

University of Pune



*A
Project Report*

On

"E-HEALTH CARE FACILITIES"

Submitted by

**Ms. AANCHAL OSWAL
Mr. ROHIT PITRE
Mr. MUSTAFA BADSHAH
Ms. VACHANA SHETTY**

Under the Guidance of

Prof.(Mrs.) MANALI VASHI Mr. SAURABH VAIDYA



K J's Educational Institutes

K J College of Engineering and Management Research

Department of Computer Engineering

2014-2015

K J's Educational Institutes
K J College of Engineering and Management Research
Department of Computer Engineering



CERTIFICATE

This is to certify that Ms. AANCHAL OSWAL, Mr. ROHIT PITRE, Mr. MUSTAFA BADSHAH and Ms. VACHANA SHETTY, of K J College of Engineering and Management Research has submitted the **Project Report** entitled

”E-HEALTH CARE FACILITIES”

They have satisfactorily completed and submitted **Project Report** as prescribed by **University of Pune** for Fourth Year Engineering (Computer Engineering) for the Academic Year 2013-2014.

Place : Pune

Date :

Prof. S.M.Patil
[Project Co-ordinator]

Prof.(Mrs.) Manali Vashi
[Project Guide]

Mr. Saurabh Vaidya
[External Guide]

Prof. Deepak C. Mehetre
[Head of Department]

Dr. Sanjeev J. Wagh
[Principal]

[External Examiner]

Date :

Acknowledgment

First and foremost, we would like to thank our guides, **Prof.(Mrs.) Manali Vashi** and **Mr. Saurabh Vaidya**, for their guidance and support. We will forever remain grateful for the constant support and guidance extended by our guides, in making this report. Through our many discussions, they helped us to form and solidify ideas. The invaluable discussions we had with them, the penetrating questions they put to us and the constant motivation, has all led to the development of this project.

We are also thankful to our family members for encouragement and support.

We wish to express our sincere thanks to the Head of department, **Prof. Deepak C. Mehetre**, as well as our principal **Dr. Sanjeev J. Wagh**, also grateful thanks to our departmental staff members for their support.

We would also like to thank to our friends for listening to our ideas and providing feedback and suggestions for improving those ideas.

Ms. Aanchal Oswal
Mr. Rohit Pitre
Mr. Mustafa Badshah
Ms.Vachana Shetty

K J College of Engineering and Management Research, Pune
May 2015

Abstract

The problems we face these days with our convention systems are missing on the doctor due to lack of an appointment, hampering our work when we lose or damage important medical records, missing out on a crucial dose of our medicine while we are busy in our work, etc. These are few of the major problems each of us face in our hectic and busy life.

This project deals with the development and implementation of a smart-phone application that lets you make a pre-appointment with your doctor, digitally stores all our medical records on a secure cloud server and gives our loved ones peace of mind by reminding us to take our medicine on time. Users can easily make appointments with a doctor of their preference which is suitable to their convenience of time and date. As well as, users can input the symptoms that he/she is suffering from, and will be able to get an idea of the disease what he/she is suffering from. Medical records of the specific user will be uploaded by the doctor or hospital which can be easily accessed and downloaded by the user whenever needed. Using our phones facilities the application can enable reminders to take the specific medicine at the specific time according to our medical prescription which is also uploaded by our doctors. Additional features of calling emergency services, giving location based details of hospitals and medical care facilities and general tips for health care are also incorporated in the application.

Contents

1	Introduction	1
1.1	Overview	1
1.2	Brief Description	2
1.3	Problem Definition, Project Objectives, Goal	3
1.4	Applying software engineering approach	3
2	Literature Survey	5
2.1	Data mining for disease diagnosis	6
2.1.1	Diabetes	6
2.1.2	Heart disease	6
2.1.3	Cancer	6
2.1.4	Multiple diseases	6
2.2	Feasibility Study	7
2.2.1	Economical Feasibility	7
2.2.2	Legal Feasibility	7
2.2.3	Time Feasibility	7
2.2.4	S/W and H/W Availability	7
3	Project Plan	9
3.1	Project Plan, Schedule and Team Structure	9
4	Software Requirements Specification	11
4.1	Introduction	11
4.1.1	Purpose	11
4.1.2	Project Scope	12
4.1.3	Product Features	12
4.1.4	Use Classes and Characteristics	13
4.1.5	Operating Environment	16
4.1.6	Design and Implementation Constraints	18
4.1.7	Assumptions and Dependencies	18
4.2	System Features	19
4.2.1	System Feature	19
4.3	External Interface Requirements	21

4.3.1	Hardware Interfaces	21
4.3.2	Software Interfaces	21
4.3.3	Communication Interfaces	22
4.4	Non-functional Requirements	23
4.4.1	Performance and Scalability Requirements	23
4.4.2	Safety Requirements	23
4.4.3	Software Quality Attributes	23
4.5	Other Requirements	25
4.5.1	System Hardware Requirements	25
4.5.2	System Software Requirements	25
4.6	Performance Expectations	26
4.7	User and human factors	26
4.8	Physical Environment	26
4.9	Interface Requirements	26
4.10	Security Requirements	26
4.11	Quality Requirements	27
4.11.1	Application Quality (Non-Functional) Requirements	27
4.11.2	Availability	27
4.12	System Analysis Models	28
5	System Design	32
5.1	System Architecture and Business logic	32
6	Technical Specifications	36
6.1	Technology details to be used in the project	36
6.2	References to Technology	36
7	Conclusion	37
8	References	38
8.1	Papers	38
8.2	Websites	38
	Appendices	39
	Appendix A Mathematical Models	39
A.1	Mathematical Model	39
A.1.1	Set Theory	39
A.2	Functional Decomposition of the system	39
A.2.1	Venn Diagram	40
A.2.2	Functional Mapping-1	40
A.2.3	Functional Mapping-2	41

Appendix B Testing/Reliability of Design	42
B.1 Test Plan for Design	42
B.2 Details of Design Testing	43
Appendix C Project Management and Project Status	45
C.1 Project Management Approach	45
C.2 Status of Project Work done	45

List of Figures

1.1	<i>Waterfall Model</i>	4
4.1	<i>Class Diagram</i>	14
4.2	<i>Use Case Diagram</i>	15
4.3	<i>Operating Environment 1</i>	16
4.4	<i>Operating Environment 2</i>	17
4.5	<i>Data Flow Diagram - Level 0</i>	28
4.6	<i>Data Flow Diagram - Level 1</i>	29
4.7	<i>Entity-Relationship Diagram</i>	30
4.8	<i>State Transition Diagram</i>	31
5.1	<i>System architecture</i>	32
5.2	<i>Interaction Diagram (Sequence Diagram)</i>	33
5.3	<i>Package Diagram</i>	34
5.4	<i>Component Diagram</i>	35

List of Tables

2.1	Comparison of disease diagnosis algorithms	8
3.1	Project Plan and Schedule	9
3.2	Task Sheet	10
4.1	System Feature: View Hospital Details	19
4.2	System Feature: Take Appointment	19
4.3	System Feature: View Medical History	20
4.4	System Feature: View Prescription	20
4.5	System Feature: Alert/Reminder	20
C.1	Project Status	45

Chapter 1

Introduction

The healthcare environment is information rich but knowledge poor. Our project deals with the development and implementation of a smart-phone application for electronic health care facilities, which is simple and easy to use. A user (patient) can get a brief idea about what disease is he/she suffering from, just by inputting the symptoms using our smart-phone application. Even users can make a pre-appointment with your doctor, can digitally store all our medical records on a secure cloud server, reminding them to take medicine on time. Users can easily make appointments with a doctor of their preference which is suitable to their convenience of time and date.

1.1 Overview

We propose android mobile application E-health care facilities for various things, such as taking pre-appointment with the doctors of users' choice. User will enter out the symptoms which he is suffering from, and using Data Mining techniques, he will come to know that possibly which disease he has caught. Also there is facility of storing the patient records digitally, so that no tension of losing the paper type records. This will lead to reduce the use of paper; ultimately will lead to save the trees. There is reminder facility for users or patients to take their medicines on time, so that no missing of medicine will cause any fatal problems to the patients. Again there are chances to extend the system, which will provide emergency service for accidental cases, by locating nearest hospitals from the place where incidence is done, even more, nearest hospitals will come to know where accident has occurred and they can provide possible help as soon as possible.

1.2 Brief Description

Basically there are some existing systems which remind us to do anything like in our mobile phones, there is facility to add reminder for any type of work. Our smart-phone android application has somewhat same facility as it reminds patients to take medicines on time, as per the schedule given by respective doctor. No patient is required to add any reminder as like we do in our regular mobile systems, automatic reminding is possible in our system. Again, it will remind the user (patient) for doctors next appointment if any there.

Basically there are some existing systems which remind us to do anything like in our mobile phones, there is facility to add reminder for any type of work. Our smart-phone android application has somewhat same facility as it reminds patients to take medicines on time, as per the schedule given by respective doctor. No patient is required to add any reminder as like we do in our regular mobile systems, automatic reminding is possible in our system. At the backend, it will read the digital data or record of patient uploaded by doctor, to add reminders. Again, it will remind the user (patient) for doctors next appointment if any there.

As per the doctors time available, this application will help user (patient) to get appointed to any particular doctor according to users choice. It will use some scheduling algorithms for giving appointments to users; so that it will be convenient for users as well as for doctors. With this android smart-phone application, user is allowed to enter the symptoms what he/she is suffering from, and according to inputted symptoms, the disease what he/she has will be predicted as a result. Though this predicted result will be the actual one, but that will be closer to the actual disease. This facility will help the doctors to predict the actual disease in one go only. For prediction of disease, data mining techniques will be used, as the dataset for this particular function is very vast. Using data mining techniques

Another facility of this application may extend to some emergency cases. In case of accidents, user will be able to locate the nearest hospital or after giving indication by user, nearest hospital will come to know about accident, so that user in hazardous conditions will not be helpless. This facility will be using GPS positioning system for finding the nearest hospital from the place of accident.

1.3 Problem Definition, Project Objectives, Goal

Our Project aims:

1. To develop an android based hospital management system to deal with disease prediction, pre-appointments, reminding and scheduling system.
2. To ease the process of taking appointment.
3. Provide an easy interface to users.
4. Store information securely in data base.
5. To maintain distinct the accounts which can be used by multiple users and would be independent of the devices. The user can log in through any mobile device having the app installed.

1.4 Applying software engineering approach

We are using waterfall model in project. Its details are as follows:

- This is the most crucial phase for the whole project here project team along with the customer makes a detailed list of user requirements. The project team chalks out the functionality and limitations of the software they are developing, in detail.
 - The document which contains all this information is called SRS, and it clearly and unambiguously indicates the requirements. A small amount of top-level analysis and design is also documented. This document is verified and endorsed by the customer before starting the project. SRS serves as the input for further phases.
1. **System Design and Software Design:** Using SRS as input, system design is done. System design included designing of software and hardware i.e. functionality of hardware and software is separated-out. After separation design of software modules is done. The design process translates requirements into representation of the software that can be assessed for quality before generation of code begins. At the same time test plan is prepared, test plan describes the various tests which will be carried out on the system after completion of development.
 2. **Implementation and Unit Testing:** Now that we have system design, code generation begins. Code generation is conversion of design into machine-readable form. If designing of software and system is

done well, code generation can be done easily. Software modules are now further divided into units. A unit is a logically separable part of the software. Testing of units can be done separately. In this phase unit testing is done by the developer itself, to ensure that there are no defects.

3. **Integration and System testing:** Now the units of the software are integrated together and a system is built. So we have complete software at hand which is tested to check if it meets the functional and performance requirements of the customer. Testing is done, as per the steps defined in the test plan, to ensure defined input produces actual results which agree with the required results. A test report is generated which contains test results.
4. **Operation and Maintenance:** Now that we have completed the tested software, we deliver it to the client. His feed-backs are taken and any changes, if required, are made in this phase. This phase goes on till the software is retired

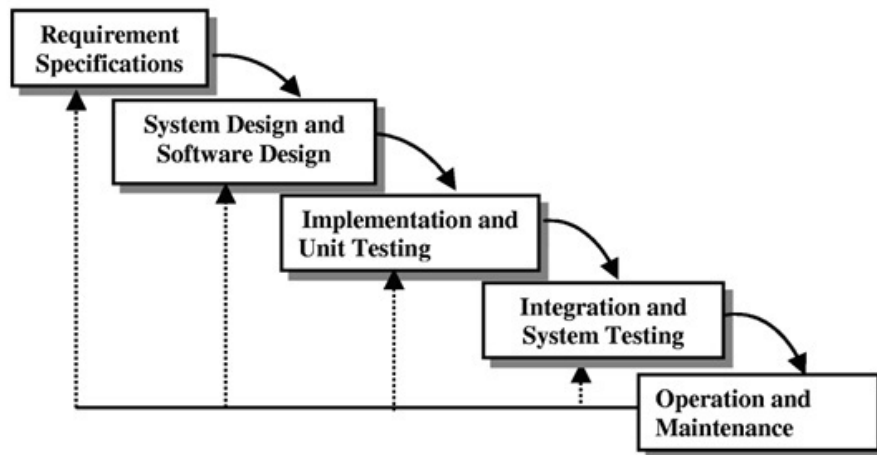


Figure 1.1: *Waterfall Model*

Chapter 2

Literature Survey

Medical data mining has been a great potential for exploring hidden patterns in data sets of medical domain. In healthcare, though data mining is not widely used but it has become increasingly popular. Data mining can improve decision-making by discovering patterns and trends in large amounts of complex data. There are two primary goals of data mining-prediction and description. Prediction involves some variables or fields in the data set to predict unknown or future values of other variables of interest. On the other hand Description focuses on finding patterns describing the data that can be interpreted by humans.

With the rapid increase in population the number of diseases occurring is also increasing. Many diseases have closely related symptoms which make it difficult for the doctors to predict the exact disease in one go. This is where data mining comes into assistance; it helps in predicting the disease which is nearly accurate. Even though the prediction is not very accurate it at least gives the doctor a brief idea what the disease could be. Thus, in no way disease prediction using data mining is a substitute to doctors whereas; its a compliment to the doctors.

We have done a survey on disease diagnosis research, there are two types of systems are available for this purpose-

- Domain specific which is for doing diagnosis of particular diseases such as system involved with diagnosis of heart diseases only it wont identify any other diseases.
- Second types of system which concentrate on multiple diseases diagnosis.

2.1 Data mining for disease diagnosis

2.1.1 Diabetes

In [1], there is prediction of diabetes using amalgam K-NN with K-means. The inconsistent data from Pima Indian Diabetes Database (PIDD) is identified and corrected using K-means clustering algorithm thereafter K-NN classification algorithm is used to classify the data.

2.1.2 Heart disease

Diagnosis of heart disease is a significant and tedious task in medicine. [3] gives us a simple technique to predict risk of heart attacks. The data classification is based on MAFIA algorithms which result in accuracy. C4.5 algorithm is used as the training algorithm to show rank of heart attack with the decision tree. Finally, the heart disease database is clustered using the K-means clustering algorithm. The results showed that the system is capable of predicting the heart attack successfully.

2.1.3 Cancer

In paper [5] the author has used J48 classification algorithm for detecting breast cancer using a two level diagnosis. In the first level the diagnosis is done on the basis of Wisconsin Breast Cancer Dataset (WBCD); the result obtained from the WBCD is classified into malignant and benign classes. At the second level diagnosis is based on the pathological and physiological parameters of malignant breast cancer dataset and then classified into various types.

2.1.4 Multiple diseases

In [7], for a small set, iterative search is applied, and if the larger data set is there, K-nn algorithm is applied. As complexity of k-nn algorithm is more for Euclidean distance method, LAMSTAR algorithm is used for assigning weights to the elements in dataset to reduce K-nn algorithm complexity.

2.2 Feasibility Study

The feasibility study is major factor which contributes to analysis of system. In earlier stages of S/W development, it is necessary to check whether system is feasible or not. There are 4 aspects of checking feasibility. Detail study was carried out to check workability of proposed system, so the feasibility study is system proposal regarding to its workability, impact on organization, ability to meet user requirements and effective use of resources thus when application progresses, it normally goes through a feasibility study and risk analysis.

Four key considerations are involved in feasibility analysis :

1. Economic Feasibility
2. Legal Feasibility
3. Time Feasibility
4. S/W and H/W Availability

2.2.1 Economical Feasibility

System has been developed using resource S/W. Cost involved in deploying system, such that there should be private network is required. Benefit of this product will definitely outweigh its cost once system has been setup, we will get the reliability function.

2.2.2 Legal Feasibility

We shall be using licensed version of various development tools and software required. Thus this project is legally feasible.

2.2.3 Time Feasibility

There is a lot of learning curve to all the components involved in this project. The project mainly deals with software. The time that will be required to do this project is decided to be 6 months including the planning (taken place). The project is timed to be deployable in the last week of March. The total project is feasible in terms of time.

2.2.4 S/W and H/W Availability

The project requires Microsoft Windows based operating system and some software tools that are easily available in the market for sale and it's possible to request it from the college as well.

Paper	Author names	Algorithms	Limitations	Various measures used
An amalgam KNN to predict Diabetes Mellitus	Nirmala Devi.M, Appavu Bal-amurugan, Swathi U.V	Amalgam of K-means and K-NN	It is lazy learner. There is no thumb rule to determine value of parameter k (Number of nearest neighbours).	Accuracy, sensitivity and specificity
Decision Tree Discovery for the Diagnosis of Type II Diabetes	Asma A. Al-Jarullah	J48 Decision Tree Algorithm and C4.5	There are other risk factors that the data collection does not consider. Dataset contains data of only female patients.	Accuracy
Predicting the Analysis of Heart Disease Symptoms Using Medicinal Data Mining Methods	V. Manikantan and S. Latha	K-Mean based MAFIA with ID3 and C4.5	Prediction of heart attack using patient prescription is not included.	Accuracy.
Classification of Heart Disease Using K-Nearest Neighbor and Genetic Algorithm	M.Akhil jabbar, B.L Deekshatulu, Priti Chandra	KNN and genetic algorithm	(KNN+GA) was not successful for breast cancer and primary tumour. As the k value goes on increasing accuracy of data sets is decreasing.	Accuracy
Two level Diagnosis of Breast Cancer using data mining	Rajkumar Gaur Grewal, Babita Pandey	J48 classification algorithm	J48 is not feasible when larger dataset is used, as small change in dataset reflects in larger modification in decision tree.	Sensitivity.
Masses Detection Using SVM Classifier Based on Textures Analysis	Fatima Eddaoudi, Fakhita Regragui, Abdelhak Mahmoudi and Najib Lamouri	SVM classification based on Haralick vector, Algorithm developed by S.M. Kwok and R. Chandrasekhar	The displacement and the orientation used for the calculation of co-occurrence matrix significantly affect the results. The mixing of two approaches (co-occurrence and contours) just gave satisfactory results	95% of classification rate can be achieved by using pre-segmented mammograms by maxima thresholding.

Table 2.1: Comparison of disease diagnosis algorithms

Chapter 3

Project Plan

3.1 Project Plan, Schedule and Team Structure

Activity	Start Date	Due Date	Milestone	Priority
Architecture & Planning			SRS & Prototype	2
Design Phase			PSD file	1
Development Phase			First version of working app	3
Testing Phase			Test Plan & Test case document	4
Deployment Phase			Project Completion Doc	5

Table 3.1: Project Plan and Schedule

TASK	Start Date	Due Date	Priority
1) Pre-project discussion regarding modules to be developed and concepts to be implemented in planned system.			High
2) Detailed discussion on scope of the project and feasibility issues regarding project.			Medium
3) Submission of abstract and finalization of topic.			High
4) Research on previously developed system based on same technology and collection of research papers.			High
5) Collection of some more research papers and study of different technologies that can be implemented.			Medium
6) Preparation of critique on different research papers that are being referred.			Low
7) Technical research on different protocols and working of different technologies.			High
9) Block diagram and UML diagrams for the system to be developed.			High

Table 3.2: Task Sheet

Chapter 4

Software Requirements Specification

4.1 Introduction

This project deals with the development and implementation of a smart-phone application that lets you make a pre-appointment with your doctor, digitally stores all our medical records on a secure cloud server and gives our loved ones peace of mind by reminding us to take our medicine on time. Users can easily make appointments with a doctor of their preference which is suitable to their convenience of time and date. As well as, users can input the symptoms that he/she is suffering from, and will be able to get an idea of the disease what he/she is suffering from. Medical records of the specific user will be uploaded by the doctor or hospital which can be easily accessed and downloaded by the user whenever needed. Using our phones facilities the application can enable reminders to take the specific medicine at the specific time according to our medical prescription which is also uploaded by our doctors. Additional features of calling emergency services, giving location based details of hospitals and medical care facilities and general tips for health care are also incorporated in the application.

4.1.1 Purpose

The main aim of our project is:

- To provide easiest way to deal with disease prediction, pre-appointments, reminding and scheduling system.
- To maintain distinct the accounts which can be used by multiple users and would be independent of the devices. The user can log in through any mobile device having the app installed.

- To ease the process of taking appointment.
- Store information securely in data base.
- The core concept for this project is to avoid waiting in the queue to take appointment.

4.1.2 Project Scope

- Proposed application can be used by all and is user friendly.
- This application will allow the user to take appointment, cancel appointment with the doctor of his choice.
- This application will be innovative in its own aspect as it will accommodate the whole process of adding reminders for taking medicines and will minimize the dependency of user on other available options.
- The application will also consist of facility to predict the disease just by entering the symptoms.
- This app would combine a number of functionalities into one, so the user need not download a number of applications for performing different tasks.
- The future scope would be to use the GPS system to deal with some emergency services in case of accidents.

4.1.3 Product Features

- The idea of our app differs as it can be used by multiple users. For example, our app would require the user to create his account which he would access from any android device.
- The data provided by the user in this app during signup would be saved in the database and would be independent of the devices.
- Our app would work like a social networker where the user would be able to create an account and use this app from any device which supports android OS, to book the ticket.

4.1.4 Use Classes and Characteristics

1. List of actors and their details

Actors in this case are the users/patients which would interact with the system to book the tickets. Also the doctor to confirm the Appointment, update schedule, and upload patient history and the hospital just registers the user/patient.

2. Use case description

The main purpose of the use case diagram is to make the user understand the basic functionalities of the particular project. The use case describes how the user uses the application to fix an appointment with the doctor or to check his medical history, etc. For instance, the user should sign up first that is user registration should be done and that information of user will be stored in the database. After user registration, user can perform following operations like fix an appointment with the doctor, check his/her medical history, view the prescription to be taken, set a reminder for his medicines. The doctor here confirms the appointment, updates the daily schedule, uploads the patient history and uploads the prescription to be taken by the patient. With all these we are also providing the delete, update and validate operations on the server side

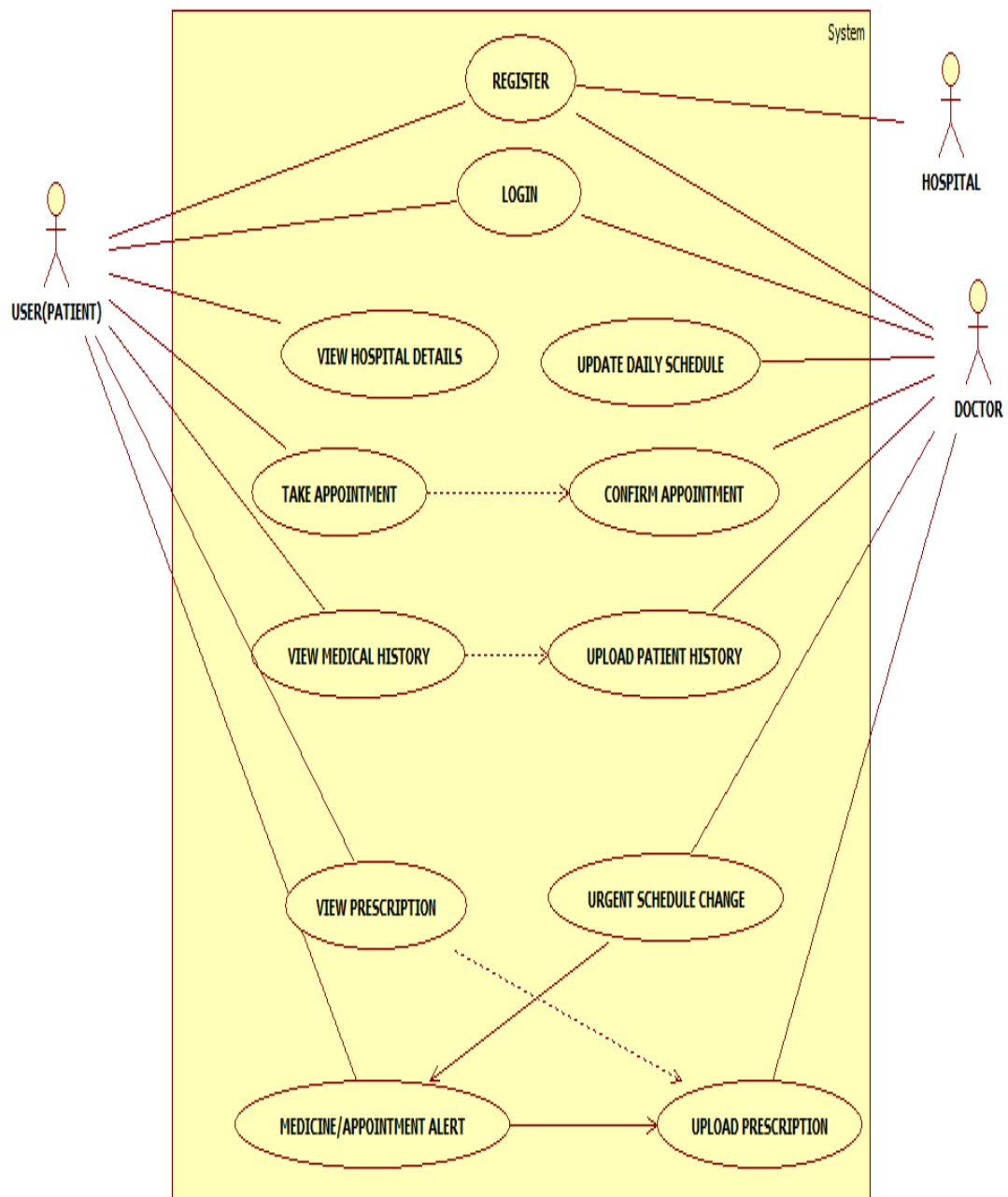
3. Preconditions

- There should be Internet connectivity available.
- The user should have android compatible smart device.
- The application should be installed by the user.
- Servers should be available at the time of request.

• Use Case Suite

1. User To-Do list management

- User registers with the system.
- User logs into the system.
- User views the hospital details.
- User takes an appointment.
- User views his medical history.
- User views the prescription.
- User can set an alert/reminder.
- User can cancel the appointment.
- User logs out from the system.

Figure 4.1: *Class Diagram*

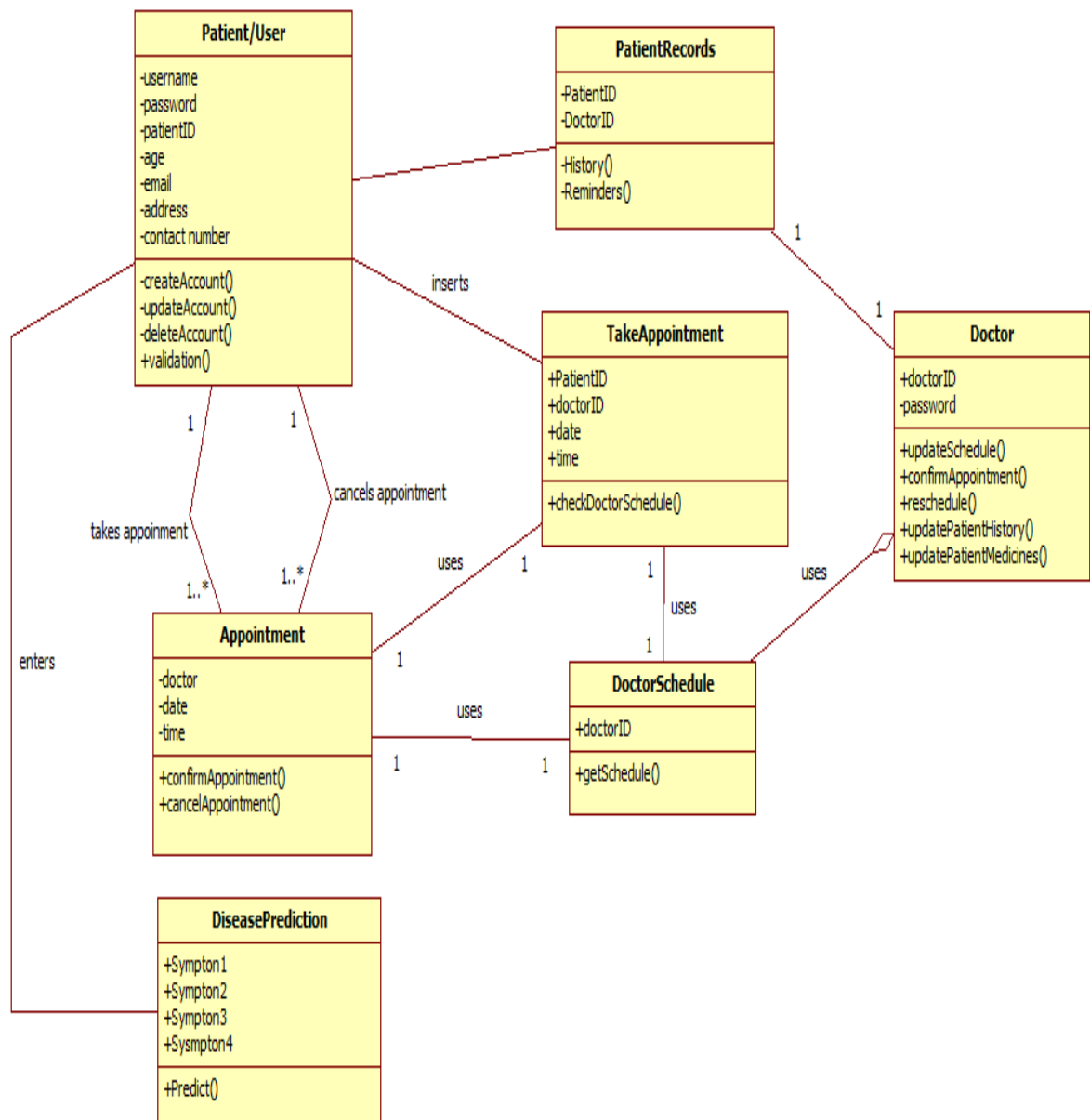


Figure 4.2: Use Case Diagram

2. Checker To-Do list management

- Registers the patient.

3. Admin To-Do list management

- Doctor confirms the appointment.
- Doctor updates daily schedule.
- Doctor uploads patient history.
- Doctor makes urgent schedule changes.
- Doctor uploads prescription.

• User Story

User enters his username and password to log in to the system. User fixes an appointment with the doctor. Doctor confirms the appointment and updates the schedule. User sets an alert of this appointment. User logs out of the system.

4.1.5 Operating Environment

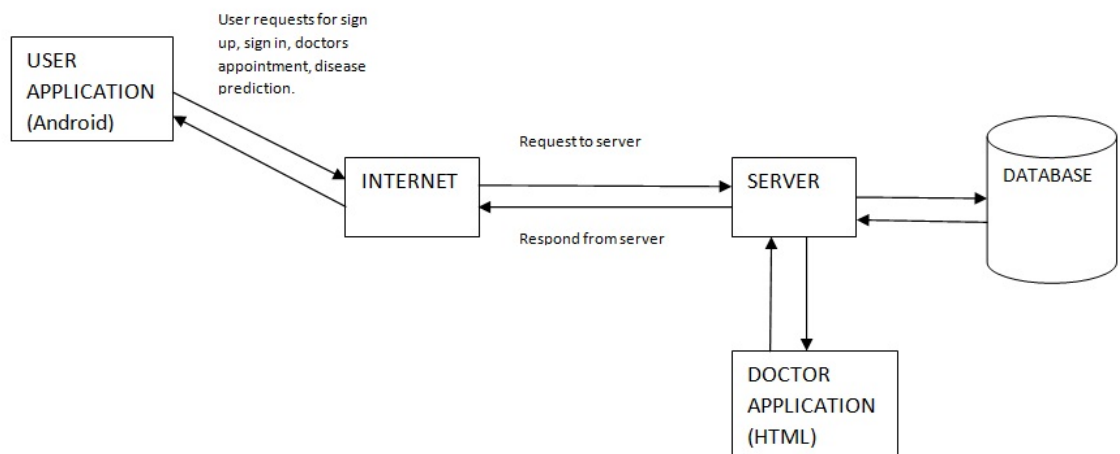
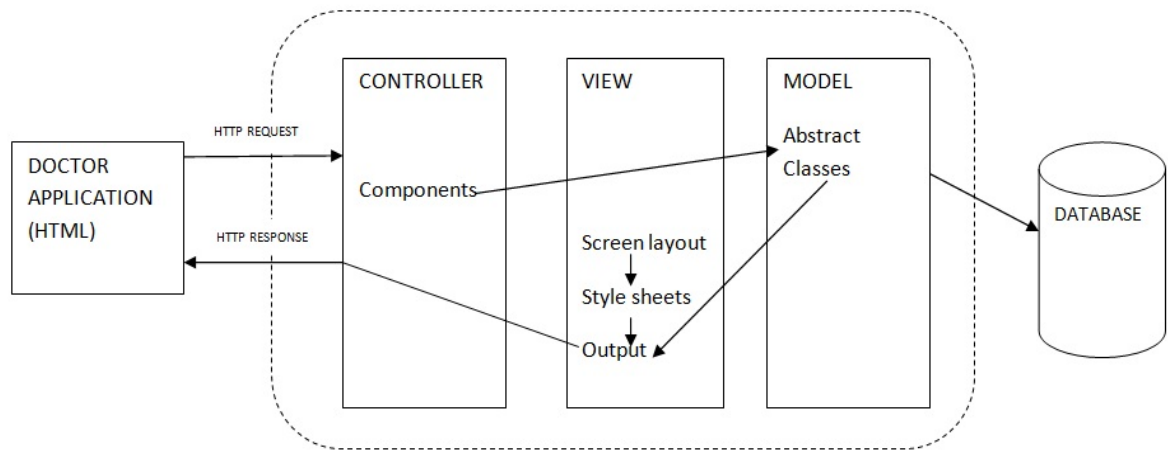


Figure 4.3: *Operating Environment 1*

Figure 4.4: *Operating Environment 2*

Step 1: The work here starts during the first time installation of our application where the user has to sign up. During sign up the patients personal and basic details need to be entered like the first name, last name, date of birth, any genetic disease, or any other health ailments. These details will be gathered and will be stored into the database. Once the user has an account he can sign in directly. Thus the user can use different android phones or the webpage to log in and will not be restricted to only his phone.

Step 2: Once the user registers, he can sign in anytime. The user can check the hospital details and perform various actions. The user can fix an appointment with the doctor of his choice by checking the doctors daily updated schedule. With this schedule he can check for the doctors availability.

Step 3: If the doctor is available, then he acknowledges the request by the user and confirms the appointment. The user gets a confirmation and can set a reminder/alert as to when the appointment is scheduled.

Step 4: The doctor can make urgent schedule changes also which will lead to cancellation or confirmation of the appointment. The doctor can also upload the prescription of the medicines to be taken by the user.

4.1.6 Design and Implementation Constraints

- Regulatory Policies: There are no regulatory policies.
- Hardware limitations: There are no hardware limitations.
- Interfaces to other applications: There shall be no interfaces.
- Parallel operations: There are no parallel operations.
- Audit Functions: There shall be no audit functions.
- Control Functions: There shall be no control functions.
- Higher order Language Functions: SQL shall be used for the database information.
- Signal handshake protocols: There are no signal handshake protocols.
- Reliability Requirements: Total number of bugs in the system shall not exceed 1
- Criticality of application: The server application shall be available 365 days.
- Safety and Security issues: The password and a valid user name are the security issues. Data protection shall be satisfied by the backup process at the server side.

4.1.7 Assumptions and Dependencies

1. The user must have the ability to use the Internet.
2. The user must be connected to the Internet to use the system.
3. The users mobile phone must be equipped with the Android Operating System.
4. The accuracy of information of the users is the responsibility of all users.

4.2 System Features

It consists of the features which includes the following:

1. View Hospital Details
2. Take Appointment
3. View Medical History
4. View Prescription
5. Alert/Reminder

4.2.1 System Feature

ID	<i>View Hospital Details</i>
Description	<i>This includes information related to the hospital and the various doctors practising in that hospital.</i>
Actors	<i>The user will be the actor.</i>
Preconditions	<i>To view the hospital details the user must be registered on the application only then he will be able to use this service.</i>
Basic Steps	<i>The user will select the view hospital details option. The database checks for the details related to the selected hospital and provide the user with the information.</i>
Exceptions	<i>Exception which could arise would be the user selects a hospital which is not available in the systems database.</i>

Table 4.1: System Feature: View Hospital Details

ID	<i>Take Appointment</i>
Description	<i>Here the user takes an appointment with the doctor of his choice after checking that doctors schedule for availability.</i>
Actors	<i>The user will be the actor.</i>
Preconditions	<i>To take an appointment, the user must be registered on the application only then he will be able to use this service. Also the doctor should be available to confirm the appointment.</i>
Basic Steps	<i>The user checks the doctors schedule, checks for availability and then takes an appointment with that doctor if he is available.</i>
Exceptions	<i>Exception occurs if the doctor hasnt updated his schedule then it would lead to clashes with the appointment timings.</i>

Table 4.2: System Feature: Take Appointment

ID	<i>View Medical History</i>
Description	<i>This includes the medical history of the patient which will be uploaded by the doctor.</i>
Actors	<i>The user will be the actor.</i>
Preconditions	<i>To view the medical history, the user must be registered on the application only then he will be able to use this service. Also the medical history has to be uploaded by the doctor.</i>
Basic Steps	<i>After logging in to the application the user can view his medical history which is uploaded by his doctor.</i>
Exceptions	<i>It wouldnt display the medical history if it is not uploaded in the database by the doctor.</i>

Table 4.3: System Feature: View Medical History

ID	<i>View Prescription</i>
Description	<i>The medicines prescribed by the users doctor are displayed here.</i>
Actors	<i>The user will be the actor.</i>
Preconditions	<i>To view the prescription the user must be registered on the application only then he will be able to use this service. Also the doctor should have prescribed some medicines to the patient.</i>
Basic Steps	<i>The user selects the view prescription option and checks what medicines the doctor has prescribed for him.</i>
Exceptions	<i>If the doctor has not prescribed any medicines for the user then it will not display anything.</i>

Table 4.4: System Feature: View Prescription

ID	<i>Alert/Reminder</i>
Description	<i>The user can set a reminder/alert as to when his next appointment is or what time he has to take his medicines.</i>
Actors	<i>The user will be the actor.</i>
Preconditions	<i>To set an alert/reminder the user must be registered on the application only then he will be able to use this service. Also either the user must have an appointment scheduled or some medicines prescribed to set an alert for the above.</i>
Basic Steps	<i>Once the users appointment is confirmed he can set a reminder to ensure that he doesnt miss it. Similarly he can set an alert for his medicine ingestion.</i>
Exceptions	<i>If the user has no appointment scheduled or no medicines prescribed he cannot use this service.</i>

Table 4.5: System Feature: Alert/Reminder

4.3 External Interface Requirements

E-health care facility system provides a group of works with interface environments. Also there will be a database which will keep all the records of the user while visiting the page.

4.3.1 Hardware Interfaces

There is no need of any hardware interface for this System.

4.3.2 Software Interfaces

1. **Two product options for viewing**

- A. Name: Android device**

Version number: Android GingerBread or later versions.

Source: Google inc.

Purpose: The operating system specified above is required as the container of the client software at the client site in order to execute the client site of User interface.

Definition of the interface: Android is a smart phone OS which provides an interface to work on any smartphone.

- B. Name: BlueStacks**

Version Number: - BlueStacks version 0.7.7.813

Source: - Opensource Application developed by Silicon Valley-based software company BlueStacks

Purpose: The software specified above enables users to use this app on desktops and computers that are not equipped with Android.

Definition of the interface: BlueStacks is a software that simulates an Android OS.

2. **Name: Apache HTTP Server**

Version Number: 2.0.5.5

Source: The Apache Software Foundation.

Purpose: In order to execute the client site part, the web server specified above is required as the provider of the client software at the server site.

Definition of the Interface: The Apache Server Project is an effort to develop and maintain an open source server for modern OS. The goal of this project is to provide a secure, efficient and extensible server that provides services in sync with the current standards.

3. Name: MySQL

Version Number: 5.0

Source: Oracle Corporation.

Purpose: Required as database server.

Definition of the interface: MySQL is the worlds most popular open source DB software. With superior speed, and ease of use, MySQL has become the preferred choice of corporate IT managers because it eliminates the major problems associated with downtime, maintenance, administration and support.

4.3.3 Communication Interfaces

The default communication protocol for data transmission between the server and the client is the Transmission Control Protocol/ Internet Protocol (TCP/IP). At the upper level, Hyper Text Transfer Protocol (HTTP, default port=80, default of Apache port=8080) will be used for communication between the web server and the client.

4.4 Non-functional Requirements

4.4.1 Performance and Scalability Requirements

The application can sustain long periods of continuous usage by one user or multiple users. The database that will be accessible by the server will hold tables, tables that will contain information about the users, detailed information about locations and fares. Authentication information like username and password needs to be stored. The database will grow as number of users increase. The scalability requirements of the system are another important issue as well as the performance requirements. The application should work efficiently with 1 thousand users approximately using the application simultaneously. The system will have ability to provide all users with efficient support, which will not be broken down.

4.4.2 Safety Requirements

Any safety problem will not take place throughout the lifecycle of the software system. Every data can be accessed and seen just after data entrance. Safety factors will be supplied through:

- Physical server security.
- Disaster recovery plan.
- Back up of data.

4.4.3 Software Quality Attributes

1. Usability

This project deals with the development and implementation of a smart-phone application for E-Health care facilities. To use this application we need to use a smart phone with android as operating system.

2. Maintainability and Upgradeability

Making changes or upgradeability in the system will not be that much difficult. By having some knowledge of programming, some features of the system might be converted to a new version. According to the needs of upgrade, system requirements might change such as change in operating system or not.

3. Supportability and Operateability

Supportability will be provided over the whole product life of the system. System will be quite easy to use but educational support will be given if needed. The application is a multi-user web based app, can

be run on every smart phone (Android) and Internet connection has to be established before using this application. The user is expected to be comfortable using android OS and have basic knowledge of English.

4. **Business Lifecycle**

The application is designed for everyone. It can be used by business professionals or students. Hence, this system is feasible for a range group of business and great number of people in any sector. Some innovations in the system may be performed and can have a greater range of business life.

4.5 Other Requirements

4.5.1 System Hardware Requirements

- Smart Phone (Android).
- A server to process all the functions having high power and multiple core processors.

4.5.2 System Software Requirements

1. Language: Java J2SE and JDK

J2SE (Java 2 Standard Edition) Java would be the required as language for development of the project. JDK is the development kit used to compile java programs.

2. IDE: Eclipse, SDK Just like visual studio provides development environment for VB and .Net, Eclipse provides an integrated development environment (IDE) for Java.

3. Database, Data Library

Serialized Objects / Serialization - Database in Java In case the project needs database this is how it is handled in java.

- First step is to use data structures like Vectors and Lists. These come under Java Collections API.
- Secondly we declare our own classes using these data structures. E.g. a class Student to hold all the student information. Now these classes need to be pre-compiled and called within Java application as libraries. This is called as a Java Class Library
- Now class objects cannot be saved to hard drive directly. We need to convert these objects to bytes so that they can be saved to hard drive. To do this we must use a concept called as Serialization. Basically it is a concept where in objects are converted to byte streams so that they can be saved to hard drive or sent via internet and vice versa. The reverse process is called as deSerialization.
- Finally to save these bytes to hard drive or to send them via network we need Java I/O.

4. GUI

AWT and SWING are used for GUI design.

5. Computing Architecture

In order to implement a architecture or a Software As A Service (SaaS architecture) we need Web Service we need to implement a web

service. GlassFish Server to host web service SOAP API to be able to call web service at client side we need to use SOAP API or even XML.

6. Rich Client Side Applications

When implementing client-server applications or even based applications, the client side applications can be implemented using architecture of java called as (JWS) Java Web Start. This allows us to create applications with rich GUIs which are also called as Rich Internet Applications (RIA). These are smarter than implementing conventional web pages.

4.6 Performance Expectations

The application can sustain long periods of continuous usage by one user or multiple users. It will load the home page quickly within 2 seconds. The database that will be accessible by the server will hold tables, tables that will contain information about the users, detailed information about locations and fares. The database will grow as number of users increase.

4.7 User and human factors

The user is expected to be comfortable using a smart phone with android OS, Internet and have basic knowledge of English

4.8 Physical Environment

The application will be a multi user application. The user can use this application once downloaded from hosted site. The Internet connection should be established before using this application.

4.9 Interface Requirements

No, we do not need an interface with any other system for it to function properly.

4.10 Security Requirements

Security breaches on these kinds of application and are of a major concern because it can involve both enterprise information and private

customer data. Security includes processes for authentication, authorization and information handling. Building security into the app from the beginning can be more effective and less disruptive in the long run. In our application, security is required for maintaining privacy and preventing illegal use so that the authentic user should be able to access his own account. This is provided by authenticating username and password. If the user enters the wrong password three times then his account will be kept on hold and he won't be able to access it for 24hrs.

4.11 Quality Requirements

4.11.1 Application Quality (Non-Functional) Requirements

The application will be user friendly so the user will be able to use it efficiently. Also it will work efficiently on all android platforms. The system is expected to sustain long periods of continuous usage by one user or multiple users. The app will provide good security to the user so that the user feels secure using this app and providing information. The database will grow as number of users increase. It will load the home page quickly within 2 seconds.

4.11.2 Availability

The application can sustain long period of continuous usage by one or multiple users.

4.12 System Analysis Models

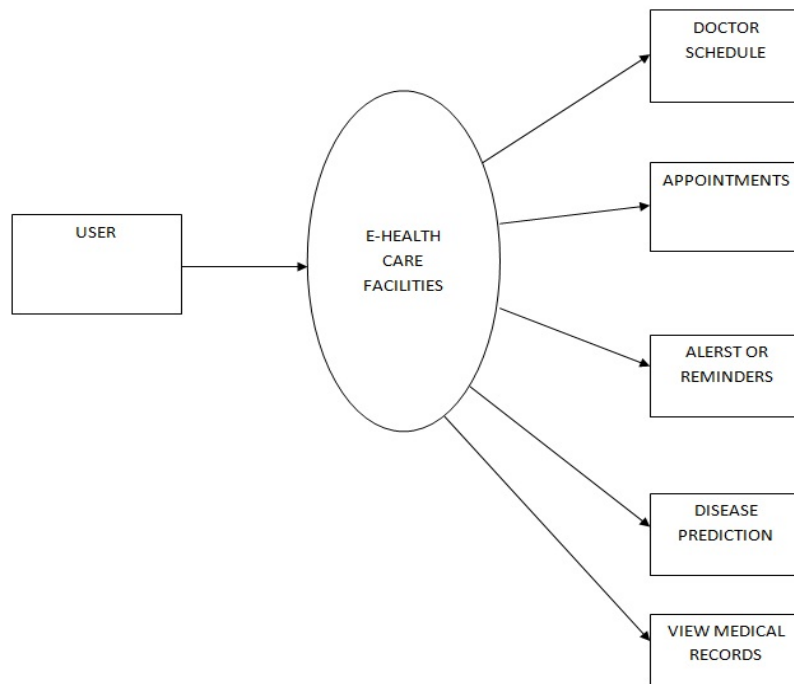
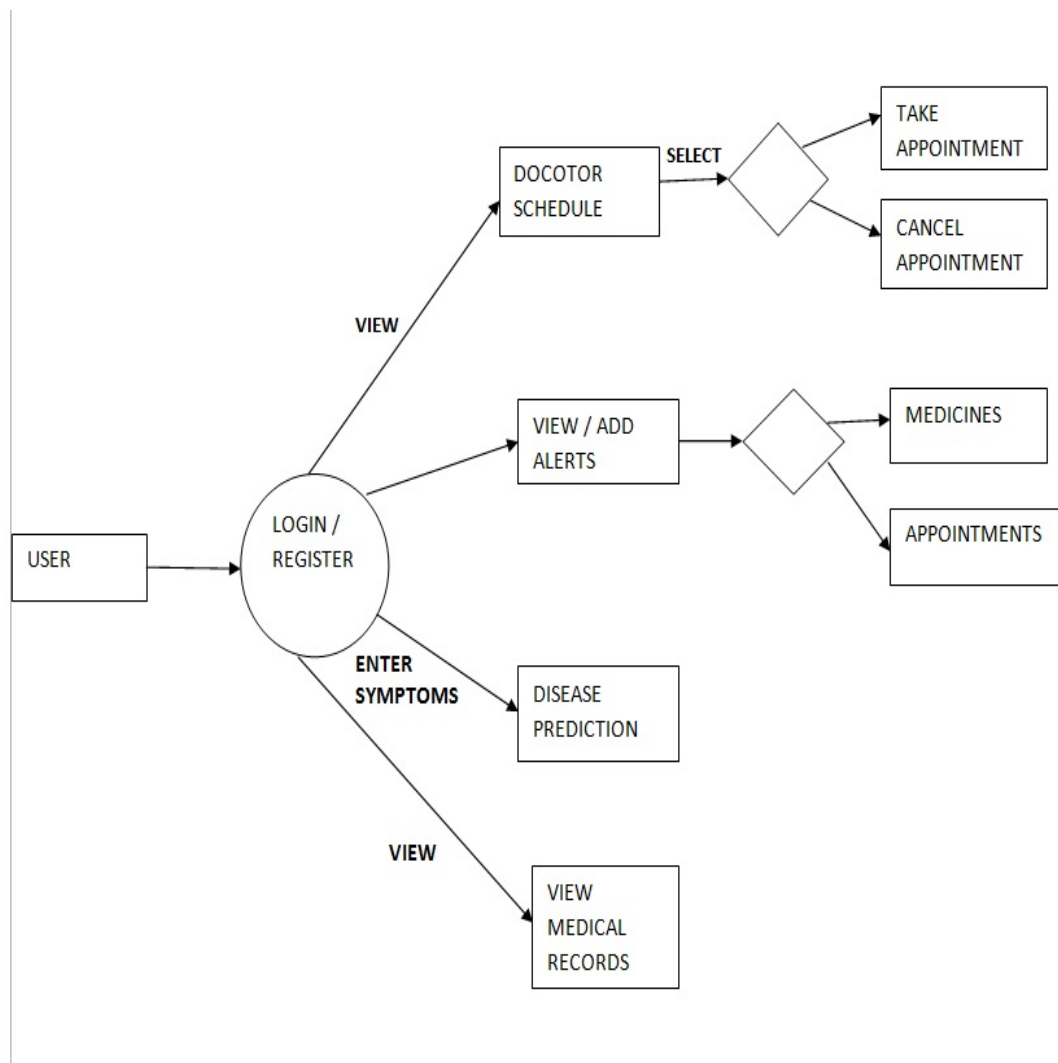
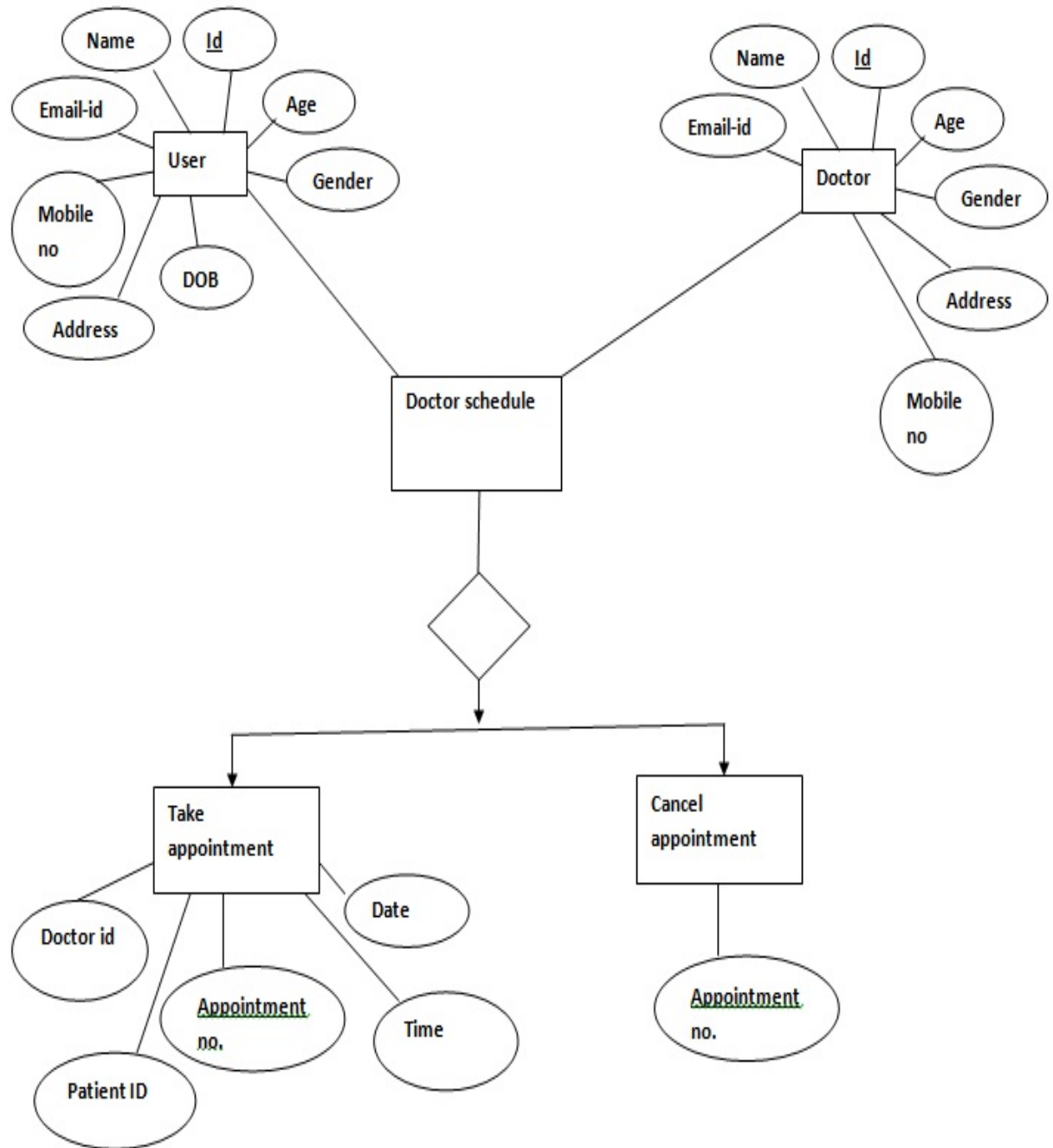
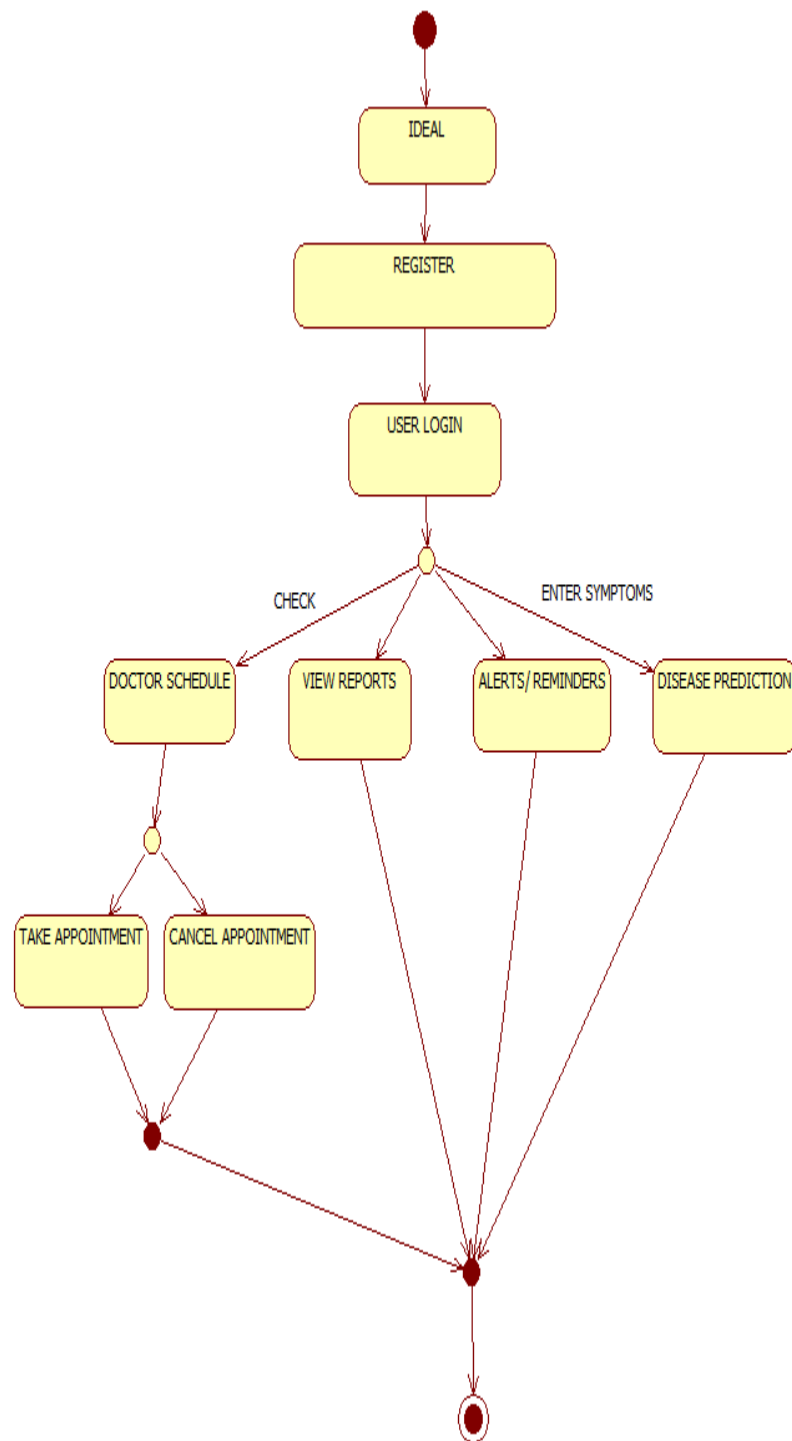


Figure 4.5: *Data Flow Diagram - Level 0*

Figure 4.6: *Data Flow Diagram - Level 1*

Figure 4.7: *Entity-Relationship Diagram*

Figure 4.8: *State Transition Diagram*

Chapter 5

System Design

5.1 System Architecture and Business logic

The structure of system divided into two components:

1. The user application which considers User requests for sign up, sign in, doctors appointment, disease prediction, personal information gathering, etc.
2. The doctor application which considers users' request for appointment, scheduling the appointments according to doctor's convenience, storing doctors' personal data along with hospital information, etc.

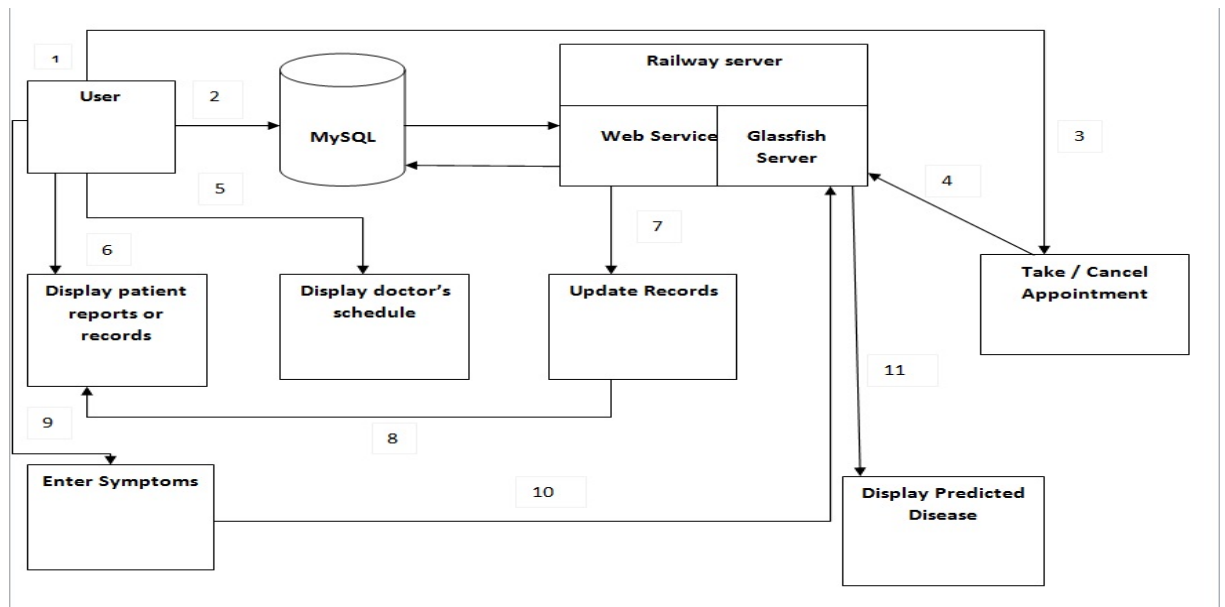


Figure 5.1: *System architecture*

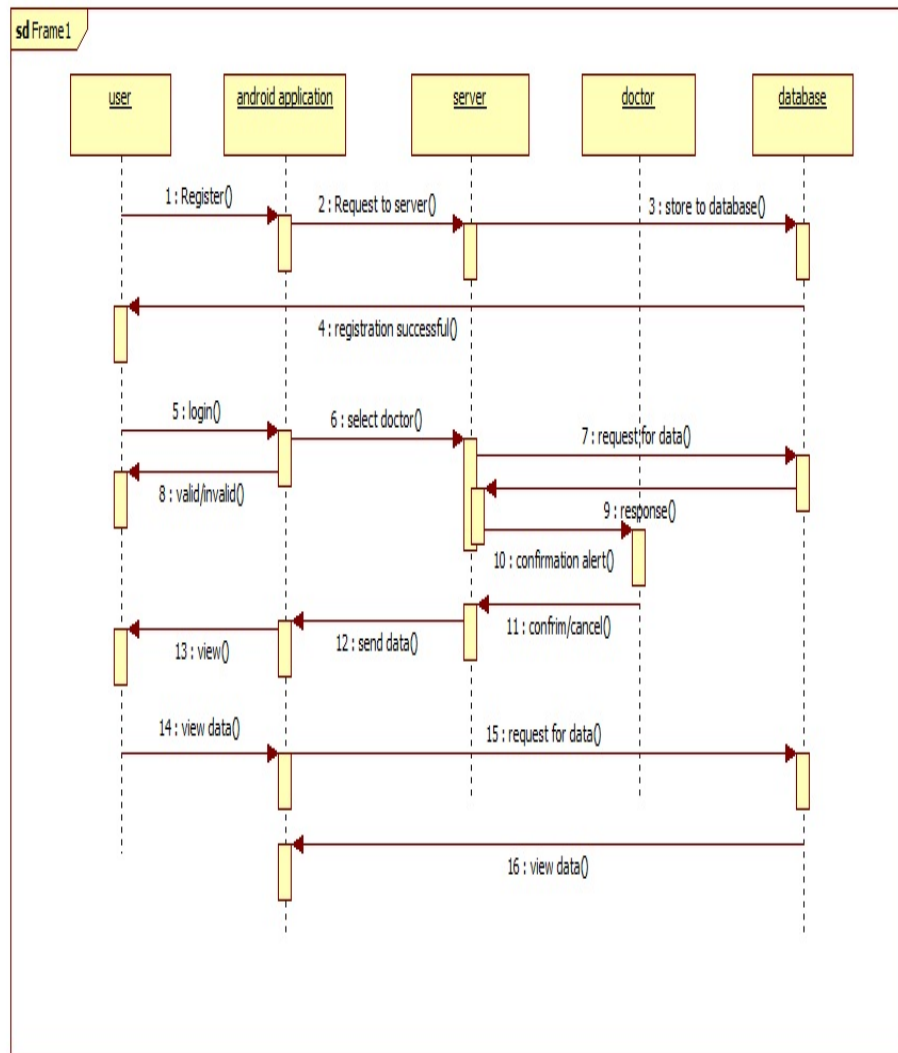
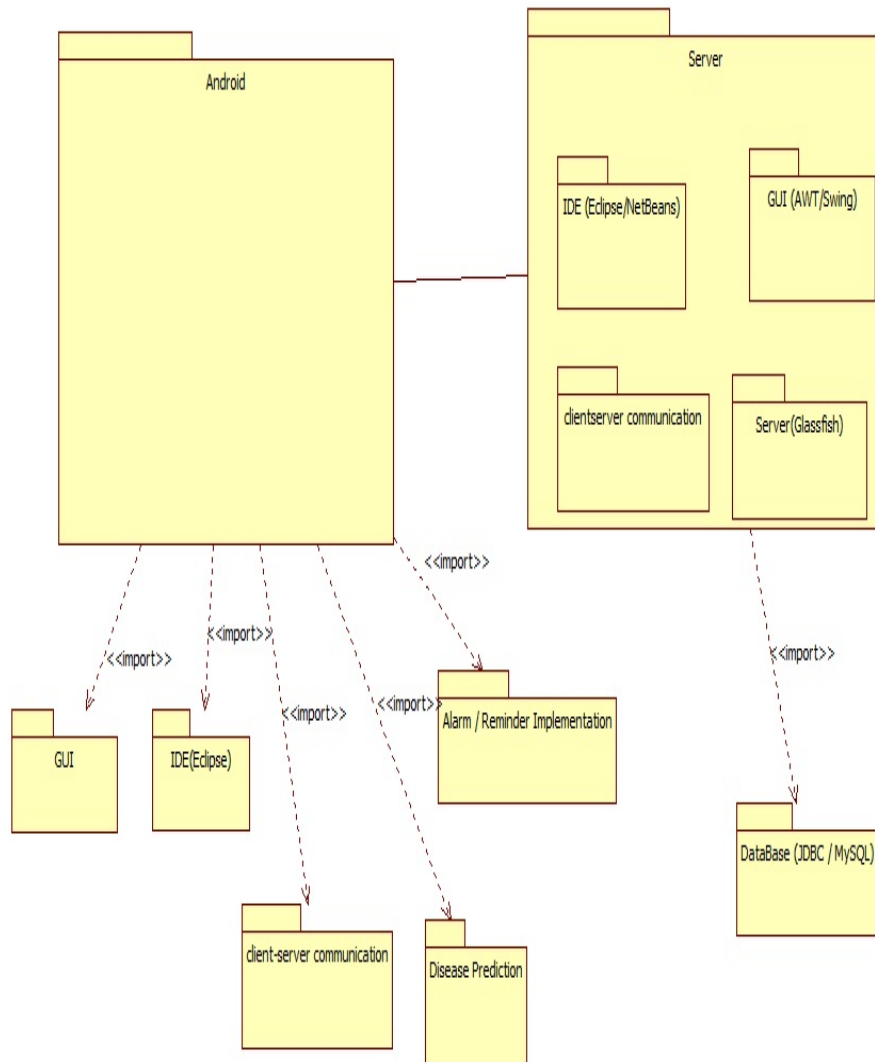
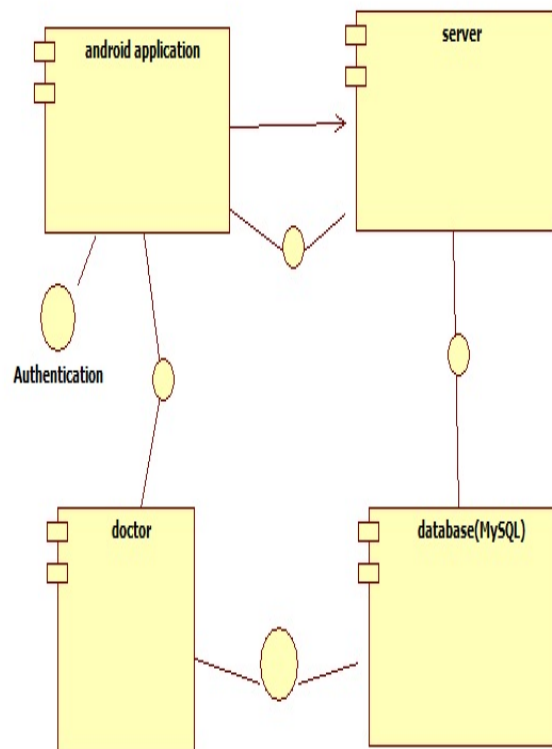


Figure 5.2: Interaction Diagram (Sequence Diagram)

Figure 5.3: *Package Diagram*

Figure 5.4: *Component Diagram*

Chapter 6

Technical Specifications

6.1 Technology details to be used in the project

Frontend

Android application (built using Eclipse ADT plug-in)

Backend

Database: MYSQL

6.2 References to Technology

- Abraham Silberschatz and Henry Korth, Database System Concepts,Alpha,2001
- Herbert Schildt, The Complete Reference Java seventh edition, Tata Mcgraw-Hill,2007
- Zigurd Mednieks, Android App Development, OReiley,2005
- www.lynda.com- Android App Development with Java Essential Training
- www.javatutorialhub.com
- www.developer.android.com

Chapter 7

Conclusion

Our proposed application will be feasible for use in the work environment as well as for family and friends. Also it will be user friendly and can be used with ease by the novice users as well as professional users. The proposed application will be used for the process of taking appointment, cancelling it. adding reminders for medical dosages. It will also predict the disease according to the inputted symptoms by user. This application will be innovative in its own aspect as it will minimize the dependency of user on the devices. This app will combine a number of functionalities into one, so the user need not download a number of applications for having access to information regarding ticketing process or booking a ticket.

The plus points of the applications are as follows:

1. It eliminates paper!
2. It eliminates the need to wait in queue for taking appointment.
3. It is secured and keeps personal data private from other users.
4. Improved customer experience Its interactive and in real time! Mobile app.

Thus what we are seeing is a significant force that could potentially change the entire healthcare industry.

Chapter 8

References

8.1 Papers

- NirmalaDevi.M, Appavu alias Balamurugan.S, Swathi U.V An amalgam KNN to predict Diabetes Mellitus,
- Asma A. AlJarullah, Decision Tree Discovery for the Diagnosis of Type II Diabetes
- Predicting the Analysis of Heart Disease Symptoms Using Medicinal Data Mining Methods
- Rajkumar Gaur Grewal, Babita Pandey Two level Diagnosis of Breast Cancer using data mining.
- Fatima Eddaoudi, Fakhita Regragui, Abdelhak Mahmoudi and Najib Lamouri. Masses Detection Using SVM Classifier Based on Textures Analysis
- Rahul Isola, Student Member, IEEE, Rebeck Carvalho, Student Member, IEEE, Amiya Kumar Tripathy, Member, IEEE. Knowledge Discovery in Medical Systems Using Differential Diagnosis, LAMSTAR, and k-NN.

8.2 Websites

- <http://en.wikipedia.org>
- <http://www.google.co.in>

Appendix A

Mathematical Models

A.1 Mathematical Model

A.1.1 Set Theory

$$Z = \{U, S, D\}$$

Where,

$$U = \text{Set of user} = \{P, H\}$$

$$P = \{\text{patient1, patient2 ...}\}$$

$$H = \{\text{hospital1, hospital2 ...}\}$$

$$U = P \cup H$$

$$D = \text{Set of database} = \{Dt\}$$

$$Dt = U \cup S$$

$$S = \text{Set of services} = \{\text{Access Records, Take Appointment, Reminders, Predict Diseases}\}$$

A.2 Functional Decomposition of the system

DIFFERENT FUNCTIONS:

Let

$f(g)$ = function of sign up.

$f(h)$ = function of appointment scheduling.

$f(i)$ = function of appointment conformation.

$f(j)$ = function of accessing medical records.

$f(k)$ = function of reminders.

$f(l)$ = function of disease prediction.

$f(m)$ = function of uploading records.

FUNCTIONAL DEPENDENCY:

$$f(g) : Dt \rightarrow U$$

$$A: f(i) \rightarrow f(h)$$

$$B: f(j) \rightarrow f(m)$$

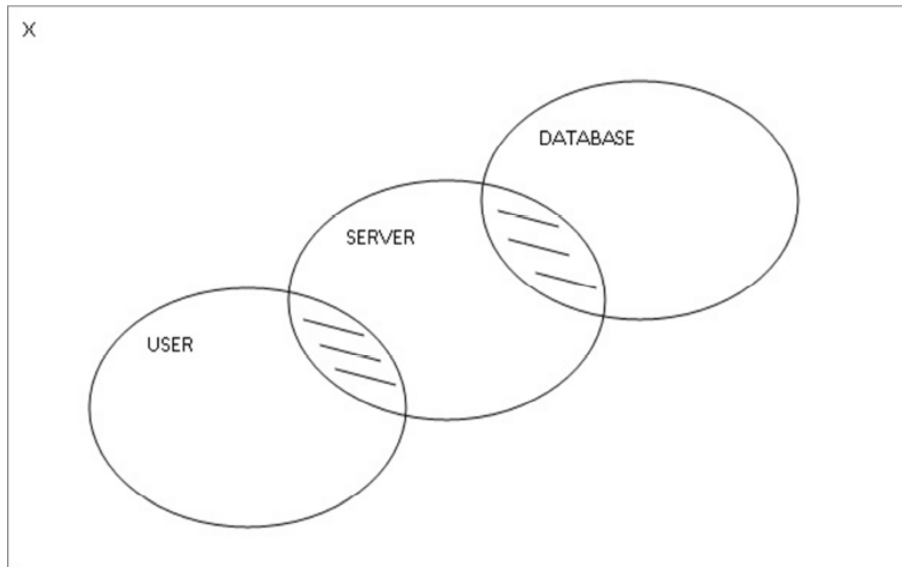
$$C: f(k) \rightarrow f(i) \cup f(m)$$

$$D: f(l) \rightarrow Dt$$

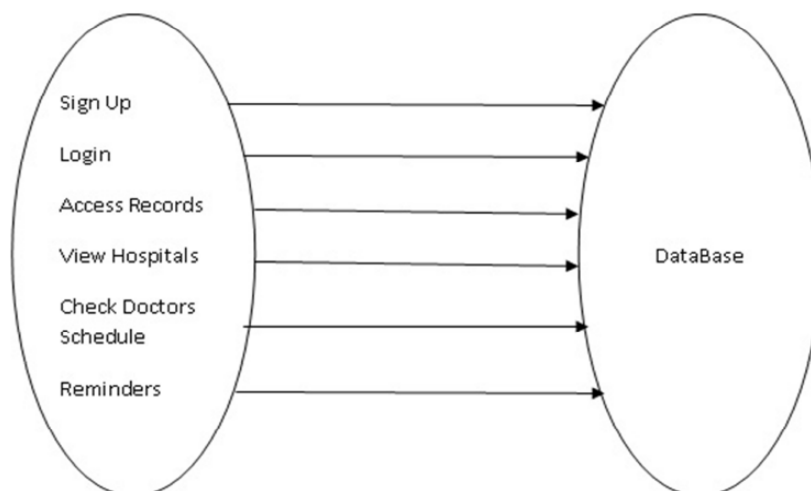
$$E: f(l) \rightarrow f(k)$$

$$F: f(m) \rightarrow f(l)$$

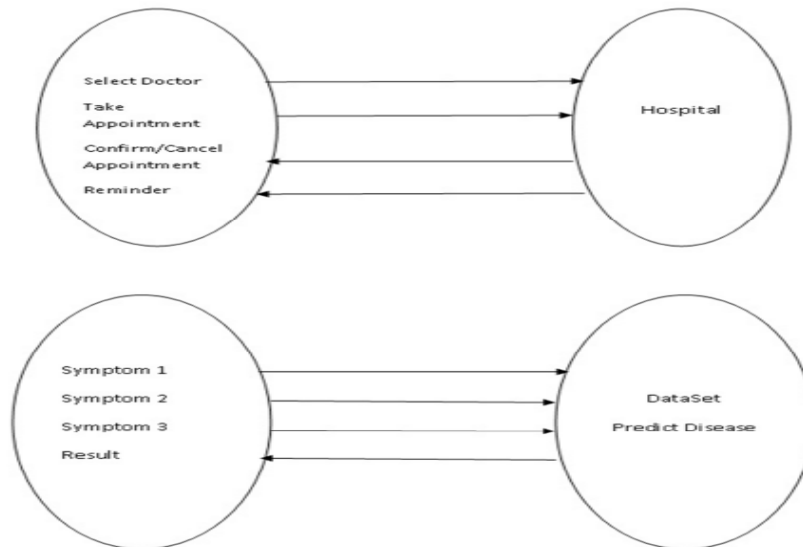
A.2.1 Venn Diagram



A.2.2 Functional Mapping-1



A.2.3 Functional Mapping-2



Appendix B

Testing/Reliability of Design

B.1 Test Plan for Design

Purpose of this plan is to explain step by step testing methodology for a developed android Urban Railway Ticketing application and is to be used this as a guide for the testing activity.

This test plan is designed to test android urban railway ticketing application. The test cases are intended to be ran manually. The test plan centers around ensuring correct functionality at the code and application levels.

The first level of testing is unit testing to ensure that the code that has been written functions correctly and continues to function correctly as the code is changed.

The next level of testing is manual testing to ensure that the functionality of the application are correctly implemented as per the user requirement (SRS and wireframe Documents.).

Testing Stages covered under manual testing.

1. Pre - Test.
2. User Interface.
3. Navigation.
4. Functionality testing.
5. Performance

Objective of this test plan is to ensure that the urban railway ticketing app functionality and component of the application are tested to the high standard and quality requirements are fulfilled.

1. Test plan Defining tasks and responsibilities
2. Identify required resources and related information.
3. Identify testing risks
4. Provide testing Schedule.

Testing Requirements

1. Each urban railway ticketing application program or method should be tested using Unit test at the time of development.
2. Graphical user interface should be tested using manual testing.
3. Functionality of software will be tested using functional testing.

Testing Strategy

The steps in testing consist:

1. Creation of all the test scenarios and test cases
2. Preparation of a test case document that has a brief description of the test case , steps to conduct tests and expected result.
3. Defect Report generation.

B.2 Details of Design Testing

The main testing types that will be performed.

1. **Pre Test:** This Pre - Testing provides general tests about launching the application.
2. **User Interface:** This User Interface testing provides tests about delivery, running the application, display, and using of the application.
3. **Navigation:** This Navigation testing provides tests about all menus, widgets and functions are accurate and correctly working.
4. **Functionality testing:** This User Interface testing provides tests about implementation of all the features in the application and work as per expectation. Sources for the information are SRS and wireframe Documents

5. **Performance Testing:** This Performance testing provides tests about response time (speed) and utilization of Application to execute its required functions in comparison with performance standard and The Application should perform at its best.

Appendix C

Project Management and Project Status

C.1 Project Management Approach

We have defined one tracking sheet for BE COMP where we are updating plan and actual dates as per the phase of project. HOD and Project coordinator track this sheet on monthly basis. Project status review takes place with project guide, project coordinator and HOD-COMP. Project guide reviews the documentation and project milestone in detail. Project coordinator tracks the timeliness and correctness of concepts. HOD ensures that all project team members are contributing to the project, team work is happening and project is within the framework as defined by the university guidelines.

Project team members meet once a week with the sponsoring company to provide the status and plan the activities for the current and next milestone.

C.2 Status of Project Work done

Sr. No	Task	Status
1	Synopsis	Completed
2	SRS Document	Completed
3	Design Completion	Yet to be completed
4	Coding	Yet to be completed
5	Testing	Yet to be completed

Table C.1: Project Status