Department of Computer Science and Engineering

Siddaganga Institute of Technology

Tumakuru - 572103

CERTIFICATE

Certified that the Project Report entitled **"Smart Health Monitoring System (a web application)"** is a bonafide work carried out by **Likhith M (1SI13CS141)** in the partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Computer Science and Engineering , Visvesvaraya Technological University, Belagavi during the year 2016-17. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

....	....
Guide	**Group Convener**
Prof. M. Raghavendra	**Dr. Y S Nijagunarya**
Asst. Professor	**Professor**
Dept of CSE, SIT	**Dept of CSE, SIT**
....	....
Dr. N R Sunitha	**Dr. Shivakumaraiah**
Professor and Head	**Principal**
Dept of CSE, SIT	**SIT, Tumakuru**

Name of the Examiners Signature with Date

1. Prof.

2. Prof.

Acknowledgements

With the successful completion of our project, it becomes part of our duty to express our gratitude to all those who made this work possible.

We are thankful to God for having given us an opportunity to associate ourselves with an esteemed personality like His Holiness **Dr. Shree Shree Shivakumara Swamijigalu**, who has been a great source of inspiration and guidance to all of us.

We express our immense gratitude to our respected Director, **Dr. M N Channabasappa** for his constant support in fulfilling of endeavours.

We would also like to thank our principal, **Dr. Shivakumariah** for providing us with various facilities for carrying out this project.

We are thankful to our HOD **Dr. N R Sunitha** for giving us all the freedom, inspiration and encouragement to carry out this project successfully.

We would like to express our heartiest gratitude to our group convener **Dr. Y S Nijagunarya** and guide Asst. Prof **M. Raghavendra** for his constructive, skilled guidance and encouragement at every step which helped us to complete our work successfully. We are also thankful to the panel for their invaluable suggestions.

We would also like to express our gratitude to our proctor Asst. Prof **Chandraprabha K S** and other faculty members.

Last but not the least, we would like to thank our parents for the help that they have provided. We thank our friends, who has always been there to help us. Our sincere thanks to all those who helped us directly or indirectly in the successful completion of the project.

Project Associate
Likhith M

Abstract

Healthcare is a sector in which error-free record keeping and communication are acute and is one of the area in which the use of computing and networking technology straggles behind other fields or sectors. Healthcare professionals(like Physicians), patients and even the Pharmacists are usually discomfited with computers, and feel that computers are not halfway to their healthcare purpose, even though they agree that acute record keeping and conveyance are consequential to good healthcare. In the present healthcare, data is transmitted from one healthcare professional to another via paper notes or personal communication. For example, in India, electronic communication between pharmacists and is not typically employed but, rather, the physician writes a prescription on paper and gives it to the patient. The patient will carry the prescription to the pharmacy, waits in queue to give it to a pharmacist, and waits for the pharmacist to fill the prescription. To improve this process, the prescriptions could be communicated electronically from the physician to the pharmacist, and the human computer interfaces for the physicians, nurses, pharmacists and other healthcare professionals could be voice enabled.

This project describes a distributed smart healthcare system that uses the Service-Oriented Architecture as a means of designing, implementing, and maintaining healthcare services. The login credentials of each user will be stored and maintained in the database. Thus,a particular user through his Id and password can login to the website and view or update the information available in his blog.

The main objective is to provide a user friendly platform and promisng the people to maintain a better health. Effective and timely communication between patients, physicians, nurses, pharmacists, and other healthcare professionals is essential to good healthcare. Current communication mechanisms, based largely on paper records and prescriptions, are old-fashioned, inefficient, and unreliable. When multiple healthcare professionals and facilities are involved in providing healthcare for a patient, the healthcare services provided aren't often coordinated. Typically, a physician writes a prescription on paper and gives it to the patient. The patient carries the prescription to the pharmacy, waits in line to hand the prescription to the pharmacist, and waits for the pharmacist to fill the prescription. The pharmacist might be unable to read the physician's handwriting, the patient could modify or forge the prescription, or the physician might be unaware of medications prescribed by other physicians. These and other problems indicate the need to improve the quality of healthcare.

A distributed electronic healthcare system based on the service-oriented architecture (SOA) can address some of these issues and problems. Multimedia input and output—with text, images make the system less computer- like and more appealing and friendly to the users who aren't computer-oriented.

Smart health monitoring system uses Service oriented architecture[3] to enforce basic software architecture principles and provide interoperability between different computing platforms and applications that communicate with each other. Although our distributed healthcare system provides user-friendly interfaces for busy healthcare professionals and patients, security and privacy are particularly important in this area, so we designed the prototype with security and privacy in mind. The system authenticates users(can be patient, physician or pharmacist or admin) and logs session information and attaches resources to the resource creator, so that only privileged users can view or modify the data.

Along with all these features,there is also a supporting website where you can give feedback regarding the actual website or any related queries in the website. This will help us to maintain a friendly relation with our users.

Contents

List of Figures

Chapter 1

Introduction

Preamble

This section clearly explicates the overview of the Project. It describes the basic motivation behind the project Smart Health Monitoring System. Emphasizes on the relavance of the project to the society and the industry is of essential importance. A detailed discussion of why this project was chosen is presented. Also,the previous works and related works along with their limitations are presented. Followed by this,the problem statement and main objevtives of the projects are discussed. At the end, it gives a picture of the organization of the report.

1.1 Background Study

1.1.1 Motivation

In existing Healthcare systems has focused on record keeping and databases. It has also focused on access and security for recording and communicating healthcare information. Human errors are more by exploiting electronic communication and record keeping. There are as many as 7,000 deaths from illegible handwriting of doctors every year in the country[1]. This is enough to indicate that there is an immediate need to reduce the errors in healthcare.

This system aims to reduce human errors by exploiting electronic communication and record keeping, and by providing user-friendly input and output capabilities. Modern technology is use to expose the functionality of our e-healthcare system as Web Services based on the Service Oriented Architecture, so that both humans and applications can use the services provided. It provides services that involve patients, physicians, nurses and pharmacists as well as medical monitoring devices, whereas their framework focuses specifically on the use of medical monitoring devices.

1.1.2 Relavance of the Project to the society and the industry

Healthcare is a sector in which error-free record keeping and communication are acute and is one of the area in which the use of computing and networking technology straggles behind other fields or sectors. Healthcare professionals(like Physicians),patients and even the Pharmacists are usually discomfited with computers, and feel that computers are not halfway to their healthcare purpose, even though they agree that acute record keeping and conveyance are consequential to good healthcare. In the present healthcare, data is transmitted from one healthcare professional to another via paper notes or personal communication. For example, in India, electronic communication between pharmacists and is not typically employed but, rather, the physician writes a prescription on paper and gives it to the patient. The patient will carry the prescription to the pharmacy, waits in queue to give it to a pharmacist, and waits for the pharmacist to fill the prescription. To improve this process, the prescriptions could be communicated electronically from the physician to the pharmacist, and the human computer interfaces for the physicians, nurses, pharmacists and other healthcare professionals could be voice enabled.

Typically, a physician writes a prescription on paper and gives it to the patient. The patient carries the prescription to the pharmacy, waits in line to hand the prescription to the pharmacist, and waits for the pharmacist to fill the prescription. The pharmacist might be unable to read the physician's handwriting; the patient could modify or forge the prescription; or the physician might be nescient of medications prescribed by other physicians. These and other problems indicate the need to improve the quality of healthcare.

The smart health monitoring system uses service oriented architecture to enforce basic software architecture principles and provide interoperability between different computing platforms and applications that communicate with each other. Although our distributed healthcare system provides user-friendly interfaces for busy healthcare professionals and patients, security and privacy are particularly important in this area, so we designed the prototype with security and privacy in mind. The system authenticates users(can be patient, physician or pharmacist or admin) and logs session information and attaches resources to the resource creator, so that only privileged users can view or modify the data.

The patients can also make use of the system and obtain the predictions based on the various data that is already present. The patient can know what disease he has by just giving the symptoms. Thus, even if the doctor is not available the patient can take some pre-cuationary measures. Also, this helps the doctor to use their precious time on more important things. Even the Pharmacists have an advantage since they will updated electronically regarding the medicines and their updated versions.So overall it is an unique and benificial platform to the health care.

1.2 Related Works

Title of the work:E-healthcare: an analysis of key themes in research.

Authors: Avinandan Mukherjee, John McGinnis.

Publication details:International Journal of Pharmaceutical and Healthcare Marketing 2007.

Description:Healthcare is among the fastest-growing sectors in both developed and emerging economies. E-healthcare is contributing to the explosive growth within this industry by utilizing the internet and all its capabilities to support its stakeholders with information searches and communication processes. The purpose of this paper is to present the key themes. Design/methodology/approach – A review of the literature in the marketing and management of e-healthcare was conducted to determine the major themes pertinent to e-healthcare research as well as the commonalities and differences within these themes. Findings – Based on the literature review, the five major themes of e-healthcare research identified are: cost savings; virtual networking; electronic medical records; source credibility and privacy concerns; and physician-patient relationships. Originality/value – Based on these major themes, managerial implications for e-healthcare are formulated. Suggestions are offered to facilitate healthcare service organizations attempts to further implement and properly utilize e-healthcare in their facilities. These propositions will also help these stakeholders develop and streamline their e-healthcare processes already in use. E-healthcare systems enable firms to improve efficiency, to reduce costs, and to facilitate the coordination of care across multiple facilities.

Title of the work: Role Based Access Control Models for E-Healthcare Systems.
Authors: Jarrod Williams .
Publication details: Florida A and M University - Department of Computer and Information Sciences.
Description:E-healthcare is the use of web-based systems to share and deliver information across the internet.With this ability, privacy and security must be maintained according to the Health Insurance Portability and Accountability Act(HIPAA) standards[1].
The reasonable approach to developing a system that can meet these requirements is a system that utilizes role-based models. Role-based access control(RBAC)is important because personnel could change but the position and access to the safe information keeps stable. With a role-based model it becomes easier to maintain access control, assign privileges, and personnel to the appropriate role.

Implementation is based off the security policy, which is a critical component of any system because it defines which roles or people have access to what information. An extensible markup language (XML) is used to enforce this policy because it is a web-based technology that is good for data transportation and security. Within this research project, we are able to give an overview for the state-of-art of secure e-healthcare system, and better understand a way of implementing a secure e-healthcare system that meets HIPAA standards and can share information to patients and healthcare facilities via the web service.

Title of the work: A Distributed System Based on the Service Oriented Architecture.
Authors: Kart, F. Gengxin Miao Moser, L.E. Melliar-Smith.
Publication Details:Dept. of Electr. and Comput. Eng., Univ. of California, Santa Barbara, CA;
Description: Large-scale distributed systems, such as e-healthcare systems, are difficult to develop due to their complex and decentralized nature. The service oriented architecture facilitates the development of such systems by supporting modular design, application integration and interoperation, and software reuse. With open standards, such as XML, SOAP, WSDL and UDDI, the service oriented architecture supports interoperability between services operating on different platforms and between applications implemented in different programming languages. In this paper we describe a distributed e-healthcare system that uses the service oriented architecture as a basis for designing, implementing, deploying, invoking and managing healthcare services. The e-healthcare system that we have developed provides support for

physicians, nurses, pharmacists and other healthcare professionals, as well as for patients and medical devices used to monitor patients. Multi-media input and output, with text, images and speech, make the system more user friendly than existing e-healthcare systems.

1.3 Project Problem Statement and Objectives

Healthcare is a field in which accurate record keeping and communication are critical. In current healthcare, information is conveyed from one healthcare professional to another through paper notes or personal communication. The pharmacist might be unable to read the physician's handwriting. The patient could modify or forge the prescription .The physician might be unaware of medications prescribed by other physicians. To improve this process, the prescriptions could be communicated electronically.

The main objectives of this project are :-

- To provide effective and timely communication between patients, physicians, pharmacists etc.
- Reducing human errors by providing user-friendly input and output capabilities.
- Promising individuals to maintain a better health.

1.4 Organization of the Report

This Chapter gives a brief introduction to the project. It also discusses the motivation behind the decision of why is this topic chosen for implementation as the final year major project, Related works studied that supports the idea and the objectives of the project.

The Chapter 2 describes the design part of the project. It gives a brief description of what software development methodology is used, describes the main architecture of the project, discusses each module and also the functional requirements of project.

The Chapter 3 emphasizes thoroughly on design part of the project. The user interface of the applications, UML diagrams along with the screen shots of the application are presented in this chapter.

The Chapter 4 gives detailed description on the implementation part. What are the tools and technologies used and why, how the integration of all modules takes place and their work flows are discussed here. This chapter also covers the nonfunctional requirements of the project.

The Chapter 5 focuses on the testing details. The work flow of different tests are briefed here. Each test case is tested and the results of the same are described in this chapter.

The Chapter 6 concludes and proposes future improvements to the project. The detailed discussion of each chapter can be seen in the coming chapters.

Chapter 2

High-level Design

This section describes the design part of the project. It highlights on the Software development methodology used in the development of the project. The overall architecture of the project is discussed in this section.Along with the system architecture various levels of data flow diagrams,module diagrams are being discussed in this section. This forms the high level design of the project,Smart Health Monitoring System.

2.1 Software development methodology

Iterative and Incremental Development

Iterative and incremental software development is a method of software development that is modeled around a gradual increase in feature additions and a cyclical release and upgrade pattern. Iterative and incremental software development begins with planning and continues through iterative development cycles involving continuous user feedback and the incremental addition of features concluding with the deployment of completed software at the end of each cycle. The procedure itself consists of the initialization step, the iteration step, and the Project Control List. The initialization step creates a base version of the system. The goal for this initial implementation is to create a product to which the user can react. It should offer a sampling of the key aspects of the problem and provide a solution that is simple enough to understand and implement easily. To guide the iteration process, a project control list is created that contains a record of all tasks that need to be performed. It includes such items as new features to be implemented and areas of redesign of the existing solution. The control list is constantly being revised as a result of the analysis phase. Iterative and incremental development can be grouped into the following phases:

1. First, it deals with the scope of the project, requirements and risks at higher levels.
2. Delivers working architecture that moderates risks identified in the inception phase and satisfies nonfunctional requirements.
3. Fills in architecture components incrementally with production-ready code, which is produced through the analysis, implementation, design and testing of functional requirements.
4. Delivers the system to the production operating environment.

The iterative and incremental development model leads to software reuse, and reusability provides software engineers with a number of measurable benefits. Iterative and incremental development used to develop Smart Healthcare Monitoring System as follows:

1. First, we go through several iterations of analysis and design, until we are pretty sure these phases are mostly complete. We can then select the content and schedule of the increments to be developed.Now let's say we have identified three modules Clinic, Pharmacy and Physyian respectively and their respective functionalities are identified which will be designed one after the other in an incremental way. The overall arhitecture is as shown in Figure 2.1.
2. For increment A, we iterate through all required phases, until the software can be released. We expect to rework the analysis and design very little. That is, a small amount of improvement or addition can be seen in each module. i.e. At this stage, we can observe samll improvements that will be reflected when the patient, Physician and Pharmacist logins successfully as a user. Its working was checeked in the browser after first cycle.
3. For increment B, we iterate through all required phases, until the software can be released. Here we can expect almost no analysis or design work. i.e. Here we just concentrate on adding new functinalities to the previous prototype which is available after increment A. No analysis and design of any module occurs till the end of this increment cycle.
4. For increment C, we will iterate through all required phases, until the software can be released.This will be the final increment where the all the functionalities or requirements listed will be added and any issues in the previous prototypes are considered and relased as a working software.

This project can can picture this process as four applications of the above process. The amount of analysis and (architectural) design decreases in each step. The process delivers functionality after the second, third and fourth steps.

2.2 Architecture

In this project we describe a distributed smart healthcare system that uses the Service- Oriented Architecture as a means of designing, implementing, and maintaining healthcare services. The login credentials of each user will be stored and maintained in the database.The complete architecture of the system is as shown in figure 2.1. A particular user through his Id and password can login to the website and view or update the information available in his user space. A distributed electronic healthcare system[6] based on the service-oriented architecture (SOA) can address some of these issues and problems. Multimedia input and output—with text, images, and speech—make the system less computer- like and more appealing and friendly to the users who aren't computer-oriented.

This Service oriented architecture will enforce basic software architecture principles and provide interoperability between different computing platforms and applications that communicate with each other. Although our distributed healthcare system provides user-friendly interfaces for busy healthcare professionals and patients, security and privacy are particularly important in this area, so we designed the prototype with security and privacy in mind. The system authenticates users(can be patient, physician or pharmacist or admin) and logs session information and attaches resources to the resource creator, so that only privileged users can view or modify the data.It is clear from the architecture that the process is a cycle where the data will flow in a cyclic manner from each user. We can see two data storage devices or databases in the figure which helps in data exchange between patient-physician and pharmacist-patient.

The patient will fill all his personal details and store it in the database.The patient will submit the details to physician along with the appointment date. The patient can also review the medical advice and the information related to medicines from physician and pharmasict web servers. The Physician will view the records of a patient between a particualar duration and fills in the prescription and will be sumitted to both pharmacist and the patient. The physician can also fill details related to particular disease which will be availble in the patient web server. Finally the pharmacist will view the list of patients,their Physicians and the corresponding medicines and their dosages from his user interface.Based on this he will issue the medicines to the respective patients.

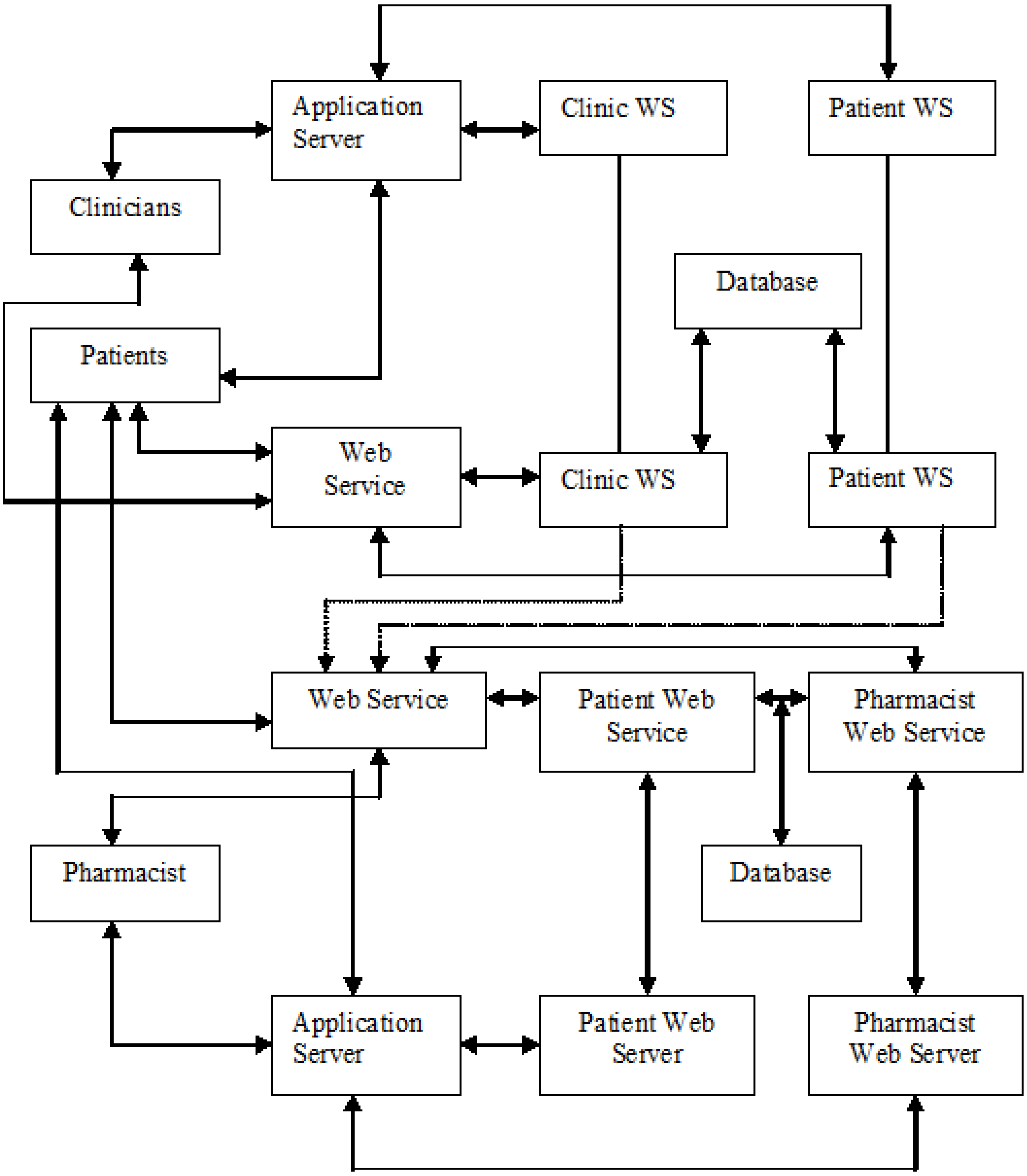


Figure 2.1: Complete Architecture of the System

2.3 Modules Description

2.3.1 Clinic Module

The Clinic module exposes two interfaces, a Web Server and a Web Service, for the clinic staff, the patients and the medical monitoring devices. The Web Server interface is intended for users who prefer to use a Web browser to access the healthcare services. Humans or devices to communicate with the e-healthcare system can use the Web Service interface. The Web Server uses the Web Services to access the data. The Clinic module provides support for routine activities of the physician. It maintains information, such as the physician's appointments for a specific day/week, the patients that s/he has examined, notes related to the patients, etc. The Clinic module sends prescriptions from the physician to the desired pharmacies using the web service provided. The block diagram is as shown in figure 2.2.

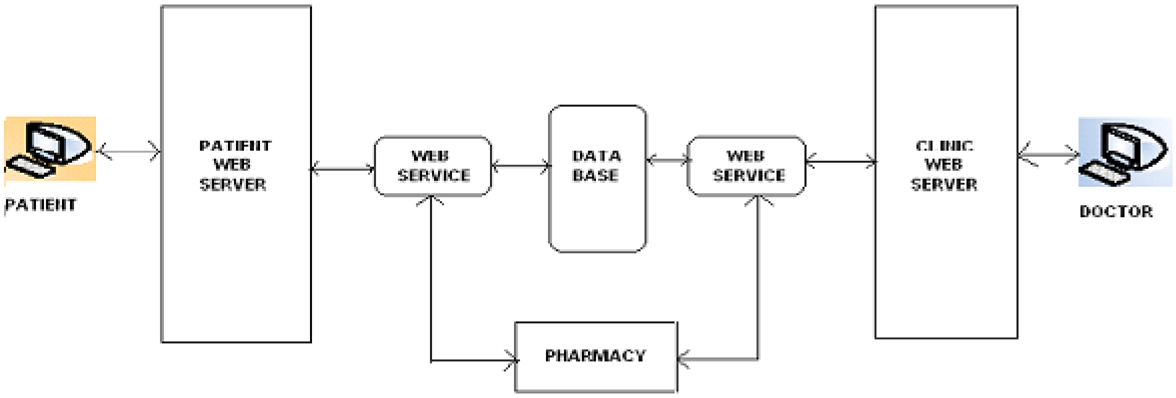


Figure 2.2: Clinic Module

2.3.2 Pharmacy Module

The Pharmacy module exposes Web Server and Web Service interfaces. The Web Server interface allows the users to access the e-healthcare system at the pharmacy using a browser. The Web Service interface provides access for applications deployed at the pharmacy and can also be used by humans and devices. The Pharmacy module provides services to the pharmacist, patients and devices used at the pharmacy. The Pharmacy module keeps a record of the patient's prescriptions for the pharmacist's and the patient's reference. When the physician submits a new prescription to the pharmacy, the Clinic module at the physician's office communicates with the Pharmacy module at the pharmacy. The pharmacist can view the outstanding prescriptions for the patients, as they are received from the physicians. The Pharmacy module updates the status of the prescriptions as the pharmacist fills them. The patient can determine, via the Web Server or Web Service, whether a prescription has been filled and is ready for pick up or delivery. The block diagram is as shown in figure 2.3.

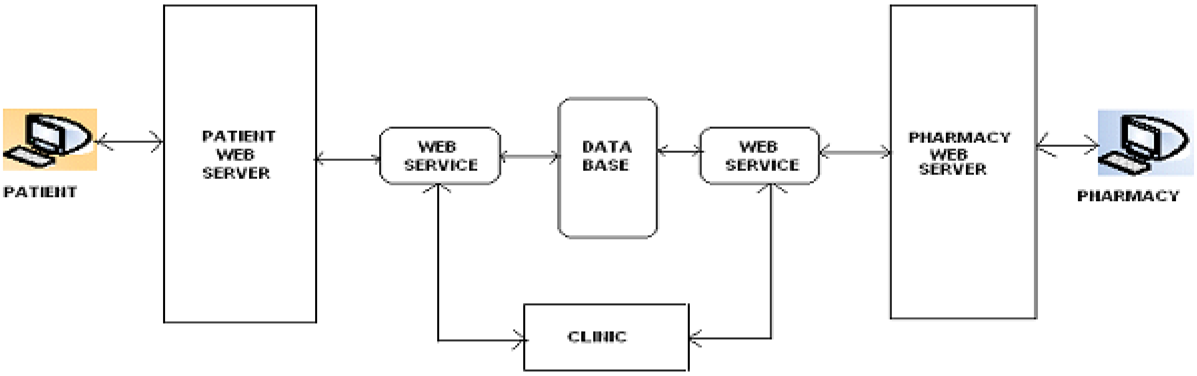


Figure 2.3: Pharmacy Module

2.4 Use Cases

2.4.1 Register as a new user

Name of the module: Registration module.

Parameters: Name of the user, password, email id, date of birth, phone number etc.

Purpose: This is a module common for patient,physician/doctor and pharmacist. In order to access the other functionalities of the website, every user must fill all the details and submit, which will be saved in the database.

2.4.2 Get Appointment

Name of the module: Physician Appointment.

Parameters: Physician Id, Appointment date.

Purpose: This is used to book an appointment by the patient with a particular physician. Here, the patient will enter the physician Id and the appointment date. on submission, these details will now be available to the physicain.

2.4.3 Get patient details

Name of the module: Patient details.

Parameters: Patient Id.

Purpose: This is used to get the details of the patient by the physician. The physician can view the details of the patient who has booked an appointment by giving the patient Id as input and he can know the basic information of that particular patient.

2.4.4 Submit diagnosis details

Name of the module: Diagnosis Details

Parameters: Name, sex, date of birth, symptoms, previous diagnosis.

Purpose: This is used to save the diagnosis details of a patient by the physician. After submitting the details, it updates the information of a particular patient and it will be saved in doctors database which he may use for the diagnosis of that patient in the future.

2.4.5 View Pharmacy details

Name of the module: Pharmacy details.

Parameters: Start date, End date .

Purpose: This is used to view pharmacists day/weekly/all list. He will either chose today's date or he will input the start and end date. On submission, it will show the details of patient, his physician, the medicine to be given and its dosage.

2.5 Data Flow Diagrams

2.5.1 Level-0 Data Flow diagram

Here we can see three entities patient, clinic and pharmacy respectively. First, the data flows from the patient module where he will submit all the details to the Physician(clinic module).Now, based on the diagnosis result the physician will update the prescriptions of that particular patient to the pharmacist.Corresponding medicines will be given to the patient by the pharmacist.This process is represented as a cycle in the figure 2.4.

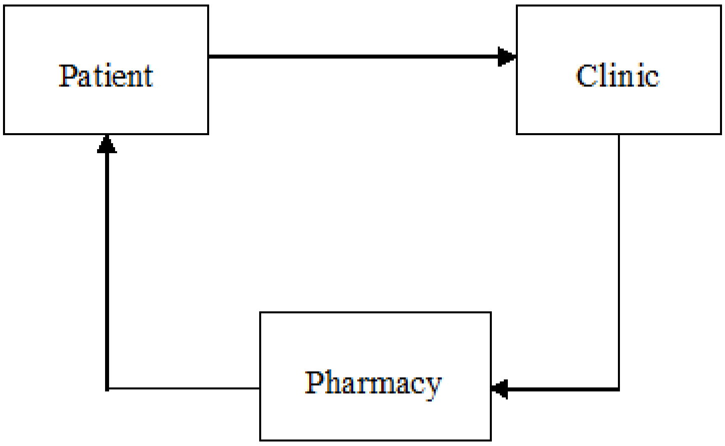


Figure 2.4: Data Flow Diagram Level-0

2.5.2 Level-1 Data Flow diagram

This part will highlight the interaction of patient and Physician. The level-1 data flow diagram is as shown in figure 2.5. Here on the patient side,he/she will submit the details like patient Id, symptoms, physician Id and any previous medicine if he/she is taking. After submitting this the patient can get an appointment by specifying the Physician Id and the date. All these details will be stored in the central server. The Physician on the other hand will access the stored data and fills the prescriptions to the particular patient. Along with this,the medical prescription details will be submitted to the Pharmacist as well. The physician can also provide the details of few common diseases to the patient prior to the diagnosis.

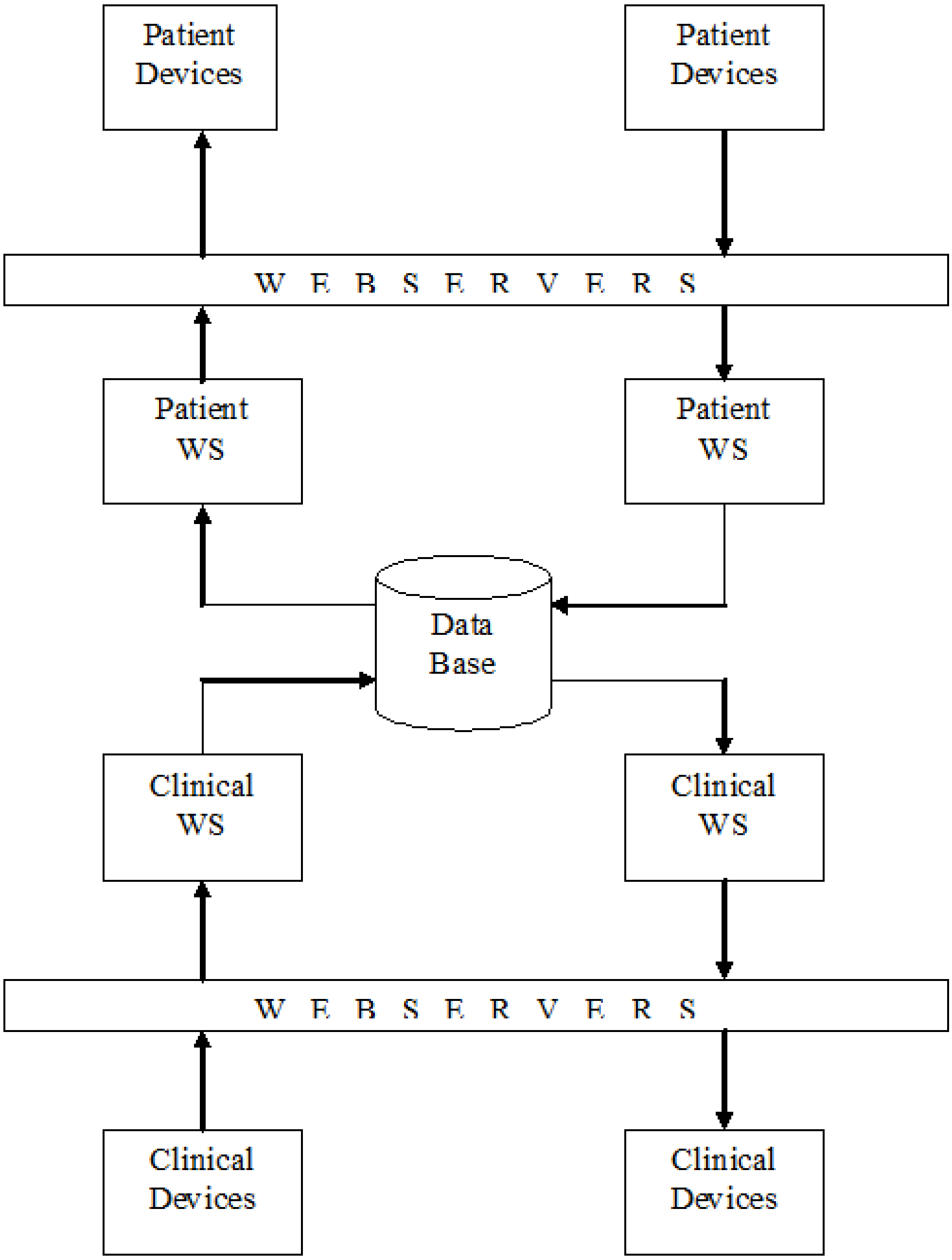


Figure 2.5: Data Flow Diagram Level-1

2.5.3 Level-2 Data Flow diagram

The figure 2.6 shows the level-2 Data flow diagram. It emphasizes the pharmacy and the patient modules. The patient will login to his/her user space and will interact with the database through our website.In the database the patient can find the available medicines and any updates in the medicines. All these will be performed through patient web server. The Pharmacist on the other hand based on the physicians prescriptions will store the medicine details and update them through his user space.He/she can view the list of all records based on monthly/weekly/ daily basis. It contains the patient Id,name,physician name,medicine and its dosage respectively.The Database is common for both the users and any changes in the database will be reflected in both the web servers.Thus,any number of users can use the website at any time.

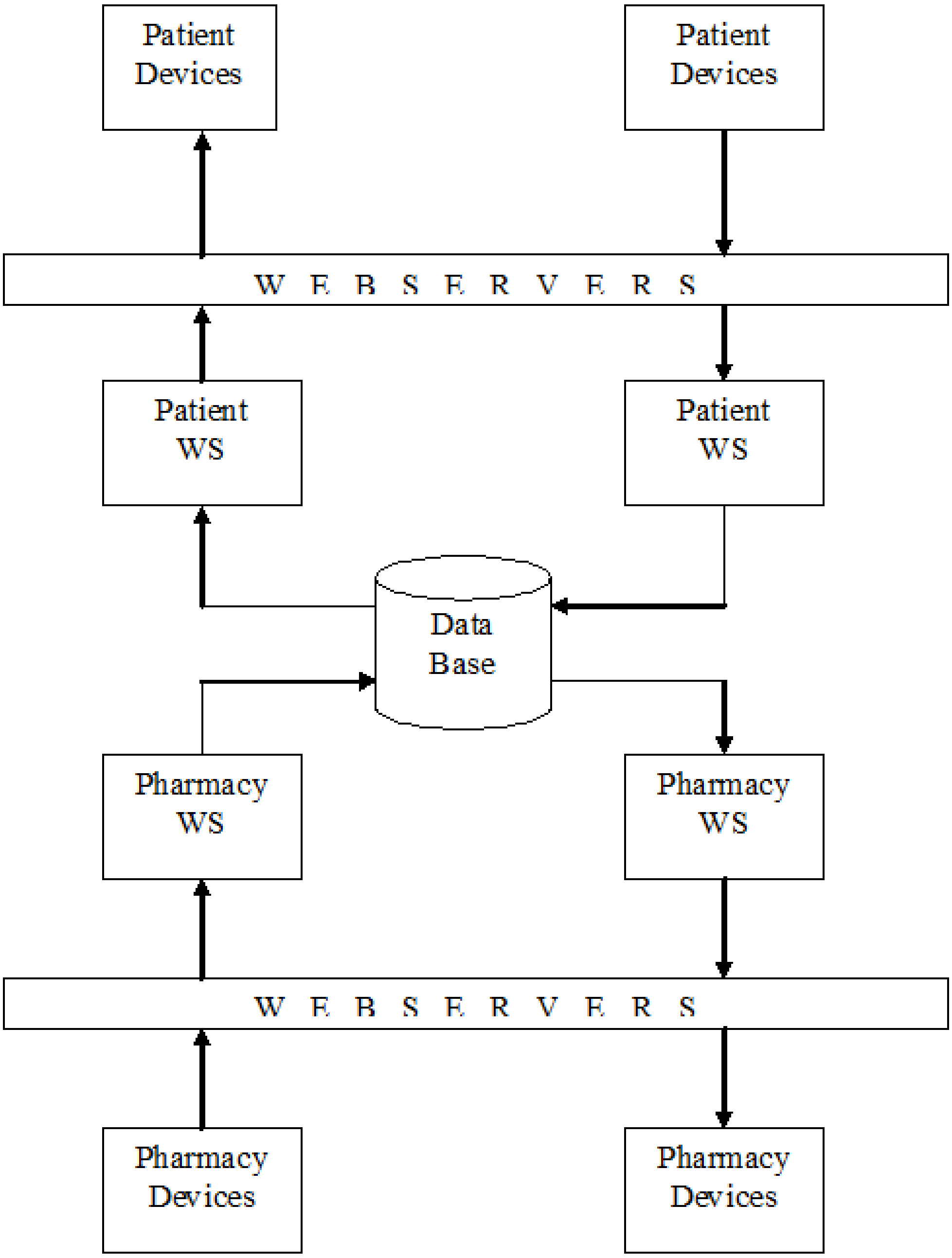


Figure 2.6: Data Flow Diagram Level-2

Chapter 3

Detailed Design

Preamble

This chapter describes the detailed design part of the project. It highights on the interface that how software(website here) interacts with users and with other softwares. Along with this a detailed description on data structures and algorithms used, UML diagrams and database used are presented in this section.

3.1 Interface design

The interface design is that part of a system with which a user interacts in order to undertake tasks and reach the goals. It is concerned with describing user behavior and defining how the system will accommodate and react to that behavior. Interface design is the design of user interfaces for machines and software with the focus on expanding the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals. Good user interface design facilitates finishing the task at hand without drawing unnecessary attention influencing how the user performs certain interactions and improving the visual appeal of the design which may enhance or detract from the ability of users to use the functions of the interface in a particular device.

Interface design activities

We have a number of activities performed for designing user interface. The process of GUI design and implementation is alike SDLC. Any model can be used for GUI implementation among Waterfall, Iterative or Spiral Model.The Interface design cycle is as shown in figure. A model used for GUI design and development should fulfill these GUI specific steps:-

1. GUI Requirement Gathering - Designers may like to have list of all functional and non-functional requirements of GUI.This can be taken from user and their software solution which is already existing.

2. User Analysis - The designer studies who is going to use the software GUI. The target audience matters as the design details change according to the knowledge and competency level of the user. If user is technical savvy, advanced and complex GUI can be incorporated. For a novice user, more information is included on how-to of software.

3. Task Analysis - Designers have to analyze what task is to be done by the software solution. Here in GUI, it does not matter how it will be done. Tasks can be represented in hierarchical manner taking one major task and dividing it further into smaller sub-tasks. Tasks provide goals for GUI presentation. Flow of information among sub-tasks determines the flow of GUI contents present in the software.

4. GUI Design and implementation - Designers after having information about requirements, tasks and user environment, design the GUI and implements into code and embed the GUI with working or dummy software in the background. It is then self-tested by the developers.

5. Testing - GUI testing can be done in various ways. Organization can have inhouse inspection, direct involvement of users and release of beta version are few of them. Testing may include usability,, user acceptance, compatibility etc.

3.1.1 Interface Design of Smart Health Monitoring

Home Page: An attractive welcome screen. The welcome screen is the very first page user views. which involves Service,Doctor profile,Facicities and user login pages followed by the contact details at the bottom. The Home page is as shown in figure.

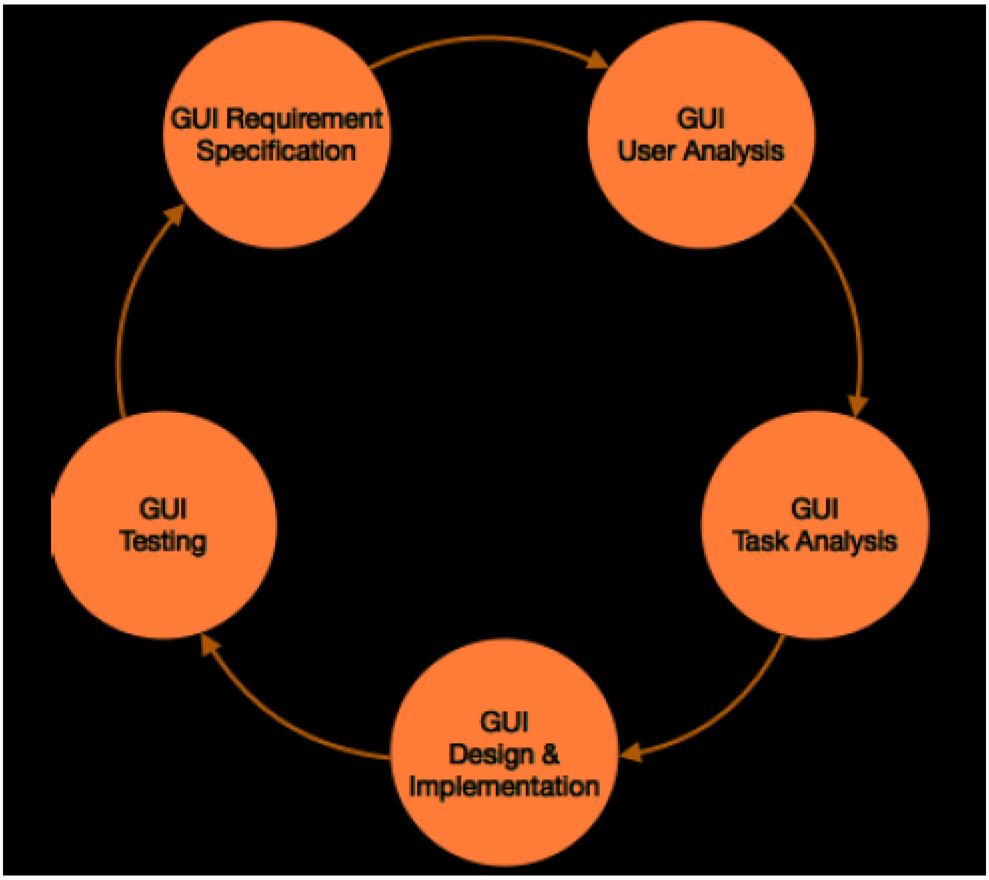
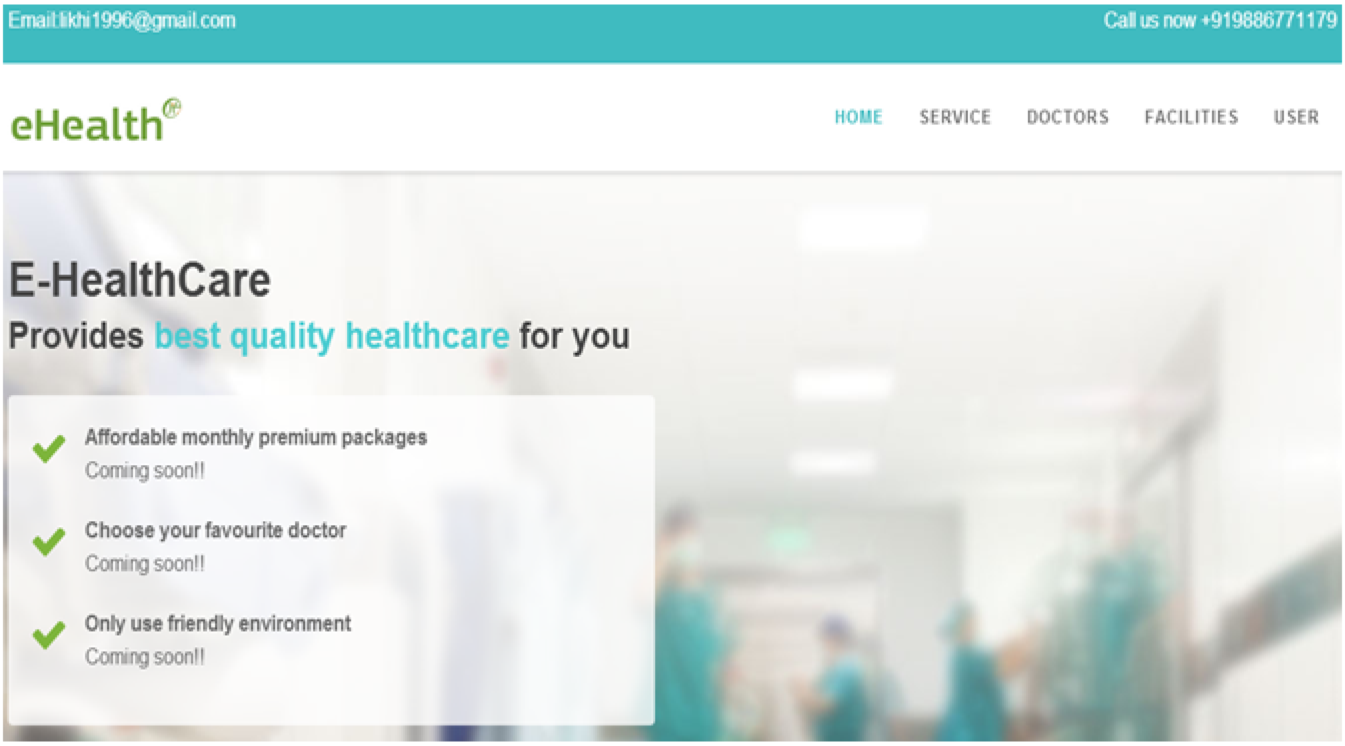


Figure 3.1: Interface design



Email:likhi1996@gmail.com

Call us now +919886771179

eHealth®

HOME SERVICE DOCTORS FACILITIES USER

E-HealthCare

Provides best quality healthcare for you

Affordable monthly premium packages

Coming soon!!

Choose your favourite doctor

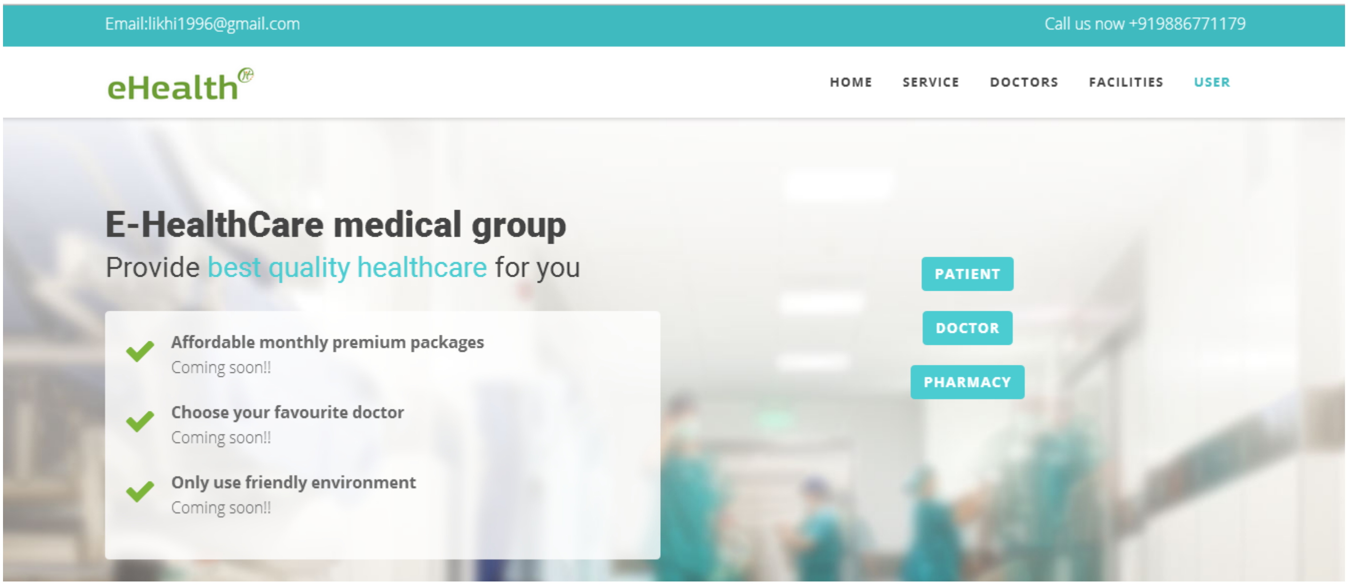
Coming soon!!

Only use friendly environment

Coming soon!!

Figure 3.2: Home Page

Login page: Consists of dialog box to enter the user Id and Password. User can be a patient, doctor or Pharmacist. The login page is as shown in figure



Email:likhi1996@gmail.com

Call us now +919886771179

E-HealthCare medical group

Provide best quality healthcare for you

Affordable monthly premium packages

Coming soon!!

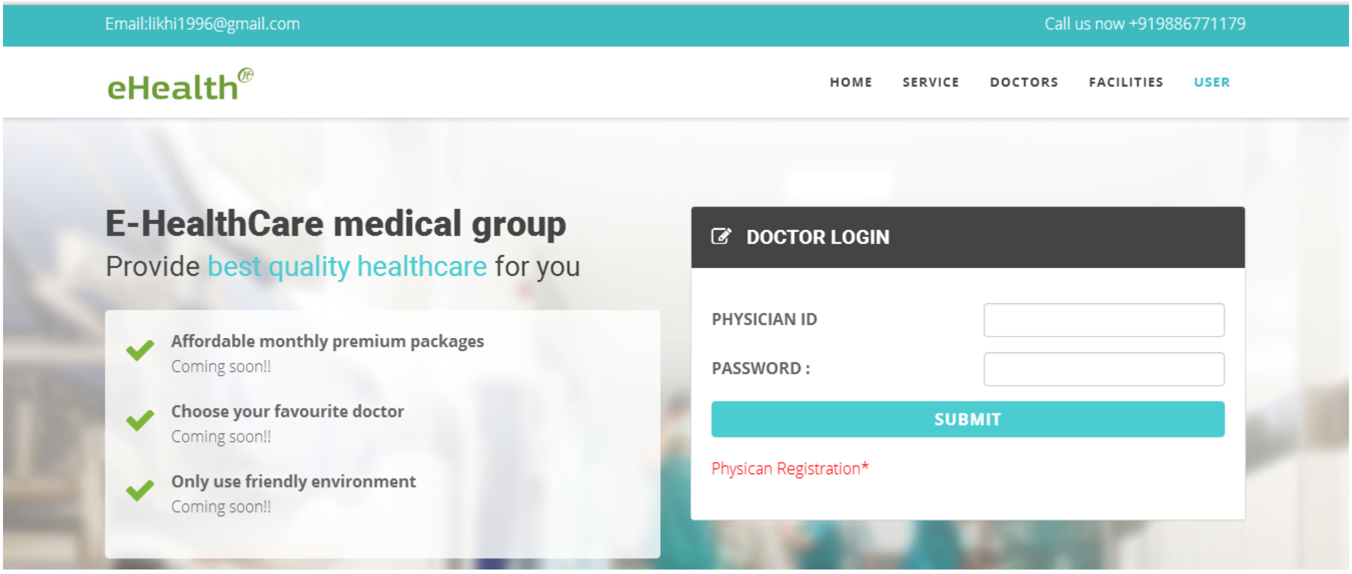
Choose your favourite doctor

Coming soon!!

Only use friendly environment

Coming soon!!

Figure 3.3: User Login selection page



Email:likhi1996@gmail.com

Call us now +919886771179

E-HealthCare medical group

Provide best quality healthcare for you

Affordable monthly premium packages

Coming soon!!

Choose your favourite doctor

Coming soon!!

Only use friendly environment

Coming soon!!

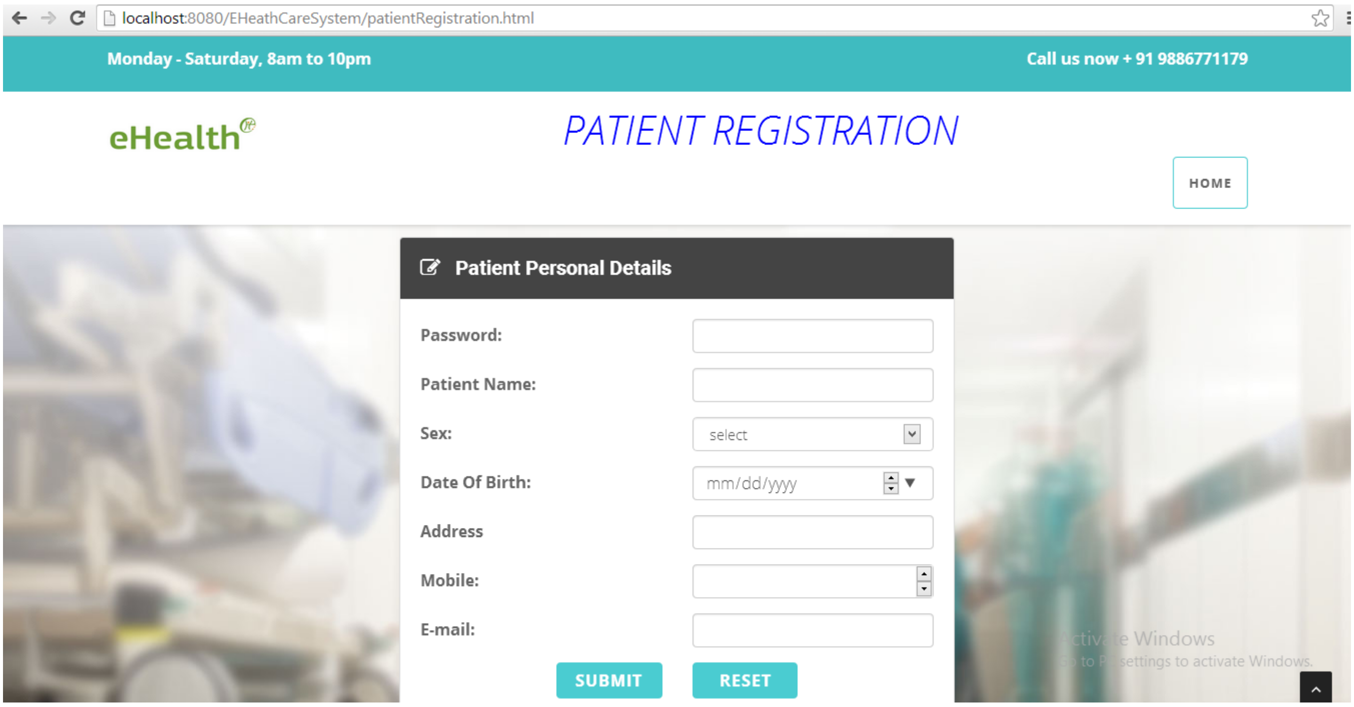
DOCTOR LOGIN

PHYSICIAN ID

PASSWORD :

Figure 3.4: Physician Login page

Registration: New users will be provided an interactive form to fill in there details for registering with the server to access the featurer provided for him. The registration page is as shown in figure.



localhost:8080/EHeathCareSystem/patientRegistration.html

Monday - Saturday, 8am to 10pm

Call us now + 91 9886771179

eHealth

PATIENT REGISTRATION

Patient Personal Details

Password:	
Patient Name:	
Sex:	select
Date Of Birth:	mm/dd/yyyy
Address	
Mobile:	
E-mail:	

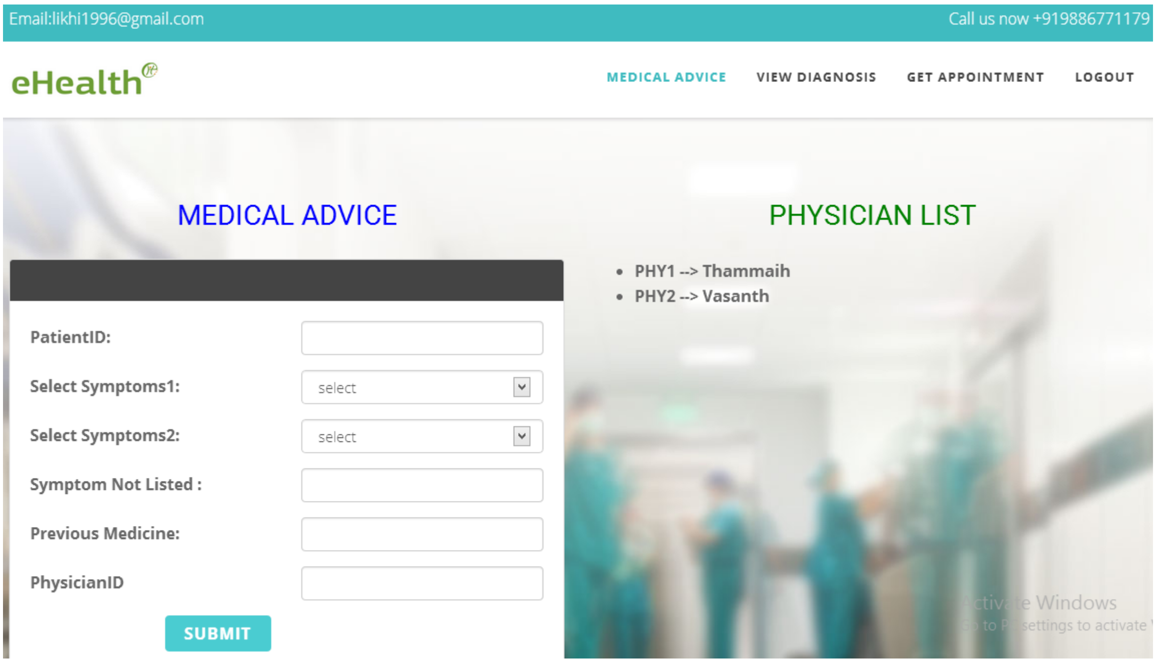
Figure 3.5: Registration page

Functionality: There will be 3 users who will login to the website based on their role. Patient, Physician and Pharmacist. It is discussed as follows:-

1. **Patient:** After Login, patient can choose to book an appointment with a doctor, send the medical advice or give some information to the doctor prior to meeting. Also, he can get the diagnosis details. Its UI is shown in the below figure 3.6.

2. **Physician:** After signing in,Physician can view the cases, get the patient details, send the medical prescription to patient and pharmacist and get the details of apppointments b/w date. Its UI is as shown in the figure 3.7.

3. **Pharmacist:** He can use the website to see the list of patients and their medical prescriptions which indicates the medicine to be given along with dosage and patient Id. He can view the pharmacy details b/w date, all or of a particular day etc. His UI is as shown in figure 3.8.



MEDICAL ADVICE

PatientID:

Select Symptoms1:

Select Symptoms2:

Symptom Not Listed :

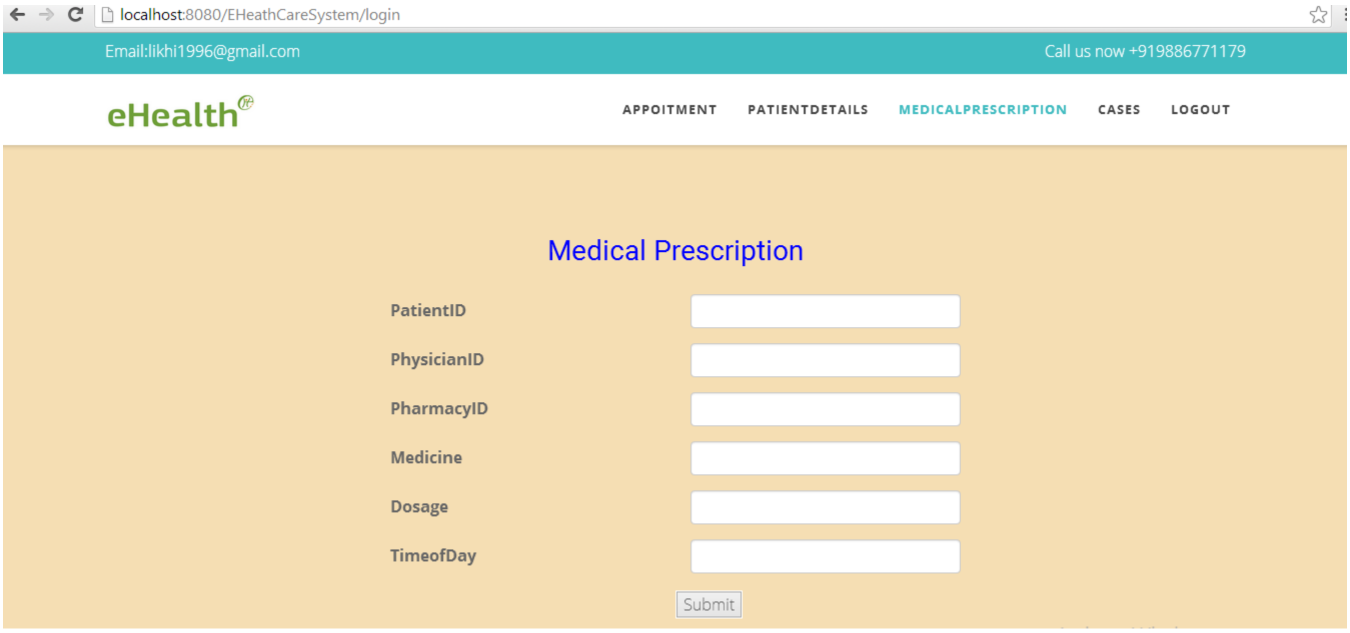
Previous Medicine:

PhysicianID

PHYSICIAN LIST

- PHY1 --> Thammaih
- PHY2 --> Vasanth

Figure 3.6: Patient Login Success Page


localhost:8080/EHeathCareSystem/login
Email:likhi1996@gmail.com
Call us now +919886771179
eHealth
APPOITMENT
PATIENTDETAILS
MEDICALPRESCRIPTION
CASES
LOGOUT
Submit

Medical Prescription

PatientID

PhysicianID

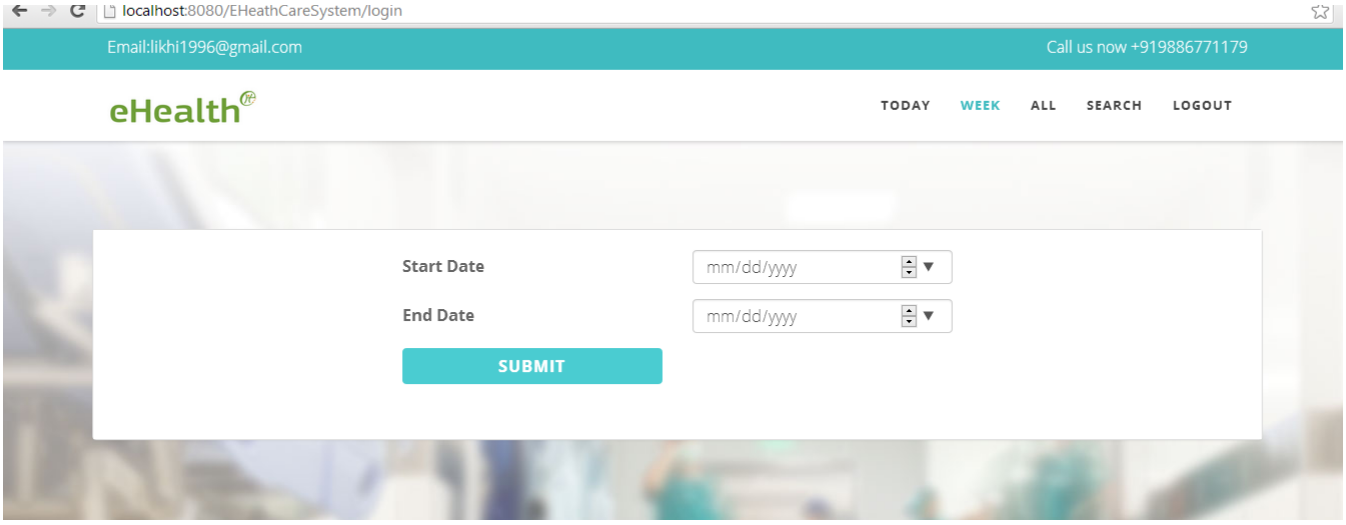
PharmacyID

Medicine

Dosage

TimeofDay

Figure 3.7: Physician's Medical Prescription Page



localhost:8080/EHeathCareSystem/login

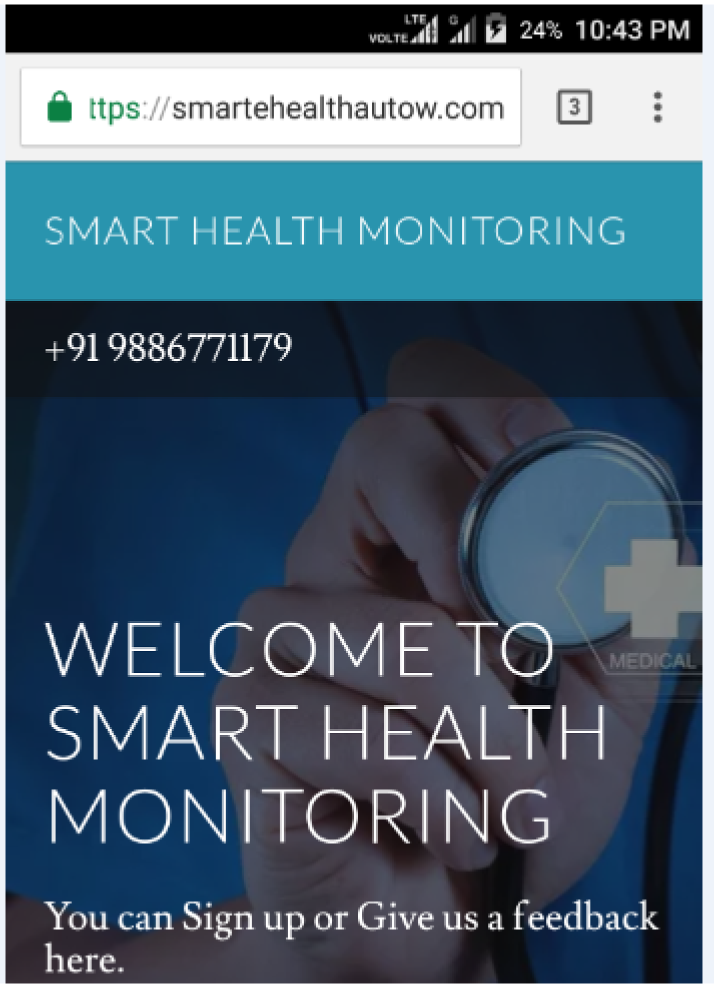
Email:likhi1996@gmail.com

Call us now +919886771179

Start Date mm/dd/yyyy

End Date mm/dd/yyyy

Figure 3.8: Pharmacist Login Success Page



ttps://smartehealthautow.com

SMART HEALTH MONITORING

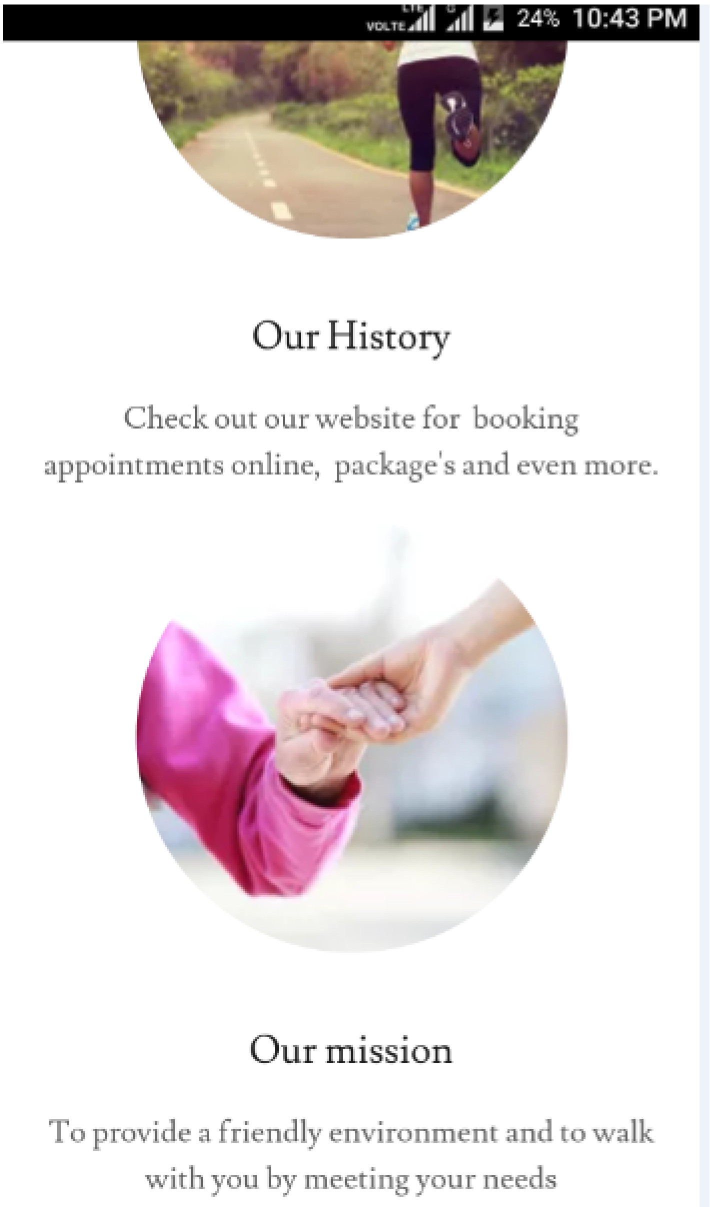
+91 9886771179

WELCOME TO SMART HEALTH MONITORING

You can Sign up or Give us a feedback here.

Figure 3.9: Welcome Page

Feedback : Since this is a user friendly website, we have considered to take the comments or suggestions from our website users. The Interface of that page is as shown in the screenshots(figures)3.9,3.10,3.11 and 3.12 respectively.



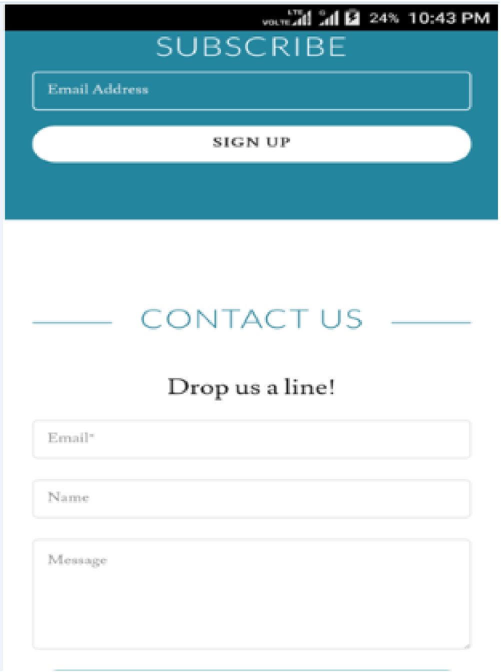
Our History

Check out our website for booking appointments online, package's and even more.

Our mission

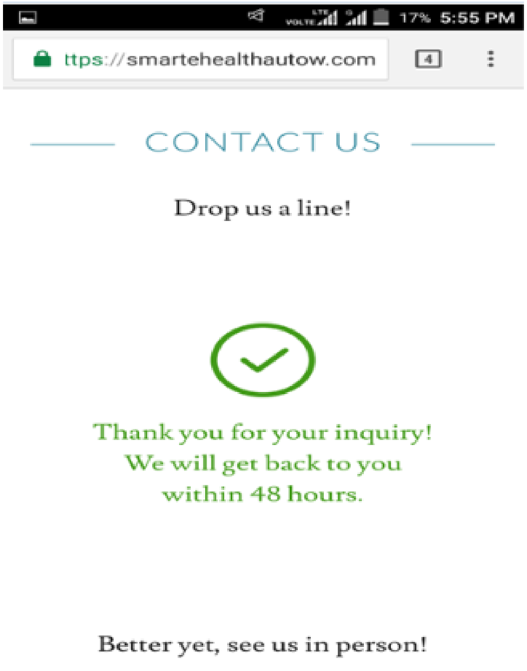
To provide a friendly environment and to walk with you by meeting your needs

Figure 3.10: A Glimpse of Mission and History



Drop us a line!

Figure 3.11: Drop line



Drop us a line!

Thank you for your inquiry!
We will get back to you
within 48 hours.

Better yet, see us in person!

Figure 3.12: Successfull Submisssion of Feedback

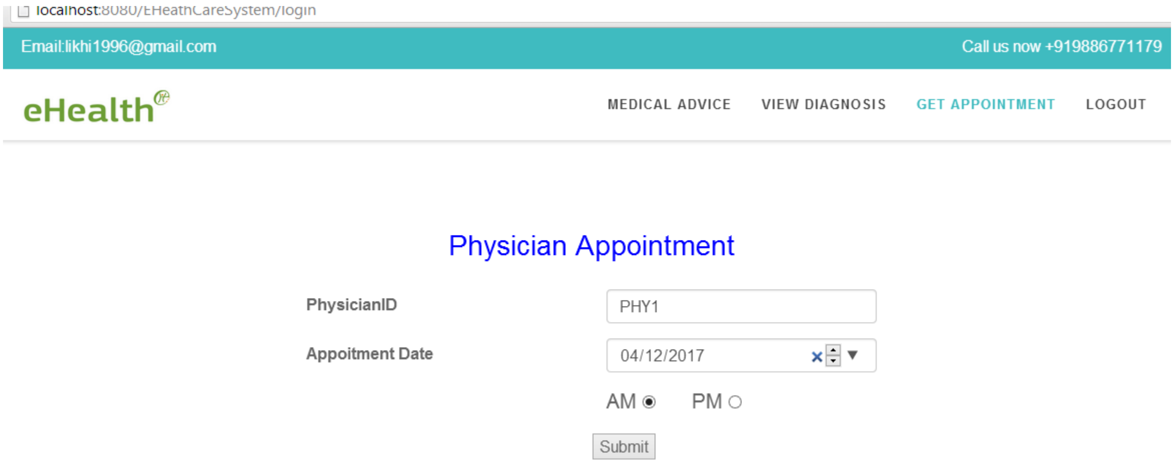
3.2 Functional Requirements

3.2.1 Patient Module

3.2.1.1 Appointment Submission

Purpose: The patient will use this to book an appointment with a respective physician. He may give the details of symptoms and previous medicines along with the appointment date and time.

Error Handling: Booking appointments with Unregistered Physicians is avoided by providing list of registered users.


localhost:8080/EHeathCareSystem/login
Email:likhi1996@gmail.com
Call us now +919886771179
eHealth
MEDICAL ADVICE
VIEW DIAGNOSIS
GET APPOINTMENT
LOGOUT
Submit

Physician Appointment

PhysicianID: PHY1

Appoitment Date: 04/12/2017

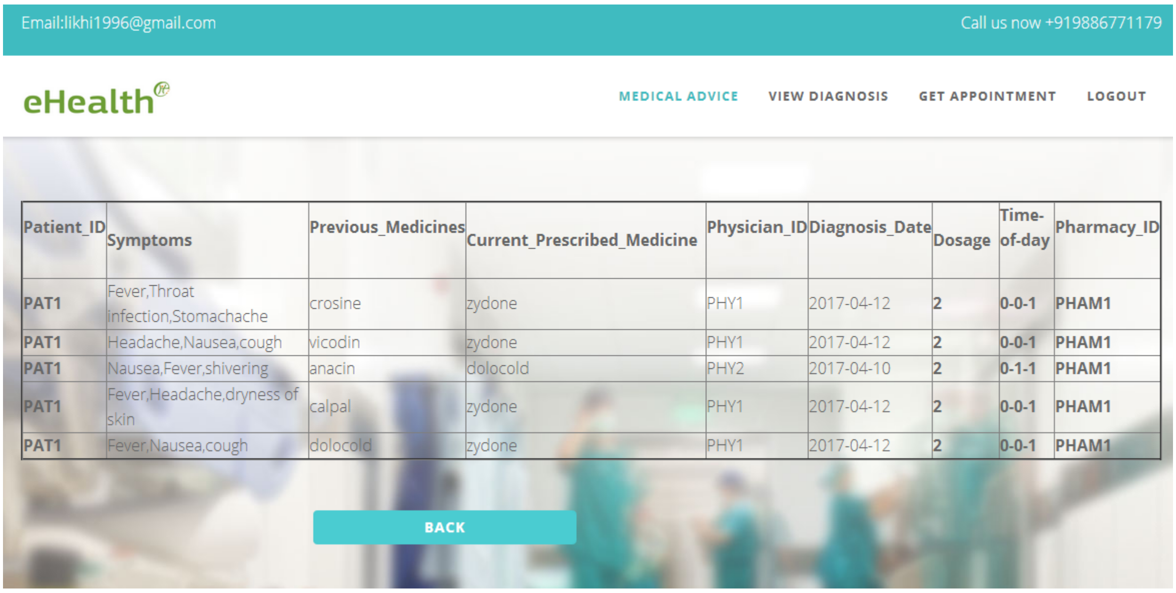
AM ● PM ○

Figure 3.13: Appointment submission

3.2.1.2 View Diagnosis Details

Purpose: The patient can view the new prescription given by the physician he met along with the dosage,time of day he has to take a particular medicine and pharmacy Id etc.

Error Handling: Until the physician submits the prescription, it will not be updated in the view diagnosis page and only that patients details will be available, thus maintaining privacy.



Patient_ID	Symptoms	Previous_Medicines	Current_Prescribed_Medicine	Physician_ID	Diagnosis_Date	Dosage	Time-of-day	Pharmacy_ID
PAT1	Fever,Throat infection,Stomachache	crosine	zydone	PHY1	2017-04-12	2	0-0-1	PHAM1
PAT1	Headache,Nausea,cough	vicodin	zydone	PHY1	2017-04-12	2	0-0-1	PHAM1
PAT1	Nausea,Fever,shivering	anacin	dolocold	PHY2	2017-04-10	2	0-1-1	PHAM1
PAT1	Fever,Headache,dryness of skin	calpal	zydone	PHY1	2017-04-12	2	0-0-1	PHAM1
PAT1	Fever,Nausea,cough	dolocold	zydone	PHY1	2017-04-12	2	0-0-1	PHAM1

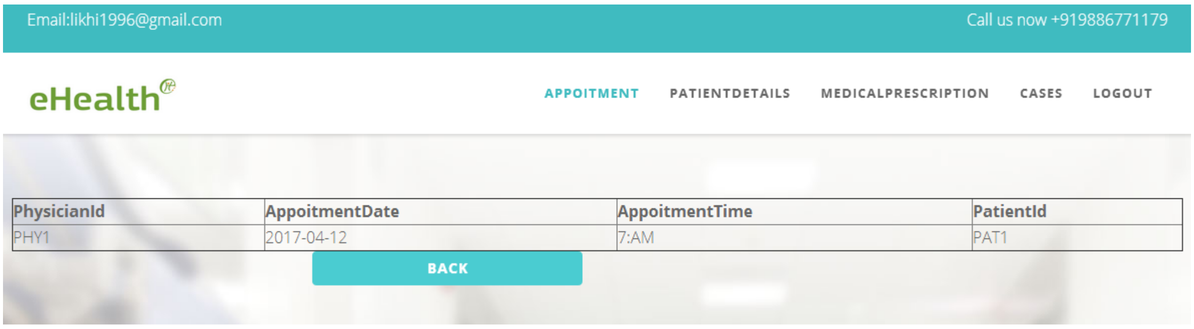
Figure 3.14: View Diagnosis details

3.2.2 Physician Module

3.2.2.1 Check Today's Patient

Purpose: The Physician can check the appointments for that particular day in order with the help of this function.

Error Handling: If there are no appointments an dialog box will be dispayed saying ”there are no cases”.



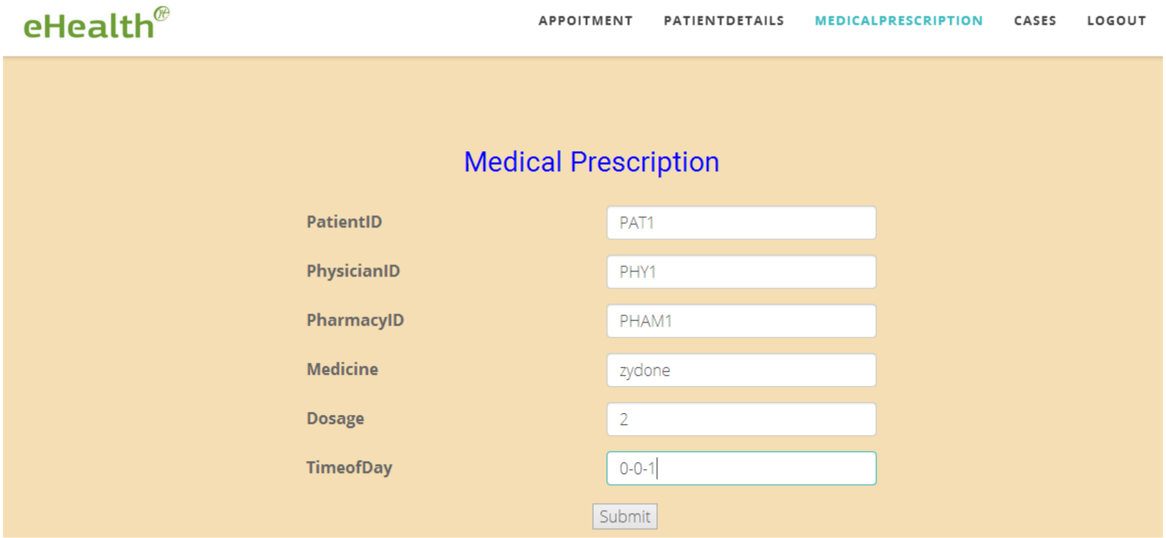
PhysicianId	AppoitmentDate	AppoitmentTime	PatientId
PHY1	2017-04-12	7:AM	PAT1

Figure 3.15: Today's Patient

3.2.2.2 Medical Prescription Submission

Purpose: This is used by the physician to submit the prescription to respective patient server via online and also related data will get transfered to the specified pharmacist server.

Error Handling: The details will be updated in that particular patient server if successfully submitted by physician else it will not be updated. The same happens in case of pharmacist



Medical Prescription

PatientID	PAT1
PhysicianID	PHY1
PharmacyID	PHAM1
Medicine	zydone
Dosage	2
TimeofDay	0-0-1

Figure 3.16: Medical Prescription Submission

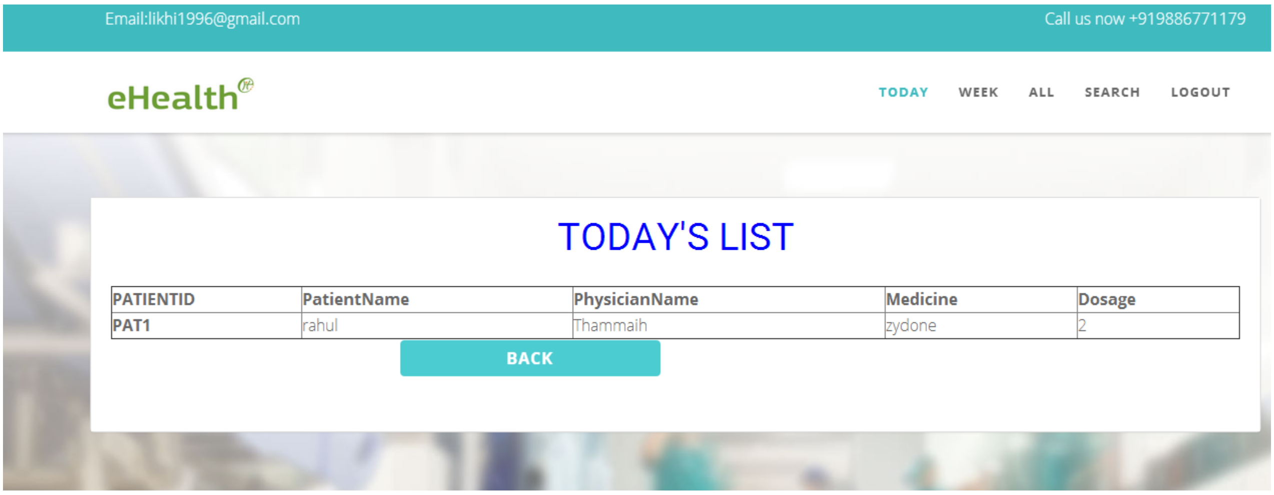
3.2.3 Pharmacist Module

3.2.3.1 View Todays Pharmacy

Purpose: This will be used by the pharmacist to check the details of the patient and their medicines which he need to deliver when that particular patient arrives on the current day.
Error Handling: Only current day's details are displayed. If there are no records for that day an empty message is displayed.

3.3.3.2 View All Pharmacy

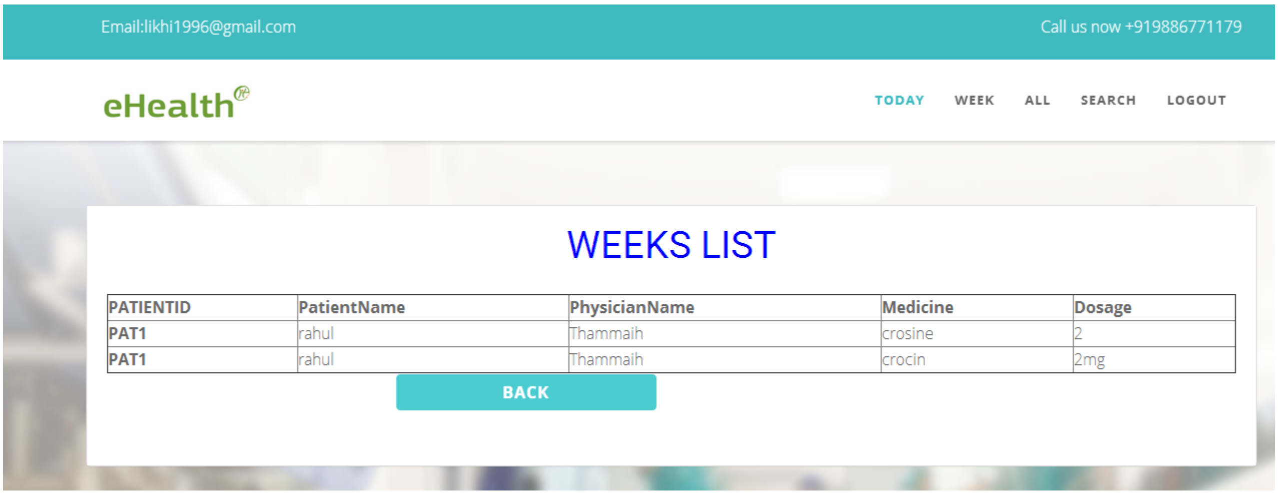
Purpose: This will be used by the pharmacist to check the details of the patient and their medicines which he need to deliver or has delivered during that week.
Error Handling: Only current weeks details are displayed. If there are no records for that week an empty message is displayed.



TODAY'S LIST

PATIENTID	PatientName	PhysicianName	Medicine	Dosage
PAT1	rahul	Thammaih	zydone	2

Figure 3.17: Today's List



WEEKS LIST

PATIENTID	PatientName	PhysicianName	Medicine	Dosage
PAT1	rahul	Thammaih	crosine	2
PAT1	rahul	Thammaih	crocin	2mg

Figure 3.18: Week List

3.3 UML diagrams with discussions

3.3.1 Activity diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows).Activity diagrams show the overall flow of control. Activity diagrams may be regarded as a form of flowchart. Typical flowchart techniques lack constructs for expressing concurrency. However, the join and split symbols in activity diagrams only resolve this for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops.

The figure 3.19 below illustrates the activity diagram of our project in a general manner right from login to logout and what are all the activities each individual can perform in their workpace.

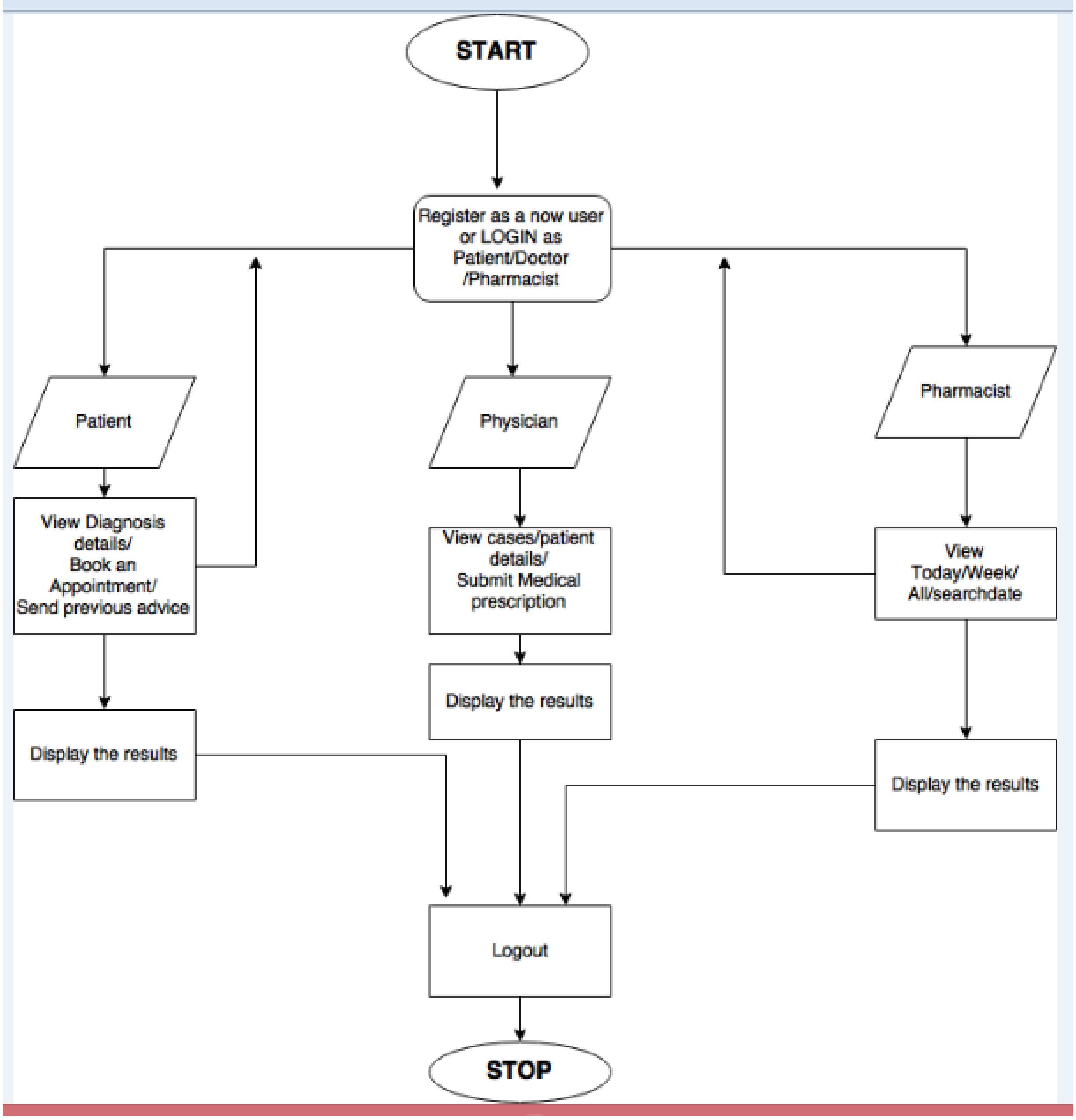

START
Register as a now user or LOGIN as Patient/Doctor /Pharmacist
Patient
Physician
Pharmacist
View Diagnosis details/ Book an Appointment/ Send previous advice
View cases/patient details/ Submit Medical prescription
View Today/Week/ All/searchdate
Display the results
Display the results
Display the results
Logout
STOP

Figure 3.19: Activity Diagram

3.3.2 Sequence Diagrams

Sequence diagrams describe interaction between different components of the module in terms of messages over time. They help to predict how system behaves during the process. Here the blocks represent the objects involved. They describe the way an object will behave in the context. The vertical dotted line, called lifeline represents the persistency of the component. Activation boxes represent the time needed for an object to complete the task. The data exchange that happens between each of the components is represented by horizontal arrows along with the messages.

Few sequences diagrams of our project are as shown in figures 3.20,3.21,3.22.

3.3.2.1 Sequence diagram for Appointment

The figure 3.19 shows the process flow in patient module. After successful login, patient will book an appointment, the data will be stored in the database and validated. The patient will recieve a message as 'Appointment submitted successfully' on completion.

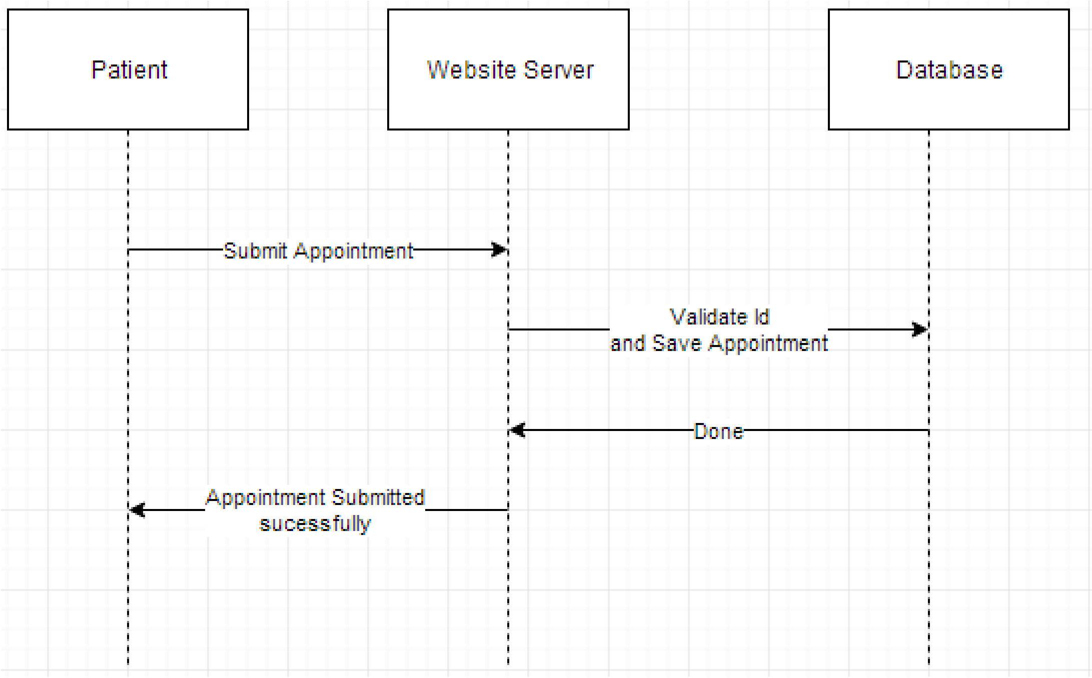


Figure 3.20: Sequence Diagram for Appointment

3.3.2.2 Sequence diagram for Medical Prescription

The figure 3.20 shows the process flow in physician module. After successful login, doctor will fill the prescription, the data will be stored in the database and validated. The physician/doctor will recieve a message as 'Medical prescription submitted successfully' on completion.

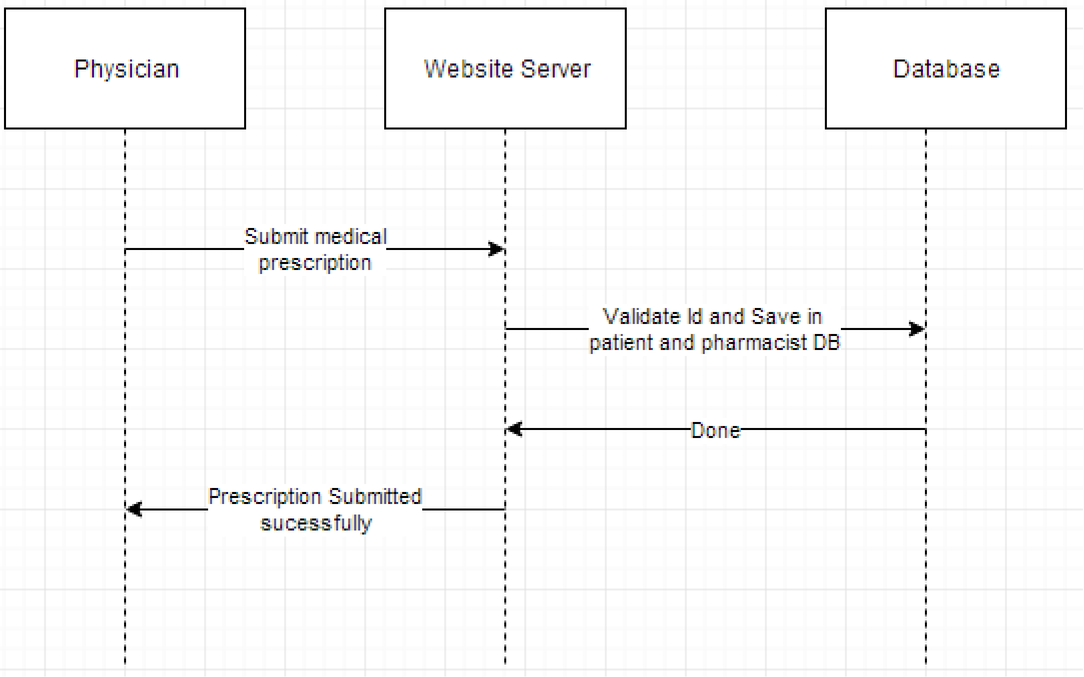

Physician
Website Server
Database
Submit medical prescription
Validate Id and Save in patient and pharmacist DB
Done
Prescription Submitted sucessfully

Figure 3.21: Sequence Diagram for Medical Prescription

3.3.2.3 Sequence diagram for Pharmacy Details

The figure 3.21 shows the process flow in pharmacist module. After successful login, pharmacist search for either todays/week's/all/particular details of the medicines and patients list, the data will be retrieved from the database. The pharmacist will recieve a message as 'Details updated successfully' on completion.

3.3.3 Class Diagrams

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. In class diagram, classes are represented with boxes that contain three compartments:

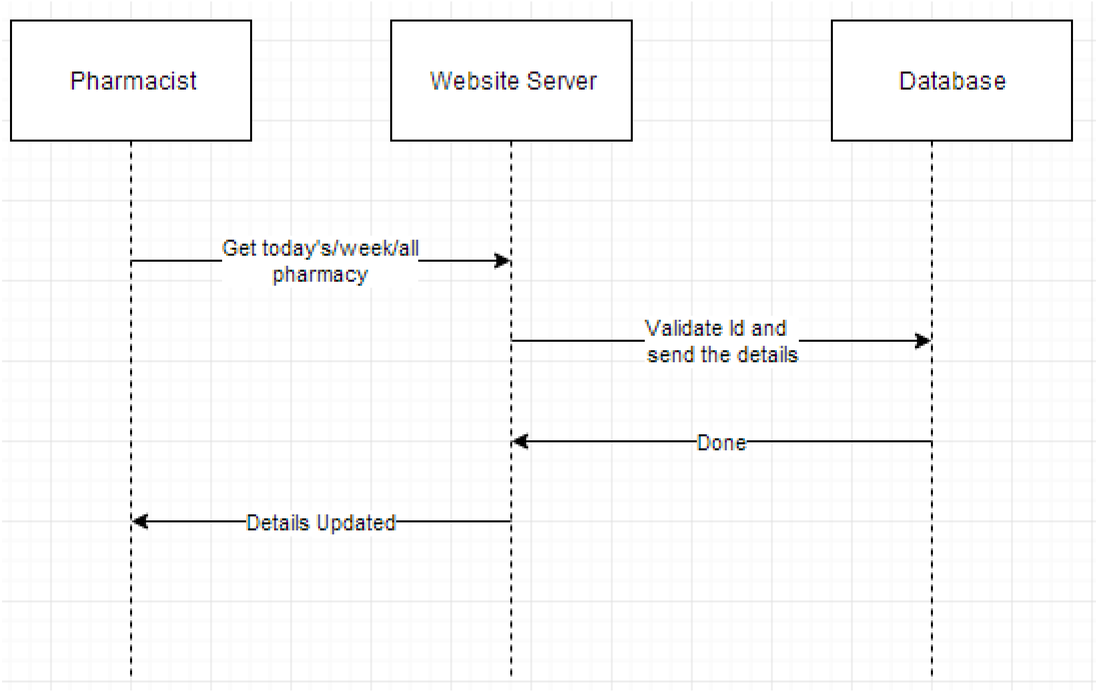

Pharmacist
Website Server
Database
Get today's/week/all pharmacy
Validate Id and send the details
Done
Details Updated

Figure 3.22: Sequence Diagram for Pharmacy Details

1. The top compartment contains the name of the class.

2. The middle compartment contains the attributes of the class.

3. The bottom compartment contains the operations the class can execute.

The figure shows the three main classes of smart health monitoring.

1. **Name of the class:** Patient

   Attributes: Password, unique phone id.

   Operations: Get diagnosis,get appointment,submit any advice.

2. **Name of the class:** Physician

   Attributes: Password, unique PHY id.

   Operations: Get cases, get patient details, submit prescription.

3. **Name of the class:** Pharmacist

   Attributes: Password, unique PHAM id.

   Operations: View todays pharmacy,weeks pharmacy,all pharmacy.

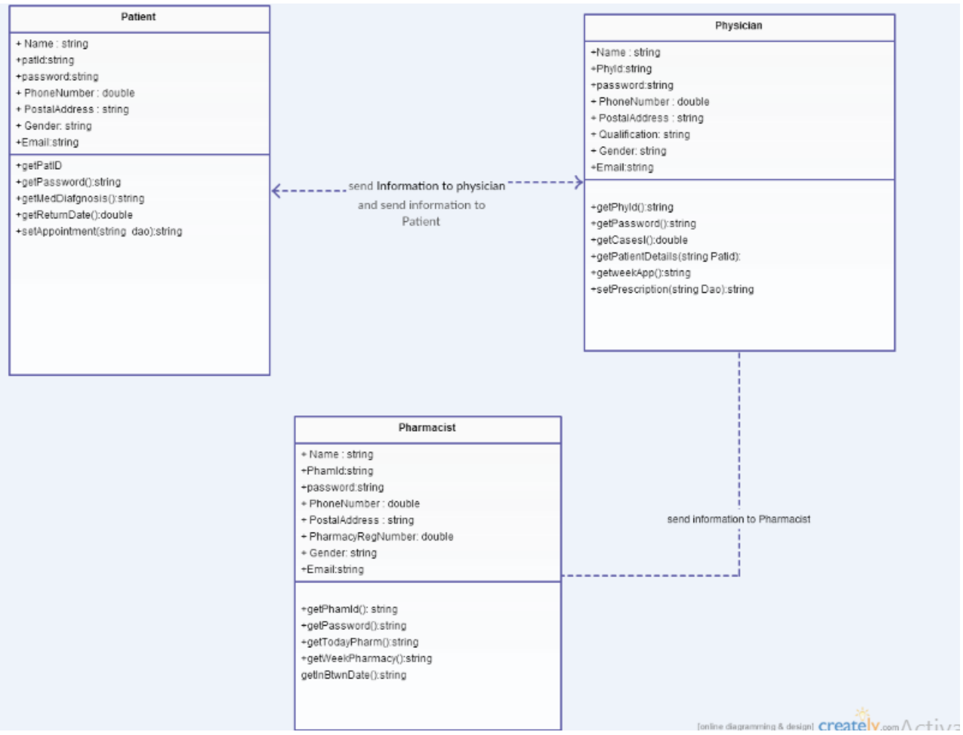


Figure 3.23: Class Diagram

3.4 Data Source/Database used and Formats

3.4.1 Data Source

In computer programming, source data or data source is the primary location from where data comes. The data source can be a database, a dataset, a spreadsheet or even hard-coded data. When data is displayed in a web page or application, in a column-row format or other formats, the data is retrieved from its data source and presented in the format defined in the code. A common type of database is an SQL database, but some applications can use other types of databases, like Microsoft Access. While less stable, a series of spreadsheets can be used as a data source, but this is less common due to the chance of spreadsheets becoming corrupted as their size increases. Databases are often part of a larger data backup system and can be maintained more efficiently, thus making them more popular choices as data sources. Data source is a name given to the connection set up to a Database from a server. The name is commonly used when creating a query to the database.

Smart Health Monitoring extracts data dynamically from the server.When the desired event is triggered, the data is intimated to the local database. At the end of the event, the data at the local database, that reflects the behaviour of the event performed in that web page is updated to the central server in the mysql database.Here are some of the important events that trigger the data for the project Smart Health Monotoring.

1. When a new user registers as a patient or doctor or pharmacist.
2. When patient gives details regarding his symptoms and appointments.
3. When doctor gives the prescription to the patient.
4. When doctor forwards the medicine details to the pharmacist.

3.4.2 Database used and Formats

MySQL for central database server

MySQL is a database system used by many websites on the Internet. It is based on SQL. Many ways of doing things in SQL are similar in MySQL. MySQL is a free, widely used SQL engine. It can be used as a fast database as well as a rock-solid DBMS using a modular engine architecture and JDBC is used to perform the related queries.

Smart Health Monotoring aims to use the local server software like Apache Tomcat server and free servers that provided by the service providers like Hostinger and GoDaddy. The Hostinger will help us in hosting the main website and which stores the database in its cloud. The GoDaddy's website is used as suppporting website for Smart Health Monitoring System which whose page will be redirected from the main website.

Chapter 4

Implementation

4.1 Tools and Technologies

4.1.1 Eclipse Indigo

The Eclipse Indigo is an free and open-source integrated development environment (IDE) for developing (especially) Java applications. The project ”Smart Health Monitoring” uses Eclipse for its back end and front end programming.

4.1.2 Database

We have used ’SqlYog’ which is a GUI tool for the RDBMS MySQL. In our application, all the user data are being saved in to this database which contains many tables with respect to the application.

4.1.3 Client/Server Technology

The Client/Server Technology used in our web application is jsp(java server pages) and servlet. Tomcat-7 is used to create the server. We will run in the website in the localhost using this technology.

4.1.4 User Interface

We have used HTML 5 and CSS 3(Cascading Style Sheets) for designing the front end of our application. The CSS is a advanced version usually used along with html for a better UI

4.1.5 JUnit 4.0

It is a testing tool provided by Eclipse for testing our application before running it in the browser i.e. without running the server.

4.2 Coding Standards Followed

The coding standards, sometimes referred to as programming styles or coding conventions, are a very important asset in writing any code. The most prominent example of this is Mozilla Firefox, the open source internet browser which makes all its code freely available online. This is so other developers can look over the source code, possibly identifying possible security flaw in the code as well as extending the code. It would be highly difficult for other developers to recognizing potential flaws in the program if it did not follow a particular style convention. Also, Firefox has a huge database of extensions, some of which it created itself but many of which were created by external developers. External extensions would be very complicated to understand and integrate into Firefox if they did not follow a coding standard, and Firefox may not have been such a popular internet browser as it is today. Some of the coding standards that has been followed is mentioned below.

4.2.1 Eclipse and Java Language

We follow standard Java coding conventions. We have added a few Java specific rules.

Package and Import Statements

The first non-comment line of Java source files is a package statement, followed by import statements.

package com.ehealth.controller;
import java.io.IOException;

Project Structure

The Application involes various kinds of files and classes which will be integrated later. The Project Structure involves the following:-

1. **Class Files:** The Class file name is same as the component name. For eg: PatientRegistrationServlet.Java, CasesServlet.Java

2. **Drawable Files:** It involves the assests, which can be bullets, box or any Pictures in PNG format. For eg: bullet.png, timer.png

3. **Xml and Html files** we have a single servlet class and multiple html clases. For eg: web.xml,intex-form.html,patientLogin.html

The Java Langauage involves:-

1. Fully qualify imports: This project uses fully qualified imports. For eg:import java.io.IOException; import java.text.ParseException; import java.text.SimpleDateFormat; import java.util.AbstractList;

2. Annotations: Predefined annotations used in the project is: @Override protected void doPost(HttpServletRequest request, HttpServletResponse response)

3. Log: Logging methods provided by the Log class to print out error messages that are used in the project. For eg: Log.e(String tag, String msg) (error)

4.3 Code Integration Details

This project uses Continuous Integration(CI) approach for integration. All the modules were coded seperately and posted at the central version control(CVS). We have used many pacakages like Controller, Service, Data Access Objects(DAO) and Transfer Object which consists of modules related to the appplication. Though they are separated we will map them by xml and they work concurrently when run on a server.

Unit tests were conducted on each individual modules before integrating it with other part of the project.

The Xml code for servelt-mapping is as shown:-

```
<servlet-name>patientservlet</servlet-name>
<servlet-class>com.ehealth.controller.PatientRegistrationServlet
</servlet-class>
</servlet>
```

```
<servlet-name>doctorservlet</servlet-name>
<servlet-class>com.ehealth.controller.DoctorRegistrationServlet
</servlet-class>
</servlet>

<servlet-mapping><servlet-name>patientservlet>/servlet-name>
<url-pattern>/patient>/url-pattern>
</servlet-mapping>

<servlet-mapping>
<servlet-name>doctorservlet</servlet-name>
<url-pattern>/doctor>/url-pattern>
</servlet-mapping >
```

4.4 Implementation Work Flow

4.4.1 Environment setup

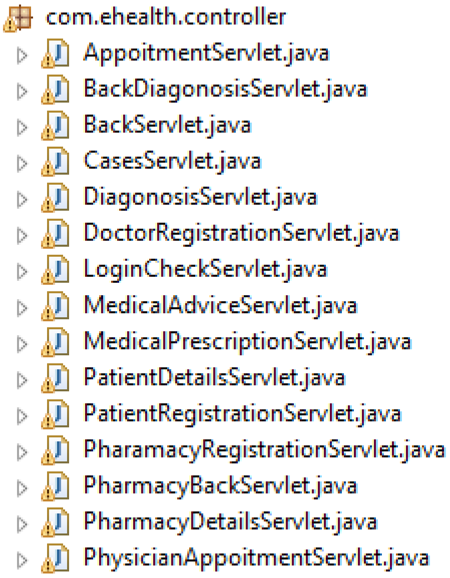
The development environment was installed and setup.Tomcat-7 server was created on which the application runs and is executed in the browser(google chrome or mozilla firefox).

4.4.2 Project Setup and Development

In our project we have made several packages and distributed the code computation in such a way that it is easy to find errors, find a particular module or class in the code and can be easily understood by anyone. The various packages used and their modules working concurrently for the same application is discussed here.

Controller

This is used to connect to the sever and linking the page to the respective java server pages based on the click made by the user. Every activity that is performed in the website is made as a java file. The various files/classes used under this package is as shown in figure.

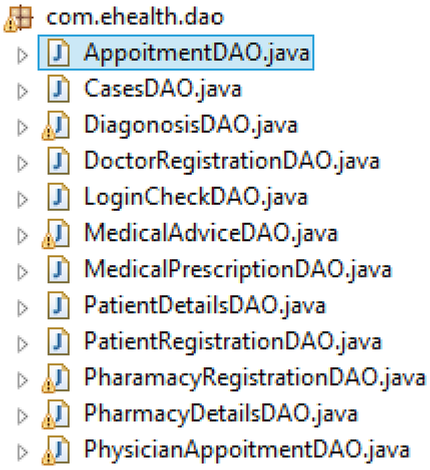


```
com.ehealth.controller
  AppoitmentServlet.java
  BackDiagonosisServlet.java
  BackServlet.java
  CasesServlet.java
  DiagonosisServlet.java
  DoctorRegistrationServlet.java
  LoginCheckServlet.java
  MedicalAdviceServlet.java
  MedicalPrescriptionServlet.java
  PatientDetailsServlet.java
  PatientRegistrationServlet.java
  PharamacyRegistrationServlet.java
  PharmacyBackServlet.java
  PharmacyDetailsServlet.java
  PhysicianAppoitmentServlet.java
```

Figure 4.1: Controller package

Data Access Object

Any queries that should be performed are done here. The contents that are fetched are being copied to an object and later the related fields are displayed based on the situation. The various files/classes used under this package is as shown in figure.

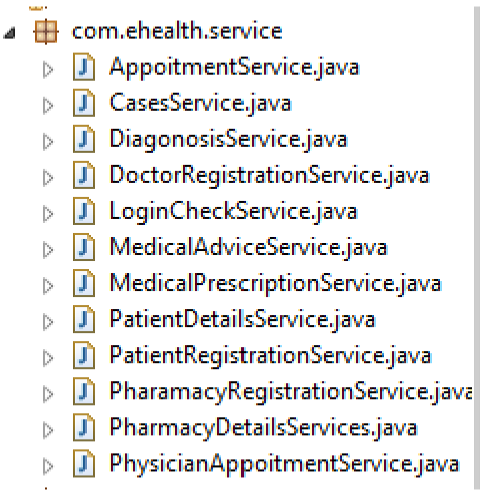


```
com.ehealth.dao
  AppoitmentDAO.java
  CasesDAO.java
  DiagonosisDAO.java
  DoctorRegistrationDAO.java
  LoginCheckDAO.java
  MedicalAdviceDAO.java
  MedicalPrescriptionDAO.java
  PatientDetailsDAO.java
  PatientRegistrationDAO.java
  PharamacyRegistrationDAO.java
  PharmacyDetailsDAO.java
  PhysicianAppoitmentDAO.java
```

Figure 4.2: Data Access Object Package

Service

This just returns the string to the Data access object class. The various files/classes used under this package is as shown in figure.

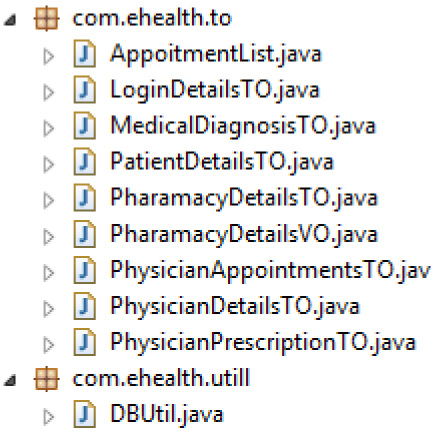


- com.ehealth.service
  - AppoitmentService.java
  - CasesService.java
  - DiagonosisService.java
  - DoctorRegistrationService.java
  - LoginCheckService.java
  - MedicalAdviceService.java
  - MedicalPrescriptionService.java
  - PatientDetailsService.java
  - PatientRegistrationService.java
  - PharamacyRegistrationService.java
  - PharmacyDetailsServices.java
  - PhysicianAppoitmentService.java

Figure 4.3: Service package

Transfering Objects and Database Connection

The required parameters that we want are declared in the class and all of them are transfered as an object at the end. The Util file is used to set the conncetion to the database using the valid username and password. The various files/classes used under this package is as shown in figure.

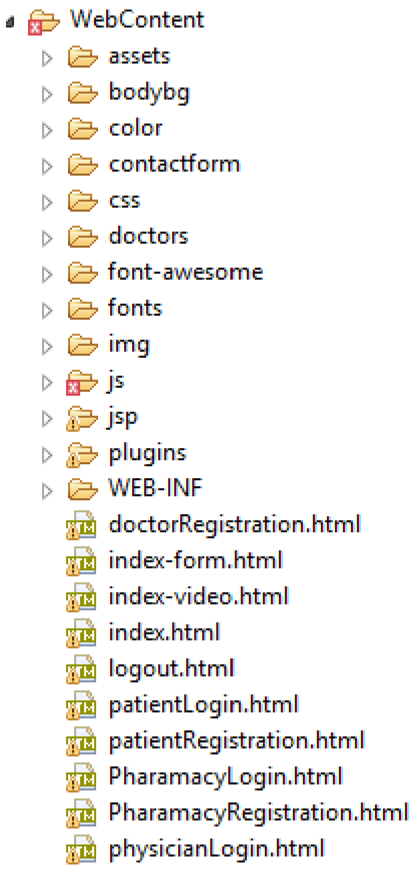


- com.ehealth.to
  - AppoitmentList.java
  - LoginDetailsTO.java
  - MedicalDiagnosisTO.java
  - PatientDetailsTO.java
  - PharamacyDetailsTO.java
  - PharamacyDetailsVO.java
  - PhysicianAppointmentsTO.jav
  - PhysicianDetailsTO.java
  - PhysicianPrescriptionTO.java
- com.ehealth.utill
  - DBUtil.java

Figure 4.4: Transfer Object and Util package

Web Contents

In this folder we have CSS,fonts,colour,assests and other .png format pictures. The Web-Inf folder contains the html files for different user interface. The various files/classes used under this package is as shown in figure.



```
WebContent
  assets
  bodybg
  color
  contactform
  css
  doctors
  font-awesome
  fonts
  img
  js
  jsp
  plugins
  WEB-INF
  doctorRegistration.html
  index-form.html
  index-video.html
  index.html
  logout.html
  patientLogin.html
  patientRegistration.html
  PharamacyLogin.html
  PharamacyRegistration.html
  physicianLogin.html
```

Figure 4.5: Web Contents

4.4.3 Building

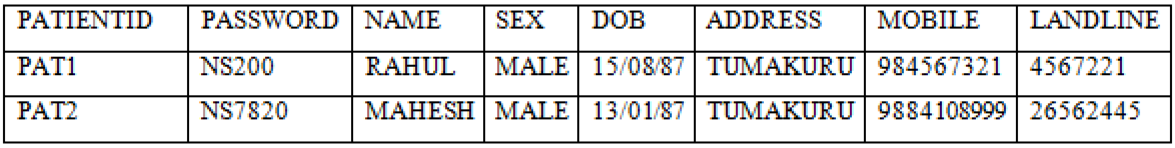
The project was built and run on the tomcat-7 server in debug mode. The log messages(logcat) were used to debug the application.The execution was checked using the Google Chrome or Mozilla Firefox browser.

4.5 Execution Results and Discussions

After Implementation, the code is executed. The data is entered by the respective user and the results are being checked in various tables. Here we dicuss the results of the patient details, physician appointments, medical diagnosis, physician details, physician prescription etc.

4.5.1 Patient Details

The figure 4.6. shows the details of the patient after he/she has successfull registeration in the website. Any blank data which he/she has not filled will appear as null.These details will be filled by the patient when he/she is a new user to the system. He/She will click on the registration link and submit it to the admin. These details are also viewed by the physician who he is going to daignoize a particular patient who has booked an apppointment with him.

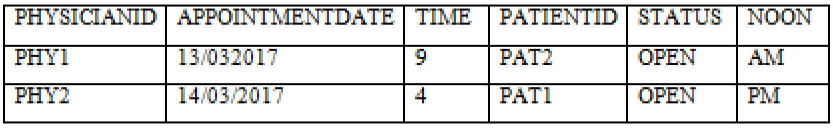


PATIENTID	PASSWORD	NAME	SEX	DOB	ADDRESS	MOBILE	LANDLINE
PAT1	NS200	RAHUL	MALE	15/08/87	TUMAKURU	984567321	4567221
PAT2	NS7820	MAHESH	MALE	13/01/87	TUMAKURU	9884108999	26562445

Figure 4.6: Patient Details

4.5.2 Physician Appointments

The figure 4.7 shows the physician appointment details. Any blank data which he/she has not filled will appear as null.This will be available when a particular patient will book an appointment with a particular physician.It shows the Physician Id, Appointment date, Time, Patient Id, Status etc. The physician will be diagnoizing only the patients whose status are active at that particular moment of time.

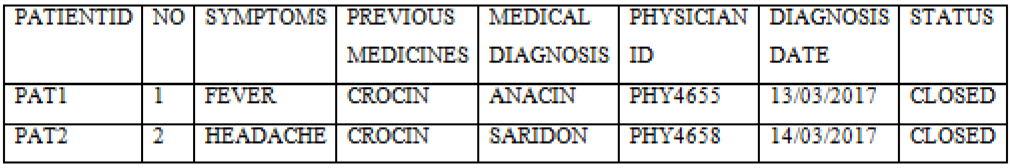


PHYSICIANID	APPOINTMENTDATE	TIME	PATIENTID	STATUS	NOON
PHY1	13/032017	9	PAT2	OPEN	AM
PHY2	14/03/2017	4	PAT1	OPEN	PM

Figure 4.7: Physician Appointments

4.5.3 Medical Diagnosis

The figure 4.8 shows the Medical Diagnosis details. Any blank data which he has not filled will appear as null.This will be available after a particular patient has booked an appointment with a particular physician and that physician fills in the prescription and forwards to the patient and the pharmacist. It mainly shows the patientId, symptoms, previous medicines, current medicine, diagnosis date, physician Id etc.

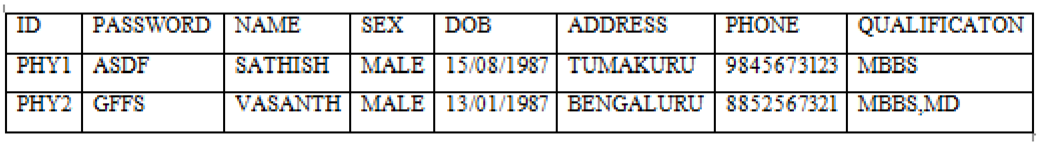


PATIENTID	NO	SYMPTOMS	PREVIOUS MEDICINES	MEDICAL DIAGNOSIS	PHYSICIAN ID	DIAGNOSIS DATE	STATUS
PAT1	1	FEVER	CROCIN	ANACIN	PHY4655	13/03/2017	CLOSED
PAT2	2	HEADACHE	CROCIN	SARIDON	PHY4658	14/03/2017	CLOSED

Figure 4.8: Medical Diagnosis

4.5.4 Physician Details

The figure 4.9. shows the details of the physician after he/she has successfull registeration in the website. Any blank data which he/she has not filled will appear as null.These details will be filled by the physician when he/she is a new user to the system. He/She will click on the registration link and submit it to the admin. These details will be available to any user and can view their achievements, area of specialization wtc in the doctor profile section.

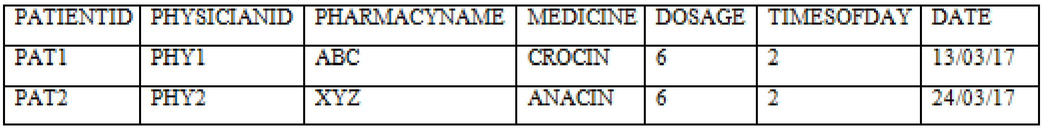


ID	PASSWORD	NAME	SEX	DOB	ADDRESS	PHONE	QUALIFICATON
PHY1	ASDF	SATHISH	MALE	15/08/1987	TUMAKURU	9845673123	MBBS
PHY2	GFFS	VASANTH	MALE	13/01/1987	BENGALURU	8852567321	MBBS,MD

Figure 4.9: Physician Details Table

4.5.5 Physician Prescription

The figure 4.10 shows the physician prescription details. Any blank data which he/she has not filled will appear as null.This will be available when a particular physician forwads the details of particular patient to the pharmacist and the patient himself.It mainly involves Id's of physician,patient and pharmacist along with medicine, dosage, date and time of day the medicine has to be taken.



PATIENTID	PHYSICIANID	PHARMACYNAME	MEDICINE	DOSAGE	TIMESOFDAY	DATE
PAT1	PHY1	ABC	CROCIN	6	2	13/03/17
PAT2	PHY2	XYZ	ANACIN	6	2	24/03/17

Figure 4.10: Physician Prescription Table

4.6 Non-functional requirements results

1. **Performance:** The response times, processing times and reporting times of the application are as per expectation since it did not show any considerable delay.

2. **Economical:** The application has been designed using Eclipse Indigo as an web application. It does not require any additional infrastructure and hence can be deployed economically.

3. **Usability:** The user interface is designed using CSS, which gives a user friendly environment to understand, learn and operate the application.

4. **Mantainability:** The application as a whole is implemented as different modules. So any changes can be done easily without manupulating the other parts of the code

5. **Storage:** The application stores the data of patients,doctors and pharmacists in separate database tables and also their related information in seperate tables which can be accessed.

6. **Availability:** The application is accessible by every valid user in his or her PC whenever he wishes to and is connected to the Internet.

Chapter 5

Testing

5.1 Test Workflow

Testing is a must before it can be used by the people or any organization. In order to remove the errors or defects present in various phases we have the concept of levels of testing. The basic levels of testing is shown below.

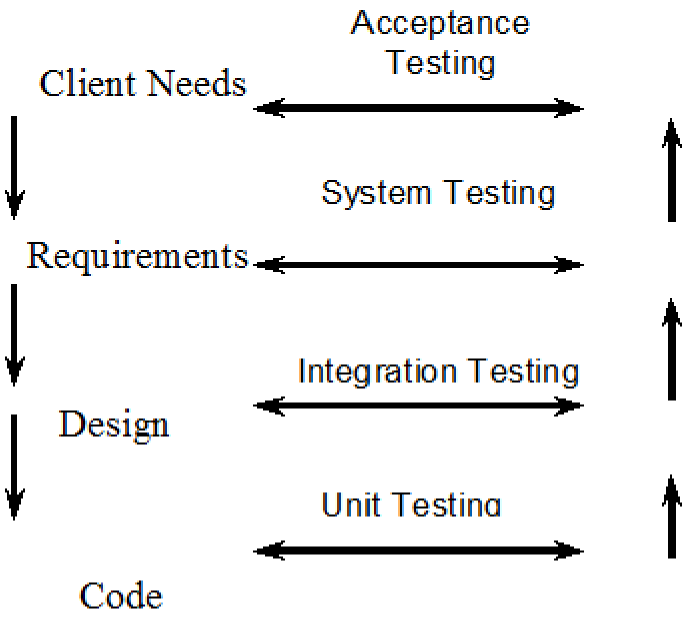


Figure 5.1: Levels of Testing

The test objectives are:-

1. All the field entries must work correctly.
2. The pages must be activated from the specified link.
3. The messages,responses and entry screen must not be delayed.
4. To verify that the entries are of the proper format.
5. Making sure no duplicate entries are allowed.
6. All the links should take the user to the proper page.

5.1.1 Unit Testing

Unit testing will involve the design of test cases that validate that the internal program logic is functioning correctly, and that program input produce valid outputs. Here we have tested on the local server(specifically Tomcat version 7.0) and each module was tested seperately and it works without any errors in all the cases.

5.1.2 Integration Testing

Integration testing is the incremental integration testing of two or more integrated software components on a isolated platform to show the lack of success caused by interface defects. All the modules which were tested individually in the unit testing were integrated and tested.
Test Results: All the test cases mentioned in the test objectives passed successfully. No failures were encountered.

5.1.3 System Testing

System testing fortifies that the complete integrated software system meets the requirements. It tests a configuration to fortify known and predictable outcomes. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, laying stress on pre-driven process links and integration points.

5.1.4 Acceptance Testing

User Acceptance Testing is the most important phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements. Test Results: All the test cases mentioned in the test objectives passed successfully. No failures were encountered.

5.2 Test Case Details

5.2.1 Test case 1

Unit to test: Authenticating user.

Assumptions: Website is used by the concerned user.

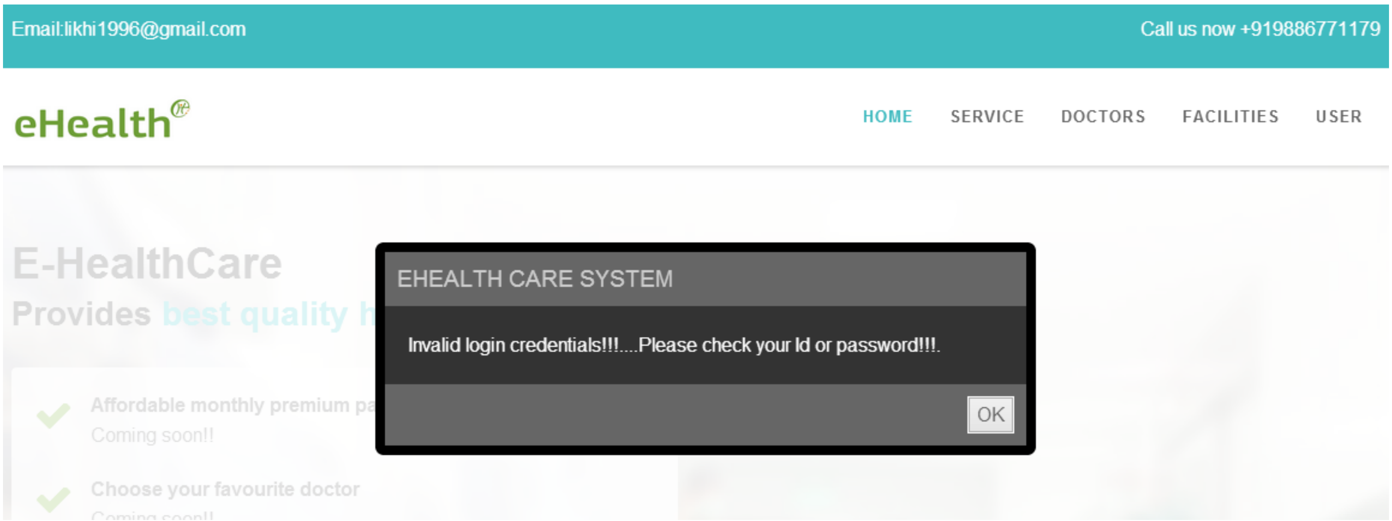
Test data: User logs.

Steps to be executed: Open the website and enter the username and password.

Expected results: If the user gives wrong password, should not link to next page.

Actual result: Doesn't redirect to next page.

Pass/Fail: Pass.



EHEALTH CARE SYSTEM

Invalid login credentials!!!....Please check your Id or password!!!.

Email:likhi1996@gmail.com
Call us now +919886771179
eHealth
HOME
SERVICE
DOCTORS
FACILITIES
USER
E-HealthCare
Provides best quality h
Affordable monthly premium pa
Coming soon!!
Choose your favourite doctor
Coming soon!!
OK

Figure 5.2: Test case1 result

5.2.2 Test case 2

Unit to test: Authenticating user

Assumptions: Website is used by the concerned user.

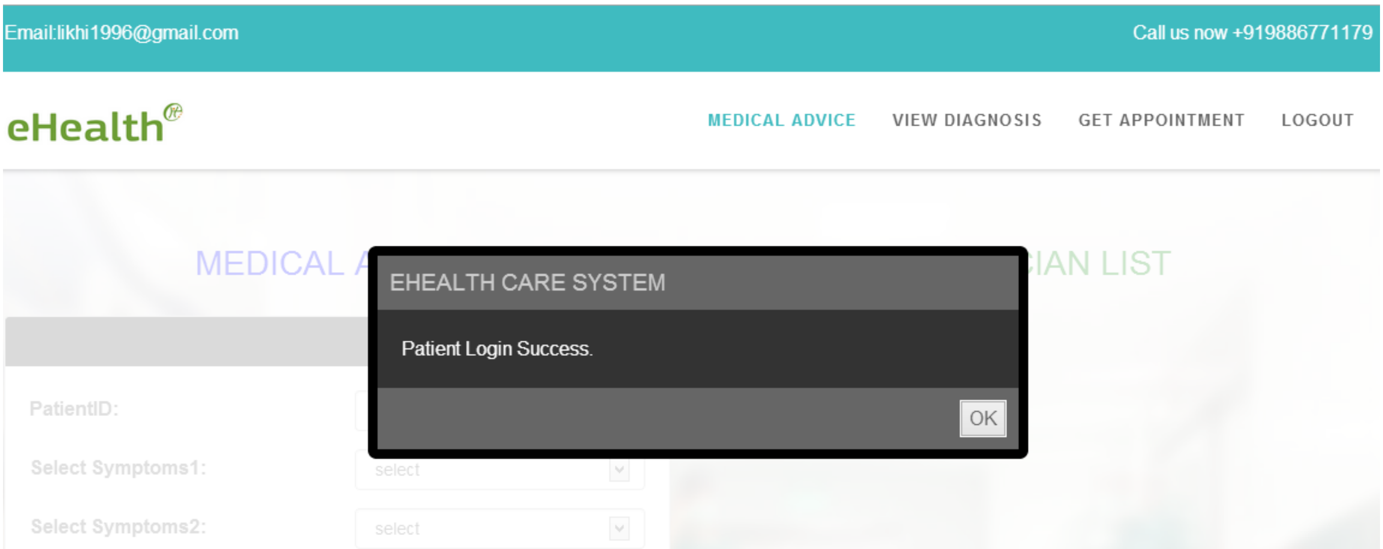
Test data: User logs

Steps to be executed: Open the website and enter the username and password.

Expected results: If the user is registered and tries to login, link to next page.

Actual result: Redirects to next page.

Pass/Fail: Pass.



Email:likhi1996@gmail.com

Call us now +919886771179

EHEALTH CARE SYSTEM

Patient Login Success.

PatientID:

Select Symptoms1:

Select Symptoms2:

Figure 5.3: Test case2 result

5.2.3 Test case 3

Unit to test: Booking appointment

Assumptions: Website is used by the concerned user.

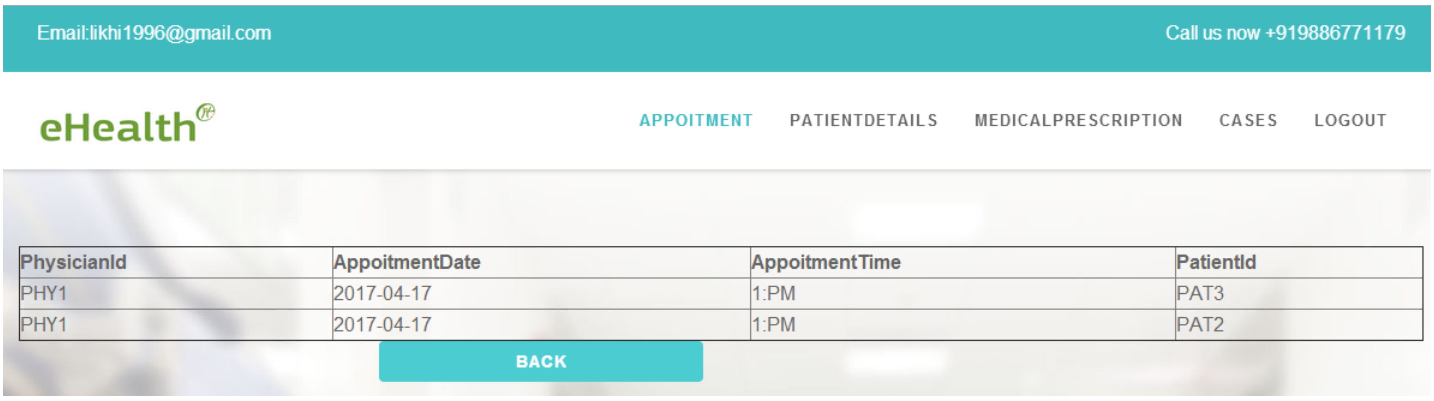
Test data: Physician Id and Appointment date.

Steps to be executed: Loging in as patient and entering the Phy Id and date before submitting.

Expected results: The data must be updated in doctors database and should be accessable.

Actual result: The Data is updated and accessable.

Pass/Fail: Pass



Email:likhi1996@gmail.com

Call us now +919886771179

PhysicianId	AppoitmentDate	AppoitmentTime	PatientId
PHY1	2017-04-17	1:PM	PAT3
PHY1	2017-04-17	1:PM	PAT2

Figure 5.4: Test case3 result

5.2.4 Test case 4

Unit to test: Checkinga a specific patient details.

Assumptions: Website is used by the concerned user.

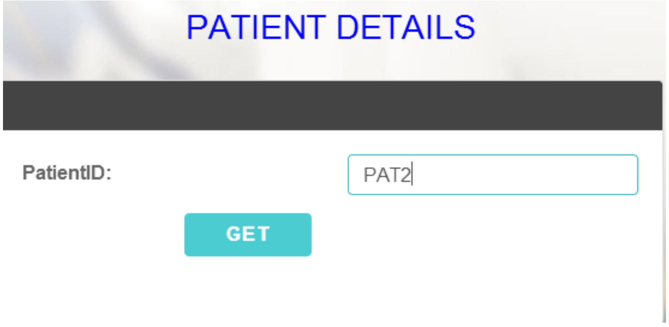
Test data: Patient Id.

Steps to be executed: Loging in as doctor and entering the start and end date.

Expected results: If the doctor gives the patient Id, display his details.

Actual result: Patient details are displayed.

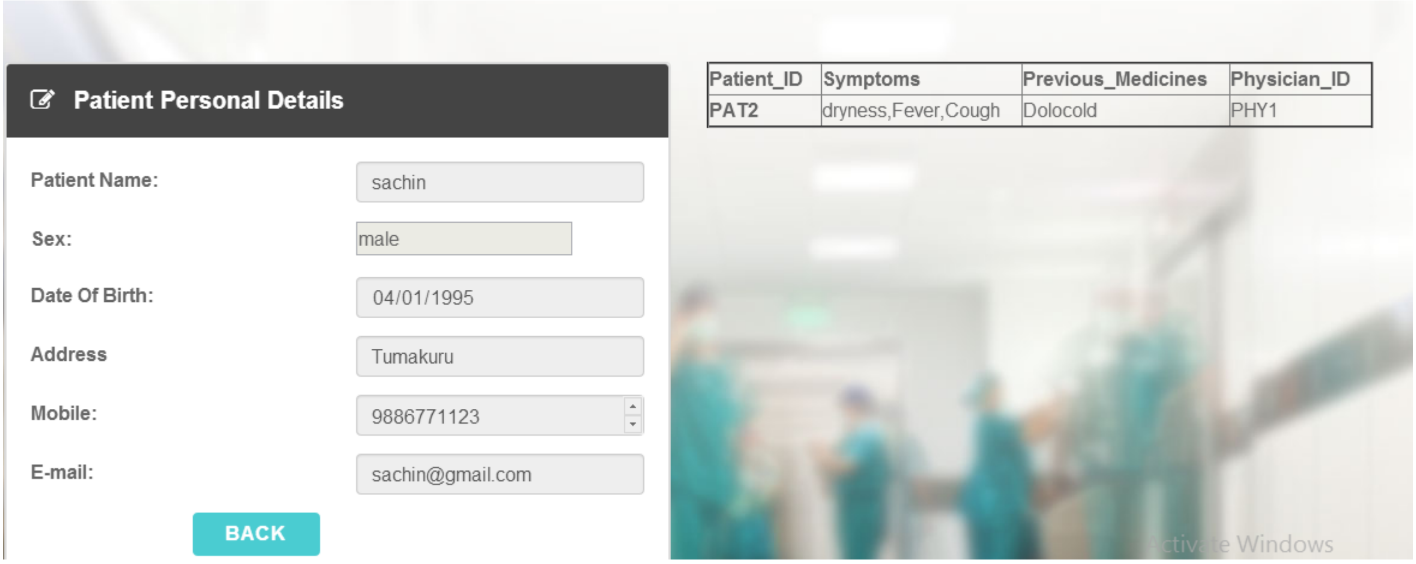
Pass/Fail: Pass.



PATIENT DETAILS

PatientID: PAT2

Figure 5.5: Test case4 result



Patient Personal Details

Patient Name: sachin

Sex: male

Date Of Birth: 04/01/1995

Address Tumakuru

Mobile: 9886771123

E-mail: sachin@gmail.com

Patient_ID	Symptoms	Previous_Medicines	Physician_ID
PAT2	dryness,Fever,Cough	Dolocold	PHY1

Figure 5.6: Test case4 result

5.2.5 Test case 5

Unit to test: When there is no appointments

Assumptions: Website is used by the concerned user.

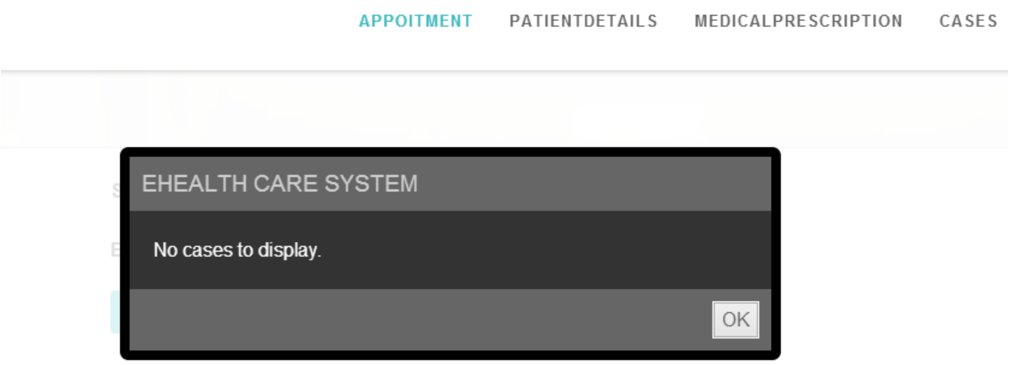
Test data: Start date and end date.

Steps to be executed: Loging in as doctor and entering the start and end date.

Expected results: If the doctor gives start and end date, display no cases

Actual result: A dialog box saying no cases is displayed.

Pass/Fail: Pass.



EHEALTH CARE SYSTEM

No cases to display.

Figure 5.7: Test case5 result

5.2.6 Test case 6

Unit to test: Submitting diagnosis details

Assumptions: Website is used by the concerned user.

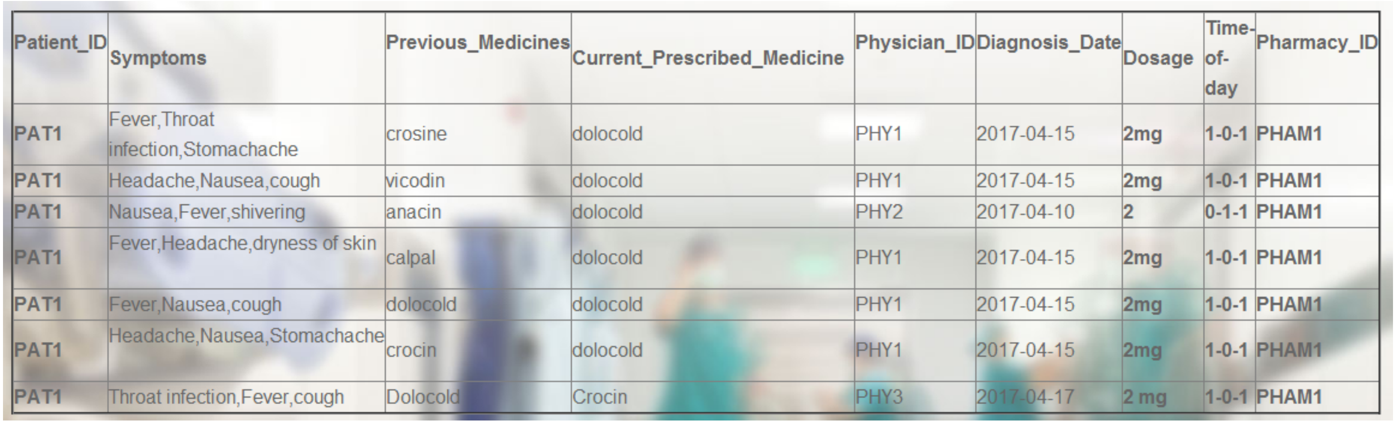
Test data: patient details, prescription etc.

Steps to be executed: Loging in as doctor and entering the diagnosis details

Expected results: The diagnosis details should be updated in patient's database and available.

Actual result: Diagnosis details are updated and available for patient.

Pass/Fail: Pass



Patient_ID	Symptoms	Previous_Medicines	Current_Prescribed_Medicine	Physician_ID	Diagnosis_Date	Dosage	Time-of-day	Pharmacy_ID
PAT1	Fever,Throat infection,Stomachache	crosine	dolocold	PHY1	2017-04-15	2mg	1-0-1	PHAM1
PAT1	Headache,Nausea,cough	vicodin	dolocold	PHY1	2017-04-15	2mg	1-0-1	PHAM1
PAT1	Nausea,Fever,shivering	anacin	dolocold	PHY2	2017-04-10	2	0-1-1	PHAM1
PAT1	Fever,Headache,dryness of skin	calpal	dolocold	PHY1	2017-04-15	2mg	1-0-1	PHAM1
PAT1	Fever,Nausea,cough	dolocold	dolocold	PHY1	2017-04-15	2mg	1-0-1	PHAM1
PAT1	Headache,Nausea,Stomachache	crocin	dolocold	PHY1	2017-04-15	2mg	1-0-1	PHAM1
PAT1	Throat infection,Fever,cough	Dolocold	Crocin	PHY3	2017-04-17	2 mg	1-0-1	PHAM1

Figure 5.8: Test case6 result

5.2.7 Test case 7

Unit to test: Checking Patient's medicine record details

Assumptions: Website is used by the concerned user.

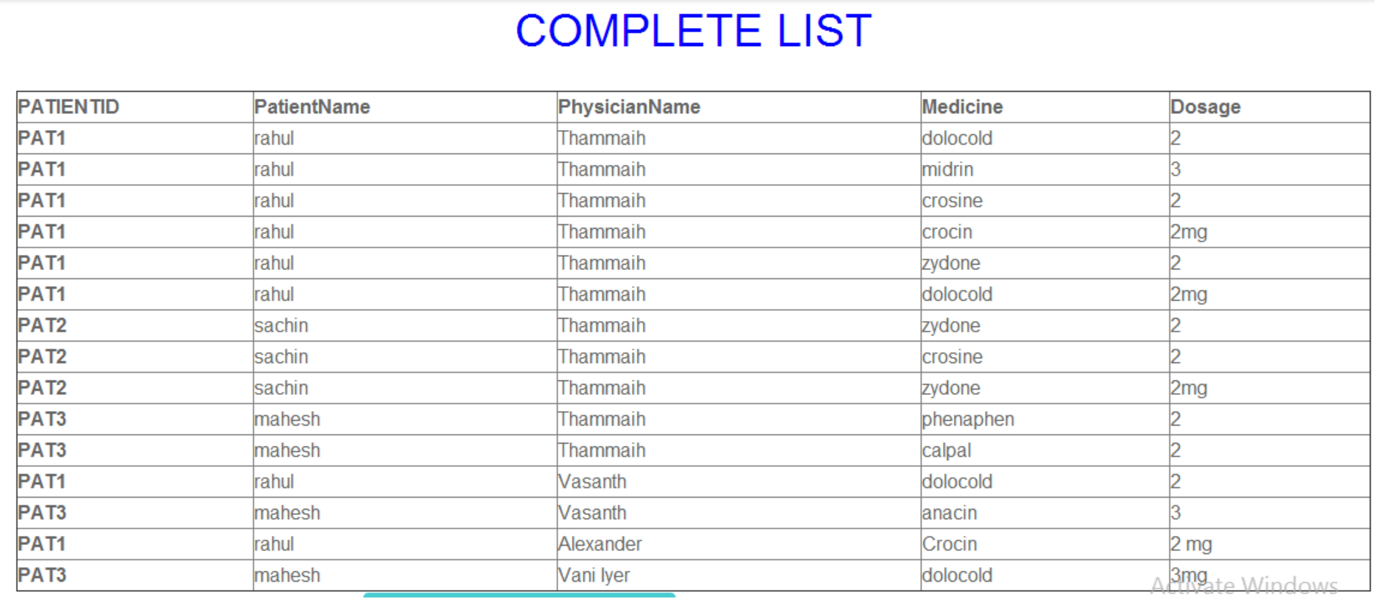
Test data: Start date and end date

Steps to be executed: Loging in as pharmacist and entering the start and end date.

Expected results: If the Pharmacist gives start and end date, display medicine record details.

Actual result: Medicine details are displayed

Pass/Fail: Pass.



COMPLETE LIST

PATIENTID	PatientName	PhysicianName	Medicine	Dosage
PAT1	rahul	Thammaih	dolocold	2
PAT1	rahul	Thammaih	midrin	3
PAT1	rahul	Thammaih	crosine	2
PAT1	rahul	Thammaih	crocin	2mg
PAT1	rahul	Thammaih	zydone	2
PAT1	rahul	Thammaih	dolocold	2mg
PAT2	sachin	Thammaih	zydone	2
PAT2	sachin	Thammaih	crosine	2
PAT2	sachin	Thammaih	zydone	2mg
PAT3	mahesh	Thammaih	phenaphen	2
PAT3	mahesh	Thammaih	calpal	2
PAT1	rahul	Vasanth	dolocold	2
PAT3	mahesh	Vasanth	anacin	3
PAT1	rahul	Alexander	Crocin	2 mg
PAT3	mahesh	Vani Iyer	dolocold	3mg

Figure 5.9: Test case7 result

5.2.8 Test case 8

Unit to test: Checking a Patient's medicine record details on a particular date

Assumptions: Website is used by the concerned user.

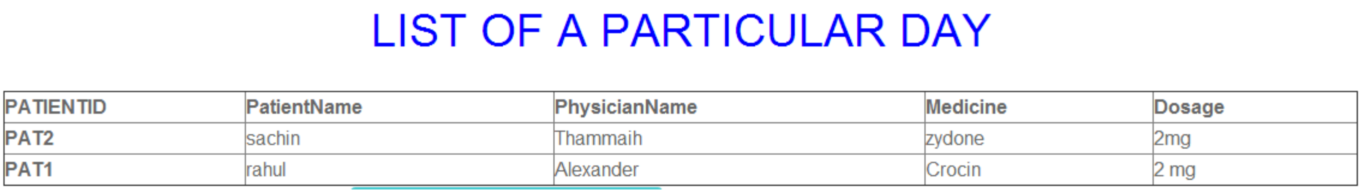
Test data: date

Steps to be executed: Loging in as pharmacist and entering specific date.

Expected results: If the Pharmacist gives the date, display medicine record details of that date.

Actual result: Medicine details of that date are displayed

Pass/Fail: Pass.



LIST OF A PARTICULAR DAY

PATIENTID	PatientName	PhysicianName	Medicine	Dosage
PAT2	sachin	Thammaih	zydone	2mg
PAT1	rahul	Alexander	Crocin	2 mg

Figure 5.10: Test case8 result

Chapter 6

Conclusions and Future Scope

Smart healthcare is a platform where information has to be stored and maintained correctly. This domain needs to create a user-friendly system, which advices users at all steps they need to carry out in it. The data provided by the users must be kept private, as the healthcare details is very much intimate. The prescriptions for a particular patient are forwarded electronically to the pharmacy. This prevents the unnecessary time taken by the patient to carry the prescription to the pharmacict in the pharmacy store.

Thus our healthcare website is much secured in giving authentication to the user. Our project advices the user to the action they need to perform. Our project is user-friendlier than all other healthcare systems.

Our website currently concentrates on the relationships between patients, physicians and pharmacists. I plan to extend this web application to other healthcare facilities and professionals, like laboratory technicians who perform and report the tests and analyses requested by physicians. We also plan to scrutinize whether our Clinic and Pharmacy modules can be interfaced to applications supplied by pharmaceutical companies that provide data on medications and dosages. In addition, we plan to investigate drug delivery devices, such as e-pillboxes, that remind and control the uniform and timely consumption of medications.

Chapter 7

Bibliography

7.1 List of Book references

[1] A Distributed e-Healthcare System Based on the Service Oriented Architecture Kart, F. Gengxin Miao Moser, L.E. Melliar-Smith, P.M.Dept. of Electr. and Comput. Eng., Univ. of California, Santa Barbara, CA;

[2] Towards a flexible, process-oriented IT architecture for an integrated healthcare network-Nicosia, Cyprus.

[3] A Service Oriented Architecture for a Health Research Data Network Rohan Baxter and Ross Sparks and Uma Srinivasan and Mark Cameron and Laurent Lefort.

[4] Unconfined E-healthcare system using UMTS-WLAN H. Qu, Q. Cheng, and E. Yaprak.

[5] Understanding and Classifying Requirements for Computer-Aided Healthcare workflows Xiping Song Hwong, B. Matos, G. Rudorfer, A. Siemens Corp. Res. Inc., Princeton;

[6] E-healthcare system design: a consumer preference approach Zhiping Walter and Y. Alex Tung

7.2 List of Web references

1. www.java.sun.com/developer/technicalArticles/WebServices/soa2/

2. www.research.ibm.com/journal/sj/444/niblett.html

3. www.sap.com/platform/soa/customers/index.epx

4. www.webreference.com/programming/soa/

5. www.tutorialspoint.com

6. www.W3schools.com