E-HEALTHCARE SYSTEM

MINI-PROJECT REPORT

submitted in partial fulfillment of the requirements
for the award of the degree in

BACHELOR OF TECHNOLOGY
In
COMPUTER SCIENCE AND ENGINEERING

By

BELLAM BIPIN CHAKRAVARTHY(191061101032)
CHAKKA SATEESH KUMAR(191061101038)
ELLA KANAKA MANOJ KUMAR(191061101052)

Dr.M.G.R.
Educational and Research Institute
(DEEMED TO BE UNIVERSITY)
(An ISO Certified Institution)
UNIVERSITY WITH SPECIAL AUTONOMY STATUS
Maduravoyal , Chennai - 600 095

DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING

MAY 2022

Dr.M.G.R.
Educational and Research Institute
(DEEMED TO BE UNIVERSITY)
(An ISO Certified Institution)
UNIVERSITY WITH SPECIAL AUTONOMY STATUS
Maduravoyal , Chennai - 600 095

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that the Mini Project Report is the bonafide work of

Mr./Ms. **BELLAM BIPIN CHAKRAVARTHY** Reg.No. **191061101032**,

Mr./Ms. **CHAKKA SATEESH KUMAR** Reg.No. **191061101038**, and

Mr./Ms. **ELLA KANAKA MANOJ KUMAR** Reg.No. **191061101052** , who carried out the mini-project entitled "**E-HEALTHCARE SYSTEM**" under our supervision from Jan 2022 to May 2022.

Mini Project Coordinator 1	**Mini Project Coordinator 2**	**HoD**
Mrs.V.Vidhya	**Dr.G.Gunashekaran**	**Dr.S.Geetha**
Department of CSE	Department of CSE	Department of CSE
Dr. M.G.R. Educational	Dr. M.G.R. Educational	Dr. M.G.R. Educational
and Research Institute	and Research Institute	and Research Institute
Deemed to be University	Deemed to be University	Deemed to be University

Submitted for Viva Voce Examination held on_____

Internal Examiner **External Examiner**

DECLARATION

We **BELLAM BIPIN CHAKRAVARTHY (191061101032), CHAKKA SATEESH KUMAR (191061101038), ELLA KANAKA MANOJ KUMAR (191061101052),** hereby declare that the Mini Project Report entitled **"E-HEALTHCARE SYSTEM"** is done by us under the guidance of "**Mrs.V.Vidhya & Dr.G.Gunashekaran"** is submitted in partial fulfillment of the requirements for the award of the degree in Bachelor of Technology in Computer Science and Engineering.

1. BELLAM BIPIN CHAKRAVARTHY.
2. CHAKKA SATEESH KUMAR.
3. ELLA KANAKA MANOJ KUMAR.

SIGNATURE OF THE CANDIDATE(S)

DATE: 23/05/2022

PLACE: CHENNAI

ACKNOWLEDGEMENT

We would first like to thank our beloved Chancellor **Thiru A.C.SHANMUGAM, B.A, B.L.** and **Er. A.C.S. ARUNKUMAR, B.Tech, M.B.A,** and for all the encouragement and support extended to us during the tenure of this project and also our years of studies in his wonderful University.

We express our heartfelt thanks our Vice Chancellor **Prof. Dr. S. Geethalakshmi** in providing all the support of our Mini Project.

We express our heartfelt thanks to our Head of the Department, **Prof. Dr. S. Geetha,** who has been actively involved and very influential from the start till the completion of our Mini Project.

Our sincere thanks to our Project Coordinators **Mrs.V.Vidhya & Dr.G.Gunashekaran**, for their continuous guidance and encouragement throughout this work, which had made this the Mini Project a success.

We would also like to thank all the teaching and non-teaching staffs of Computer Science and Engineering Department, for their constant support and the encouragement given to us while we went about to achieving my project goals.

ABSTRACT

The purpose of the project "E-HEALTHCARE SYSTEM" is to computerize the management of hospital to develop software which is user friendly simple, fast, and cost – effective. It deals with the collection of patient's information and other details, etc. Typically, it was done manually. The main function of the system is register and store patient details and doctor details and retrieve these details as and when required, and also to manipulate these details meaningfully. System input contains patient details and other details, while system output is to get these details on to the screen. The E-Healthcare System can be logged in using a username and password. It is accessible by an administrator. Only he/she can add data into the database. The data can be retrieved easily. The data are well protected for personal use and makes the data processing very fast.

INDEX

CHAPTER-1

INTRODUCTION

1.1 Introduction

The project E-Healthcare system is designed for any hospital to replace their existing manual paper based system. It consists of two modules, admin, doctor and user, the user module is known as Patient module. It includes registration of patients, storing their details into the system and retrieve it when required. It includes a search facility for a patient to take an appointment from a specific doctor. User can search availability of a doctor.

The E-Healthcare system can be entered using a username and password. It is accessible by an administrator. Only they can add data into the database .The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast.

E-Healthcare system is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals.

E-Healthcare system is designed for multi-speciality hospitals, to cover a wide range of hospital administration and management processes. It is an integrated end-to-end E-Healthcare System that provides relevant information across the hospital to support effective decision making for patient care in a seamless flow.

E-Healthcare system is a software product suite designed to improve the quality and management of hospital management in the areas of clinical process analysis by prescribing necessary medication to the patient. E-Healthcare system enables you to develop your organization and improve its effectiveness and quality of work. Managing the key processes efficiently is critical to the success of the hospital helps you manage your processes.

1.2 Problem Introduction

In hospital, it is very important and difficult to maintain information of every patient and also need to retrieve when required. Preparation of accurate and prompt reports is very difficult and also more paper work is required. Manual calculations are error prone and takes lot of time. It is very important to maintain efficient software to handle information of a Hospital. This is a php based web application which is useful to record all the information and to access it in a simple way.

1.2.1 Problem Statement

In this busy world we don't have the time to wait in infamously long hospital queues. The problem is, queuing at hospital is often managed manually by administrative staff, then take a token there and then wait for our turn then ask for the doctor and the most frustrating thing - we went there by traveling a long distance and then we come to know the doctor is on leave or the doctor can't take appointments.

E-HCS will help us overcome all these problems because now patients can book their appointments at home, they can check whether the doctor they want to meet is available or not. Doctors can also confirm or decline appointments, this help both patient and the doctor because if the doctor declines' appointment then patient will know this in advance and patient will visit hospital only when the doctor confirms' the appointment this will save time and money of the patient.

Patients can also pay the doctor's consultant fee online to save their time. E-HCS is essential for all healthcare establishments, be it hospitals, nursing homes, health clinics, rehabilitation centers, dispensaries, or clinics. The main goal is to computerize all the details regarding the patient and the hospital. The installation of this healthcare software results in improvement in administrative functions and hence better patient care, which is the prime focus of any healthcare unit.

1.2.2 Benefits of implementing a E-Healthcare System

- **Appointment booking**
  - Helps patients cut the long queue and saves their time
  - Is equipped with features like automated email and text message reminders
- **Role-Based Access Control**
  - Allows employees to access only the necessary information to effectively perform their job duties
  - Increases data security and integrity
- **Overall cost reduction**
  - Cuts down paper costs as all the data are computerized
  - No separate costs for setting up physical servers
- **Data accuracy**
  - Removes human errors
  - Alerts when there's a shortage of stock
- **Data security**
  - Helps to keep patients records private
  - Restricts access through role-based access control
- **Revenue management**
  - Makes daily auditing simple
  - Helps with statistics and other financial aspects

1.3 Objective of the Project

- Provide top management at single point of control.
- Develop a user friendly, fast and cost effective software for suitable requirements.
- Better co-ordination among different departments.
- Design a system for better patient care.
- Easy access for patient to get an appointment from a doctor based on their specilizations.
- Recording information about the Patients that come.
- Recording information related to medications and prescriptions given to Patients.
- Reduce hospital operating costs.

1.4 Modules in the Project

Admin Panel

Admin has the full access to the system which means he is able to manage any activity with regard to the system. He is the highest privileged user who can access to the system.

- Access patient record, doctor Record.
- Add new doctor entry in system database.
- View Records.(Total no of patients treated, doctor added/remove).
- Watch appointment of doctors.
- View Prescriptions of patients

Doctor Panel

Doctors can view the patient appointment list and provide the confirmation or make changes in the appointment list if required. Doctors have access to only records of those patients whom they are treating.

- Confirmation of appointment.
- Cancellation of appointment.
- Modification of appointment list.
- Add Prescription.
- Provide medication.

Patient Panel

Patients can choose the best preferred appointments from the options provided and can also change the appointment schedule or cancel it. After appt. is confirmed by the respective doctor they can pay their consultant fee online. Patients have access to only their records.

- Make appointment.
- Cancel appointment.
- View appointment list and status with doctors
- View prescription details.
- View medication from doctor

Available Features

- Patient Panel
- Doctor Panel
- Admin Panel
- Book an appointment
- Appointment history
- Prescription list
- Manage appointment (doctor)
- Prescribe
- Manage Doctors
- View Available Patients
- Search Records
- List overall appointment, prescriptions

CHAPTER-2

REQUIREMENTS ANALYSIS

2.1 Existing System

Hospitals currently use a manual system for the management and maintenance of critical information. The current system requires numerous paper forms, with data stores spread throughout the hospital management infrastructure. Often information is incomplete or does not follow management standards. Forms are often lost in transit between departments requiring a comprehensive auditing process to ensure that no vital information is lost. Multiple copies of the same information exist in the hospital and may lead to inconsistencies in data in various data stores. If a patient wants to take an appointment from a doctor, the patient goes to the Hospital for an appointment which is a time waste process. Also, the Doctor doesn't have proper schedule to deal with the patients.

2.2 Proposed System

The E-Healthcare System is designed for any hospital to replace their existing manual paper based system. The new system is to control the information of patients. These services are to be provided in an efficient, fast, cost effective manner, with the goal of reducing the time and resources currently required for such tasks. The new system helps the Doctor to manage their schedule. Also has an easy access for a patient to get an appointment from a doctor based on their specilizations. In this system the prescription can be given by there itself so, there will be no worry of loosing the prescription.

Advantages

- The system automates the manual procedure of managing hospital activities.
- Doctors can view their patients' treatment records and details easily.
- The system is convenient and flexible to be used.
- It saves their time, efforts, money and resources.

Disadvantages

- Requires large database.
- The admin has to manually keep updating the information by entering the details in the system.
- Need Internet connection.

2.3 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following below discuss about the various aspects of minimal hardware requirements.

PROCESSOR	:	Intel dual Core ,i3
RAM	:	2 GB
HARD DISK	:	80 GB

2.4 Software Requirements

Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

OPERATING SYSTEM	:	Windows 7/8/9/10 operating system
FRONT END	:	HTML, CSS, Java Script.
SERVER SIDE SCRIPT	:	PHP.
DATABASE	:	MySQL.

2.4.1 Software Specification

HTML

The **HyperText Markup Language** or **HTML** is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a webpage semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by *tags*, written using angle brackets. Tags such as <**img** /> and <**input** /> directly introduce content into the page. Other tags such as <**p**> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

CSS

Cascading Style Sheets (**CSS**) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

JavaScript

JavaScript often abbreviated **JS**, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users' devices.

JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O. JavaScript engines were originally used only in web browsers, but are now core components of some servers and a variety of applications. The most popular runtime system for this usage is Node.js. Although Java and JavaScript are similar in name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

PHP

PHP is an open-source, interpreted, and object-oriented scripting language that can be executed at the server-side. PHP is well suited for web development. Therefore, it is used to develop web applications (an application that executes on the server and generates the dynamic page.). Some important points need to be noticed about PHP are:

- PHP stands for Hypertext Preprocessor.
- PHP is an interpreted language, i.e., there is no need for compilation.
- PHP is faster than other scripting languages, for example, ASP and JSP.
- PHP is a server-side scripting language, which is used to manage the dynamic content of the website.
- PHP can be embedded into HTML.
- PHP is an object-oriented language.
- PHP is an open-source scripting language.
- PHP is simple and easy to learn language.

Why use PHP

PHP is a server-side scripting language, which is used to design the dynamic web applications with MySQL database.

- It handles dynamic content, database as well as session tracking for the website.
- You can create sessions in PHP.
- It can access cookies variable and also set cookies.
- It helps to encrypt the data and apply validation.
- PHP supports several protocols such as HTTP, POP3, SNMP, LDAP, IMAP, and many more.
- Using PHP language, you can control the user to access some pages of your website.
- As PHP is easy to install and set up, this is the main reason why PHP is the best language to learn.
- PHP can handle the forms, such as - collect the data from users using forms, save it into the database, and return useful information to the user. **For example** - Registration form.

MySQL

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

MySQL is a Relational Database Management System (RDBMS) software that provides many things, which are as follows:

- It allows us to implement database operations on tables, rows, columns, and indexes.
- It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.
- It provides the Referential Integrity between rows or columns of various tables.
- It allows us to updates the table indexes automatically.
- It uses many SQL queries and combines useful information from multiple tables for the end-users.

MySQL follows the working of Client-Server Architecture. This model is designed for the end-users called clients to access the resources from a central computer known as a server using network services. Here, the clients make requests through a graphical user interface (GUI), and the server will give the desired output as soon as the instructions are matched. The process of MySQL environment is the same as the client-server model.

The core of the MySQL database is the MySQL Server. This server is available as a separate program and responsible for handling all the database instructions, statements, or commands. The working of MySQL database with MySQL Server are as follows:

- MySQL creates a database that allows you to build many tables to store and manipulate data and defining the relationship between each table.
- Clients make requests through the GUI screen or command prompt by using specific SQL expressions on MySQL.
- Finally, the server application will respond with the requested expressions and produce the desired result on the client-side.

CHAPTER-3

DESIGN

3.1 System Design

3.1.1 Introduction to UML

UML Design

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the software system and its components. It is a graphical language , which provides a vocabulary and set of semantics and rules. The UML focuses on the conceptual and physical representation of the system. It captures the decisions and understandings about systems that must be constructed. It is used to understand, design, configure, maintain, and control information about the systems.

The UML is a language for:

- Visualizing
- Specifying
- Constructing
- Documenting

Visualizing

Through UML we see or visualize an existing system and ultimately we visualize how the system is going to be after implementation. Unless we think, we cannot implement. UML helps to visualize, how the components of the system communicate and interact with each other.

Specifying

Specifying means building, models that are precise, unambiguous and complete UML addresses the specification of all the important analysis design, implementation decisions that must be made in developing and deploying a software system.

Constructing

UML models can be directly connected to a variety of programming language through mapping a model from UML to a programming language like JAVA or C++ or VB. Forward Engineering and Reverse Engineering is possible through UML.

Documenting

The Deliverables of a project apart from coding are some Artifacts, which are critical in controlling, measuring and communicating about a system during its developing requirements, architecture, desire, source code, project plans, tests, prototypes releasers, etc...

3.1.2 UML Approach

UML Diagram

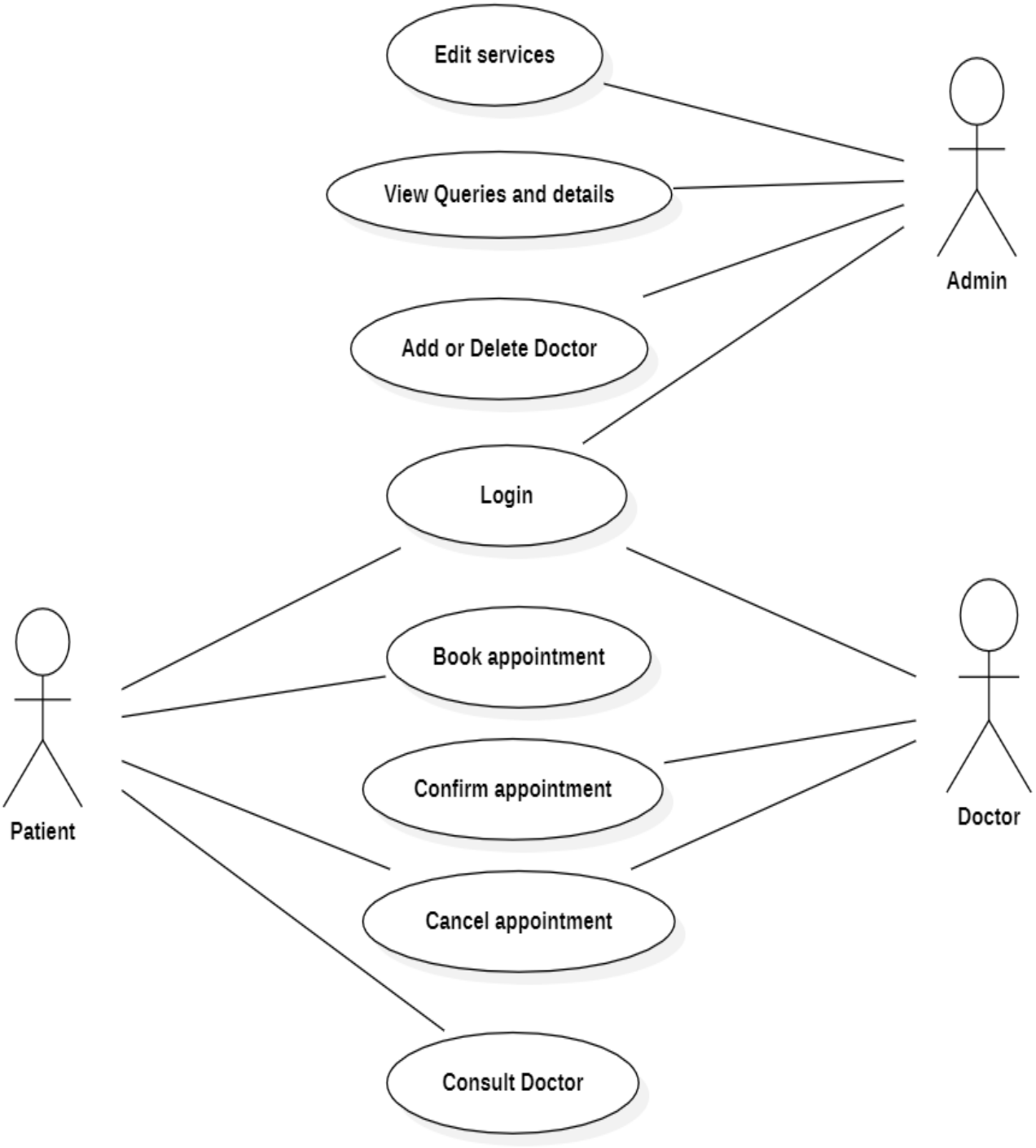
A diagram is the graphical presentation of a set of elements, most often rendered as a connected graph of vertices and arcs . you draw diagram to visualize a system from different perspective, so a diagram is a projection into a system. For all but most trivial systems, a diagram represents an elided view of the elements that make up a system. The same element may appear in all diagrams, only a few diagrams , or in no diagrams at all. In theory, a diagram may contain any combination of things and relationships. In practice, however, a small number of common combinations arise, which are consistent with the five most useful views that comprise the architecture of a software-intensive system.

Use Case Diagram

A usecase diagram in the Unified Modeling Language(UML) is atype of behavioral diagram defined by and created from a use-case analysis.its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals(represented as use cases),and any dependencies between those use cases.

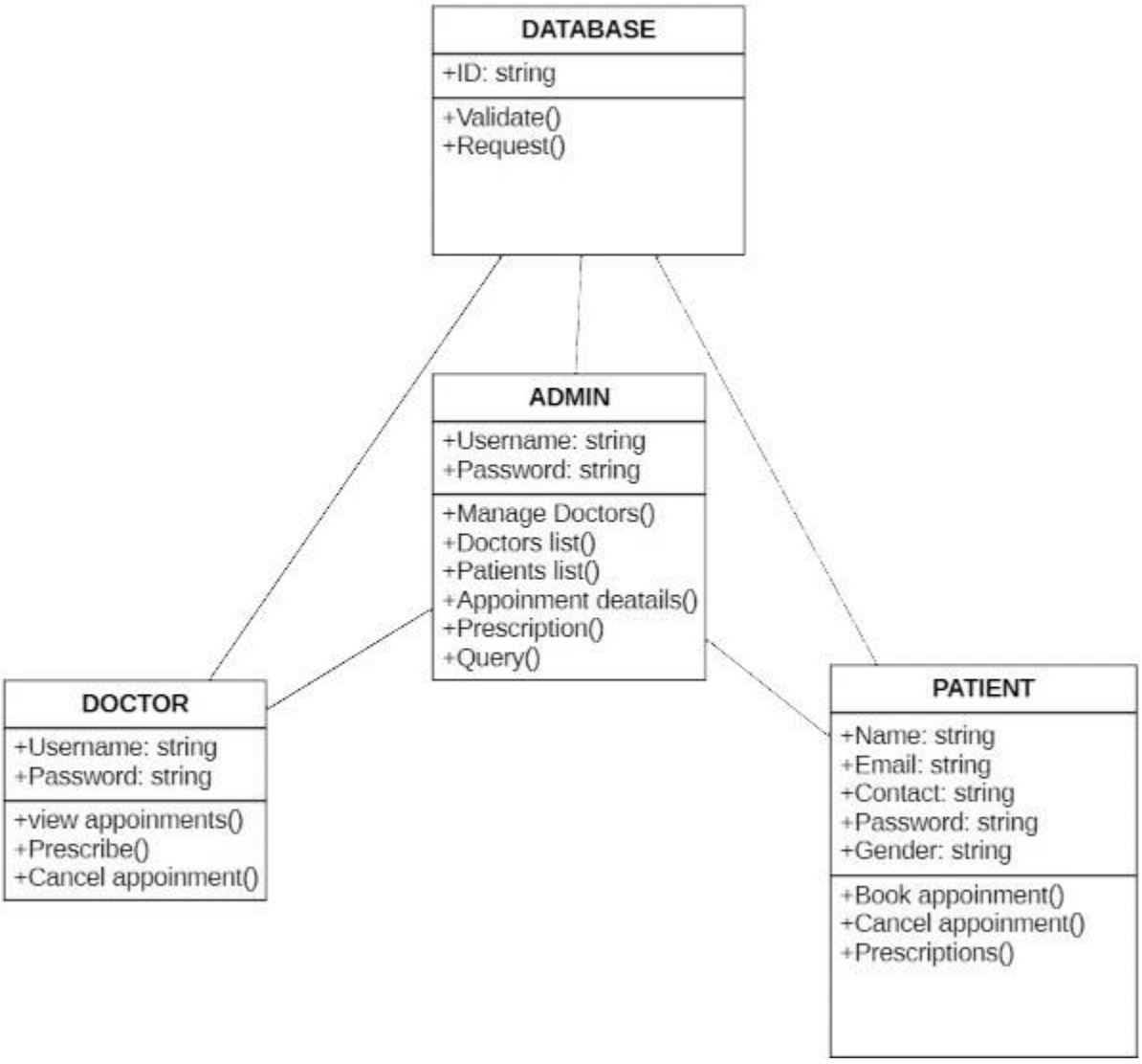
Use case diagrams are formally included in two modeling languages defined by the OMG:theunfied modeling language(UML) and the systems modeling language(sysML).

E-HEALTHCARE SYSTEM



Class Diagram

A Class is a category or group of things that has similar attributes and common behavior. A Rectangle is the icon that represents the class it is divided into three areas. The upper most area contains the name, the middle; area contains the attributes and the lowest areas show the operations. Class diagrams provides the representation that developers work from. Class diagrams help on the analysis side, too.



CHAPTER-4

IMPLEMENTATION

4.1 Modules Description

This simple E-Healthcare System project in PHP focuses mainly on dealing with numerous hospital records. Also, the system displays all the available records of patients and their appointments. In addition, the system allows registering doctors too. The project is divided into three categories: Admin, Doctor, and Patient. A clean and simple dashboard is presented with simple color combinations for greater user experience while using this Hospital Management System Project in PHP MySQL. For its UI elements, a free open-source CSS framework.

Admin Panel

On the other hand, the admin has full control of the system. An admin plays an important role in the proper flow of this whole system. An admin has the right to manage a doctor's account. And for the removal of the doctor's account, he/she has to enter up doctor's email address to proceed. The admin can list all available patients, doctors, and their appointments with listed prescriptions. In fact, an admin can have an overview of all records around the system. He/she can view all the prescriptions from each doctor. Besides, the admin can perform search functions on each section which helps to filter up the records on time. Additionally, the admin can view all the contact queries with names, emails, and messages of each.

Doctor Panel

Coming towards the doctor's panel, it all depends upon the patient's appointment request. The main responsibility of a doctor's account is to approve the appointment and prescribe medicines. As soon as the patient book an appointment, that particular doctor gets a request. Each field contains the patient's name, gender, email, date, and time. Here, he/she can cancel or prescribe the appointment. After approving an appointment, the doctor has to fill up certain fields. These fields include the name of the disease, allergies, and prescriptions. After submission, the patient can view his/her prescription and proceed towards bills payment. Besides, the doctor can view appointment and prescription histories.

Patient Panel

The patient can proceed with booking an appointment. For this, the user has to select doctor, specialization, date, and time. After the selection of the doctor, the system displays the total fees for it. Besides, the patient can view his/her appointment history and prescription details. Also, the patient can cancel an appointment anytime.

4.2 Sample Code

index.php

```
<!DOCTYPE html>
<html>
<head>
<title>E-HCS</title>
<link rel="shortcut icon" type="image/x-icon" href="images/favicon.png" />
<link rel="stylesheet" type="text/css" href="style1.css">
<style >
.form-control {
border-radius: 0.75rem;
}</style>
<script>
var check = function() {
if (document.getElementById('password').value ==
document.getElementById('cpassword').value) {
document.getElementById('message').style.color = '#5dd05d';
document.getElementById('message').innerHTML = 'Matched';
} else {
document.getElementById('message').style.color = '#f55252';
document.getElementById('message').innerHTML ='Password fields doesnot match';}}
function alphaOnly(event) {
var key = event.keyCode;
return ((key >= 65 && key <= 90) || key == 8 || key == 32);};
function checklen()
{ var pass1 = document.getElementById("password");
if(pass1.value.length<6){
alert("Password must be at least 6 characters long. Try again!");
return false;  }}
</script></head>
<body>
<nav class="navbar navbar-expand-lg navbar-dark fixed-top" id="mainNav" >
<div class="container">
<a class="navbar-brand js-scroll-trigger" href="#" style="margin-top: 10px;margin-left:-
65px;font-family: 'IBM Plex Sans', sans-serif;"><h4><i class="fa fa-hospital-o" aria-
hidden="true"></i>  E-HEALTHCARE SYSTEM</h4></a>
<button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-
label="Toggle navigation">
<span class="navbar-toggler-icon"></span></button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item" style="margin-right: 40px;">
<a class="nav-link js-scroll-trigger" href="index.php" style="color: white;font-family: 'IBM
Plex Sans', sans-serif;"><h6>HOME</h6></a></li>
<li class="nav-item">
```

```
<a class="nav-link js-scroll-trigger" href="contact.html" style="color: white;font-family: 'IBM Plex Sans', sans-serif;"><h6>CONTACT</h6></a>
</li></ul></div></div></nav>
<div class="container register" style="font-family: 'IBM Plex Sans', sans-serif;">
<div class="row">
<div class="col-md-3 register-left" style="margin-top: 10%;right: 5%">
<img src="https://image.ibb.co/n7oTvU/logo_white.png" alt=""/>
<h3>WELCOME</h3></div>
<div class="col-md-9 register-right" style="margin-top: 40px;left: 80px;">
<ul class="nav nav-tabs nav-justified" id="myTab" role="tablist" style="width: 40%;">
<li class="nav-item">
<a class="nav-link active" id="home-tab" data-toggle="tab" href="#home" role="tab" aria-controls="home" aria-selected="true">Patient</a></li>
<li class="nav-item">
<a class="nav-link" id="profile-tab" data-toggle="tab" href="#profile" role="tab" aria-controls="profile" aria-selected="false">Doctor</a>
</li>
<li class="nav-item">
<a class="nav-link" id="profile-tab" data-toggle="tab" href="#admin" role="tab" aria-controls="admin" aria-selected="false">Admin</a></li></ul>
<div class="tab-content" id="myTabContent">
<div class="tab-pane fade show active" id="home" role="tabpanel" aria-labelledby="home-tab">
<h3 class="register-heading">Register as Patient</h3>
<form method="post" action="func2.php">
<div class="row register-form">
<div class="col-md-6">
<div class="form-group">
<input type="text" class="form-control"  placeholder="First Name *" name="fname" onkeydown="return alphaOnly(event);" required/></div>
<div class="form-group">
<input type="email" class="form-control" placeholder="Your Email *" name="email" /></div>
<div class="form-group">
<input type="password" class="form-control" placeholder="Password *" id="password" name="password" onkeyup='check();' required/></div>
<div class="form-group">
<div class="maxl">
<label class="radio inline">
<input type="radio" name="gender" value="Male" checked>
<span> Male </span> </label>
<label class="radio inline">
<input type="radio" name="gender" value="Female">
<span>Female </span> </label></div>
<a href="index1.php">Already have an account? Login Now</a>
</div></div>
<div class="col-md-6"><div class="form-group">
<input type="text" class="form-control" placeholder="Last Name *" name="lname" onkeydown="return alphaOnly(event);" required/></div>
```

```
<div class="form-group">
<input type="tel" minlength="10" maxlength="10" name="contact" class="form-control"
placeholder="Contact *"  /></div>
<div class="form-group">
<input type="password" class="form-control"  id="cpassword" placeholder="Confirm
Password *" name="cpassword"  onkeyup='check();' required/><span
id='message'></span></div>
<input type="submit" class="btnRegister" name="patsub1" onclick="return checklen();"
value="Register"/>
</div></div></form></div>
<div class="tab-pane fade show" id="profile" role="tabpanel" aria-labelledby="profile-tab">
<h3  class="register-heading">Login as Doctor</h3>
<form method="post" action="func1.php">
<div class="row register-form">
<div class="col-md-6">
<div class="form-group">
<input type="text" class="form-control" placeholder="User Name *" name="username3"
onkeydown="return alphaOnly(event);" required/>
</div></div>
<div class="col-md-6"><div class="form-group">
<input type="password" class="form-control" placeholder="Password *" name="password3"
required/></div>
<input type="submit" class="btnRegister" name="docsub1" value="Login"/>
</div></div></form></div>
<div class="tab-pane fade show" id="admin" role="tabpanel" aria-labelledby="profile-tab">
<h3  class="register-heading">Login as Admin</h3>
<form method="post" action="func3.php">
<div class="row register-form">
<div class="col-md-6">
<div class="form-group">
<input type="text" class="form-control" placeholder="User Name *" name="username1"
onkeydown="return alphaOnly(event);" required/>
</div></div>
<div class="col-md-6">
<div class="form-group">
<input type="password" class="form-control" placeholder="Password *" name="password2"
required/></div>
<input type="submit" class="btnRegister" name="adsub" value="Login"/>
</div></div></form></div></div></div></div></div></body>
</html>
```

admin-panel.php

```
<!DOCTYPE html>
<?php
$con=mysqli_connect("localhost","root","","hospitalms");
include('newfunc.php');
if(isset($_POST['docsub']))
{
  $doctorname=$_POST['doctorname'];
  $doctor=$_POST['doctor'];
  $dpassword=$_POST['dpassword'];
  $demail=$_POST['demail'];
  $spec=$_POST['special'];
  $docFees=$_POST['docFees'];
  $query="insert into doctb(doctorname,username,password,email,spec,docFees) values
('$doctorname' , '$doctor','$dpassword','$demail','$spec','$docFees')";
  $result=mysqli_query($con,$query);
  if($result)
    {
      echo "<script>alert('Doctor added successfully!');</script>";
  }
}
if(isset($_POST['docsub1']))
{
  $demail=$_POST['demail'];
  $query="delete from doctb where email='$demail';";
  $result=mysqli_query($con,$query);
  if($result)
    {
      echo "<script>alert('Doctor removed successfully!');</script>";
  }
  else{
    echo "<script>alert('Unable to delete!');</script>";
  }
}
?>
```

doctor-panel.php

```
<!DOCTYPE html>
<?php
include('func1.php');
$con=mysqli_connect("localhost","root","","hospitalms");
$doctor = $_SESSION['dname'];
if(isset($_GET['cancel']))
  {
    $query=mysqli_query($con,"update appointmenttb set doctorStatus='0' where
ID = '".$_GET['ID']."'");
    if($query)
```

```
    {
      echo "<script>alert('Your appointment successfully cancelled');</script>";
    }
  }
   if(isset($_GET['prescribe']))
   {
     $pid = $_GET['pid'];
     $ID = $_GET['ID'];
     $appdate = $_GET['appdate'];
     $apptime = $_GET['apptime'];
     $disease = $_GET['disease'];
     $prescription = $_GET['prescription'];
     $query=mysqli_query($con,"insert into
prestb(doctor,pid,ID,appdate,apptime,disease,prescription) values
('$doctor',$pid,$ID,'$appdate','$apptime','$disease','$prescription');");
     if($query)
     {
       echo "<script>alert('Prescribed successfully!');</script>";
     }
     else{
       echo "<script>alert('Unable to process your request. Try again!');</script>";
     }
   }
?>
```

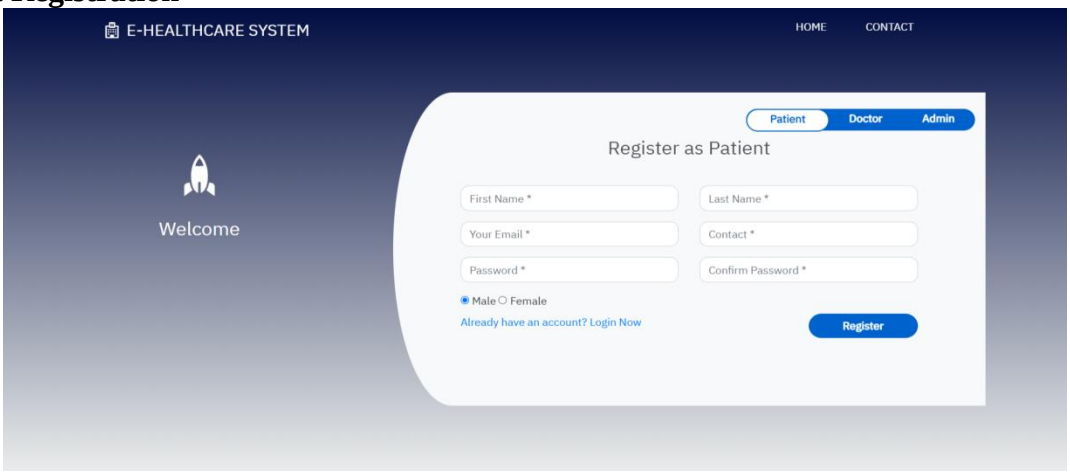
patient-panel.php

```
<!DOCTYPE html>
<?php
include('func.php');
include('newfunc.php');
$con=mysqli_connect("localhost","root","","hospitalms");
  $pid = $_SESSION['pid'];
  $username = $_SESSION['username'];
  $email = $_SESSION['email'];
  $fname = $_SESSION['fname'];
  $gender = $_SESSION['gender'];
  $lname = $_SESSION['lname'];
  $contact = $_SESSION['contact'];
if(isset($_POST['app-submit']))
{
  $pid = $_SESSION['pid'];
  $username = $_SESSION['username'];
  $email = $_SESSION['email'];
  $fname = $_SESSION['fname'];
  $lname = $_SESSION['lname'];
  $gender = $_SESSION['gender'];
  $contact = $_SESSION['contact'];
  $doctor=$_POST['doctor'];
```

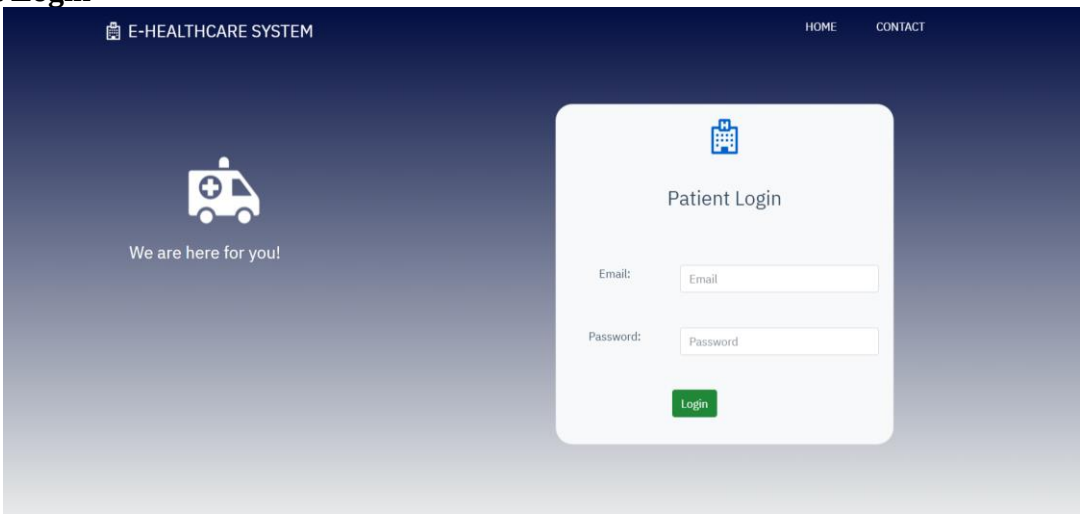
```
  $email=$_SESSION['email'];
  # $fees=$_POST['fees'];
  $docFees=$_POST['docFees'];
  $appdate=$_POST['appdate'];
  $apptime=$_POST['apptime'];
  $cur_date = date("Y-m-d");
  date_default_timezone_set('Asia/Kolkata');
  $cur_time = date("H:i:s");
  $apptime1 = strtotime($apptime);
  $appdate1 = strtotime($appdate);
  if(date("Y-m-d",$appdate1)>=$cur_date){
    if((date("Y-m-d",$appdate1)==$cur_date and date("H:i:s",$apptime1)>$cur_time) or
date("Y-m-d",$appdate1)>$cur_date) {
      $check_query = mysqli_query($con,"select apptime from appointmenttb where
doctor='$doctor' and appdate='$appdate' and apptime='$apptime'");
        if(mysqli_num_rows($check_query)==0){
          $query=mysqli_query($con,"insert into
appointmenttb(pid,fname,lname,gender,email,contact,doctor,docFees,appdate,apptime,userStatu
s,doctorStatus)
values($pid,'$fname','$lname','$gender','$email','$contact','$doctor','$docFees','$appdate','$appti
me','1','1')");
          if($query){
            echo "<script>alert('Your appointment successfully booked');</script>";}
          else{
            echo "<script>alert('Unable to process your request. Please try again!');</script>";
          }}else{
        echo "<script>alert('We are sorry to inform that the doctor is not available in this time or
date. Please choose different time or date!');</script>";
      }}else{
      echo "<script>alert('Select a time or date in the future!');</script>";
    }}else{
      echo "<script>alert('Select a time or date in the future!');</script>";
  }}
if(isset($_GET['cancel'])){
    $query=mysqli_query($con,"update appointmenttb set userStatus='0' where ID =
'".$_GET['ID']."'");
    if($query){
      echo "<script>alert('Your appointment successfully cancelled');</script>";
    }}
function get_specs(){
  $con=mysqli_connect("localhost","root","","hospitalms");
  $query=mysqli_query($con,"select username,spec from doctb");
  $docarray = array();
    while($row =mysqli_fetch_assoc($query)){
       $docarray[] = $row;
    }
    return json_encode($docarray);
}
?>
```

4.3 Sample Screenshots

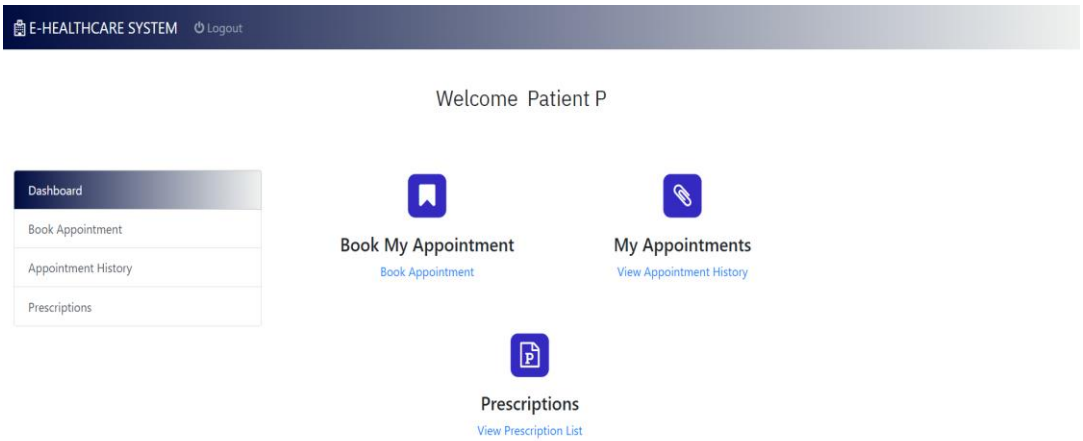
Patient Registration



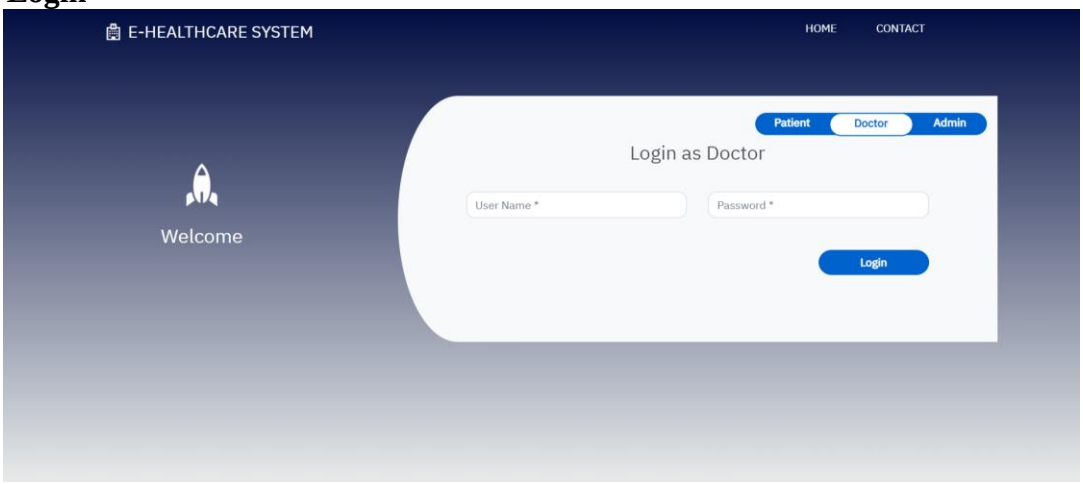
Patient Login



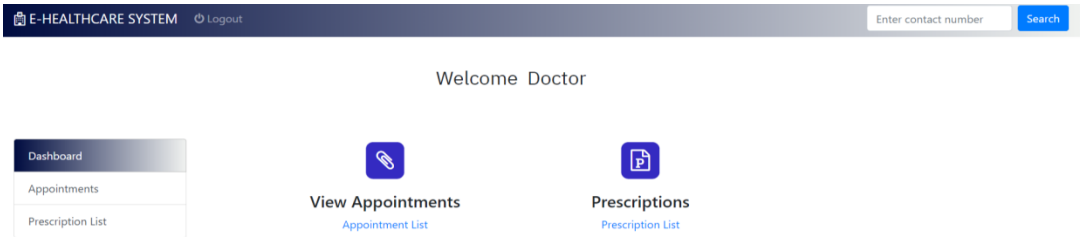
Patient Dashboard



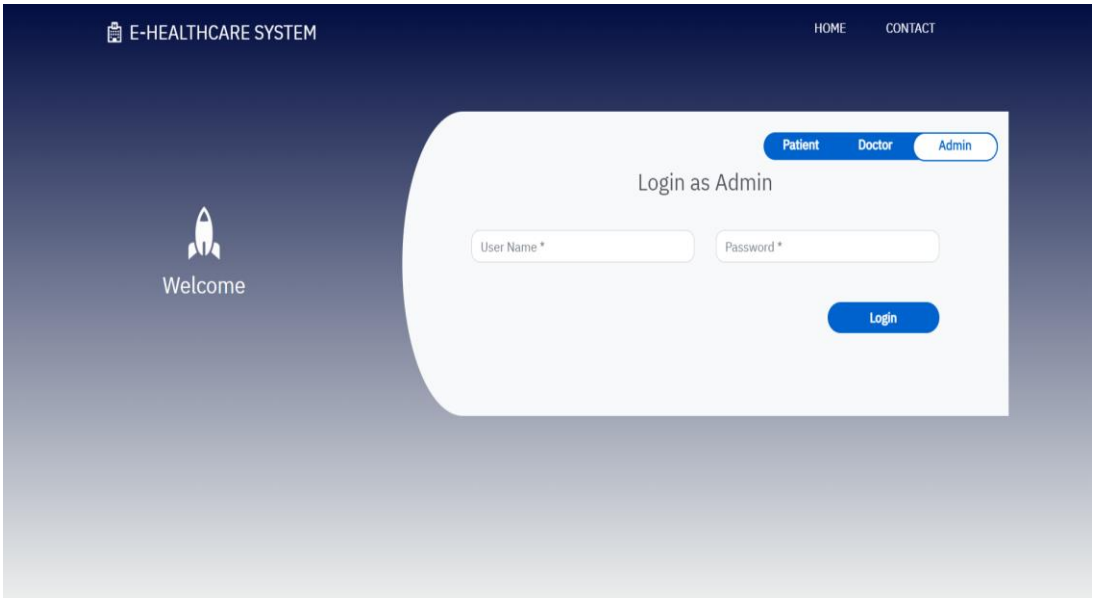
Doctor Login



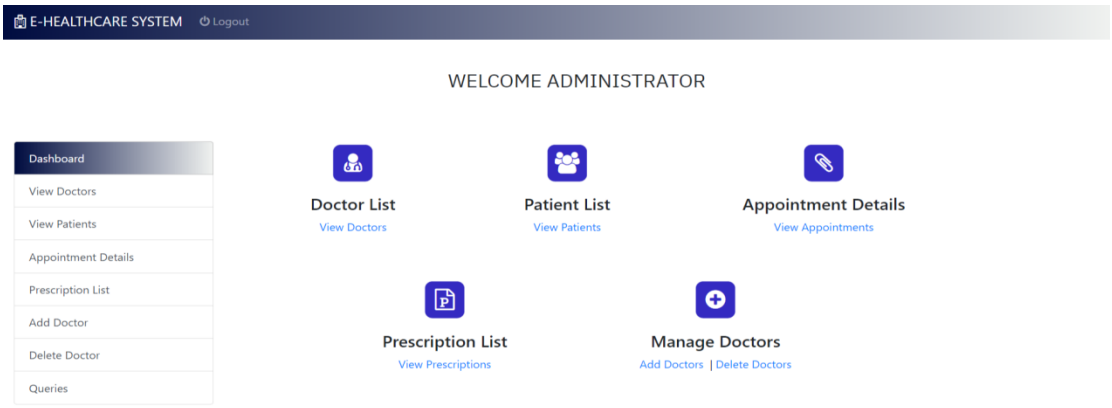
Doctor Dashboard



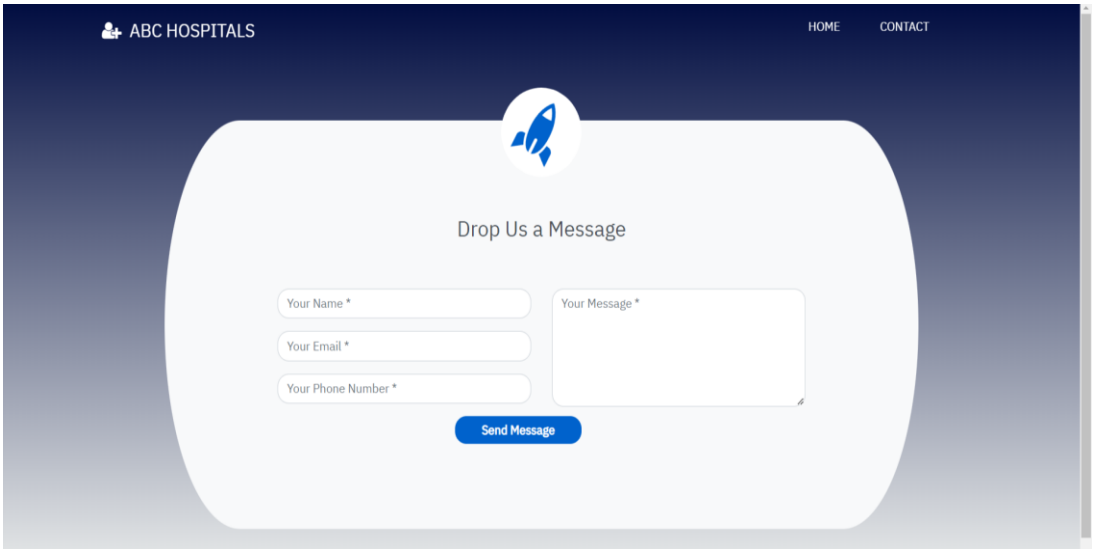
Admin Login



Admin Dashboard



Contact us page



CHAPTER-5

CONCLUSION

Since we are entering details electronically in the " E-Healthcare System", data will be secured. Using this application we can retrieve patient's history with a single click. Thus processing information will be faster. It guarantees accurate maintenance of Patient details. It easily reduces the book keeping task and thus reduces the human effort and increases accuracy speed.

Working on the project was an excellent experience. It helped us to understand the importance of planning, designing and implementation so far we have learnt in our theory books. It helped us unleashing our creativity while working in a team. It also realized the importance of team working, communication as a part of this project. The project was successfully completed after a lot of efforts and work hours. This project underwent number of compiling, debugging, removing errors, making it bug free, adding more facilities in E-Healthcare System and interactivity making it more reliable and useful.

Finally, we like to conclude that we put all our efforts throughout the development of our project and tried to fulfill most of the requirements of the user.

REFERENCES/BIBLOGRAPHY