulers, or a person's physical presence.

Long waiting lines have created an unpleasant scenario for healthcare facilities. Manual appointments require the physical presence of both parties (patients and doctors). Patients, in particular, demand an easy way to schedule an appointment, and they integrated healthcare system capable of providing seamless service.

Healthcare facilities are organizations that are in charge of delivering health care and treatment to everyone. But how can a clinic deliver a fast and more efficient service if their day-to-day procedures are still done manually? The majority of clinics still use a manual procedure to schedule appointments for their clients/patients.

Project Context

St. Patrick Polyclinic in Ormoc City is one of the few healthcare facilities that still uses conventional methods in their daily activities, such as scheduling appointments and patient flow management. Patients who wish to have a medical

examination must arrive early and speak with the medical personnel in order to be included on the patients' appointment list.

When patients arrive at the clinic, the medical staff writes their names in the patients' registration book. This record will be used to queue the patients' appointments flow, which is based on a first-come, first-served basis. In that case, the patients would have to wait longer before being tested properly. The method of making appointments takes a long time.

Another issue at St. Patrick Clinic is that their specialists/physicians are not full-time clinic personnel; they have other patients in other hospitals to visit, which is why patients often must wait three to five hours for the doctors to arrive. If the doctor wants to cancel the appointment the medical staff will manually send a message through SMS prior to the scheduled date of appointment.

In relation to this, we will build a system called St. Patrick Polyclinic Scheduling and Queuing Management System. Since the emergence of online appointment systems, which provide timely and efficient access to health care services, this will alleviate the current issue of booking patients' appointments at St. Patrick Polyclinic.

In this scheme, doctors are given the authority to set a weekly appointment schedule given the date and time of his/her availability.

Patients can schedule appointments using an appointment scheduler by accessing the website through a web browser. Patients must register first using some of their personal information and once it is done the system will send a verification code to the patients email and verify their account. Once the patient

accessed the system, it will provide choices relevant to the services that St. Patrick Polyclinic provide.

The patient must choose any specific services that they need, and after that the device will generate the appointment QR code. The appointment queues would use the scanned QR codes to manage the patients flow.

While the patients wait, information screens and digital signs indicate the current position of the queue. It will also notify patients of their turn by showing their name on the LCD Monitor and directing them to the appropriate doctor's office.

Purpose and Description of the Project

The St. Patrick Polyclinic Scheduling and Queuing Management System is a proposal system for outpatients. It was created specifically for St. Patrick Polyclinic in Ormoc City. It will increase outpatient in a coherent manner, allowing management to efficiently manage resources, and patient queues so that patients receive the appropriate treatment at the appropriate time in a relaxed and hassle-free setting.

Using an online appointment scheduler is the most effective way to assist outpatients in making consultations with their physicians for medical check-ups without difficulty. It will benefit not only outpatients, but the entire clinic administration, as it will appease their clients/patients, resulting in increased revenue.

In addition, the queuing control system will handle outpatient traffic, resulting in shorter wait times for outpatient appointments.

Objectives of the Study

General Objectives

St. Patrick Polyclinic Scheduling and Queuing Management System main objective is to provide an efficient and economical way to make an appointment and manage queues. It is wanted that patients' needs an easy and fast way of making an appointment.

Specific Objectives

This project will provide organized and well managed scheduled appointments and to smoothly managed appointment traffic. It will also give transparency and reduced patients waiting times and avoid no show and other conflicts. It will also provide a real- time dashboards and reports in monitoring appointments schedules of patients.

Scope and Limitations of the Project

This project is basically designed for St. Patrick Polyclinic. The clinic provides services to patients like daily treatment of patients. The study was focused on two types; Online appointment scheduling and queue management system development. To define the scope and constraints of this investigation, the researcher developed the scope and limitations of this project.

Scope

- Appointment Scheduler.
- Management of Scheduled Appointments.
- Generate Quick Response Code used per queue.
- Queuing Management.

Limitations

The St. Patrick Polyclinic scheduling and queuing management system limits itself on the following:

- Generating of billing statements.
- Medical records of patients.
- Generating medical prescription.
- Other information that are not related to scheduling of appointment.

The system is unable to generate billing statements for patients because it lacks a module that can perform this function. There is no module in the system that can store information about a patient's medical history or reports. The system also doesn't produce online medical prescriptions or anything else that isn't connected to online appointment scheduling.

Significance of the Study

St. Patrick Polyclinic scheduling and queuing management system is to be utilizing by the administration, doctors, and patients. The user will find it useful because of its numerous benefits.

The result of this study is beneficial to the following: **Clinic, Doctors, Clinical Staff, Patients, Researchers and Future Researchers**. By offering improved services, the technology will help to save time spent with customers by simplifying and automating daily clinic procedures. It will aid in the establishment and notification of the doctors' weekly appointment schedules.

This study will benefit clinical staff greatly because they are an essential part of the clinic's scheduling and queuing management system. Patients are the doctor's most valuable asset, which is why this study is being undertaken to provide a solution for doctors to better their services to their patients. In making this project researcher will develop their analysis in writing for the documentation and gain more knowledge about the capstone project proposal.

Other researchers will profit from this study if they want to conduct a similar investigation since the results of this study will serve as a template for them to adapt in their research.

CHAPTER 2

REVIEW OF RELATED LITERATURE AND SYSTEMS

This chapter presents the theoretical background, related literature, and systems used by the proponents in order to have a clear view and knowledge in how to develop the proposed system. Those that were included in this chapter helped in familiarizing information that is relevant and similar to the present study.

Review of the related literature, helped the researchers to accustom himself with current knowledge in the field or area in which he is going to conduct his research and to review all related literature that enables the researchers to identify the limits of his/her field.

It helped the researchers to define problem, avoid unprofitable and ineffective problem area, avoid accidental duplication of well-established findings, and gain knowledge to choose the problem given in the previous research, as suggestions for further studies.

Theoretical Background

Patients should be able to receive services from health systems that are safe, efficient, and smooth. Several important reimbursement changes, increasing critics and economic pressures on the system, and increased demand for quality and efficacy from highly aware and educated patients as a result of technological and telecommunications improvements have begun to place additional pressure on healthcare manager to respond to these concerns.

Queuing theory is an example of its application in healthcare. It primarily deals with patient flow through the system; if it is good, patient queuing is minimized; if it is bad, the system may lose money and patients may experience significant queuing delays. Health care system can be visualized as a complex queuing network in which delays can be reduced through the following ways:

- Scheduling of resources (doctors and nurses) to match patterns of arrival
- Constant system monitoring (tracking number of patients waiting by location) linked to immediate action.

Scheduling and Queuing Management System for St. Patrick Polyclinic Ormoc City "Our Journey towards the implementation of a Scheduling and Queuing Management system to the Critical Care Environment" according to the above statement, there has been a significant advance in medical technology used in patient treatment and care.

A comprehensive "Scheduling and Queuing Management System" is intended to respond quickly to a patient's request for an appointment. The biggest problem at St. Patrick Polyclinic, according to the clinic's director, is that they use a manual system for recording, tracking, and monitoring patients' requests, which might lead to delayed transactions among clients.

These studies were conducted to create a scheduling and queuing management system that will be used as a proposal to assist doctors and staff in saving time and resources by automating everyday clinic operations.

Related Literature

Obulor R., Eke B. (2016) This research presents an efficient queuing model for proper appointment system to give the solution to the long waiting times in the hospitals. This system provides better utilization of resources and reduces patients waiting times in the general OPD before consultation with the Doctor. Nidhi Malik, "Application of Queuing Theory to Patient Satisfaction at Combined Hospital, Srinagar Garhwal Uttarakhand". This paper given a complete idea about the patients' demographic characteristics, nature of their illness, time they spent before reaching the hospital, on the queue to see the doctor and with the doctor. It also describes the patient's view about queue and their behavior in the queue.

Zao, Yoo, Lavoie, Lavoie, and Simoes (2017) This study aims to find evidence to expound the benefits and challenges while implementing appointment scheduling and queuing management system. This appointment scheduling is more concern with the patient's safety, this system shows positive changes not just for patients but also to the doctors and staff. It lessens the work of staff as well as the waiting time of every patient, and reduced contact with other people. Appointment scheduling and queuing management system has positive changes in such a way that it is patient-centered, however some instances in the case of healthcare providers and patients there is slow adoption of appointment scheduling. Patients doubt to adopt this queuing management system was influenced by their preceding experiences using computers and internet connection.

Ikwunne (2016) During this catastrophic event worldwide, people get to seek relevant ways to suffice peoples concern about scheduling their visit to the hospitals or clinic. As observed, patient(s) tend to wait outside of the hospitals or clinic for their daily, weekly, or monthly check-ups. This scenario bears huge burden to those who are sick or just having check-ups. According to Ikwunne an Onyesolu, 2016, "*The major problem faced by the patients in that department (hospital) is overcrowding as a result of delay in consultation.*" This kind of situation is also present in our country, leading to conflicts and deprivation of rights and worst questioning one's profession that is inefficient.

Sherly (2016) Filipino people are fond about reservations, as we investigate it, whatever we do we always reserve something that is beneficial to us. In the past years companies uses logbooks to record the reservations made by their clients. Today, as technology keeps on progressing, presently used in every aspect such as recording and keeping files and make it easy to find. According to, Irin Sherly et. al. 2016, "In classical system patients has to go to the hospital and wait in a queue at the appointment desk to make the reservation and get an appointment but, they generally finish up waiting endlessly for a long-time interval. The patient might choose to fix an appointment, but this choice is always not possible and does not likely work well for all the people involved in the system. People involved are as follows: The medical personnel, the hospital, and the patient. The patient longs for effortlessly convenient and accessible times. When they cannot discover a quick appointment, they feel like waiting endlessly. The patient also expects to be seen suddenly or within few minutes of their visit to the hospital or clinic."

Onoja and Kembe (2018) Queuing theory is a renowned and tested mathematical approach to the analysis of waiting lines. The queuing characteristics at the Ante-natal care clinic of JUTH were analyzed using a Multi-server Queuing Model and the Waiting and service Costs. The results of the analysis demonstrated that average queue length, waiting time of pregnant women as well as over usage of doctors can reduce when the capacity service level of doctors at the clinic is increased.

Geng, Wen J (2019) The random arrivals of walk-in patients significantly affect the daily operations of healthcare facilities. This paper attempts to make an appointment schedule by considering walk-ins and the waiting time target (WTT) for appointment patients. A finite-horizon Markov Decision Process model is formulated to establish the optimal real-time scheduling policy. Numerical experiments based on real data are performed to investigate the influence of different parameters and to compare different schedules.

Senderovich and Weidlich (2016) Conformance checking and performance improvement in scheduled processes - This work targets the analysis of resource-driven, scheduled processes based on event logs. It suggests adaptations of the scheduling policy of the service process to decrease the tardiness (non-punctuality) and lower the flow time. The improvement technique yields a decrease in median tardiest and flow time by more than 20%. It is based on areal-world dataset comprising clinical pathways of an outpatient clinic that have been recorded by a real-time location system (RTLS).

Berger (2016) To maximize operating room (OR) utilization, better estimates of case duration lengths are needed. We used computerized simulation to determine whether scheduling OR cases using a statistically driven system that incorporates patient and surgery-specific factors improves OR throughout and utilization. They modeled surgical and anesthetic length of vascular surgical procedures as a function of patient and operative characteristics.

Related System

Tufail (2018) Aims to bring together all practitioners and patients. By using this application user will be able to know and access the professional profile of each registered doctor. Doctor's profile includes information regarding their professional experience, practicing license, educational background, working days, clinic accessibility and other patient reviews. New practitioners can easily join the platform and start taking appointments without the need of any expensive advertisement. The doctor can easily get access to the patient's profile and update medical record. Users can sign up online, search for nearby doctor and book appointment while sitting at their homes by using a web browser. Two different types of actors are using this system: user actor (patient) and administrative actor (physician).

Rodnic and Olga (2016-2017) By the use of online scheduling and queuing system, the hospital can accommodate clients/patients its most valuable services efficiently, free from patient' fulmination and other undesirable events. According to Olga, et. Al 2016-17 *"... we consider that all the requirements/objectives were fulfilled accordingly. We can say that the development team has learned several techniques during the development of this website making possible to further*

improve it and maintain it with necessary efficiency." Clearly, our hospitals must look necessary fully equipped ways to lessen the burden of their patients. Every hospital's must have their own website guiding their patients/clients concern about visiting the place. In that way, money and efforts of the people will not be wasted.

HSC.COM.MY (2016) The need to wait in line for a long time to see a doctor has been reduced thanks to technological advances such as the internet. HSC Medical Center is one such company that has adapted to the recent technological period. HSC Medical Center's appointment system does not require an ID or password to log in before making an appointment, and the appointment is valid for one week.

This system has flexible appointment templates, code-driven procedures, and powerful patient control are all available with this method.

CHAPTER 3

TECHNICAL BACKGROUND

Technicality of the Project

Today's technology advances at such a quick pace that most individuals mold their everyday lives around their machines. Internet, mobile, social networking, and other technologies have made it easier for us to connect with one another and have provided us with a great deal of convenience.

The system that will be used at St. Patrick Polyclinic is a web-based system, and only the clinic's administration, doctors, and patients will have access to the system's information.

Technically, the system will employ a variety of technologies to execute and meet all of the requirements necessary to achieve the goals. It will make use of the following current hardware and software:

Details of the technologies to be used:

The manual appointment scheduling technique is used at St. Patrick Polyclinic, which entails coming at the clinic as early as possible to be added to the patient's appointment lists on a piece of paper for a patient/s to be medically inspected.

Software:

XAMPP is a free and open-source cross-platform web server solution stack package that consists mostly of the Apache HTTP server, MySQL database, and interpreters for PHP and Perl scripts.

This is what we went with for our database. We need a server package to test the system using PHP, therefore we chose XAMPP because the researchers have used it before, and familiarity with the software is a key element in accomplishing things on time. We used it as a development tool, allowing us to test our work on our own PCs without requiring Internet access.

PHP (Hypertext Pre-Processor) is a server-side scripting language that is primarily used for web development but may also be used for other purposes.

We, the researchers, chose PHP because the purpose is to create a web site/portal, and PHP is one of the most popular server-side languages among programmers. One of the key reasons the researchers picked PHP is because of its power. The majority of the system's operations, such as adding and/or modifying major task lists and infractions, signing in as administrators, retrieving summary reports, and others, require PHP to run and work well.

The standard markup language for creating web pages is HTML (Hypertext Markup Language). HTML files can be read by web browsers and rendered into visible or audible web pages. HTML is a markup language, not a programming language, because it semantically represents the structure of a webpage as well as presentation cues.

A web browser's job is to read HTML texts and put them together into visible web pages. We choose HTML because it is the core language that got the system up and operating. HTML is a simple language to learn. You don't need to be a coder to use it, and it's the industry standard for building websites.

Microsoft Visual Studio is the company's integrated development environment (IDE). It's used to make websites, web apps, web services, and mobile apps, among other things. We use this in developing our websites because it is convenient to write codes with this IDE.

A web browser is a piece of software that allows you to search, display, and navigate information resources on the internet. A Uniform Resource Identifier (URI/URL) identifies an information resource, which can be a web page, image, video, or other piece of material. Users can simply navigate their browsers to related resources thanks to hyperlinks in resources.

Web Browsers such as Internet Explorer, Mozilla Firefox, and Google Chrome are used to locate and display content on the World Wide Web, such as web pages, photos, and other files, during the development and deployment phases. These can be used to access the web-based system.

Bootstrap is a collection of free and open-source tools for building websites and online apps. It really helps us in designing our website. Its goal is to make developing dynamic websites and web apps easier.

Hardware:

A personal computer is a general-purpose computer whose size, capabilities, and initial sale price make it suitable for individual use, and which is designed to be operated directly by the end-user without the need for a computer operator. We do our entire task in constructing this project using a personal computer; without computers, we would be unable to complete this project because it is an electronic system.

How the Project will work?

- Planning

  The system's scopes and weaknesses are determined by the proponents' plans for how will the system operate, which are based on information from the clinic and other related studies.

- Analysis

  The proponents examined all of the information gathered as well as the existing system's flow and efficacy.

- Design and Development

  With its acceptable and friendly user interface design, the proponents create a project that will undoubtedly solve the current problem of the facilities and their processes.

- Testing

  The proponents would test all of the system's features to see whether they are operating properly.

- Implementation

  When the system is finished and ready to use, the proponents can deploy it in the facilities.

- Maintenance

  The proponents will continue to improve and give maintenance to the system.

CHAPTER 4

METHODOLOGY

This chapter will discuss about the methodology that will be used in the development of St. Patrick Polyclinic scheduling and queuing management system. The fundamental for this project is to develop a scheduling and queuing management system that can be implemented and integrated in St. Patrick Polyclinic.

Using a methodology aids in project management by breaking down the development process into individual tasks and identifying the sequence in which they should be completed as well as the interdependencies of the tasks. This aids in system design, scheduling, and monitoring.

Environment

The study entitled Scheduling and Queuing Management System for St. Patrick Polyclinic will greatly help the patients, nurses and doctors to minimize the time consumed in the utilization of paper-based. The study takes place at St. Patrick Polyclinic located at Burgos St. Ormoc City, Leyte.

Locale

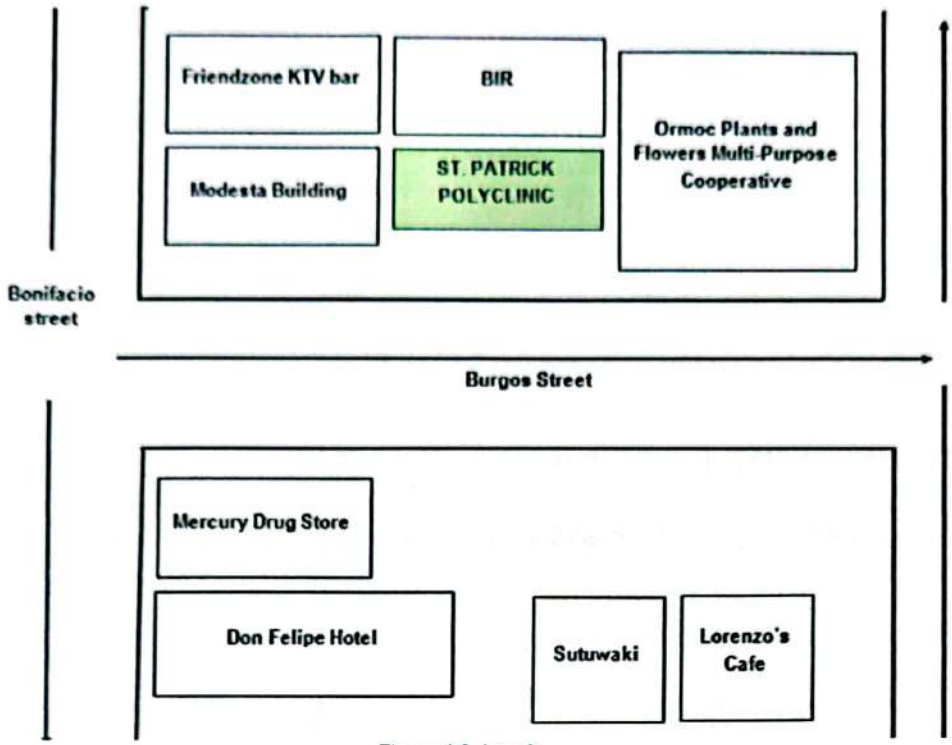


Figure 4.0: Locale

It shows the map where the study was conducted in Ormoc City, Leyte where the St. Patrick Polyclinic was located.

Population of the Study

The study's target population was designated as all the patients at St. Patrick Clinic. Patients from Orthopedic Surgeons, Internal Medicine, Pediatric Medicine, General Surgery, Obstetrics and Gynecology, and Internist/Pulmonologist are all included.

The responders were chosen at random from a total population of 49 patients on that particular day but the number of patients that was been examined was only 38.

	Specialist's List	No. of Medical Staff	No. of Scheduled Appointment	No. of Examined Patients
1	Orthopedic Surgeon	1	5	3
2	Internal Medicine	1	6	5
3	Pediatric Medicine	1	15	11
4	General Surgery	1	5	4
5	Obstetrics and Gynecology	1	10	8
6	Internist/Pulmonologist	1	8	7
	Total		49	38

Table 4.0: Distribution of Respondents

This table shows the distribution of respondents or patients at each of their physician's appointments; a total of 11 patients were not examined that day.

Organizational Chart/ Profile

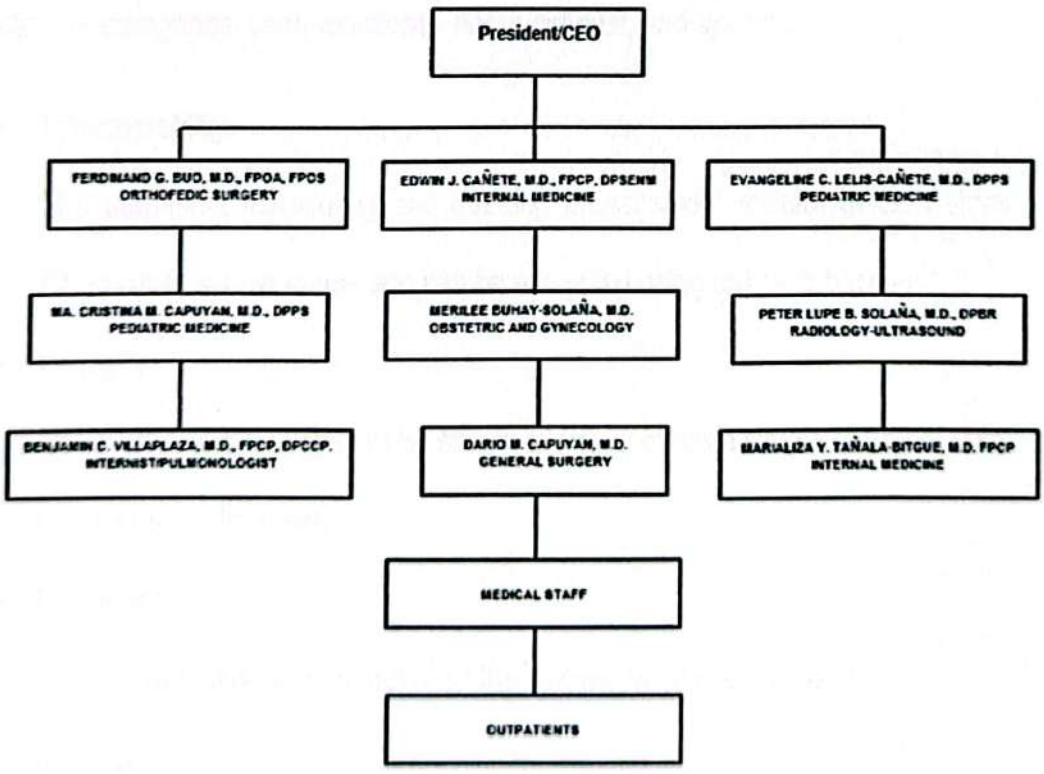


Figure 4.1: Organizational Profile

It shows the organizational chart of the St. Patrick Polyclinic in Ormoc City it conveys a company's internal structure.

Requirements Specifications

The specifications are how the project would assist St. Patrick Polyclinic in making online appointment process was addressed in this section. These are various forms of feasibility and modeling, which address the system's specifications and flow, as well as how it will operate until it is developed and implemented.

Operational Feasibility

After examining the data, we came up with a list of criteria, including user requirements, system hardware, and program attributes. The needs were divided into four categories: user, functional, non-functional, and systems.

- Interoperability

  The outpatient scheduling and queuing management system at St. Patrick Polyclinic is simple to use and can be accessed using any web browser.

- Usability

  Users will find the system to be simple to use. It offers a simple UI that is easy to understand for users.

- Maintenance

  The administrator is in control of all the maintenance in the system.

- Security

  This system has limited user data, but it is nonetheless adequately safeguarded because only the administrator, doctors, and patients have

access to it. Each user has their own username and passwords that are secured.

Fishbone Diagram

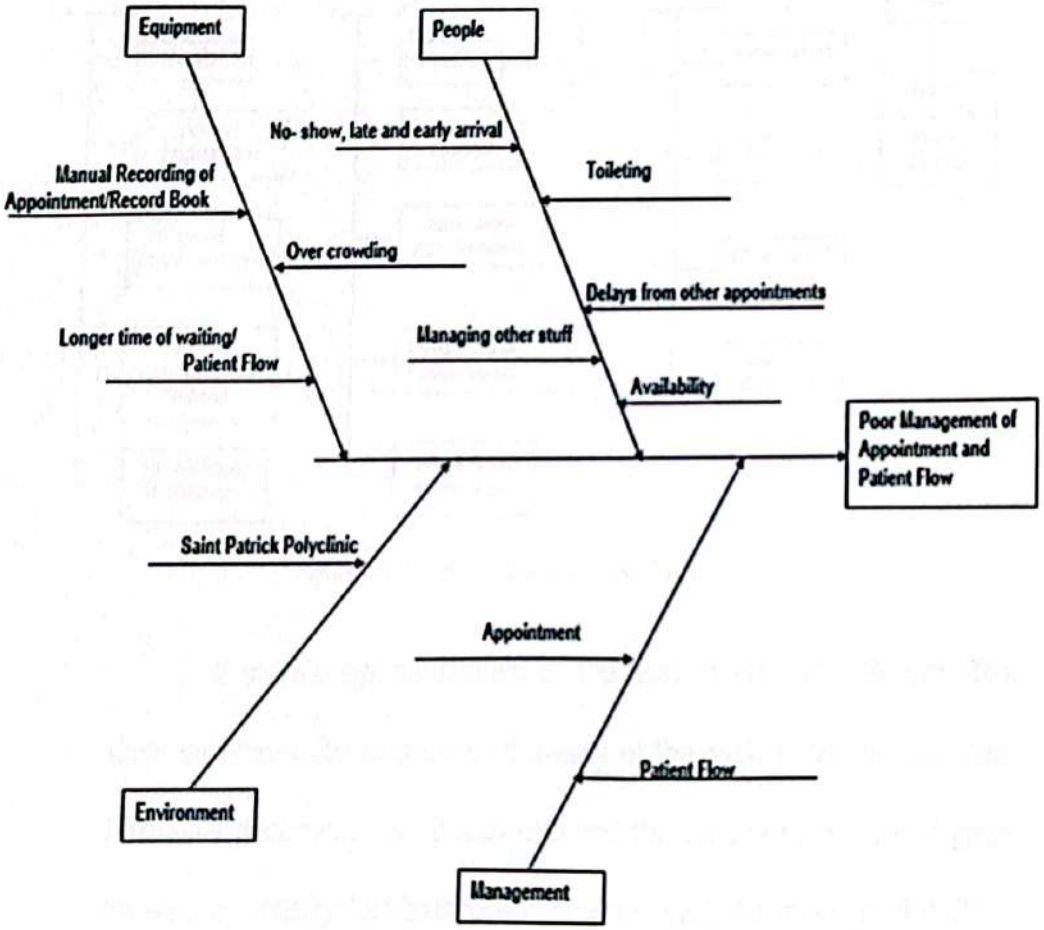


Figure 4.2: Fishbone Diagram

This image represents the St. Patrick Polyclinic fishbone diagram, a cause-and-effect diagram that aids managers in identifying reasons for flaws, variances, detects, or failures.

Functional Decomposition Diagram

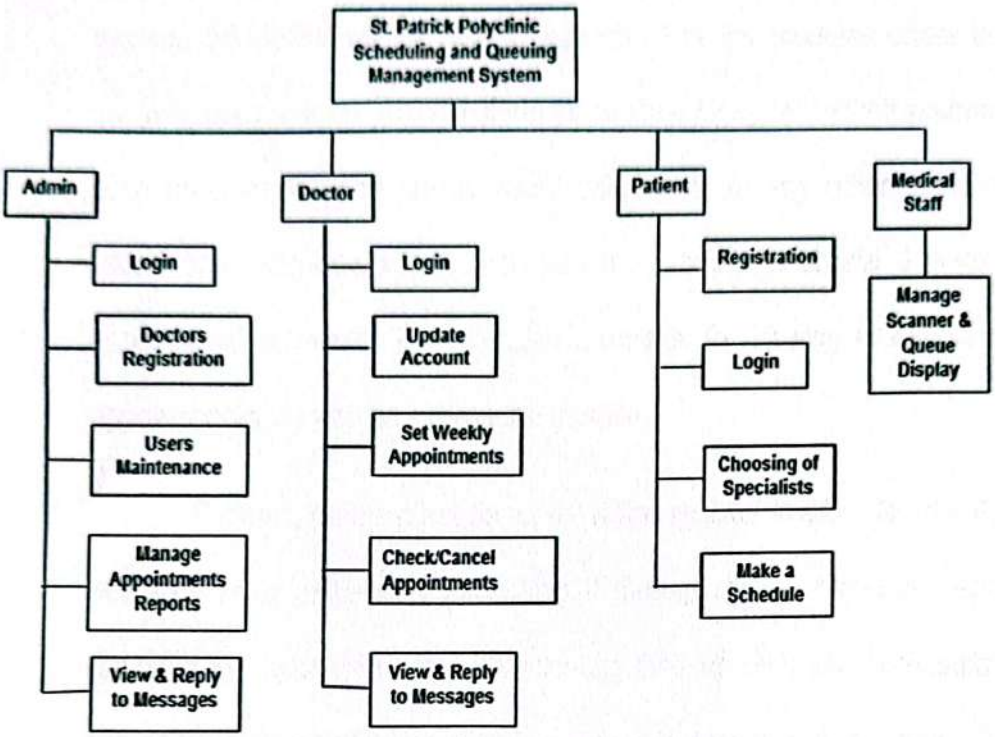


Figure 4.3: Functional Decomposition Diagram

It shows the capabilities of the system that are relevant. The diagram shows the sequence of events of the system and its top-down functional decomposition, it also exposes the system's structure. It gives the step-by-step system breakdown, beginning with the main function of the system.

The admin will first log in to the system, and then access all of the modules under the admin user modules, which include the Doctors' Registration module, user maintenance module, manage appointment reports module, and message module, after successfully logging in.

The doctor will also login using the username and password that the admin will provide to the doctor, and after successfully logging in to the system, the doctor will be able to access all of the modules under the doctor's user modules, which include the module for updating their account, such as changing passwords, email addresses, or any other personal information. The doctor can then use the module to create a weekly appointment schedule. There is also a module for viewing or canceling appointments, as well as a message module.

Patients, on the other hand, must first register in order to view the website's main page after accessing it through online browsers. After successfully registering and authenticating their account with their email, they must log in with their user and password to access their account.

They can access their dashboards in their account, where they can examine their planned appointments with its unique the QR codes, as well as the scheduler module, where they can select a doctor by clicking the specialist button. After patients schedule their medical appointments, the system provides an auto-generated rapid response code for each one. Messages and notifications can also be accessed by patients.

Technical Feasibility

Compatibility Checking

In the development of a project study, an assessment of the technical practicability of a proposed study is important so as to prove that the idealized

	Software	Hardware	Other Technologies
St. Patrick Polyclinic Scheduling and Queuing Management System	Windows 10 OS Xampp MySQL Database Visual Studio HTML, CSS, BOOTHSTRAP and PHP	Computer Set Laptops Smartphones Router	Internet Connection

Table 4.1: Compatibility

concept is technically possible. This means that the St. Patrick Polyclinic Scheduling and Queuing Management System for Outpatients can operate in the desired manner.

It also deals whether the materials in the development are technically and practically available. The materials include a computer set/ Laptops, Smartphones and Router/WIFI for internet connection for the development and the deployment of the system, a programming language is used for building the system, a database that serves as data repository, and most importantly, the patient and employee information that is needed for the system to be running.

For the development and deployment of the system, the researchers made use of the computer sets that are already available and windows 10 as operating system, Visual Studio as developing environment. The programming languages to be used are (HTML, CSS, BOOTHSTRAP and PHP).

These languages were utilized since the researchers are already familiar with these languages, thus, making it easier for them in the development process. As for the back-end database, Xampp MySQL Server will be used.

Relevance of the Technologies

Based on the assessment made, the realization of the system is achievable. The researchers are very certain that the availability of the needed materials and resources and the ability of the researchers in making use of these resources is an indication that the project, St. Patrick Polyclinic Scheduling and Queuing Management System for Outpatients, is technically feasible.

Schedule Feasibility

A Gantt chart is used to plan and keep an eye on the development of the project. In this way, the researchers are guided on the next step to follow in a certain amount of time. There are different factors to consider before the development of the project, such as the user's needs, what the system is expected to provide for its users, development cost, time constraints, and even the developer's capabilities and expertise.

Tasks	2020												2021				
	Jan.	Feb.	Mar.	Apr.	May	June	July	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	April	May
Pre-Proposal stage																	
Proposal Stage																	
Analysis																	
Design																	
Development																	
Testing																	
Defense																	
Revision																	

Table 4.2 Gantt Chart

The Gantt chart is used to illustrate the start and finish date of the different elements and/or components of the project. It also helps the researchers to be mindful of the specific time allotted for a certain activity and meet the target date.

Economic Feasibility

The cost and logical perspective for the project is referred to as economic feasibility. This table depicts the potential cost of expenses, which is less expensive but more dependable than typical document requisition methods.

Development	Costs
System Program	25,000
Consultants	2,000
Additional Fees (Programmer Maintenance)	3,000
Subtotal	30,000

Deployment	Costs
Manpower, Software and hardware (Router, Internet Connection, PC, Monitor Screen etc.)	250,000
Operation and Maintenance	30,000
Subtotal	280,000

Monthly Fees	Costs
Internet Bils	2000
Operation and Maintenance	10,000
Subtotal	12,000

Table 4.3: Budget and Cost Management

It shows the budget and cost management in developing and implementing the system in St. Patrick Polyclinic.

Requirements Modeling

A requirement modeling is a perceptive way of determining the functions of the system and its components. This refers a process of building up a specification of the system and produces useful and provable models.

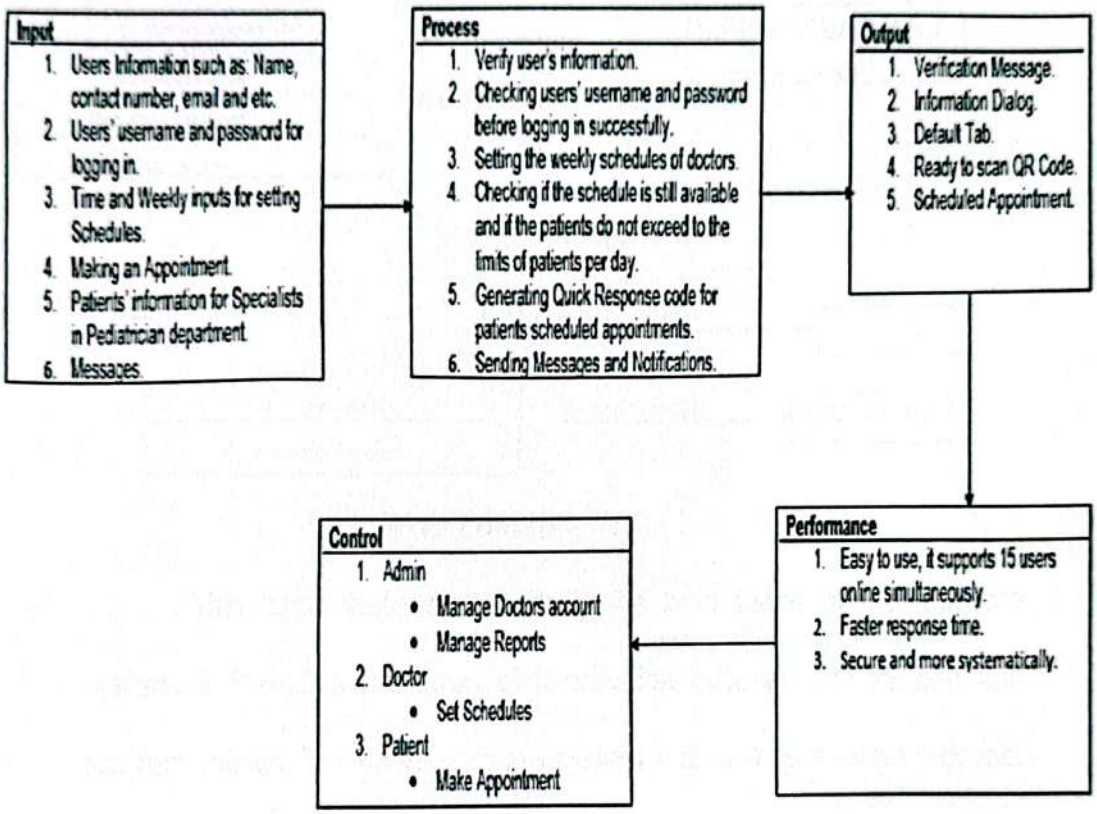


Figure 4.4: Input-Process-Output-Control-Performance

This shows the input, process, output of the system as long as the performance and its controls.

Data and Process Modeling

Context Diagram

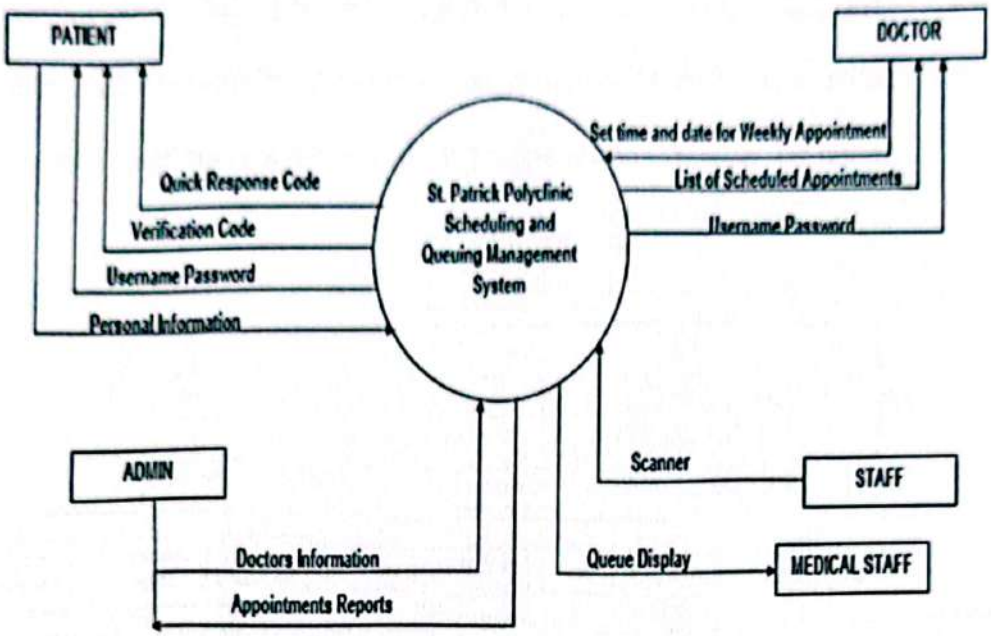


Figure 4.5: Context Diagram

This figure defines and clarify the boundaries of the software system. It identifies the flows of information between the system and external entities. The entire software system is shown as a single process.

System Flowchart

System Flowchart is a physical design tool that shows in general terms the operations that will be performed on information in an information system. The arrows on a system flowchart show the direction that data will flow in around the system rather than the order in which the operations will be carried out.

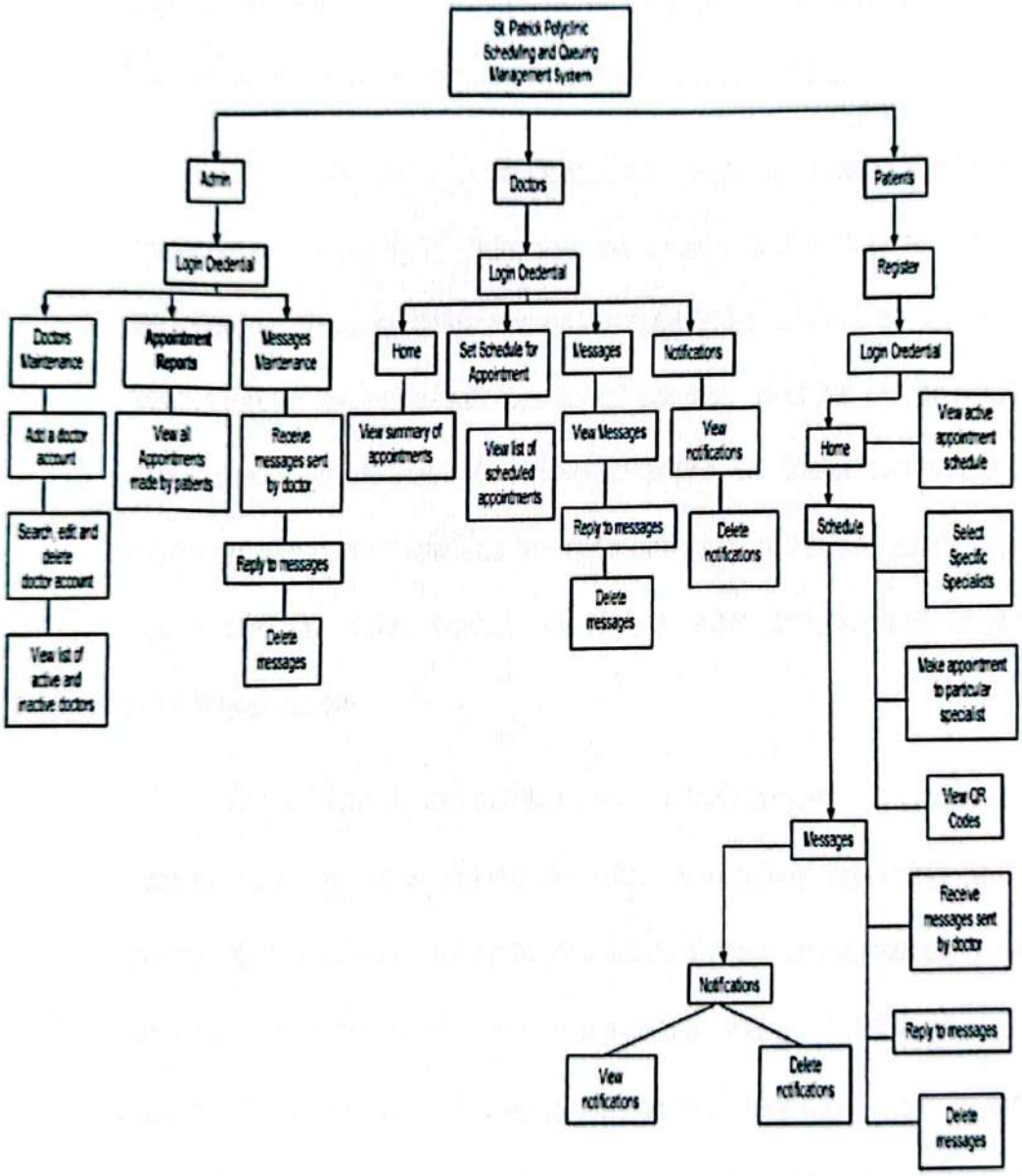


Figure 4.6: System Flowchart

The Scheduling and Queuing Management System at St. Patrick Polyclinic.

System Flow: The first user is the admin; he or she must first log in using the admin account in order to access the doctors' maintenance, appointments, reports, and messaging tab. Admin can add a new doctor account, search, update, and activate or deactivate accounts under the doctor maintenance section. Admins can view all appointments made by patients in Appointments Reports. In Messaging, administrators can receive, respond to, and remove messages sent by doctors.

The second user is the Doctor, who must log in using the log in credentials provided by the administrator. Once logged in, he can access the following tabs or modules: Home/Dashboards, Schedule, Message, and Notification. In the Home tab, the doctor can see all of his or her active appointments for the week. In the Schedule tab, the doctor can create a weekly schedule that includes the date and hour of his/her availability. Doctor can also send, receive, and reply to messages, as well as get system notifications.

The Patient is the primary user of this system: To create an account, the patient must first register. After successfully registering, he or she can log in with his or her log in credentials. Patients have access to the home tab, the schedule tab, the messaging tab, and the notifications tab. Patients can access all active appointments that they have made in the home tab, as well as the QR code for each appointment. In the schedule page, he can arrange an appointment with the appropriate doctor by clicking the specialist button and then clicking the weekly schedules in the

calendar; the QR code will be generated immediately and is ready to scan. Notifications are also available in notification tab.

Program Flowchart

A program flowchart is a diagram that depicts the flow of data when writing a program or algorithm. The tool has four simple symbols with programming code written on them.

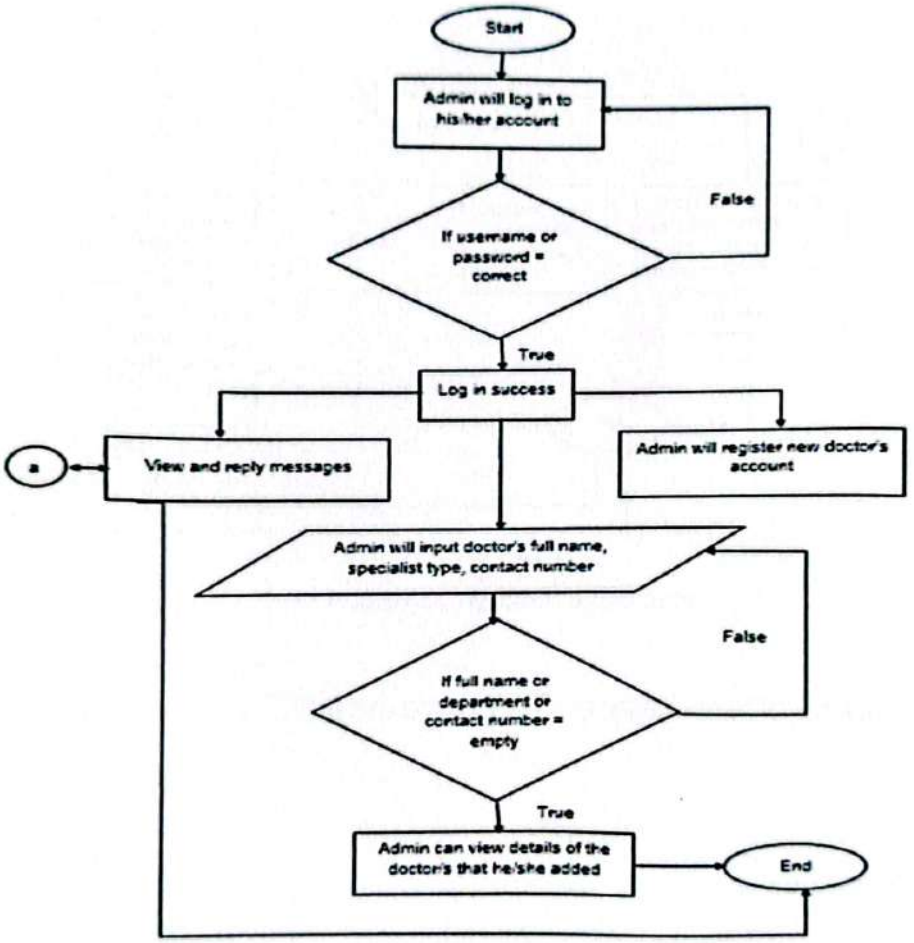


Figure 4.7: Program Flowchart/Administrator

This figure shows the Program flowchart for Administrator.

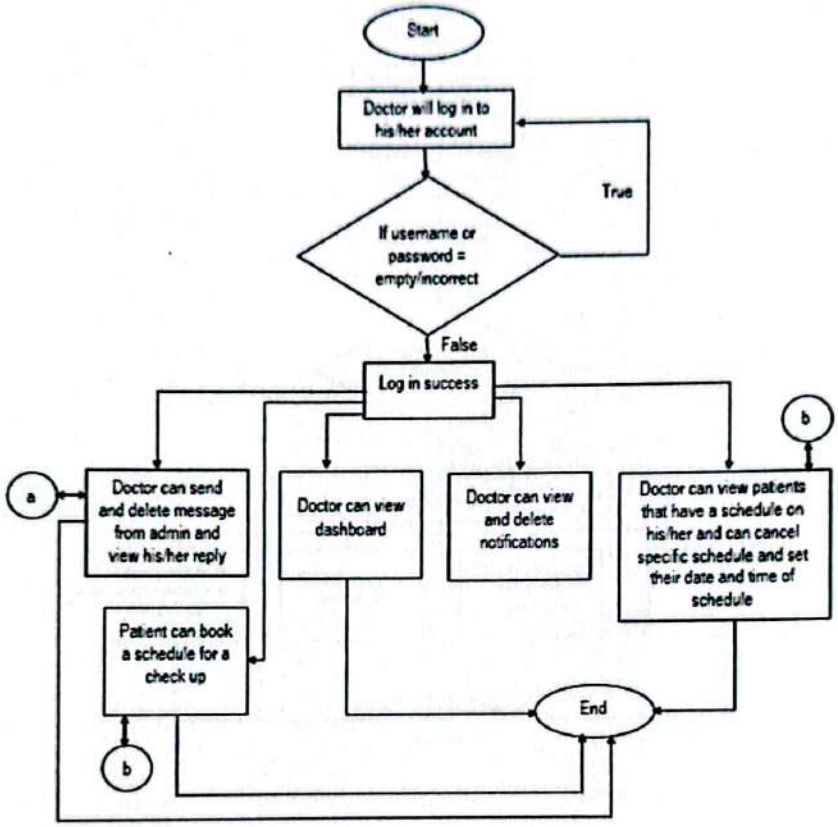


Figure 4.8: Program Flowchart/ Doctor

This figure shows the Program flowchart for doctor.

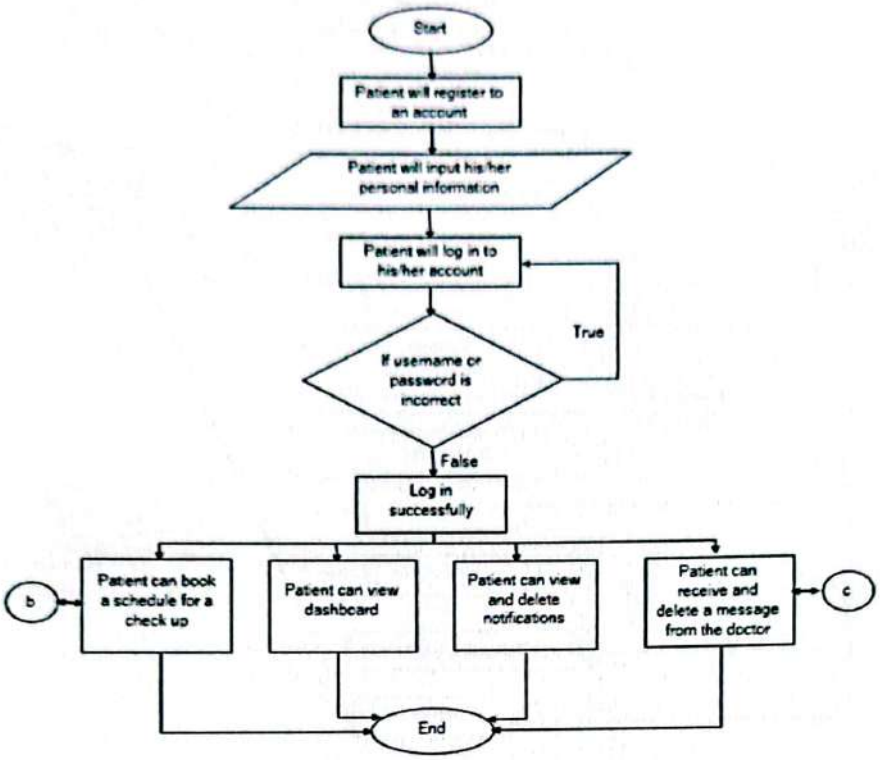


Figure 4.9: Program Flowchart/Patient

This figure shows the Program flowchart for patient.

Object Modeling

Object model diagram specifies the structure and static relationships of the classes in the system. It is sometimes referred to as instance diagrams, which are useful for exploring real world in which the system interacts were divided and the overall decomposition of the system.

Use Case Diagram

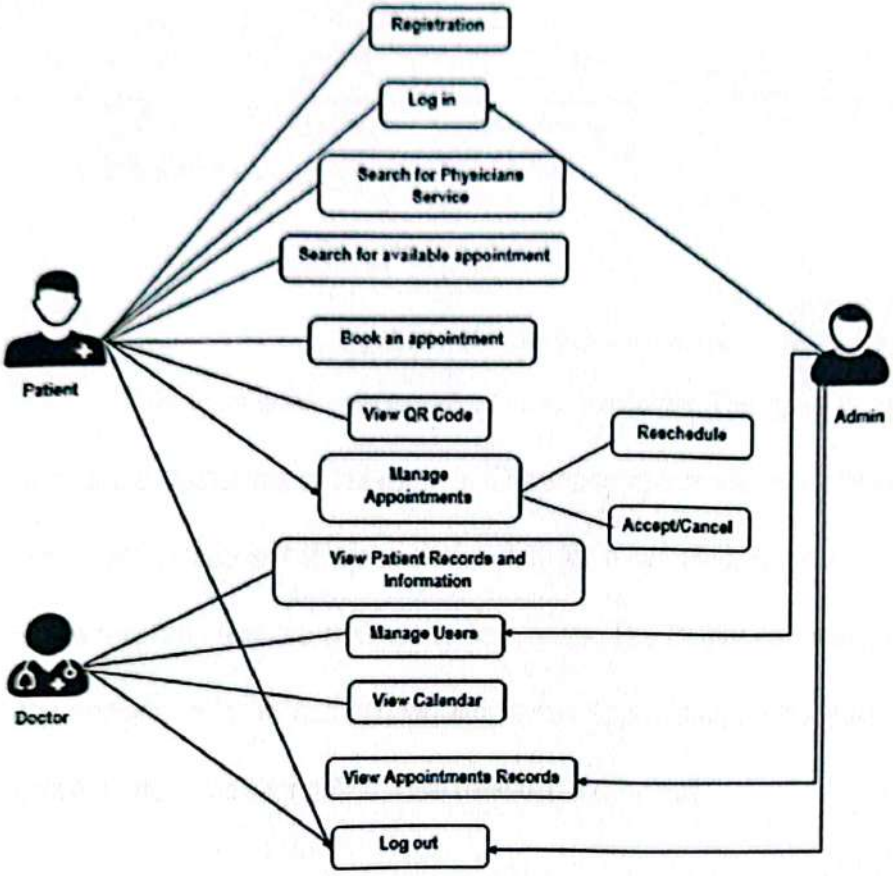


Figure 4.1.0 Use case Diagram for all Users

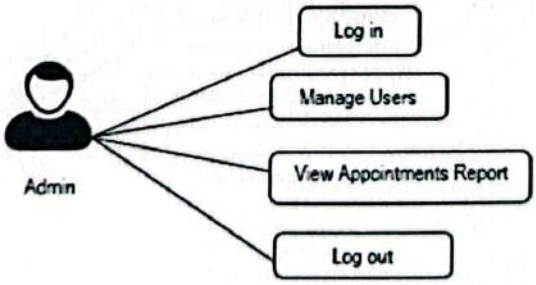


Figure 4.1.1: Use Case Diagram for the Admin

This figure is the use case diagram for the admin. The figure above shows the restrictions of the users in different system functions. There are functions that are exclusive only for a certain user as well as some functions that are common to both users.

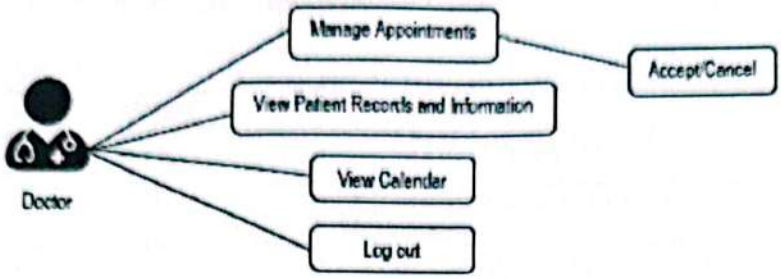


Figure 4.1.2: Use Case Diagram for the Doctor

This figure is the use case diagram for the doctor. The figure above shows the restrictions of the users in different system modules/functions. There are functions that are exclusive only for a certain user as well as some functions that are common to both users. The Doctor can manage the appointments he can accept and cancel appointment, view patient records and some information, view calendar and log out.

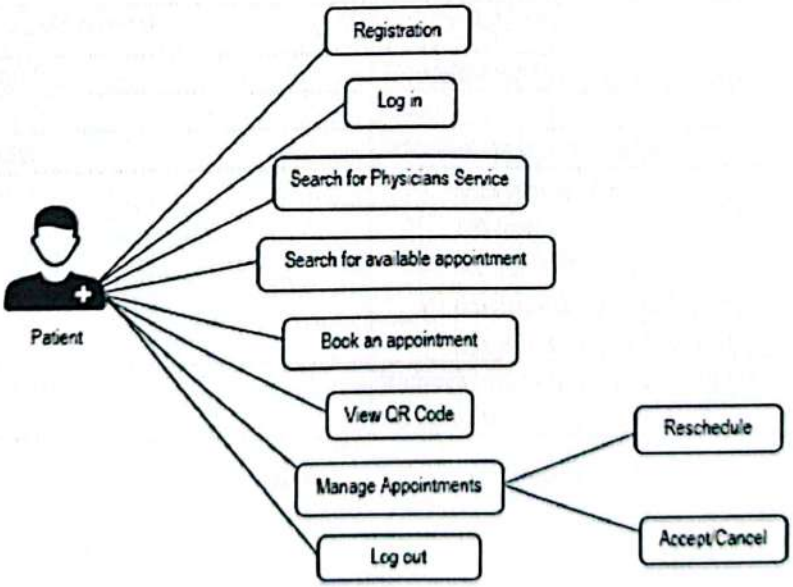


Figure 4.1.3: Use Case Diagram for the Patients

This figure is the use case diagram for the Patient. The figure above shows the restrictions of the users in different system modules/functions.

There are functions that are exclusive only for a certain user as well as some functions that are common to both users.

Use case No. SPP case-101	Login
Brief description	To Login to the system
Actor	Admin, Doctor and Patient
Precondition	Actor needs to have a valid username and passwords
Main flow	• Actor to identify itself as patient or doctor. • Actor to enter username and password. • System validates the entered username and password. • If username and password are validated, system grants access and displays the system main screen.
Alternative flow	If entered username or password is invalid, system access to the system.

Table 4.4: Use Case No. SPP case-101

This table shows the detailed explanation for the "Login" module, who's the actor/ user and how it will work.

Use case No. SPP case-102	Register new Doctor
Brief description	To register new doctor user in the system.
Actor	Admin
Precondition	Actor needs to logon to the system
Main flow	• Actor to enter all personal details of a new user. • Add a specialization. • Confirm registration and a successful registration message will be displayed.
Alternative flow	If any valid details are entered, warning message will be displayed.

Table 4.5: Use Case No. SPP case-102

This table shows the detailed explanation for the "Register" module, it shows how to register new Doctor.

Use case No. SPP case-103	Register new Patient
Brief description	To register new Patient user in the system.
Actor	Patient
Precondition	Actor needs to logon to the system
Main flow	• Actor to enter all personal details of a new user. • System will send verification code to patients' email. • Input verification code, confirm registration and a successful registration message will be displayed.
Alternative flow	If any valid details are entered, warning message will be displayed.

Table 4.6: Use Case No. SPP case-103

This table shows the detailed explanation for the "Register" module for new Patient, it shows how to register new Patient.

Use case No. SPP case-104	Setting a Weekly schedule
Brief description	To set a weekly appointment schedule
Actor	Doctor
Precondition	Actor needs to logon to the system with his user account.
Main flow	• Actor to needs to set his available schedule within a week. • Setting the time and date of schedule • Confirm and a successful registration message will be displayed.
Alternative flow	If any valid details are entered, warning message will be displayed.

Table 4.7: Use Case No. SPP case-104

This table shows the detailed explanation for "Setting weekly schedule" module for doctor, it shows how to set weekly schedule.

Use case No. SPP case-105	Taking an appointment schedule
Brief description	To set an appointment schedule to desired specialists
Actor	Patient
Precondition	Actor needs to logon to the system with his user account.
Main flow	• Actor to needs click schedule tab • Choose and click specialist's button example: Pediatrician • Set an appointment date if Pediatrician button were clicked patient's information will need to be inputted. • Confirm schedule and the system will generate the appointment QR code.
Alternative flow	If any valid details are entered, warning message will be displayed.

Table 4.8: Use Case No. SPP case-105

This table shows the detailed explanation on how to "take an appointment schedule" module for patient, it shows how will the patient take an appointment schedule.

Activity Diagram

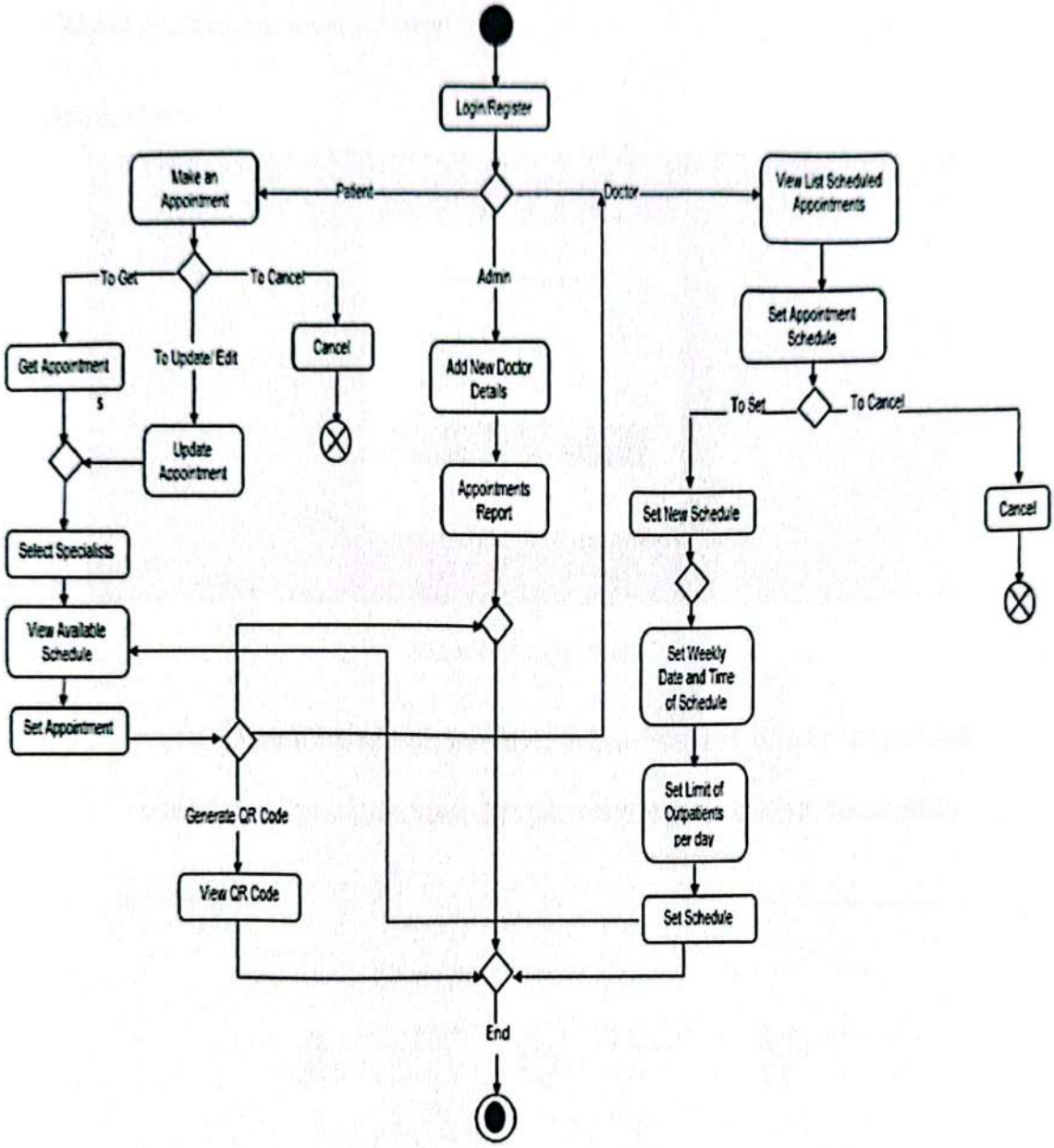


Figure 4.1.4: Activity Diagram

This figure "Activity Diagram" represents the flow of the system, it helps to understand how will the system will work.

Design

Output and User-Interface Design

Administrator

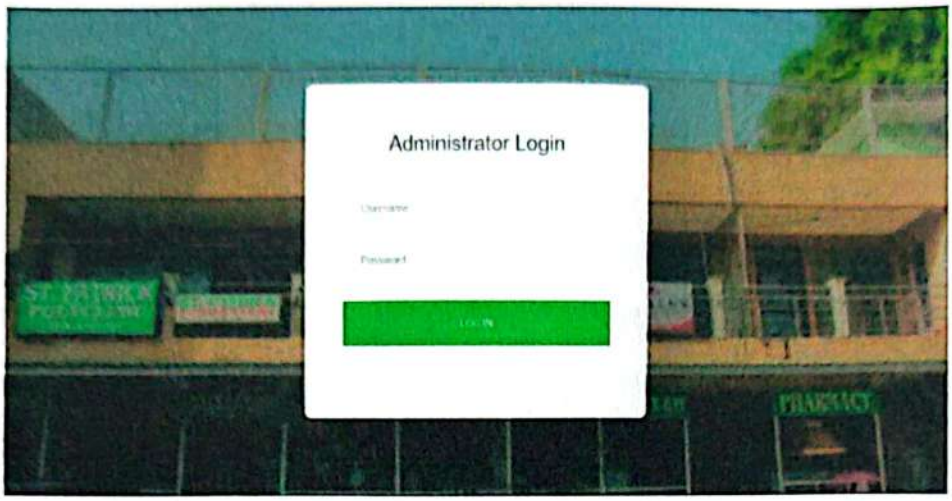


Figure 4.1.5: Log in Form

This is the administrator's log in form interface, which must be filled out with the correct username and password in order to log in successfully.

Figure 4.1.6: Dashboard/Home Page

This is the administrator's Home Page interface, through which the administrator may see all of the doctors' information as well as patient appointment records.

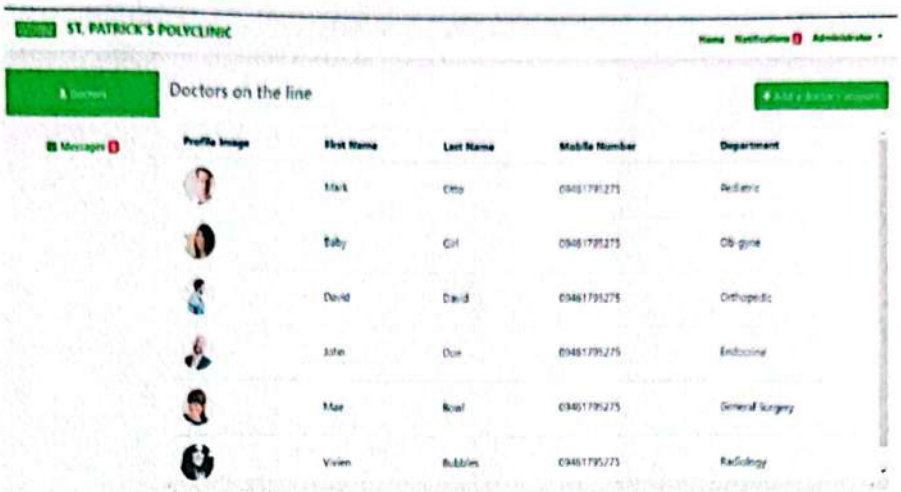


Figure 4.1.7: Doctors Lists Tab

When the administrator wishes to see the lists of active Doctors, he or she can go to the Doctors lists tab.

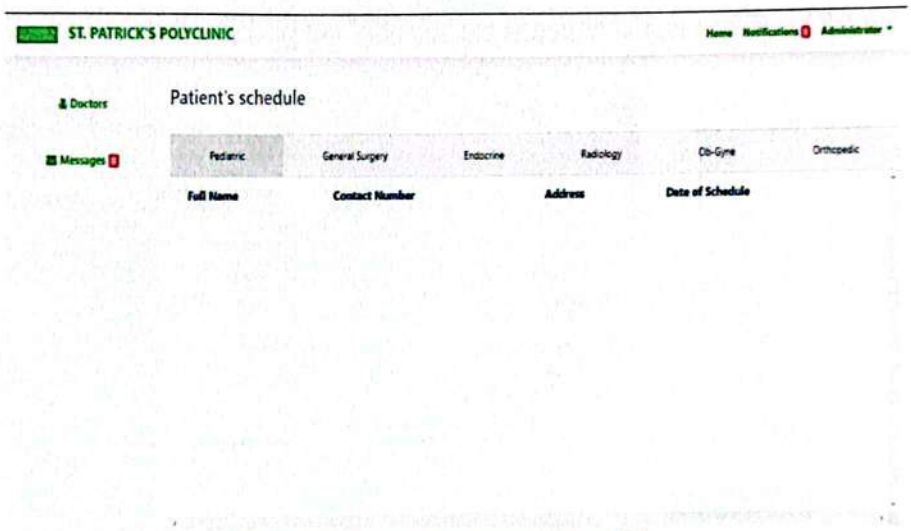


Figure 4.1.8: Patients Appointments Report Tab

When the administrator wishes to see the Patients schedule in each specialty, he or she can go to the Patients schedule tab.

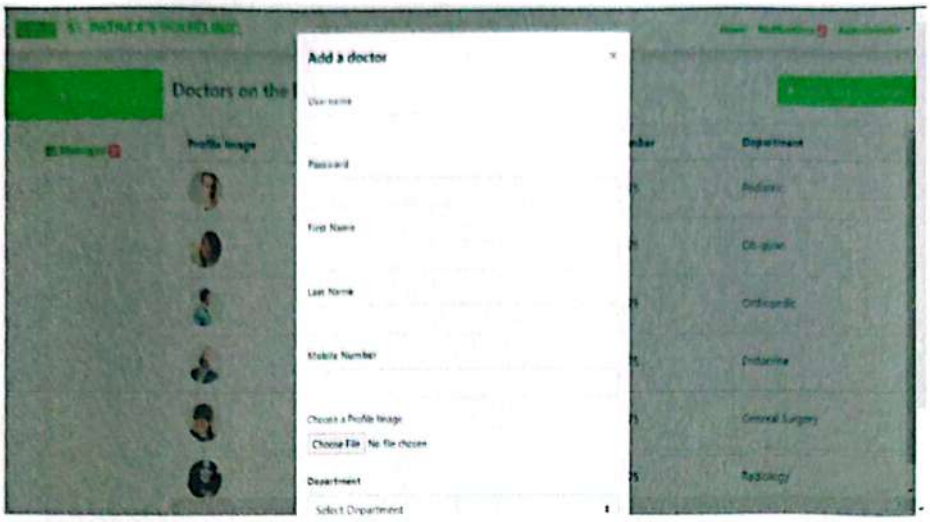


Figure 4.1.9: Add Doctors Account Form

When the administrator wants to add a new doctor's account, he can do so by clicking the "Add doctors account" button and fill out the form.

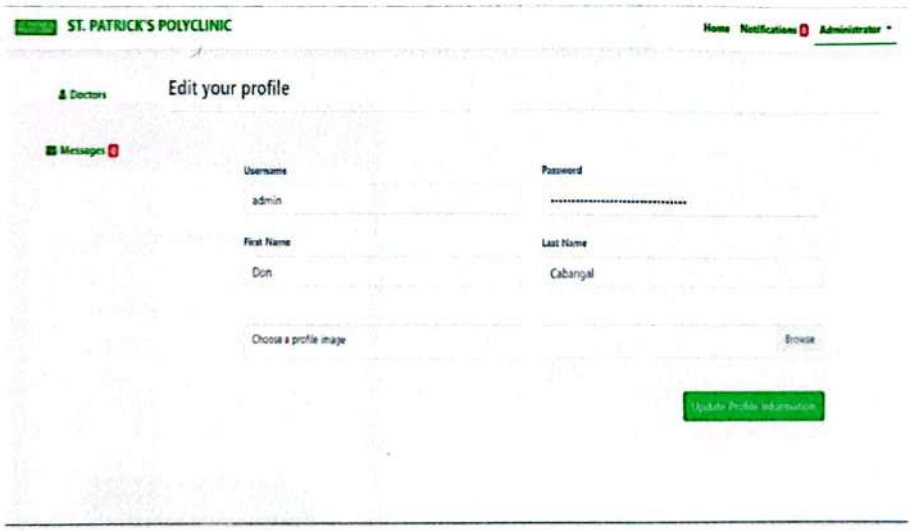


Figure 4.2.0: Updating Administrator Account

When the administrator has to change his or her account, he or she can do so by clicking the "Administrator Tab" in the upper right corner of the page.

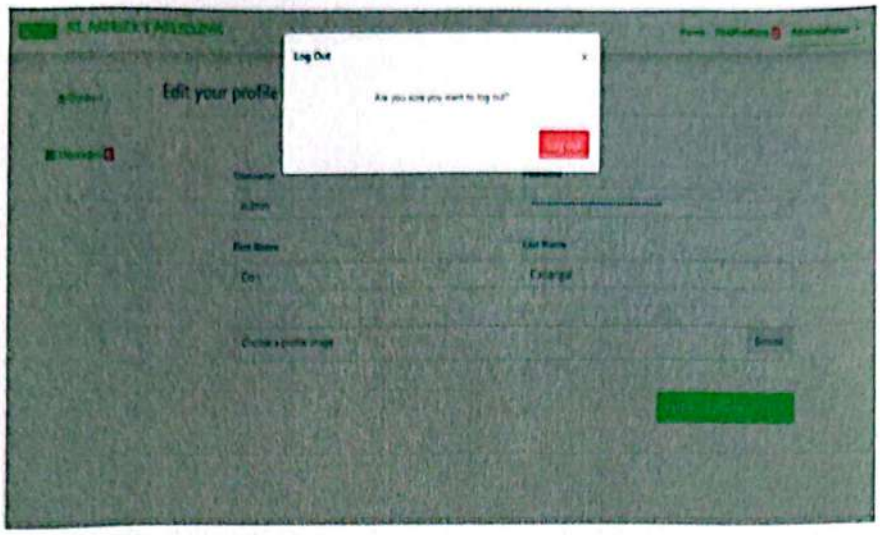


Figure 4.2.1: Log out Confirmation

When the administrator logs out of his or her account, this is the log out confirmation.

Doctor

Figure 4.2.2: Doctor Log in Form

Figure 4.2.3: Home/Dashboard

Doctors must log in using the username and password provided by the admin in order to log in successfully, as seen in Figure 4.2.2, Doctor log in interface in mobile form. The dashboard design, shown in Fig. 4.2.3,

allows the doctor to access all of the tabs as well as his appointment schedule.

Figure 4.2.4: Schedule Tab/ Lists of Patients

Figure 4.2.5: Schedule Tab/ Setting of Schedules

Within the schedule tab, the doctor can view another two tabs: the Patients Schedule (Figure 4.2.4), where the doctor can see all of the patients' scheduled appointments, and the Weekly Schedule (Figure 4.2.5), where the doctor can set his or her weekly schedule simply by filling out all of the necessary forms and submitting and creating new schedules.

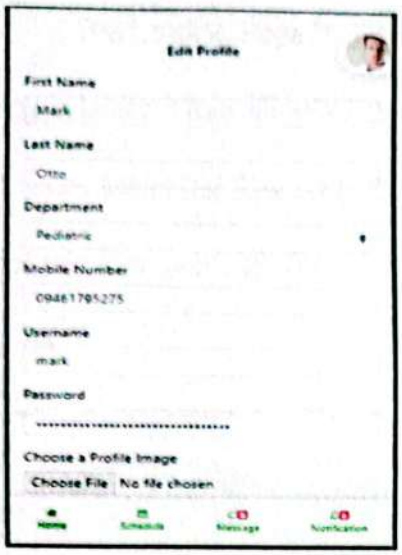


Figure 4.2.6: Updating Doctors Account

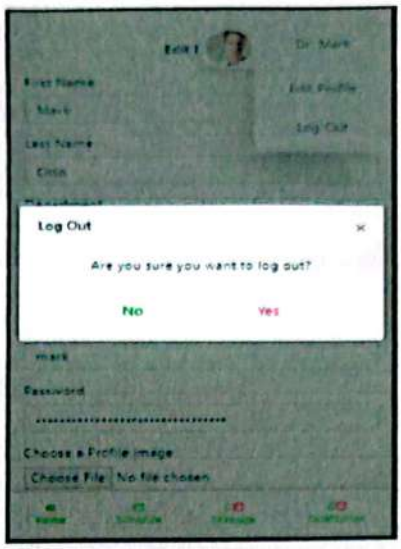


Figure 4.2.7: Log out Confirmation

Fig. 4.2.6 shows on updating doctors account, while Fig. 4.2.7 shows the log out confirmation.

Patient

Figure 4.2.8: Landing Page

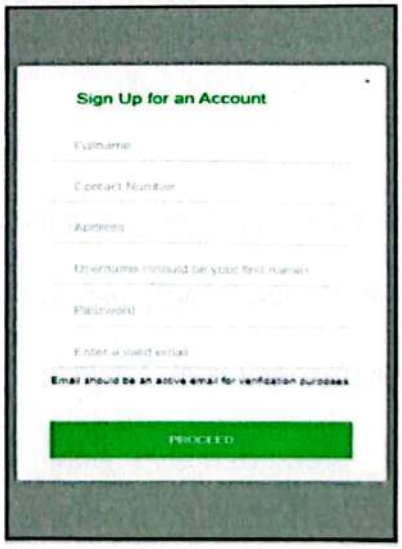


Figure 4.2.9: Sign Up Page/Registration

The Landing Page for Patients is shown in Fig. 4.2.8; they can choose the Log in button if they already have an account or the Sign-up button if they do not. If users select the Sign-up button, Fig. 4.2.9, Sign-up Page Form, they must fill out this form and submit valid email address in order to receive the verification code.

Figure 4.3.0: 2nd Phase of Registration

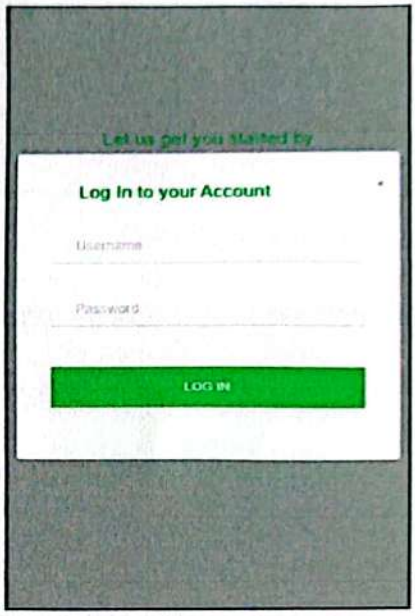


Figure 4.3.1: Log in Form

The second stage of registering a new patient account is to enter the verification code in order to properly register. The following figure shows Fig. 4.3.1, where you may now log in using the username and password that the patient created.

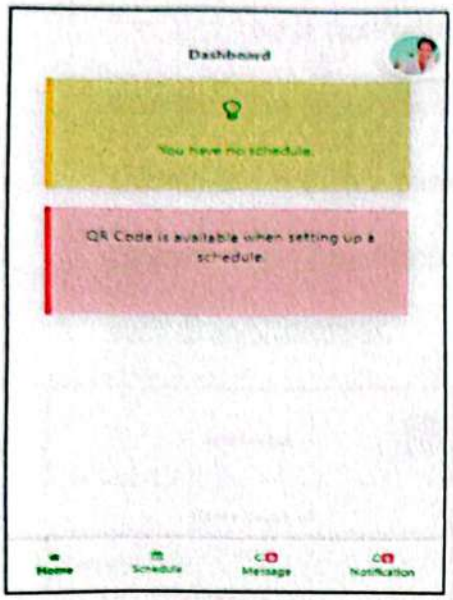


Figure 4.3.2: Dashboard/ Home Page

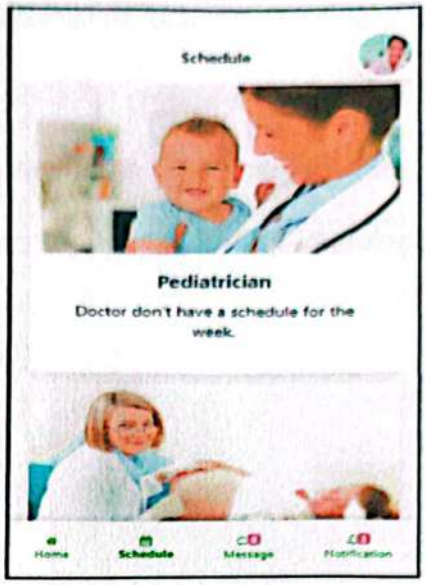


Figure 4.3.3: Schedule Tab

In the dashboard, patients can access their appointment schedules as well as their appointment QR code (Fig. 4.3.2). While under the scheduling page, the patient can select a specialist based on their medical needs (Fig. 4.3.3).

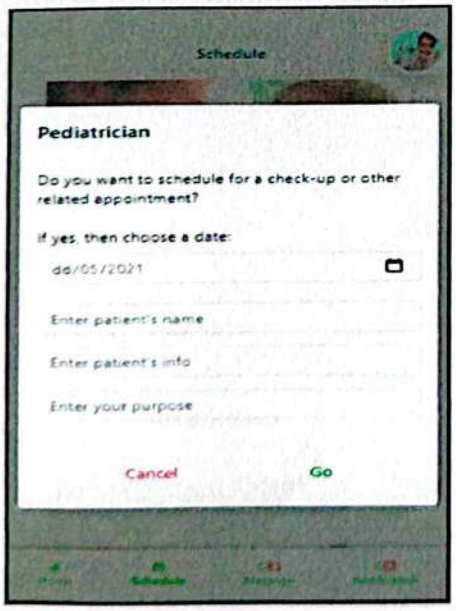


Figure 4.3.4: Pediatrician Appointment Form

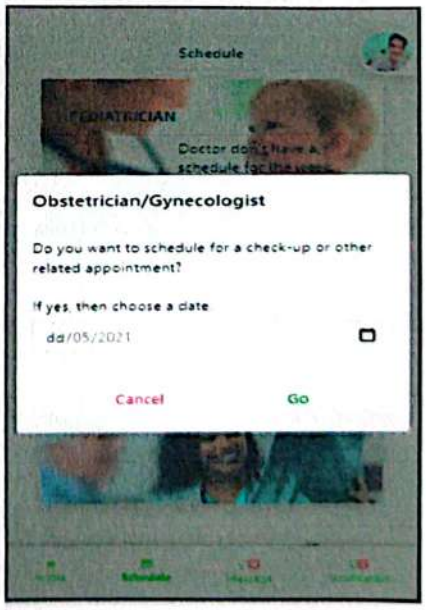


Figure 4:3.5: Sample Form/ Obstetrician

This is the Pediatrician appointment form (Fig. 4.3.4); it differs from the other appointment forms since the patients in pediatrician are children and require a guardian to arrange an appointment for them. In contrast, (Fig. 4.3.5) depicts a sample appointment form for other doctors such as an obstetrician.

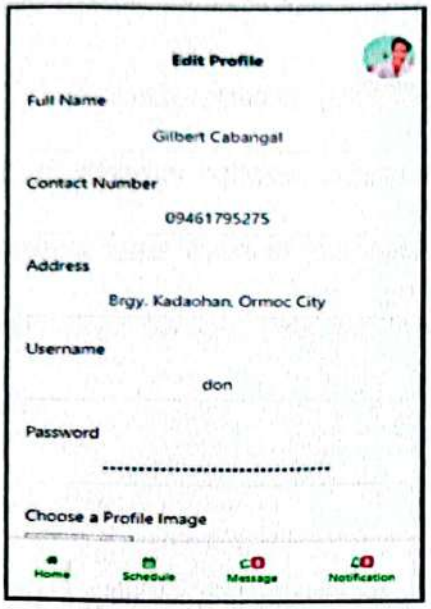


Figure 4.3.6: Updating Patients Profile

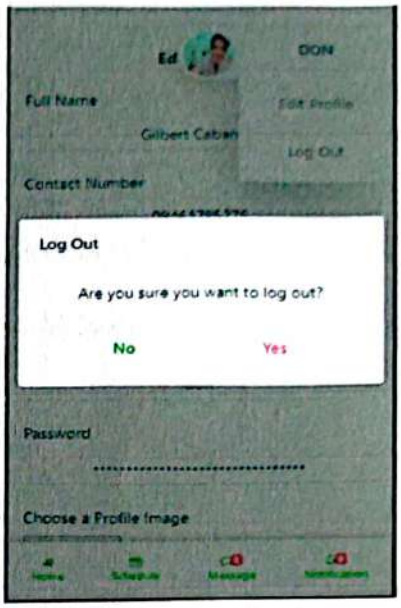


Figure 4.3.7: Log out Confirmation

Figure 4.3.8: Queue Display Status

Fig. 4.3.6 shows on updating patients profile account, while Fig. 4.3.7 shows the log out confirmation and the Fig. 4.3.8 shows the queuing display status of the patients.

Data Design

Entity- Relationship Diagram

An entity-relationship diagram (ERD) is a specialized graphic that illustrates the relationships between entities in a database. This often uses symbols to represent three types of information. This is the easiest way to describe the relationships between data and to get a picture of functionality needed for the system.

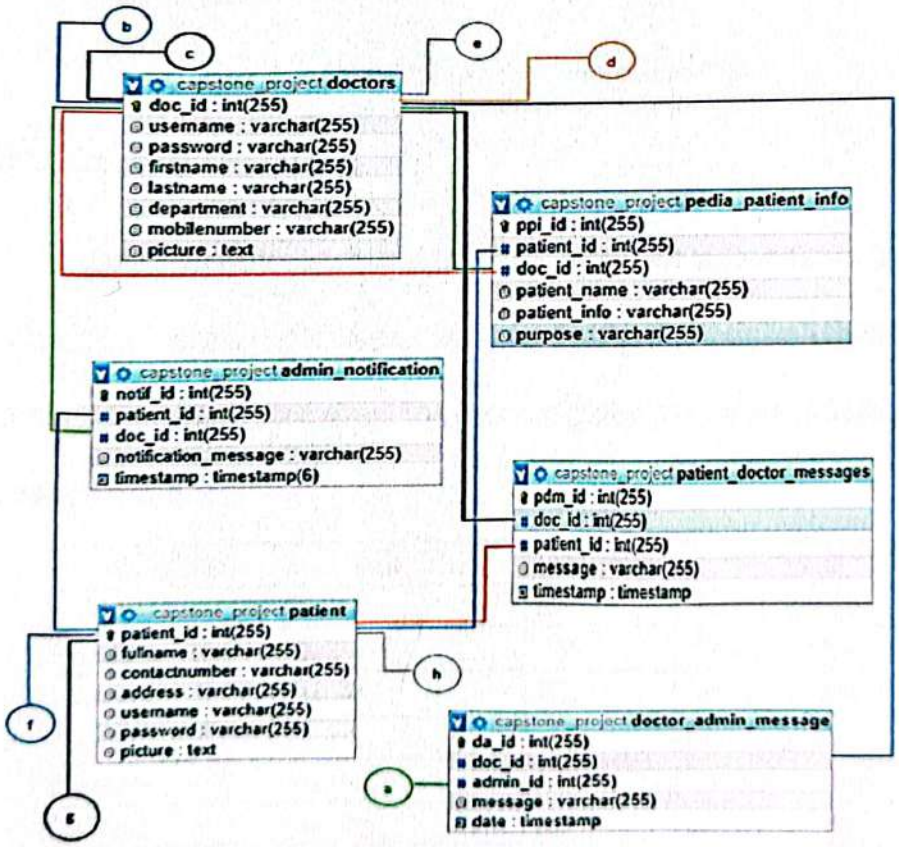


Figure 4.3.9: Entity Relationship Diagram (ERD)

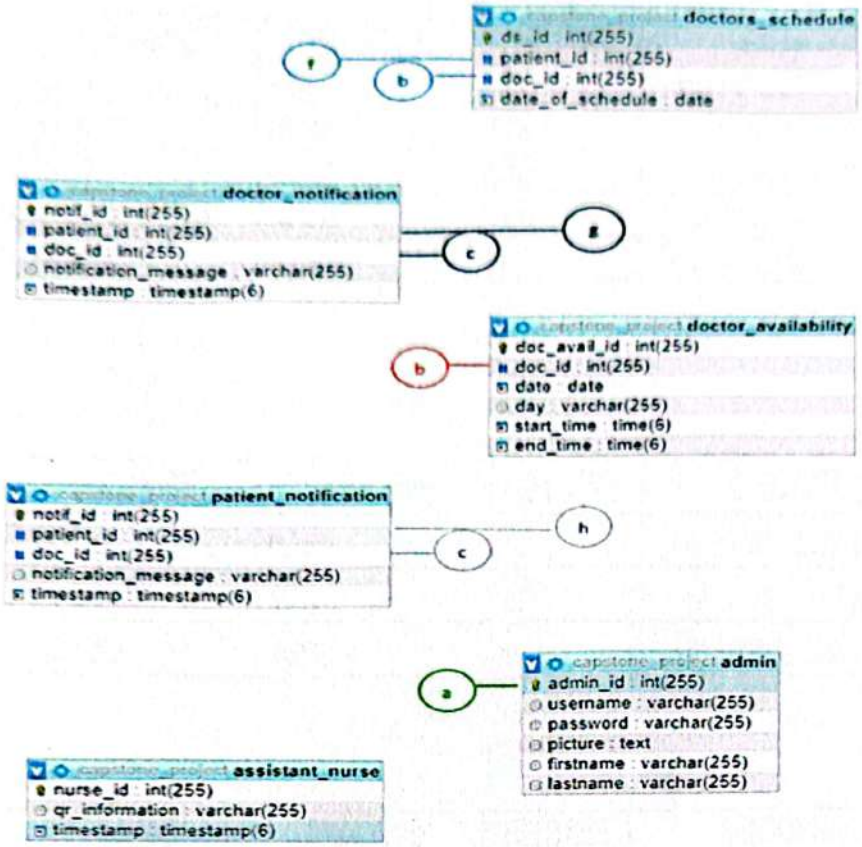


Figure 4.4.0: Entity Relationship Diagram (ERD)

Data Dictionary

Data dictionaries were used to provide definitions of the data used; these included the final data structures for the various tables and their corresponding data fields, description and sizes. The user application programs and interface were developed.

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
admin_id	integer	255	Primary Id for admin
username	varchar	255	Username of admin
password	varchar	255	Password of admin
picture	text	none	Picture of admin
firstname	varchar	255	Firstname of admin
lastname	varchar	255	Lastname of admin

Table 4.9: Data Dictionary for Admin

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
notif_id	integer	255	Primary id for admin notification
patient_id	integer	255	Primary id for patient
doc_id	integer	255	Primary id for doctor
notification_message	varchar	255	Notification message for admin
timestamp	timestamp	6	Timestamp for admin notification

Table 4.1.0: Data Dictionary for Admin Notification

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
nurse_id	integer	255	Primary id for assistant nurse
qr_information	varchar	255	Qr information for patient
timestamp	timestamp	6	Timestamp for qr information

Table 4.1.1: Data Dictionary for Queuing/Assistant Nurse

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
doc_id	integer	255	Primary id doctor
username	varchar	255	Username for doctor
password	varchar	255	Password for doctor
firstname	varchar	255	Firstname for doctor
lastname	varchar	255	Lastname for doctor
department	varchar	255	Department for doctor
mobilenumber	varchar	255	Mobile number for doctor
picture	text	none	Picture for doctor

Table 4.1.2: Data Dictionary for Doctors

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
ds_id	integer	255	Primary id for doctor's schedule
patient_id	integer	255	Primary id for patient
doc_id	integer	255	Primary id for doctor
date_of_schedule	date	255	Date of patient's schedule

Table 4.1.3: Data Dictionary for Doctors Schedule

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
da_id	integer	255	Primary id for doctor to admin message
doc_id	integer	255	Primary id for doctor
admin_id	integer	255	Primary id for admin
message	varchar	255	Message for doctor to admin message
date	timestamp	255	Date for message

Table 4.1.4: Data Dictionary for Doctors Admin Message

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
notif_id	integer	255	Primary id for doctor's notification
patient_id	integer	255	Primary id for patient
doc_id	integer	255	Primary id for doctor
notification_message	varchar	255	Notification for doctor
timestamp	timestamp	6	Timestamp for doctor's notification

Table 4.1.5: Data Dictionary for Doctors Notification

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
patient_id	integer	255	Primary id for patient
fullname	varchar	255	Fullname for patient
contactnumber	varchar	255	Contact number for patient
address	varchar	255	Address for patient
username	varchar	255	Username for patient
password	varchar	255	Password for patient
email	varchar	255	Email for patient
picture	text	none	Picture for patient

Table 4.1.6: Data Dictionary for Patient

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
pdm_id	integer	255	Primary id for doctor and patient messages
doc_id	integer	255	Primary id for doctor
patient_id	integer	255	Primary id for patient
message	varchar	255	Message for patient and doctor messages
timestamp	timestamp	none	Timestamp for patient and doctor's messages

Table 4.1.7: Data Dictionary for Doctor Admin Page

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
notif_id	integer	255	Primary id for patient notification
patient_id	integer	255	Primary id for patient
doc_id	integer	255	Primary for doctor
Notification_message	varchar	255	Notification message for patient
timestamp	timestamp	6	Timestamp for patient notification

Table 4.1.8: Data Dictionary for Doctor Patient Page

FIELD NAME	DATA TYPE	FIELD SIZE	DESCRIPTION
ppi_id	integer	255	Primary for patient pedia info
patient_id	integer	255	Primary id for patient
doc_id	integer	255	Primary if for doctor
patient_name	varchar	255	Name of patient
patient_info	varchar	255	Information of patient
purpose	varchar	255	Purpose of the patient

Table 4.1.9: Data Dictionary for Pedia Patient

System Architecture

This provides a high-level overview of the new system including the system's primary components, the services they provide how they connect. The system is built on a three-tier design that includes user interface, process management and DBMS.

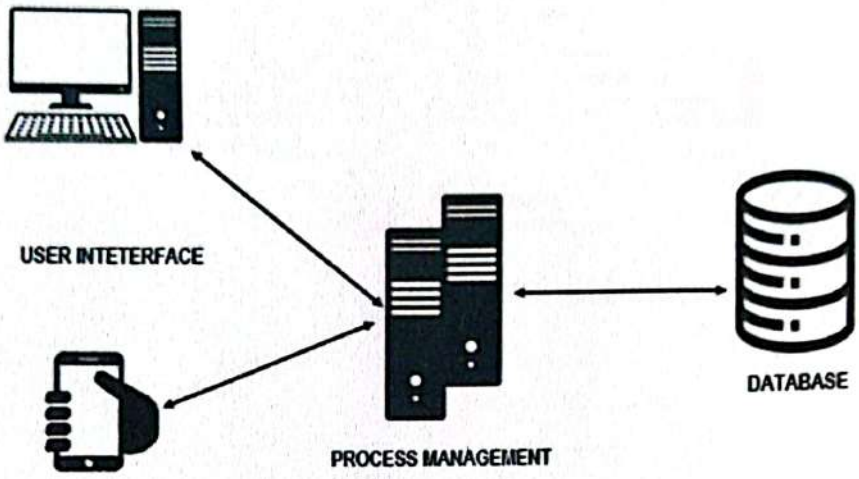


Figure 4.4.1: System Architecture

Network Model

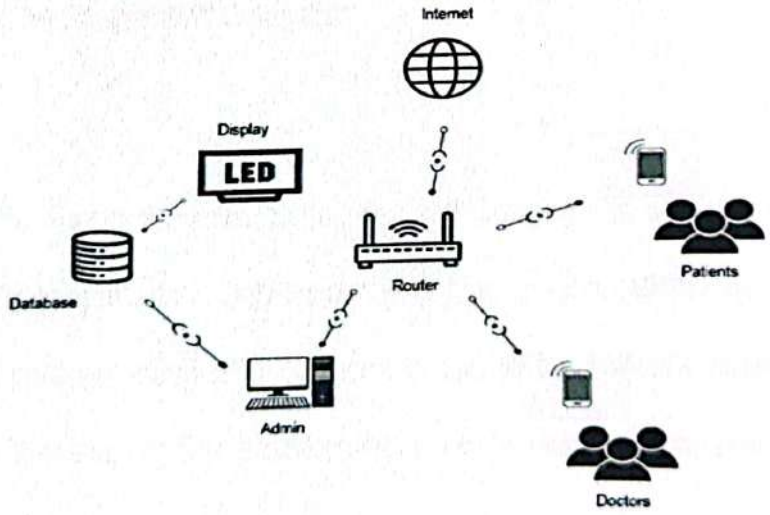


Figure 4.4.2: Network Model of the system

This figure shows the system network model it is designed as a flexible approach to represent objects and their relationships.

Network topology

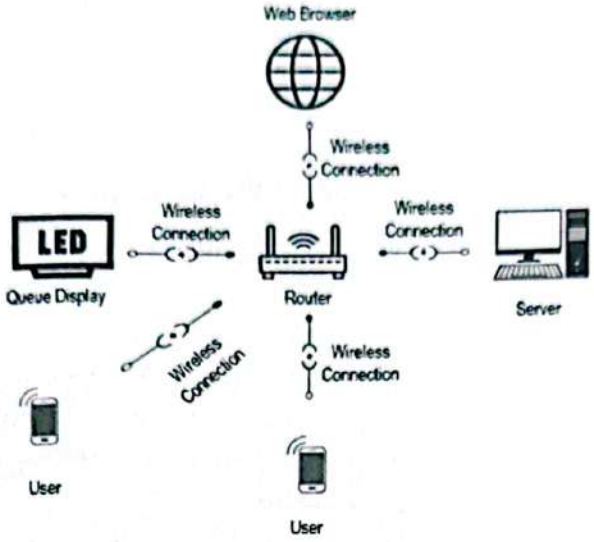


Figure 4.4.3: Star Topology

In star topology, all the components of network are connected to a central hub. It is also one of the most common network setups nowadays. All the data on the star topology passes through the central device before reaching the intended destination.

Security

The St. Patrick Polyclinic Scheduling and Queuing Management System shall protect its sensitive data. Only the admin, doctor, and the patient have access to their own personal information or accounts and to the patient's profiles and information with the use of their password that are encrypted for strong protection.

Development

In order for the system to be realized and developed, hardware and software requirements must be identified. Below are the software and hardware requirements details specified for the development of the system.

Software Specifications

Visual Studio Version 1.56.0

Xampp

Chrome

Hardware Specifications

Desktop/Personal Computer

Program Specifications

PHP

CSS

BOOTHSTRAP

MySQL

Programming Environment

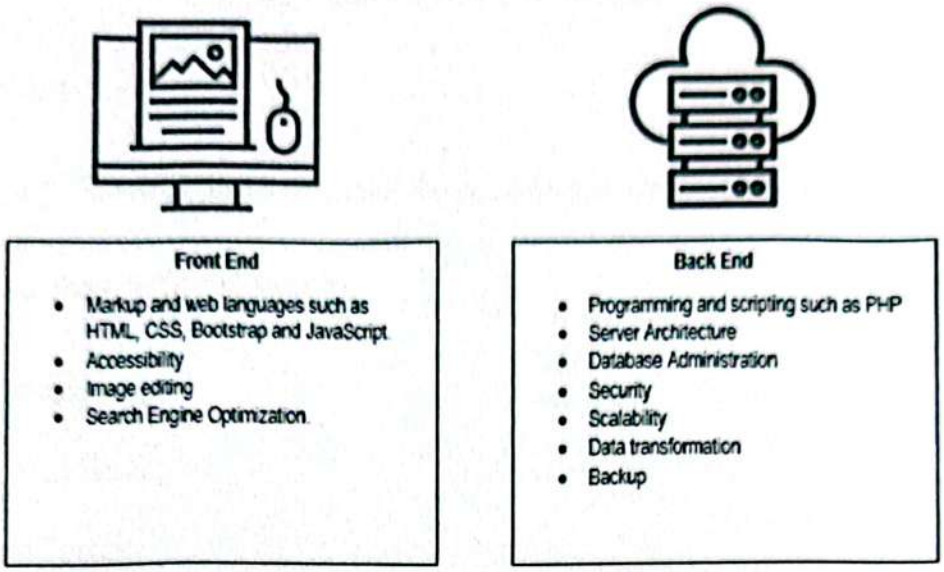


Figure 4.4.4: Front and Back End Environment

This figure shows the front end and back-end environment of the system.

Deployment Diagram

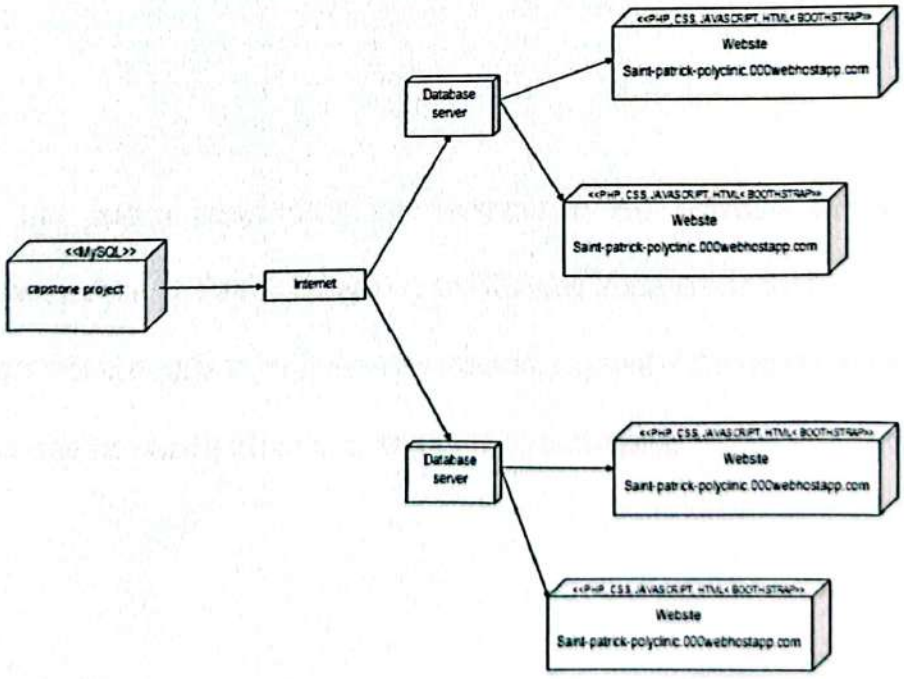


Figure 4.4.5: Deployment Diagram

This figure shows a deployment diagram it shows what software components run on each node and how it is connected.

Test Plan

Objective: To test all the modules with the possible inputs and outcomes.

Steps to make to Test all Modules:

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing

All of this test is needed in order to successfully test the overall functionality of the system.

Testing

Unit Testing

Unit testing would allow the developer to test individual software components of the St. Patrick Scheduling and Queuing Management System. The developer would be able to verify even the minutest segment of the system as well as determine the stability of modules developed on each sprint.

Module: LOG IN			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Leave the email and password empty and press "Log in" button.	Warning message displayed to prompt email and password.	As expected
2	Enter invalid email and password and press "Log in" button.	Warning message displayed to show login failed.	As expected
3	Enter valid email and password and press "Log in" button.	Greeting message displayed. Once user click "OK" on the message, home page is redirected.	As expected

Table 4.2.0: Unit Testing for Log in Module

Module: Registration			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Leave input field empty and click "proceed" button.	Warning message displayed "Please Fill out this Field"	As expected
2	Enter an invalid email like missing "@" then click "proceed" button.	Warning message displayed "Please include "@" in the email address.	As expected
3	Enter invalid or inactive email then click "Proceed" button	Message displayed "Verification code sent to your email". No Code Receive, User can't Register.	As expected
4	Enter valid or active email then click "Proceed" button.	Message displayed "Verification code sent to your email". No Code Receive, User can't Register.	As expected
5	Don't put profile picture for your account then click "Register" button.	Warning message displayed "Please select a file".	As expected
6	Choose profile picture for your account then click "Register" button.	Message displayed "Registered! Your account has been successfully signed up!"	As expected
7	Enter invalid code for verification then click "Register" button.	Message displayed "Registered Failed, Invalid verification code. Please try again with the correct on".	As expected
8	Enter valid code for verification then click "Register" button.	Message displayed "Registered! Your account has been successfully signed up!"	As expected

Table 4.2.1: Unit Testing for Registration Module

Module: Update Personal Information			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Enter no value and press "save" button.	Warning message displayed to prompt email and password.	As expected
2	Leave one of the fields empty and press and press "save" button.	Warning message displayed to show login failed.	As expected
3	Enter all valid details and press "save" button.	Information message displayed to indicate successful update.	As expected

Table 4.2.2: Unit Testing for Updating Personal Information

Module: Set/update weekly schedule for Appointments			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Enter no value and press "submit" button.	Warning message displayed to prompt email and password.	As expected
2	Leave one of the fields empty and press and press "submit" button.	Warning message displayed to prompt email and password.	As expected
3	Enter all value and press "submit" button	Information message displayed Successfully set a schedule.	As expected

Table 4.2.3: Unit Testing for Update weekly Schedule of Appointments

Module: Take an Appointment			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Enter no value and press "submit" button.	Warning message displayed to prompt email and password.	As expected
2	Leave one of the fields empty and press and press "submit" button.	Warning message displayed to show login failed.	As expected
3	Enter all valid details and press "submit" button.	Information message displayed to indicate successful update.	As expected

Table 4.2.4: Unit Testing for Taking an appointment

Integration Testing

Integration testing are combined module and tested as a group. It is conducted to evaluate the component of the system with their specified requirements.

Module: Registration and Account Log in			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Enter no email and password and press "Register" button	Warning message displayed to prompt email and password.	As expected
2	Enter an email address which is already registered to the system.	Warning message displayed to prompt email and password.	As expected
3	Enter unregistered email and enter unmatched passwords. Finally, press "Register" button.	Warning message displayed to prompt email and password.	As expected
4	Enter unregistered email and enter matching passwords. Finally, press "Register" button.	Information message displayed to indicate successful registration. Update personal detail page is redirected.	As expected
5	Enter newly registered email and password and press "Login "button.	Greeting message displayed. Once user click "OK" on the message, home page is redirected.	As expected

Table 4.2.5: Integration Testing Registration and Account Log in

Module: Set Schedule and Take Appointment Schedule			
No.	**Test Case**	**Expected Result**	**Actual Result**
1	Set a weekly schedule with time and appointments limits. Finally, press "submit" button.	Information messages displayed to indicate successful setting of schedules.	As expected
2	Book an appointment based on the doctor's schedule.	Information messages displayed to indicate successful setting of schedules.	As expected
3	Generate QR code after setting an appointment schedule.	Display Qr Code successfully.	As expected
4	View Active Schedule Appointment	Display appointment schedules.	As expected

Table 4.2.6: Integration Testing Set Schedule and Taking Appointment

System Testing

System testing is a form of testing that is done on an entire integrated system to see whether it meets the specifications. All integrated components that have passed integration testing are fed into system testing.

Module: Saint Patrick Polyclinic Scheduling and Queuing Management System			
No	**Test Feature**	**Test Objective**	**Test Status**
1.	Log in	Different types of users are able to login to the system	Passed
2.	Registration	New Accounts are Registered	Passed
3.	Update personal Information	Personal information is updated accurately	Passed
4.	Set Weekly Schedule	Doctors' schedules are set weekly.	Passed
5.	Take Appointment Schedule	Patients choose appointments schedule based on the scheduled set by the doctor.	Passed
6.	Register New Doctor	New doctors account is added.	Passed
7.	View my Appointment	All the appointment made by the user can be viewed and their status.	Passed
8.	View Appointments Records	Only the admin can view all the records.	Passed
9.	Sending Reminder to email	Appointment Reminder will be sent.	Passed
10.	Generate Quick Response Code	Qr code is generated after taking an appointment, this will be used in queuing management.	Passed

Table 4.2.7: System Testing St. Patrick Polyclinic Scheduling and Queuing Management System

Acceptance Testing

This method of testing is used to decide if the software system meets the requirements. This test is used to determine whether or not the system is compliant. Ratings (1) Excellent (2) Satisfied (3) Good (4) Fair (5) Poor

Name of Tester: Jeneveb A. Estrada System User: Patient

Date of Test: June 15 2021

Graphical User Interface Evaluation						
No.	Question	Rating from Tester				
		1	2	3	4	5
1.	Easy to Access Information?	/				
2.	Appropriate Font?		/			
3.	Appropriate Color?	/				

Table 4.2.8: Acceptance Testing/ Graphical User Interface Evaluation

Human Computer Interaction Evaluation						
No.	Question	Rating from Tester				
		1	2	3	4	5
1.	System content easy to understand?	/				
2.	Is the validation Appropriate?	/				
3.	Does the error message help?	/				
4.	Is the system interruption helpful?	/				
5.	Appropriate use of textbox, drop box and etc.	/				

Table 4.2.9: Acceptance Testing/ Human Computer Interaction Evaluation

Overall Functionality Evaluation						
No.	Question	Rating from Tester				
		1	2	3	4	5
1.	Login	/				
2.	Registration	/				
3.	Sending of Verification Code to email	/				
4.	Update Personal Information	/				
5.	Set a new Appointment Schedule	/				
6.	Get new Appointment Schedule	/				
7.	Updating Appointment	/				
8.	Cancel Appointment	/				
9.	Generate QR Code	/				
10.	View My Appointment		/			
11.	List of Scheduled Appointments	/				
12.	View Doctor Record	/				
13.	View Appointments/Patients Report	/				
14.	Sending of Notifications	/				
15.	Logout	/				

Table 4.3.0: Acceptance Testing/ Overall Functionality Evaluation

Chapter V

Implementation Plan

Implementations

In this section, we will go over the processes for effectively implementing this system at the ST. Patrick Polyclinic in Ormoc City if this will be implemented. First, we must analyze all of the requirements for the system to function properly. We will require three different types of resources.

Hardware: All physical devices needed.

First of all, we will state the hardware needed. The website will be developed firstly on localhost since it is much easier. After testing everything on our computer, we can integrate it on the web hosting for user's access. Therefore, this is what we will be needed in implementing the system.

- Internet Connection and a Router.
- PC (Personal Computer): We don't need a powerful computer, merely 2GB RAM minimum or 8GB suggested. We won't require much secondary memory because our system/website will be hosted on a remote server (300GB will be more than enough). An Intel i3 CPU will suffice as well.

  This computer will be used by the administrator.

  All this hardware is already included to the budget calculations.
- Other hardware needed is smartphone any type of Smartphone that can be use in scanning the QR codes.

The doctor and the staff will just use their smartphones in accessing the website, so that the management can make a less in budget.

Software: Programs needed for the hardware.

All software needed;

- My SQL Server/ Apache web Server
- Xampp
- PHP
- Web Host: 000webhost
- Modern Internet Browsers

Staff: people need in order to implement the system well.

The management team includes many doctors and medical personnel who will provide all clinic-related services. For website development and future maintenance, we will need to engage a system administrator. An external system administrator is ideal since it will be significantly less expensive than hiring a full-time system administrator, and the website is not large enough to warrant a full-time system administrator.

Implementation Checklists		
1. Set-Up the Hardware: Router and Internet Connection PC	In implementing the system, we must set-up first the hardware that we will going to use. The router and the internet connection we must choose an internet provider with a good and faster connection in the area. We must also set-up the PC that we will going to use as a server and also for the system administrator, we will connect it to the router through LAN (Local Area Network).	Notes:
2. Install all the needed software's, applications, or programs such as: My SQL Server Internet Browsers PHP	After we finish setting-up all the hardware that we will needed, the next thing that we will going to do is to install all the needed software's to the PC Server. Then try to test the it, if it will be going to work. If it worked then try to test the website/ system using the PC Server and try to connect also your smartphones if it will be going to work.	Notes:
3. Staff: Doctors, Medical Staff and other Staff (operates the Qr code Scanner).	After we finish setting -up all the things that we needed, the next thing that we will going to do is to teach or train all the staff on how to use the system. We must teach them and make them familiarize to the new system that they will going to use in order to implement the system successfully.	Notes:
4. Testing for Deployment	The Last thing that we will going to do is to have a testing following the system flow. Like making an account for the doctor, then setting up a schedule, patients' registration and making an appointment, to queuing flow and so on.	Notes:

Table 4.3.1: Implementation Checklists

This table shows the checklists that are needed when implementing the system.

Implementation Contingency

Scenario	Trigger	Response	Who to Inform?	Key Responsibilities	Timeline	
				What	What	When
No Internet Connection	Depicts of Router or Internet Receiver	Use Mobile data in viewing scheduled patients	Administrator	Check the possible depicts of router or receiver.	While the admin checks for any potential depictions, doctors and employees can use cellphone data as a stand-in.	As soon as possible

Table 4.3.2: Implementation Contingency

Infrastructure

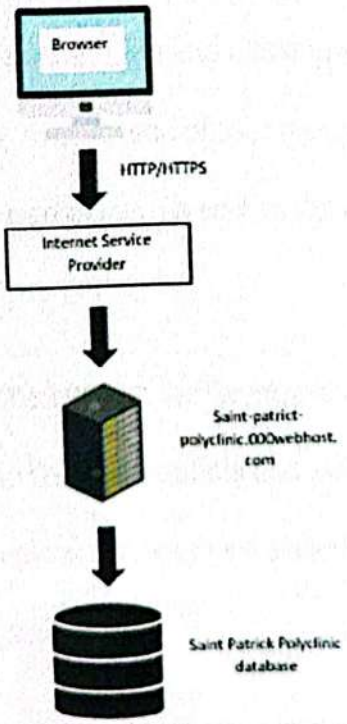


Figure 4.4.6: Infrastructure

Chapter VI

Conclusion and Recommendation

Conclusion

During the project's development, the following objectives were fulfilled in order to achieve the goal of building an online appointment scheduling and queuing management system that users (patients, doctors, and administrators) could utilize in their everyday transactions.

The developer analyzed all of the necessary data or information to meet the main objectives of this project; to provide an easy and cost-effective method of scheduling an appointment, it is desired to make an effortless and fastest way of making an appointment for the patients in St. Patrick Polyclinic.

The clinic's scheduling and queuing systems went through several stages, including the construction of user groups and the assignment of appropriate access credentials, as well as the use extensions to create pages with specific criteria.

Despite the fact that the developers had to overcome numerous obstacles in order to create an orderly and well-managed schedule of appointments, as well as to smooth the patients' appointment traffic and achieve the other study objectives.

We believe that all of the requirements and objectives were satisfied in the end. Throughout the creation of this system, the development team learned

several tactics, allowing it to be developed and maintained with the essential efficiency.

Recommendations

As future enhancements, we may suggest a variety of features that would make the system more useful.

- User Feedback: We may solicit feedback from users about the system and what they would want to see added to it.
- Online Consultations: Originally, users had to visit to the clinic to get inspected. If people express an interest in online consultations that do not require a physical examination, this could be a useful option.
- Online Prescriptions: If we incorporate online consultations, perhaps we can also integrate online doctor prescriptions to patients, which would help to reduce the use of paper or prescription paper.
- Online Payments: Until now, all payments have been made in person at the clinic. We could incorporate this function if we integrate online consultations or if people indicate a preference for paying online rather than in cash at the clinic.

REFERENCES

1] Obulor R., Eke B. O. (2016) Outpatient Queuing Model Development for Hospital Appointment System http://ijseas.com/volume2/v2i4/ijseas20160404.pdf

2] Peng Zao, Illhoi Yoo, Jaie Lavoie, Beau James Lavoie, Eduardo Simoes (2017) Web-Based Medical Appointment Systems: A systematic Review https://www.researchgate.net/publication/303470976_A_simulation_of_a_hospital_queuing_system_-_Case_study_Thika_Level_Five_Hospital

3] Tochukwo A. Ikwunne (2016) "Optimality Test for Multi- Server Queuing Model with Homogenous Server in the Out-Patient Department (OPD) of Nigeria Teaching Hospitals https://www.researchgate.net/publication/301241366_Optimality_Test_for_Multi-Sever_Queuing_Model_with_Homogenous_Server_in_the_Out-Patient_Department_OPD_of_Nigeria_Teaching_Hospitals

4] Iren Sherly (2016) "Online Appointment Reservation and Scheduling for Healthcare https://docplayer.net/17551943-Online-appointment-reservation-and-scheduling-for-healthcare-a-detailed-study.html

5] A.A Onoja and Msugh Kembe (2018) "The Application of Queuing theory on Patient Waiting time in Ante- Natal Care Clinic https://www.researchgate.net/publication/323643720

6] Pan X- Geng N-Xie X – Wen J (2019) Managing Appointments with waiting time targets and random walk-ins www.sciencedirect.com/science/article/pii/S0305048318307126?via%3Dihub

7] Arik Senderovich and Matthias Weidlich (2016) Conformance checking and performance improvement in scheduled processes

8] David H. Berger(2016) Use of simulation to assess a statistically driven surgical scheduling system

9] Maryam Tufail (August 2018) "Online Polyclinic Appointment And Database Management System" Master's thesis Master's Degree in Information Technology

https://www.theseus.fi/bitstream/handle/10024/152390/Tufail_Maryam.pdf?sequence

10] Cara Rudnic, Olga (2016-2017) "Online Medical Appointment Scheduling System" Trabajo Fin de Grado Grado en Ingeniería Informática https://riunet.upv.es/bitstream/handle/10251/88831/CARA%20-%20Sistema%20de%20cita%20online%20para%20una%20consulta%20m%C3%A9dica.pdf?sequence=1

APPENDIX A: RELEVANT SOURCE CODE

Source Code (Patient)

Confirm_cancel_schedule.php

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
  <link rel="stylesheet" href="../style/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
  <link href="../custom-styles/style.css" rel="stylesheet">
 </head>
 <body>
<?php
 require ("../database/dbconnect.php");
 $ds_id = $_REQUEST['ds_id'];
 $patient_id = $_REQUEST['patient_id'];
 $doc_id = $_REQUEST['doc_id'];
 $date_of_schedule = $_REQUEST['date_of_schedule'];
 $notif_message = "Your have cancelled a schedule on " . date("F j, Y", strtotime(
$date_of_schedule)) . ".";
 $doc_notif_message = "A patient cancelled a schedule on " . date("F j, Y", strtoti
me($date_of_schedule)) . ".";
 $cancel_schedule = mysqli_query($connection,"DELETE from `doctors_schedule
` where ds_id='$ds_id'");
 $insert_patient_notification = mysqli_query($connection,"INSERT INTO `patient_
notification`(`patient_id`, `doc_id`, `notification_message`, `timestamp`) VALUES ('
$patient_id','$doc_id','$notif_message', 'NOW()')");
 $insert_doctor_notification = mysqli_query($connection,"INSERT INTO `doctor_n
otification`(`patient_id`, `doc_id`, `notification_message`, `timestamp`) VALUES ('$
patient_id','$doc_id','$doc_notif_message','NOW()')");
 if (!$cancel_schedule && !$insert_patient_notification && !$insert_doctor_notificat
ion && !$insert_admin_notification)
 {
```

```
  echo"<script>alert('Incorrect Query!')</script>";
 }
 else
 {
  echo '<script>
  setTimeout(function() {
     swal({
        title: "Cancelled",
        text: "Your schedule has been cancelled successfully.",
        type: "success"
     }, function() {
        window.location = "../contents/dashboard.php";
     });
  }, 1000);
 </script>';
 }
  <?php include '../includes/scripts.php'; ?>
 </body>
<html>
```

Confirm_reschedule_patient.php

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
  <link href="../custom-styles/style.css" rel="stylesheet">
 </head>
 <body>
  <?php
   require ("../database/dbconnect.php");
   $ds_id = $_REQUEST['ds_id'];
   $patient_id = $_REQUEST['patient_id'];
   $doc_id = $_REQUEST['doc_id'];
   $date_of_schedule = $_REQUEST['date_of_schedule'];
```

```
    $date_of_reschedule = $_REQUEST['date_of_reschedule'];
    $notif_message = "You have rescheduled your appointment on " . date("F j, Y"
, strtotime($date_of_schedule)) . " to " . date("F j, Y", strtotime($date_of_reschedul
e)) . " successfully.";
    $doc_notif_message = "A patient reschceduled an appointment on " . date("F j
, Y", strtotime($date_of_schedule)) . " to " . date("F j, Y", strtotime($date_of_resch
edule)) . ".";
    $compareday = date("l", strtotime($date_of_reschedule));
    $flag=0;
    $query ="SELECT * FROM doctor_availability WHERE doc_id = '$doc_id'";
    $results = $connection->query($query);
    while($rs=$results->fetch_assoc())
    {
     if($rs["day"]==$compareday)
     {
      $flag++;
      break;
     }
    }
    if ($flag==0)
    {
     echo '<script>
     setTimeout(function() {
      swal({
       title: "Doctor unavailable on selected date, please choose another date",

       text: "Tap OK to continue",
       type: "error"
      }, function() {
       window.location = "../contents/dashboard.php";
      });
     }, 1000);
     </script>';
    }
    else
    {
     $reschedule = mysqli_query($connection,"UPDATE `doctors_schedule` SET
`patient_id`='$patient_id',`doc_id`='$doc_id',`date_of_schedule`='$date_of_resche
dule' WHERE ds_id = '$ds_id'");
```

```
    $insert_patient_notification = mysqli_query($connection,"INSERT INTO `pati
ent_notification`(`patient_id`, `doc_id`, `notification_message`, `timestamp`) VALU
ES ('$patient_id','$doc_id','$notif_message', 'NOW()')");
    $insert_doctor_notification = mysqli_query($connection,"INSERT INTO `doct
or_notification`(`patient_id`, `doc_id`, `notification_message`, `timestamp`) VALUE
S ('$patient_id','$doc_id','$doc_notif_message','NOW()')");
    if (!$reschedule && !$insert_patient_notification && !$insert_doctor_notificatio
n && !$insert_admin_notification)
    {
     echo"<script>alert('Incorrect Query!')</script>";
    }
    else
    {
     echo '<script>
     setTimeout(function() {
        swal({
           title: "Rescheduled Success.",
           text: "Your appointment has been rescheduled successfully.",
           type: "success"
        }, function() {
           window.location = "../contents/dashboard.php";
        });
      }, 1000);
     </script>';
     }
    }
  <?php include '../includes/scripts.php'; ?>
 </body>
<html>
```

Register_patient.php

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
```

```
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
  <link href="../custom-styles/style.css" rel="stylesheet">
 </head>
 <body>
  <?php
   require("../database/dbconnect.php");
   if(isset($_POST['submit']))
   {
    if($_POST['code'] == "sppormoc")
    {
     $fullname =$_POST['fullname'];
     $contactnumber =$_POST['contactnumber'];
     $address =$_POST['address'];
     $username =$_POST['username'];
     $password =md5($_POST['password']);
     $email = $_POST['email'];
     $picture = $_FILES['picture']['name'];
     $filetmpname = $_FILES['picture']['tmp_name'];
     move_uploaded_file($filetmpname,"../images/$picture");
     $saveinfo= mysqli_query($connection,"INSERT INTO `patient`(fullname,con
tactnumber,address,username,password,email,picture)
                          values ('$fullname','$contactnumber','$address','$user
name','$password','$email','$picture')");
     echo '<script>
     setTimeout(function() {
       swal({
         title: "Registered",
         text: "Your account has been successfully signed up!",
         type: "success"
       }, function() {
         window.location = "../index.php";
       });
     }, 1000);
     </script>';
    }
    else
    {
     echo '<script>
     setTimeout(function() {
```

```
            swal({
              title: "Registration Failed",
              text: "Invalid verification code. Please try again with the correct one.",
              type: "error"
            }, function() {
              window.location = "../index.php";
            });
          }, 1000);
          </script>';
        }
      }
    ?>
    <?php include '../includes/scripts.php'; ?>
  </body>
</html>
```

Schedules.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
    <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
    <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
    <link rel="stylesheet" href="../style/sweetalert.css">
    <link href="../custom-styles/style.css" rel="stylesheet">
  </head>
  <body>
    <?php
      include('../library/phpqrcode/qrlib.php');
      require("../database/dbconnect.php");
      if(isset($_POST['schedule_to_pedia'])){
        $username = ($_SESSION['login_user']);
        $patient_id = $_REQUEST['patient_id'];
        $date = strtotime($_POST['date']);
        $fullname = $_POST['fullname'];
        $address = $_POST['address'];
        $contactnumber = $_POST['contactnumber'];
```

```
$newdate = date('Y-m-d', $date);
$patient_name = $_POST['patient_name'];
$patient_info = $_POST['patient_info'];
$purpose = $_POST['purpose'];
$patient_count = 0;
$compareday = date("l", $date);
$flag=0;
$query ="SELECT * FROM doctor_availability WHERE doc_id = '1'";
$results = $connection->query($query);
while($rs=$results->fetch_assoc()){
 if($rs["day"]==$compareday){
  $flag++;
  break;
 }}
if ($flag==0) {
 echo '<script>
 setTimeout(function() {
  swal({
   title: "Doctor unavailable on selected date, please choose another
          date",text: "Tap OK to continue",
   type: "error"
  }, function() {
   window.location = "../contents/schedule.php";
  });
 }, 1000);
 </script>';
}else {
  $patient_count1 = "SELECT date_of_schedule, COUNT(date_of_schedule)
FROM `doctors_schedule` GROUP BY date_of_schedule HAVING COUNT(date_
of_schedule) > 1;";
  $count = $connection->query($patient_count1);
  while($count1=$count->fetch_assoc()){
   $count1++;
   break;
  }if($count1<=30){
   $tempdir = '../library/qrimages/';
   $filename = $username;
   $codecontents = 'Fullname: '.$fullname.' - Address: '.$address.' - Contact
Number: '.$contactnumber.' - Date of Schedule: '.date("F j, Y", strtotime($newdate)
).' - Department: Pediatrics - Code: stptrckplyclnc';
```

```
        $code = QRcode::png($codecontents, $tempdir."".$filename.'.png', QR_EC
LEVEL_L, 5);
        $saveinfo1 = mysqli_query($connection,"INSERT INTO `doctors_schedule
`(patient_id,doc_id,date_of_schedule) VALUES ('$patient_id','1','$newdate')");
        $saveinfo2 = mysqli_query($connection,"INSERT INTO `pedia_patient_inf
o`(`patient_id`, `doc_id`, `patient_name`, `patient_info`, `purpose`) VALUES ('$pati
ent_id','1','$patient_name','$patient_info','$purpose')");
        echo '<script>
        setTimeout(function() {
         swal({
           title: "Scheduled to Pediatrician",
           text: "QR Code Generated. PLease see Home tab for other details";
           type: "success"
         }, function() {
           window.location = "../contents/schedule.php";
         });
        }, 1000);
        </script>';
       }else{
        echo '<script>
        setTimeout(function() {
         swal({
           title: "Sorry, schedule is full for this day. Try another day.",
           text: "Tap OK to continue",
           type: "warning"
         }, function() {
           window.location = "../contents/schedule.php";
         });
        }, 1000);
        </script>';
       }}}
    if(isset($_POST['schedule_to_obgyne'])){
      $username = ($_SESSION['login_user']);
      $patient_id = $_REQUEST['patient_id'];
      $date = strtotime($_POST['date']);
      $fullname = $_POST['fullname'];
      $address = $_POST['address'];
      $contactnumber = $_POST['contactnumber'];
      $newdate = date('Y-m-d', $date);
      $compareday = date("l", $date);
      $flag=0;
```

```
$query ="SELECT * FROM doctor_availability WHERE doc_id = '2'";
$results = $connection->query($query);
while($rs=$results->fetch_assoc()){
 if($rs["day"]==$compareday){
  $flag++;
  break;
 }}if ($flag==0) {
 echo '<script>
 setTimeout(function() {
  swal({
   title: "Doctor unavailable on selected date, please choose another
          date",text: "Tap OK to continue",
   type: "error"
  }, function() {
   window.location = "../contents/schedule.php";
  });
 }, 1000);
 </script>';
}else {
 $patient_count1 = "SELECT date_of_schedule, COUNT(date_of_schedule)
FROM `doctors_schedule` GROUP BY date_of_schedule HAVING COUNT(date_
of_schedule) > 1;";
 $count = $connection->query($patient_count1);
 while($count1=$count->fetch_assoc()){
 $count1++;
 break;
 }if($count1<=30){
  $tempdir = '../library/qrimages/';
  $filename = $username;
  $codecontents = 'Fullname: '.$fullname.'\nAddress: '.$address.'\nContact N
umber: '.$contactnumber.'\nDate of Schedule: '.$newdate.' - Department: Ob-
Gyne - Code: stptrckplyclnc';
  $code = QRcode::png($codecontents, $tempdir."$filename.'.png', QR_EC
LEVEL_L, 5);
  $saveinfo1 = mysqli_query($connection,"INSERT INTO `doctors_schedule
`(patient_id,doc_id,date_of_schedule) VALUES ('$patient_id','2','$newdate')");
  echo '<script>
  setTimeout(function() {
   swal({
    title: "Scheduled to Ob-Gynecologist",
```

```
        text: "QR Code Generated. PLease see Home tab for other details. ",

        type: "success"
      }, function() {
        window.location = "../contents/schedule.php";
      });
    }, 1000);
    </script>';
  }else{
    echo '<script>
    setTimeout(function() {
      swal({
        title: "Sorry, schedule is full for this day. Try another day.",
        text: "Tap OK to continue",
        type: "warning"
      }, function() {
        window.location = "../contents/schedule.php";
      });
    }, 1000);
    </script>';
  }}}
if(isset($_POST['schedule_to_ortho'])){
  $username = ($_SESSION['login_user']);
  $patient_id = $_REQUEST['patient_id'];
  $date = strtotime($_POST['date']);
  $fullname = $_POST['fullname'];
  $address = $_POST['address'];
  $contactnumber = $_POST['contactnumber'];
  $newdate = date('Y-m-d', $date);
  $compareday = date("l", $date);
  $flag=0;
  $query ="SELECT * FROM doctor_availability WHERE doc_id = '3'";
  $results = $connection->query($query);
  while($rs=$results->fetch_assoc()){
    if($rs["day"]==$compareday){
      $flag++;
      break;
    }}if ($flag==0) {
    echo '<script>
    setTimeout(function() {
      swal({
```

```
        title: "Doctor unavailable on selected date, please choose another
                date",text: "Tap OK to continue",
        type: "error"
       }, function() {
        window.location = "../contents/schedule.php";
       });
      }, 1000);
      </script>';
     }else {
      $patient_count1 = "SELECT date_of_schedule, COUNT(date_of_schedule)
FROM `doctors_schedule` GROUP BY date_of_schedule HAVING COUNT(date_
of_schedule) > 1;";
      $count = $connection->query($patient_count1);
      while($count1=$count->fetch_assoc()){
      $count1++;
      break;
      }if($count1<=30){
       $tempdir = '../library/qrimages/';
       $filename = $username;
       $codecontents = 'Fullname: '.$fullname.'\nAddress: '.$address.'\nContact N
umber: '.$contactnumber.'\nDate of Schedule: '.$newdate.' - Department: Orthope
dics - Code: stptrckplyclnc';
       $code = QRcode::png($codecontents, $tempdir."$filename.'.png', QR_EC
LEVEL_L, 5);
       $saveinfo1 = mysqli_query($connection,"INSERT INTO `doctors_schedule
`(patient_id,doc_id,date_of_schedule) VALUES ('$patient_id','3','$newdate')");
       echo '<script>
       setTimeout(function() {
        swal({
         title: "Scheduled to Orthopedist",
         text: "QR Code Generated. PLease see Home tab for other details. ",
      type: "success"
        }, function() {
         window.location = "../contents/schedule.php";
        });
       }, 1000);
       </script>';
      }else{
       echo '<script>
       setTimeout(function() {
        swal({
```

```
            title: "Sorry, schedule is full for this day. Try another day."
            text: "Tap OK to continue",
            type: "warning"
          }, function() {
            window.location = "../contents/schedule.php";
          });
        }, 1000);
        </script>';
      }}}
if(isset($_POST['schedule_to_endo'])){
    $username = ($_SESSION['login_user']);
    $patient_id = $_REQUEST['patient_id'];
    $date = strtotime($_POST['date']);
    $fullname = $_POST['fullname'];
    $address = $_POST['address'];
    $contactnumber = $_POST['contactnumber'];
    $newdate = date('Y-m-d', $date);
    $compareday = date("l", $date);
    $flag=0;
    $query ="SELECT * FROM doctor_availability WHERE doc_id = '4'";
    $results = $connection->query($query);
    while($rs=$results->fetch_assoc()){
      if($rs["day"]==$compareday){
        $flag++;
        break;
      }}
    if ($flag==0) {
      echo '<script>
      setTimeout(function() {
        swal({
          title: "Doctor unavailable on selected date, please choose another
                                    date", text: "Tap OK to continue",
          type: "error"
        }, function() {
          window.location = "../contents/schedule.php";
        });
      }, 1000);
      </script>';
    }else {
```

```
        $patient_count1 = "SELECT date_of_schedule, COUNT(date_of_schedule)
FROM `doctors_schedule` GROUP BY date_of_schedule HAVING COUNT(date_
of_schedule) > 1;";
        $count = $connection->query($patient_count1);
        while($count1=$count->fetch_assoc()){
        $count1++;
        break;
        }if($count1<=30){
         $tempdir = '../library/qrimages/';
         $filename = $username;
         $codecontents = 'Fullname: '.$fullname.'\nAddress: '.$address.'\nContact N
umber: '.$contactnumber.'\nDate of Schedule: '.$newdate.' - Department: Endocrin
ologist - Code: stptrckplyclnc';
         $code = QRcode::png($codecontents, $tempdir."".$filename.'.png', QR_EC
LEVEL_L, 5);
         $saveinfo1 = mysqli_query($connection,"INSERT INTO `doctors_schedule
`(patient_id,doc_id,date_of_schedule) VALUES ('$patient_id','4','$newdate')");
         echo '<script>
          setTimeout(function() {
            swal({
              title: "Scheduled to Endocrinologist",
              text: "QR Code Generated. PLease see Home tab for other details. ",
         type: "success"
            }, function() {
              window.location = "../contents/schedule.php";
            });
         }, 1000);
         </script>';
        }else{
         echo '<script>
         setTimeout(function() {
          swal({
            title: "Sorry, schedule is full for this day. Try another day.",
            text: "Tap OK to continue",
            type: "warning"
          }, function() {
            window.location = "../contents/schedule.php";
          });
         }, 1000);
         </script>';
        }}}
```

```
if(isset($_POST['schedule_to_gensur'])){
  $username = ($_SESSION['login_user']);
  $patient_id = $_REQUEST['patient_id'];
  $date = strtotime($_POST['date']);
  $fullname = $_POST['fullname'];
  $address = $_POST['address'];
  $contactnumber = $_POST['contactnumber'];
  $newdate = date('Y-m-d', $date);
  $compareday = date("l", $date);
  $flag=0;
  $query ="SELECT * FROM doctor_availability WHERE doc_id = '5'";
  $results = $connection->query($query);
  while($rs=$results->fetch_assoc()){
   if($rs["day"]==$compareday){
    $flag++;
    break;
   }}
 if ($flag==0) {
    echo '<script>
    setTimeout(function() {
     swal({
      title: "Doctor unavailable on selected date, please choose another
            date", text: "Tap OK to continue",
      type: "error"
     }, function() {
      window.location = "../contents/schedule.php";
     });
    }, 1000);
    </script>';
   }else {
    $patient_count1 = "SELECT date_of_schedule, COUNT(date_of_schedule)
FROM `doctors_schedule` GROUP BY date_of_schedule HAVING COUNT(date_
of_schedule) > 1;";
    $count = $connection->query($patient_count1);
    while($count1=$count->fetch_assoc()){
    $count1++;
    break;
    }if($count1<=30){
     $tempdir = '../library/qrimages/';
     $filename = $username;
```

```
        $codecontents = 'Fullname: '.$fullname.'\nAddress: '.$address.'\nContact N
umber: '.$contactnumber.'\nDate of Schedule: '.$newdate.' - Department: General
Surgery - Code: stptrckplyclnc';
        $code = QRcode::png($codecontents, $tempdir."$filename.'.png', QR_EC
LEVEL_L, 5);
        $saveinfo1 = mysqli_query($connection,"INSERT INTO `doctors_schedule
`(patient_id,doc_id,date_of_schedule) VALUES ('$patient_id','5','$newdate')");
        echo '<script>
        setTimeout(function() {
         swal({
           title: "Scheduled to General Surgeon",
           text: "QR Code Generated. PLease see Home tab for other details. ",
      type: "success"
          }, function() {
           window.location = "../contents/schedule.php";
         });
        }, 1000);
        </script>';
       }else{
        echo '<script>
        setTimeout(function() {
         swal({
           title: "Sorry, schedule is full for this day. Try another day."
           text: "Tap OK to continue",
           type: "warning"
         }, function() {
           window.location = "../contents/schedule.php";
         });
        }, 1000);
        </script>';
       }}}
  if(isset($_POST['schedule_to_radio'])){
   $username = ($_SESSION['login_user']);
   $patient_id = $_REQUEST['patient_id'];
   $date = strtotime($_POST['date']);
   $fullname = $_POST['fullname'];
   $address = $_POST['address'];
   $contactnumber = $_POST['contactnumber'];
   $newdate = date('Y-m-d', $date);
   $compareday = date("l", $date);
   $flag=0;
```

```
$query ="SELECT * FROM doctor_availability WHERE doc_id = '6'";
$results = $connection->query($query);
while($rs=$results->fetch_assoc()){
 if($rs["day"]==$compareday){
  $flag++;
  break;
 }}
if ($flag==0) {
 echo '<script>
 setTimeout(function() {
  swal({
   title: "Doctor unavailable on selected date, please choose another
          date", text: "Tap OK to continue",
   type: "error"
  }, function() {
   window.location = "../contents/schedule.php";
  });
 }, 1000);
 </script>';
}else {
 $patient_count1 = "SELECT date_of_schedule, COUNT(date_of_schedule)
FROM `doctors_schedule` GROUP BY date_of_schedule HAVING COUNT(date_
of_schedule) > 1;";
 $count = $connection->query($patient_count1);
 while($count1=$count->fetch_assoc()){
 $count1++;
 break;
 }if($count1<=30){
  $tempdir = '../library/qrimages/';
  $filename = $username;
  $codecontents = 'Fullname: '.$fullname.'\nAddress: '.$address.'\nContact N
umber: '.$contactnumber.'\nDate of Schedule: '.$newdate.' - Department: Radiolog
y - Code: stptrckplyclnc';
  $code = QRcode::png($codecontents, $tempdir."".$filename.'.png', QR_EC
LEVEL_L, 5);
  $saveinfo1 = mysqli_query($connection,"INSERT INTO `doctors_schedule
`(patient_id,doc_id,date_of_schedule) VALUES ('$patient_id','6','$newdate')");
  echo '<script>
  setTimeout(function() {
   swal({
    title: "Scheduled to Radiologist",
```

```
                text: "QR Code Generated. PLease see Home tab to see other details
 .",            type: "success"
               }, function() {
                 window.location = "../contents/schedule.php";
               });
             }, 1000);
             </script>';
            }else{
             echo '<script>
             setTimeout(function() {
              swal({
               title: "Sorry, schedule is full for this day. Try another day.",
               text: "Tap OK to continue",
               type: "warning"
              }, function() {
               window.location = "../contents/schedule.php";
              });
             }, 1000);
             </script>';
            }}}?>
    <?php include '../includes/scripts.php'; ?>
  </body>
</html>
```

Verify_patient.php

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
  <link href="../custom-styles/style.css" rel="stylesheet">
 </head>
 <body>
  <?php
   require("../database/dbconnect.php");
   require("../phpmailer/PHPMailerAutoload.php");
```

```
    if(isset($_POST['submit'])){
      $fullname =$_POST['fullname'];
      $contactnumber =$_POST['contactnumber'];
      $address =$_POST['address'];
      $username =$_POST['username'];
      $password =$_POST['password'];
      $to =$_POST['email'];
      $mail = new PHPMailer;
      $mail->isSMTP();
          $mail->Host = 'smtp.gmail.com';
      $mail->SMTPAuth = true;
      $mail->Username = 'saintpatrickpolyclinic@gmail.com';
      $mail->Password = 'ujympwspcyoykdwn';
      $mail->SMTPSecure = 'tls';
      $mail->Port = 587;
      $mail-
>setFrom('saintpatrickpolyclinic@gmail.com', "St. Patrick's Polyclinic team");
      $mail->addAddress($to, 'Email User');
      $mail->addReplyTo('saintpatrickpolyclinic@gmail.com');
      $mail->isHTML(true);
      $mail-
>Subject = "Verification for registering of St. Patrick Polyclinic's patients.";
      $mail->Body    = 'Your code is: <b>sppormoc</b>';
      if($mail->send()) {
       $msg = "A verification code was successfully sent to your email: " . $to;}else
{
       $msg = "Verification code not sent.";
      }}?>
   <div class="container mt-5 text-center">
    <div class="alert alert-success alert-dismissible fade show mb-5" role="alert">
     <p><?php echo $msg ?></p>
     <button type="button" class="close" data-dismiss="alert" aria-label="Close">
      <span aria-hidden="true">&times;</span>
     </button>
    </div>
    <div>
     <form action="register_patient.php" method="POST" enctype="multipart/form
-data">
       <div class="form-group mb-4">
<input type="hidden"value="<?php echo $fullname?>" name="fullname"
```

```
<input type="hidden" value="<?php echo $contactnumber?>" name="contactnumb
er">
<input type="hidden" value="<?php echo $address?>" name="address">
<input type="hidden" value="<?php echo $username?>" name="username">
<input type="hidden" value="<?php echo $password?>" name="password">
<input type="hidden" value="<?php echo $to?>" name="email" required="require"
>
          <div class="mb-3 ml-4">
            <h6>Choose a profile picture for your account</h6>
          </div>
          <div class="mb-5" style="margin-left: 8rem;">
          <inputtype="file"name="picture"class="input-font" required="require">
          </div>
          <hr class="mb-5">
          <div class="mb-3 ml-4">
            <h6>Verification code for your account</h6>
          </div>
          <div class="mb-5 col-4 offset-4">
            <input type="text" name="code" class="form-
control" placeholder="Enter verification code" required="require">
          </div>
        </div>
        <div class="text-right mt-4 mb-3">
          <input type="submit" name="submit" class="btn btn-
success shadow" value="Register">
        </div>
      </form>
    </div>
  </div>
    <?php include '../includes/scripts.php'; ?>
 </body>
</html>
```

Source Code (Doctor)

Doctor_schedule.php

```
<?php
 include('../session.php');
 if(!isset($_SESSION['login_user'])){
  header("location: ../index.php");
 }
?>
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Doctor's Schedule</title>
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
 </head>
 <body>
 <?php
  require("../database/dbconnect.php");
  if(isset($_POST['submit'])){
   $doc_id = $_REQUEST['doc_id'];
   $date = $_POST['date'];
   $day = date("l", strtotime($date));
   $arrival_time = $_POST['arrival_time'];
   $depart_time = $_POST['depart_time'];
   $doc_sched = mysqli_query($connection, "INSERT INTO `doctor_availability`(
`doc_id`, `date`, `day`, `start_time`, `end_time`) VALUES ('$doc_id','$date','$day','
$arrival_time','$depart_time')");
   if(!$doc_sched){
```

```
      echo '<script>
      setTimeout(function() {
       swal({
        title: "Incorrect Query.",
        text: "Tap OK to continue",
        type: "error"
       }, function() {
        window.location = "../contents/schedule.php";
       });
      }, 1000);
      </script>';
     }else {
      echo '<script>
      setTimeout(function() {
       swal({
        title: "You have set a day in your schedule in a week.",
        text: "Tap OK to continue",
        type: "success"
       }, function() {
        window.location = "../contents/schedule.php";
       });
      }, 1000);
      </script>';
     }}
 ?>
  <?php include '../includes/scripts.php'; ?>
 </body>
</html>
```

Delete_notification.php

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
```

```
  <link rel="stylesheet" href="../style/sweetalert.css">
  <link href="../custom-styles/style.css" rel="stylesheet">
 </head>
 <body>
  <?php
   require("../database/dbconnect.php");
   $notif_id = $_REQUEST['notif_id'];
   $delete_notif = mysqli_query($connection,"DELETE from `doctor_notification`
where notif_id ='$notif_id'");
   if(!$delete_notif){
    echo '<script>
    setTimeout(function() {
      swal({
        title: "Opps!",
        text: "Notification not deleted successfully. Please try again.",
        type: "error"
      }, function() {
        window.location = "../contents/notification.php";
      });
    }, 1000);
   </script>';
  }else{
   echo '<script>
   setTimeout(function() {
     swal({
       title: "Deleted.",
       text: "Notification deleted successfully.",
       type: "success"
     }, function() {
       window.location = "../contents/notification.php";
     });
   }, 1000);
  </script>';
  }
  ?>
  <?php include '../includes/scripts.php'; ?>
 </body>
<html>
```

Done_schedule.php

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
  <link href="../custom-styles/style.css" rel="stylesheet">
 </head>
 <body>
  <?php
   require("../database/dbconnect.php");
   $ds_id = $_REQUEST['ds_id'];
   $patient_id = $_REQUEST['patient_id'];
   $doc_id = $_REQUEST['doc_id'];
   $date_of_schedule = $_REQUEST['date_of_schedule'];
   $patient_notif_message = "Your schedule for " . date("F j, Y", strtotime($date_
of_schedule)) . " is already done.";
   $doc_notif_message = "Your schedule for " . date("F j, Y", strtotime($date_of_
schedule)) . " is already done.";
   $done_sched = mysqli_query($connection,"DELETE from `doctors_schedule`
where ds_id ='$ds_id'");
   $patient_name = mysqli_query($connection,"SELECT fullname FROM `patient
` WHERE patient_id = $patient_id");
   while ($pname = mysqli_fetch_array($patient_name)){
    $fullname = $pname['fullname'];
}
   $delete_sched_from_scan_qr = mysqli_query($connection,"DELETE FROM `a
ssistant_nurse` WHERE qr_information LIKE '%$fullname%'");
   $insert_patient_notification = mysqli_query($connection,"INSERT INTO `patie
nt_notification`(`patient_id`, `doc_id`, `notification_message`) VALUES ('$patient_i
d','$doc_id','$patient_notif_message')");
   $insert_doctor_notification = mysqli_query($connection,"INSERT INTO `doctor
_notification`(`patient_id`, `doc_id`, `notification_message`, `timestamp`) VALUES
('$patient_id','$doc_id','$doc_notif_message','$date_of_schedule')");
   if(!$done_sched && !$insert_patient_notification && !$insert_doctor_notificatio
n && !$delete_sched_from_scan_qr){
    echo '<script>
```

```
        setTimeout(function() {
          swal({
            title: "Oops!",
            text: "Your schedule not done successfully. Please try again.",
            type: "error"
          }, function() {
            window.location = "../contents/schedule.php";
          });
        }, 1000);
      </script>';
    }else{
      echo '<script>
      setTimeout(function() {
        swal({
          title: "Done schedule.",
          text: "Your schedule for this day is done.",
          type: "success"
        }, function() {
          window.location = "../contents/schedule.php";
        });
      }, 1000);
    </script>';
    }?>
    <?php include '../includes/scripts.php'; ?>
  </body>
</html>
```

Message_admin.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
    <link rel="stylesheet" href="../bootstrap/font-awesome/css/font-
awesome.min.css">
    <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
    <link rel="stylesheet" href="../style/bootstrap.min.css">
    <link rel="stylesheet" href="../style/sweetalert.css">
    <link href="../custom-styles/style.css" rel="stylesheet">
```

```
</head>
<body>
<?php
  require("../database/dbconnect.php");
   if(isset($_POST['send_message_to_admin'])){
    $doc_id = $_REQUEST['doc_id'];
    $message_admin = $_POST['message_admin'];
    $saveinfo= mysqli_query($connection,"INSERT INTO `doctor_admin_messa
ge`(doc_id,admin_id,message)values ('$doc_id','1','$message_admin')");
    echo '<script>
     setTimeout(function() {
        swal({
           title: "Yehey!",
           text: "Your message to Admin has been sent!",
           type: "success"
        }, function() {
           window.location = "../contents/message.php";
        });
     }, 1000);
    </script>';
}?>
   <?php include '../includes/scripts.php'; ?>
 </body>
</html>
```

Update_doctor.php

```
<?php
 require ("../database/dbconnect.php");
 $doc_id =$_REQUEST['doc_id'];

 $firstname =$_POST['firstname'];
 $lastname =$_POST['lastname'];
 $department =$_POST['department'];
 $mobilenumber =$_POST['mobilenumber'];
 $username =$_POST['username'];
 $password=$_POST['password'];
 $picture = $_FILES['picture']['name'];
 $filetmpname = $_FILES['picture']['tmp_name'];
 move_uploaded_file($filetmpname,"../../admin/images/$picture");
```

```
 $updateData = mysqli_query($connection,"UPDATE `doctors` set firstname='$fir
stname',lastname='$lastname',department='$department',
                    mobilenumber='$mobilenumber',username='$username',passwor
d='$password',picture='$picture' where doc_id='$doc_id'");
 if(!$updateData){
   echo"<script>alert('Incorrect Query!');window.location.href='../index.php';</script
>";
 }else{
   echo"<script>alert('Your profile information has been updated!');window.location
.href='../index.php';</script>";
 }?>
```

Source Code (Admin)

Delete_message.php

```
<!DOCTYPE html>
<html lang="en">
 <head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Delete Messages</title>
   <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
   <link rel="stylesheet" href="../style/bootstrap.min.css">
   <link rel="stylesheet" href="../style/sweetalert.css">
 </head>
 <body>
   <?php
    require ("../database/dbconnect.php");
    $da_id =$_REQUEST['da_id'];
    $deleteMessage = mysqli_query($connection,"DELETE from `doctor_admin_
message` where da_id='$da_id'");
    echo '<script>
      setTimeout(function() {
        swal({
          title: "Deleted",
          text: "Message has been successfully deleted.",
          type: "success"
        }, function() {
          window.location = "../contents/messages.php";
```

```
        });
      }, 1000);
     </script>';
   ?>
   <script src="../style/sweetalert.min.js"></script>
  </body>
</html>
```

Delete_notification.php

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Delete Messages</title>
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
 </head>
 <body>
  <?php
   require ("../database/dbconnect.php");
   $notif_id =$_REQUEST['notif_id'];
   $deleteMessage = mysqli_query($connection,"DELETE from `admin_notificati
on` where notif_id='$notif_id'");
   echo '<script>
    setTimeout(function() {
     swal({
      title: "Deleted",
      text: "Notification has been successfully deleted.",
      type: "success"
     }, function() {
      window.location = "../contents/notifications.php";
     });
    }, 1000);
   </script>';
  ?>

  <script src="../style/sweetalert.min.js"></script>
</body>
```

```
</html>
```

Savedoc.php

```
<!DOCTYPE html>
 <html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Add a Doctor</title>
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
 </head>
 <body>
  <?php
   require("../database/dbconnect.php");
   $username =$_POST['username'];
   $password =md5($_POST['password']);
   $firstname =$_POST['firstname'];
   $lastname =$_POST['lastname'];
   $mobile_number =$_POST['mobile_number'];
   $department =$_POST['department'];
   $picture = $_FILES['picture']['name'];
   $filetmpname = $_FILES['picture']['tmp_name'];
   move_uploaded_file($filetmpname,"../images/$picture");
   $saveinfo= mysqli_query($connection,"INSERT INTO `doctors`(`username`, `p
assword`, `firstname`, `lastname`, `department`, `mobilenumber`, `picture`)
                    values ('$username', '$password', '$firstname', '$lastname', '$de
partment', '$mobile_number', '$picture')");
    echo '<script>
    setTimeout(function() {
     swal({
      title: "Added",
      text: "You have successfully added a doctor.",
      type: "success"
     }, function() {
      window.location = "../contents/doctors.php";
     });
    }, 1000);
    </script>';
  ?>
```

```
    <script src="../style/sweetalert.min.js"></script>
  </body>
</html>
```

Update_admin.php

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Update Profile</title>
  <link rel="stylesheet" href="../bootstrap/css/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
 </head>
 <body>
  <?php
   require ("../database/dbconnect.php");
   $admin_id =$_REQUEST['admin_id'];

   $username =$_POST['username'];
   $password =$_POST['password'];
   $firstname =$_POST['firstname'];
   $lastname =$_POST['lastname'];
   $picture = $_FILES['picture']['name'];
   $filetmpname = $_FILES['picture']['tmp_name'];
   move_uploaded_file($filetmpname,"../images/$picture");
   $updateData = mysqli_query($connection,"UPDATE `admin` set username='$
username',password='$password',picture='$picture',firstname='$firstname',lastna
me='$lastname' where admin_id='$admin_id'");
   echo '<script>
   setTimeout(function() {
    swal({
     title: "Updated",
     text: "Your profile has been successfully updated.",
     type: "success"
    }, function() {
     window.location = "../contents/homepage.php";
    });
   }, 1000);
  </script>';
```

```
  ?>
  <script src="../style/sweetalert.min.js"></script>
 </body>
</html>
```

Source Code (Assistant Nurse)

Insert_info_qrcode.php

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="../style/bootstrap.min.css">
  <link rel="stylesheet" href="../style/sweetalert.css">
  <title>Save Information</title>
 </head>
 <body>
  <?php
   require("../database/dbconnect.php");
   if (isset($_POST['text'])) {
    $text = $_POST['text'];
    $code = "stptrckplyclnc";
    if(strpos($text, $code) !== false){
     $save_info_qrcode = mysqli_query($connection,"INSERT INTO `assistant_
nurse`(qr_information, timestamp) VALUES('$text',NOW())");
     if(!$save_info_qrcode){
      echo '<script>
       setTimeout(function() {
        swal({
```

```
        title: "Error!",
        text: "Cannot read QR Code information.",
        type: "error"
       }, function() {
        window.location = "../contents/scan_qr.php";
       });
      }, 1000);
     </script>';
    }else {
     echo '<script>
      setTimeout(function() {
        swal({
          title: "Scanned Successfully!",
          text: "QR Code information read successfully.",
          type: "success"
        }, function() {
          window.location = "../contents/scan_qr.php";
        });
      }, 1000);
     </script>';
    }}
   else{
    echo '<script>
     setTimeout(function() {
      swal({
       title: "Warning!",
       text: "QR Code is not valid for this system.",
       type: "warning"
      }, function() {
       window.location = "../contents/scan_qr.php";
      });
     }, 1000);
    </script>';
   }}?>
  <script src="../style/sweetalert.min.js"></script>
 </body>
</html>
```

Scan_qr.php

```
<html>
```

```
<head>
 <meta name="viewport" content="width=device-width; initial-
scale=1.0; maximum-scale=1.0; user-scalable=0;"/>
 <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/webrtc-
adapter/3.3.3/adapter.min.js"></script>
 <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.1.
10/vue.min.js"></script>
 <link rel="stylesheet" href="../style/bootstrap.min.css">
 <link rel="icon" type="image/png" href="../images/logo.png">
 <script type="text/javascript" src="../style/instascan.min.js"></script>
 <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/instasca
n/2.0.0-rc.4/instascan.min.js"></script>
 <title>Scan QR Code</title>
</head>
<body>
 <div class="container">
  <div class="row mt-5">
   <div class="col-10 offset-1 col-sm-10 offset-sm-1 col-md-8 offset-md-2 col-lg-
6 offset-lg-3 ">
    <h4 class="mt-5 mb-3 text-center">Scan for QR Code</h4>
    <video id="preview" width="100%"></video>

    <form action="../process/insert_info_qrcode.php" method="POST">
     <input type="hidden" name="text" id="text" readonyy="" class="form-
control">
    </form>
   </div>
  </div>
  <div class="row">
   <div class="col-10 offset-1 col-sm-10 offset-sm-1 col-md-8 offset-md-2 col-lg-
6 offset-lg-3 mt-5 text-center">
    <a href="view_patients.php" class="btn btn-
success">Click to view patient schedules</a>
   </div>
  </div>
 </div>
 <script>
  let scanner = new Instascan.Scanner({ video: document.getElementById('previ
ew')});
  Instascan.Camera.getCameras().then(function(cameras){
   if(cameras.length > 0 ){
```

```
            scanner.start(cameras[0]);
        } else{
            alert('Error. No cameras found');
        }
    }).catch(function(e) {
        console.error(e);
    });
    scanner.addListener('scan',function(c){
        document.getElementById('text').value=c;
        document.forms[0].submit();
    });
  </script>
 </body>
</html>
```

View_patients.php

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="../style/bootstrap.min.css">
  <link rel="icon" type="image/png" href="../images/logo.png">
  <title>View Patients</title>
 </head>
 <body>
  <div class="container">
   <table class="table">
    <thead>
     <tr>
      <td class="text-center">Patient Information</td>
     </tr>
    </thead>
    <tbody>
     <?php
      require("../database/dbconnect.php");
      $show_patients = mysqli_query($connection,"SELECT * FROM `assistant
_nurse` ORDER BY timestamp DESC");
```

```
            if(mysqli_num_rows($show_patients) > 0) {
             while($show_patient = mysqli_fetch_array($show_patients))
             {?>
             <tr>
              <td class="text-
 center"><?php echo $show_patient['qr_information'];?></td>
             <?php }?>
            <?php
            }else{?>
             <td class="text-center">No patients yet.</td>
             </tr>
            <?php }?>
       </tbody>
      </table>
      <div class="text-center">
       <a href="scan_qr.php" class="btn btn-
success">Back to QR Code scanner</a>
      </div>
    </div>
  </body>
</html>
```

Source Code (LED Queue Display)

Index.php

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <meta ht tp-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="style/bootstrap.min.css">
  <link rel="stylesheet" href="style/sweetalert.css">
  <title>Display Patients Queue</title>
  <link rel="icon" type="image/png" href="images/logo.png">
 </head>
 <body>
{
  <div class="container mt-5">
   <h4 class="text-center">Patients Queue on every Doctor</h4>
```

```
<div class="card-columns" style="margin-top: 50px;">
 <div class="card text-center shadow">
  <div class="card-body">
   <h5 class="card-title text-success">Pediatrician</h5>
   <?php
    require("database/dbconnect.php");
    $display = mysqli_query($connection, "SELECT * FROM doctors WHERE
doc_id = '1'");
    while ($display_doctor = mysqli_fetch_array($display))
    {?>
     <p class="card-text"><small class="text-
muted">Dr. <?php echo $display_doctor['firstname'];?> <?php echo $display_doct
or['lastname'];?></small></p>
    <?php }?>
   <table class="table">
    <thead>
     <tr>
      <th class="border border-bottom-0 border-right-0 border-left-
0" scope="col">Patient's Full Name</th>
     </tr>
    </thead>
    <tbody>
     <?php
      require("database/dbconnect.php");
      $display1 = mysqli_query($connection, "SELECT * FROM `assistant_n
urse` WHERE qr_information LIKE '%Pediatrics%' ORDER BY timestamp DESC L
IMIT 1");
      while ($display_patient = mysqli_fetch_array($display1))
      {
      ?>
       <tr>
        <td><?php echo substr($display_patient['qr_information'], 10, -
138);?></td>
       </tr>
      <?php }?>
    </tbody>
   </table>
  </div>
 </div>
 <div class="card text-center shadow">
  <div class="card-body">
```

```
            <h5 class="card-title text-success">Obstetrician Gynecologist</h5>
            <?php
             require("database/dbconnect.php");
             $display = mysqli_query($connection, "SELECT * FROM doctors WHERE
  doc_id = '2'");
             while ($display_doctor = mysqli_fetch_array($display))
             {?>
              <p class="card-text"><small class="text-
 muted">Dr. <?php echo $display_doctor['firstname'];?> <?php echo $display_doct
 or['lastname'];?></small></p>
             <?php }?>
            <table class="table">
             <thead>
              <tr>
               <th class="border border-bottom-0 border-right-0 border-left-
 0" scope="col">Patient's Full Name</th>
              </tr>
             </thead>
             <tbody>
              <?php
               require("database/dbconnect.php");
               $display1 = mysqli_query($connection, "SELECT * FROM `assistant_n
 urse` WHERE qr_information LIKE '%Ob-
 Gyne%' ORDER BY timestamp DESC LIMIT 1");
               while ($display_patient = mysqli_fetch_array($display1))
               {
               ?>
                <tr>
                 <td><?php echo substr($display_patient['qr_information'], 10, -
 138);?></td>
                </tr>
               <?php }?>
             </tbody>
            </table>
           </div>
          </div>
          <div class="card text-center shadow">
           <div class="card-body">
            <h5 class="card-title text-success">Orthopedist</h5>
            <?php
             require("database/dbconnect.php");
```

```
                $display = mysqli_query($connection, "SELECT * FROM doctors WHERE
        doc_id = '3'");
                while ($display_doctor = mysqli_fetch_array($display))
                {?>
                 <p class="card-text"><small class="text-
      muted">Dr. <?php echo $display_doctor['firstname'];?> <?php echo $display_doct
      or['lastname'];?></small></p>
              <?php }?>
             <table class="table">
              <thead>
               <tr>
                <th class="border border-bottom-0 border-right-0 border-left-
    0" scope="col">Patient's Full Name</th>
               </tr>
              </thead>
              <tbody>
               <?php
                require("database/dbconnect.php");
                $display1 = mysqli_query($connection, "SELECT * FROM `assistant_n
     urse` WHERE qr_information LIKE '%Orthopedics%' ORDER BY timestamp DES
     C LIMIT 1");
                while ($display_patient = mysqli_fetch_array($display1))
                {
                ?>
                 <tr>
                  <td><?php echo substr($display_patient['qr_information'], 10, -
    138);?></td>
                 </tr>
               <?php }?>
             </tbody>
            </table>
           </div>
          </div>
          <div class="card text-center shadow">
           <div class="card-body">
            <h5 class="card-title text-success">Endocrinologist</h5>
            <?php
             require("database/dbconnect.php");
             $display = mysqli_query($connection, "SELECT * FROM doctors WHERE
    doc_id = '4'");
            while ($display_doctor = mysqli_fetch_array($display))
```

```
            {?>
              <p class="card-text"><small class="text-
 muted">Dr. <?php echo $display_doctor['firstname'];?> <?php echo $display_doct
 or['lastname'];?></small></p>
            <?php }?>
           <table class="table">
            <thead>
             <tr>
              <th class="border border-bottom-0 border-right-0 border-left-
 0" scope="col">Patient's Full Name</th>
             </tr>
            </thead>
            <tbody>
             <?php
              require("database/dbconnect.php");
              $display1 = mysqli_query($connection, "SELECT * FROM `assistant_n
 urse` WHERE qr_information LIKE '%Endocrinologist%' ORDER BY timestamp D
 ESC LIMIT 1");
              while ($display_patient = mysqli_fetch_array($display1))
              {
              ?>
               <tr>
                <td><?php echo substr($display_patient['qr_information'], 10, -
 138);?></td>
               </tr>
              <?php }?>
           </tbody>
          </table>
         </div>
        </div>
        <div class="card text-center shadow">
         <div class="card-body">
          <h5 class="card-title text-success">General Surgeon</h5>
          <?php
           require("database/dbconnect.php");
           $display = mysqli_query($connection, "SELECT * FROM doctors WHERE
 doc_id = '5'");
           while ($display_doctor = mysqli_fetch_array($display))
           {?>
```

```
            <p class="card-text"><small class="text-
  muted">Dr. <?php echo $display_doctor['firstname'];?> <?php echo $display_doct
  or['lastname'];?></small></p>
            <?php }?>
           <table class="table">
            <thead>
             <tr>
              <th class="border border-bottom-0 border-right-0 border-left-
  0" scope="col">Patient's Full Name</th>
             </tr>
            </thead>
            <tbody>
             <?php
              require("database/dbconnect.php");
              $display1 = mysqli_query($connection, "SELECT * FROM `assistant_n
  urse` WHERE qr_information LIKE '%General Surgery%' ORDER BY timestamp D
  ESC LIMIT 1");
              while ($display_patient = mysqli_fetch_array($display1))
              {
              ?>
               <tr>
                <td><?php echo substr($display_patient['qr_information'], 10, -
  138);?></td>
               </tr>
              <?php }?>
            </tbody>
           </table>
          </div>
         </div>
         <div class="card text-center shadow">
          <div class="card-body">
           <h5 class="card-title text-success">Radiologist</h5>
           <?php
            require("database/dbconnect.php");
            $display = mysqli_query($connection, "SELECT * FROM doctors WHERE
  doc_id = '6'");
           while ($display_doctor = mysqli_fetch_array($display))
           {?>
            <p class="card-text"><small class="text-
  muted">Dr. <?php echo $display_doctor['firstname'];?> <?php echo $display_doct
  or['lastname'];?></small></p>
```

```
          <?php }?>
         <table class="table">
          <thead>
           <tr>
            <th class="border border-bottom-0 border-right-0 border-left-
0" scope="col">Patient's Full Name</th>
           </tr>
          </thead>
          <tbody>
           <?php
            require("database/dbconnect.php");
            $display1 = mysqli_query($connection, "SELECT * FROM `assistant_n
urse` WHERE qr_information LIKE '%Radiology%' ORDER BY timestamp DESC L
IMIT 1");
            while ($display_patient = mysqli_fetch_array($display1))
            {
            ?>
             <tr>
              <td><?php echo substr($display_patient['qr_information'], 10, -
138);?></td>
             </tr>
            <?php }?>
          </tbody>
         </table>
        </div>
       </div>
      </div>
     </div>
    </body>
   </html>
```

Dbconnect.php

```
<?php
$connection = mysqli_connect("localhost","root","");
if(!$connection)
{
  die("Database connection failed!");
}
$db_connect = mysqli_select_db($connection,"capstone_project" );
if(!$db_connect)
{
  die("Database connection failed!".mysqli_error());
}
?>
```

APPENDIX B: EVALUATION TOOL

This is a sample result of our evaluation for the ST. Patrick Polyclinic Scheduling and Queuing Management System.

Graphical User Interface Evaluation						
No.	Question	Rating from Tester				
		1	2	3	4	5
1.	Easy to Access Information?	/				
2.	Appropriate Font?		/			
3.	Appropriate Color?	/				

Human Computer Interaction Evaluation						
No.	Question	Rating from Tester				
		1	2	3	4	5
1.	System content easy to understand?	/				
2.	Is the validation Appropriate?	/				
3.	Does the error message help?	/				
4.	Is the system interruption helpful?	/				
5.	Appropriate use of textbox, drop box and etc.	/				

Overall Functionality Evaluation						
No.	Question	Rating from Tester				
		1	2	3	4	5
1.	Login	/				
2.	Registration	/				
3.	Sending of Verification Code to email	/				
4.	Update Personal Information	/				
5.	Set a new Appointment Schedule	/				
6.	Get new Appointment Schedule	/				
7.	Updating Appointment	/				
8.	Cancel Appointment	/				
9.	Generate QR Code	/				
10.	View My Appointment		/			
11.	List of Scheduled Appointments	/				
12.	View Doctor Record	/				
13.	View Appointments/Patients Report	/				
14.	Sending of Notifications	/				
15.	Logout	/				

This table shows the overall evaluation result and its interpretation.

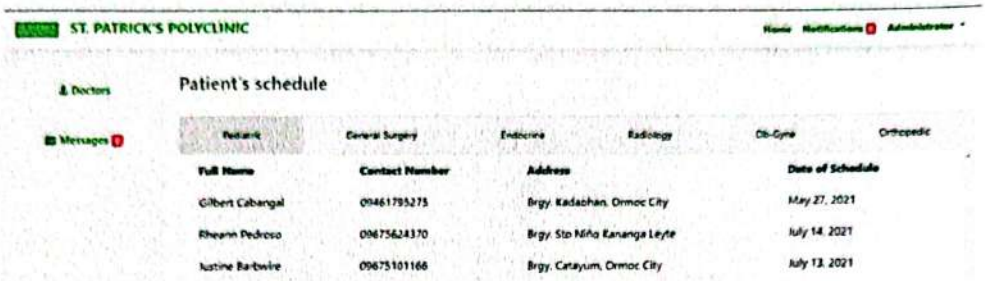
EVALUATION	5 (E)	4 (S)	3 (G)	2 (F)	1 (UNDECIDED)	TOTAL	INTERPRETATION
Graphical User Interface Evaluation	(8) 6	(5) 4				(13) 10	(E)
Human Computer Interaction Evaluation	(10) 7	(4) 4				(14) 11	(E)
Overall Functionality Evaluation	(3) 3	(8) 7				(11) 10	(E)

Legend:

10 -5	Excellent- (E)
4 - 3.5	Satisfied- (S)
3 - 2.5	Good - (G)
2 - 1.5	Fair - (F)
1- 0	Undecided – (U)

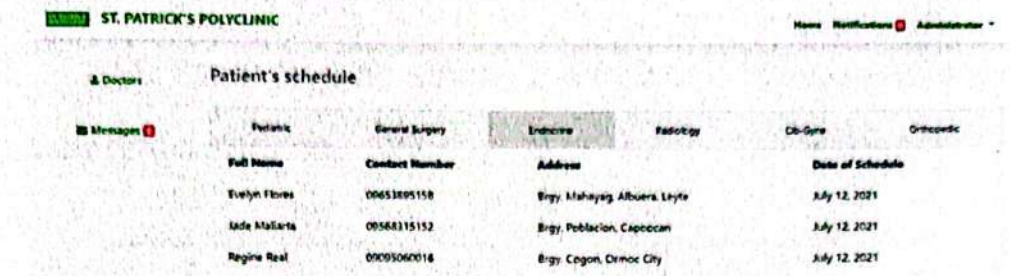
APPENDIX C: SAMPLE REPORTS (EXISTING SYSTEM)

ST. Patrick Polyclinic Scheduling and Queuing Management System sample reports.

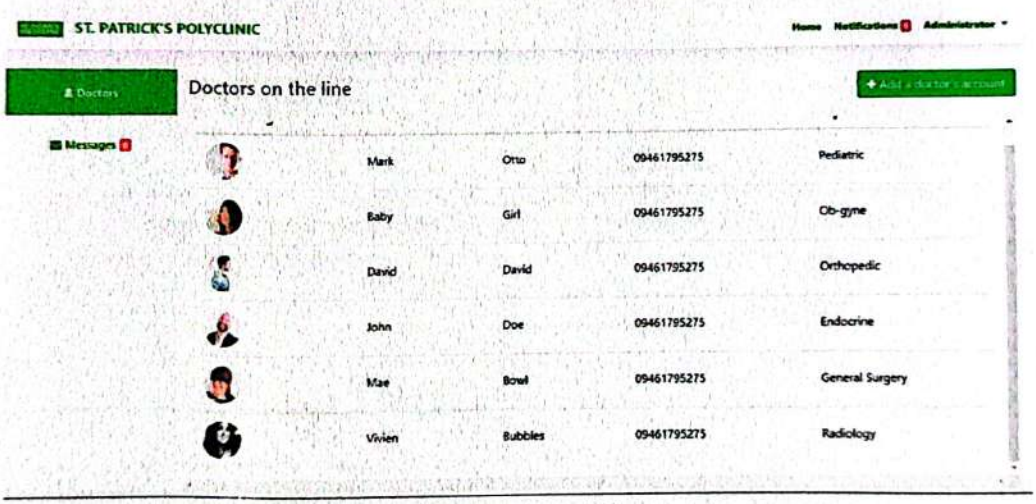


Sample Reports for Patients Appointments in Pedia

Sample Reports for Patients Appointments in Orthopedic



Sample Reports for Patients Appointments in Endocrine

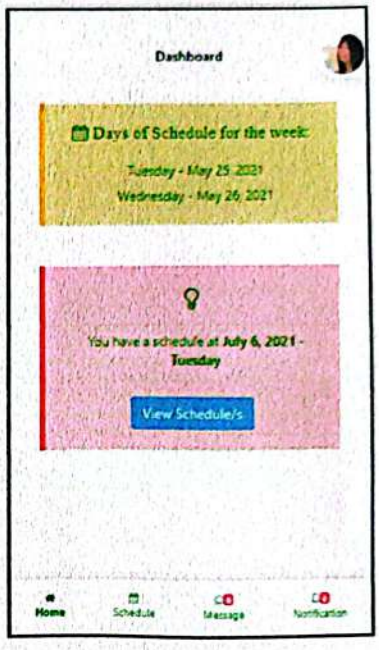


Sample Reports for the Active and In Active Doctors

APPENDIX D: SYSTEM GENERATED OUTPUTS

ST. Patrick Polyclinic Scheduling and Queuing Management System sample generated outputs.

Unique QR Code for every Patient Appointments



Doctors Lists of Schedules

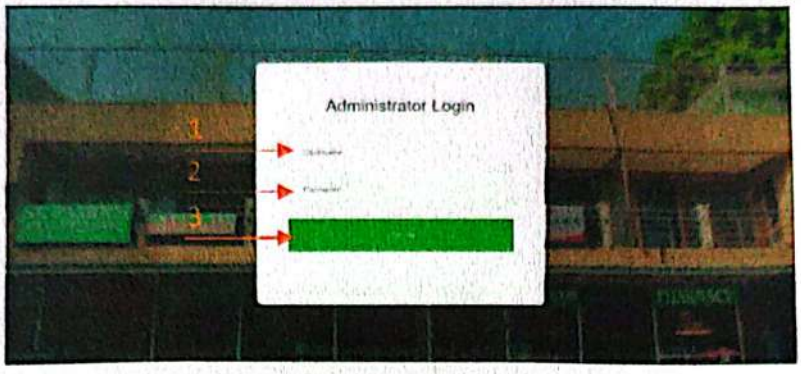
Scanned QR Codes: Patient Appointments Detail

Patients Queue on every Doctor

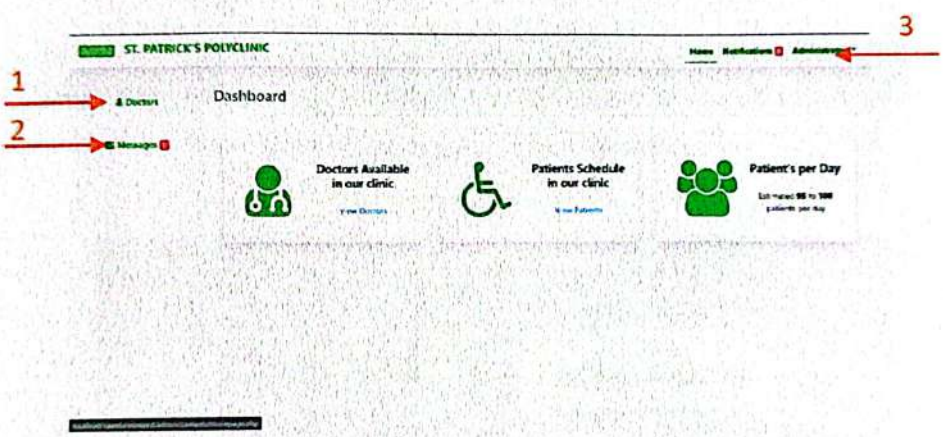
Pediatrician	Orthopedist	General Surgeon
Dr. Mark Oro	Dr. David David	Dr. Alla Bold
Patient's Full Name	**Patient's Full Name**	**Patient's Full Name**
Obstetrician Gynecologist	**Endocrinologist**	**Radiologist**
Dr. Baby Girl	Dr. John Doe	Dr. Vivien Bubbles
Patient's Full Name	**Patient's Full Name**	**Patient's Full Name**
Gilbert Cabangal		

Queuing Display

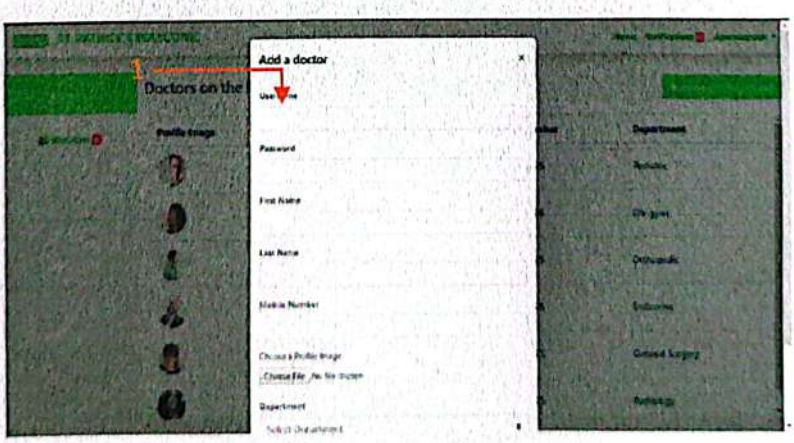
APPENDIX E: USER'S GUIDE/MANUAL



Fill out the (1) Username and (2) Password using the Administrator Log in Credentials, then click (3) LOG IN button.



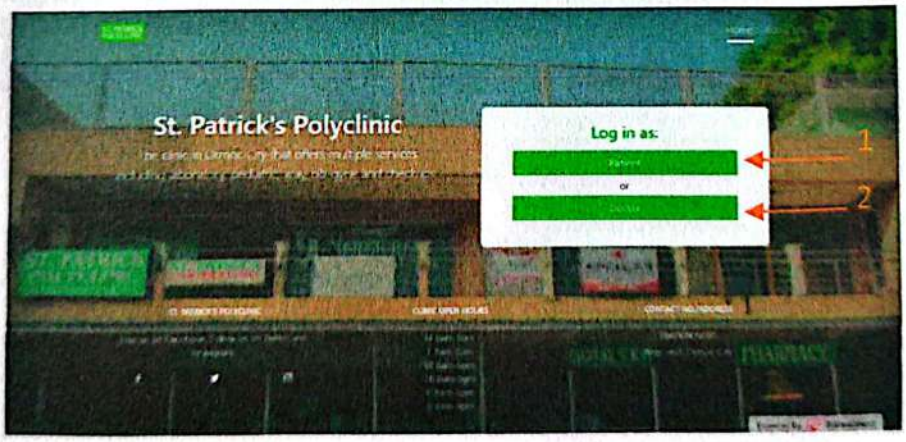
Click (1) Doctors to Add New Doctor, (2) To make new Message and (3) to edit Admin Profile



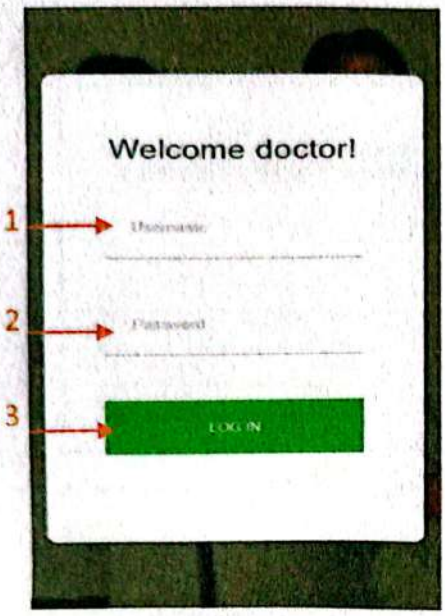
Fill out all the needed information (1) to add new Doctor

Click (2) Log out.

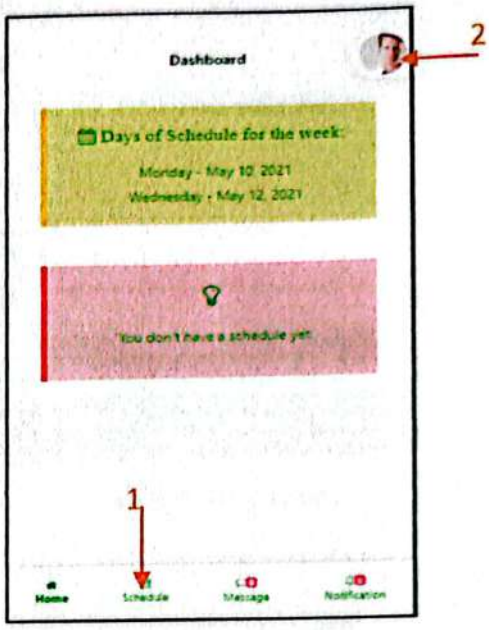
Search (1) search saint-patrick-polyclinic.000webhostapp.com



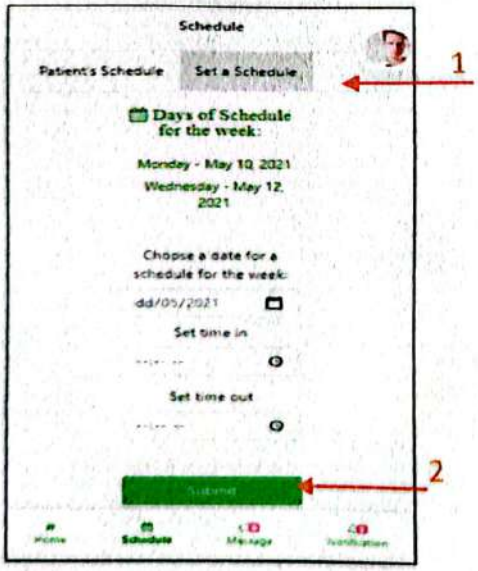
Click (1) if you are a Patient User then (2) if a Doctor User



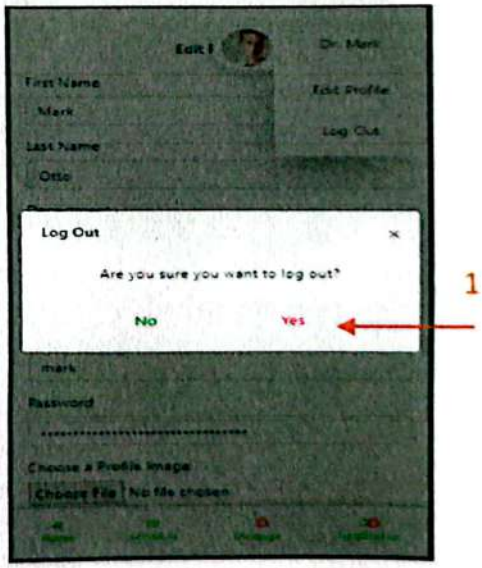
Fill out the (1) Username and (2) Password using the Doctor Log in Credentials, then click (3) LOG IN button.



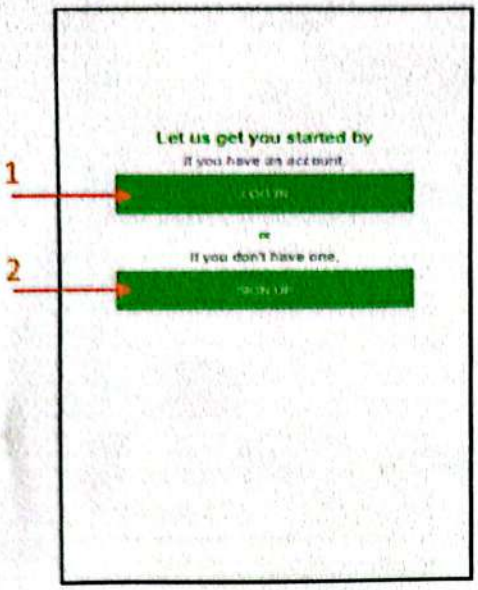
Click (1) Schedule to add new Schedule and (2) To view or update Doctor Profile



Click (1) to set new schedule and (2) to Submit

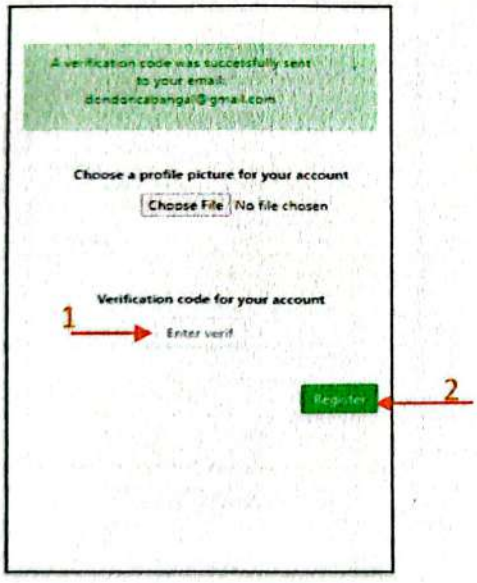


Click (1) Yes to Log out

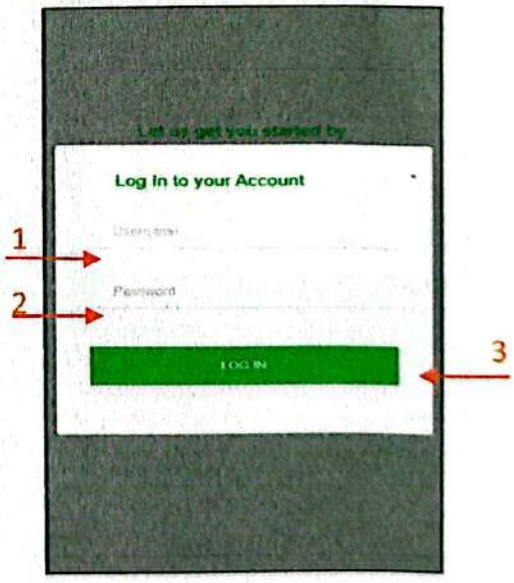


Click (1) to Log In and (2) to Sign- Up

Click (1) to Proceed

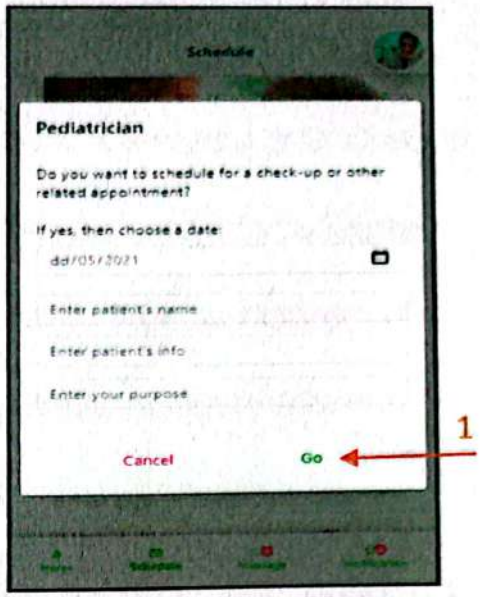


Input (1) Verification code then Click (2) Register

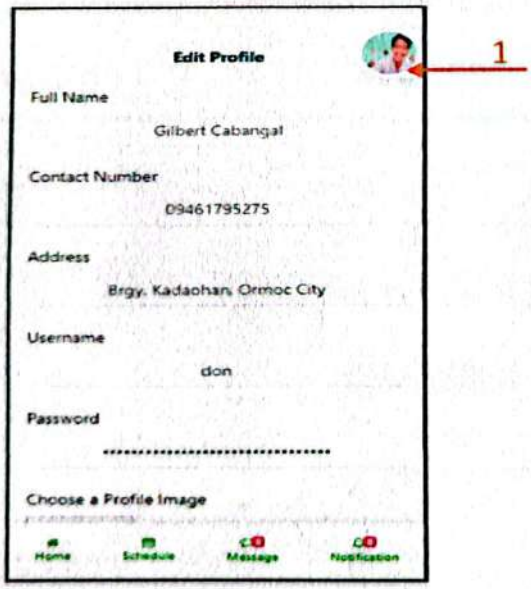


Fill out the (1) Username and (2) Password using the Doctor Log in Credentials, then click (3) LOG IN button.

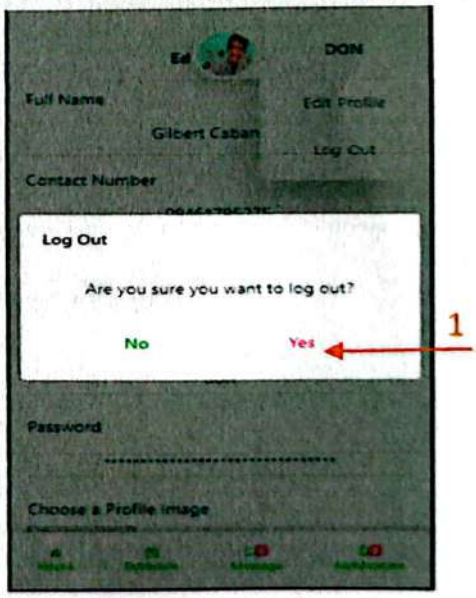
Click (1) Schedule to make an appointment to a Doctor



Fill out all the necessary details then click (1) Go to proceed



Click (1) Update profile



Click (1) Log out

APPENDIX F: ACCEPTANCE SHEET

The thesis proposal entitled "ST. Patrick Polyclinic Scheduling and Queuing System" of Eastern Visayas State University prepared and submitted by Rasalie C. Pedroso, Gilbert T. Cabangal, and Zherra Stifannie D. Serat in fulfillment of the degree of Bachelor of Science in Information Technology is hereby accepted and recommended for approval and acceptance.

MR. WILFERD JUDE PERANTE

ADVISER

Approved by the members of the Evaluating Panel in Oral Examination with the mark of PASSED.

ENGR. JOSEPH JAYMEL S. MORPOS, MSIT

CHAIRMAN

ENGR. EDWARD B. BERTULFO, MSIT

MEMBER

MR. ALEXANDER B. HIPE

MEMBER

Accepted in fulfillment of the requirement for the degree of **Bachelor of Science in Information Technology.**

ENGR. JOSEPH JAYMEL S. MORPOS, MSIT

Adviser, BS Information Technology

ENGR. RUDERICO ENDRIANO JR.

Department Head

APPENDIX G: GRAMMARIAN'S CERTIFICATION

July 7 2021

GRAMMARIAN'S CERTIFICATE

This is to certify that the undersigned has reviewed and went through all the pages of the proposed project study / research entitled " ST. Patrick Polyclinic Scheduling and Queuing Management System" as against the set of structural rules that govern the composition of sentences, phrases, and words in the English language.

Signed:

CARLOTA S. CAÑETE
Signature over Printed Name

Conforme:

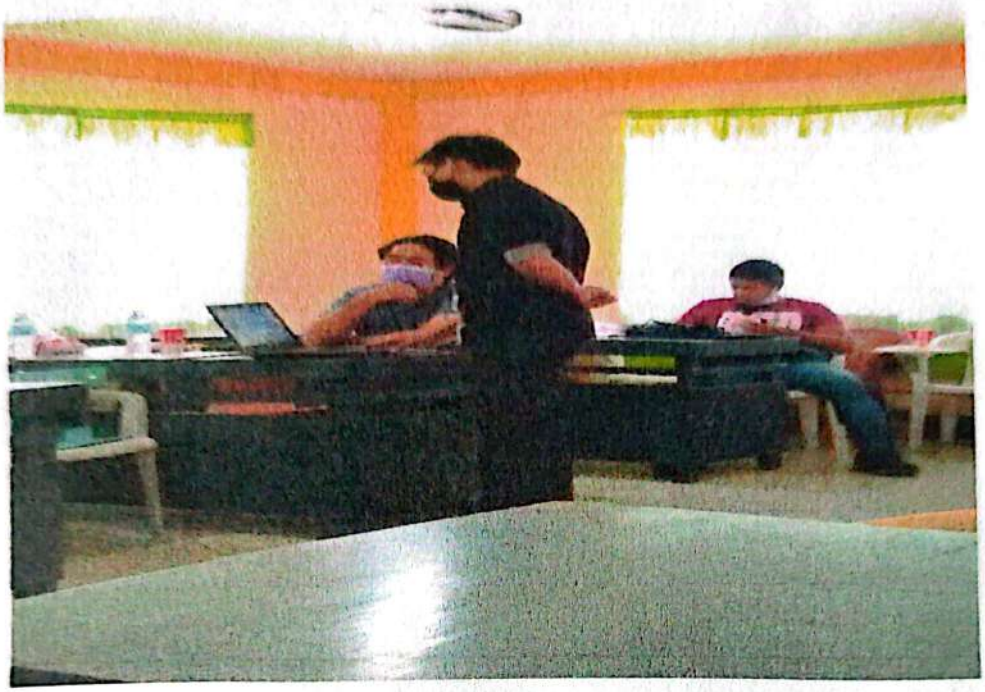
Rasalie C. Pedroso

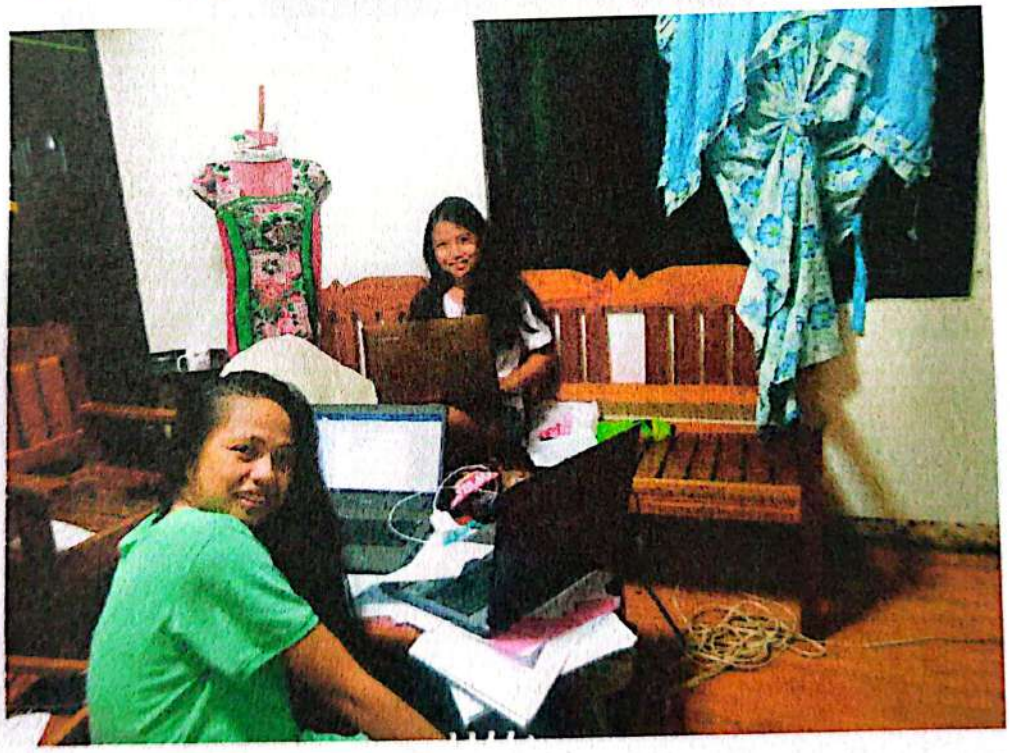
Gilbert T. Cabangal

Zherra Stifannie D. Serat

APPENDIX H: PICTURES

This are some of our pictures while developing this capstone project.





CURRICULUM VITAE

RASALIE C. PEDROSO

Brgy. Sto. Niño Kananga,Leyte,6531
Contact No. +639675101166
E-mail Address: rasaliep@gmail.com

Objective:

To be able to attach this document for thesis purposes.

Personal Information:

Nickname: Ras/Salie
Date of Birth: March 03 1998
Place of Birth: Malagutay, Zamboanga City
Religion: Born Again Christian (JMCIM)

Nationality: Filipino
Age: 23 y/o
Premarital Status: Single

Educational Attainment:

Tertiary — **Eastern Visayas State University**
Bachelor of Science In Information Technology
Brgy. Don Felipe Larrazabal, Ormoc City, 6531
SY: 2020-2021

Secondary — **Kananga National High School**
Poblacion Kananga, Leyte
SY: 2013-2014

Primary — **Sto. Nino Elementary School**
Kananga, Leyte
SY: 2009-2010

GILBERT TURTOGA CABANGAL

Brgy. Kadaohan, Ormoc City, Leyte 6541

09661356539

gilbert.cabangal@evsu.edu.ph

Objectives: To be able to attach this document for thesis purposes.

Experience: No experience yet.

Education:

Elementary: Valencia Central School – Brgy. Valencia Ormoc City (2003-2009)

Secondary: Valencia National High School – Brgy. Valencia Ormoc City (2009-2013)

Tertiary: Eastern Visayas State University – Brgy. Don Felipe Larrazabal Ormoc City

Course: Bachelor of Science in Civil Engineering (2013-2015),

Bachelor of Science in Information Technology (2016-Present)

ZHERRA STIFANNIE DIGNOS SERAT

Sitio Gardenia Brgy. Libongao Kananga, Leyte 6531, Philippines

09564479502

zherrastifannie.serat@evsu.edu.ph

Objectives: To be able to attach this document for thesis purposes.

Experience: No experience yet.

Education:

Elementary: Kananga Central School – Brgy. Poblacion Kananga, Leyte (2004-2010)

Secondary: Kananga National High School – Brgy. Poblacion Kananga, Leyte (2010-2014)

Tertiary: Visayas State University – Brgy. Cagnocot Villaba, Leyte

Course: Bachelor of Secondary Education (2014-2015)

Eastern Visayas State University – Brgy. Don Felipe Larrazabal Ormoc City

Course: Bachelor of Science in Information Technology (2017-Present)