“Ayur” Patient Management and Medicine Stock Management System for Peramuna Ayurveda Medical Center

A dissertation submitted for the Degree of Master of Information Technology

K.R. Kaluarachchi

University of Colombo School of Computing

2019

ABSTRACT

The developed “Ayur Patient Management and Medicine stock Management” web and mobile system for Peramuna Ayurvedic Center is helpful to overcome the problems and difficulties which were created by the current manual system. This system is helpful to automatically execute the appointment process, doctor prescription handling, pharmacist and cashier function handling, keep track of the past patient diagnosis and treatments, sent SMS/Email alerts and notifications to users, Online chatting service for the patients, track and monitor monthly income, trending treatments, etc. , store all information and Generate reports. Also, this system facilities to maintain user levels as well. Authorized personnel will be allowed to execute the relevant system functions.

The Developed system has been overcome the difficulties of appointment handling, even though the mobile application has been made easier. The Patient will be notified with all the details such as reminders when the particular patient’s number is the next one. Doctors can handle all the patient prescriptions and past records online without any bargain. The pharmacist will be notified when the medicine stocks need to refill and expirations of medicine. Also, they can easily handle medicine issuing via the system. In an instance, the cashier will be print the invoices and easily do the payments.

The system was implemented as a web and mobile-based system by Java 10 with Spring Boot framework and IntelliJ Idea for the web and Android Studio with Android APIs level 24 for the mobile. Mongo DB will be used to handle databases and Apache Tomcat 9.0.1 will be used as the web server.

The project completed successfully with the requested functionalities resulting in the Peramuna Ayurvedic Center to manage their business effectively by overcoming the difficulties made by the manual system.

DECLARATION

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: Kasuni Rashmika Kaluarachchi

Registration Number: 2016MIT028

Index Number: 16550282

Signature: Date:

This is to certify that this thesis is based on the work of

Prof. G.K.A. Dias

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Prof. G.K.A. Dias

Signature: Date:

ACKNOWLEDGEMENT

I take this opportunity to express my gratitude to those who helped me throughout this research and guided me to successful completion.

I would first like to thank my thesis advisor Prof. G.K.A. Dias for all the help he provided me with completing this thesis. He consistently allowed this dissertation to be my own work but steered me in the right direction whenever he thought I needed it.

Also, I like to thank Dr. Shyama Peramauna (Owner/Chief Doctor – Peramuna Ayurvedic Center) for providing me with the necessary information and allow me to monitor the current workflow. My sincere thanks to all the staff members who helped me throughout the process.

Finally, I must express my very profound gratitude to my parents and to my husband and my family for providing me with unfailing support and continuous encouragement throughout my years of study and through the out this project implementing and writing this dissertation.

This accomplishment would not have been possible without them. Thank you.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AUT	Application Under Test
API	Application Programming Interface
CPU	Central Processing Unit
DB	Database
ERD	Entity Relationship Diagram
GB	Gigabyte
GHZ	Gigahertz
JDK	Java Development Kit
LAN	National Identity Card
MVC	Model-View-Controller
MIT	Master of Information Technology
PC	Personal Computer
RAM	Random Access Memory
SMS	Short Message Service
URL	Local Area Network

CHAPTER 01: INTRODUCTION

1.1.Problem Statement

Ayurveda medicine has been originated in Sri Lanka many years ago. It is expanding in Sri Lanka since then rapidly. It is one of the most ancient forms of Sri Lanka and still practicing in here. [1] Currently many more people are into Ayurveda medicines than western medicines. “Peramuna Ayurveda Medical Center” is an ayurvedic wellness medical center, provides treatments like Ayurvedic Treatments, Panchakarma [2], Beauty Treatments, Skin & Hair Care Treatments, etc. [3] They have a team of experienced & qualified doctors to examine patients. Their main objective is to increase its customer base within a short period and provide all their services efficiency.

Currently, they handle all their reservations via telephone and note down each and every reservation in a book. They serve a large number of patients daily and it is very difficult to maintain reservation details, customer details, payment details, medicine details, etc. manually. It wastes a lot of time and requires them to do a lot of manual paperwork. Also, they have faced many difficulties with tracking past records. Patients also had to face many difficulties during taking an appointment and had to wait a long time to take their medicines because of the manual system. Also, doctors faced with many difficulties in tracking patient history records, previous prescription details, etc. To overcome all these issues and difficulties, it was proposed to implement a fully computerized software system, which will minimize the paperwork and reduce the waiting time for the patients and help to reduce the hazel with using a manual paper-based system providing easy data access, history tracking. Also, it will improve data security as well.

1.2.Motivation

Ayurvedic medicine sector is one of the most popular medical industries in Sri Lanka. It has a unique process and most of the Ayurveda centers do not use fully computerized systems. Since Sri Lankan Ayurvedic medicine is famous among foreigners and also it will be easy if they can use a system to handle their needs.

At the moment “Peramuna Ayurveda Medical Center” is using a manual system to handle the overall process and they found difficulties in that. Since they cater local as well as foreign customers in different social status, people are not satisfied with the current system handling. Most of the people use computers as well as mobile phones to do their day to day activities. The main motivation of developing a computer/mobile based system is to increase the customer base and modified the current process in a friendly manner.

The system will provide many services and information based on customer needs since customers will be happy to receive their treatments without wasting their time and they will encourage to use the center’s services rapidly. Ayurveda center can plan and manage their information, records accurately and can be accessed anytime. That will motivate the doctors as well as patients to use the system rapidly [4] [5] [6].

1.3.Objectives

The main objective of this project is to implement a computer-based solution to manage the overall process of the Peramuna Ayurveda Medical Center and improve the efficiency of the day to day tasks. The following can be identified as the main objectives of the proposed system.

- Implement a system that can handle patients and medicine stocks management process.
- Analyze the current process and potential capability to expand the patient base.
- Schedule all the bookings online and improve the channeling facilities.
- Facilitate to generate different reports for analytical decision-making purposes.
- Improve medicine stock handling by notifying about out of stock medicines.
- Implement a process for prescription handling and reduce the paperwork to be done.
- Implement a payment process to handle patient payments efficiently.

1.4.Methodology

The software development life cycle model that I am going to adopt is agile methodology [7]. In this methodology, it uses incremental, iterative work sequences that are commonly known as sprints. The proposed system will be deployed as a web-based system with a mobile application. Because this particular application can be accessed via the internet for the users anywhere in the world. The main concern of implementing this system is to maximize the usability, so via mobile application, any user can access the application so easily. The architecture of the proposed system will be Model-View-Controller (MVC). It supports parallel and rapid development and also supports the development of SEO friendly web pages and applications.

Java 10 with Spring Boot framework [8] [9] and IntelliJ Idea will be used to develop the web application and Android Studio [10] with android APIs level 24 will be used to develop the mobile application. Mongo DB will be used to handle databases. Apache Tomcat 9.0.1 will be used as the web server.

1.5.Scope of Project

Scope of the system is described as below;

- Manage patient details and doctor details.
    1. The system will facilitate patients to search treatment details and give suggestions (treatments, doctors, etc.) before the booking. Also, have the facility to view comments regarding previous patient treatments and allow to online chat regarding any information.
    2. All the patients need to be registered to the system upon appointment booking with a unique patient ID and the medical center will be able to track the patient's details through the system.
    3. Maintain a calendar for long term treatments.
    4. All the Doctors details need to register to the system and patient will be enabled to view doctor details upon searching booking details.

- Handle online patients' appointments
    1. Upon booking a number for a particular doctor, the patient will be notified via SMS/Email.
    2. The doctor will be able to view daily reservation details.
    3. The system will send reminders via SMS/Email regarding appointments to patients.
    4. The system will allow the patient to check doctor availability and track appointment details.
    5. The system will store appointment details.

- Facilitate all prescription details to be computerized.
    1. The system will allow doctors to enter the prescription details of a particular patient and send those details to the pharmacy. Upon prescription details entering, the system will notify if the medicines are out of stock.
    2. The system will allow printing the prescriptions that the patient needs to buy from outside medicine shops (each medicine will be printed with an image for identification purposes).
    3. The system will allow patients to search for medicine shops for particular medicines.
    4. The system will allow doctors to track previous prescription details of patients.

- Handle all patient payment process.

    1. The system will handle doctor channeling payments online. Upon payment medical center and the patient will be notified regarding payment and appointment details via email.
    2. The system will automatically calculate and save the payment amount upon medicine issuing.
    3. The system will allow receipt printing facility and store the payment details.

- Maintain medicine stock details.
    1. The system will notify on medicine out of stock/expire.
    2. The system will facilitate to track all medicine stock details.

- Facilitate to generate accurate reports for analysis purposes.
    1. Current patient base
    2. Appointment details (daily, monthly, yearly)
    3. Treatment details – Can be analyzed the trending treatments and buy more medicines for those.
    4. Patient payment details - (daily, monthly, yearly)
    5. Medicine stocks - Can be analyzed the most issuing medicines and buy more.
    6. Patient diagnosis history – If a regular patient or long-term patient, doctors can be analyzed patient illnesses and make decisions easily.

1.6.Chapter Break Down

The dissertation is the document that contains overall information of the project in chapter wise. This dissertation contains six main chapters followed by reference and appendix.

Chapter 01: Introduction

The Introduction chapter describes the project problem and it justifies the study, summarizes the objectives, scope and includes the information to motivate the reader. Also, it includes a chapter breakdown of the dissertation.

Chapter 02: Background

The Background chapter describes the background of the project that is going to be implemented. Also, it contains an analysis of similar systems, tool, and technologies which will be used for the implementation.

Chapter 03: Methodology

All the functional, non- functional requirements, information gathering techniques, methodologies and tools which are used to design the system will be included in this chapter. For the system designing, User interfaces, database structures, information diagrams will be designed. And also this chapter describes how specifications were gathered (in analysis and design phases) are converted into executable program. Furthermore, chapter incorporates with Implementing technologies and concepts.

Chapter 04: Evaluation

This chapter describes how the system was tested by using various testing methods. It also reported errors and how to overcome those errors and how to modify the system.

Chapter 05: Conclusion

This chapter includes the all summarize details of the projects and also includes findings and lessons learned during the project with further improvements of the project.

.

CHAPTER 02: BACKGROUND

2.1.Background

“Peramuna Ayurveda Medical Center” is well known ayurvedic center in Kottawa area which is functioning around 10 years. Since the start, day by day their customer base is increasing. This ayurvedic center provides many ayurvedic medical treatments. Currently, a few numbers of doctors examine patients and provide treatments here and they issued medicines as well. They conduct a pharmacy to provide medicines for the customers as well. Since they issued many medicines daily, they need a proper medicine stock handling process and as well as a patient management process. Therefore, this system will implement to determine both patient and medicine management of the ayurvedic center.

Many techniques have been conducted to gather background requirements of the current system as below;

1. Work environment Observation
2. Study existing documents (Inventory books, prescriptions, payment bills, medicine log books)
3. Interviews with personals (Doctors, cashier, pharmacist)

Study existing documentation and observation of the work environment were used as the primary fact-gathering techniques. Then interviewed the people who are related to the main process to gather specific information about the system and to find out their expectations from the proposed system.

The existing system mainly consists of four subsystems.

Refer the ‘Figure 2.1’ for the basic use case diagram of the existing subsystems.

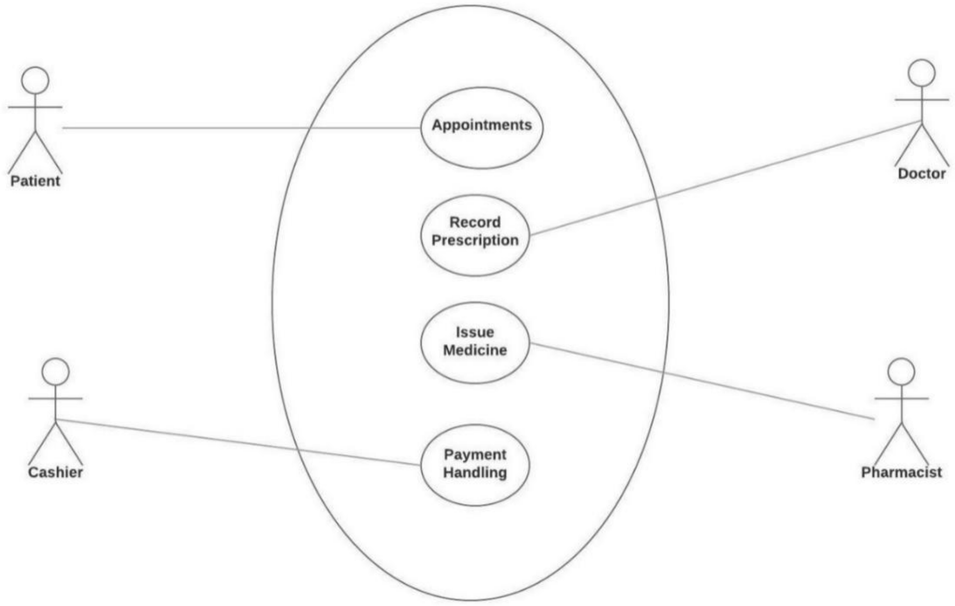


Figure 2.1 Use case Diagram of Existing Sub Systems

2.2.Functional Requirements

The following requirements were identified as the functional requirements of the proposed system. The proposed system should have three four levels with the ability of do the following functionalities. The five user levels are ‘Patient’, ‘Doctor’, ‘Cashier’, ‘Pharmacist’ and System Administrator. Functional requirements identified for each user level can be described as follows;

Patient

- Search treatment details
- Online chatting regarding information
- Schedule Appointments
- View Doctor details
- Check doctor availability
- Track appointment details
- Search medicine shops

Doctor

- Calendar maintain for long term treatments
- Enter prescription details
- Send prescription details
- View daily reservation details
- Track previous prescription details
- View reports
- Print reports

Cashier

- Manage Online Appointments
- Payment Handling
- View Reports
- Print Reports

Pharmacist

- View prescription details
- View Reports
- Print Reports

System Administrator

- User Management
- Maintain Patient and doctor details

System Functions

- Notify patient appointments via SMS/Email
- Notify medicine stocks availability
- Send appointment reminders
- Print prescription
- Handle doctor channeling payments
- Notify payment details
- Manage payments

- Report Generation

Common Functions

- User Login
- Change password
- System Logout

2.3.Non-Functional Requirements

These are the requirements that specify criteria that can be used to judge the operation of the system, rather than specific behaviors. Non -Functional requirements can be described as follows;

- **User friendliness of the system** - The system should be easy to learn for the users, and the user interfaces should be simple easy to use.
- The system should be **Reliable**. The ability to make backups and restore them can be categorized under this requirement.
- The system should be **Accurate** and **Consistent** – Accuracy and consistency of the functions are essential non-functional requirements of the system
- **Security** – Only authorized users should be able to operate the system and the availability of different user levels protects confidential information.
- **Maintainability** - The system should be able to maintain in the future. That means functionalities could be added, removed or modified without affecting the core functions of the system. In order to achieve this, proper documentation of the system

2.4.Similar Systems Available in the Market

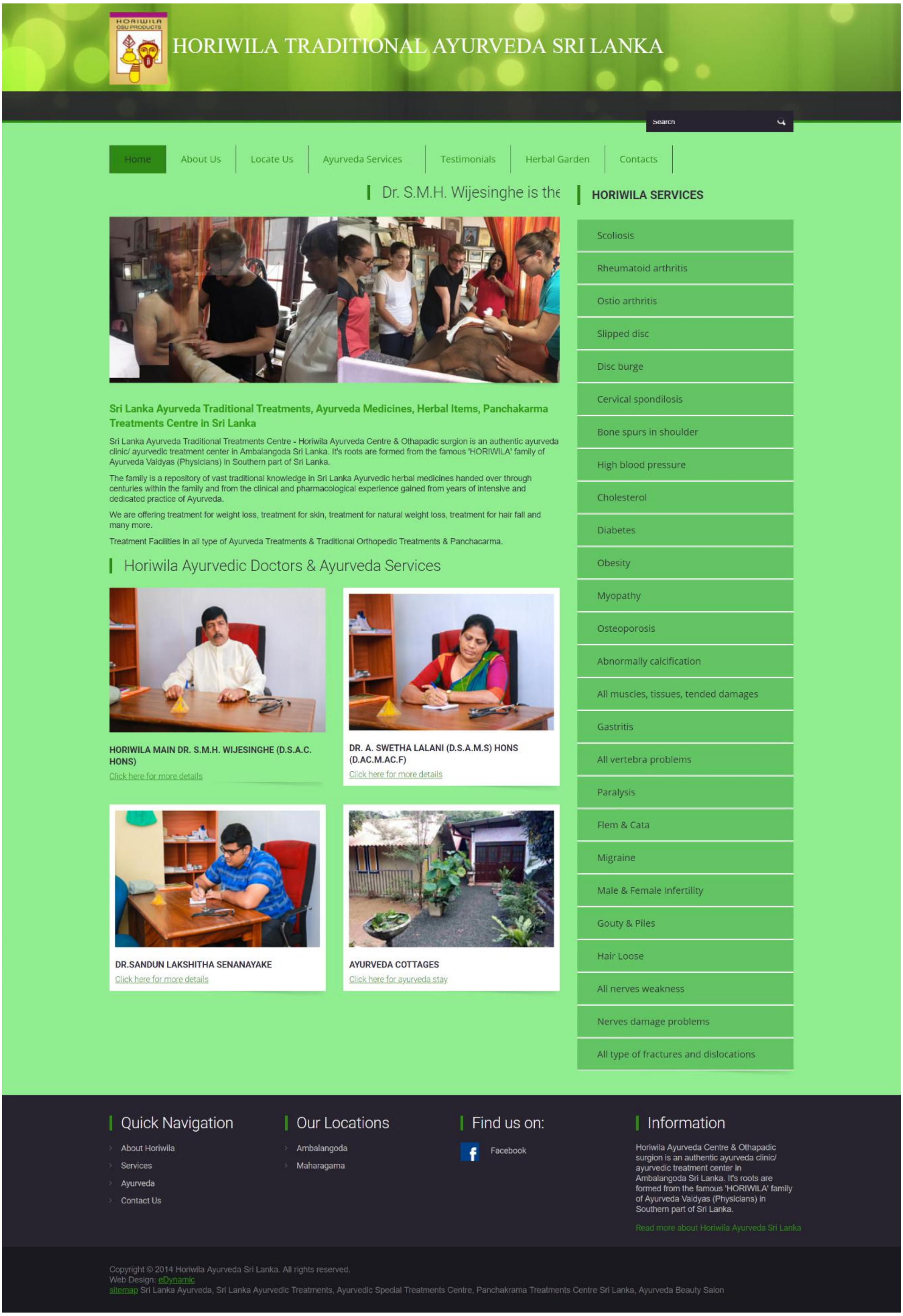
Few Systems [11] [12] in the market were analyzed to check whether they cater to the requirement of the Peramuna Ayurveda medical center.

- Horiwila Traditional Ayurveda Sri Lanka:

URL: http://www.horiwilaayurvedic.com/index.html

Horiwila Traditional Ayurveda is a web application which supports medical activities of the Horiwilla Ayurveda center. This a very basic web application that provides patients to inform information and increases the customer base. This application provides information on all the treatments and services they provide, Doctor Details and their background, Testimonials, Herbal garden picture gallery, and their contact details.

Figure 2.2 illustrates the main view of "Horiwila Traditional Ayurveda". Web application



HORIWILA TRADITIONAL AYURVEDA SRI LANKA

Dr. S.M.H. Wijesinghe is the

Sri Lanka Ayurveda Traditional Treatments, Ayurveda Medicines, Herbal Items, Panchakarma Treatments Centre in Sri Lanka

Sri Lanka Ayurveda Traditional Treatments Centre - Horiwila Ayurveda Centre & Othapadic surgion is an authentic ayurveda clinic/ ayurvedic treatment center in Ambalangoda Sri Lanka. It's roots are formed from the famous 'HORIWILA' family of Ayurveda Vaidyas (Physicians) in Southern part of Sri Lanka.

The family is a repository of vast traditional knowledge in Sri Lanka Ayurvedic herbal medicines handed over through centuries within the family and from the clinical and pharmacological experience gained from years of intensive and dedicated practice of Ayurveda.

We are offering treatment for weight loss, treatment for skin, treatment for natural weight loss, treatment for hair fall and many more.

Treatment Facilities in all type of Ayurveda Treatments & Traditional Orthopedic Treatments & Panchacarma.

Horiwila Ayurvedic Doctors & Ayurveda Services

HORIWILA MAIN DR. S.M.H. WIJESINGHE (D.S.A.C. HONS)
Click here for more details

DR. A. SWETHA LALANI (D.S.A.M.S) HONS (D.AC.M.AC.F)
Click here for more details

DR.SANDUN LAKSHITHA SENANAYAKE
Click here for more details

AYURVEDA COTTAGES
Click here for ayurveda stay

HORIWILA SERVICES

- Scoliosis
- Rheumatoid arthritis
- Ostio arthritis
- Slipped disc
- Disc burge
- Cervical spondilosis
- Bone spurs in shoulder
- High blood pressure
- Cholesterol
- Diabetes
- Obesity
- Myopathy
- Osteoporosis
- Abnormally calcification
- All muscles, tissues, tended damages
- Gastritis
- All vertebra problems
- Paralysis
- Flem & Cata
- Migraine
- Male & Female Infertility
- Gouty & Piles
- Hair Loose
- All nerves weakness
- Nerves damage problems
- All type of fractures and dislocations

Quick Navigation
- About Horiwila
- Services
- Ayurveda
- Contact Us

Our Locations
- Ambalangoda
- Maharagama

Find us on:
Facebook

Information

Horiwila Ayurveda Centre & Othapadic surgion is an authentic ayurveda clinic/ ayurvedic treatment center in Ambalangoda Sri Lanka. It's roots are formed from the famous 'HORIWILA' family of Ayurveda Vaidyas (Physicians) in Southern part of Sri Lanka.

Read more about Horiwila Ayurveda Sri Lanka

Copyright © 2014 Horiwila Ayurveda Sri Lanka. All rights reserved.
Web Design: eDynamic
sitemap Sri Lanka Ayurveda, Sri Lanka Ayurvedic Treatments, Ayurvedic Special Treatments Centre, Panchakrama Treatments Centre Sri Lanka, Ayurveda Beauty Salon

Figure 2.2 "Horiwila Traditional Ayurveda Sri Lanka" System

Drawbacks of Horiwila Traditional Ayurveda Sri Lanka system

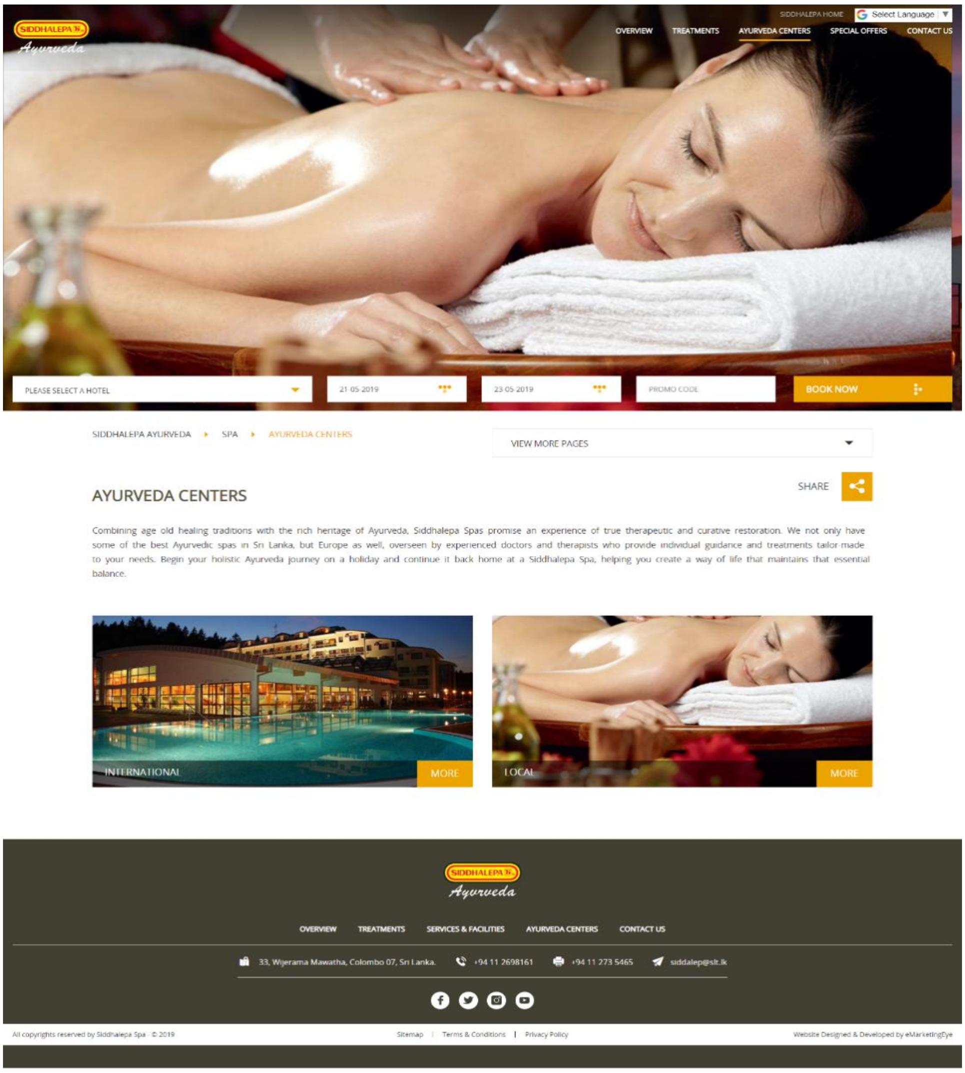
- This application didn’t cater to patient management handling and medicine stock handling.
- This can be only used as an information application.
- This didn’t cater to different user groups (user feasibility).

- Siddhalepa :

URL: http://www.siddhaleparesort.com/

Siddhalepa is a web application which supports medical activities of the Siddhalepa group booking facility. This web application also an information site that provides information about their background and services, their resorts in the hotels, Siddhalepa Ayurveda hospital, spas, their products and offers. This web application is mainly targeted on foreigners. Users can enter their booking dates and check availability. This is basically clients can book a package with accommodation in one of their hotels for the ayurvedic treatments and services. This application supports many different languages as well.

Figure 2.3 illustrates the main view of the “Siddhalepa” Web application.



21-05-2019

23-05-2019

AYURVEDA CENTERS

Combining age old healing traditions with the rich heritage of Ayurveda, Siddhalepa Spas promise an experience of true therapeutic and curative restoration. We not only have some of the best Ayurvedic spas in Sri Lanka, but Europe as well, overseen by experienced doctors and therapists who provide individual guidance and treatments tailor-made to your needs. Begin your holistic Ayurveda journey on a holiday and continue it back home at a Siddhalepa Spa, helping you create a way of life that maintains that essential balance.

33, Wijerama Mawatha, Colombo 07, Sri Lanka. +94 11 2698161 +94 11 273 5465 siddalep@slt.lk

All copyrights reserved by Siddhalepa Spa · © 2019

Website Designed & Developed by eMarketingEye

Figure 2.3 “Siddhalepa” System

Drawbacks of Siddhalepa system

- This application didn’t cater to patient management handling and medicine stock handling.
- This application is basically targeted by foreign customers.
- This did not cater day today casual local patients.

Every patient needs to pay for the whole package with accommodation.

2.5.Alternative Design Strategies

For software modeling, there are many models such as waterfall, iterative, spiral, etc. The waterfall method can be only used if the requirements are stable in a project. In here each phase must be completed before the next. For this project, I am going to use the agile model. It is a combination of iterative and incremental process. Break the working product into small incremental builds. This will be cost-effective way than waterfall. For the Software development free and open source software will be used as this would decrease the development cost. The client-server environment will be the hardware environment. For the system, software will be used GNU/Linux for mobile development. Since all these sources are free, the cost will be very low comparing to other alternative strategies.

CHAPTER 03: METHODOLOGY

3.1.Introduction

This chapter includes software analysis and design and implementation methodologies. Software analysis and design includes all the activities converting human requirements into the implementation. This include both low-level component and algorithm design and high-level, architecture design. All the requirements have been identified in the requirements gathering phase and convert them to design models in the design phase.

3.2.Alternative Design Solution Evaluation

Below mentioned are the best alternative technological solutions that can be identified to design the proposed system.

Option 1: Standalone System

Standalone software [13] is any software application that does not need any software package to run. That is a portable application that does not need to install on your computer. These types of systems are useful only if, the user does not require to access information through multiple computers. This software can create problems once it needs to integrate with other packages.

Option 2: Web-Based System

Web based system is a software package that can be accessed through the web browser. The software and database are residing on a central server. Users can access this type of system from anywhere in the world. Hence when developing a web-based system, security should be the main concern to protect the confidential information of the users.

Option 3: Network System

This refers to a system [14] which includes multiple computers and a common database server to access by each computer networked in the system. Separate instances of the software will be installed in each computer and multiple users will be able to access and exchange information using the common database. This type of system does not require an internet connection to communicate between the respective computers. All computers and servers will be interconnected via a Local area network (LAN).

Option 4: Mobile Based System

This is a type of software designed to run on a mobile device. Mobile applications mostly serve to provide users with similar features and services to those accessed on desktops. Apps are small, mostly individual software units with limited function.

Among the 4 options mentioned in the above, it was decided to select Option 2: Web

Based System and Option 4: Mobile Based System due to the following reasons;

- The system will be required to access by the patients, to schedule appointment. (Hence 'Option 1: Standalone System' and 'Option 3: Network System', were removed from the consideration)
- Web and mobile-based application gives access to business information from anywhere in the world at any time.
- Web and mobile-based application also facilitates you to save time & money and improve the interactivity with your customers.
- Web-based applications are very easy to use and also can be developed without interrupting your existing work process.
- Currently, with the rapid adoption of smartphones, the mobile application is the best solution for nowadays.

3.3.System Design Methodology

There are several system design methodologies such as waterfall incremental, iterative, agile, scrum [15] , extreme programming, etc. Here I have been selected agile methodology due to below reasons;

- Transparency – By using the agile methodology system complexity can be reduced.
- Collaborative Effort - Clients are obligated to actively participate in the requirements definition process, thus reducing unknowns and making the overall process more efficient.
- Project size is medium.
- The customer is available throughout the project.
- Customer requirements are not stable.

3.4.Design Diagrams

The following diagrams were designed based on the functional requirements.

- Use Case Diagram and respective use case descriptions
- Entity Relationship Diagram
- Database Design [16]
- User Interface Design

In this chapter use case diagrams, Entity relationship diagram, database design, and few user interfaces are added. Refer the 'Appendix A – Design Documentation' for other complete design diagrams.

3.4.1. Use Case Diagram

The user case diagram [17] used to demonstrate the user interactions with the system. These highlight scenarios of the systems, goals that the system help actors to achieve and the scope of the system.

The figures 3.1,3.2,3.3,3.4 and 3.5 shows the use case diagrams based on the user for the proposed system.

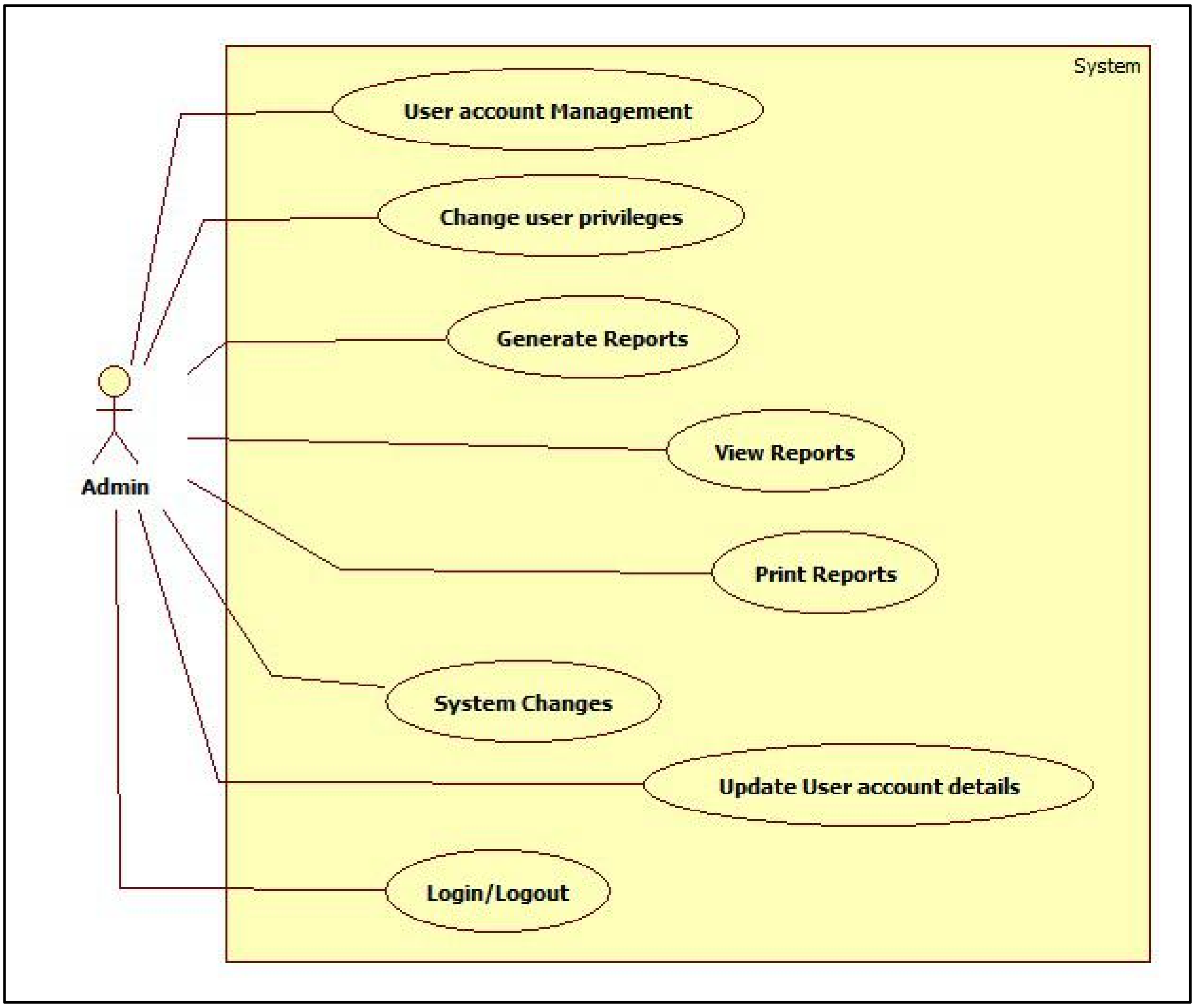


Figure 3.1 Use Case Diagram -Administrator

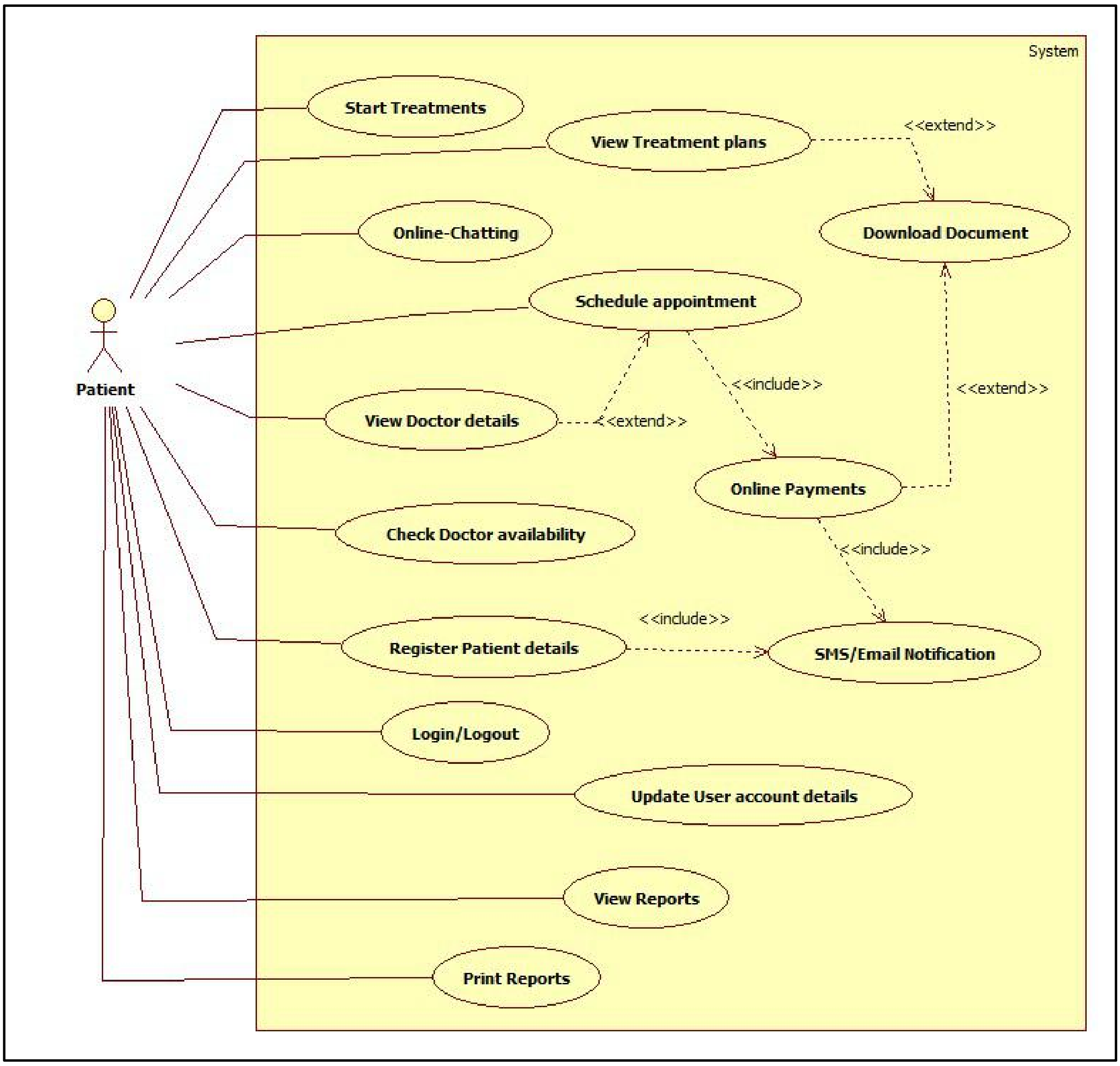


Figure 3.2 Use Case Diagram -Patient

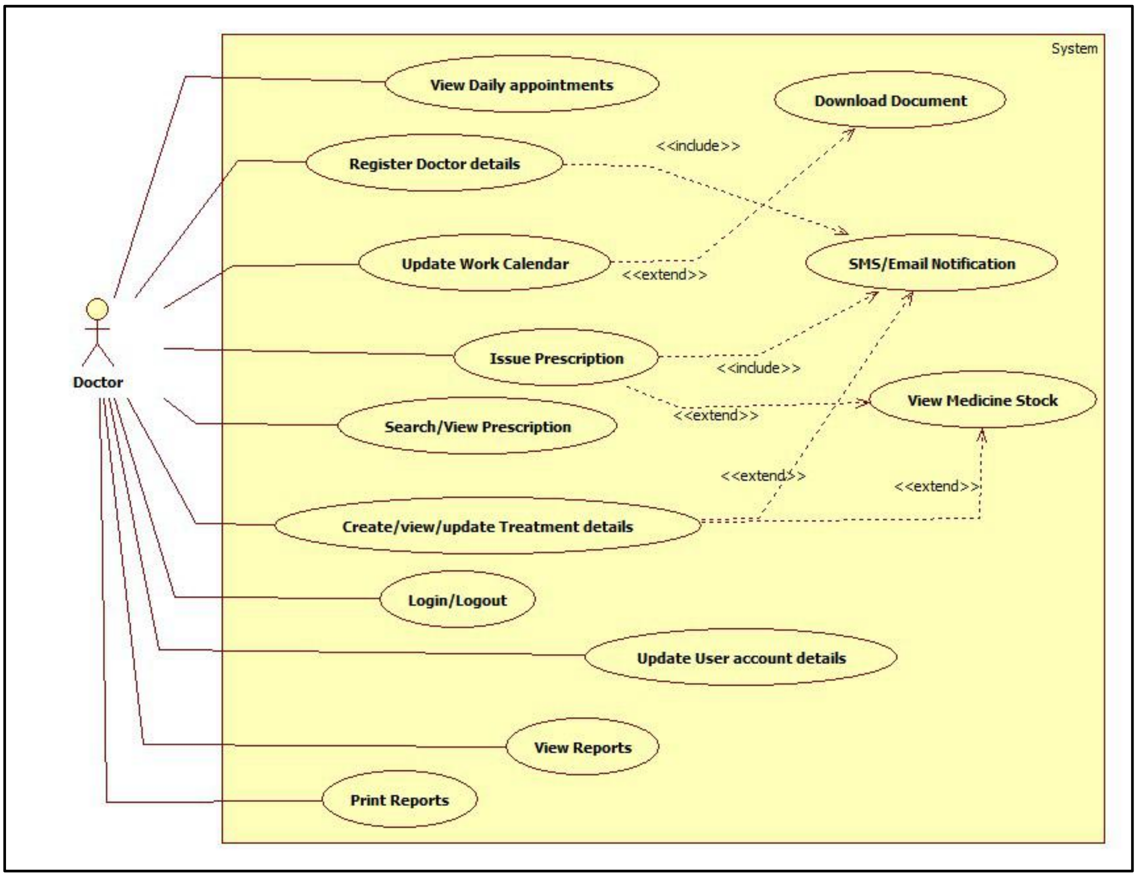


Figure 3.3 Use Case Diagram –Doctor

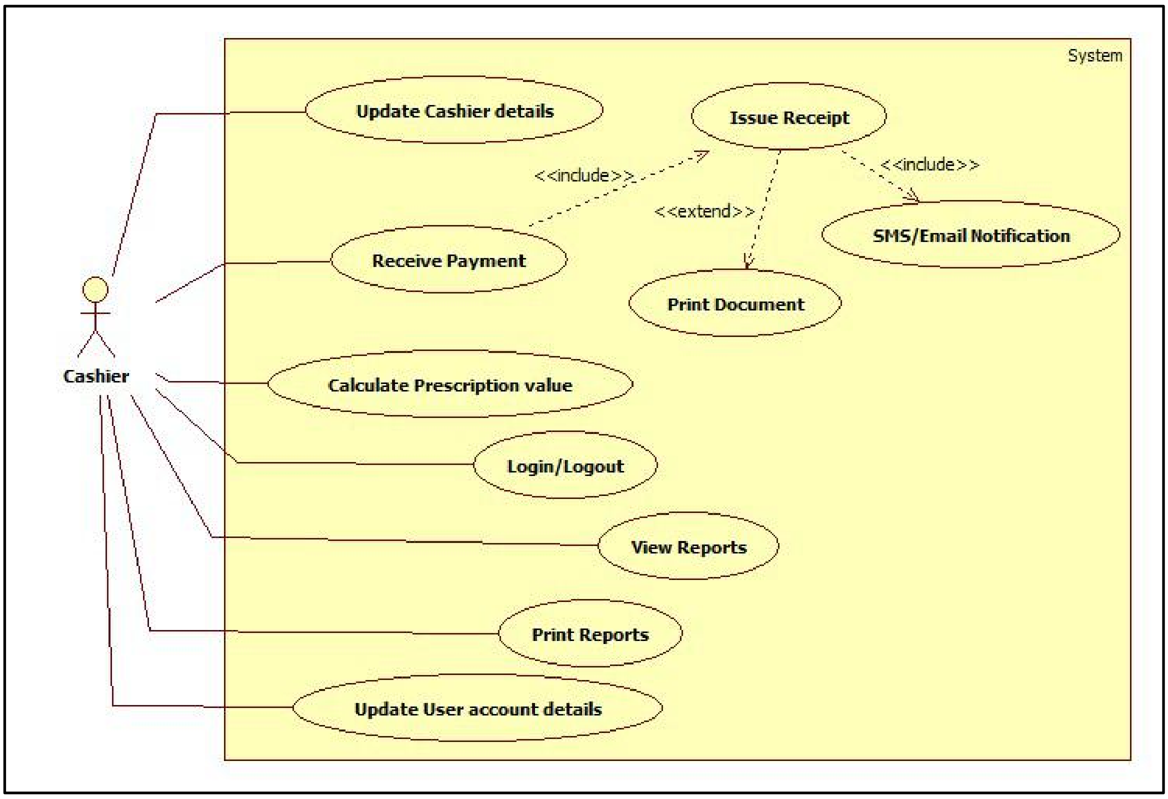


Figure 3.4 Use Case Diagram -Cashier

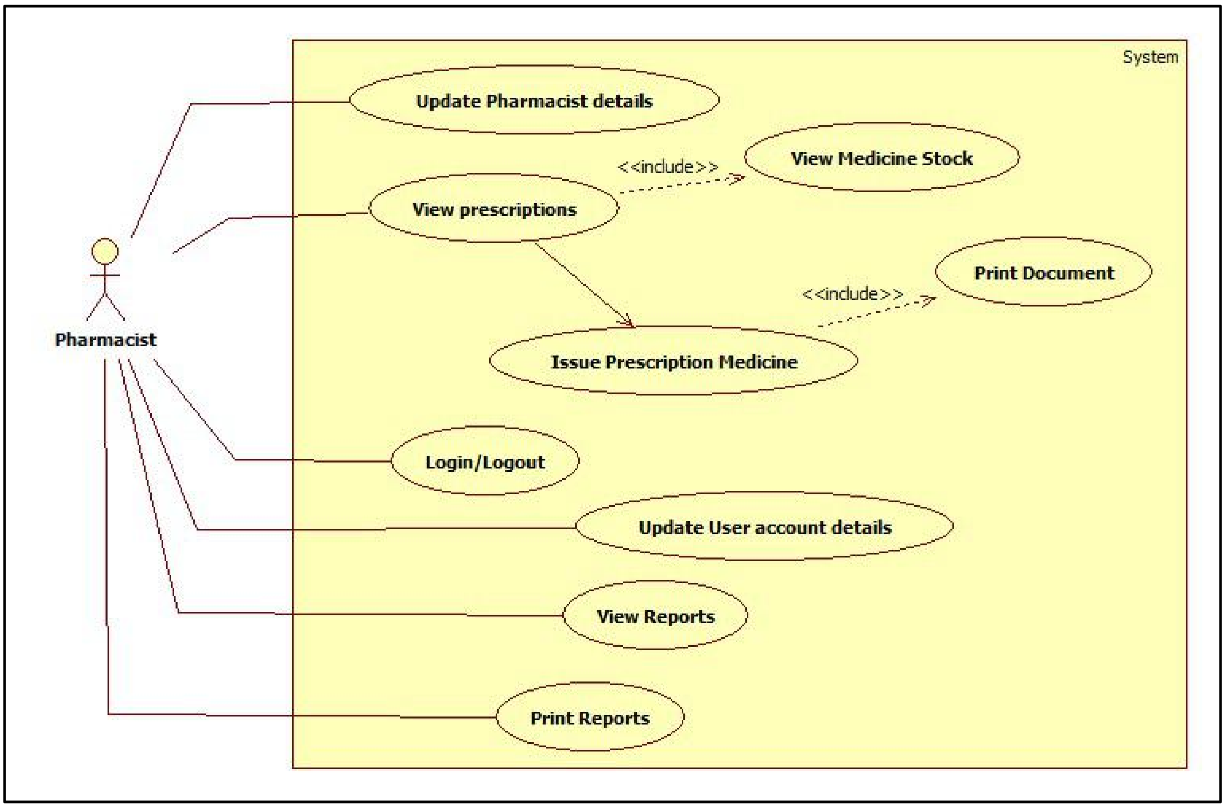


Figure 3.5 Use Case Diagram -Pharmacist

3.4.2. Entity Relationship Diagram

Entity relationship diagram is a graphical representation of an information system that depicts the relationships among people, objects, places, concepts or events within that system. An ERD is a data modeling technique that can help define business processes and be used as the foundation for a relational database. Figure 3.6 shows the ER diagram of the system.

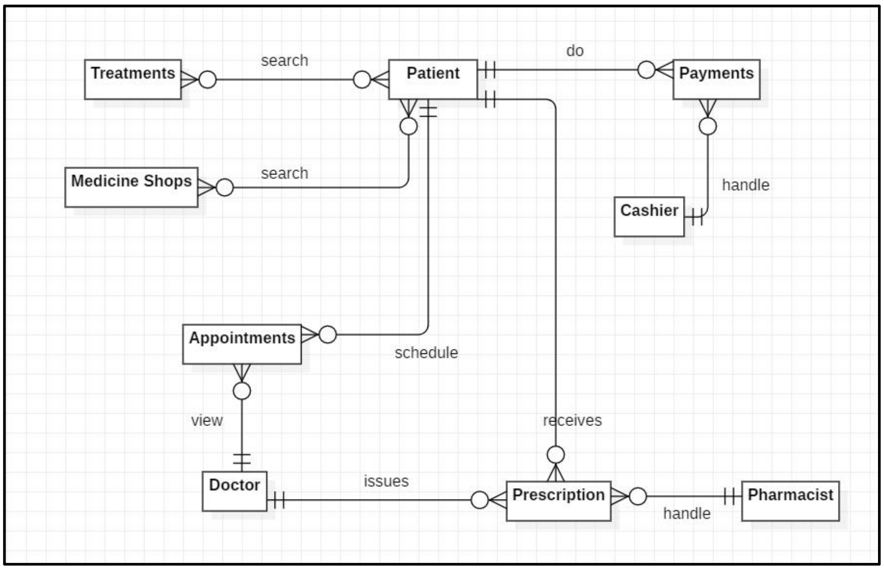


Figure 3.6 Entity Relationship Diagram

3.4.3. Database Design

Database Design for the proposed system can be identified as figure 3.7 below;

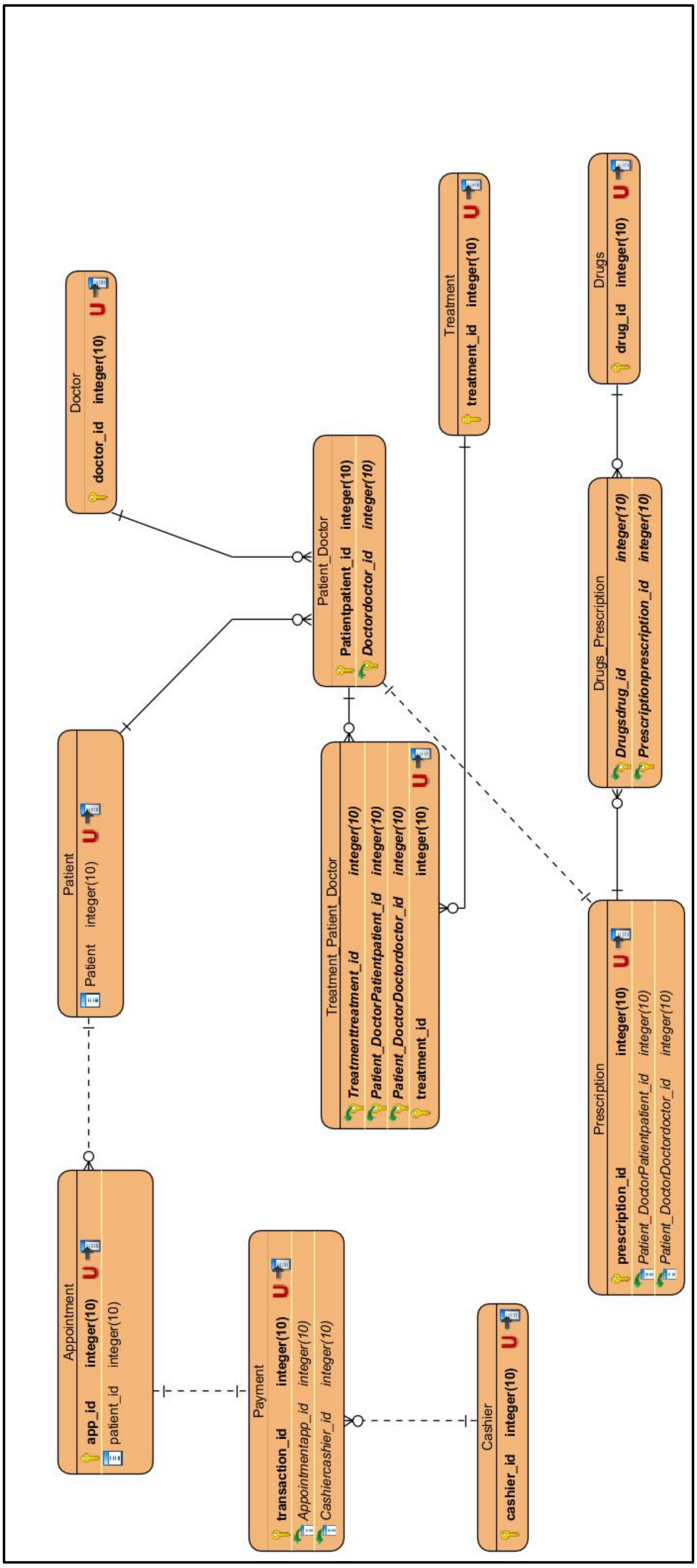

Doctor
doctor_id integer(10)
Treatment
treatment_id integer(10)
Drugs
drug_id integer(10)
Patient_Doctor
Patientpatient_id integer(10)
Doctordoctor_id integer(10)
Drugs_Prescription
Drugsdrug_id integer(10)
Prescriptionprescription_id integer(10)
Patient
Patient integer(10)
Treatment_Patient_Doctor
Treatmenttreatment_id integer(10)
Patient_DoctorPatientpatient_id integer(10)
Patient_DoctorDoctordoctor_id integer(10)
treatment_id integer(10)
Prescription
prescription_id integer(10)
Patient_DoctorPatientpatient_id integer(10)
Patient_DoctorDoctordoctor_id integer(10)
Appointment
app_id integer(10)
patient_id integer(10)
Payment
transaction_id integer(10)
Appointmentapp_id integer(10)
Cashiercashier_id integer(10)
Cashier
cashier_id integer(10)

Figure 3.7 Database Design

3.4.4. Class Diagram

Class Diagram for the proposed system can be identified as figure 3.8 below;

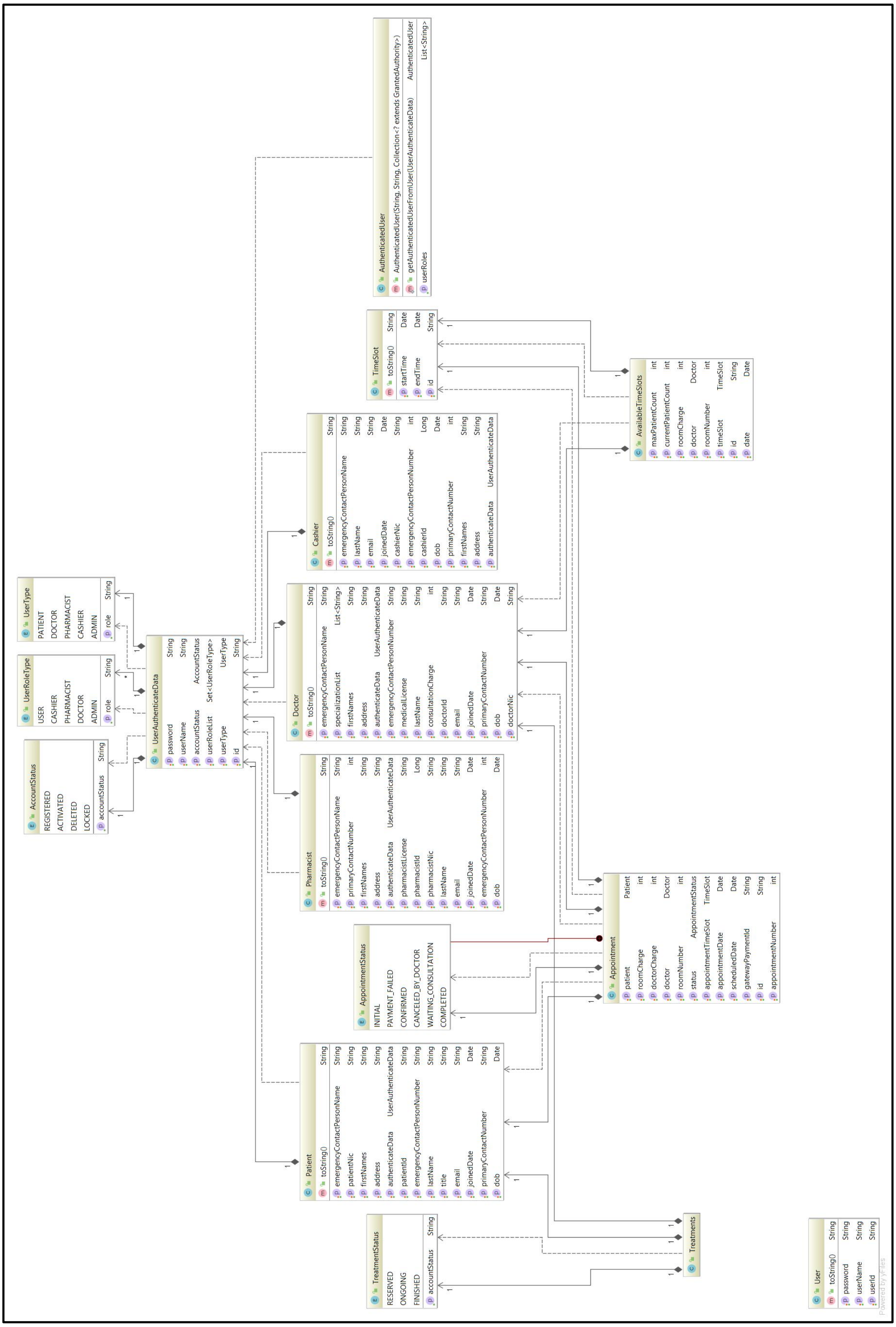


Figure 3.8 Class Diagram

3.4.5. Sequence Diagram

Sequence diagram for the main process can be identified as figure 3.9 below;

Refer "Appendix-A Design Documentation for other diagrams"

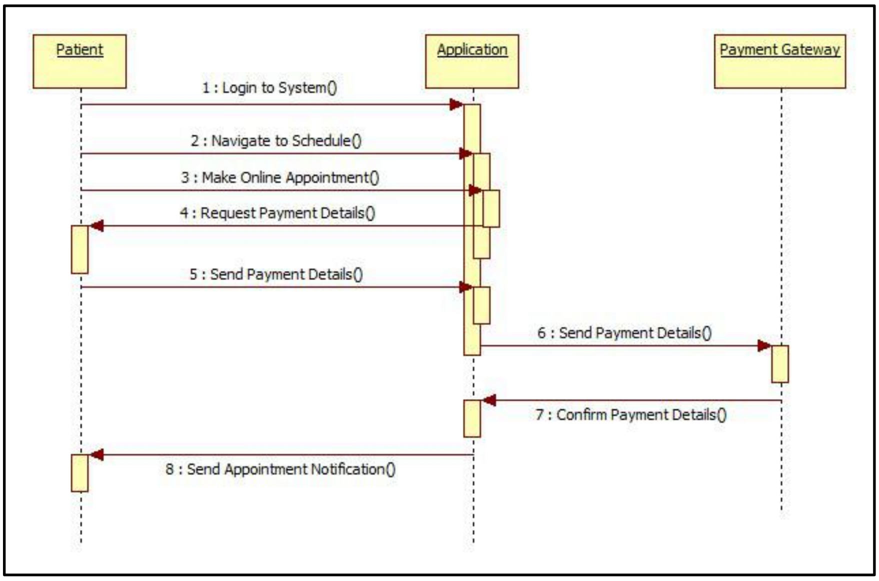

Patient
Application
Payment Gateway
1 : Login to System()
2 : Navigate to Schedule()
3 : Make Online Appointment()
4 : Request Payment Details()
5 : Send Payment Details()
6 : Send Payment Details()
7 : Confirm Payment Details()
8 : Send Appointment Notification()

Figure 3.9 Sequence Diagram – Appointment Process

3.4.6. User Interface Design

User interfaces [18] are the access points where users interact with designs. Following are some interface designs specifically designed for this web and mobile application. Those can by identify as 3.10,3.11,3.12,3.13,3.14 and 3.15 figures below;

3.4.6.1. User Interface Design – Web Application

For all the user interfaces has been designed as a combination of light blue and white colors. All the forms formats has been designed with more usability and all minum required details has been added as fields. Most of the data has been loaded by drop down, then it willb easy for the user to select. Also all the mandorty fields are marked.For all the valaidation breaks proper error messages will be promot.Also with the header navigation part user can navigate to any page any time.

In the home page for the doctor search funcutality user cannot select past dates and also each time user can saerch only one doctor or specaification with a valid date.Also all the other page navaigate is possible from home page as well.

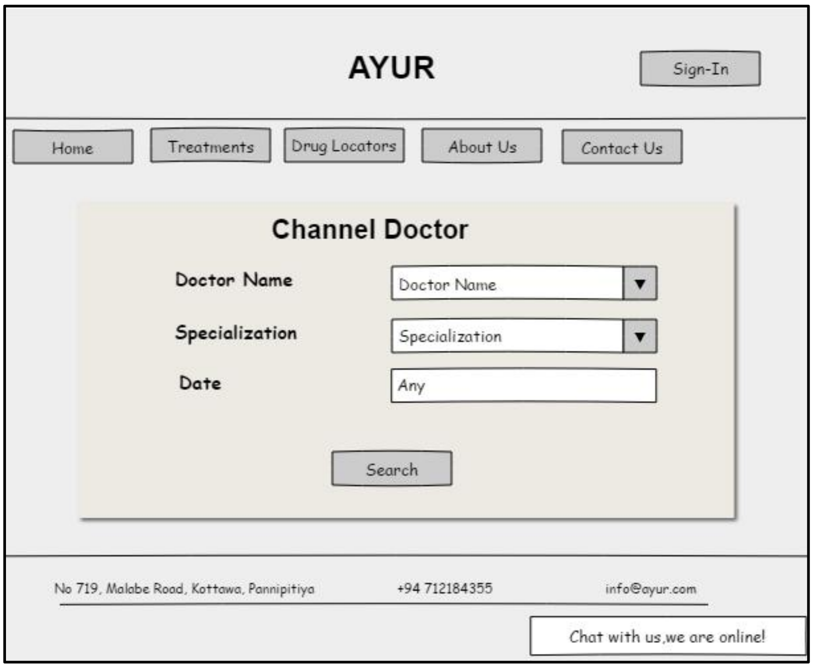


Figure 3.10 User Interface Design –Home Screen

In the login form,user can be logged with the valid user name and password.For the inavalid user login details proper validation messages will be promt.this interface has been designed with forgot password and sign up links as well.

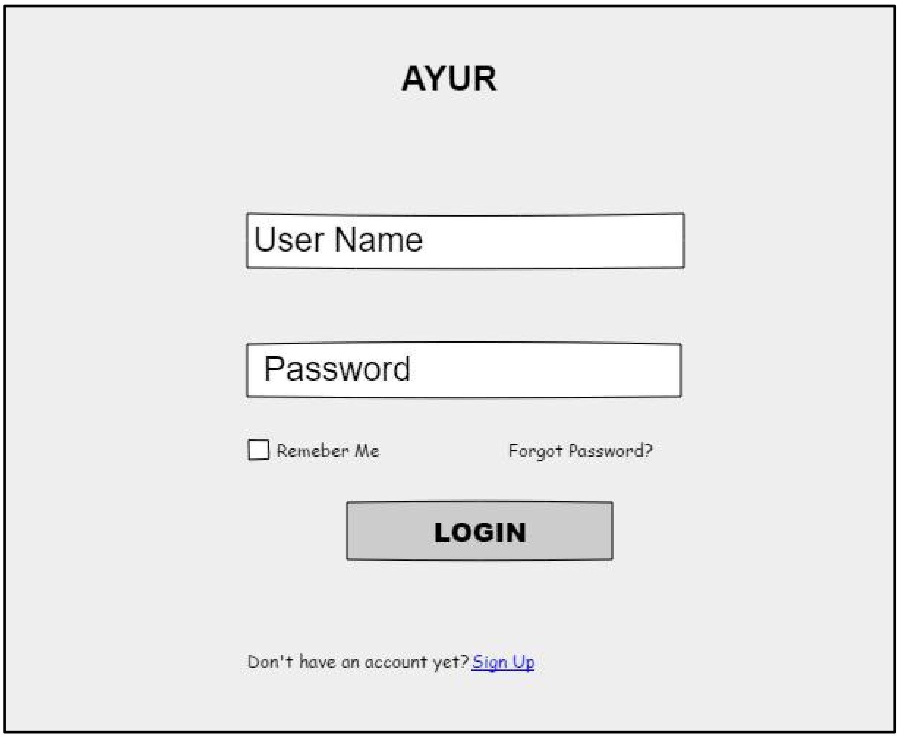
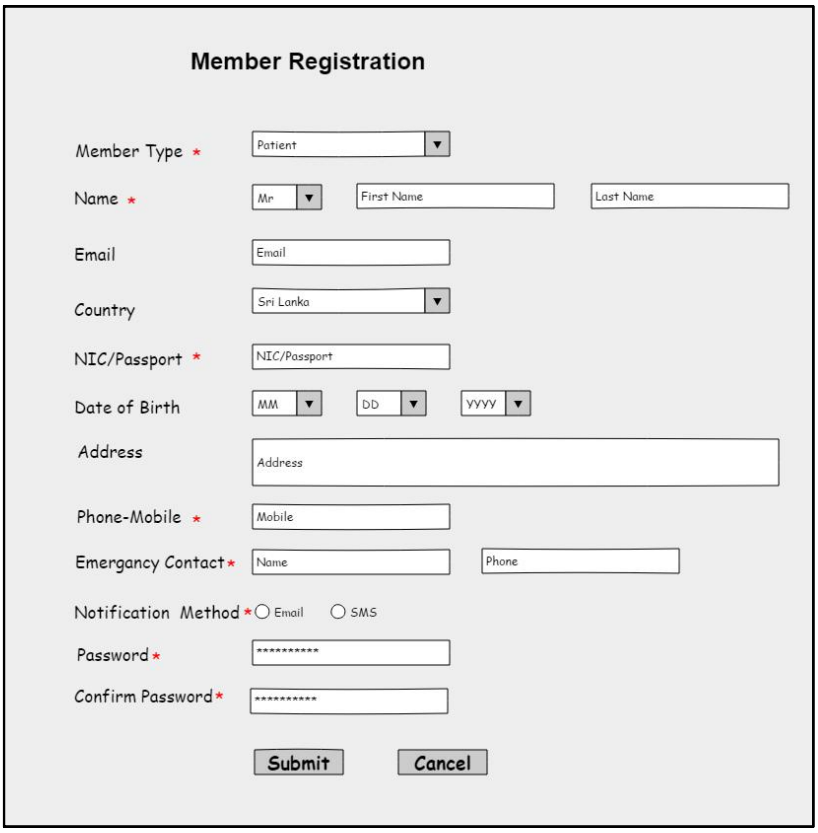


Figure 3.11 User Interface Design –Login Screen

In the Registration form,this form fields will be designed according to the sealcted member type and fields will be diifer according to the selected member type.All the mandotry field s have been marked with "*"and proper validation message will be promot for the invalid data (email format,phone number format,NIC format,password etc.)



Member Registration

Member Type *	Patient ▼
Name *	Mr ▼ First Name Last Name
Email	Email
Country	Sri Lanka ▼
NIC/Passport *	NIC/Passport
Date of Birth	MM ▼ DD ▼ YYYY ▼
Address	Address
Phone-Mobile *	Mobile
Emergancy Contact*	Name Phone
Notification Method*	○ Email ○ SMS
Password*	**********
Confirm Password*	**********

Figure 3.12 User Interface Design –Member Registration Screen

3.4.6.2. User Interface Design – Mobile Application

Mobile application has been designed with the simple interfaces and with increasing the userbility.All the mandotry fields will be marked and proper validation messages will be promot.All the validations are handeled same as the web application. Also forms has been desgined similar to web application with the similar colour combiniation and fields. With this design user can do all the functulaties with filling the minimal requiremnts.

In the login form all the fields validations will be checked and proper validation message will be prompt. In the member registration user can sign up with the minimal fields fillings.All the validations will be handeled same as the web application.

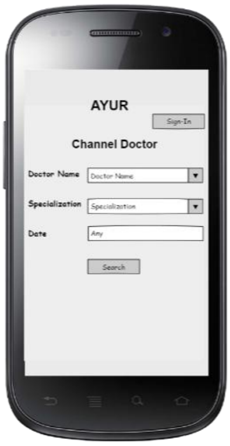


Figure 3.13 User Interface Design –Home Screen

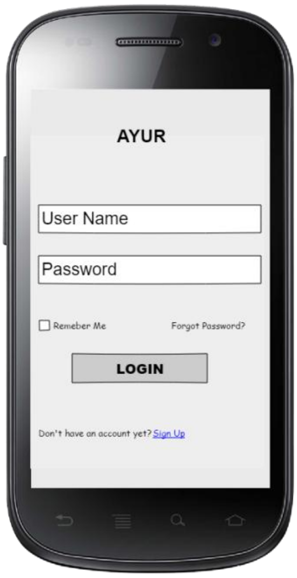


Figure 3.14 User Interface Design –User Login

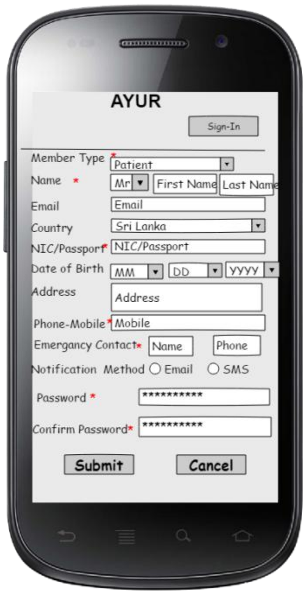


Figure 3.15 User Interface Design –Member Registration

3.5.Implementation Environment

3.5.1. Client/Server Architecture

This software is run in a client/server architecture. A client is any process that requests some services from server processes and a server is a process that provides some requested services for clients. The server should be a Low server with 8 GB RAM and 300 GB Storage. It is better if the client match with the below minimum hardware/software requirements.

Hardware	Software
2.4GHz CPU or upwards	Windows 2007 (64 bit) or higher Operating System
2GB RAM	Android 8.1 or higher
Printer for report printing	Mongo DB 4.0.4
	Java 11

Table 3.1 User Interface Design –Client/Server Architecture

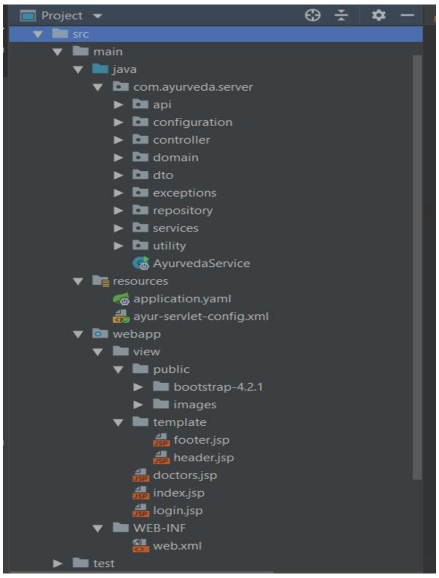
3.6. Development Tools and Techniques

All the tools and techniques are listed below;

- Angular 7
  This is the front-end development tool used. Because it lets the developer turn HTML-based documents into dynamic content easily.
- Open JDK 11
  This is the programming language was used to develop the web and mobile solution. Because it can be used with any application development and other open source tools to improve the performance in open source implementation model.
- Spring Boot 2.1.1.
  This framework was used to simplify the development of the services. Because it can quickly set up and run standalone, web applications and micro services at very less time.
- Mongo DB 4.0.4
  Mongo DB was used as the database. Because it is a schema less database. That means we can have any type of data in a sperate document also the performances are very high
- HTML with Bootstrap 4
  This was used to build the interfaces of the system. Because it is a consistent framework that supports major of all browsers and CSS compatibility fixes and also it has responsive structures and styles.
- JavaScript
  JavaScript was used to handle all the client-side validations. Because it handles client-side security and also supports different browsers.

3.7. Application Development Architecture

This system was developed by using MVC architecture. The Model is used to maintain the data of the application. View display data, using models and controller handles user requests. Figure 3.16 shows the folder structure for development and figure 3.17 shows the architecture. For the code snippets refer appendix F.



```
▼ src
  ▼ main
    ▼ java
      ▼ com.ayurveda.server
        ▶ api
        ▶ configuration
        ▶ controller
        ▶ domain
        ▶ dto
        ▶ exceptions
        ▶ repository
        ▶ services
        ▶ utility
          AyurvedaService
    ▼ resources
        application.yaml
        ayur-servlet-config.xml
    ▼ webapp
      ▼ view
        ▼ public
          ▶ bootstrap-4.2.1
          ▶ images
        ▼ template
            footer.jsp
            header.jsp
          doctors.jsp
          index.jsp
          login.jsp
      ▼ WEB-INF
          web.xml
  ▶ test
```

Figure 3.16 Development Architecture

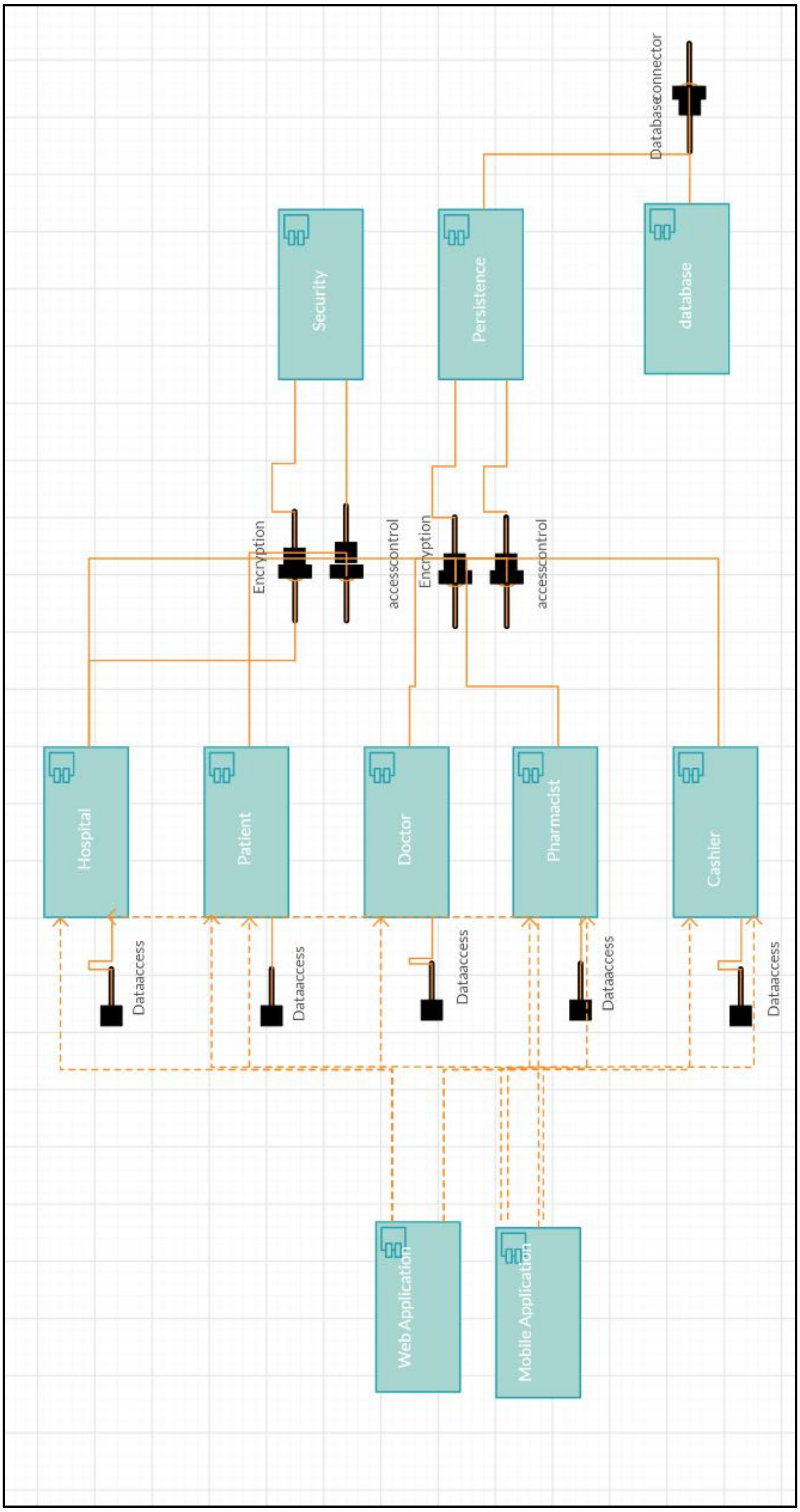


Figure 3.17 Component Diagram

CHAPTER 04: EVALUATION

4.1.Introduction

This chapter includes software testing and user evaluation details. Software testing means verification and validation of the application under test (AUT). It helps to identify the missing requirements, gaps, errors that make a conflict to the actual requirements. Main Levels of Software testing are functional and Nonfunctional. User evaluation helps to know how well users can learn a product and use a product to achieve their goals. It also how satisfied users are with the process.

4.2.Software Testing Levels

Functional Testing

Functional testing is a used to verifies/validates each and every function of the software operates in conformance with the requirement specification.

1. **Unit Testing**
   These kinds of testing [19] helps to test each module separately. It checks whether the individual components are fulfilling the functionalities or not.

2. **Integration Testing**
   In this testing phase, individual software modules are combined and tested as a group to make sure the system is ready for system testing.

3. **System testing**
   System testing will be performed on a complete integrated system. It tests overall functionalities of the system and involves load testing, performance testing, security testing, etc.

4. **Acceptance Testing** [20]
   This is the software testing process where a system is tested for acceptability. The main target of this test type is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery or not. Acceptance testing is normally done by the end user or the customer.

Non-Functional Testing

Non-functional testing is a type of testing to check non-functional facts (performance, usability, reliability, etc.) of a software.

1. **Performance Testing** [21]
   This is a testing type executed to determine the system parameters (users) in terms of reactiveness and solidity under the various workloads.

2. **Usability Testing**
   This is a testing type executed to expose usability defects. Mainly focus on ease of using the application.

4.3.Test Plan

The test plan [22] is designed to describe the scope, objectives, resources, and schedule of a software testing effort of this web and mobile application.

4.3.1. Test Strategy

4.3.1.1 Scope of Testing

- **Features to be tested – (Functional & Non -Functional)**

Features	**Phase**
Treatment Search	Phase 1
Drug Locators Search	Phase 1
Online Chatting	Phase 1
User Login	Phase 1
Change Password	Phase 1
Member Registration	Phase 2
Doctor Channeling	Phase 2
Doctor Prescription Handling	Phase 2
Pharmacy Drug Management	Phase 3
Payment Handling	Phase 3
Mobile Testing	Phase 4
Performance Testing	Phase 4

Table 4.1 Features to be tested

- **Features not to be tested**

  ✓ Hardware Interfaces
  ✓ Database Logical
  ✓ Communications Interfaces

4.3.2. Test Type

In the project, there are 6 types of testing should be conducted.

1. Unit Testing

2. Integration Testing
3. System testing
4. Acceptance Testing
5. Mobile Testing
6. Performance Testing

4.3.3. Risk and Issues

Risk [23]	**Mitigation**
The project schedule is too tight; it's hard to complete this project on time	Set Test Priority [24] for each of the test activity.
Lack of the required skills for performance testing.	Self-study performance testing
Wrong project testing time estimation	Establish the scope before beginning work, pay a lot of attention to project planning and constantly track and measure the progress

Table 4.2 Risk and Issues

4.3.4. Test Criteria

4.3.4.1 Suspension Criteria

If **40%** of test cases **failed**, suspend testing until fixe all the failed cases.

4.3.4.2 Exit Criteria

Specifies the criteria that denote **successful** completion of a test phase.

- The **Run** rate of the test cases is mandatory to be **100%** unless a clear reason is given.

- The **pass** rate is **80%,** achieving the pass rate is **mandatory**.

4.4.Test Cases and Test Results

Refer the ‘Appendix D– Test Results’ for Mobile and performance testing [25] in Phase 4.All the functional testing cases has been tested manually and performance testing has been tested by using JMeter.

4.4.1. Treatment Search

Test Case ID	Test Scenario	Test Case	Expected Results	Actual Result		
				Phase 1	Phase 2	Phase 3
A01	Search treatments with valid data	Search with Selecting only a treatment type	Searched Details will be viewed	Passed	Passed	Passed
		Search with entering only existing key word for treatment description	Searched Details will be viewed	Passed	Passed	Passed
		Search with selecting a treatment type and entering an existing key word for treatment description	Searched Details will be viewed	Passed	Failed	Passed
A02	Search treatments with invalid data	Search with remain all fields empty	Prompt No result found	Passed	Passed	Passed
	Validations	Search with entering 255 characters (max length) in the treatment description	Searched Details will be viewed	Failed Reason: maximum length has been set as 150	Passed	Passed
		Search with entering (characters+ numeric+ special characters) the treatment	Searched Details will be viewed	Failed Reason: cannot enter special characters	Passed	Passed

		description				
A03	Search Results-Enlarge Images	Enlarge the images by clicking on it	Images will be enlarged	Failed Reason: cannot enlarge	Passed	Passed

Table 4.3 Test Cases and Results – Treat Search Module

4.4.2. Drug Locators Search

Test Case ID	Test Scenario	Test Case	Expected Results	Pass/Fail		
				Phase 1	Phase 2	Phase 3
A04	Search drug locators with valid data	Search with Selecting only an area	Searched Details will be viewed	Passed	Passed	Passed
A05	Search treatments with invalid data	Search with remain all fields empty	Prompt No result found	Passed	Passed	Passed

Table 4.4 Test Cases and Results – Drug Locators Search Module

4.4.3. Online Chatting

Test Case ID	Test Scenario	Test Case	Expected Results	Pass/Fail		
				Phase 1	Phase 2	Phase 3
A06	Chat Send	Chat send by user	User can be chatted	Failed Reason: chat was not working	Passed	Passed
A07	Chat Reply	Reply by the admin	Replies can be sent	Failed Reason : chat was not working	Failed	Passed

Table 4.5 Test Cases and Results – Online Chatting Module

4.4.4. User Login

Test Case ID	**Test Scenario**	**Test Case**	**Expected Results**	**Pass/Fail**		
				Phase 1	**Phase 2**	**Phase 3**
A08	Login to the system with invalid data	Login with empty username and password	Prompt error messages	Passed	Passed	Passed
		Login with empty username and correct password	Prompt error messages	Passed	Passed	Passed
		Login with correct username and empty password	Prompt error messages	Passed	Failed	Passed
		Login with incorrect username and password	Prompt error messages	Passed	Passed	Passed
A09	Login to the system with valid data	Login with correct username and correct password	Successfully Log in to the system	Passed	Passed	Passed
A10	Store Login credentials in the session	Login with correct username and correct password by checking remember me check box	Login credentials will be stored in the session	Passed	Passed	Passed
A11	Screen Link Navigations	Navigating to change password screen by clicking the particular link	User should navigate to change password screen	Passed	Passed	Passed
		Navigating to member registration screen by clicking the particular link	User should navigate to change member registration screen	Passed	Passed	Passed

Table 4.6 Test Cases and Results – User Login Module

4.4.5. Change Password

Test Case ID	**Test Scenario**	**Test Case**	**Expected Results**	**Pass/Fail**		
				Phase 1	**Phase 2**	**Phase 3**
A12	Change Password with invalid data	Change the password with all fields empty	Prompt error messages	Passed	Passed	Passed
		Change the password with old password empty and other two filled	Prompt error messages	Passed	Passed	Passed
		Change the password with new password empty and other two filled	Prompt error messages	Passed	Passed	Passed
		Change the password with confirm password empty and other two filled	Prompt error messages	Passed	Passed	Passed
		Change the password with incorrect old password and other two correct	Prompt error messages	Passed	Passed	Passed
		Change the password with correct old password and other two incorrect	Prompt error messages	Passed	Passed	Passed
		Change the password with correct old password and new password with incorrect confirm	Prompt error messages	Passed	Passed	Passed

		password				
A13	Change Password with invalid data	Change the password with correct old password and new password with incorrect confirm password	Successfully Log in to the system	Passed	Passed	Passed
A14	Store Login credentials in the session	Login with correct username and correct password by checking remember me check box	Login credentials will be stored in the session	Failed Reason: not working	Passed	Passed
A15	Screen Link Navigations	Navigating to change password screen by clicking the particular link	User should navigate to change password screen	Passed	Passed	Passed
		Navigating to member registration screen by clicking the particular link	User should navigate to change member registration screen	Failed Reason: navigate to wrong pages	Passed	Passed

Table 4.7 Test Cases and Results – Change Password Module

4.4.6. Member Registration

Test Case ID	Test Scenario	Test Case	Expected Results	Pass/Fail		
				Phase 1	Phase 2	Phase 3
A16	Submit Member details with invalid data	Remain all the fields empty	Prompt error messages - mandatory fields	Passed	Passed	Passed

A17	Submit Member details with valid data	Filled all the fields correctly	Successfully registered the member	Passed	Passed	Passed
			User can proceed the channeling	Passed	Passed	Passed
A18	Validations	Enter all the text field maximum length with characters+ numeric+ special characters	Successfully registered the member	Failed Reason: once “&” sign entered cannot save	Passed	Passed
		Email with incorrect format	Error message prompt	Passed	Passed	Passed
		NIC/Passport in invalid format	Error message prompt	Passed	Failed	Passed
		Phone Mobile/Contact No with characters + special characters	Error message prompt	Passed	Passed	Passed
		Correct password and incorrect confirm password	Error message prompt	Passed	Passed	Passed

Table 4.8 Test Cases and Results – Member Registration Module

4.4.7. Doctor Channeling

Test Case ID	**Test Scenario**	**Test Case**	**Expected Results**	**Pass/Fail**		
				Phase 1	**Phase 2**	**Phase 3**
Doctor Details Search						
A19	Search Doctor with valid data	Select only doctor name	Matched Doctor details will be viewed	Passed	Passed	Passed
		Select only specialization	Matched Doctor details will	Passed	Passed	Passed

			be viewed			
		Select only date	Matched Doctor details will be viewed	Passed	Failed	Passed
A20	Search Doctor with invalid data	Select a doctor and incorrect specialization	Error message prompt	Passed	Passed	Passed
		Select doctor and specialization and incorrect date	Error message prompt	Passed	Passed	Passed
A21	Channel doctor	Channel a doctor from the searched data	User will navigate to date slots details page	Passed	Failed	Passed
Channel Date Slots						
A22	Available date slot book	Channel a date for the doctor	User will navigate to patient details page	Failed Reason: Navigate to home page	Failed	Passed
A23	Full date slot book	Channel a date for the doctor	Unable to book	Passed	Passed	Passed
A24	Holiday Date slot book	Channel a date for the doctor	Unable to book	Passed	Passed	Passed
Patient Details Fill						
A25	Existing user channeling doctor	Submit the auto filled patient details	User should navigate to payment screen	Passed	Passed	Passed
A26	Unregistered user channeling doctor	Submit the entered patient details	User should navigate to payment screen	Passed	Passed	Passed

Table 4.9 Test Cases and Results – Doctor Channeling Modul

4.4.8. Cashier Booking Confirm

Test Case ID	**Test Scenario**	**Test Case**	**Expected Results**	**Pass/Fail**		
				Phase 1	**Phase 2**	**Phase 3**
A27	Confirm Appointment with valid data	Confirm by entering reference no	Channel details should be auto filled	Failed Reason: Reference number won't generate	Passed	Passed
		Confirm by entering NIC/ Passport no	Channel details should be auto filled	Failed Reason: details won't auto filled	Passed	Passed
		Confirm by entering patient name	Channel details should be auto filled	Passed	Passed	Passed
			View previous prescriptions	Passed	Passed	Passed
A28	Confirm Appointment with invalid data	Confirm by entering incorrect reference no	Error message prompt	Passed	Passed	Passed
		Confirm by entering incorrect NIC/passport no	Error message prompt	Passed	Passed	Passed
		Confirm by entering incorrect patient name	Error messages will be prompt	Passed	Failed	Passed
A29	Confirm Appointment	Confirm Appointment	Send appointment details to	Passed	Passed	Passed

	details	details	doctor			
A30	Print Appointment details	Print Appointment details	Appointment details can be printed	Passed	Passed	Passed

Table 4.10 Test Cases and Results – Cashier Booking Confirmation Module

4.4.9. Doctor Prescription Handling

Test Case ID	Test Scenario	Test Case	Expected Results	Pass/Fail		
				Phase 1	Phase 2	Phase 3
A31	View Session	Doctor login to system and view the session	Viewed session details	Failed Reason: won't view session details	Passed	Passed
A32	Start Session	Click to start the session	No 1 and patient details will be appeared	Passed	Passed	Passed
A33	Enter details	Enter details with maximum characters length	Details can be entered	Passed	Passed	Passed
			View previous prescriptions	Passed	Passed	Passed
A34	Calendar treatments schedule	Mark next treatments in the calendar	Marked calendar details will be viewed	Failed Reason : can mark for past dates as well	Passed	Passed
A35	Enter Internal Prescriptio	Add drug items by selecting details	Drug items can be added	Passed	Passed	Passed

	n details - available drugs		to data grid			
A36	Enter Internal Prescription details - unavailable drugs	Add drug items by selecting details	Error messages will be prompt	Passed	Failed	Passed
			Drug items will be added to external prescription data grid	Passed	Passed	Passed
A37	Send Prescription details	Send prescription details	Prescription details will be sent to pharmacy	Passed	Passed	Passed
A38	Print prescription details	Print prescription details	Prescription details can be printed	Failed Reason : once click print will navigate to registration page	Passed	Passed

Table 4.11 Test Cases and Results – Doctor Prescription Handling Module

4.4.10. Pharmacist Medicine Issuing

Test Case ID	Test Scenario	Test Case	Expected Results	Pass/Fail		
				Phase 1	Phase 2	Phase 3
A39	Send Prescription details	Send Prescription details to cashier	Prescription details can be sent	Passed	Passed	Passed
A40	Print Prescription details	Print Prescription details	Prescription details can be print with drug item pictures	Passed	Passed	Passed
			View previous prescriptions	Passed	Passed	Passed

Table 4.12 Test Cases and Results – Pharmacist Drug Issuing Module

4.4.11. Cashier Payment Handling

Test Case ID	Test Scenario	Test Case	Expected Results	Pass/Fail		
				Phase 1	Phase 2	Phase 3
A41	Print Payment details	Print Payment details	Payment details can be printed	Passed	Passed	Passed

Table 4.13 Test Cases and Results – Cashier Payment Handling Module

The summary of test results in each phase can be identified in Table 4.12, Table 4.13 and Table 4.14. All the failed test cases has been reports as bugs and most of them are priority medium .All the defects has been fixed before the next release. Most of the defect has been occurred because unit testing is not enough and some gaps with the requirements and the developed software. All the bugs have been fixed before releasing to the end user.

Lessons Learnt

1. Need to align with the requirement
2. Need to conduct more unit testing
3. Need to identify the gaps between requirement and the software
4. Need to analysis the estimate time for the development

Phase 1											
	Treatment Search	Drug Locators Search	Online Chatting	User Login	Change Password	Member Registration	Doctor Channeling	Cashier Channel	Prescription Handling	Drug Issuing	Payment Handling	Total
Passed	3	2	0	8	9	7	11	7	7	3	1	58
Failed	3	0	2	0	2	1	1	2	2	0	0	13
Blocked	1	0	1	0	0	0	1	0	0	0	0	3
Total	7	2	3	8	11	8	13	9	9	3	1	74

Table 4.14 Summary of Test Results in Test Phase 1

Phase 2												
	Treatment Search	Drug Locators Search	Online Chatting	User Login	Change Password	Member Registration	Doctor Channeling	Cashier Channel	Prescription Handling	Drug Issuing	Payment Handling	Total
Passed	6	2	1	8	11	7	9	6	8	3	1	62
Failed	1	0	1	0	0	1	2	1	1	0	0	7
Blocked	0	0	1	0	0	0	0	1	0	0	0	2
Total	7	2	3	8	11	8	10	8	9	3	1	71

Table 4.15 Summary of Test Results in Test Phase 2

Phase 3												
	Treatment Search	Drug Locators Search	Online Chatting	User Login	Change Password	Member Registration	Doctor Channeling	Cashier Channel	Prescription Handling	Drug Issuing	Payment Handling	Total
Passed	7	2	2	8	11	7	11	7	9	3	1	68
Failed	0	0	0	0	0	0	0	0	0	0	0	0
Blocked	0	0	0	0	0	0	0	0	0	0	0	0
Total	7	2	2	8	11	7	11	7	9	3	1	68

Table 4.16 Summary of Test Results in Test Phase 3

4.5.Acceptance Test Results

The main purpose of the system built is to satisfy the customer requirements, the final system should be accepted by the end users to the complete project as a successful project. Thus, a questionnaire was designed and make system assess to a set of the users by capturing the customer feedbacks on the survey. The following is a sample questionnaire created.

“Ayur” Patient Management and Medicine Stock Management System **User Evaluation Form**					
Date:		**Name:**		**Designation:**	
I appreciate your help in evaluating new software to your shop by filling short answers to below questions. Put “X” to appropriate column.					
#	Question	Strongly Agree	Agree	Disagree	Strongly Disagree
1	Interfaces are simple and understandable				
2	System display information more familiar				
3	Easily navigate through system				
4	Functions are more reliable				
5	System provide sufficient functionalities				
6	Degree of Information provided in Reports.				
7	Ability to maintain data, to keep it up-to date.				
8	Ease of entering/ handling form fields, and handle manipulations.				
Comments 1					
Thank You					

Table 4.17 Acceptance Test-User Evaluation Form

The Whole Feedback ratings given by each user were taken and converted into a graphical evaluation chart. The following bar chart in Figure 4.15 displays the overall feedback.

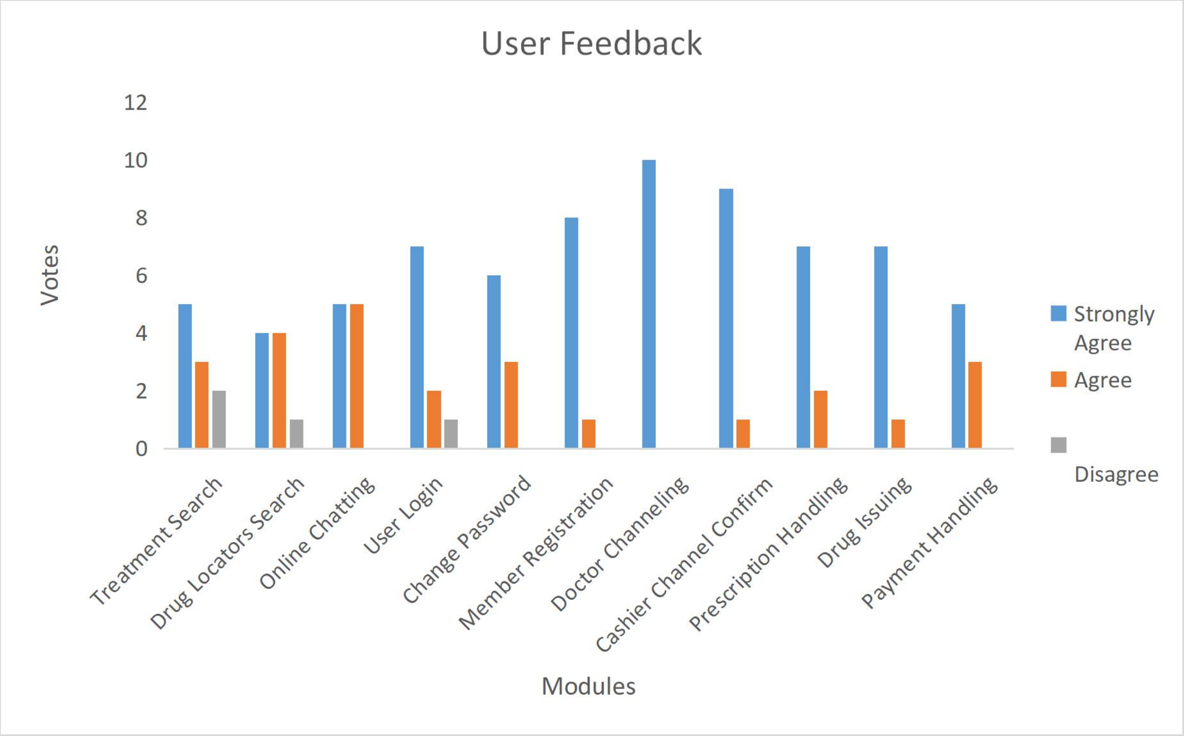


Table 4.18 Acceptance Test-Overall Feedback

The final feedback received from the survey was significantly positive. Most of the users strongly agree for all the functionalities and usability etc. Most agreed module (75% votes) is doctor channeling module and users found very easy to channel doctors. 80% users found all the interfaces are very easy to use and very simple. Most of the doctors (75%) found very easy to maintain patient data. Also, some (10%) disagree for the drug store search module, they found difficult to search the areas. Also, some suggestions were taken and hope to consider them at the Phase 4.

CHAPTER 05: CONCLUSION

5.1.Introduction

This chapter discusses the objectives achieved and the future enhancements of the developed system. “Peramuna Ayurveda Medical Center” is an ayurvedic wellness medical center, provides all types of ayurvedic treatments. Their main objective is to increase its customer base within a short period and provide all their services efficiency. All the current processes of the center are currently handled manually.

The implementation of the new system could be used to overcome the drawbacks in the manual system. It has simple and effective user interface which allow users to easily manage their tasks.

5.2.Problem Encountered

1. **Requirements were not Clear and Stable**
   Although the client has a clear idea about the manual process of their work, they had not any idea how this process can be converted to an automatic system. Since the client does not have any technical knowledge or experience, it was very hard to finalize what the system does according to the client needs. So, it took much time to finalized the requirement with the client and changed in many times as well.

2. **Lack of Development Knowledge and Experience**
   To cater the client requirement, I needed to implement a web application as well as a mobile application. Due to a lack of development knowledge, this was very difficult in the start. To overcome this needed to refer many tutorials, video tutorials, Ebooks, etc.

   Implementing a web application and mobile application at the same with the limited time was the real challenge. Since all the web and mobile technologies used for these implementations are very new to me and I had a great experience and improved my knowledge as well.

5.3.Lesson Learnt

The knowledge added throughout the project was actually valuable. Starting from the requirement gatherings to the end of development this process gave incomparable experience in many ways. This project gave a chance to test and implement the most important theories and technologies learned throughout the MIT degree program. It also facilitated to learn very interesting new and updated technologies (Spring Boot framework, Mongo DB, etc.) in order to improve the system performance. Moreover, special efforts were taken to learn the frameworks.

Furthermore, working on the project encouraged me to improve technical skills as well as intellectual skills by collaborating with many individuals from collective fields. I also gained good knowledge in project management including time and resource management and the report writing knowledge, which was acquired by writing this dissertation, will be very important for my future career.

5.4.Future Work

According to the suggestions and comments gained by the client through the acceptance testing phase some functional and non-functional requirements could be added to improve the system.

The following functions could be added to improve the effectiveness and functionality of the system.

1. Integrated with more ayurvedic hospital and implement a centralized system for all the ayurvedic hospital, medical centers, etc.
2. Create a process to handle a drug delivery service, where users can upload prescriptions and give their details and send orders via web and mobile applications.
3. Create a process to consult a doctor online via applications and send user symptoms and ask for a prescription online.
4. Create a process to make tentative reservations and pay later.

REFERENCES

[1] "Wikipedia," Wikimedia Foundation, Inc, 17 May 2019. [Online]. Available: https://en.wikipedia.org/wiki/Sri_Lankan_traditional_medicine. [Accessed 28 August 2018].

[2] "Atuyrvedic Healing," Ayurvedic Healing Inc, [Online]. Available: https://www.ayurvedichealing.net/panchakarma/. [Accessed 27 August 2018].

[3] "EDB Blog," Sri Lanka Export Development Board, 26 November 2018. [Online]. Available: http://www.srilankabusiness.com/blog/roots-of-sri-lankan-traditional-medicine.html. [Accessed 28 August 2018].

[4] B. a. K. R. Chowdhury, "RFID-based hospital real-time patient management system," in *6th IEEE/ACIS international conference on computer and information science (ICIS 2007)*, IEEE, 2007, pp. 363-368.

[5] 2. P. K. 3. P. S. 1Digvijay H. Gadhari, "Hospital Managment System," *International Journal for Research in Engineering Application & Management (IJREAM),* vol. 01, no. 11, pp. 1-4, 2016.

[6] H. Y. T. E. Yönetimi, "LEAN MEDICAL INVENTORY MANAGEMENT IN HOSPITALS," *International Journal of Academic Value Studies,* vol. 3, no. 8, pp. 1-9, 2017.

[7] "Agile Alliance," Agile Alliance, [Online]. Available: https://www.agilealliance.org/agile101/. [Accessed 30 August 2018].

[8] "Baeldung," 7 May 2019. [Online]. Available: https://www.baeldung.com/jlink. [Accessed 10 January 2019].

[9] M. Raible, "Okta," Okta, 22 August 2018. [Online]. Available: https://developer.okta.com/blog/2018/08/22/basic-crud-angular-7-and-spring-boot-2. [Accessed 11 January 2019].

[10] "android," android, [Online]. Available: https://developer.android.com/training/basics/firstapp. [Accessed 25 January 2019].

[11] "Ayurveda Sri Lanka," [Online]. Available: https://ayurvedasrilanka.net/en/. [Accessed 20 December 2018].

[12] "doc," Dailog, [Online]. Available: https://www.doc.lk/. [Accessed 12 December 2018].

[13] "David Darlings," David Darlings, [Online]. Available: http://www.daviddarling.info/encyclopedia/S/AE_stand-alone_system.html. [Accessed 03 November 2018].

[14] "wikipeadia," Wikimedia Foundation, Inc., 17 May 2019. [Online]. Available: e Wikimedia Foundation, Inc.. [Accessed 03 November 2018].

[15] "Scrum," Scrum Org., [Online]. Available: https://www.scrum.org/resources/what-is-scrum. [Accessed 10 November 2018].

[16] "Datanamic," Datanamic, [Online]. Available: https://www.datanamic.com/support/lt-dez005-introduction-db-modeling.html. [Accessed 10 November 2018].

[17] "Lucidchart," Lucidchart, [Online]. Available: https://www.lucidchart.com/pages/uml-use-case-diagram. [Accessed 12 November 2018].

[18] H. Umbach, "Fresh tilled soil," Fresh tilled soil, 18 April 2018. [Online]. Available: https://www.freshtilledsoil.com/what-is-user-interface-design/. [Accessed 16 November 2018].

[19] "Guru99," [Online]. Available: https://www.guru99.com/levels-of-testing.html. [Accessed 3 February 2019].

[20] "TutorialsPoint," [Online]. Available: https://www.tutorialspoint.com/software_testing_dictionary/acceptance_testing.htm. [Accessed 20 February 2019].

[21] "Guru99," [Online]. Available: https://www.guru99.com/performance-testing.html. [Accessed 04 February 2019].

[22] "Software Testing Fundamentals," [Online]. Available: http://softwaretestingfundamentals.com/test-plan/. [Accessed 8 February 2019].

[23] A. Aitken, "Developer," 2 March 2018. [Online]. Available: https://www.developer-tech.com/news/2018/mar/02/when-take-risk-software-testing-guide/. [Accessed 19 February 2019].

[24] "Testing Excellence," 2 December 2018. [Online]. Available: https://www.testingexcellence.com/severity-and-priority-difference/. [Accessed 10 February 2019].

[25] J. Barlett, "Testlodge," [Online]. Available: https://blog.testlodge.com/how-to-write-test-cases-for-software-with-sample/. [Accessed 10 February 2019].

[26] "Visual Paradigm," [Online]. Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/. [Accessed 18 February 2019].

[27] "User Focus," [Online]. Available: https://www.userfocus.co.uk/articles/usermanuals.html. [Accessed 3 March 2019].

[28] "CIO2019," [Online]. Available: https://whatis.ciowhitepapersreview.com/definition/report-management/. [Accessed 8 March 2019].

[29] "Tutorials Point," [Online]. Available: https://www.tutorialspoint.com/mongodb/. [Accessed 20 March 2019].

[30] "Computer Hope," [Online]. Available: https://www.computerhope.com/jargon/d/database-server.htm. [Accessed 20 March 2019].

[31] "NGINX," [Online]. Available: https://www.nginx.com/blog/. [Accessed 3 April 2019].

Appendix A – Design Documentation

Refer the following use case descriptions to gain a clear idea about the System which was designed for the “Peramuna Ayurvedic Center”.

Use Case Description – User Management

Refer the ‘Table A.1 for the user management use case description

User Case ID	001
Use Case	User Management
Actor(s)	Doctor, Cashier, Pharmacist, Administrator
Description	
Administrator can create users, update user details, active/inactive users and other users only can update the details	
Pre-conditions	
1.User credentials have been validated 2.User Should be connected to the internet	
Normal Cause	
1. Enter user details 2. Click Submit button 3. User will be successfully created.	
Alternative Causes	
1. If the duplicate user details entered, a relevant error message will be displayed. 2.If the mandatory fields have not been filled, a relevant error message will be displayed. 3.If the password won’t be matched, a relevant error message will be displayed.	
Post-conditions	
1.User will be created successfully. 2.User will be received an email regarding user details 3.Users can be logged to the system upon change the password 4.Users will be granted for permission that has been assigned to the particular user role.	

Table A.1 Use Case Description – User Management

Use Case Description – Password Change

Refer the ‘Table A.2’ for the password change use case description

User Case ID	002
Use Case	Password Change
Actor(s)	Patient, Doctor, Cashier, Pharmacist, Administrator
Description	
All the registered users can be changed their current password	
Pre-conditions	
1.User credentials have been validated 2.User Should be connected to the internet	
Normal Cause	

1. Enter password details 2. Click Change button 3. User will be successfully logged out from the system and navigate back to login page.
Alternative Causes
1. If the invalid password format entered, a relevant error message will be displayed. 2.If the mandatory fields have not been filled, a relevant error message will be displayed. 3.If the password won't be matched, a relevant error message will be displayed.
Post-conditions
1.User can be logged into the system with the changed password

Table A.2 Use Case Description – Change Password

Use Case Description – Login

Refer the 'Table A.3' for the login use case description

User Case ID	003
Use Case	Login
Actor(s)	Patient, Doctor, Cashier, Pharmacist, Administrator
Description	
All the registered users can be logged into the system and execute functions that allowed to that particular user.	
Pre-conditions	
1.User credentials have been validated 2.User Should be connected to the internet	
Normal Cause	
1. Enter user name and password 2. Click Login button 3. User will be successfully logged and navigate to home page.	
Alternative Causes	
1. If the login credentials are invalid, a relevant error message will be displayed.	
Post-conditions	
1.User logged into the system 2.User can proceed to system functions	

Table A.3 Use Case Description – Login

Use Case Description – Member Registration

Refer the 'Table A.4 for the Member registration use case description

User Case ID	004
Use Case	Member Registration
Actor(s)	Patient
Description	
Unregistered users can be restarted here.	
Pre-conditions	
1.User should be unregistered member 2.User Should be connected to the internet	

Normal Cause
1. Enter member details 2. Click Submit button 3. Member will be successfully created.
Alternative Causes
1. If the duplicate user details entered, a relevant error message will be displayed. 2.If the mandatory fields have not been filled, a relevant error message will be displayed. 3.If the password won't be matched, a relevant error message will be displayed. 4.If any field validations fired, a relevant error message will be displayed.
Post-conditions
1.Member will be created successfully. 2.Member will be received an email regarding user details 3.Members can be logged to the system. 4.Members will be granted for permission that has been assigned to the particular user role.

Table A.4 Use Case Description – Member Registration

Use Case Description – Home

Refer the 'Table A.5' for the home use case description

User Case ID	005
Use Case	Home
Actor(s)	Patient -Un registered Members
Description	
All the un registered users can be channeled a doctor via home screen.	
Pre-conditions	
1.User Should be connected to the internet	
Normal Cause	
1. Select doctor name, specialization and date 2. Click Search button 3. User will be successfully navigated to doctor date schedule view page.	
Alternative Causes	
N/A	
Post-conditions	
1.User will be navigated to doctor date schedule view page. 2.User can select an appropriate date for the channel	

Table A.5 Use Case Description – Doctor Channel

Use Case Description – Channel Doctor

Refer the 'Table A.6' for the channel doctor use case description

User Case ID	006
Use Case	Channel Doctor
Actor(s)	Patient -Registered Member
Description	
All the registered users can be channeled a doctor.	

Pre-conditions
1.User should be registered. 2.User should be logged to the system. 2.User Should be connected to the internet.
Normal Cause
1. Select doctor name, specialization and date 2. Click Search button 3.Select a doctor to channel by clicking Channel now button 4. User will be successfully navigated to doctor date schedule view page.
Alternative Causes
N/A
Post-conditions
1.User will be navigated to doctor date schedule view page. 2.User can select an appropriate date for the channel

Table A.6 Use Case Description – Doctor Channel

Use Case Description – Appointment Date Schedule

Refer the 'Table A.7' for the appointment date schedule use case description

User Case ID	007
Use Case	Appointment Date Schedule
Actor(s)	Patient -Registered/unregistered Member
Description	
All the users can be select a date for a particular doctor.	
Pre-conditions	
1.User should select the doctor via channel doctor screen	
Normal Cause	
1.Click available button to proceed the appointment.	
Alternative Causes	
1.If the user clicked the full and holiday dates as appointment date, a relevant error message will be displayed.	
Post-conditions	
1.User will be navigated to patient details filling screen.	

Table A.7 Use Case Description – Appointment Date Schedule

Use Case Description – Patient Details Management

Refer the 'Table A.8' for the patient details use case description

User Case ID	008
Use Case	Patient Details Management
Actor(s)	Patient -Registered/unregistered Member

Description
All the users can be enter patient details
Pre-conditions
1.User should select the date for appointment via date schedule screen
Normal Cause
1.Enter patient details. 2.Click Submit button.
Alternative Causes
1.If the user is a registered user all the relevant data will be auto filled. 2.If the duplicate user details entered, a relevant error message will be displayed. 3.If the mandatory fields have not been filled, a relevant error message will be displayed.
Post-conditions
1.User will be navigated to payment screen.

Table A.8 Use Case Description – Patient Details

Use Case Description – Payment Management

Refer the ‘Table A.9’ for the payment management use case description

User Case ID	009
Use Case	Payment Management
Actor(s)	Patient -Registered/unregistered Members
Description	
All the users can be proceed their payments for the channeling.	
Pre-conditions	
1.User should enter patient details via patient details screen	
Normal Cause	
1.Select payment method 2.Click Pay Button. 3.Enter payment details and proceed	
Alternative Causes	
N/A	
Post-conditions	
1.Payment will be done successfully. 2.Email/SMS will be sent to patient.	

Table A.9 Use Case Description – Payment

Use Case Description – Appointment Confirm

Refer the ‘Table A.10’ for the payment management use case description

User Case ID	010
Use Case	Appointment Confirm
Actor(s)	Cashier

Description
Cashier can enter user/appointment details and confirm it and print the details.
Pre-conditions
1.Appiontment should be successfully completed.
Normal Cause
1.Enter Details 2.Click Search Button. 3.Details will be auto filled 4.Clcik Confirm Appointment button 5.Click print button to print
Alternative Causes
1.If the invalid appointment details entered, relevant error message will be pop upped.
Post-conditions
1.Appointment will be confirmed.

Table A.10 Use Case Description – Appointment Confirm

Use Case Description – Prescription Handling

Refer the ‘Table A.11’ for the payment management use case description

User Case ID	011
Use Case	Prescription Handling
Actor(s)	Doctor
Description	
All the patient prescription, history, details will be handled.	
Pre-conditions	
1.User should log to the system	
Normal Cause	
1.Click Start Session to start the session. 2.Enter all details and click save to save patient details for that particular session 3.Enter drug details and click send button to send that details to pharmacist.	
Alternative Causes	
1. All the session patients will be received an email/SMS regarding session start. 2.Once data send to pharmacy, it will notify. 3.If the drugs are not available in current stock system will notify.	
Post-conditions	
1.Prescription will be sent successfully. 2.User will be notified when his/her no is next.	

Table A.11 Use Case Description – Prescription Handling

Use Case Description – Drug Stock Handling

Refer the ‘Table A.12’ for the payment management use case description

User Case ID	012
Use Case	Drug Stock Handling

Actor(s)	Pharmacist
Description	
Pharmacist can view prescriptions and issue the prescriptions.	
Pre-conditions	
1.User should log to the system 2.Prescription has been received	
Normal Cause	
1.View the prescription details 2.Issue the drugs 3.Print the prescription with drug image and print the external prescription as well. 4.Send the details to cashier.	
Alternative Causes	
1. If the drug stock not available system will be notified.	
Post-conditions	
1.Prescription will be sent successfully to the cashier.	

Table A.12 Use Case Description – Drug Stock Handling

Use Case Description – Cashier Payment Handling

Refer the 'Table A.13' for the payment management use case description

User Case ID	013
Use Case	Cashier Payment Handling
Actor(s)	Cashier
Description	
Cashier can view the payment details and print	
Pre-conditions	
1.User should log to the system 2.Prescription payment details has been received	
Normal Cause	
1.View the prescription payment details 2.print the receipt	
Alternative Causes	
N/A	
Post-conditions	
1.Prescription payment will be completed successfully.	

Table A.13 Use Case Description – Cashier Payment Handling

Use Case Description – Treatment Search

Refer the ‘Table A.14’ for the payment management use case description

User Case ID	014
Use Case	Treatment Search
Actor(s)	Patient
Description	
Patient can search treatment details and chat online.	
Pre-conditions	
1.User should be connected to internet connection.	
Normal Cause	
1.Enter Treatment details 2.Click Search button 3.Search details will be viewed.	
Alternative Causes	
1.If there are no search details, error message will be pop upped. 2.User can open the chat box and chat regrading treatments, drugs, illness etc.	
Post-conditions	
1.Search Results will be viewed.	

Table A.14 Use Case Description – Treatment Search

Use Case Description –Drug Store Search

Refer the ‘Table A.15’ for the payment management use case description

User Case ID	015
Use Case	Drug Store Search
Actor(s)	Patient
Description	
Patient can search drug store details	
Pre-conditions	
1.User should be connected to internet connection.	
Normal Cause	
1.Enter drug store details 2.Click Search button 3.Search details will be viewed.	
Alternative Causes	
1.If there are no search details, error message will be pop upped.	
Post-conditions	
1.Search Results will be viewed.	

Table A.15 Use Case Description – Drug Store Search

Sequence Diagrams

The Figure A.16 shows the Sequence diagram [26] for the Consultation Process.

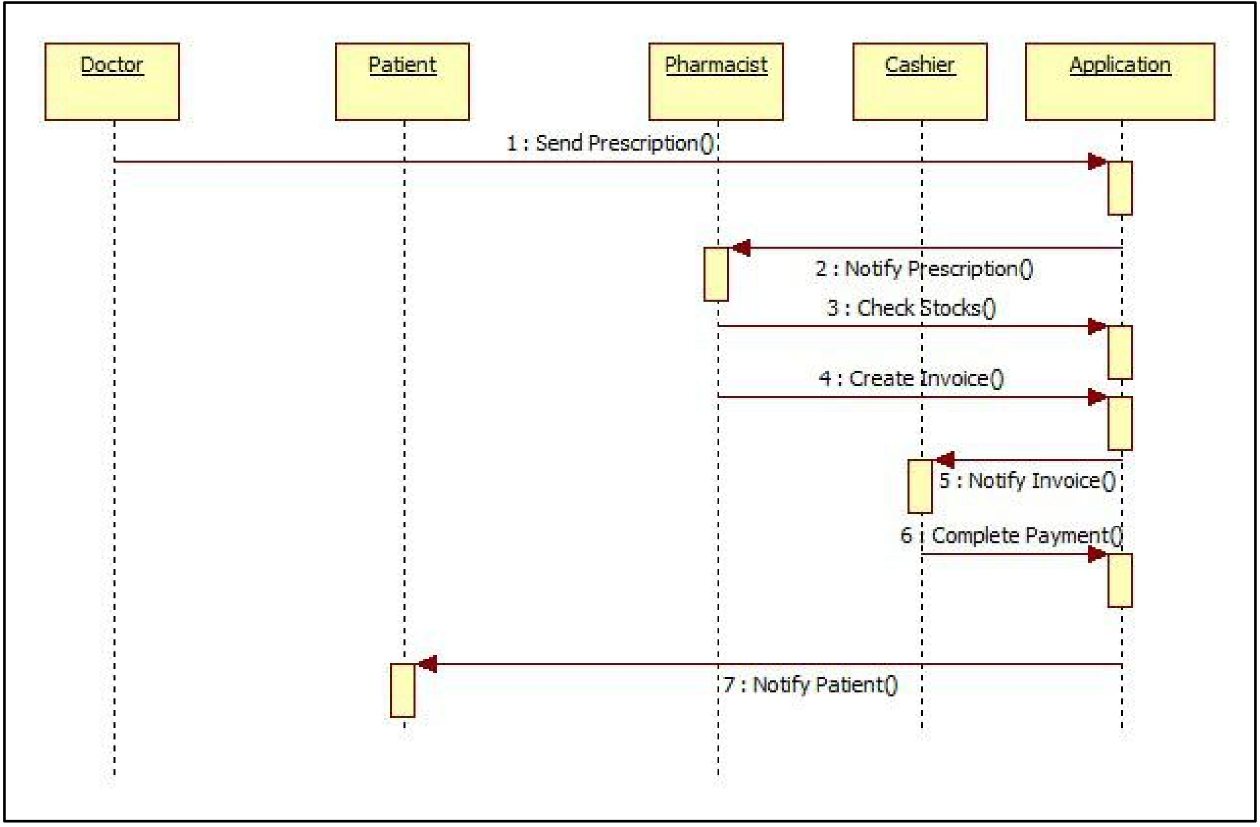


Table A.16 Sequence Diagram – Consultation Process

Appendix B – User Documentation

The Patient Mangment and Medince Stock Management System which was developed for "Peramuna Ayurvedic Center" is intended to use by four user categories which are Patient, Doctor,Cashier,Pharmacist. For the All the system This manual [27] is guidance for users how to use system features which helps to adhere management expectations.

Web application will be facilitated to operate all the module describe below (for all users) and mobile application will be facilitated only the module include in online appointment process (only for patients). Administrator has all the rights to operate User Management Modules and Report Generation Module only. The main purpose of this Appendix is to include a user guide for the main functions of this System.

Home

The Figure B.1 and B.2 shows the "Home" interface. Any person can be viewed (authorized/unauthorized) the home screen.Users can select doctor name,specification and date to search the doctor details to channel.If the user is already registred user,user can click the sign-in button.Once the user click the sign-in button,user will naviagte to login screen.Also user have the online chatting facility as well.If the user click the treatments,drug locators,about us and contact us buttons,user will navigate to respective pages clicked.

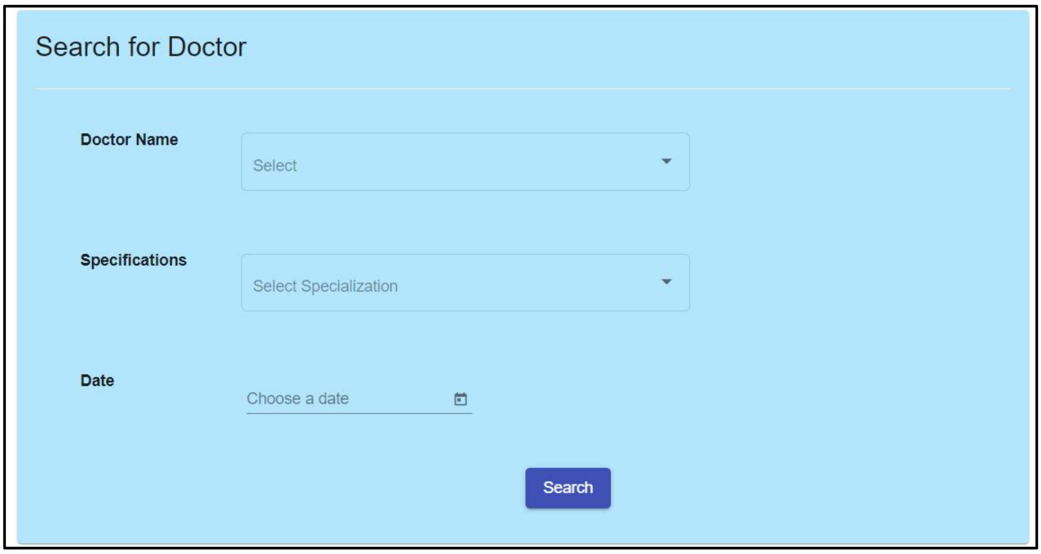


Figure B.1 Home - Home

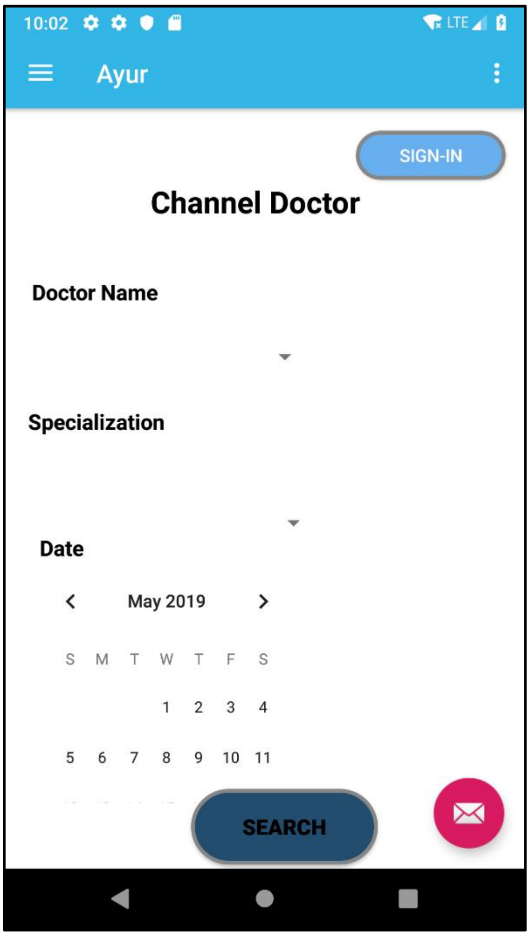
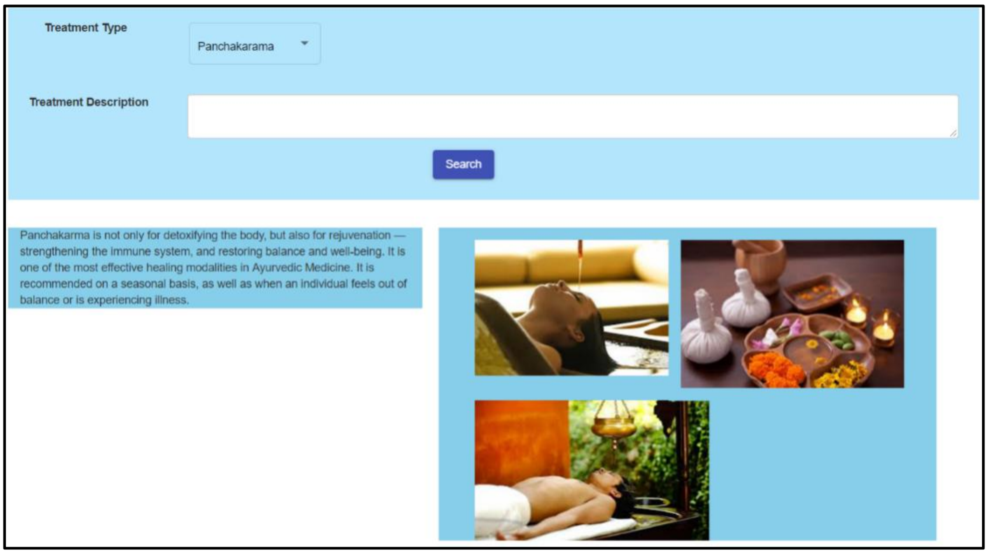


Figure B.2 Home - Mobile

Treatments

The Figure B.3 shows the "Treatment" interface. Any person can be viewed (authorized/unauthorized) the this screen.Users can search about treatments by selecting the tretmentment type and entering descriptions.If search results available result will be viewed.

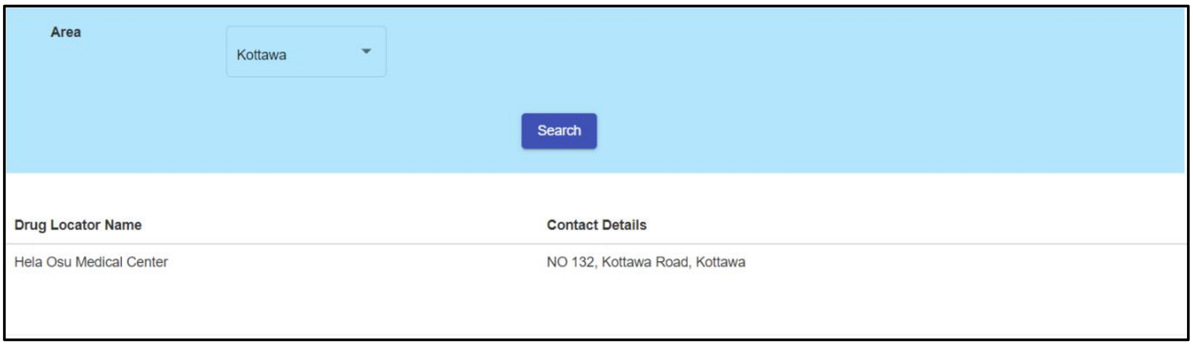


Panchakarma is not only for detoxifying the body, but also for rejuvenation — strengthening the immune system, and restoring balance and well-being. It is one of the most effective healing modalities in Ayurvedic Medicine. It is recommended on a seasonal basis, as well as when an individual feels out of balance or is experiencing illness.

Figure B.3 Treatments –Web

Drug Locators

The Figure B.4 shows the “Drug Locators” interface. Any person can be viewed (authorized/unauthorized) the this screen.Users can search drug locators by selecting the area.If search results available result will be viewed.



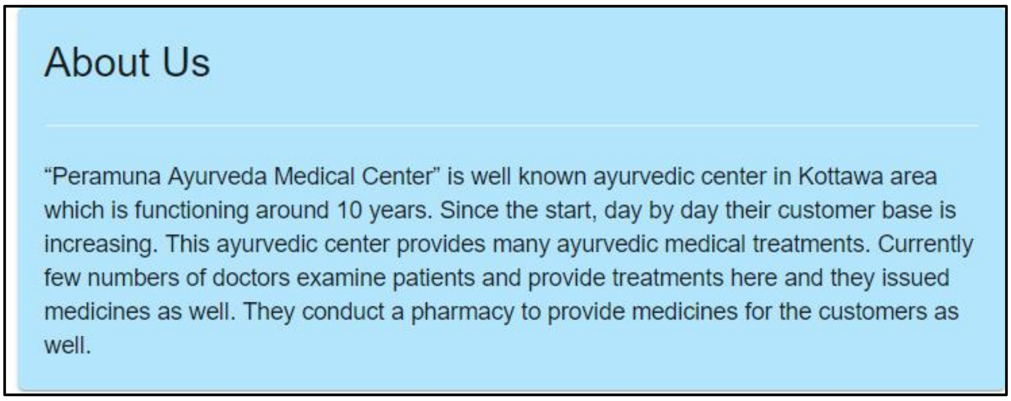
Area: Kottawa

Drug Locator Name	Contact Details
Hela Osu Medical Center	NO 132, Kottawa Road, Kottawa

Figure B.4 Drug Locators –Web

About Us

The Figure B.5 shows the “About Us” interface. Any person can be viewed (authorized/unauthorized) the this screen.Users can view the information about the ayurvedic center by clciking on this button.



About Us

“Peramuna Ayurveda Medical Center” is well known ayurvedic center in Kottawa area which is functioning around 10 years. Since the start, day by day their customer base is increasing. This ayurvedic center provides many ayurvedic medical treatments. Currently few numbers of doctors examine patients and provide treatments here and they issued medicines as well. They conduct a pharmacy to provide medicines for the customers as well.

Figure B.5 About Us –Web

Contact Us

The Figure B.6 shows the “Contact Us” interface. Any person can be viewed (authorized/unauthorized) the this screen.Users can enter own data and send mail to the center.

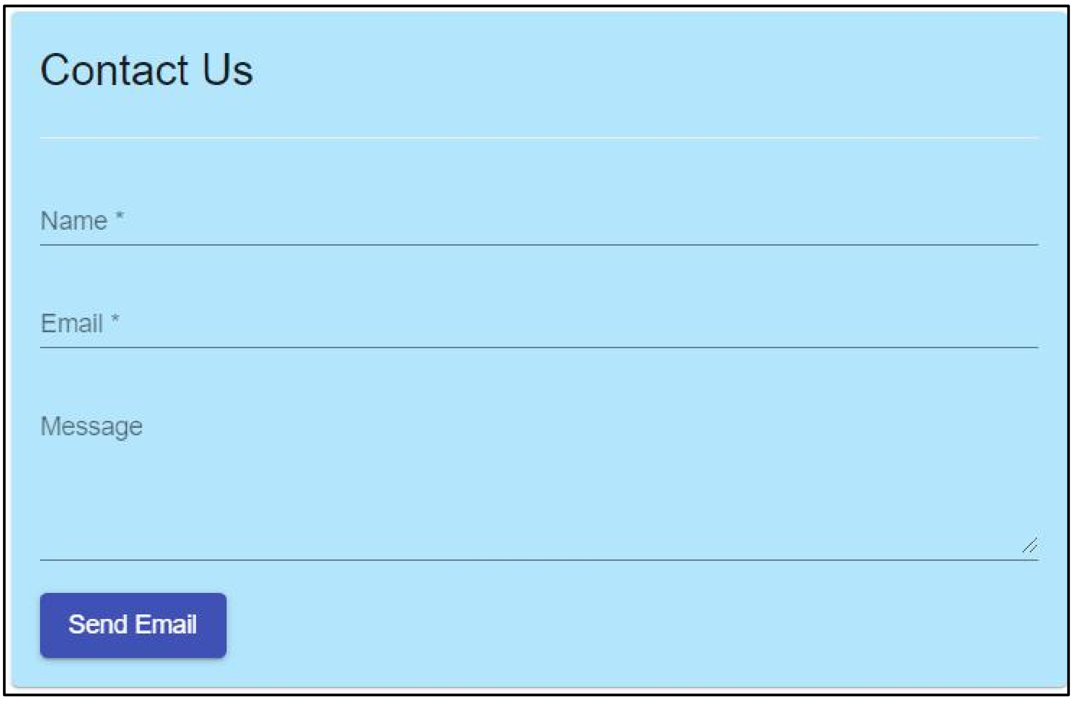


Figure B.6 Contact Us –Web

User Login

The Figure B.7 and B.8 shows the “User Login” interface. Registered patients can be logged to the system by entring valid user name and password.User can remember the login details by checking the remember me check box. Once the user click forget password link user will navigate to change password screen.If the user is unregistered,for the registration can be click the sign-up link which redirect to member registration page.Once the user logged to the system,channel doctor screen will be viewed.

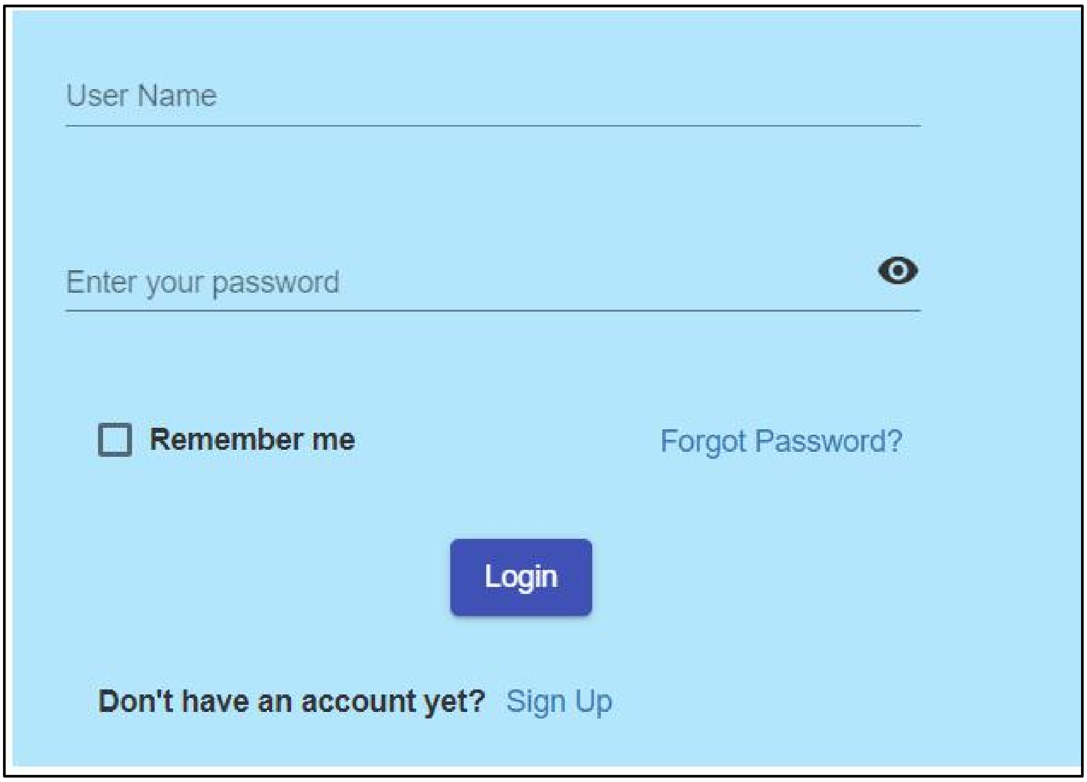


Figure B.7 User Login –Web

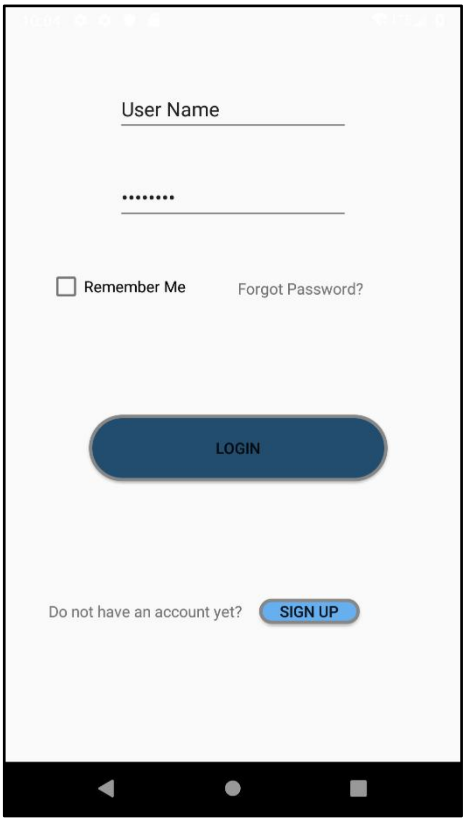


Figure B.8 User Login –Mobile

Member Registration

The Figure B.9 and B.10 shows the “Member Registration” interface. Unregistered patients can be registered via this screen by entering valid data.Once the user submitted the form,user will navigate to login screen.

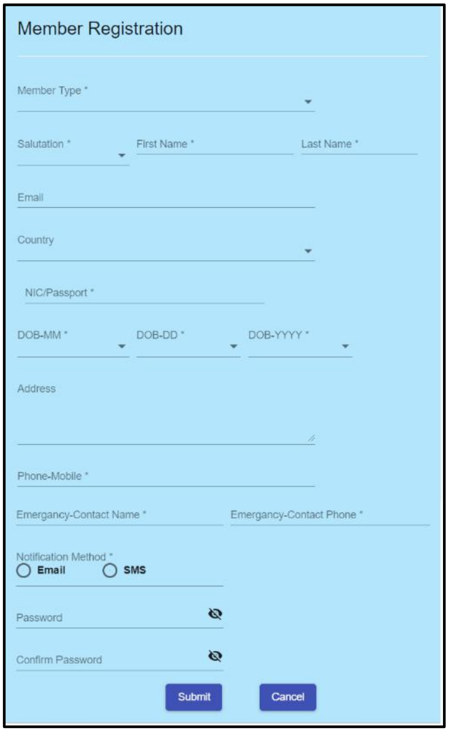


Figure B.9 Member Registration –Web

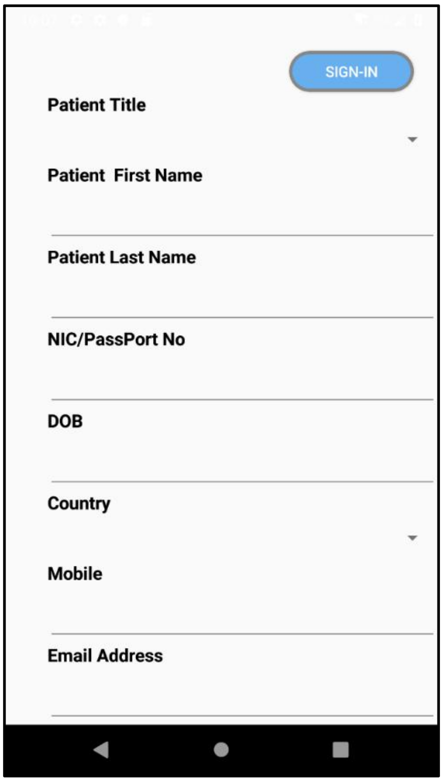


Figure B.10 Member Registration –Mobile

Channel Doctor

The Figure B.11 shows the “Channel Doctor” interface. Paient can be viewed the search results according to the user search critirea.Once the patient click the channel now button ,user will navigate to date view screen.

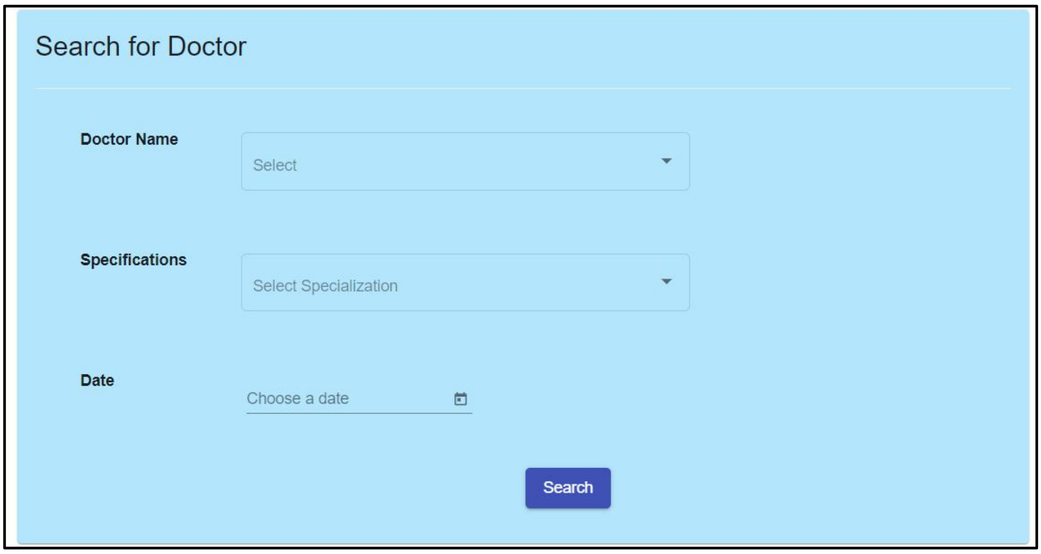
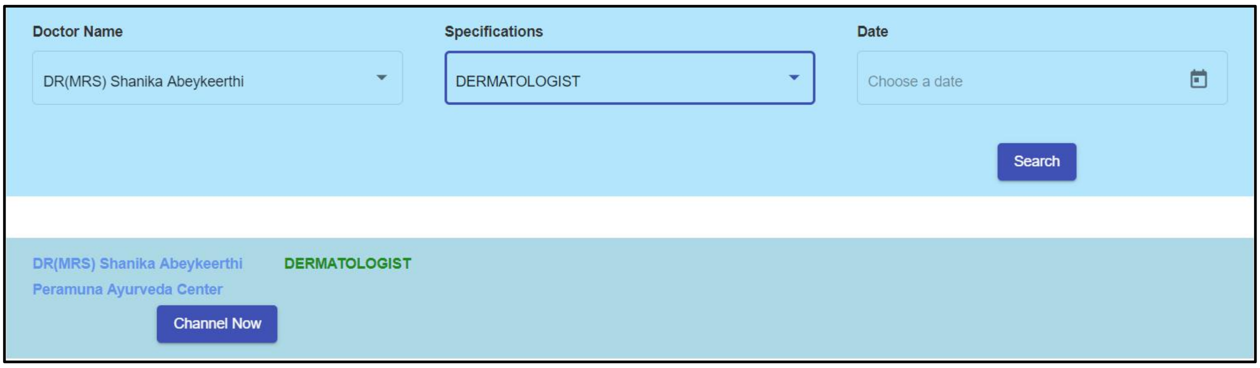


Figure B.11 Channel Doctor –Web

Channel Details View

The Figure B.12 and B.13 shows the “Channel Details View” interface. entring screen.This screen will be viewed all the matching details for the search critirieas.



Doctor Name: DR(MRS) Shanika Abeykeerthi

Specifications: DERMATOLOGIST

Date: Choose a date

DR(MRS) Shanika Abeykeerthi DERMATOLOGIST
Peramuna Ayurveda Center

Figure B.12 Channel Details View –Web

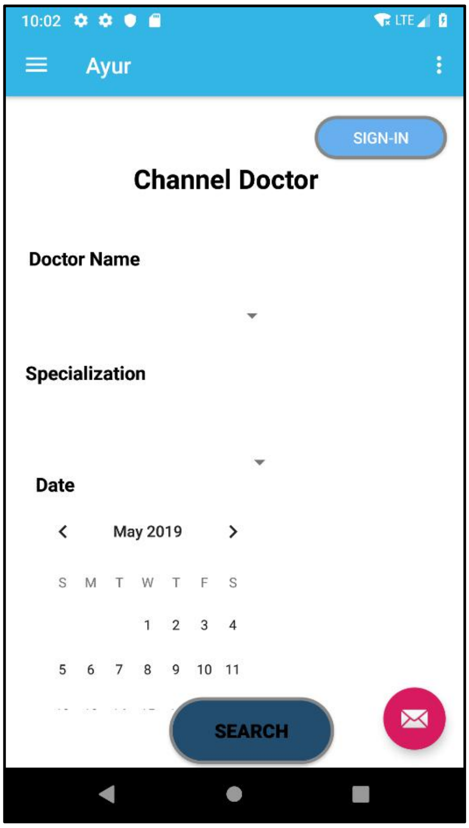
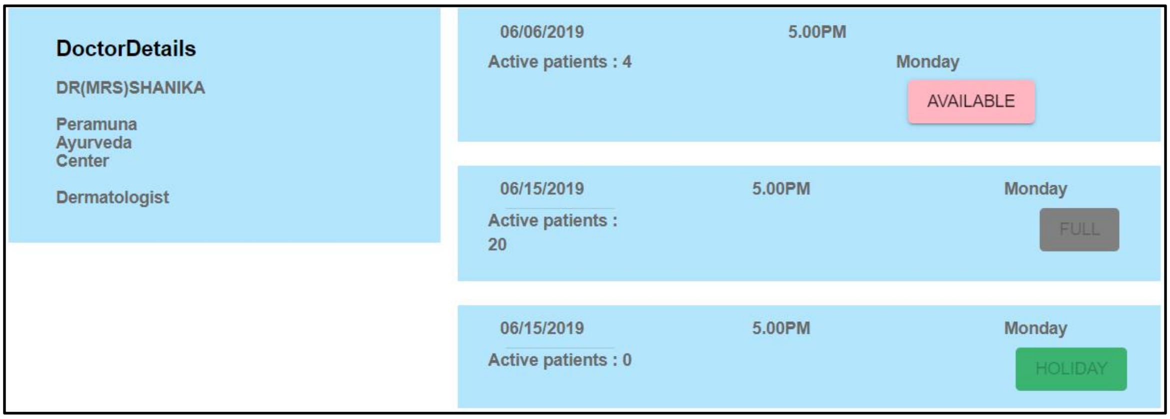


Figure B.13 Channel Details View –Mobile

Channel Date Schedule

The Figure B.14 and B.15 shows the “Channel Date Schedule” interface. All the time slots (avialbe.holiday,full) will be viewed of the search doctor.appointmnet can be schedule only for the avialble time slots.Once the user click the available button,user will navigate to ptient details entring screen.



DoctorDetails

DR(MRS)SHANIKA

Peramuna Ayurveda Center

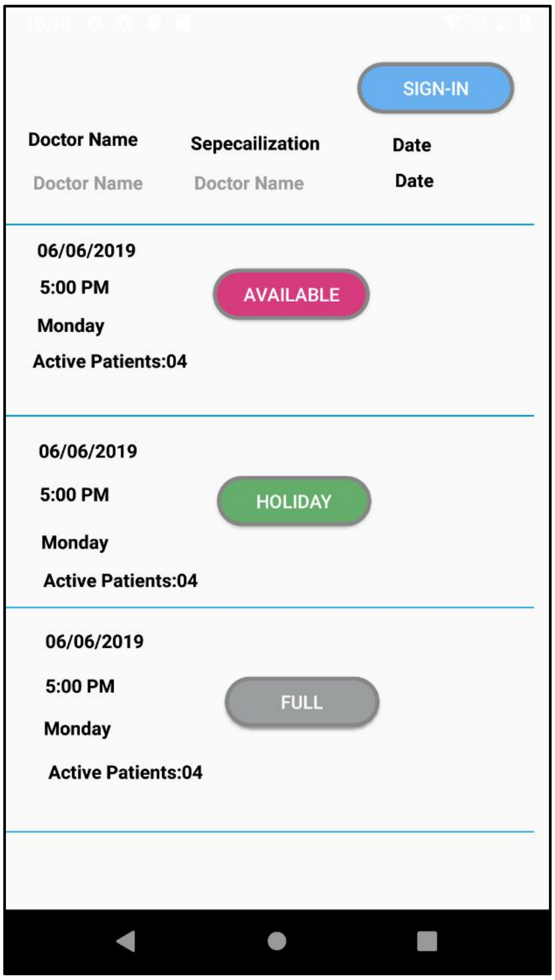
Dermatologist

06/06/2019 5.00PM
Active patients : 4 Monday

06/15/2019 5.00PM Monday
Active patients : 20

06/15/2019 5.00PM Monday
Active patients : 0

Figure B.14 Channel Doctor –Web



Doctor Name	Sepecailization	Date
Doctor Name	Doctor Name	Date

06/06/2019
5:00 PM
Monday
Active Patients:04

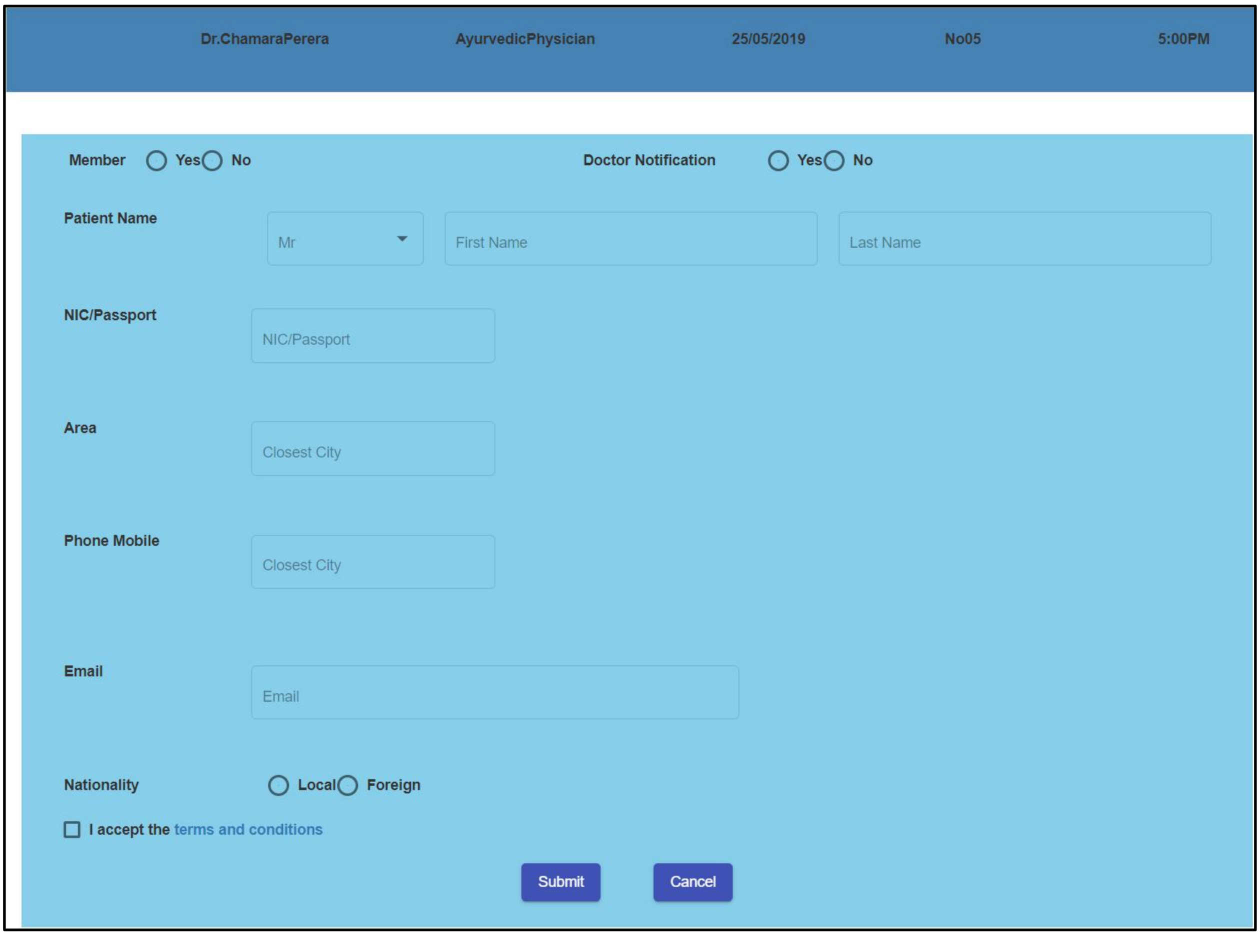
06/06/2019
5:00 PM
Monday
Active Patients:04

06/06/2019
5:00 PM
Monday
Active Patients:04

Figure B.15 Channel Doctor –Mobile

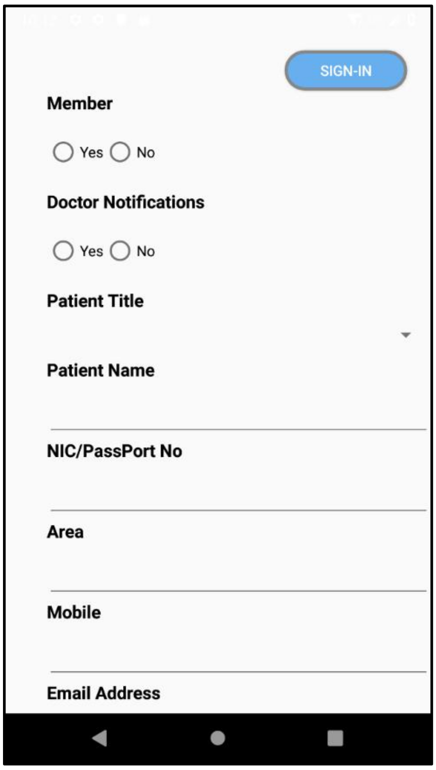
Patient Details Fill

The Figure B.16 and B.17 shows the “Patient Details Fill” interface. If the user is a registered user,all the data will be automatically filled , if not need to enter all details and submit the form.Once the user filled the details,will be navigate to payment screen to proceed payment of the online booking.Once the paymnet is done,Patient will received a email with all the booking details.Patient will be received emial/SMS once the doctor arrived to center and also reminders before the number will be sent.



Dr.ChamaraPerera AyurvedicPhysician 25/05/2019 No05 5:00PM

Figure B.16 Patient Details –Web



Member

○ Yes ○ No

Doctor Notifications

○ Yes ○ No

Patient Title

Patient Name

NIC/PassPort No

Area

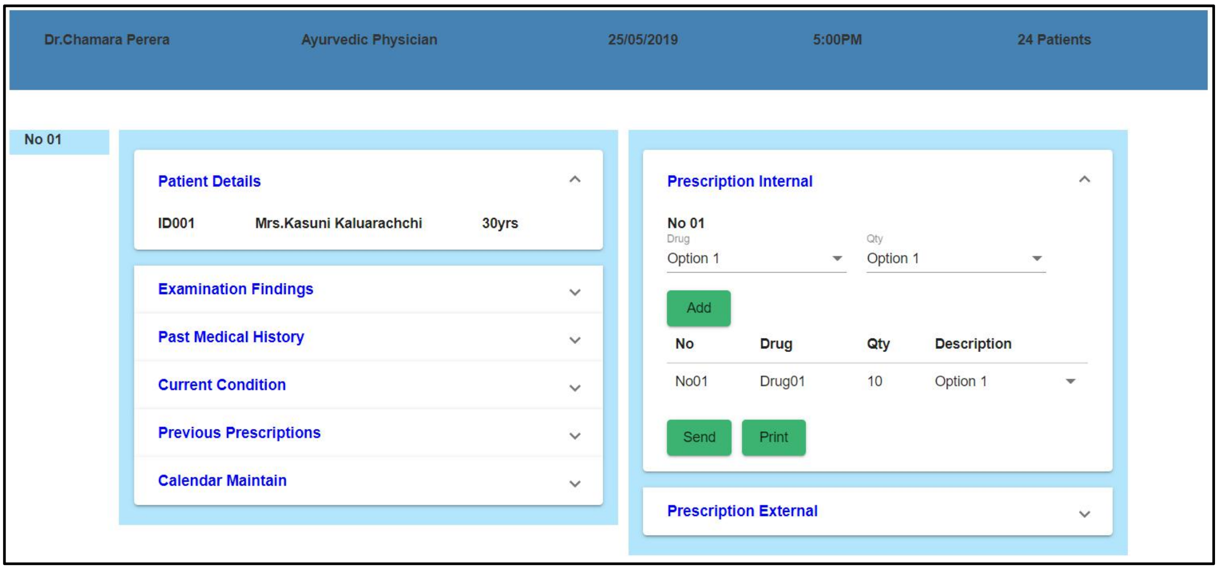
Mobile

Email Address

Figure B.17 Patient Details –Mobile

Doctor Prescription Fill

The Figure B.18 shows the "Doctor Prescription Fill" interface. Doctor can start the session by clicking the start session button.patient no will be viewed and automaticaly changed.Doctor can enter all the patient details and have the option to add drugs to the prescription (internal/external).If a particular drug is not availbe system will be notified and will be added to external prescription list.To send the prescrpition to pharamcy doctor need to click send button.



Dr.Chamara Perera Ayurvedic Physician 25/05/2019 5:00PM 24 Patients

No 01

Patient Details

ID001 Mrs.Kasuni Kaluarachchi 30yrs

Examination Findings

Past Medical History

Current Condition

Previous Prescriptions

Calendar Maintain

Prescription Internal

No 01

Drug: Option 1

Qty: Option 1

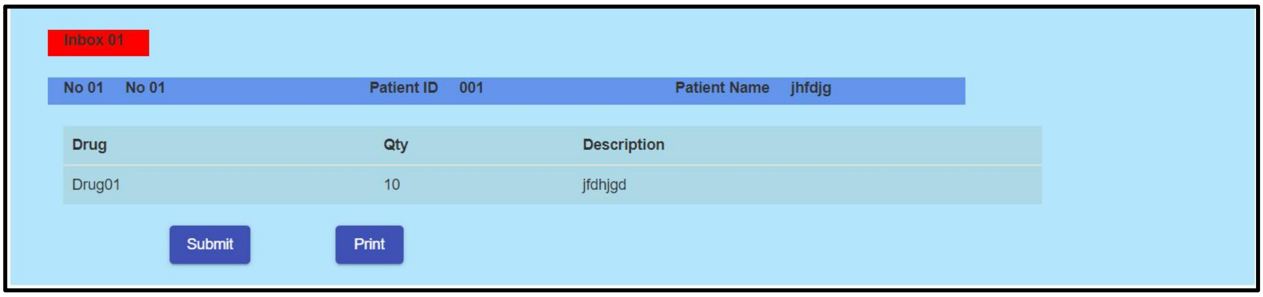
No	Drug	Qty	Description
No01	Drug01	10	Option 1

Prescription External

Figure B.18 Doctor Prescription –Web

Pharmacist Prescription Handeling

The Figure B.19 shows the “Pharmacist Prescription Handeling” interface. Once a prescription is recevied it will be notified to pharmacist.Also phramacist will be recived notifactions from the system regarding drug stocks (out of stock,reorder level).Once the drugs issued to patient ,pharmacist can send the prescriptions to cashier and also have the option to print the prescriptions.



Inbox 01

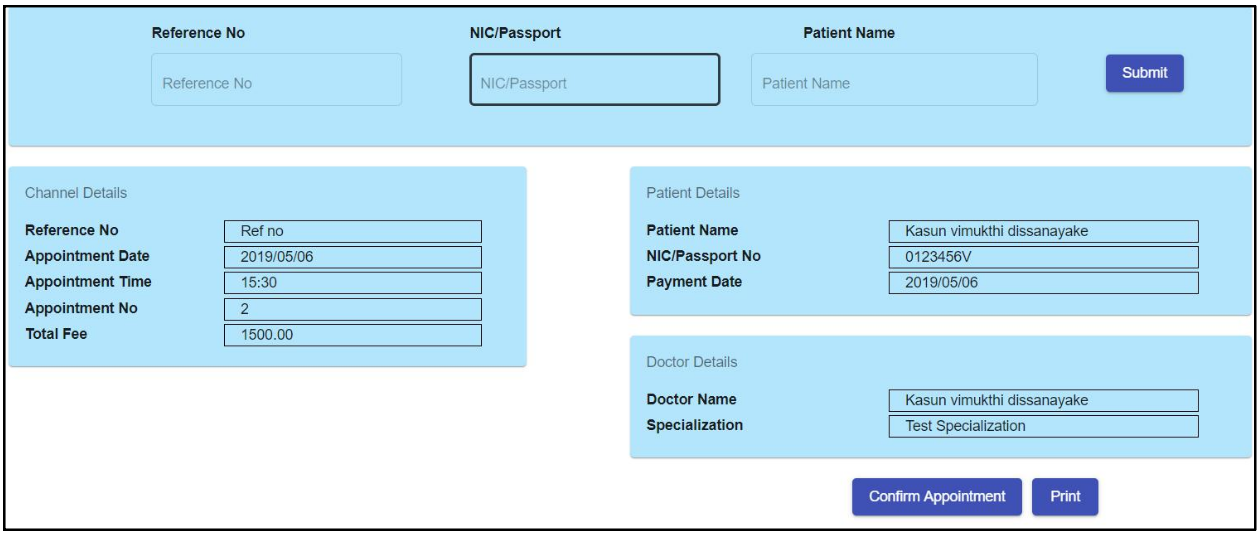
No 01 No 01 Patient ID 001 Patient Name jhfdjg

Drug	Qty	Description
Drug01	10	jfdhjgd

Figure B.19 Doctor Prescription –Web

Channel Confirmation

The Figure B.20 shows the “Channel Confirmation” interface. Once the patient complete the online booking for the doctor appointment patient can go to casheir to confirm the channeling on the appointment day.Once the patient show the email of appoinment, casheir can search the details by reference no,NIC/Passport no,patient name.Cashier can click confirm button to confirm the appoinment and print the appoinmnet for the patient.



Reference No	NIC/Passport	Patient Name
Reference No	NIC/Passport	Patient Name

Channel Details

Reference No	Ref no
Appointment Date	2019/05/06
Appointment Time	15:30
Appointment No	2
Total Fee	1500.00

Patient Details

Patient Name	Kasun vimukthi dissanayake
NIC/Passport No	0123456V
Payment Date	2019/05/06

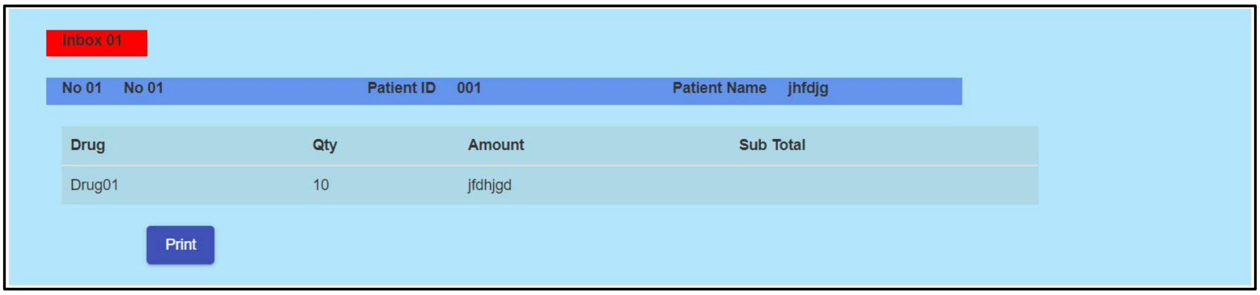
Doctor Details

Doctor Name	Kasun vimukthi dissanayake
Specialization	Test Specialization

Figure B.20 Channel Confirmation –Web

Cashier Payments

The Figure B.21 shows the “Cashier Payments” interface. Once a prescription (sent from phramcist) is recevied it will be notified to cashier.Cashier can print the payment slip.



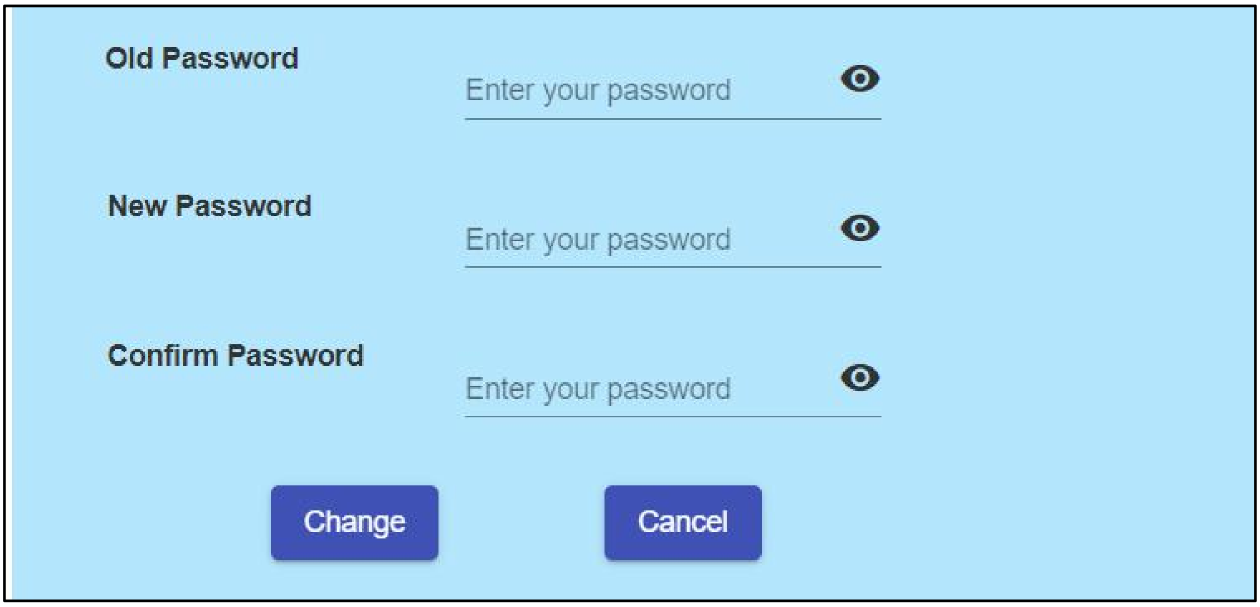
No 01 No 01 Patient ID 001 Patient Name jhfdjg

Drug	Qty	Amount	Sub Total
Drug01	10	jfdhjgd	

Figure B.21 Cashier Payments –Web

Change Password

The Figure B.22 shows the “Change Password” interface. If the user forgot the password ,user can reset the password via this screen.



Old Password: Enter your password

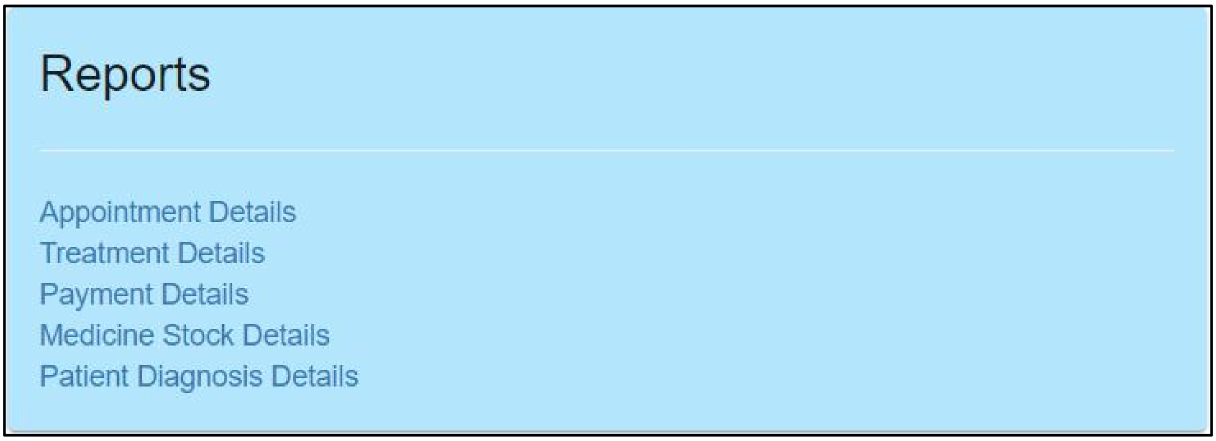
New Password: Enter your password

Confirm Password: Enter your password

Figure B.22 Change Password –Web

Appendix C – Management Reports

Figure C.1 shows the main report [28] view screen.System administrator will be have access to these reports.Once the user click the rleveant report link,user will be navigate to report view screen.



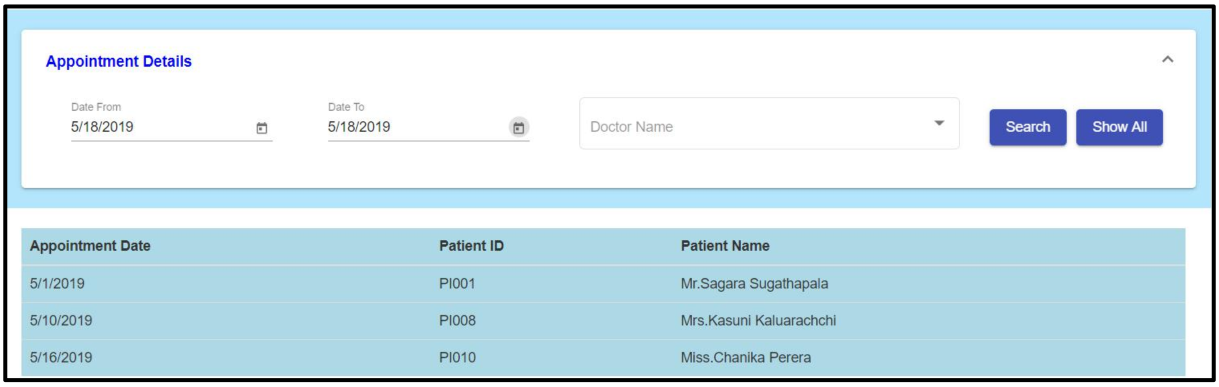
Reports

Appointment Details
Treatment Details
Payment Details
Medicine Stock Details
Patient Diagnosis Details

Figure C.1 Reports View –Web

Appointment Details

The Figure C.2 shows the “Appointment Details -Report” interface. System Administrator and Doctor are authorized to view and print the report.



Appointment Details

Date From: 5/18/2019

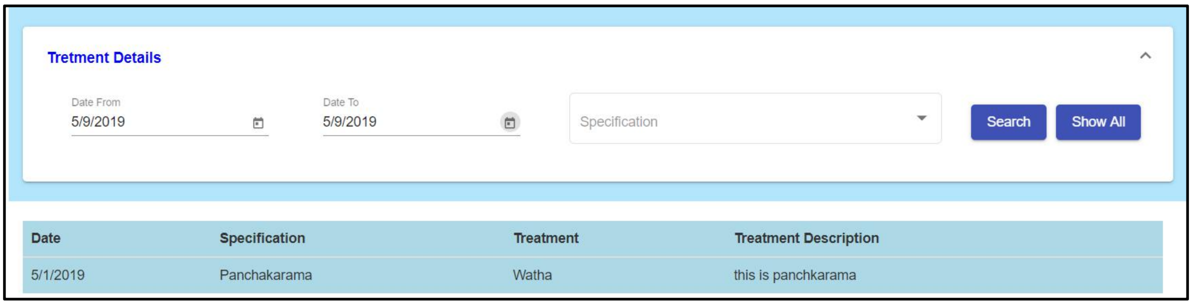
Date To: 5/18/2019

Appointment Date	Patient ID	Patient Name
5/1/2019	PI001	Mr.Sagara Sugathapala
5/10/2019	PI008	Mrs.Kasuni Kaluarachchi
5/16/2019	PI010	Miss.Chanika Perera

Figure C.2 Report–Appointment Details

Treatment Details

The Figure C.3 shows the “Treatment Details -Report” interface. System Administrator and Doctor are authorized to view and print the report.



Tretment Details

Date From 5/9/2019

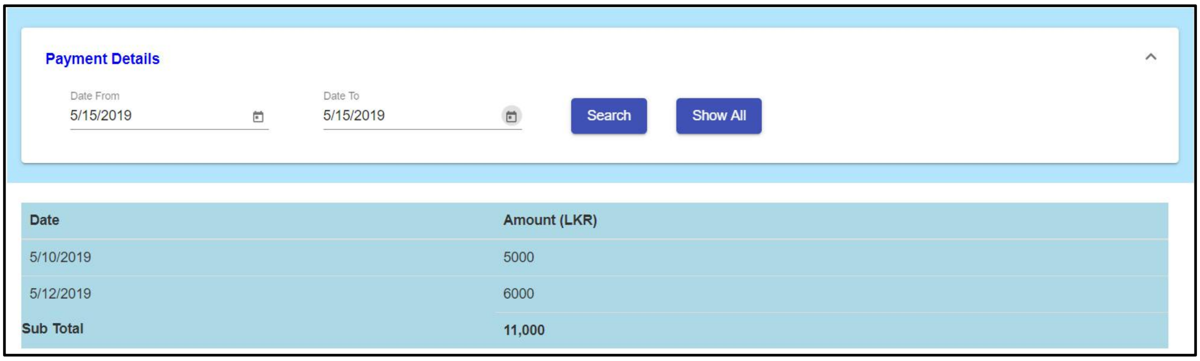
Date To 5/9/2019

Date	Specification	Treatment	Treatment Description
5/1/2019	Panchakarama	Watha	this is panchkarama

Figure C.3 Reports–Treatment Details

Payment Details

The Figure C.4 shows the “Payme Details -Report” interface. System Administrator and Cashier are authorized to view and print the report.



Payment Details

Date From 5/15/2019

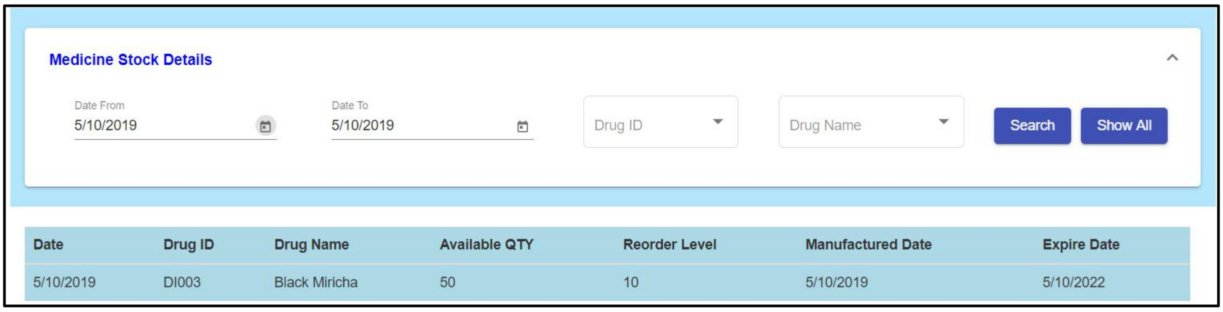
Date To 5/15/2019

Date	Amount (LKR)
5/10/2019	5000
5/12/2019	6000
Sub Total	**11,000**

Figure C.4 Reports–Payment Details

Medicine Stock Details

The Figure C.5 shows the “Medicine Stock Details -Report” interface. System Administrator and Cashier are authorized to view and print the report.



Medicine Stock Details

Date From 5/10/2019

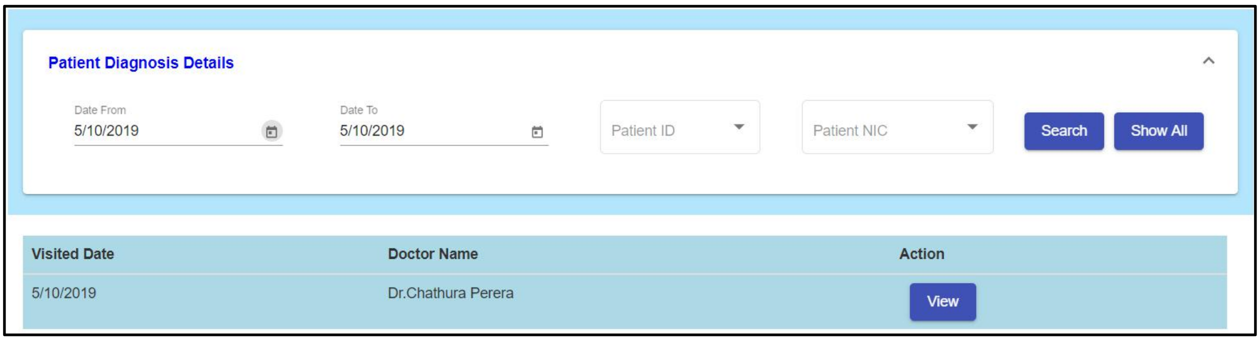
Date To 5/10/2019

Date	Drug ID	Drug Name	Available QTY	Reorder Level	Manufactured Date	Expire Date
5/10/2019	DI003	Black Miricha	50	10	5/10/2019	5/10/2022

Figure C.5 Reports–Medicine Stock Details

Patient Diagnosis Details

The Figure C.6 shows the “Patient Diagnosis Details -Report” interface. System Administrator and Cashier are authorized to view and print the report.



Patient Diagnosis Details

Date From: 5/10/2019

Date To: 5/10/2019

Visited Date	Doctor Name	Action
5/10/2019	Dr.Chathura Perera	View

Figure C.6 Reports–Patient Diagnosis Details

Appendix D – Test Results

Mobile Testing

Table D.1 represents the Test results of the Mobile Testing.

Test Case ID	Test Scenario	Test Case	Expected Results	Actual Result
				Phase 4
M01	Mobile Network Coverage Testing	2G Mobile coverage	All system functions can be executed	Passed
		3G Mobile coverage	All system functions can be executed	Passed
		4G Mobile coverage	All system functions can be executed	Passed
M02	Data Connectivity	WIFI	All system functions can be executed	Passed
		Mobile Data	All system functions can be executed	Passed
M03	Background Applications Execution	Open multiple applications in the background while using the app	All system functions can be executed	Passed
M04	Phone Call Receiving	While Using the application, received an incoming call	All system functions can be executed	Passed
M05	Hardware Testing	Test application in different phone models (OS -Android)	All system functions can be executed	Passed

Table D.1 Test Results –Mobile

Appendix E – System Documentation

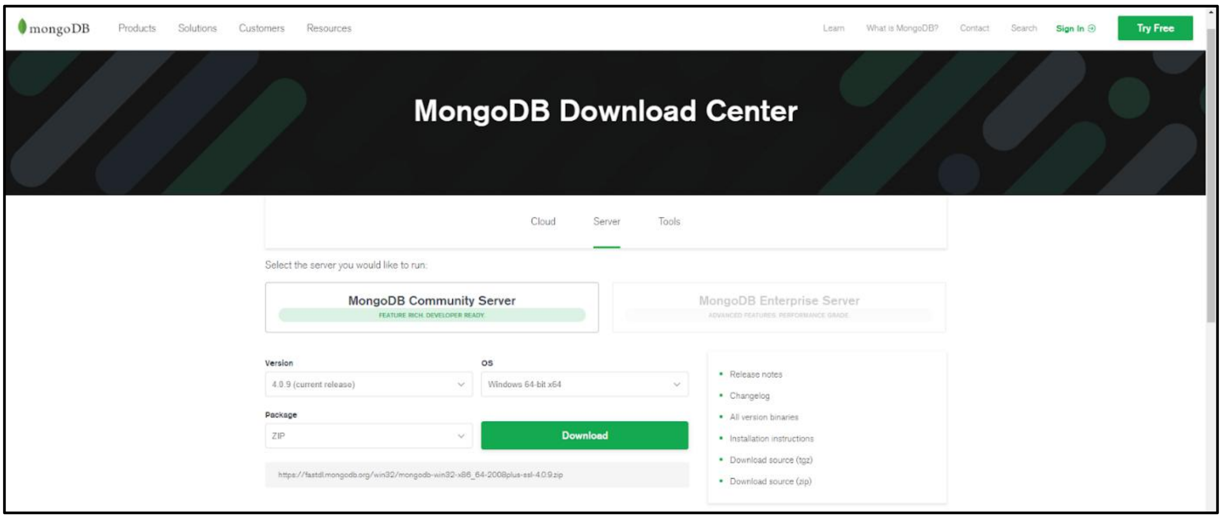
This System Documentation provide the guidelines to install the Ayur web service backend and the respective frontend in a relevant server pc. Since the system is created as a web-based software there will be no special configuration required for the user to access the system other than a working internet connection.
Although the following Requirements should be available with the server to host both frontend and the backend services.

- Openjdk 11
- Mongodb
- Nginx

Below described the process to configuring each item in deployment environment and how to start all the services

Configuring Deployment Environment: DB Setup

Step 1: Navigate to https://www.mongodb.com/download-center/community and download the mongo DB 4.0.9 [29] , When downloading get the zip file rather than msi setup



Select the server you would like to run:

Version: 4.0.9 (current release)

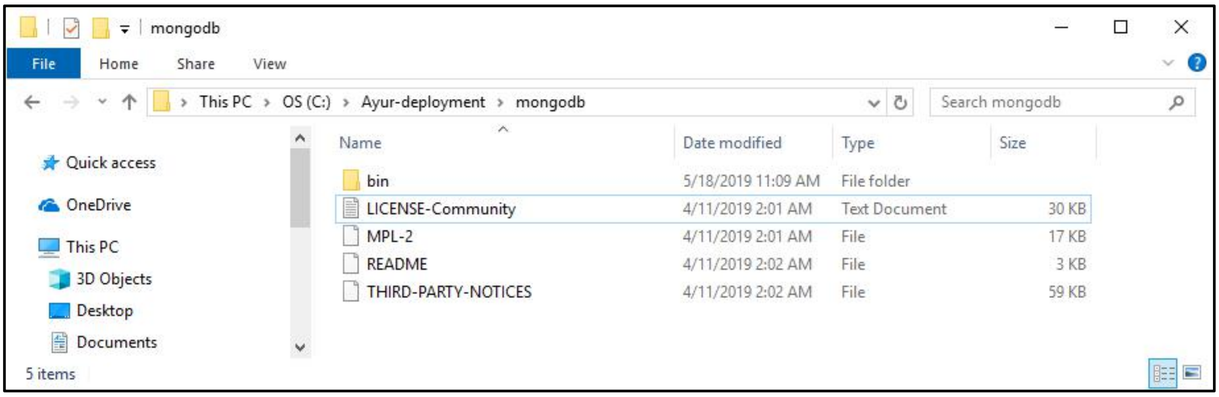
OS: Windows 64-bit x64

Package: ZIP

https://fastdl.mongodb.org/win32/mongodb-win32-x86_64-2008plus-ssl-4.0.9.zip

Figure E.1 Download Mongo DB

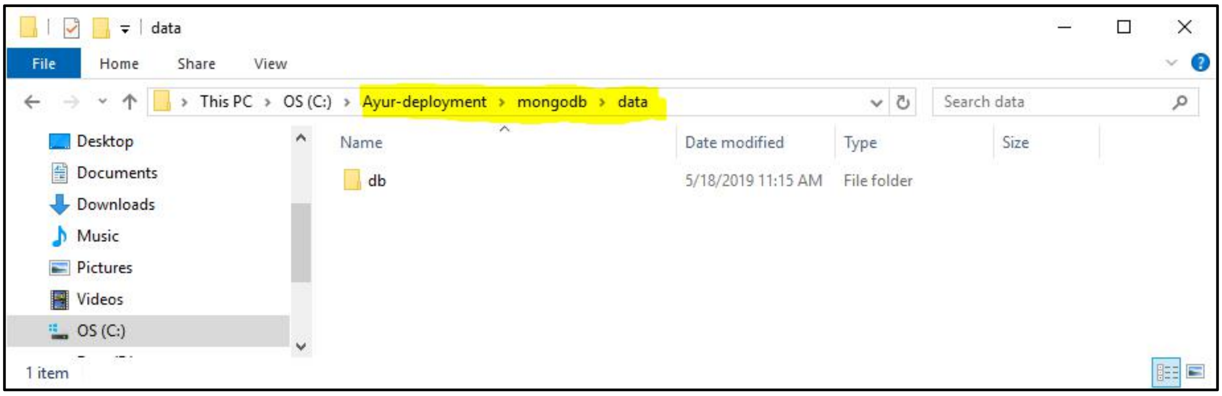
Step 2: extract the downloaded zip file to the deployment location and rename the folder in appropriate manner as shown in Figure E.2. Then create data\db folder in the extracted location as shown in Figure E.3.



This PC › OS (C:) › Ayur-deployment › mongodb

Name	Date modified	Type	Size
bin	5/18/2019 11:09 AM	File folder	
LICENSE-Community	4/11/2019 2:01 AM	Text Document	30 KB
MPL-2	4/11/2019 2:01 AM	File	17 KB
README	4/11/2019 2:02 AM	File	3 KB
THIRD-PARTY-NOTICES	4/11/2019 2:02 AM	File	59 KB

Figure E.2 Extracted Zip File Content

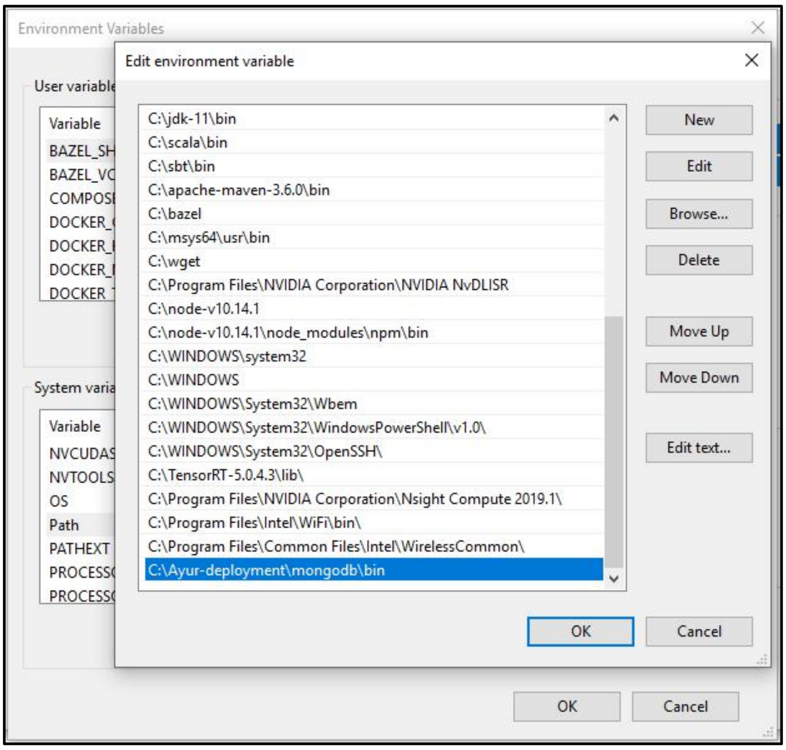


This PC › OS (C:) › Ayur-deployment › mongodb › data

Name	Date modified	Type	Size
db	5/18/2019 11:15 AM	File folder	

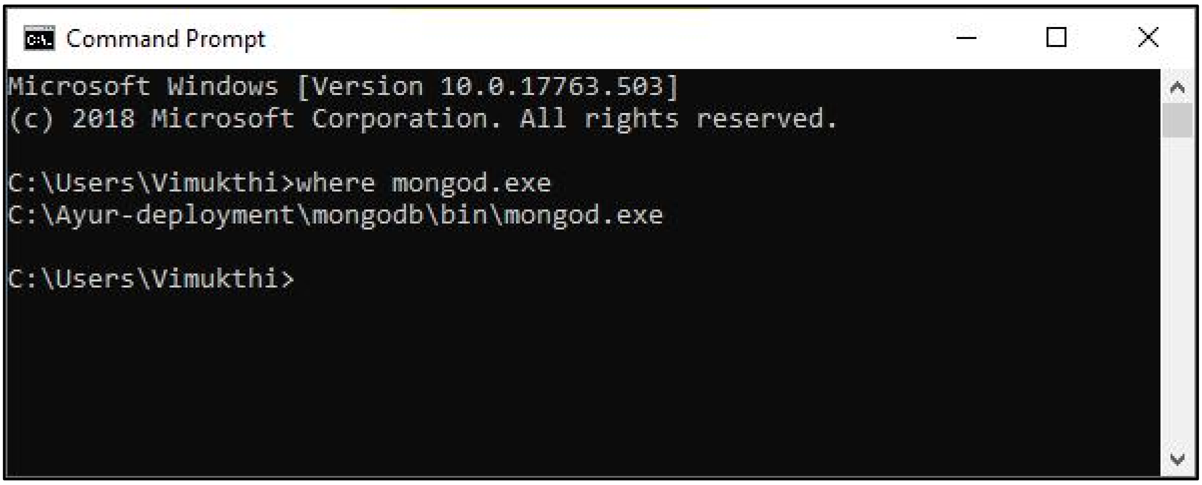
Figure E.3 Create Data folders

Step 3: Add the mongodb/bin path to the windows Path variable as shown in Figure E.4 and verify that the path is properly set by executing the command "where mongod.exe" in a terminal window as shown in Figure E.5.



```
C:\jdk-11\bin
C:\scala\bin
C:\sbt\bin
C:\apache-maven-3.6.0\bin
C:\bazel
C:\msys64\usr\bin
C:\wget
C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR
C:\node-v10.14.1
C:\node-v10.14.1\node_modules\npm\bin
C:\WINDOWS\system32
C:\WINDOWS
C:\WINDOWS\System32\Wbem
C:\WINDOWS\System32\WindowsPowerShell\v1.0\
C:\WINDOWS\System32\OpenSSH\
C:\TensorRT-5.0.4.3\lib\
C:\Program Files\NVIDIA Corporation\Nsight Compute 2019.1\
C:\Program Files\Intel\WiFi\bin\
C:\Program Files\Common Files\Intel\WirelessCommon\
C:\Ayur-deployment\mongodb\bin
```

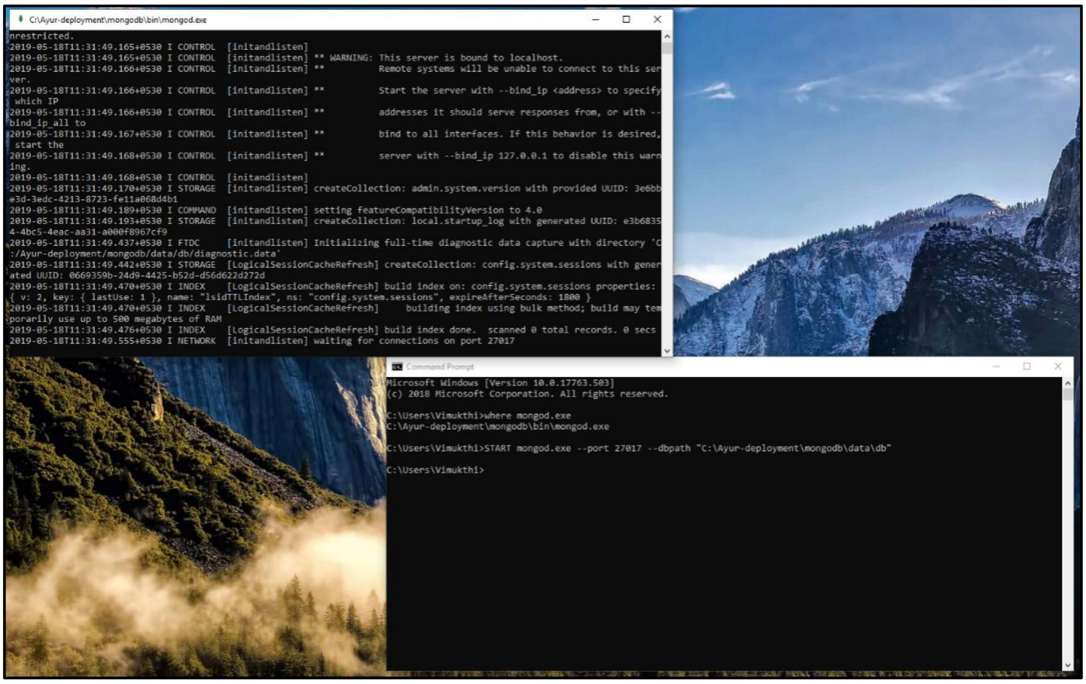
Figure E.4 Update Windows Path Variable



```
Microsoft Windows [Version 10.0.17763.503]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Vimukthi>where mongod.exe
C:\Ayur-deployment\mongodb\bin\mongod.exe

C:\Users\Vimukthi>
```

Figure E.5 Verify Path

Step 4: Start the mongodb server from terminal using the following command,
START mongod.exe --port 27017 --dbpath "C:\Ayur-deployment\mongodb\data\db"
Once it’s started you can see a separate terminal window which is running the mongo server (Figure E-6)



```
nrestricted.
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten]
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten] ** WARNING: This server is bound to localhost.
2019-05-18T11:31:49.166+0530 I CONTROL  [initandlisten] **          Remote systems will be unable to connect to this ser
ver.
2019-05-18T11:31:49.166+0530 I CONTROL  [initandlisten] **          Start the server with --bind_ip <address> to specify
 which IP
2019-05-18T11:31:49.166+0530 I CONTROL  [initandlisten] **          addresses it should serve responses from, or with --
bind_ip_all to
2019-05-18T11:31:49.167+0530 I CONTROL  [initandlisten] **          bind to all interfaces. If this behavior is desired,
 start the
2019-05-18T11:31:49.168+0530 I CONTROL  [initandlisten] **          server with --bind_ip 127.0.0.1 to disable this warn
ing.
2019-05-18T11:31:49.168+0530 I CONTROL  [initandlisten]
2019-05-18T11:31:49.170+0530 I STORAGE  [initandlisten] createCollection: admin.system.version with provided UUID: 3e6bb
e3d-3edc-4213-8723-fe11a068d4b1
2019-05-18T11:31:49.189+0530 I COMMAND  [initandlisten] setting featureCompatibilityVersion to 4.0
2019-05-18T11:31:49.193+0530 I STORAGE  [initandlisten] createCollection: local.startup_log with generated UUID: e3b6835
4-4bc5-4eac-aa31-a000f8967cf9
2019-05-18T11:31:49.437+0530 I FTDC     [initandlisten] Initializing full-time diagnostic data capture with directory 'C
:/Ayur-deployment/mongodb/data/db/diagnostic.data'
2019-05-18T11:31:49.442+0530 I STORAGE  [LogicalSessionCacheRefresh] createCollection: config.system.sessions with gener
ated UUID: 0669359b-24d9-4425-b52d-d56d622d272d
2019-05-18T11:31:49.470+0530 I INDEX    [LogicalSessionCacheRefresh] build index on: config.system.sessions properties:
{ v: 2, key: { lastUse: 1 }, name: "lsidTTLIndex", ns: "config.system.sessions", expireAfterSeconds: 1800 }
2019-05-18T11:31:49.470+0530 I INDEX    [LogicalSessionCacheRefresh]     building index using bulk method; build may tem
porarily use up to 500 megabytes of RAM
2019-05-18T11:31:49.476+0530 I INDEX    [LogicalSessionCacheRefresh] build index done.  scanned 0 total records. 0 secs
2019-05-18T11:31:49.555+0530 I NETWORK  [initandlisten] waiting for connections on port 27017
```

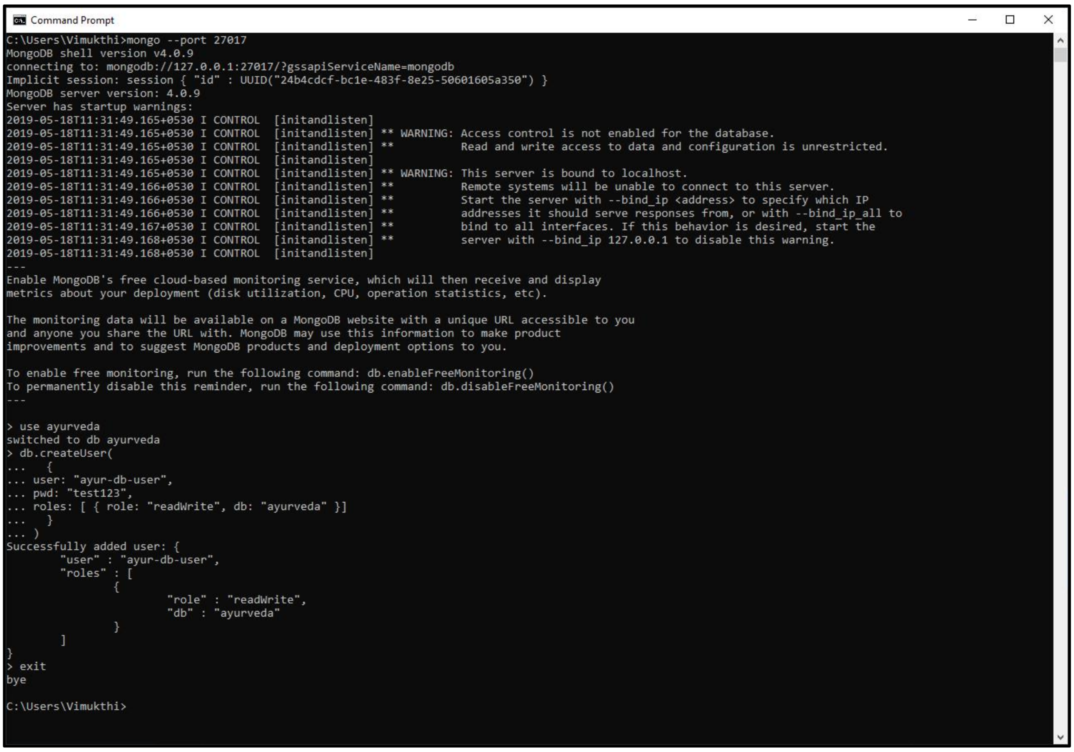
```
Microsoft Windows [Version 10.0.17763.503]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Vimukthi>where mongod.exe
C:\Ayur-deployment\mongodb\bin\mongod.exe

C:\Users\Vimukthi>START mongod.exe --port 27017 --dbpath "C:\Ayur-deployment\mongodb\data\db"

C:\Users\Vimukthi>
```

Figure E.6 Running Mongo DB Server

Step 5: Connect to the mongo server and create the relevant user account for the DB server [30] by using following commands (Figure E.7 and Figure E.8).

```
use ayurveda
db.createUser(
  {
        user: "ayur-db-user",
        pwd: "test123",
        roles: [ { role: "readWrite", db: "ayurveda" }]
  }
)
```

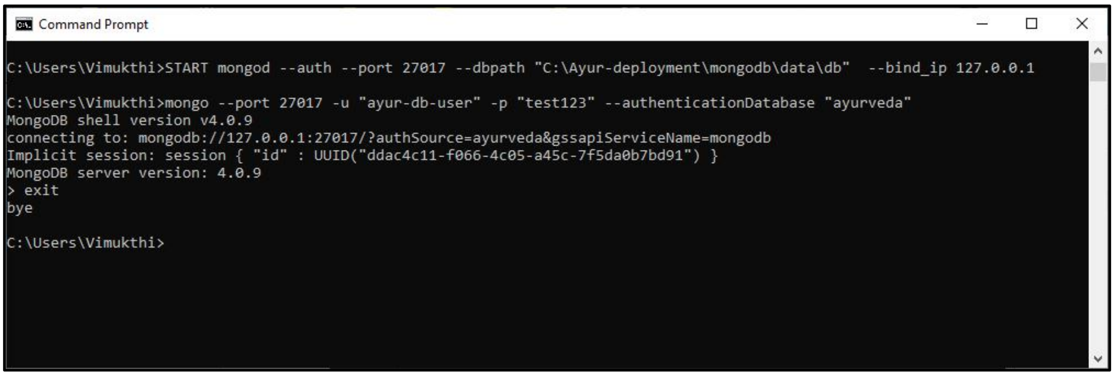
Figure E.7 Create User Account for server



```
C:\Users\Vimukthi>mongo --port 27017
MongoDB shell version v4.0.9
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("24b4cdcf-bc1e-483f-8e25-50601605a350") }
MongoDB server version: 4.0.9
Server has startup warnings:
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten]
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten]
2019-05-18T11:31:49.165+0530 I CONTROL  [initandlisten] ** WARNING: This server is bound to localhost.
2019-05-18T11:31:49.166+0530 I CONTROL  [initandlisten] **          Remote systems will be unable to connect to this server.
2019-05-18T11:31:49.166+0530 I CONTROL  [initandlisten] **          Start the server with --bind_ip <address> to specify which IP
2019-05-18T11:31:49.166+0530 I CONTROL  [initandlisten] **          addresses it should serve responses from, or with --bind_ip_all to
2019-05-18T11:31:49.167+0530 I CONTROL  [initandlisten] **          bind to all interfaces. If this behavior is desired, start the
2019-05-18T11:31:49.168+0530 I CONTROL  [initandlisten] **          server with --bind_ip 127.0.0.1 to disable this warning.
2019-05-18T11:31:49.168+0530 I CONTROL  [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---

> use ayurveda
switched to db ayurveda
> db.createUser(
...   {
... user: "ayur-db-user",
... pwd: "test123",
... roles: [ { role: "readWrite", db: "ayurveda" }]
...   }
... )
Successfully added user: {
        "user" : "ayur-db-user",
        "roles" : [
                {
                        "role" : "readWrite",
                        "db" : "ayurveda"
                }
        ]
}
> exit
bye

C:\Users\Vimukthi>
```

Figure E.8 Create User or Server

Step 6: Once the user is created shutdown the db server and restart the server with authentication enabled by using following commands and when the server is started verify the created user can login to the db server (Figure E.9).

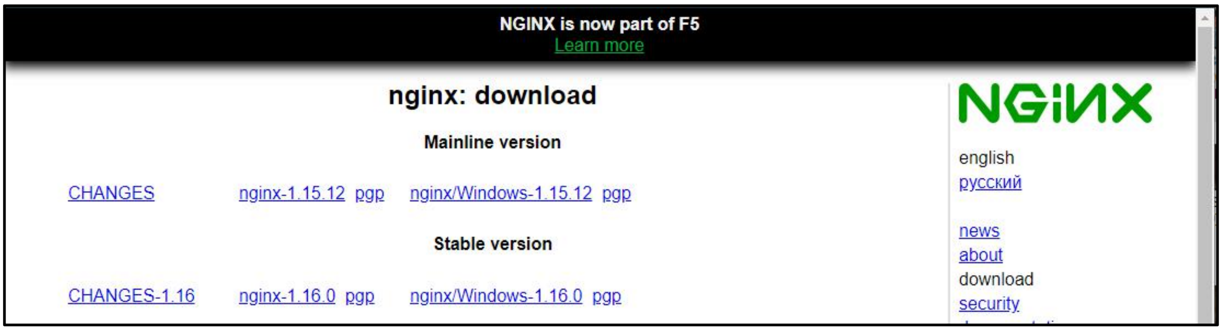


```
C:\Users\Vimukthi>START mongod --auth --port 27017 --dbpath "C:\Ayur-deployment\mongodb\data\db"  --bind_ip 127.0.0.1

C:\Users\Vimukthi>mongo --port 27017 -u "ayur-db-user" -p "test123" --authenticationDatabase "ayurveda"
MongoDB shell version v4.0.9
connecting to: mongodb://127.0.0.1:27017/?authSource=ayurveda&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("ddac4c11-f066-4c05-a45c-7f5da0b7bd91") }
MongoDB server version: 4.0.9
> exit
bye

C:\Users\Vimukthi>
```

Figure E.9 Start Secure DB Server and Test Connection

Configuring Deployment Environment: Web Server Setup

Step 1: Navigate to http://nginx.org/en/download.html and download the nginx 1.16.0 [31](Figure E-10).



NGINX is now part of F5
Learn more

nginx: download

Mainline version

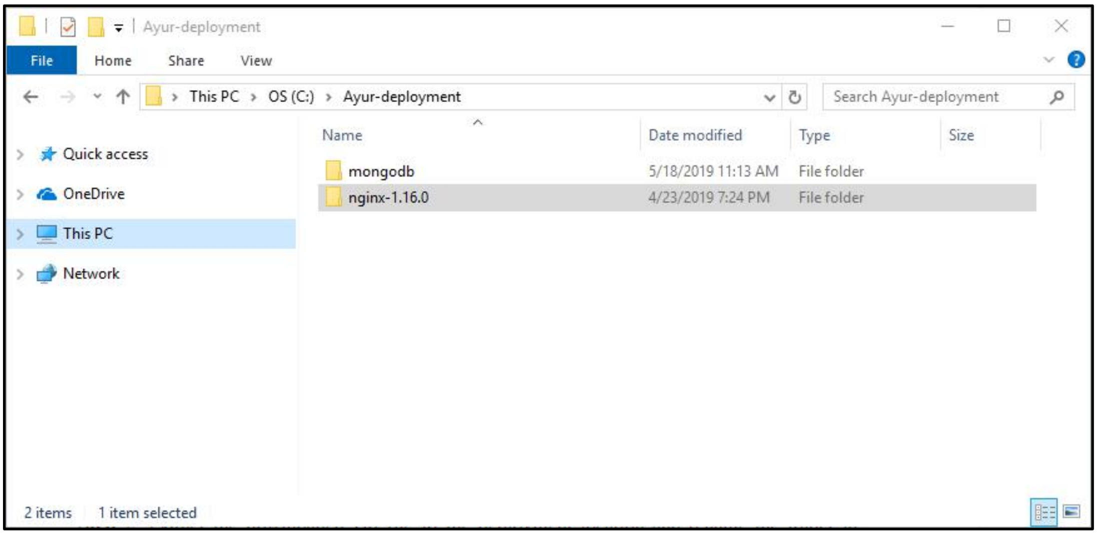
CHANGES	nginx-1.15.12 pgp	nginx/Windows-1.15.12 pgp

Stable version

CHANGES-1.16	nginx-1.16.0 pgp	nginx/Windows-1.16.0 pgp

Figure E.10 Download NGINX

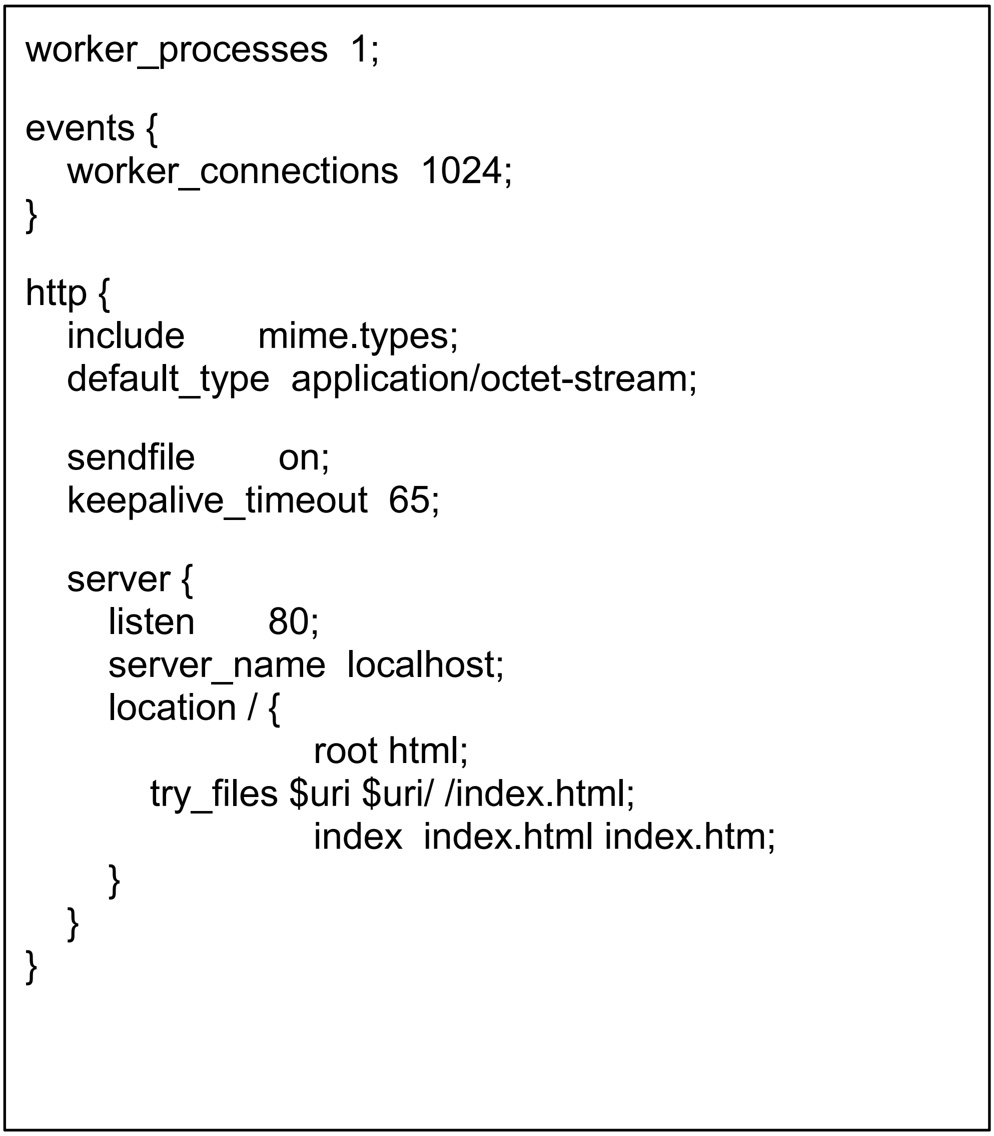
Step 2: extract the downloaded zip file to the deployment location and rename the folder in appropriate manner as shown in Figure E.11.



Name	Date modified	Type	Size
mongodb	5/18/2019 11:13 AM	File folder	
nginx-1.16.0	4/23/2019 7:24 PM	File folder	

Figure E.11 Extract NGNIX in Deployment Path

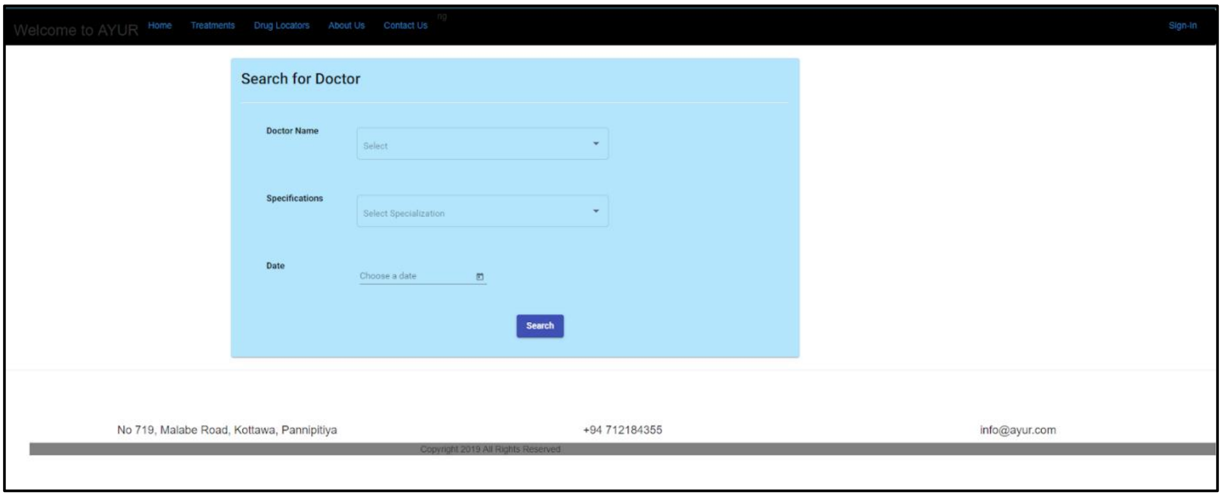
Step 3: change the C:\Ayur-deployment\nginx-1.16.0\conf\nginx.conf with the configuration given below (Figure E.12)



```
worker_processes  1;

events {
   worker_connections  1024;
}

http {
   include       mime.types;
   default_type  application/octet-stream;

   sendfile        on;
   keepalive_timeout  65;

   server {
      listen       80;
      server_name  localhost;
      location / {
                      root html;
         try_files $uri $uri/ /index.html;
                      index  index.html index.htm;
      }
   }
}
```

Figure E.12 NGINX Config

Step 4: copy the frontend files to C:\Ayur-deployment\nginx-1.16.0\html folder and start the web server by executing C:\Ayur-deployment\nginx-1.16.0\nginx.exe. The access the http://localhost with a web browser to verify that the frontend is running (Figure E.13).



Search for Doctor

Doctor Name: Select

Specifications: Select Specialization

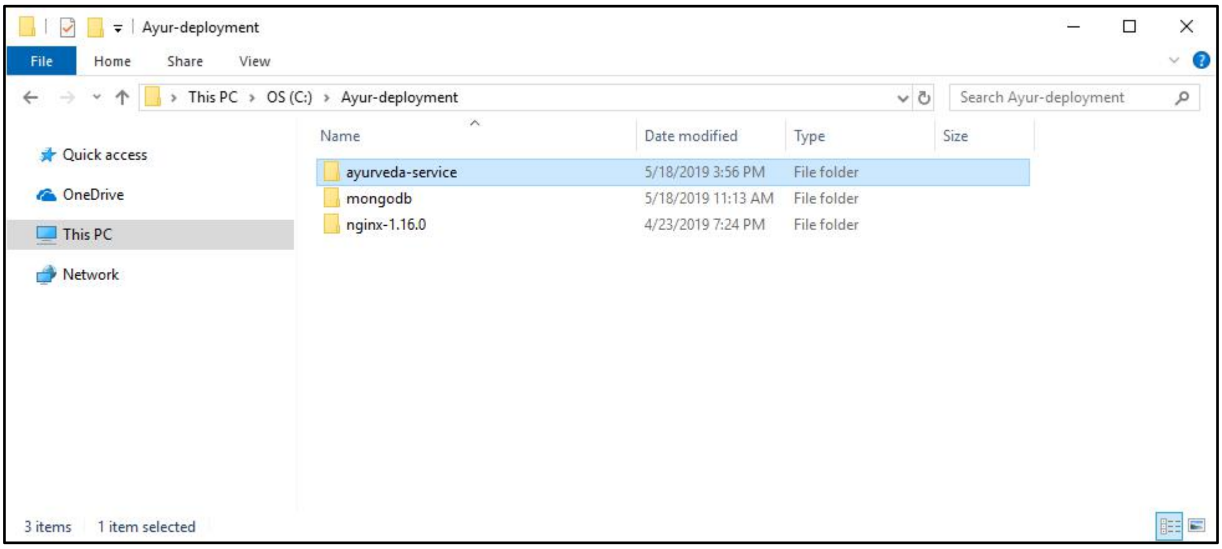
Date: Choose a date

No 719, Malabe Road, Kottawa, Pannipitiya
+94 712184355
info@ayur.com
Copyright 2019 All Rights Reserved

Figure E.13 Home Page

Configuring Deployment Environment: Deploy backend server

Step 1: Get the backend server binary zip file and extract it to the deployment location and rename the folder appropriately as shown in Figure E.14.

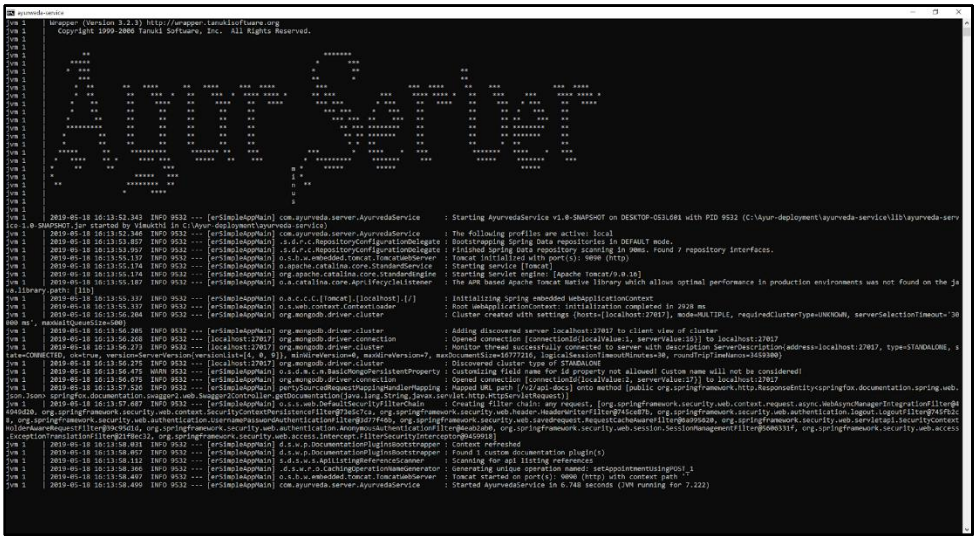


Name	Date modified	Type	Size
ayurveda-service	5/18/2019 3:56 PM	File folder	
mongodb	5/18/2019 11:13 AM	File folder	
nginx-1.16.0	4/23/2019 7:24 PM	File folder	

3 items 1 item selected

Figure E.14 Extract API Server

Step 2: Change directory to C:\Ayur-deployment\ayurveda-service\bin thru a terminal window and enter the command “ayur-service console” to start the service in console mode (Figure E.15)



```
jvm 1    | Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
jvm 1    |   Copyright 1999-2006 Tanuki Software, Inc.  All Rights Reserved.
jvm 1    | 2019-05-18 16:13:52.343  INFO 9532 --- [erSimpleAppMain] com.ayurveda.server.AyurvedaService      : Starting AyurvedaService v1.0-SNAPSHOT on DESKTOP-053L601 with PID 9532 (C:\Ayur-deployment\ayurveda-service\lib\ayurveda-serv
ice-1.0-SNAPSHOT.jar started by Vimukthi in C:\Ayur-deployment\ayurveda-service)
jvm 1    | 2019-05-18 16:13:52.346  INFO 9532 --- [erSimpleAppMain] com.ayurveda.server.AyurvedaService      : The following profiles are active: local
jvm 1    | 2019-05-18 16:13:53.857  INFO 9532 --- [erSimpleAppMain] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data repositories in DEFAULT mode.
jvm 1    | 2019-05-18 16:13:53.957  INFO 9532 --- [erSimpleAppMain] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 90ms. Found 7 repository interfaces.
jvm 1    | 2019-05-18 16:13:55.137  INFO 9532 --- [erSimpleAppMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 9090 (http)
jvm 1    | 2019-05-18 16:13:55.174  INFO 9532 --- [erSimpleAppMain] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
jvm 1    | 2019-05-18 16:13:55.174  INFO 9532 --- [erSimpleAppMain] org.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/9.0.16]
jvm 1    | 2019-05-18 16:13:55.187  INFO 9532 --- [erSimpleAppMain] o.a.catalina.core.AprLifecycleListener   : The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found on the ja
va.library.path: [lib]
jvm 1    | 2019-05-18 16:13:55.337  INFO 9532 --- [erSimpleAppMain] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
jvm 1    | 2019-05-18 16:13:55.337  INFO 9532 --- [erSimpleAppMain] o.s.web.context.ContextLoader            : Root WebApplicationContext: initialization completed in 2928 ms
jvm 1    | 2019-05-18 16:13:56.204  INFO 9532 --- [erSimpleAppMain] org.mongodb.driver.cluster               : Cluster created with settings {hosts=[localhost:27017], mode=MULTIPLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30
000 ms', maxWaitQueueSize=500}
jvm 1    | 2019-05-18 16:13:56.205  INFO 9532 --- [erSimpleAppMain] org.mongodb.driver.cluster               : Adding discovered server localhost:27017 to client view of cluster
jvm 1    | 2019-05-18 16:13:56.268  INFO 9532 --- [localhost:27017] org.mongodb.driver.connection            : Opened connection [connectionId{localValue:1, serverValue:16}] to localhost:27017
jvm 1    | 2019-05-18 16:13:56.273  INFO 9532 --- [localhost:27017] org.mongodb.driver.cluster               : Monitor thread successfully connected to server with description ServerDescription{address=localhost:27017, type=STANDALONE, s
tate=CONNECTED, ok=true, version=ServerVersion{versionList=[4, 0, 9]}, minWireVersion=0, maxWireVersion=7, maxDocumentSize=16777216, logicalSessionTimeoutMinutes=30, roundTripTimeNanos=3459300}
jvm 1    | 2019-05-18 16:13:56.275  INFO 9532 --- [localhost:27017] org.mongodb.driver.cluster               : Discovered cluster type of STANDALONE
jvm 1    | 2019-05-18 16:13:56.475  WARN 9532 --- [erSimpleAppMain] o.s.d.m.c.m.BasicMongoPersistentProperty : Customizing field name for id property not allowed! Custom name will not be considered!
jvm 1    | 2019-05-18 16:13:56.675  INFO 9532 --- [erSimpleAppMain] org.mongodb.driver.connection            : Opened connection [connectionId{localValue:2, serverValue:17}] to localhost:27017
jvm 1    | 2019-05-18 16:13:57.526  INFO 9532 --- [erSimpleAppMain] pertySourcedRequestMappingHandlerMapping : Mapped URL path [/v2/api-docs] onto method [public org.springframework.http.ResponseEntity<springfox.documentation.spring.web.
json.Json> springfox.documentation.swagger2.web.Swagger2Controller.getDocumentation(java.lang.String,javax.servlet.http.HttpServletRequest)]
jvm 1    | 2019-05-18 16:13:57.687  INFO 9532 --- [erSimpleAppMain] o.s.s.web.DefaultSecurityFilterChain     : Creating filter chain: any request, [org.springframework.security.web.context.request.async.WebAsyncManagerIntegrationFilter@4
4949d20, org.springframework.security.web.context.SecurityContextPersistenceFilter@73e5c7ca, org.springframework.security.web.header.HeaderWriterFilter@745ce87b, org.springframework.security.web.authentication.logout.LogoutFilter@745fb2c
8, org.springframework.security.web.authentication.UsernamePasswordAuthenticationFilter@3d77f46b, org.springframework.security.web.savedrequest.RequestCacheAwareFilter@6a995620, org.springframework.security.web.servletapi.SecurityContext
HolderAwareRequestFilter@39c95d1d, org.springframework.security.web.authentication.AnonymousAuthenticationFilter@4eab2ab0, org.springframework.security.web.session.SessionManagementFilter@5606331f, org.springframework.security.web.access
.ExceptionTranslationFilter@21f8ec32, org.springframework.security.web.access.intercept.FilterSecurityInterceptor@9459918]
jvm 1    | 2019-05-18 16:13:58.031  INFO 9532 --- [erSimpleAppMain] d.s.w.p.DocumentationPluginsBootstrapper : Context refreshed
jvm 1    | 2019-05-18 16:13:58.057  INFO 9532 --- [erSimpleAppMain] d.s.w.p.DocumentationPluginsBootstrapper : Found 1 custom documentation plugin(s)
jvm 1    | 2019-05-18 16:13:58.112  INFO 9532 --- [erSimpleAppMain] s.d.s.w.s.ApiListingReferenceScanner     : Scanning for api listing references
jvm 1    | 2019-05-18 16:13:58.366  INFO 9532 --- [erSimpleAppMain] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: setAppointmentUsingPOST_1
jvm 1    | 2019-05-18 16:13:58.497  INFO 9532 --- [erSimpleAppMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 9090 (http) with context path ''
jvm 1    | 2019-05-18 16:13:58.499  INFO 9532 --- [erSimpleAppMain] com.ayurveda.server.AyurvedaService      : Started AyurvedaService in 6.748 seconds (JVM running for 7.222)
```

Figure E.15 Start API Server

Configuring User Environment: Installing mobile app

Step 1: Get the android mobile application and copy that apk file to your mobile device and install the apk using file manager.

Appendix F – Code Snippets

Some important code snippets are as below;

Authentication controller

```
package com.ayurveda.server.controller;

import com.ayurveda.server.domain.UserAuthenticateData;
import com.ayurveda.server.domain.UserType;
import com.ayurveda.server.dto.Member;
import com.ayurveda.server.dto.User;
import com.ayurveda.server.repository.UserAuthenticationDataRepository;
import com.ayurveda.server.services.MemberService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.CrossOrigin;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import java.security.Principal;

@RestController
@RequestMapping(path = "/user")
@CrossOrigin("*")
public class UserController {

  private UserAuthenticationDataRepository authenticationDataRepository;
  private MemberService memberService;

  @Autowired
  public UserController(UserAuthenticationDataRepository authenticationDataRepository, MemberService memberService) {
    this.authenticationDataRepository = authenticationDataRepository;
    this.memberService = memberService;
  }


  @GetMapping(path = "/validateLogin")
  public User validateLogin(Principal principal) {
    UserAuthenticateData authUser =
authenticationDataRepository.findUserAuthenticateDataByUserNameIs(principal.getName());
    User user = new User();
    user.setUserName(authUser.getUserName());
    user.setStatus(String.valueOf(authUser.getAccountStatus()));
    user.setUserType(authUser.getUserType());
    user.setAdminUser(authUser.getUserRoleList().stream().anyMatch(userRoleType -> userRoleType== UserType.ADMIN));
    return user;
  }
  @GetMapping(path = "/get")
  public Member getCurrentUser(Principal principal) {
    UserAuthenticateData authUser =
authenticationDataRepository.findUserAuthenticateDataByUserNameIs(principal.getName());
    return memberService.retrieveMemberData(authUser);
  }


}
```

Messaging Service

```
package com.ayurveda.server.services;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.mail.javamail.*
import org.springframework.stereotype.Service;
import javax.annotation.PostConstruct;
import javax.mail.*;
import java.io.*;
import java.net.URL;
import java.util.Properties;

@Service
public class MessagingService {
    @Value("${ayur.email.username}")
    private String username;
    @Value("${ayur.email.password}")
    private String password;
    @Value("${ayur.email.mail-host}")
    private String host;
    private JavaMailSenderImpl sender;
    @PostConstruct
    public void init() {
        this.sender = new JavaMailSenderImpl();
        this.sender.setHost(this.host);
        this.sender.setPort(587);
        this.sender.setUsername(this.username);
        this.sender.setPassword(this.password);
        this.sender.setProtocol("smtp");
        Properties javaMailProperties = new Properties();
        javaMailProperties.put("mail.smtp.starttls.enable", true);
        javaMailProperties.put("mail.smtp.auth", true);

        this.sender.setJavaMailProperties(javaMailProperties);
    }
    public boolean sendSms(String message, String receiverNo) {
        try {
            URL textit = new URL("http://textit.biz/sendmsg/index.php?id=94778132872&pw=3925&to=" + receiverNo + "&text=[Ayur]" + message);
            BufferedReader in = new BufferedReader(new InputStreamReader(textit.openStream()));
            String inputLine;
            while ((inputLine = in.readLine()) != null)
                System.out.println(inputLine);
            in.close();
            return true;
        } catch (IOException e) {
            e.printStackTrace();
        }
        return false;
    }
    public boolean sendEmail(String messageBody, String receiver) {
        try {
            MimeMessage message = sender.createMimeMessage();
            MimeMessageHelper helper = new MimeMessageHelper(message, true);
            helper.setTo(receiver);
            helper.setText(messageBody, true);
            sender.send(message);
            return true;
        } catch (MessagingException e) {
            e.printStackTrace();
        }
        return false;
    }
}
```

Patient Repository

```
package com.ayurveda.server.repository;

import com.ayurveda.server.domain.Patient;
import org.springframework.data.mongodb.repository.MongoRepository;
import org.springframework.stereotype.Repository;

import java.util.List;

@Repository
public interface PatientRepository extends MongoRepository<Patient, String> {
    @Override
    List<Patient> findAll();

    @Override
    <S extends Patient> S insert(S entity);

    Patient findPatientByPatientNicIs(String patientNic);

    void deletePatientByPatientNicIs(String doctorNic);

    List<Patient> findPatientByFirstNamesIsOrLastNameIs(String firstNames, String lastName);

    Patient findPatientByAuthenticateData_Id(String authenticateData_id);
}
```

Patient Domain Class

```
package com.ayurveda.server.domain;
import com.mongodb.lang.*Nullable;
import lombok.*;
import org.hibernate.validator.constraints.Length;
import org.springframework.data.annotation.Id;
import org.springframework.data.mongodb.core.index.Indexed;
import org.springframework.data.mongodb.core.mapping.*;
import java.time.LocalDate;
import java.util.Date;

@Document(collection = "patient")
@Getter
@Setter
@ToString
@Builder
@NoArgsConstructor
@AllArgsConstructor
public class Patient {

    @Id
    private String patientId;
    @Indexed(unique = true)
    @NonNull
    @Field(value = "patient_nic")
    private String patientNic;
    @Field(value = "first_names")
    private String firstNames;
    @Indexed
    @Field(value = "last_name")
    private String lastName;
    private String title;
    @Indexed
    @Field(value = "date_0f_birth")
    private LocalDate dob;
    @Field(value = "address")
    private String address;
    private String country;
    @Indexed
    @Nullable
    private String email;
    @NonNull
    private Date joinedDate;
    @NonNull
    @Length(max = 10, min = 10)
    @Field(value = "primary_contact")
    private String primaryContactNumber;
    @NonNull
    @Field(value = "emergency_contact_name")
    private String emergencyContactPersonName;
    @NonNull
    @Length(min = 10, max = 10)
    @Field(value = "emergency_contact_number")
    private String emergencyContactPersonNumber;
    @NonNull
    @Field(value = "user_authenticate_data")
    @DBRef(lazy = true)
    private UserAuthenticateData authenticateData;

    private String notificationMethod;
}
```