**DOCTOR E-APPOINTMENT MANAGEMENT SYSTEM**

**A CASE STUDY OF TUDOR HEALTH CARE MOMBASA**

**OKUMU HASTINGS OWINO**

**BTIT/296J/2017**

**SUPERVISOR**

**MR. OGWOKA**

**TECHNICAL UNIVERSITY OF MOMBASA**

**Research Project Submitted in Partial Fulfillment for the Degree in Bachelor of Technology in Information Communication Technology of Technical University of Mombasa**

**NOVEMBER 2022**

# **DECLARATION**

**Declaration by the student**

I do hereby declare that this project is my original work and has not been presented for a degree in any other University or for any other award.

Students Name: \_\_\_\_\_\_\_\_\_\_\_\_\_

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_

**Declaration by the supervisor**

I confirm that the work reported in this project was carried out by the candidate under my supervision.

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_

# **DEDICATION**

I dedicate this project proposal to my family, friends and my able lecturer Mr.Ogwoka who has given me great knowledge in doing research.

# **ACKNOWLEDGEMENT**

I am greatly indebted and thank all those who assisted me to make both my proposal and thesis writing a success. First to the Almighty God who gave me the good health to accomplish the work. Special thanks to my supervisor Mr.Ogwoka of Technical University of Mombasa; in his own capacity put a lot of effort throughout my project proposal and the thesis work. He gave me an invaluable guidance and assistance in form of reference and journal articles.

# **ABSTRACT**

The study was focused on designing a Doctor E-Appointment Management System. This an application that will be used to manage efficiently the appointment services within Tudor Health care. Patients will register and login and book an appointment based on doctor availability schedule data added by the doctor after login. The system will assist the institution to store, access and keep track of appointment data over cloud. The doctor will have ability to access appointments remotely. The current appointment management system is characterized by the manual way of booking and managing the appointments which has been an evident as long queues, delayed services, tedious and insecure. To curb these issues an effective and efficient system was proposed in this paper which will ease these services and save the wait time ,cut costs, increase patient satisfaction and simplify process of finding a doctor. System was guided by the following objectives: Analyzing current system to identify user requirements, designing the system, developing and testing the system. System development was guided by RAD methodology. The web server was Apache HTTP Server .The proposed system was developed and designed and managed using JAVASCRIPT,HTML,CSS for the front-end and MySQL and PHP for the back-end. On successful implementation, the system offered more efficient, cost-effective services.

Table of Contents

[**DECLARATION** i](#_Toc120111689)

[**DEDICATION** ii](#_Toc120111690)

[**ACKNOWLEDGEMENT** iii](#_Toc120111691)

[**ABSTRACT** iv](#_Toc120111692)

[**CHAPTER ONE: INTRODUCTION** 1](#_Toc120111693)

[1.0 Background of the study 1](#_Toc120111694)

[1.1 Problem Statement 2](#_Toc120111695)

[1.2 Objectives of the Study 2](#_Toc120111696)

[1.3.1 General Objective 2](#_Toc120111697)

[1.3.2Specific Objectives 2](#_Toc120111698)

[**CHAPTER TWO: LITERATURE REVIEW** 3](#_Toc120111699)

[2.0 Introduction 3](#_Toc120111700)

[2.1 Theoretical Review 3](#_Toc120111701)

[2.2 Similar Systems 5](#_Toc120111702)

[2.3 Critical Review and Research Gap Identification 7](#_Toc120111703)

[2.4 Chapter Summary 8](#_Toc120111704)

[**CHAPTER THREE: SYSTEM DEVELOPMENT METHODOLOGY** 9](#_Toc120111705)

[3.1 Introduction 9](#_Toc120111706)

[3.2 System development methodology 9](#_Toc120111707)

[3.3 System Requirement Analysis 11](#_Toc120111708)

[**3.3.1** **Functional requirements** 11](#_Toc120111709)

[**3.3.2** **Non-Functional requirements** 12](#_Toc120111710)

[**3.3.3 Unified Modelling Language** 12](#_Toc120111711)

[3.4 Database Design 19](#_Toc120111712)

[3.4.1 Table Design 19](#_Toc120111713)

[**3.4.2 3NF Normalized form** 23](#_Toc120111714)

[**3.5 Testing Design** 23](#_Toc120111715)

[**3.5.1 Test cases** 24](#_Toc120111716)

[3.5 Chapter Summary 25](#_Toc120111717)

[**4.1 Introduction** 26](#_Toc120111718)

[**4.2 Presentation of Findings** 26](#_Toc120111719)

[**4.2.1 Objective 1** 26](#_Toc120111720)

[**4.2.2 Objective 2** 26](#_Toc120111721)

[**4.2.3 Objective 3** 26](#_Toc120111722)

[4.2.1 Admin module 30](#_Toc120111723)

[4.2.3 Doctor Module 32](#_Toc120111724)

[4.2.4 Patient Module 34](#_Toc120111725)

[4.2.5 Report screenshots 36](#_Toc120111726)

[4.3 Testing 37](#_Toc120111727)

[4.3.1 User Acceptance Testing 37](#_Toc120111728)

[**Table 10: User Acceptance Testing** 38](#_Toc120111729)

[**CHAPTER FIVE: SUMMARY OF FINDINGS, CONCLUSIONS AND RECOMMENDATIONS** 40](#_Toc120111730)

[5.1 Introduction 40](#_Toc120111731)

[5.2 Summary of the Achievements 40](#_Toc120111732)

[5.3 System Constraints 40](#_Toc120111733)

[5.4 Conclusions 40](#_Toc120111734)

[5.5 Recommendations 41](#_Toc120111735)

[REFERENCES 42](#_Toc120111736)

[APPENDICES: 43](#_Toc120111737)

[APPENDIX A: BUDGET. 43](#_Toc120111738)

[APPENDIX B: TIMEFRAME 43](#_Toc120111739)

[APPENDIX C: SOURCE CODE 44](#_Toc120111740)

# **CHAPTER ONE: INTRODUCTION**

## **1.0 Background of the study**

Hospitals are essential parts of our lives, providing best medical facilities like doctor e-appointment management system is very crucial for a health care since it brings about convenient and efficient way of managing appointments. E-Doctor appointment management system is a web service for the health care institutions used for minimizing expected waiting in a medical appointment system.(Bosch , Dietz, 2000). doctor e-appointment management system refers to a computerized application that runs over a network specifically the internet for online scheduling of appointments on evaluating metrics of outpatient’s bookings of appointments. (Habibi, Tabesh, 2019) . Within the system one can store, retrieve, delete, update and makes other changes to that patients’ appointment data which is stored in the database. It has users like patients, administrator and doctors. The doctor will give the username and password and login into the application, approve an appointment and send a notification to the patient. and patients can log into the system and book an appointment from the doctor schedule list. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes. (Garg, A. X, McDonald 2005). Critical aspect of changing and improving the effective delivery of health care in developed countries. (Fitzgerald,1997).

Tudor Health Care does not use this kind of a system. Given the numerous benefits that come with computer-based systems, there is need to design such a system. Tudor health care uses paper-based and manual appointment booking and management system, fill in appointment forms by pen and the information is stored on paper and books which can be misplaced . This patient finds it hard to access appointment data since one must visit the institution and also creating long queues, patient data loss, very tiring and ineffective. Hence the call for the design of Doctor E-appointment system.

## **Problem Statement**

The Doctor appointment management system that is currently running at Tudor Health Care is totally paper-based and Manual. This inevitably is not only time wasting but also causes increase in chances of errors, immediate response to queries is difficult, it also consumes large volume of paper work. The inconvenience evidenced in loss of patients’ appointment booking information, data redundancy, long queues and other problems. Paper-based workarounds and limits health data exchange. (Rosemann,2015). It also tiresome to the patients since they have to go to the health care to book an appointment with the doctor. This raises the need for a web-based Doctor appointment management system that can facilitate better appointment booking solution, security of doctor data and patient data, doctors can easily schedule their available time details and it can be accessible and efficient from everywhere.

## **Objectives of the Study**

## **1.3.1 General Objective**

To design Doctor E-Appointment management system that will efficiently manage patients’ appointments so as to reduce the number of paper files in the appointments registry, reduce long queues and to manage these operations effectively and according to the needs and satisfaction of the patients and their data safety security and keep track the history all patient appointment details online.

## **1.3.2Specific Objectives**

1. To analyze current existing Doctor appointment management system and identify user requirements.

2. To design the Doctor appointment Management system with all its user interfaces, patients’ side and administrator or doctor side.

3. To develop the designed the Doctor appointment Management system.

4. To test the implemented system to ensure its performance.

# **CHAPTER TWO: LITERATURE REVIEW**

## **2.0 Introduction**

The literature review of this project covers previously researched material from different scholars

on doctor appointment management system and security. Through researchers’ attempts, various theories on how doctor appointment management system has been propagated. These theories produce systems to assist medical doctors in the easy management of patient appointment bookings, managing the information in Health care institutions, existing information reporting systems, improving management, reporting and the use of data warehousing efficiently to improve management reporting. Some existing systems will be explained and then critically evaluated in terms of usability, functionality and efficiency of user interface

## **2.1 Theoretical Review**

Theoretical review provides a benchmark into the existing systems information and assist in gaining necessary information relevant to the area of study authenticity. The internet accelerates the communication and understanding between people, which makes information unprecedented important. (Fischer, J. E., & Steurer, J. ,2001). Furthermore, it changes the way patients book appointments. The Internet has been becoming a powerful channel for business marketing and communication (Palmer, 1999).This paper analyzes and designs the administrator operations, and achieves the functions of register, log-in, appointment bookings, patient appointments booking management, and user interface in order to improve the efficiency of doctor appointment booking application and management online and in the database. Database is a logical collection of interrelated information, managed and stored as a unit, usually on some form of mass-storage system such as magnetic tape or disk. (Kroenke, 2002). ( Gupta, D., & Denton, B. ,2008). Appointment scheduling in health care: Challenges and opportunities.

According to (Armstrong, Grossi, & Sauter, 2014) an online booking system includes a computer connected to a wide-area network, the computer including a processor and a memory configured to store programming and data. An act database includes entertainment records associated with a plurality of entertainers. A booking system allows your organization to manage your resources worldwide. With one location on the internet, your members can book resources wherever they are in the world. Your members can also find who has booked a resource, when it is available and who wants it in the future.

An online system is also known as a web-based system. A web is made up of page that is commonly known as web page or website, and a website is a computer program that runs a web server that provides access to a group of related web pages. A system is a set of independent components working together to achieve a common objective. Therefore, a web-based system is a system that is accessible over the Internet by a user to perform a particular task for a given purpose. The Internet is a system that is use to connect computers and computer networks. It helps to link millions of computer systems all over the world, and it allows the users to get information stored on other computers from a long distance (James, 1999).

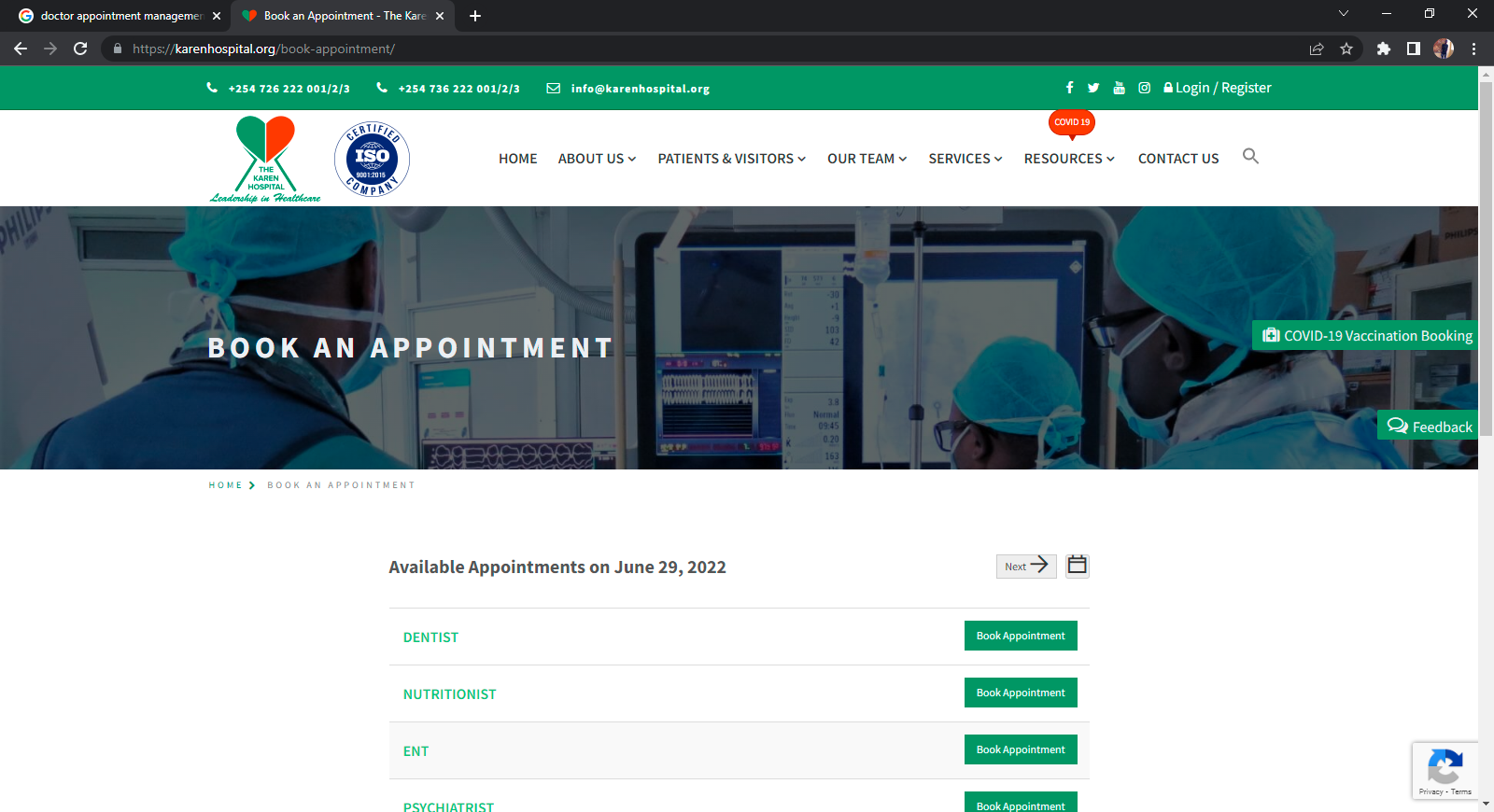
According to ( James, 2010), the public demand for the better hospitals system and the alarming number of missed appointments has forced the hospitals sector to recognize how they deliver care services. With the advance of IT technology today and seen hospitals system as a critical system, appointment booking system lies at the intersection of providing efficient, dependable and timely access to health services. The conventional way of appointment booking is via fax, phone or email. However, with the growing internet penetration, the hospitals industry is moving towards the use of an online appointment booking system.

This manual method of appointment booking procedures and doctor appointments management applied at Tudor Health care in Mombasa is characterized by numerous drawbacks, some of which insecure keeping appointments records of their patients and doctors , doctor available time schedules is difficult. Another disadvantage is that preparing appointment schedules and approval by the doctors manually will take more time. Currently the manual process takes a whole day to verify the records of patient appointments details. Increasing number of patients in Tudor health care gives the administrator more tasks in serving the long appointment queues, approval of the appointments and scheduling doctors availability time. Currently there are a number Hospitals and health cares which have adopted a web-based doctor appointment management system which have greatly made their process of online appointment bookings by patients efficient and convenient also keeping the patients records by administrators very easy. Such institutions include: The Karen hospital online doctor appointment booking system in Kenya, Docpulse Hospital appointment scheduling system in India and Checkups Medical hub appointment management system In Nairobi Kenya.

## **Similar Systems**

1. **The Karen hospital online doctor appointment System.**

The Karen hospital attract many people who needs medical consultations and attentions the online doctor appointment System that it has makes its services easy because of clarity of the system based on patients’ records appointment booking procedures improves health sectors in Kenya. IT handles most complicated cases and which require deeper attention from more qualified personnel. Though it doesn’t offer the best of services it is one of big hospitals.

Figure 1 Snapshot of Karen hospital online doctor appointment System source: https://karenhospital.org

2. **Docpulse Hospital appointment scheduling system**

It private, non-profit hospital in India where the working language is bothEnglish and Hindi . This enables its system to be accessed from anywhere, hence many people fromAfrican countries go to India for advanced treatment.

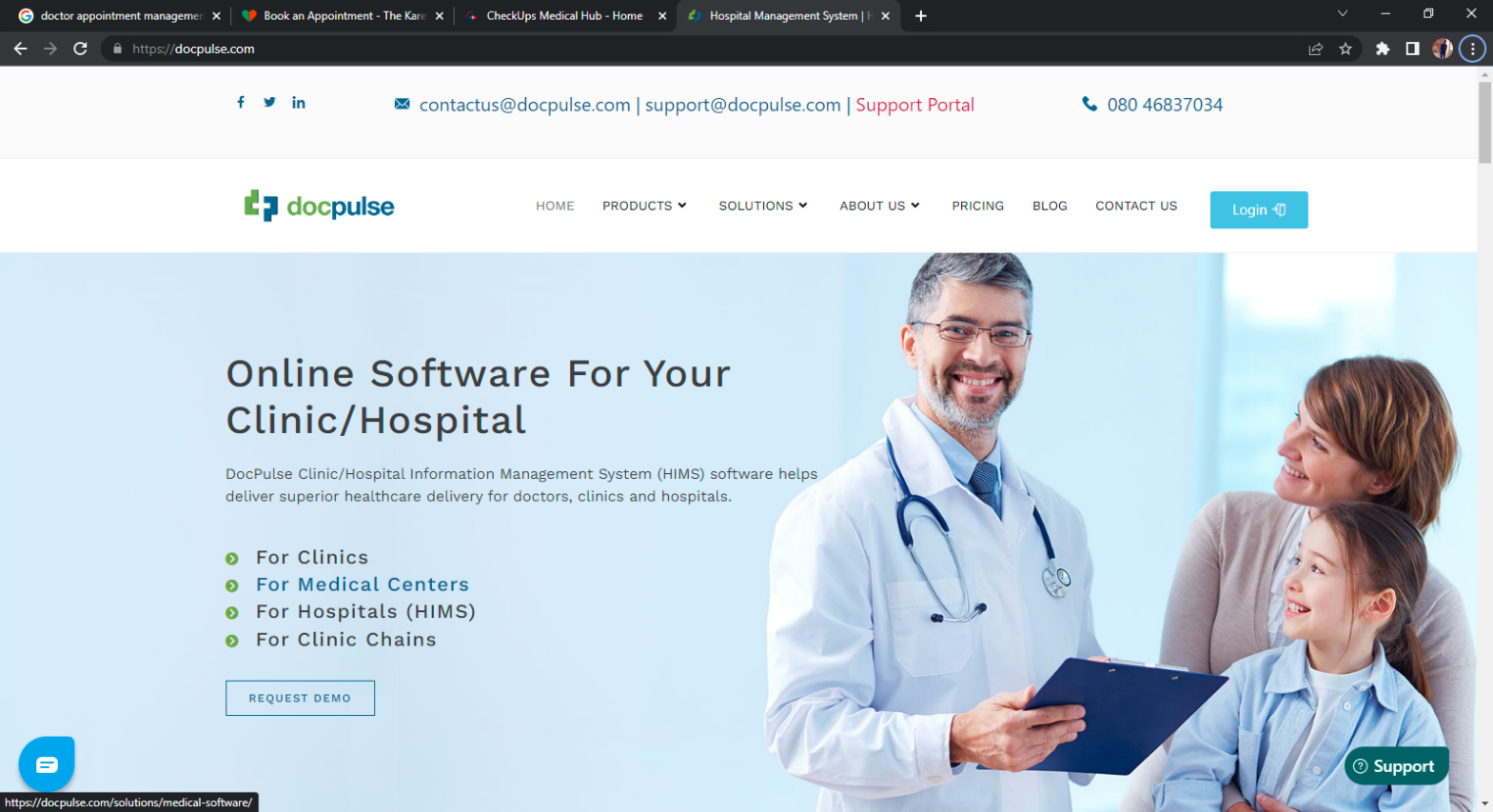
****

Figure 2: Snapshot of Docpulse Hospital Source: <https://docpulse.com/>

**3. Checkups Medical hub appointment management system**

Homepage of the Checkups Medical center . The system homepage layout is clear and user-friendly. Users can browse through to get information easily. Appointment booking also requires patients to fill in their details on a booking form.

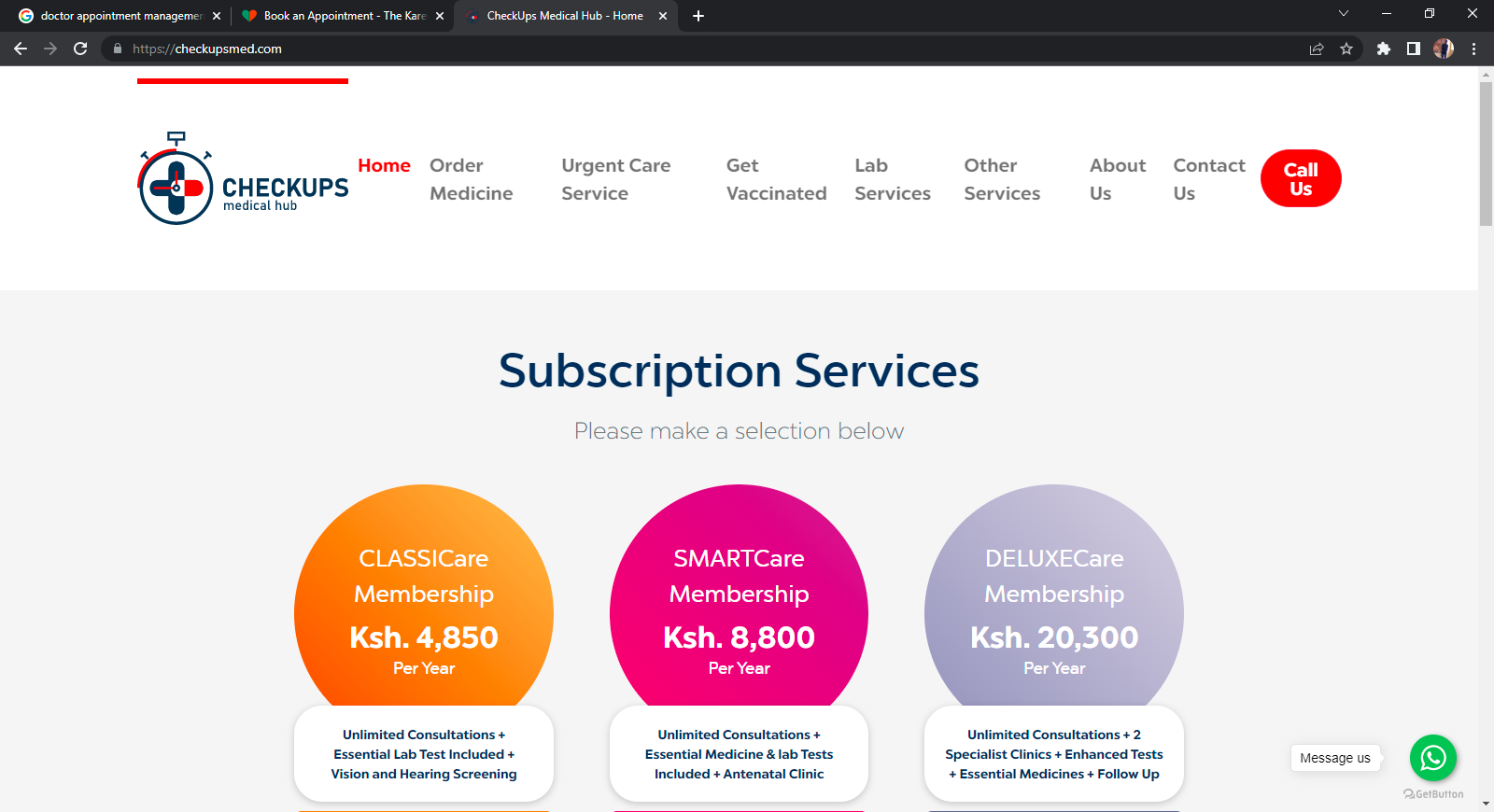
****

Figure 3: Snapshot of CHECKUPS Medical Center source: <https://checkupsmed.com>

## **Critical Review and Research Gap Identification**

The existing systems have a very unfriendly Graphical user interface the pages are very congested and some even fly from anywhere these leads to delays when a user is searching for he or she needs within the system. The systems are also generalized to different types of rooms. A good number of viewed systems requires users to follow many options during registration process which might be tiresome and some of the existing systems require human interaction thus does not automate the process. The proposed system will be one that overtakes the above challenges encountered in the existing systems as it will easily automate the doctor appointment booking process, and also provide a simple graphical user interface which will be easy to interact with since it will contain fewer steps to follow during the interactions especially registration and appointment booking process.

## **2.4 Chapter Summary**

The chapter compared the already existing similar systems with the proposed systems in order to bridge the gap to provide a user-friendly system that is a cross-platform to enable different operating systems and is efficient for both the patients and the Doctors. The proposed system is a web-based application that allows patients to make enquiries online and book for an appointment with the doctor and providing the patients required details. A patient surfs through any of the items of the proposed system. More importantly, patients can make an appointment at their own pace.

# **CHAPTER THREE: SYSTEM DEVELOPMENT METHODOLOGY**

## **3.1 Introduction**

This chapter is based on the methodology used, the type of research design used-in this case descriptive design, the major audience targeted to use the system, basically people around Tudor health care, the doctors and the registrar. It also elaborates how the data was collected, the procedure used, the instruments used, the type of collection used. It contains the functional and non-functional requirements, the methodologies used and the justification of the methodology used. It also uses Unified Modeling Language (UML) i.e., use-case diagram to describe the outline of the system.

## **3.2 System development methodology**

The proposed system was developed using PHP, HTML, CSS and MySQL. Virtual studio software was used for building the web page, chrome as the browser, and the appropriate package was XAMP. The design methodology for the research was Rapid Application Development (RAD) methodology. The RAD method of software development structures the life cycle of the project in phases. Initialization prepares the organization and determines the scope and terms of communication. The framework defines a space for objectives, solutions and resources. The design models the solution and validates its systematic coherence. Construction is realized in an active prototype (permanent validation). The completion is reduced to a final quality. RAD method seeks functional application quality and technique with RAD tools including the automation generation of code.

Online knowledge defines Rapid Application Development as “a methodology that enables organizations to develop strategically important systems faster while reducing development costs, lifespan and maintaining quality. This is achieved by using a series of proven application development techniques, within a well-defined methodology. Traditional development lifecycles are too slow and rigid to meet the business demands of today’s economy due to the increase technology. A new methodology must be implemented, one that allows organizations to build software applications faster, better and cheaper. RAD enables such development. RAD model is preferred over other traditional methodologies because of Faster delivery timesince it reduces the project development life cycle due to a reduced requirement analysis, greater customer satisfaction since it involves active participation of the customers and end users in all stages of analysis, better project management and reduced risk due to iterative prototyping.

Design methodology RAD has Four phases

1**) Requirement planning**

Focuses on why the system need to be built, the system’s requirements are clearly defined and a thorough feasibility study was done from various perspectives including technical, financial and organizational where the system users were involved fully.

2) **User design phase**

This phase will involve problem identification and solution analysis. Prediction of the problems that the developers might face while creating the system. The end-deliverables of this phase determines the method of system creation and provided guidance to developer.

3) **Construction phase**

This is the phase when the actual development of the system takes place, it entailed the actual coding of the system using the specified software and hardware specifications

4) **Implementation phase**

In this phase, the system will be tested for defects that will be rectified and finally installed. Inclusive of essential activities including training and maintenance.

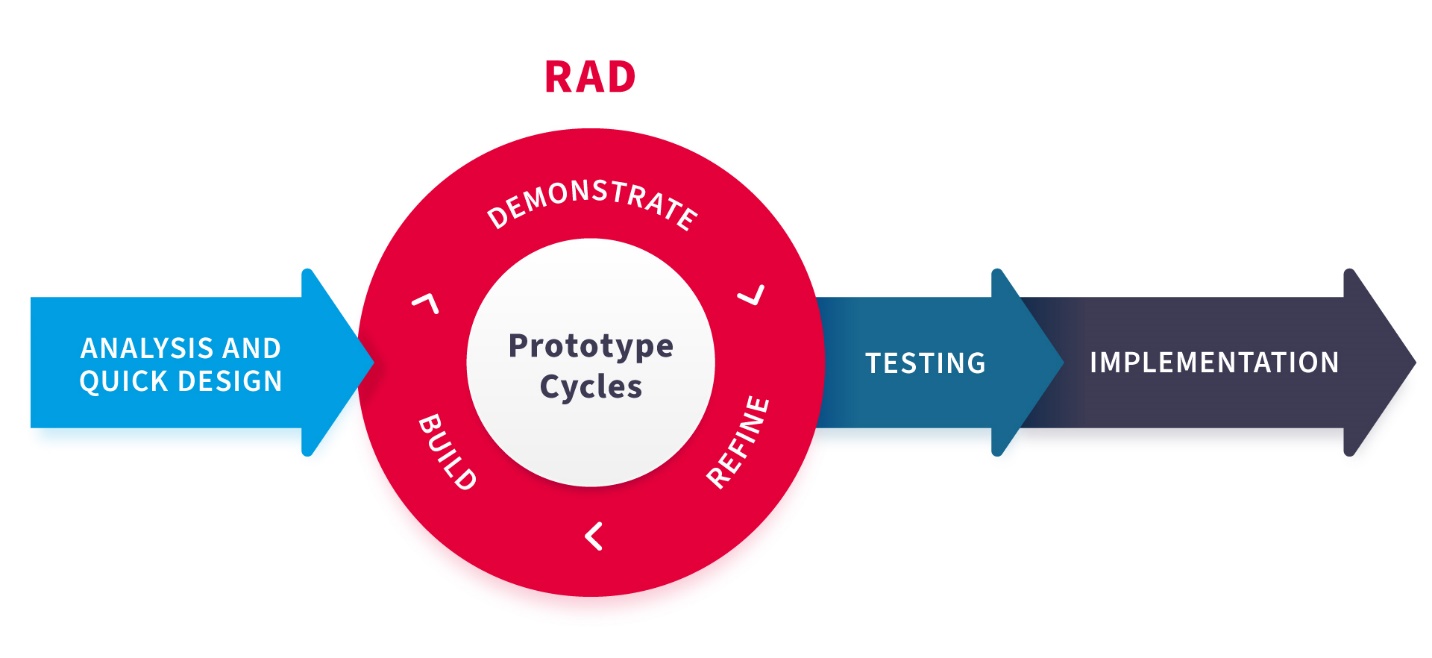


Figure 4 RAD methodology.

## **3.3 System Requirement Analysis**

It refers to the process of determining user’s expectations for a new or modified product. Requirements analysis involves frequent communication with system users to determine specific feature expectations, resolution of conflict or ambiguity in requirements as demanded by the various users or groups of users, avoidance of feature creep and documentation of all aspects of the project development process from start to finish. Requirements are of two types, functional and non-functional requirements

### **Functional requirements**

These are statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.

1.Administrator and doctor, patients should have separate privileges to log in to the system.

2.The system will provide services according to the requirements of the authorized administrator, also system will facilitate to cancel the appointments.

3.Patients should have the ability to get messages through email about their appointment booking details and also to cancel an appointment.

Admin has the following privileges:

1.Add doctor and patient details.

2.Manage patient appointment booking details(Add, Update)

3.Manage doctor schedule time details (Add, Update, Delete)

4.Approve or disapprove new patient appointment application

5.Manage the history of appointments (Add, Edit, Delete)

### **Non-Functional requirements**

The non-functional requirement elaborates a performance characteristic of the system. Non-functional requirements detail constraints, targets or control mechanisms for the new system. They describe how, how well or to what standard a function should be provided. These are constraints on the services or functions offered by the system;

1.Maintaining backups ensure the system database security and reliability

2.Ensuring Availability and High Performance

3.User interfaces should enhance the user friendliness.

4.The system should be implemented in a way to utilize reusability and maintainability of both the design and code.

### **3.3.3 Unified Modelling Language**

This is a general purpose, developmental and modelling language in the field software engineering that is intended to provide a standard way to visualize the design of a system.

#### **3.3.3.1 Use case Modelling**

According to the software and system engineering, a use case is a list of actions or events steps typically defining the interactions between a role known in the UML as an actor and a system to achieve a goal.

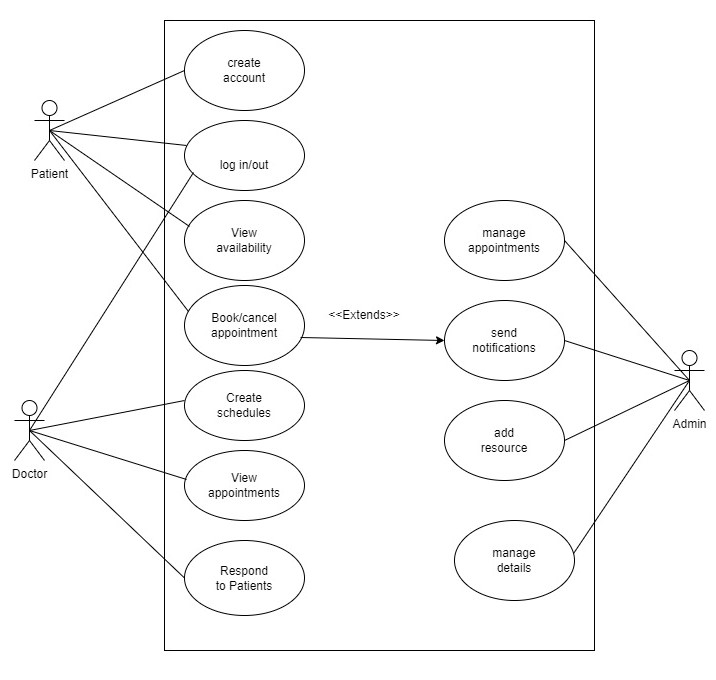


Figure 5: use case diagram source: Draw.io

**3.3.3.2 Activity Diagram**

The activity diagram has a major role in capturing dynamic behavior of a system.

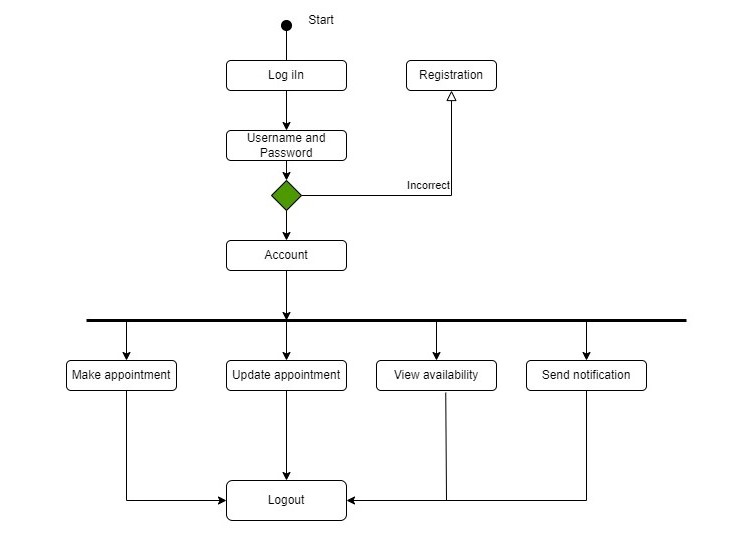


Figure 6: patient activity diagram source: Draw.io

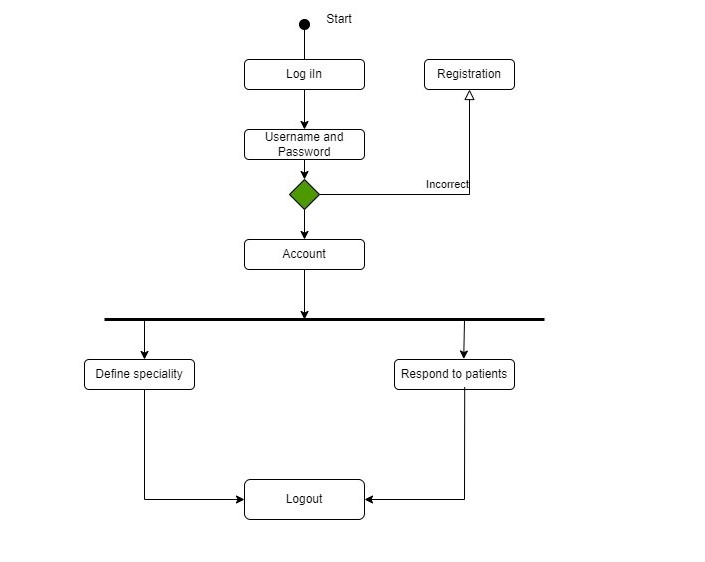
****

Figure 7:Doctor Activity diagram source :Draw.io

****

Figure 8: admin activity diagram source: Draw.io

#### **3.3.3.3 Sequence diagram**

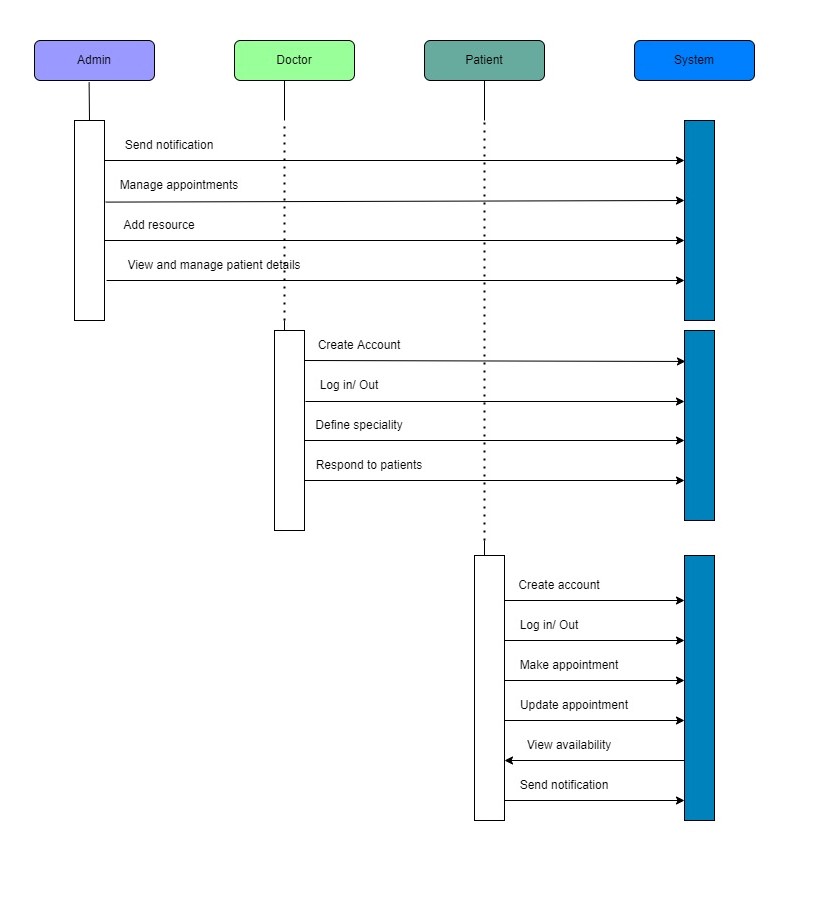
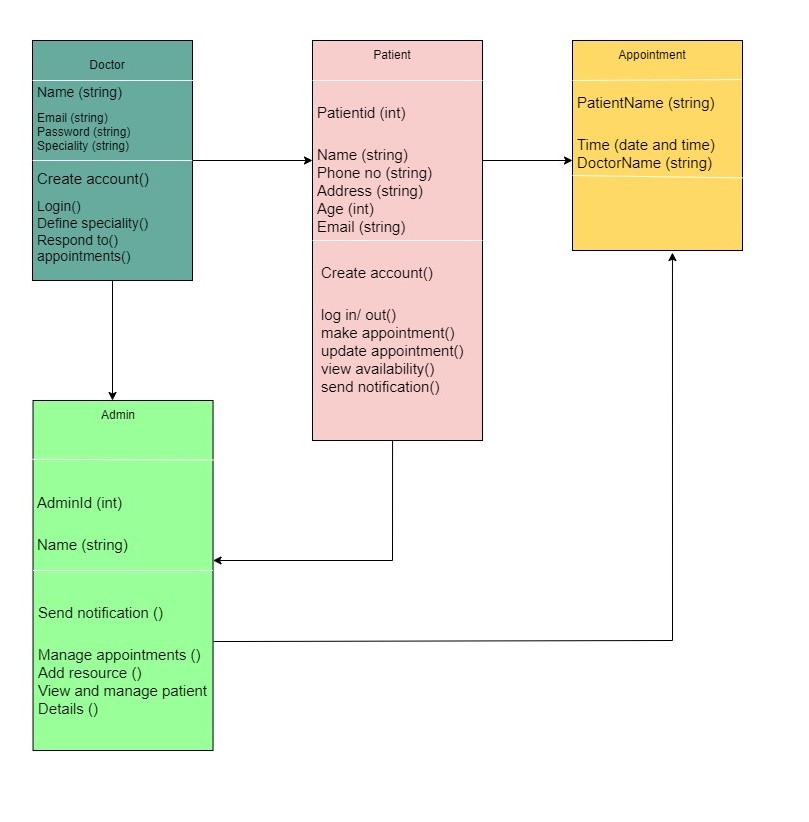
****

Figure 9: sequence diagram source: Draw.io

**3.3.3.4 Class Diagram**

****

**Figure 10: Class diagram source: Draw.io**

#### **3.3.3.5 Deployment diagram**

****

Figure 11:Deployment diagram

## **3.4 Database Design**

This is the organization of data according to database model. It determines how the data elements interrelate and how the data must be stored.

## **3.4.1 Table Design**

Table design involves arranging related data in a table format within a database. A table consists of rows and columns; they have specific number of columns but can have any number of rows

**Table 1: Patient login Table**

This table stores the user’s credentials which are used to login into the system

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATA TYPE | FIELD SIZE |
| Email | User email | VARCHAR | 100 |
| Username | Name for login | VARCHAR | 100 |
| Password | User Password | VARCHAR | 100 |

**Table 2: Admin User login Table**

Stores the admin information to enable them login in the system and manage appointment data made by users from the user interface**.**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATATYPE | FIELD SIZE |
| Email | User Email | VARCHAR | 100 |
| Username | Name for Login | VARCHAR | 100 |
| Password | User Password | VARCHAR | 100 |
| Hospital | Hospital Name | VARCHAR | 150 |
| Address | User Address | VARCHAR | 200 |

**Table 3: Doctor User login Table**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATA TYPE | FIELD SIZE |
| Id | Primary Key | INT | 50 |
| Username | Name for login | VARCHAR | 100 |
| Password | User Password | VARCHAR | 100 |

**Table 4:Patient List Table**

Table 4 below stores the patients’ information in the Database**.**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATA TYPE | FIELD SIZE |
| Patient Id | Primary Key | INT | 50 |
| First Name | Patient First name | VARCHAR | 200 |
| Last Name | Patient Last name | VARCHAR | 100 |
| Password | Patient password | VARCHAR | 200 |
| Email | Patient Email | VARCHAR | 200 |
| Phone | Patient Contact | INT | 100 |
| Address | Patient Address | VARCHAR | 150 |

**Table 5:Doctor list Table**

Table 5 below stores the doctors’ information in the Database

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATA TYPE | FIELD SIZE |
| Id | Primary Key | INT | 50 |
| Doctor ID | Doctor Job ID | VARCHAR | 50 |
| Email | Doctor Email | VARCHAR | 100 |
| Password | Doctor password | VARCHAR | 200 |
| Name | Doctor Name | VARCHAR | 200 |
| Phone | Doctor Contact | INT | 150 |
| Specialty | Specialization Area | VARCHAR | 255 |

**Table 6:Doctor Schedules Table**

Table 6 below stores the doctors’ schedule information in the Database**.**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATA TYPE | FIELD SIZE |
| Schedule Id | Primary Key | INT | 50 |
| Doctor ID | Doctor Job ID | VARCHAR | 200 |
| Schedule Date | Availability Date | DATE |  |
| Start Time | Schedule Start Time | TIME |  |
| End Time | Schedule End Time | TIME |  |
| Consulting Time | Average consult Time | TIME |  |

**Table 7:Appointments Table**

Table 7 below stores the appointments information in the Database**.**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD NAME | DESCRIPTION | DATA TYPE | FIELD SIZE |
| Appointment Id | Primary Key | INT | 50 |
| Doctor ID | Doctor Job ID | VARCHAR | 50 |
| Patient Id | Foreign Key | INT | 50 |
| Schedule Id | Foreign Key | INT | 50 |
| Status | Appointment Status | VARCHAR | 90 |
| Reason | Appointment Reason | VARCHAR | 200 |
| Comment | Doctor Comment | VARCHAR | 200 |

### **3.4.2 3NF Normalized form**

Normalization is a database design technique which organizes tables in a manner that reduces redundancy and dependency of data. Third Normal Form (3NF) is the third step in normalizing a database and it builds on the first and the second normal forms. This normalized form states that all column reference in referenced data that are not dependent on the primary key should be removed.

**Table 8:3NF form**.

|  |  |  |
| --- | --- | --- |
| **Id** | **email** | **Password** |
| 6 | kerenhappuch@gmail.com | Keren2024 |
| 8 | mandy@gmail.com | $sign |
| 9 | roselette@gmail.com | R2022 |
| 10 | kairony@gmail.com | Kai29 |

# **3.5 Testing Design**

Testing is a process of checking if a system functions as expected, more so if it meets the user requirements. The best way of testing would be to involve users so as to get their feedback. This process is repeated until the users are satisfied with the system, however most of the testing is done by the developer when debugging on where errors were corrected to satisfy the users. Different types of testing were done as shown below:

### **3.5.1 Test cases**

These are the set of actions executed to verify a particular feature or functionality of a system. It helps validate the system under test.

**Table 9: Test cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test case id** | **Test scenario** | **Test steps** | **Expected result** | **Actual result** | **Pass/Fail** |
| TC01 | Check patient login with valid data | Enter valid username  Enter valid password  Click login | User should successfully log in to the system | As expected | Pass |
| TC02 | Check patient login with invalid data | Enter invalid username  Enter invalid password  Click login | User should not log in to the system | As expected | Pass |
| TC03 | Verify an input field that can accept a maximum of 200 characters | Log in to the system  Enter 200 characters | The system should be able to accept all these characters | As expected | Pass |
| TCO4 | Verify an input field that can accept a maximum of 250 characters | Log in to the system  Key in 250 characters | The system should not be able to accept all these characters | As accepted | Pass |

## **3.5 Chapter Summary**

This chapter contains and has expounded on the system design focusing on how data will be gathered, analyzed, represented using tables and unified modelling diagrams, also it touches on how the test design should be carried out.

**CHAPTER FOUR: RESEARCH FINDINGS AND DISCUSSION**

# **4.1 Introduction**

This chapter describes how the system was developed using all the requirements found during the research process. The objectives of the project are observed while developing the system.

The chapter describes all the operations of the system.

# **4.2 Presentation of Findings**

Here a clear illustration of each functional and non-functional requirement are shown and explained with regards to the different modules in the system.

### **4.2.1 Objective 1**

The first objective is for the system to meet all the requirements identified during the research gap identification. The developed system conforms to the chosen methodology as earlier analyzed.

### **4.2.2 Objective 2**

The final system meets all the design specifications, the theoretical analysis and system analysis that were made.

### **4.2.3 Objective 3**

The system is in line with the analysis of the results, requirements, research design and the design methodologies. The results of the developed system are described below:

**1.Home page**

Upon launching the system, the user of the system is directed to the home page which has a login form where after logging in they can navigate to other pages that one is interested in.

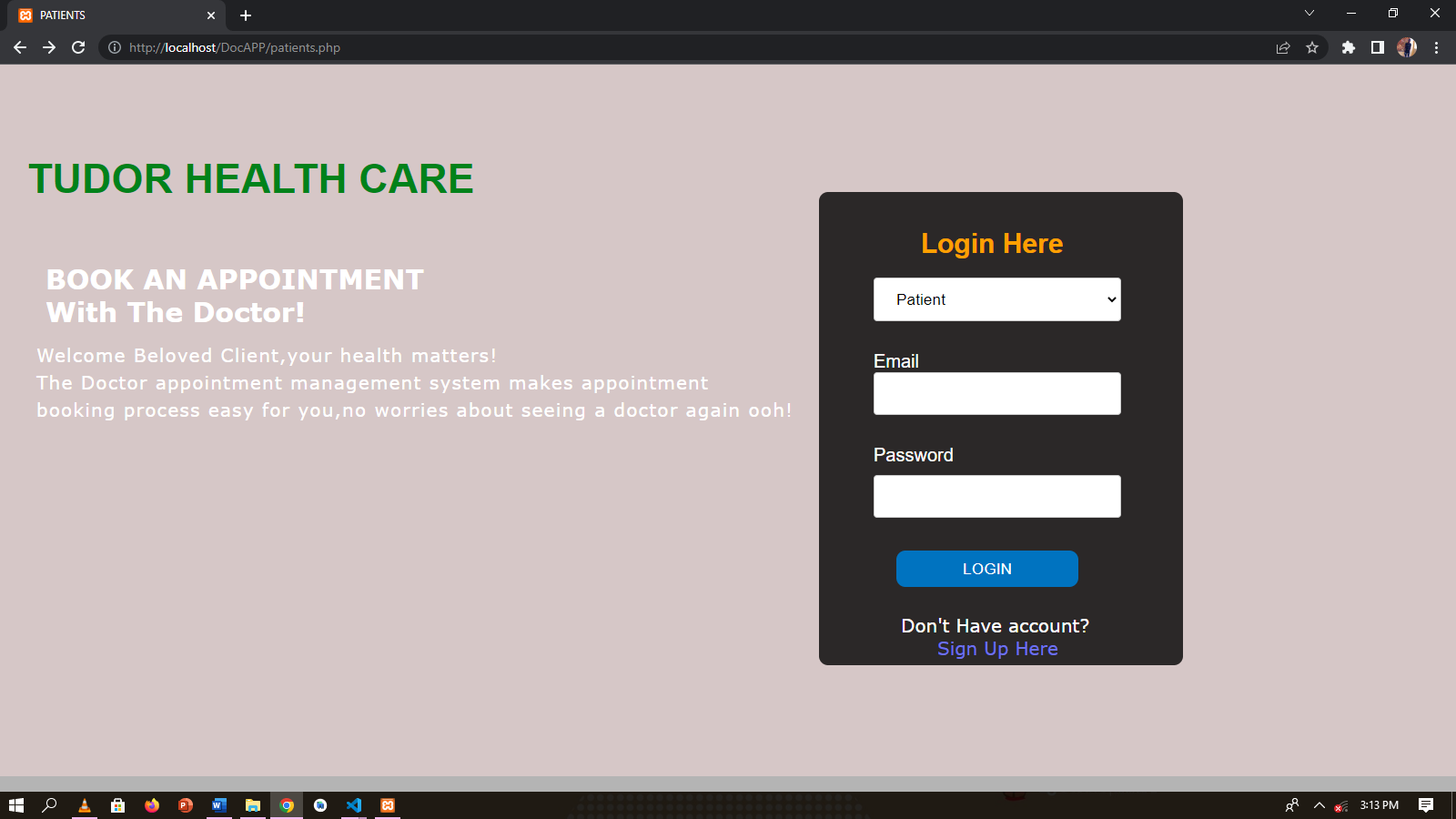
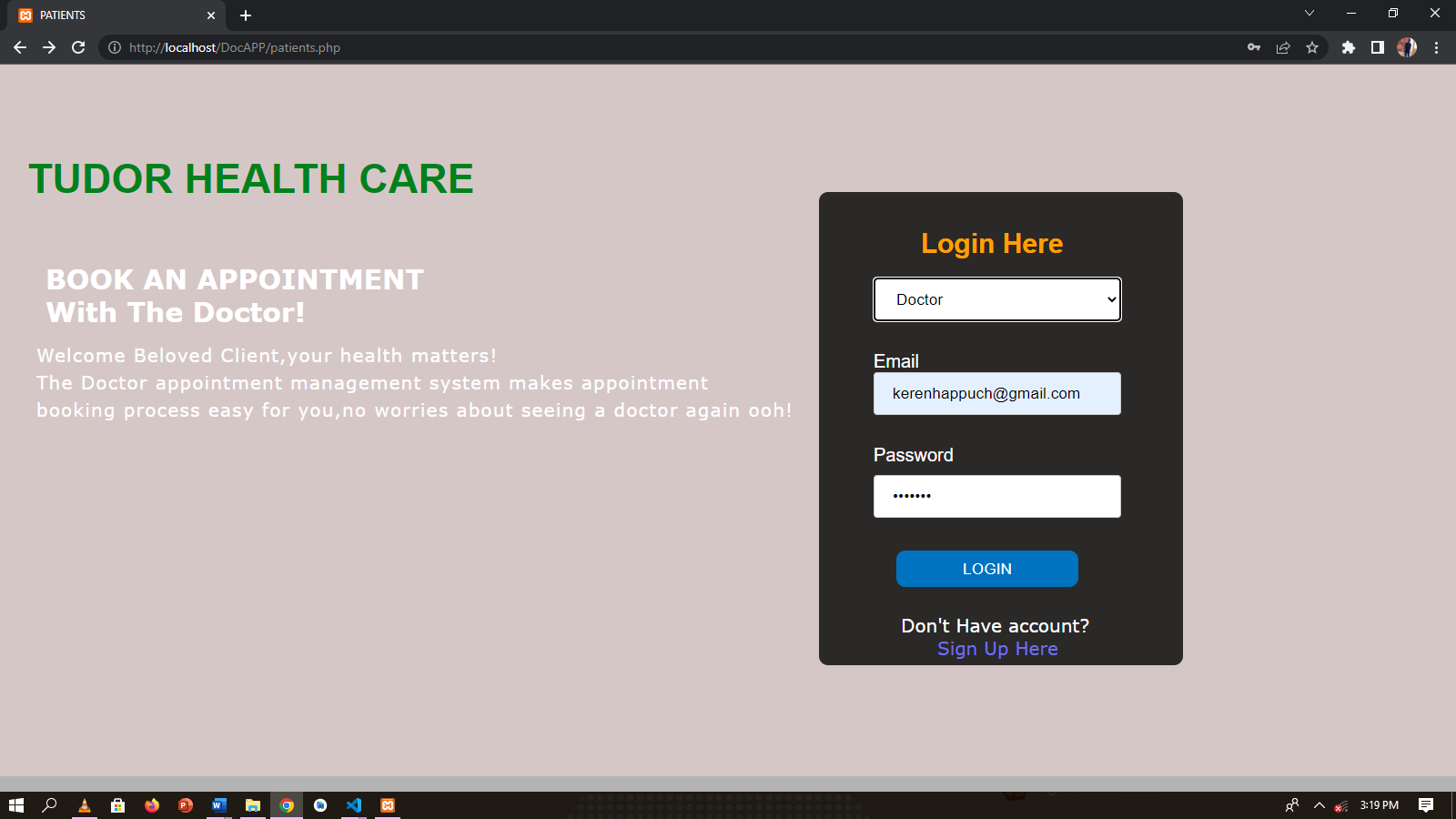


Figure 12:Homepage

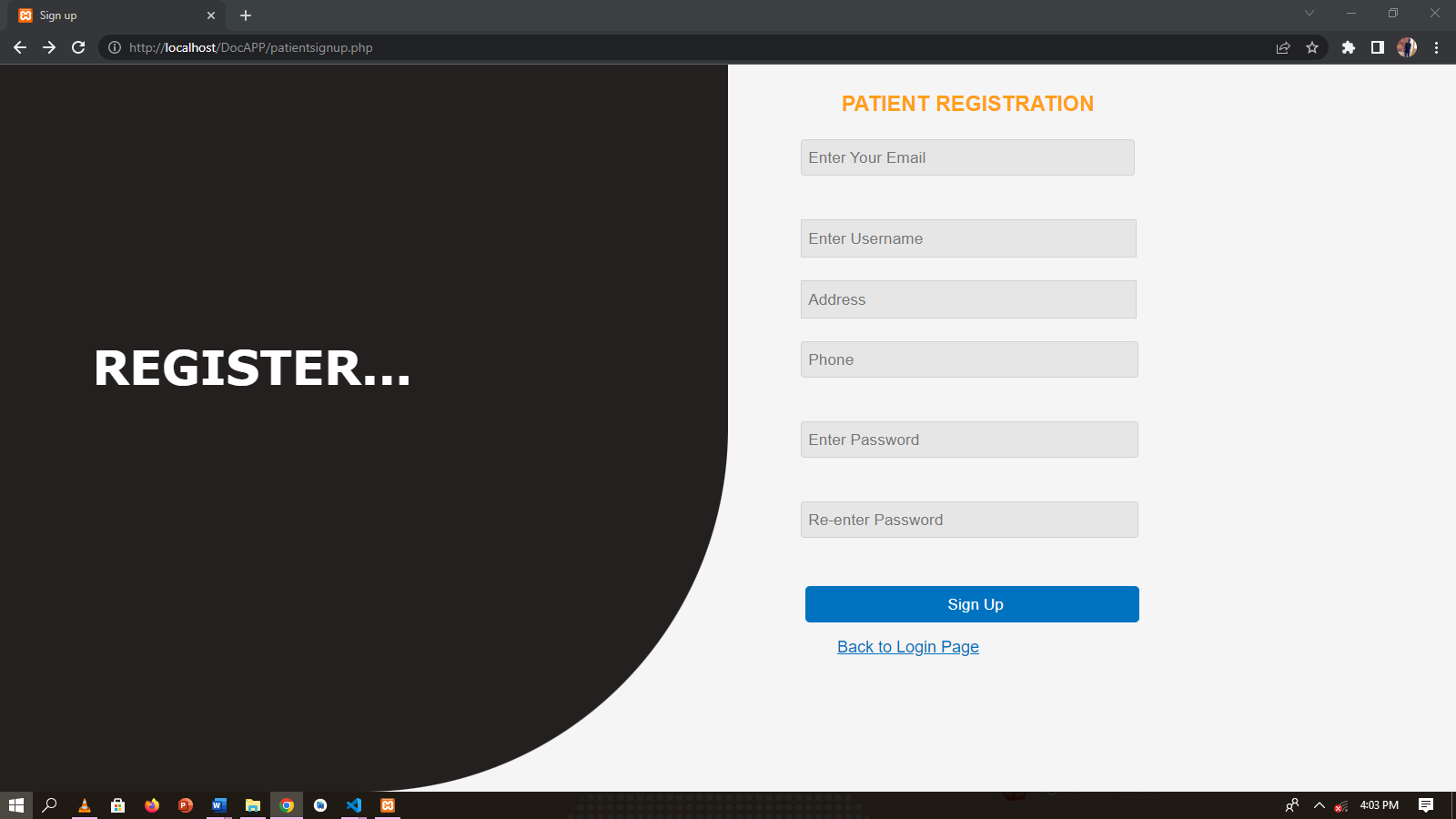
**2.Login page**

The patients, doctor and the administrator all have to login to the system to perform any particular task. By signing up user creates an account or registers into the system, where afterwards the patient can sign in after registering into the system. While signing in, the user signs either as a patient or a doctor. Administrator also signs into the system.

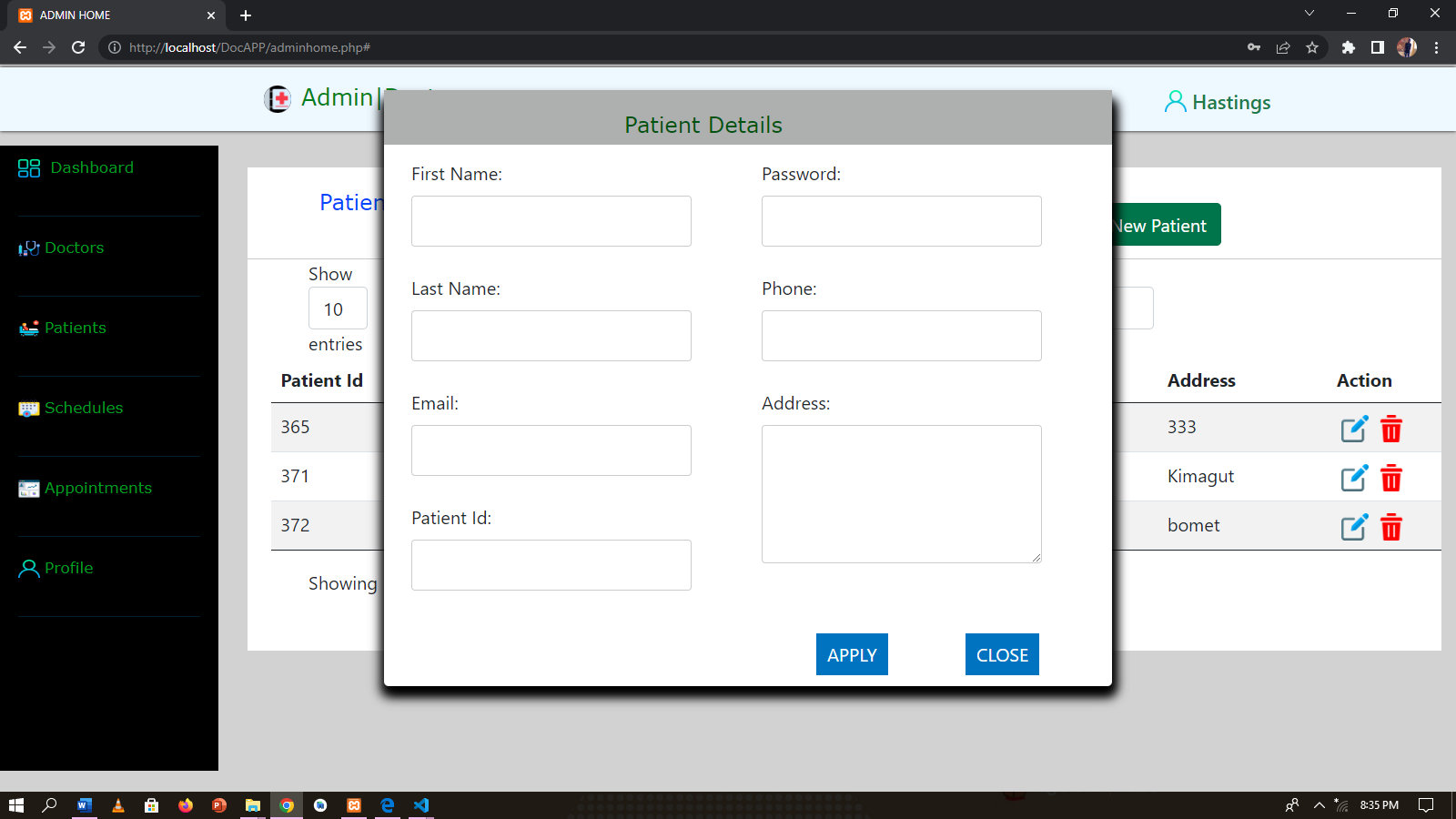
 Figure 13: Login Page

**3.Data Entry Forms**

For a user who is accessing the system for the first time, he/she will be required to sign up first in order to register into the system and hence will be required to fill a registration form.



**Figure 14: Data entry form**



**Figure 15:patient registration by Admin**

**4.Validation Form**

When a user logs in with the correct details, he/she will successfully log in, whereas when a user logs in with wrong details the system will display an error message since the details aren’t recognized by the system.

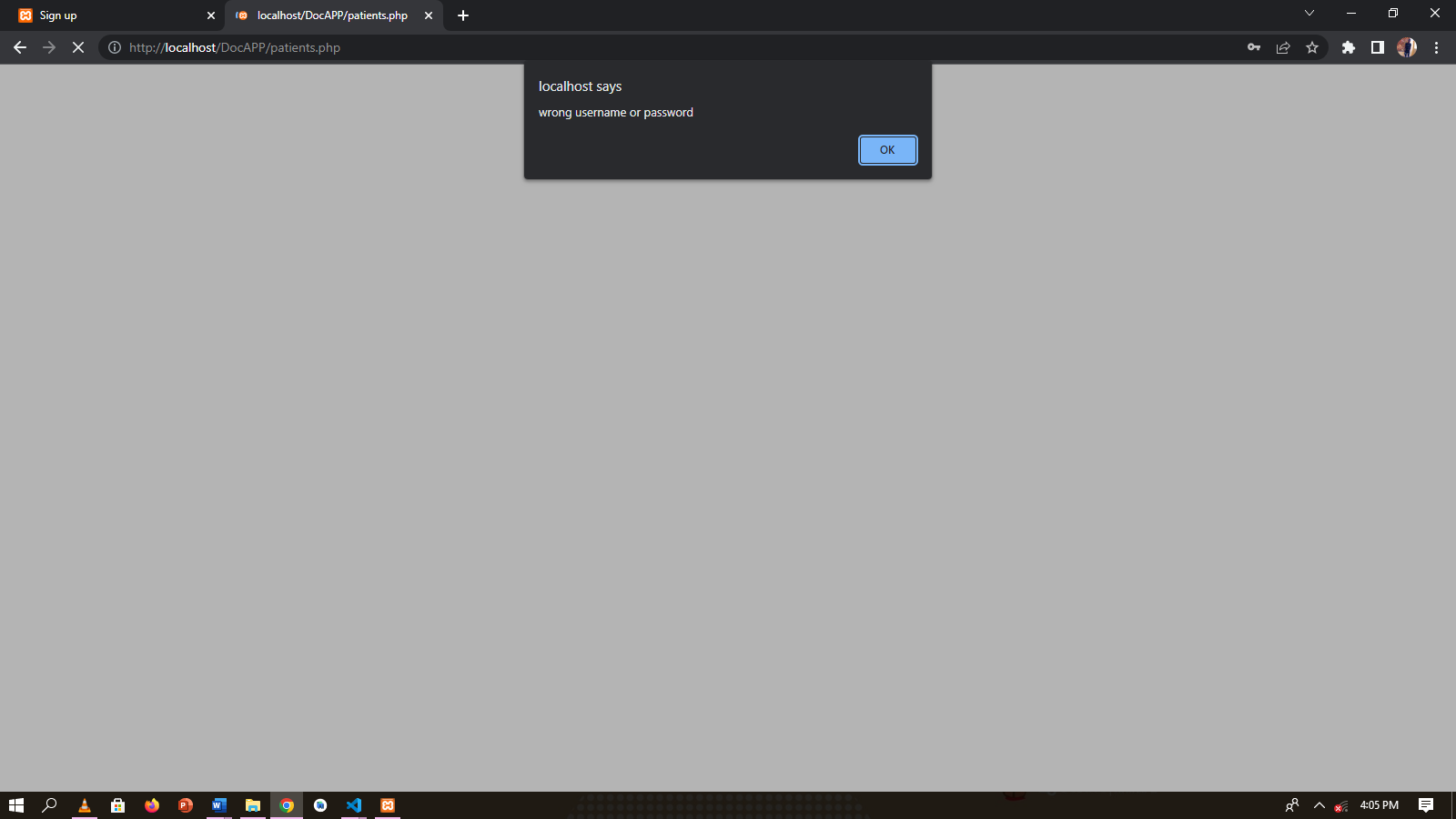


Figure 16: Validation Form

## **4.2.1 Admin module**

The administrator adds and views the patients and doctors. He can also add doctor schedule data and view appointments too.

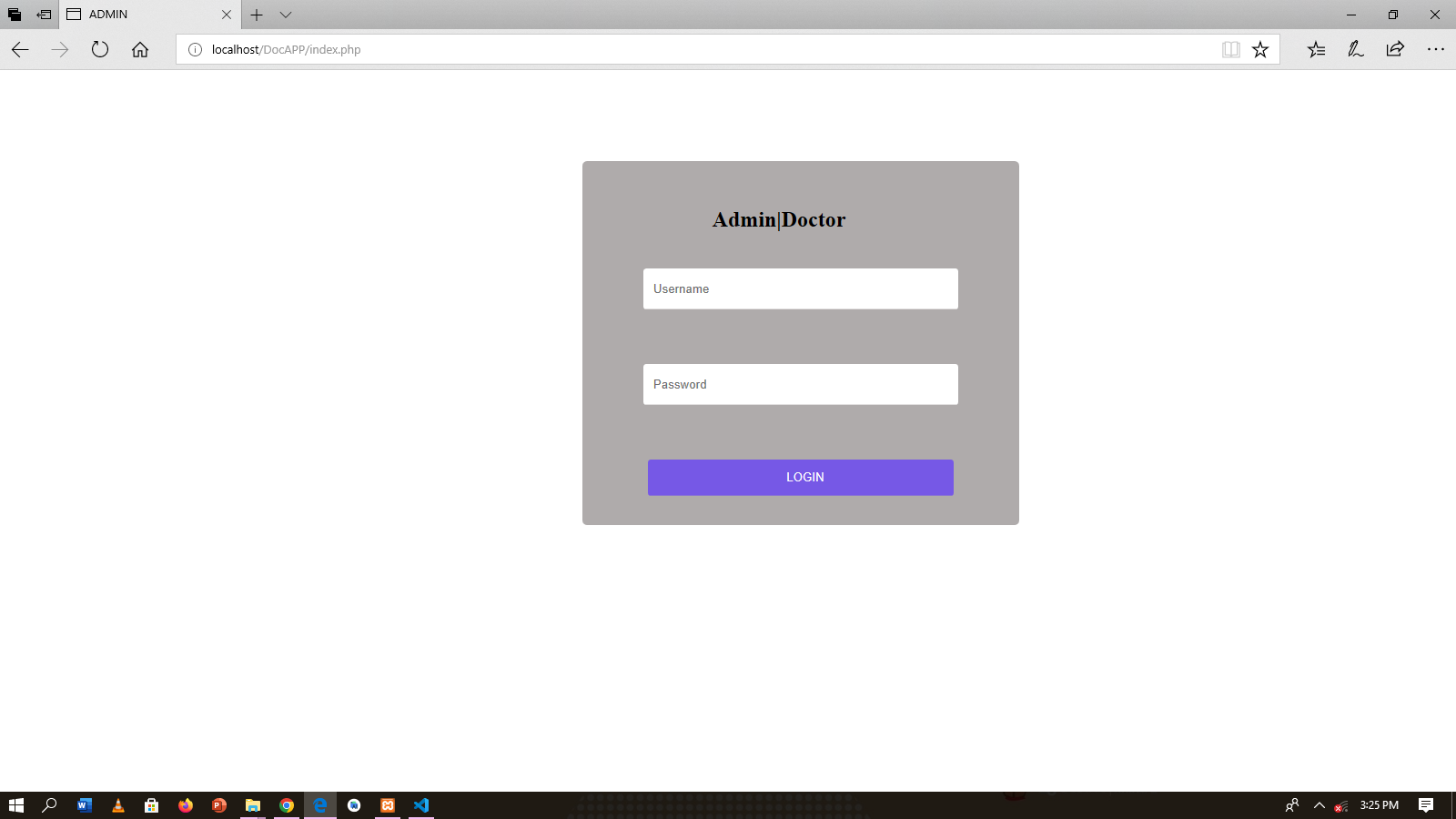


Figure 17: Admin login page

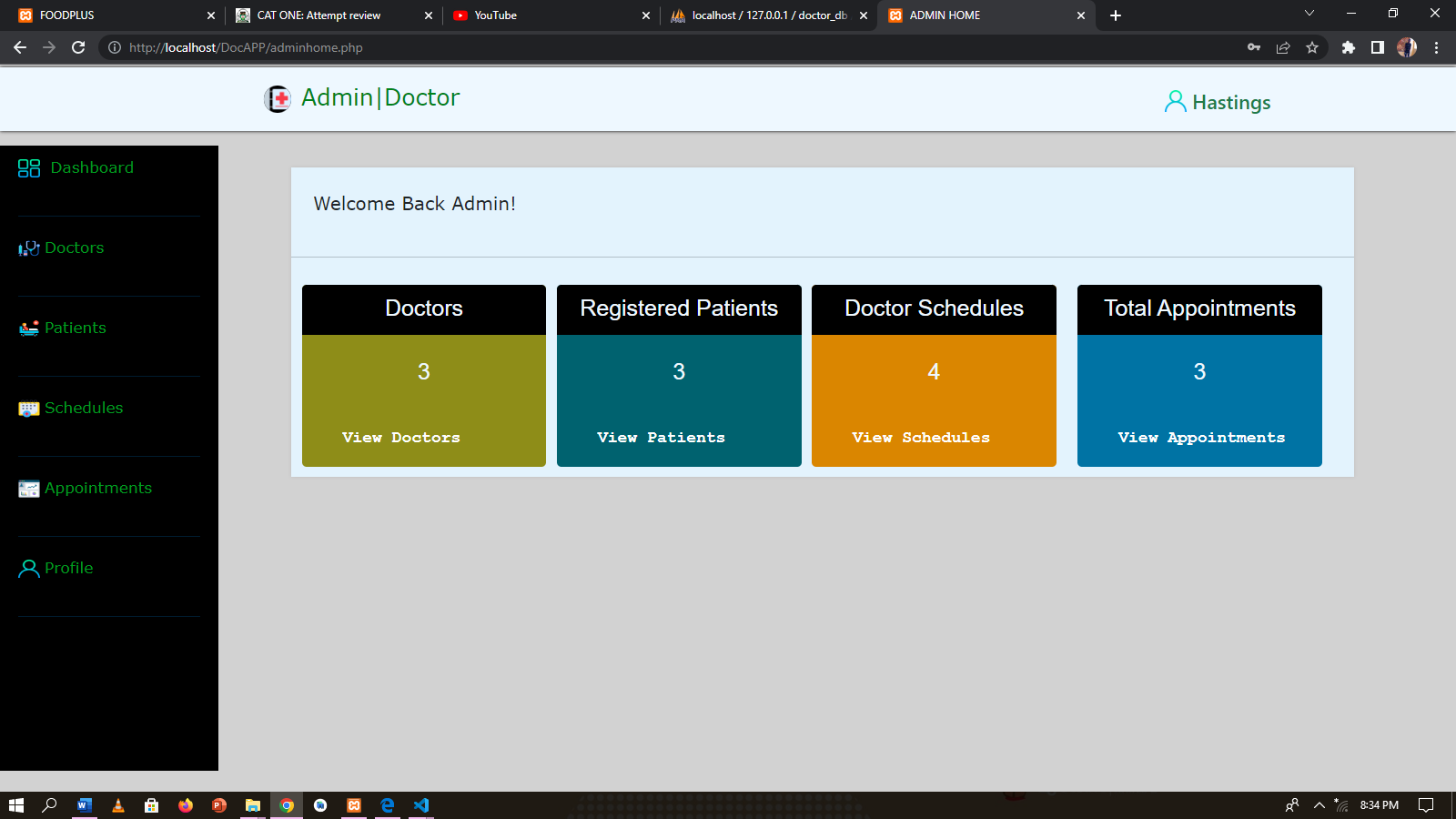


Figure 18:Admin dashboard

Figure 19:Admin user view registered doctors.



Figure 20:Admin user view registered patients.

## **4.2.3 Doctor Module**

The doctor adds his schedule availability data for patients to view and book an appointment.

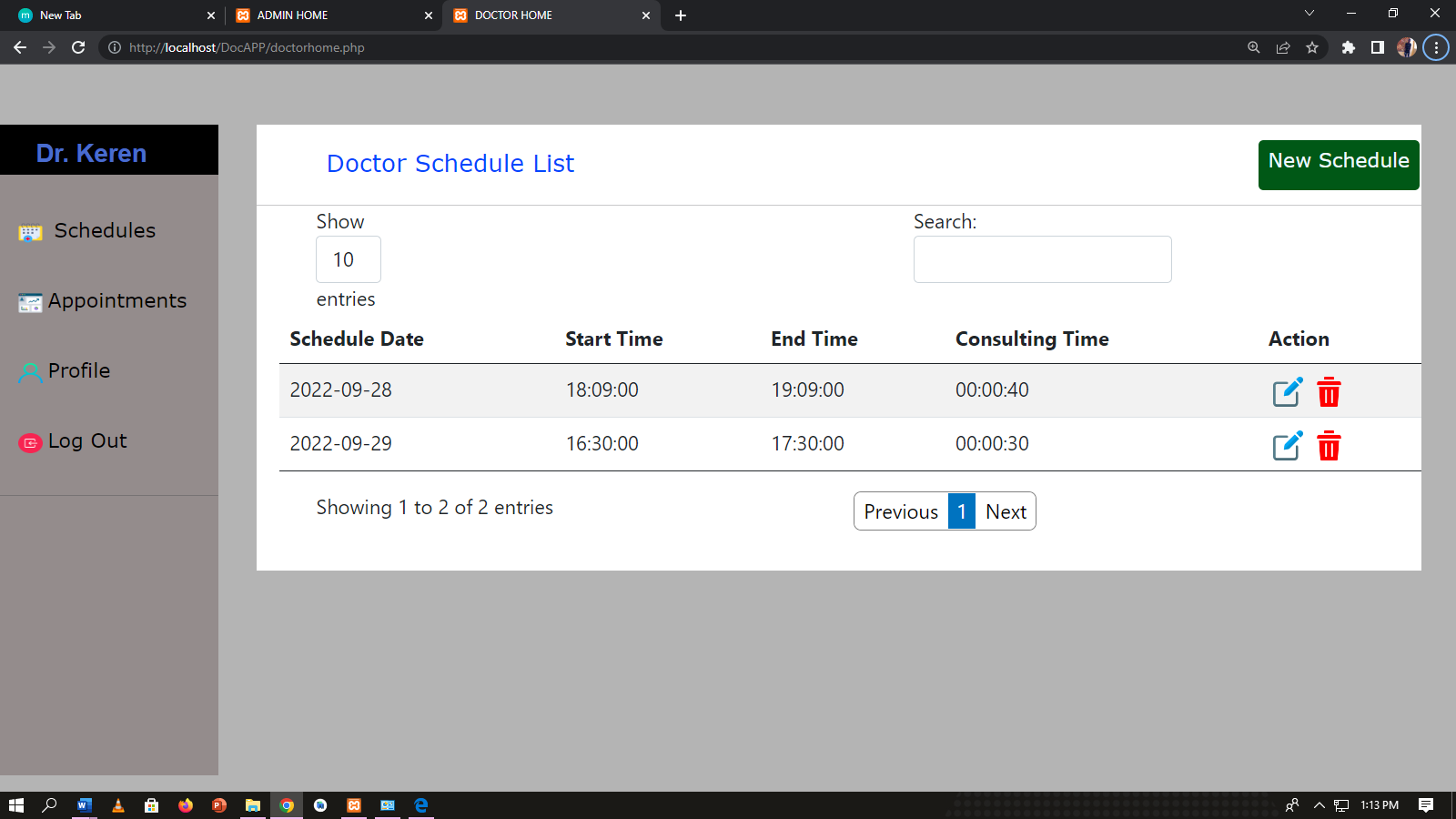


Figure 21:Doctor user view schedules list.

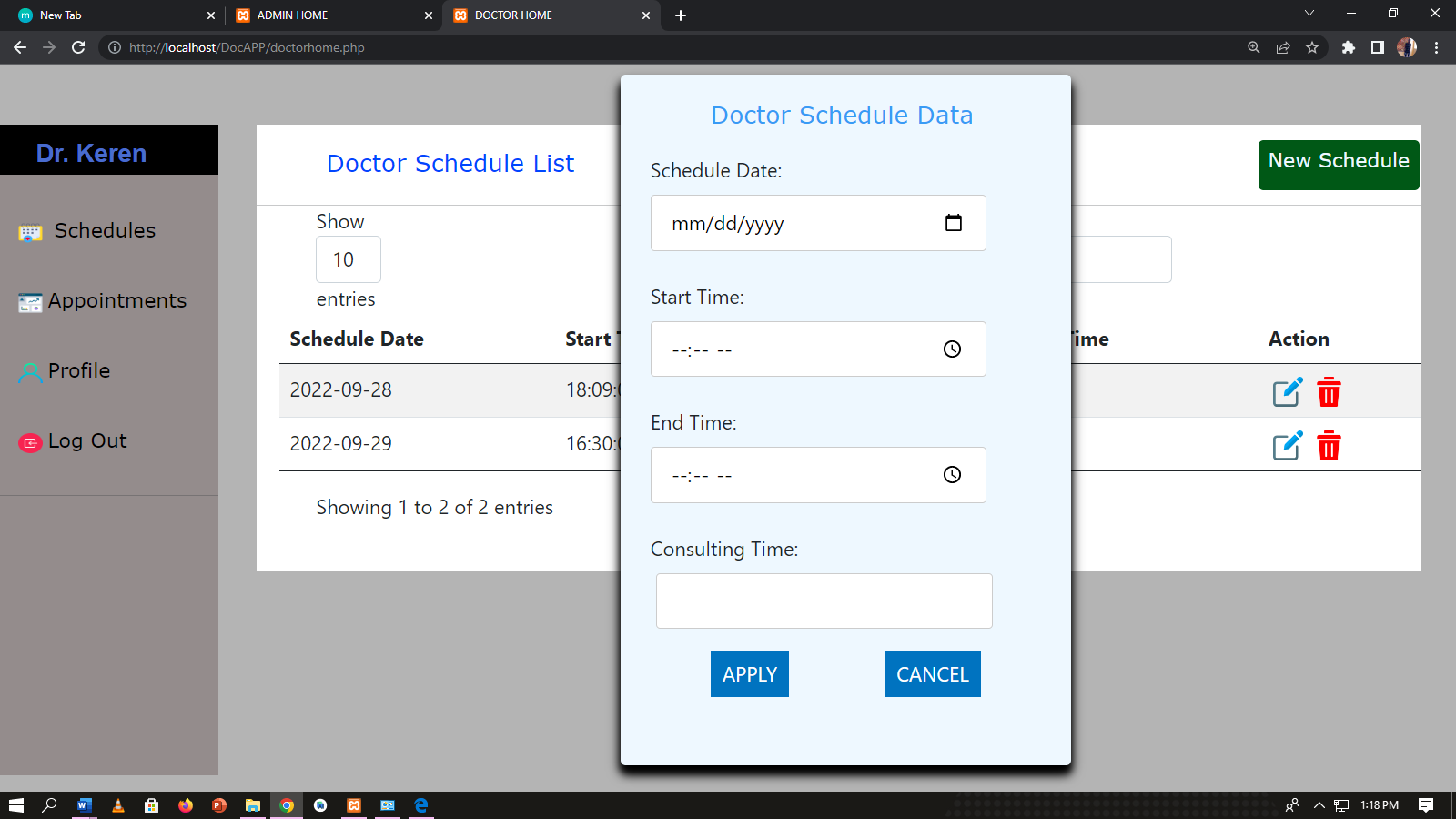


Figure 22:Doctor user schedule data entry form.

## **4.2.4 Patient Module**

This page displays a menu which what the patient can do with the system. He or she can view doctor schedule list and book an appointment and can also view appointment history and profile.

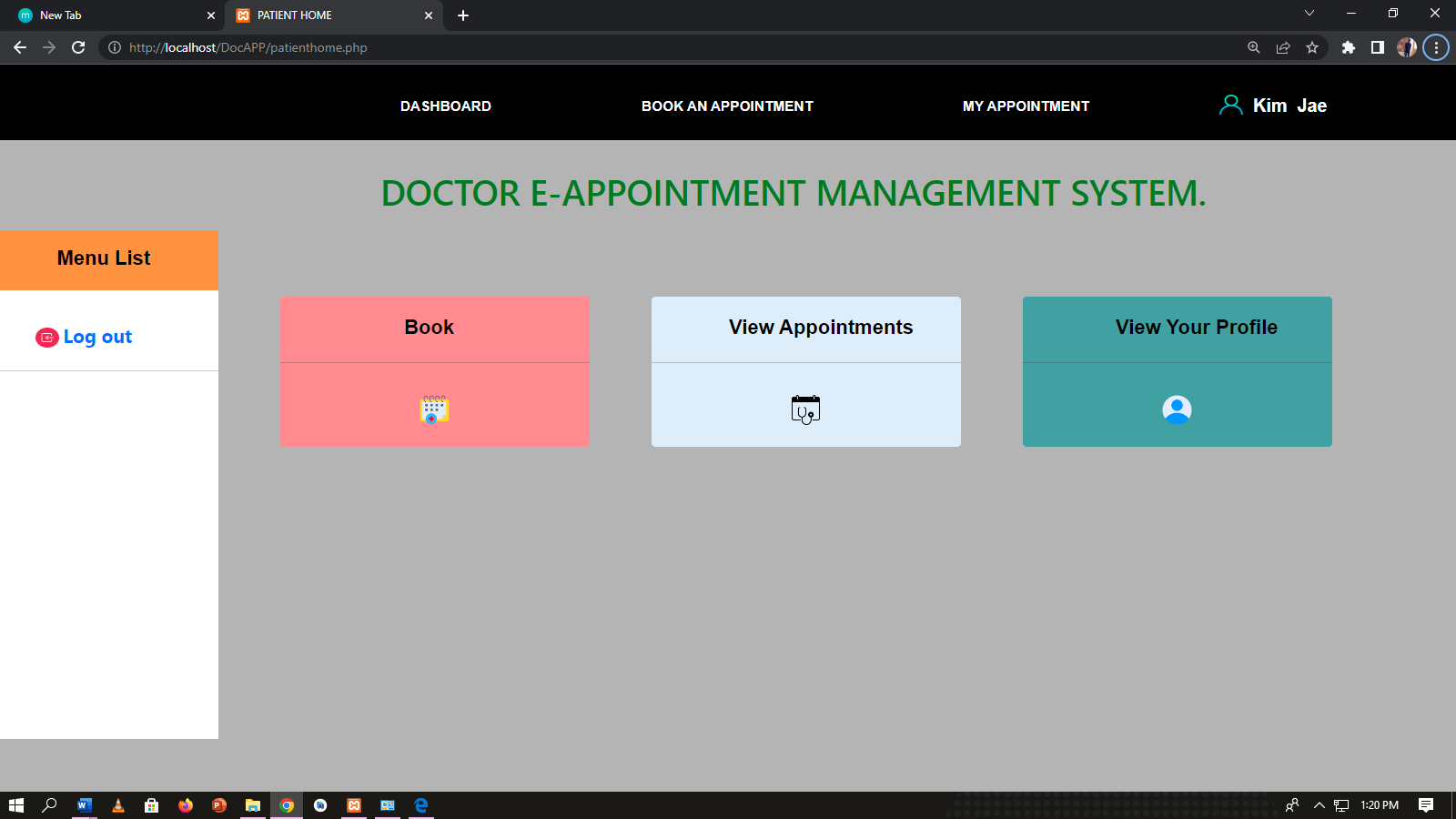


Figure 23:Patient Dashboard.

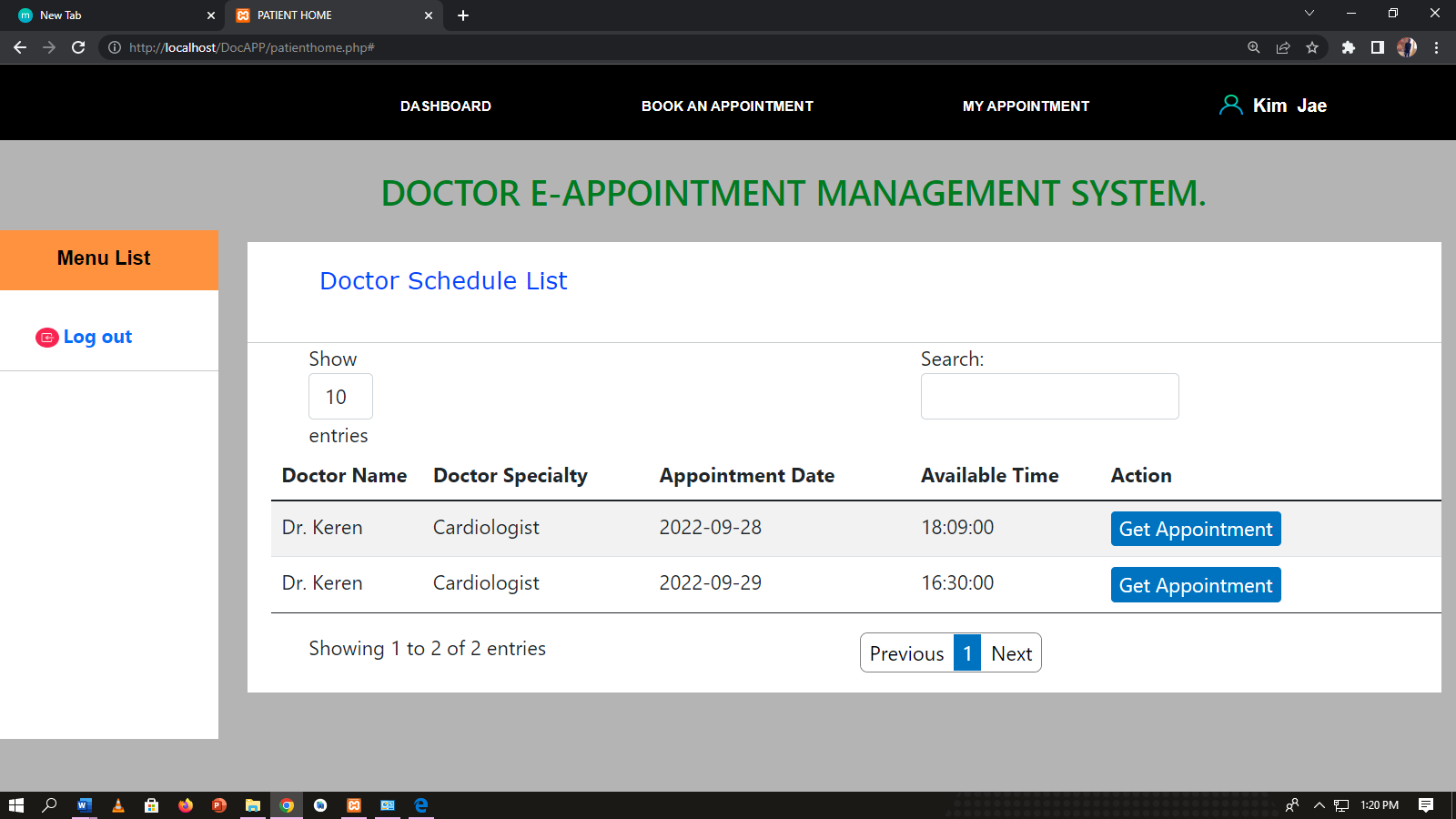


Figure 24:Patient view of doctor schedules .

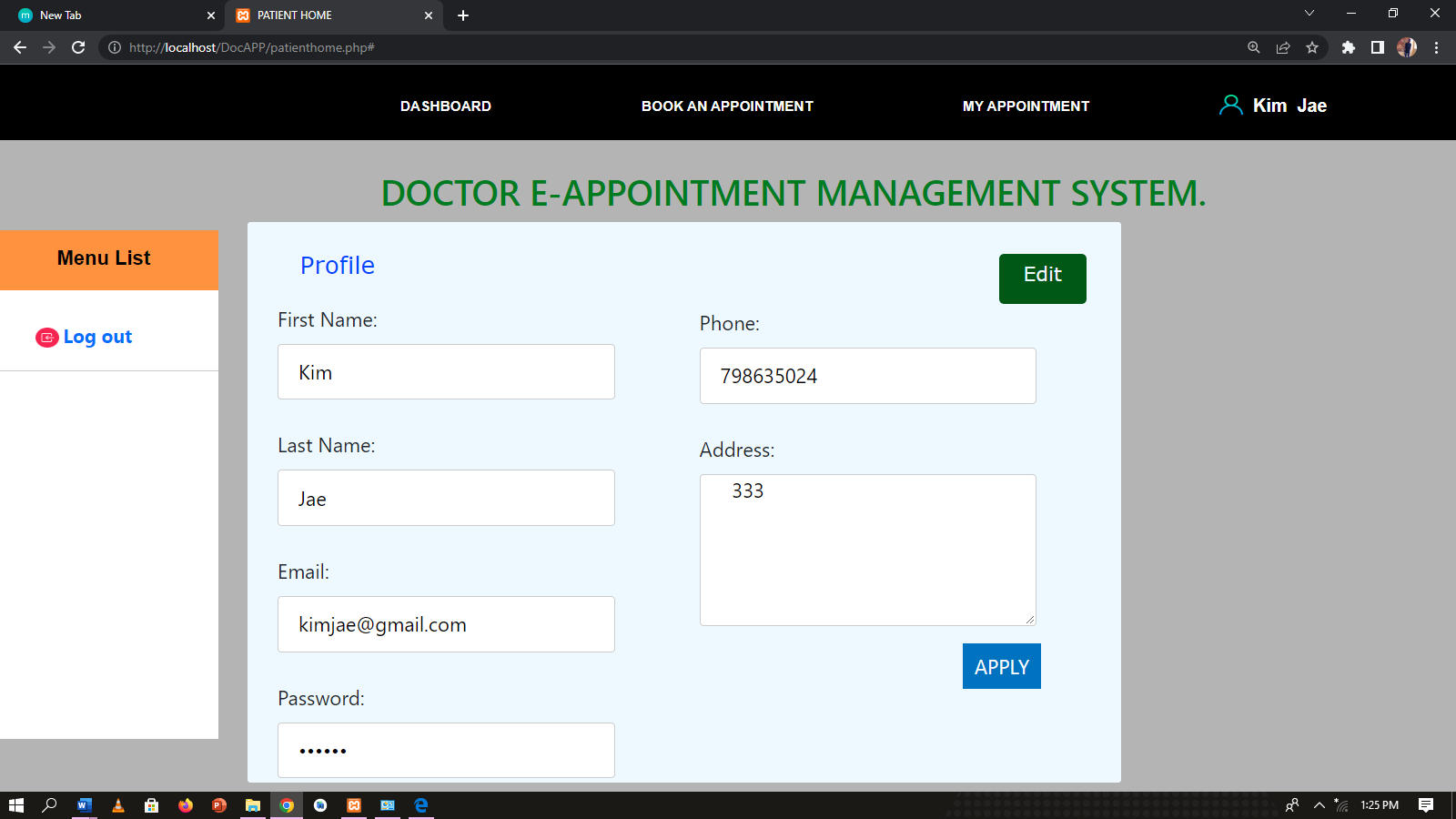


Figure 25:Patient user profile.

### 4.2.5 Report screenshots

This is the report generation in the database (MYSQL).

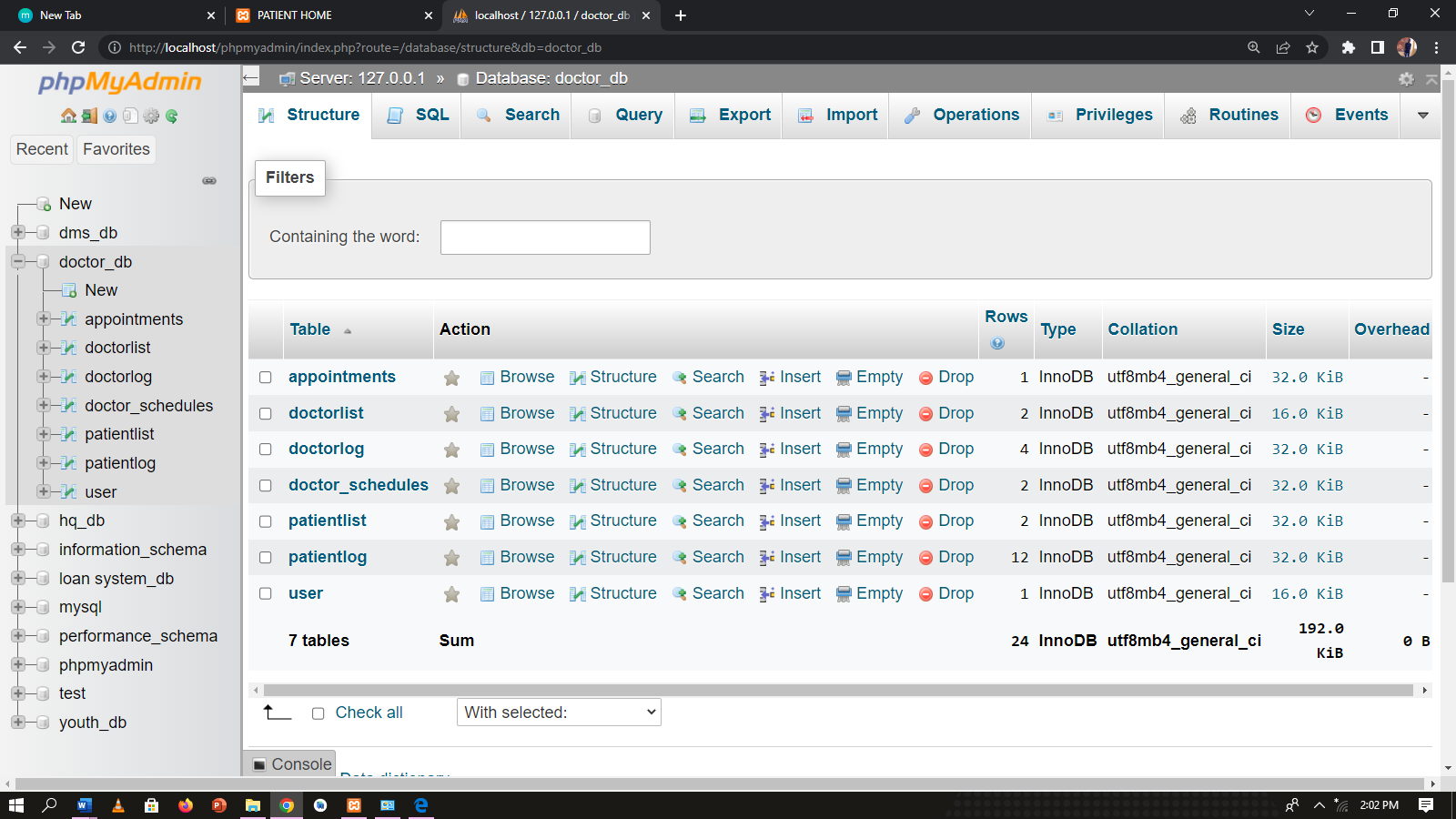


Figure 26:Reports screenshot database.

## **4.3 Testing**

Testing the system is very important to determine the success of the system. System testing determines whether the modules of the system is correct and achieving the user requirements in accordance to the objectives outlined. Testing of the system involves using the test data in the test phase. Debugging and testing was carried out in two consecutive phases. First phase was composed of Unit testing of major modules of the system. Lastly the User Acceptance Test was conducted whereby four volunteers were chosen to conduct User Acceptance Testing. The main purpose of test plan is to determine and build detailed test cases of each aspect of the application as provided by the users of the system.

## **4.3.1** **User Acceptance Testing**

Users test the system if it can handle real world scenarios. This is in accordance to the user’s specifications outlined in the objectives to ensure the expected results are met. The login module was tested and implemented using username and password. The other modules are as shown below.

# **Table 10: User Acceptance Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type of Testing** | **Module** | **Objective** | **Expectation** | **Outcome** |
| Unit testing | Add different types of user doctors and patients | To check if the administrator can add different users. | Authorized users should be able to login to the system and do their roles | Successful |
| Unit testing | View users | To check if the admin can successfully view and delete users from the database | Admin should be to add, view and delete users. | Successful |
| Unit testing | List of patients and doctors | To check if the administrator can view tenants and the houses they occupy | they should be able to get and view doctors and their schedules and patient data | Successful |

|  |  |  |  |
| --- | --- | --- | --- |
| Login | Log in the system should be done by the registered persons for the users and their correct passwords  as the administrator | Login should be successful and the user should be able to enter into the system | Fail  Pass |
| users(admin) should log in with the assigned login details as provided by the administrator | Log in should fail with an error message  “Incorrect Username |  |
|  | The system gives an error and deny login. | /Password” |  |
| User (doctor and patient) | Log in allows the user to get to the users’ dashboard | Login successfully user gets to dashboard | Pass |
| Administrator | Log in allows administrator get to admin dashboard | Log in successfully and admin gets to admin dashboard | Pass |

# **CHAPTER FIVE: SUMMARY OF FINDINGS, CONCLUSIONS AND RECOMMENDATIONS**

## 

## **5.1 Introduction**

This chapter summarizes the finding of the study and makes conclusions upon which recommendations are drawn. Suggestion for further study is also captured as a way of filling the gaps identified in the study. The chapter also summarizes the findings of the current method of the system and whether to adopt the use of the online platform for registration for swiftness of registration of as many students as possible. Also, to improve security of users’ credentials.

## **5.2 Summary of the Achievements**

The system developer was able to make the system in the following parts:

1.Saving registered patients and doctors data

2.Through the system there exist an aspect of integrity of data saved online

3.Saved on time used to review case files and data recorded online

4.Safety of data and can be extracted for future reference

## 

## **5.3 System Constraints**

Implementing an A.I system that could easily capture intruders who maliciously access doctors and patients accounts. This is a major drawback.

## **5.4 Conclusions**

In regard to the above discussion, it is noted that the doctor e-appointment management system has met the user requirements and the objectives stated. The main objective of implementing this system is to automate the appointment booking process without any physical human interaction. Considering the critical nature of the subject area, the government of Kenya should be in the forefront to spear head the innovations made in building of the systems. Currently as it stands that the number of cases where long queues are experienced in the hospital making the patients even go back home without treatment . This therefore implies that at least all health institutions should adopt use of doctor e-appointment management system for the appointment booking as it makes work easier and faster.

## **5.5 Recommendations**

It is known that for any meaningful computer-based information management system, should be integrated into any organization. Therefore, proper training and orientation has to be given both to the doctors and management. Proper Training should be given to data entry staff on how to handle the software and hardware part of the machine during back up processes. The electronic storage media should be handled properly or else vital data might be lost. The scope of the study being vast for future implementation. Therefore, the system can be updated in the future considering new user requirements at the time. The system is flexible and can be easily expanded. The following can be recommended for the future scope of the project.

1.In the future, the security of the system should be considered being that the system is at risk from intruders which may make the system very vital to hacking which can lead to mistrust from the administrators.

2.The system functionalities may be increased in the future to increase the efficiency of the system. For instance, coming up with complex algorithms to ensure the patients are able to use the system efficiently.

# 

# REFERENCES

1.Armstrong, M., Grossi, S., & Sauter, K. (2014, April). online booking systems

2. Habibi, Tabesh, H., Vakili-Arki, H., Abu-Hanna, A., & Eslami, S. (2019). Effect of an online appointment scheduling system on evaluation metrics of outpatient scheduling system: a before-after multicenter study. Journal of medical systems, *43*(8), 1-9.

3. Garg, A. X., Adhikari, N. K., McDonald, H., Rosas-Arellano, M. P., Devereaux, P. J., Beyene, J., ... & Haynes, R. B. (2005). Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. *Jama*, *293*(10), 1223-1238.

4. Kroenke, 2002,Gupta, D., & Denton, B. ,2008. Appointment scheduling in health care: Challenges and opportunities.

5.Armstrong, Grossi, & Sauter, 2014 online booking system.

6. Djalali, S., Ursprung, N., Rosemann, T., Senn, O., & Tandjung, R. (2015). Undirected health IT implementation in ambulatory care favors paper-based workarounds and limits health data exchange. *International journal of medical informatics*, *84*(11), 920-932.

7. James, T., Michael Reed, W., Wells, J. G., & Dombrowski, L. A. (1999). The effects of online multimedia project development, learning style. *Journal of Research on Computing in Education*, *31*(4), 341-355.

8. Koller, M., Grutter, R., Peltenburg, M., Fischer, J. E., & Steurer, J. (2001). Use of the Internet by medical doctors in Switzerland. *Swiss Medical Weekly*, *131*(17-18), 251.

## **APPENDICES:**

## **APPENDIX A: BUDGET.**

|  |  |
| --- | --- |
| **ITEMS** | **COSTS** |
| Laptop computer | KSHS. 40000 |
| Flash disk-64 | KSHS. 1000 |
| Microsoft windows 10 PRO operating system | KSHS. 10000 |
| Microsoft office word 2019 | KSHS. 5000 |
| Microsoft office access 2019 | KSHS. 5000 |
| Printing | KSHS. 4000 |
| Miscellaneous | KSHS. 5000 |
| **TOTAL** | **KSHS.7 0 000** |

## **APPENDIX B: TIMEFRAME**

The project involved all the activities involved in the Software Development Lifecycle. These activities are summarized in the Gantt chart below:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Weeks activity | **May** | **June** | **July** | **Aug** | **Sep** | **Oct** | **Nov** | **Dec** |
| Problem definition |  |  |  |  |  |  |  |  |
| Requirement identification |  |  |  |  |  |  |  |  |
| Analysis |  |  |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |  |  |
| Implementation  Testing |  |  |  |  |  |  |  |  |
| Documenting |  |  |  |  |  |  |  |  |

Figure 3.12 Timeframe

## **APPENDIX C: SOURCE CODE**

<!--beginning of html-->

<!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>ADMIN  HOME</title>

    <link rel="stylesheet"href="style/doc.css?V=<?php echo time();?>">

    <script src="js/jquery.js"></script>

    <script src="js/dt.js"></script>

    <script src="js/dtbt.js"></script>

    <link rel="stylesheet" href="style/bootstrap.min.css">

</head>

<body style="background-color: rgb(210, 210, 210);">

<div class="logo">

    <div class="a">

      <img src="images/im.png">

    </div>

<div class="b">

        <h2>Admin|Doctor.</h2</div

</div>

  <div id="sidebar">

   <ul class="Menu">

        <li id="btn-home"><img src="images/dash.png"> <a href="#">Dashboard</a> </li><hr>

          <li id="btn-doctors"><img src="images/doc.png "><a href="#">Doctors</a> </li><hr>

              <li id="btn-patients"><img src="images/pat.png"><a href="#">Patients</a> </li><hr>

                            <li id="btn-schedules"><img src="images/pp.png"><a href="#">Schedules</a> </li><hr>

                            <li id="btn-appointments"><img src="images/mm.png"><a href="#">Appointments</a> </li><hr>

                            <li id="btn-profiles"><img src="images/user.png"><a href="#">Profile</a> </li><hr>

                            <li id="btn-log"><img src="images/logout.png"><a href="index.php">Log Out</a></ul>

     </div><div id="panel-home"> <div class="panel-box">

      <div class="cardH"><div class="cardbody">Welcome Back Admin!</div>

</div>

<hr>

<div class="doc-col-b">

<div class="doc-card-panel">

    <h2>Doctors</h2>

  </div><div class="message"id="msg1"> <p>View Details</p>

  </div>

</div>

<div class="doc-col-c">

<div class="doc-card-panel">

    <h2>Registered Patients</h2>

  </div>

<div class="message"id="msg2">

    <p>View Patients</p>

  </div>

</div>

<div class="doc-col-d">

<div class="doc-card-panel">

    <h2>Schedules</h2>

  </div> <div class="message"id="msg3">

    <p> View Schedules </p>

  </div>

</div><div class="doc-col-a">

  <div class="doc-card-panel">

    <h2>Appointments</h2>

  </div>

  <div class="message"id="msg4">

    <p>View List</p>

  </div>

</div> </div>

      </div>

      <!--doctor start-->

      <div id="doctors">

          <div class="contain-box">

                <div class="doc-col-1">

              <div class="doc-card"><div id="app-form"><div class="app-form-body"> <label >Add Doctor Details</label>

               </div> <form action="adminhome.php"method="POST">

                 <div class="a">

  <label for="name">Name\*</label><br>

  <input type="text" id="name" name="txt-name"><br>

  <label for="name"> Password\*</label><br>

  <input type="text" id="lname" name="txt-passward"><br>

  <label>Speciality\*</label><br>

  <textarea cols="40"rows="5"placholder="Enter Purpose" name="txt-speciality"></textarea>

  </div>

  <div class="b">

  <label for="name">  Email\*</label><br>

  <input type="text" id="lname"  name="txt-email"><br>

  <label for="name"> Phone\*</label><br>

  <input type="text" id="lname"  name="txt-phone"><br>

  <label for="name"> Doctor Id\*</label><br>

  <input type="text" id="lname"  name="txt-id"><br>

  <ul><li><input type="submit"value="APPLY"name="btnapplydoc"></li>

    <li><input type="button"value="CLOSE"id="btn-cancel-application"></li>

  </ul>

  </div>

</form></div>

               <div class="doc-card-l">

                   <div class="doc-card-body-l">Doctor List</div>

                    <div class=" btn btn-success"id="btn-apply"> New Doctor</div>

                 </div><hr>

<table id="doc-table" class="table table-striped" style="width:99%;margin-top:20px;margin-left:2%;">

<thead>

            <tr> <th>Id</th>

                <th>Doctor Id</th>

                <th>Doctor Email</th>

                <th>Name</th>

                <th>Phone</th>

                <th>Speciality</th>

                <th>Action</th>

            </tr>

        </thead>

        <tbody>

  <tr>

    <td

    <td><?php echo $row['speciality'];?></td>

  <td><img src="images/DE.png">

   <img src="images/D.png" class="deletedoc">

    </td>

  </tr>

<?php

<td></td><td></td><td></td> <td></td>

        <td></td>

        <td></td>

        <td></td>

        <td></td>

       </tr>

        </tbody>

            <tfoot>

            <tr>

            </tr>

</tfoot>

</ta</div>

           </div>

           </div>

           </div>

                 </div><!--doctor end-->

               <div id="patients">

         <div class="contain-boxP">

        <div id="app-form-patient">

            <div class="app-form-patient-body"> <label > Patient Details</label>

            </div>

            <hr>

            <form action="adminhome.php"method="POST">

                 <div class="a">

  <label for="name">First Name:</label><br>

  <input type="text" id="name" name="txt-fname"><br>

  <label for="name">Last Name:</label><br>

  <input type="text" id="lname" name="txt-lname"><br>

  <label for="name">Email:</label><br>

  <input type="text" id="lname"  name="txt-email"><br>

  <label for="name">Patient Id:</label><br>

  <input type="text" id="lname"  name="txt-id"><br>

</div>

</body>

</html>