

E-MEDICAL CARE APPLICATION

BY

ANWAR HASAN SHUVO

ID: 151-15-5506

AND

S.M FAHIM FOYSAL RABBY

ID: 151-15-5127

This Report Presented in Partial Fulfillment of the Requirements for the Degree
of Bachelor of Science in Computer Science and Engineering

Supervised By

Abdus Sattar

Senior Lecturer

Department of CSE

Daffodil International University

Co-Supervised By

Mohshi Masnad

Lecturer

Department of CSE

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

10TH DECEMBER, 2018

APPROVAL

This Project titled “**E-Medical Care**”, submitted by Anwar Hasan Shuvo and S.M Fahim Foysal Rabby to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering (BSc) and approved as to its style and contents. The presentation has been held on 10th December, 2018.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain

Professor and Head

Department of CSE

Faculty of Science & Information Technology

Daffodil International University

Chairman

Dr. Sheak Rashed Haider Noori

Associate Professor and Associate Head

Department of CSE

Faculty of Science & Information Technology

Daffodil International University

Internal Examiner

Zahid Hasan

Assistant Professor & Associate Head

Department of CSE

Faculty of Science & Information Technology

Daffodil International University

Internal Examiner

Dr. Mohammad Shorif Uddin

Professor

Department of Computer Science & Engineering

Jahangirnagar University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Abdus Sattar, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Abdus Sattar
Senior Lecturer
Department of CSE
Daffodil International University

Co- Supervised by:

Mohshi Masnad
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Anwar Hasan Shuvo
ID: 151-15-5506
Department of CSE
Daffodil International University

S.M Fahim Foysal Rabby
ID: 151-15-5127
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We are grateful and indebted to our respected supervisor Abdus Sattar, the lecturer of the department of Computer Science & Engineering in Daffodil International University, Dhaka. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would also like to express our heartiest gratitude to our co-supervisor Mohshi Masnad, the lecturer of the department of Computer Science & Engineering and Syed Akhter Hossain, the head of the department of Computer Science & Engineering and other faculty members who gave necessary suggestions to make a complete project.

Finally, we will never forget the help of our friends who always motivated us and gave their valuable opinions which assisted us a lot to make this project.

ABSTRACT

The project is all about patient record system. It is made for a single hospital. It contains each and every record of all operations that occur at hospital.

In a hospital, there are many doctors and they have many patients. In the system there are four levels of users and they are admin, patient, doctor & receptionist. The admin level encompasses the other level of users. The system enables registration of new doctors, patients.

All relevant information of doctors and patients will be saved in a database. In this system, there will be an admin who will create the profile of both patients and doctors. Doctors can keep the record of patients and make prescription in the web app. Records of patients will be stored in the database and all topical records will be shown. If a patient goes to another doctor, the doctor can see the past prescription and details of the patient. Every user can find blood donor in this application and request for donating blood. Patients can see a list of doctor's relevant information to find a doctor. Patients can check the information regarding test and its price.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
Table of Contents	v- vi
List of figures	vii-viii
List of tables	ix
CHAPTER	
CHAPTER 1: INTRODUCTION	1-3
1.1 Introduction	1
1.2 Motivation	1-2
1.3 Objectives	2
1.4 Expected Outcome	2
1.5 Report Layout	3
CHAPTER 2: BACKGROUND	4-6
2.1 Introduction	4
2.2 Related Work	4
2.3 Comparative Studies	4-5
2.4 Scope of the Problem	5
2.5 Challenges	5-6

CHAPTER 3: REQUIREMENT SPECIFICATION	7-13
3.1 Business Process Modeling	7
3.2 Requirement Collection and Analysis	7-8
3.3 Use Case Modeling and Description	8-12
3.4 Design Requirement	13
3.5 Logical Data Model	13
CHAPTER 4: DESIGN SPECIFICATION	14-20
4.1 Front-end Design	14
4.2 Back-end Design	14-15
4.3 Interaction Design and UX	15-19
4.4 Implementation Requirements	20
CHAPTER 5: IMPLEMENTATION AND TESTING	21-26
5.1 Implementation of Database	21-22
5.2 Implementation of Front-end Design	22-25
5.3 Test Results and Reports	25-26
CHAPTER 6: CONCLUSION AND FUTURE SCOPE	27
6.1 Discussion and Conclusion	27
6.2 Scope for Further Developments	27
APPENDIX	28
Appendix A: Project Reflection	28
REFERENCES	29
PLAGIARISM	30

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.1: Business process modeling	7
Figure 3.3.1: Use case of Admin	9
Figure 3.3.2: Use case of Receptionist	10
Figure 3.3.3: Use case of Doctor	11
Figure 3.3.4: Use case of Patient	12
Figure 3.5: Entity Relationship Diagram	13
Figure 4.3.1: Login UI	15
Figure 4.3.2: Dashboard for Admin	16
Figure 4.3.3: Dashboard of Receptionist	16
Figure 4.3.4: Dashboard of Doctor	17
Figure 4.3.5: Dashboard of Patient	17
Figure 4.3.6: Creation of Doctor Profile	18
Figure 4.3.7: Appointment	18
Figure 4.3.8: Invoice	18
Figure: 4.3.9: Add New Schedule	19
Figure 4.3.10: Add New Service	19
Figure 4.3.11: Patient Add	19
Figure 5.1.1: Doctor table	21
Figure 5.1.2: Patient table	21
Figure 5.1.3: Appointment table	21
Figure 5.1.4: Doctor details	21
Figure 5.1.5: Login table	21
Figure 5.1.6: Service table	21
Figure 5.1.7: Invoice table	22
Figure 5.1.8: Patient medical info table	22
Figure 5.2.1: Doctor List	22
Figure 5.2.2: Patient List	23
Figure 5.2.3: Prescription List	23

Figure 5.2.4: Payment List	23
Figure 5.2.5: Receptionist List	24
Figure 5.2.6: Service List	24
Figure 5.2.7: Schedule List	24
Figure 5.2.8: Appointment List	25
Figure 5.2.9: Invoice List	25

LIST OF TABLES

TABLES	PAGE NO
Table 3.1: Use case of Admin	9
Table 3.2: Use case of Receptionist	10
Table 3.3: Use case of Doctor	11
Table 3.4: Use case of Patient	12
Table 5.3.3: Unit Testing	26
Table 5.3.4: Login Testing	26

CHAPTER 1

Introduction

1.1 Introduction

Hospitals are very significant part of our life which provides medical facility to the people who are suffering from various diseases. It's has been a mandatory for the hospitals to keep track of its day-to-day activities and records of its patients, doctors which will keep the hospital digitalized and online-based.

It's a herculean task to keep the track of patients in papers and it is error prone. It's impractical and time consuming process. It is not financially sufficient.

So to eradicate these problems we have developed an application named "Smart Patient Record System".

Our main goal is to provide such a system where doctors can keep their patients all relevant information and check previous medical reports and make the system, provide a paper-less recording system.

1.2 Motivation

Every invention of modern science is a wonderful phenomenon. With the help of technology our life has been easier than before. Now-a-days people are leading internet based life. They want to do their every task with the help of internet. When a person becomes sick, he or she needs to find a doctor for medical checkup. Doctors make prescriptions by hand writing and don't keep any record of their patients. Sometimes, prescriptions are not readable which cause a problem.

Due to the increasing number of doctors, patients there can be some problems which are given below:

- 1.The information is very difficult to retrieve and to find particular information to find out about the patient's history
2. Several changes to information like patient details are difficult to update as paper work is related.

3. Manual process takes too time to store and it may result in incorrect information.

Because of these things patients and authority face some kinds of problem. We wanted this problem to be eradicated as soon as possible. We want to introduce a new application where every user will find it useful.

1.3 Objectives

Our main goal is to make a such an app by which both doctor and patient will find it useful to use. Both doctors and patients will be benefitted by using the application as it has some fantastic features.

A patient can check his/ her medical reports in the app, communicate with the doctor, make appointment for visiting a doctor.

A doctor can keep the record of his patients in a systemic way, prescribe his patients, give schedule for checking up of a patient.

Both doctor and patients can request a blood donor for donating blood.

It is hoped that the app will bring some solutions to the field of medical for both doctors and patients.

1.4 Expected Outcome

The expected outcomes of the project are given below:

1. It will be an easy way of storing medical reports.
2. It will be an online based hospital management system.
3. There will be a proper storage and access to retrieve the information.
4. In the proposed system, there will not be any redundant information anywhere.
5. Manual system will be removed.
6. The system should be easy to use.
7. Any type of information will be available when the user required.
8. It generates timely & accurate patient information and helps in clinical audit.

1.5 Report Layout

The project report contains six chapters. Summarization is given below:

The motivation, objectives and expected outcome have been discussed in details in chapter one.

The background of our project which is covered with related works of the application and discussion of the problem and challenges of the system is discussed in chapter two.

In chapter three there will be a segment named “Requirement Specification”. Business Processing Modeling, Requirement Collection and Analysis, Use Case Modeling, Logical Data Model and Design Requirements are manifested in this segment.

Chapter four will contain design specification. Front-end design, Back-end design, UX & Implementation requirements are illustrated here.

Entity relationship diagram of database, testing results and reports are exhibited in chapter five. Finally, chapter six contains the future scope of the development.

CHAPTER 2

Background Study

2.1 Introduction

A hospital is such a place where people go to for taking treatment. Sometimes people face some sort of problems. Majority of the task are manually done by the receptionist in the paper. A lot of papers are needed to be handled and taken care of.

Doctors need to remember the relevant reasons of previous check-up of a patient. As a result, sometimes they forget the reason and find it difficult to diagnosis the disease.

Though there are quite same related applications are found, they aren't usable to maintain a hospital. Most of the hospitals aren't online based. With the development of the technology the world has come closer. Online payment system isn't developed in any system in hospital management system.

2.2 Related works

When we decided to make this application, we looked through online and found some apps that are like ours but they aren't useful. We found have many bugs and there are many features missing in the application. We will combine all the features in one app and make it useful to the common people.

Some related works are mentioned below with a short detail:

- Dr. Pad: In this app, the only user is doctor. No other option is created for a patient. No communication system has been developed. A patient is to go the doctor physically and then the app will be used. UI isn't user friendly.
- Doctorola: In this app, there is no option created for doctor. A user can find hospital, blood donor and make appointment with a doctor but won't get any confirmation.
- IbnSina: A user can see the list doctors and make appointment.

2.3 Comparative Studies

We study for some similar features sites and found that they are not same which we are developing. Needless to say that there are a few applications related to a proper hospital management system.

In the present, manual system is followed in every medical hospital and this system has many demerits.

We build a simple and user friendly site for a doctor and a patient. The main goal of the project is to make the system online based and digitalized.

2.4 Scope of the problem

It is totally a new site which brings doctors and patients side by side by its features. We are to face problems in our every day of life. Similarly, we faced many problems while making the project which were cumbersome.

The scopes of the problems are discussed below:

2.4.1 An attractive site for both patients and doctors

The application is designed for making the hospital online based. So it's very important to make the User Interface user-friendly and efficient.

2.4.2 Data analysis

We can analyze the data of the users to make predictions, get number of death. We can also get which disease has the most impact on patient & which disease's patients come most.

2.4.3 Online Payment

A patient can pay to a doctor through the application. The transaction information will be kept hidden from outside of the world.

2.4.5 Privacy

Privacy of all users should be maintained strictly. It is important to check the transaction system cautiously and privately. It will only record the user name and his given password to be identify him/her next time. Those data will be also kept safe by admin. If it keeps track of user activities and records it often it will be a strong violation of security.

2.5 Challenges

In every sphere of life there are challenges but we are to overcome the challenges to make something good. The challenges we have faced are given below:

2.5.1 User friendly UI

A successful app means an app with better UI. We tried to keep it simple and easy.

2.5.2 Problem finding

As some related applications have already been made, we tried heart and soul to find the bugs and missing features that should have been in the system.

2.5.3 Real time database

We faced some problems to choose which database management system we will use to materialize the application.

2.5.4 Time management

It was one of the difficult parts to maintain. We always kept in our mind about the deadline. If we couldn't do it in time, it would be very dangerous for us. Thus we had to divide our time to develop the site part by part.

CHAPTER 3

Requirement Specification

3.1 Business Process Modeling

Business process modeling shows the business process like work flow. The business process modeling is given below:

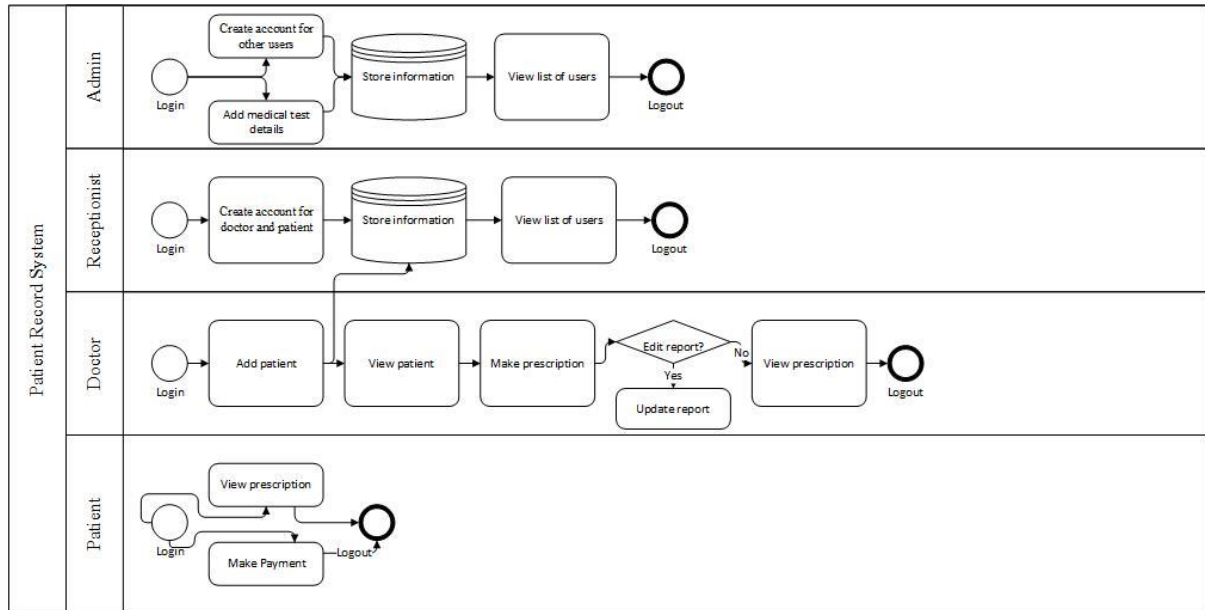


Figure 3.1: Business process modeling

3.2 Requirement Collection & Analysis

Requirement collection is one of the most important task to make an application. Without requirements development can be a herculean task. It's required to collect the requirements and the requirements are mentioned below:

1. Each user has to have login system.
2. Doctor can make prescription and give test using the application.
3. Patient can check their report in their profile.
4. Each user can search blood donator.
5. Normal user can search any doctor of the hospital.
6. Patient can make payment to book appointments using the application.

7. Receptionist can create account for patients and admin can create accounts for patient and doctors. Doctors also can add new patients.

These are the initial requirements based on the system has been developed.

3.2.1 Hardware & Software Requirement for System

First of all, It's mandatory to look after the minimum requirement of hardware and software to run it. There are some software requirements in addition to hardware to run this site properly and efficiently.

The following requirements are needed to visit the site:

1. Browser: Google Chrome, Mozilla Firebox (Recommended) or any browser which supports at least HTML 5.
2. RAM: Minimum 1 GB for pcs or less for mobile devices.
3. Operating System: Windows, Linux, Android etc.

3.2.2 Analysis

We have the confidence to develop this site completely. We studied a lot of sites which are related to our application. we decided to do that as PHP and MySQL in backend.

We decided to do backend in PHP, because PHP is a server-side scripting language for developing an online application and that is peaceful to be developed. PHP is prominent for the back-end.

3.3 Use Case Modeling & Description

Use-case diagram manifests the actions of a user. The diagrams are given below:

3.3.1 Use-case of Admin

The use case diagram of the admin is given below:

TABLE 3.1: Use case of Admin

Use case ID	3.1
Use case name	Admin.
Description	Admin will manage most of the things.
Pre-conditions	Admin needs to have an ID and password.
Standard flow	<ol style="list-style-type: none"> 1. Create and delete account of doctor, patient, receptionist. 2. View list of all users. 3. Search for blood donors 4. Seek for specialist doctors. 5. View medical test's price. 6. View and edit profile

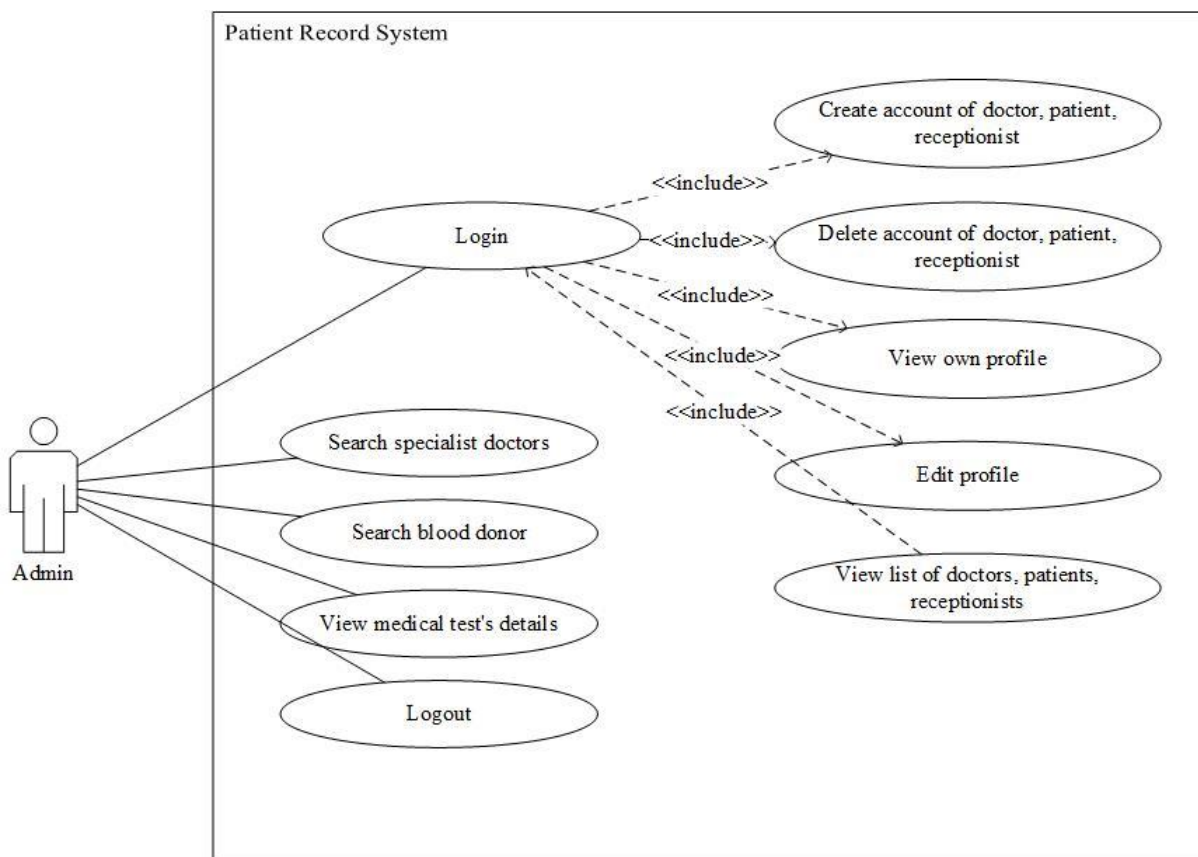


Figure 3.3.1: Use case of Admin

3.3.2 Use-case of Receptionist

TABLE 3.2: Use case of Receptionist

Use case ID	3.2
Use case name	Receptionist
Description	Receptionist is a user whose responsibility is to create the account of doctors and patients and help them.
Pre-conditions	Receptionist needs to have an ID and password.
Standard flow	<ol style="list-style-type: none"> 1. Create and delete account of doctor, patient. 2. Search for blood donors 3. Seek for specialist doctors. 4. Add medical test's details. 5. View medical test's price. 6. View and edit profile

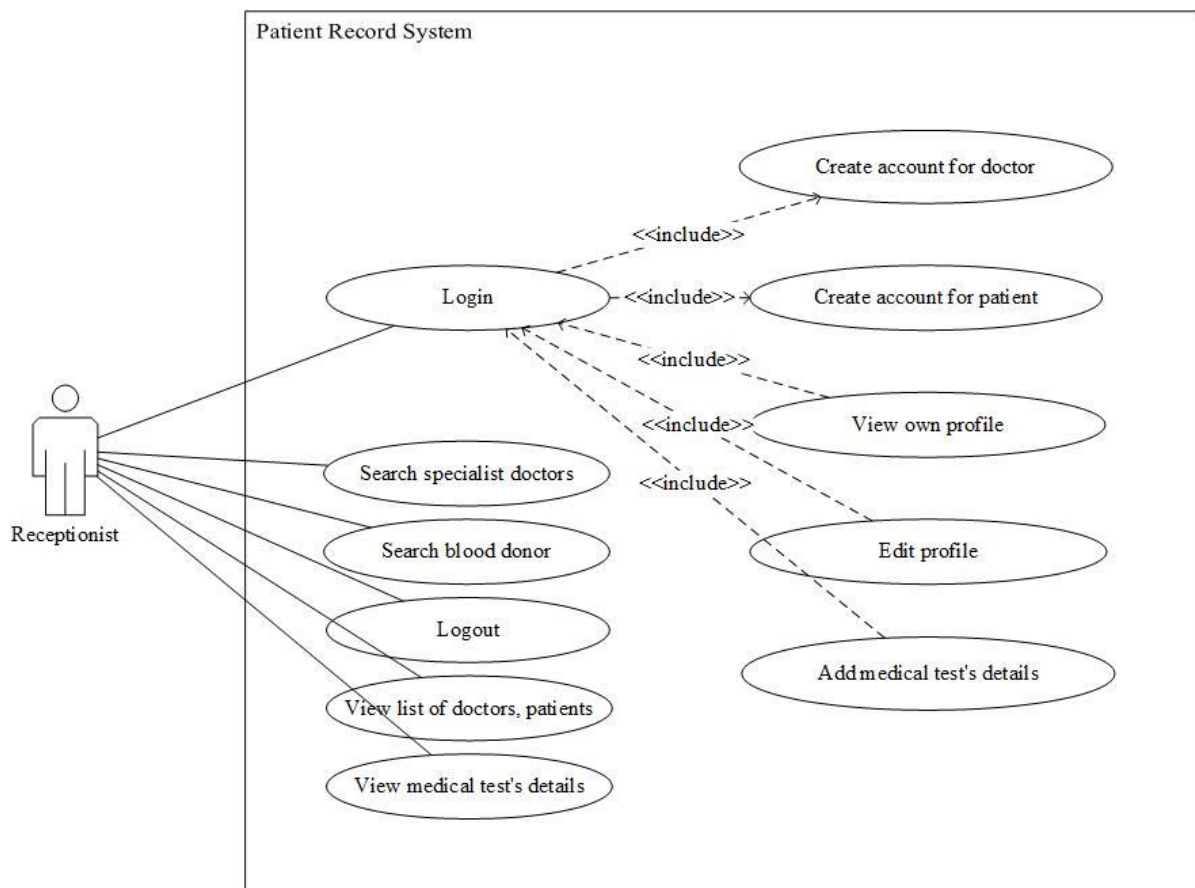


Figure 3.3.2: Use case of Receptionist

3.3.3 Use-case of Doctor

TABLE 3.3: Use case of Doctor

Use case ID	3.3
Use case name	Doctor
Description	Doctor is a user of the system who will interact with the patient.
Pre-conditions	Doctor needs to have an ID and password.
Standard flow	<ol style="list-style-type: none"> 1. Add patients. 2. Make prescription. 3. Update report. 4. View prescription. 5. View previous medical reports. 6. View and edit profile

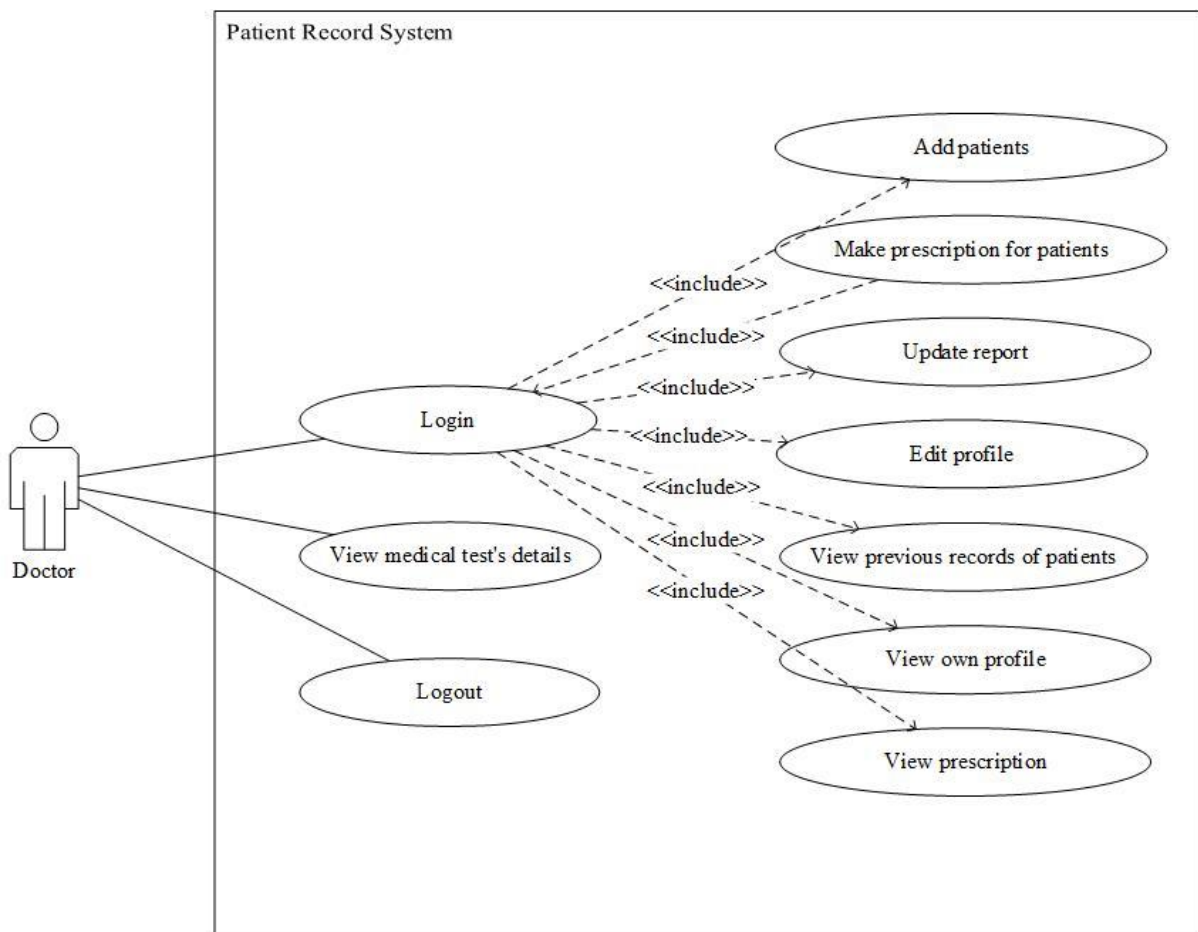


Figure 3.3.3: Use case of Doctor

3.3.3 Use-case of Patient

TABLE 3.4: Use case of Patient

Use case ID	3.4
Use case name	Patient
Description	Patient is a user of the system who will interact with the patient.
Pre-conditions	Patient needs to have an ID and password.
Standard flow	<ol style="list-style-type: none"> 1. View prescription. 2. Make payment. 3. Search for blood donors 4. Seek for specialist doctors. 5. View medical test's details. 6. View and edit profile

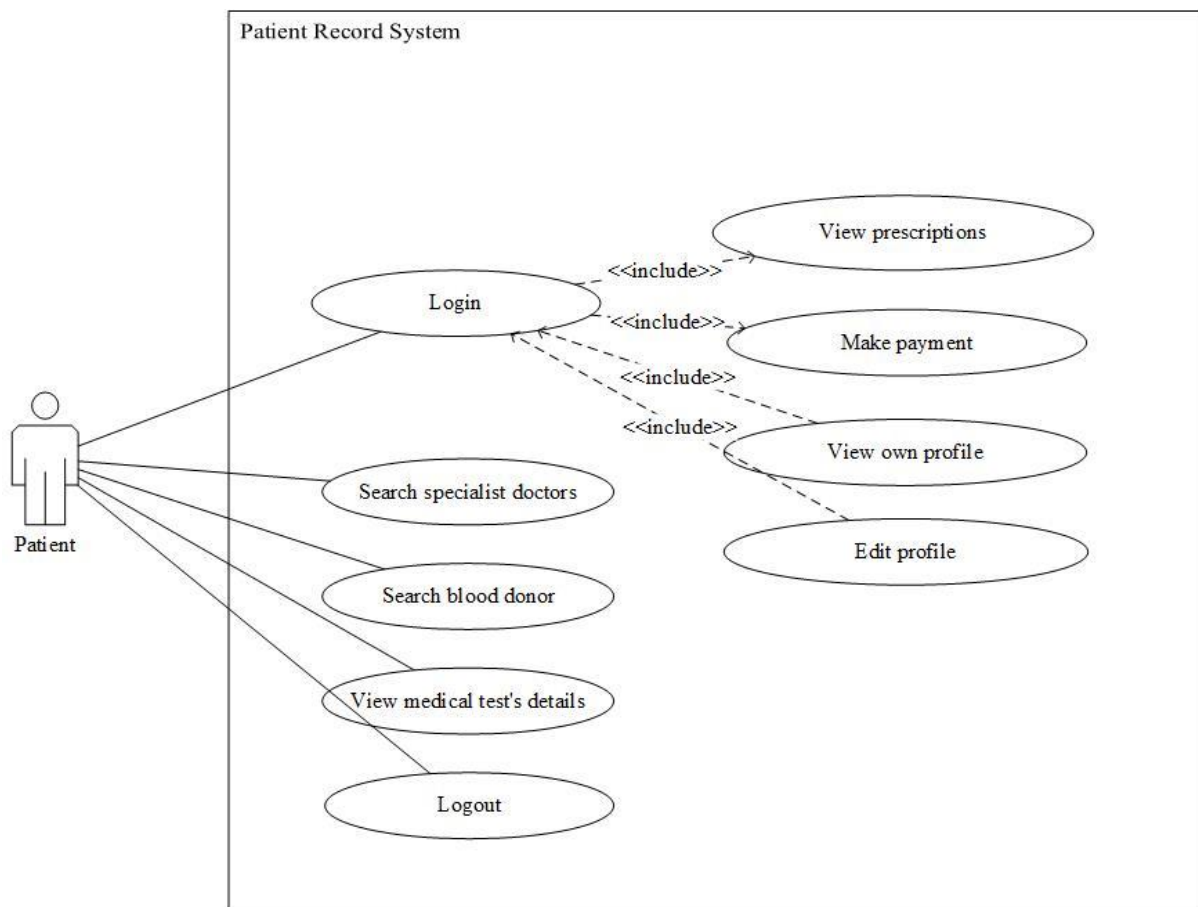


Figure 3.3.4: Use case of Patient

3.4 Design Requirements

Designing the User Interface is one of the most important tasks for making the site spanning.

The design requirements are given below:

1. User-friendly site.
2. Easy way of making prescription.
3. Easy of showing medical test's details.
4. Pleasant Dashboard.

3.5 Logical Data Model

Logical Data Model is a procedure which is used to identify and analyses requirement needed to support the business processes within the scope of corresponding information systems in organizations. The Entity-Relationship model or Entity-Relationship diagram is a logical data model, it includes the entity, attributes, table, and relationships.

Only the relationship among entities are shown here for simplicity.

The entity-relationship diagram is given below:

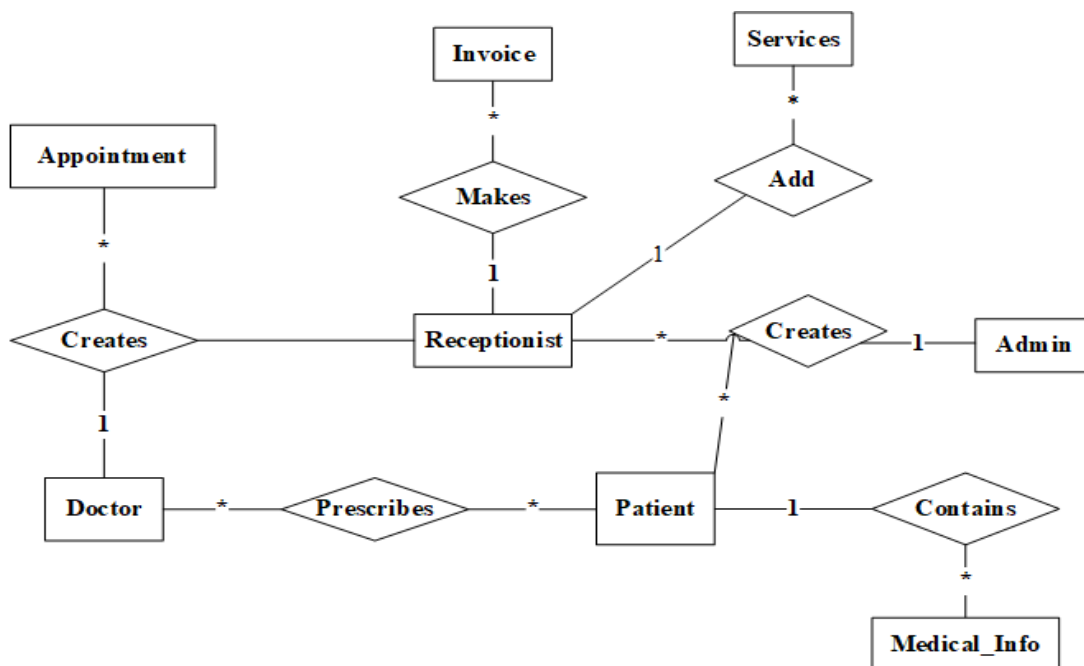


Figure 3.5: Entity Relationship Diagram

CHAPTER 4

Design Specification

4.1 Front-end Design

Front-end design defines the user interface design. Creating the main HTML, CSS and presentational JavaScript code that makes up a user interface. Design will play a vital role for any site. The designs need to be easy and user friendly. Front-end design should be simple which can be easily understood by anyone who will visit the site. Bootstrap, CSS, HTML and JavaScript have been used to design the pages.

4.1.1 CSS

CSS stands for Cascading Style Sheet. It is used for describing the style sheet of an HTML document which is written in a markup language. CSS describes how HTML elements should be displayed. We use CSS in our application to enable the separation of presentation and content including aspects such as layout, colors, margin, padding, font-size, border and background etc.

4.1.2 HTML

HTML is the standard markup language for creating web pages. HTML elements are represented by tags. We have used HTML for creating web page's elements that cannot be represented in plain text such as Heading, Paragraph, Table, Form, Stylized text, and so on.

4.1.3 Bootstrap

Bootstrap has been used to make the website responsive so that It can be fit to any screen. It helps to reload the pages fast.

4.2 Back-end Design

PHP and MySQL have been used in the project. The details are given below:

4.2.1 PHP

PHP is always distinguished as a back end. It's a server side scripting language. It can be embedded with HTML. PHP along with MySQL is used to make the website dynamic. It's compatible with most of the servers. We have used the framework named "Codeigniter" in the project.

4.2.2 MySQL

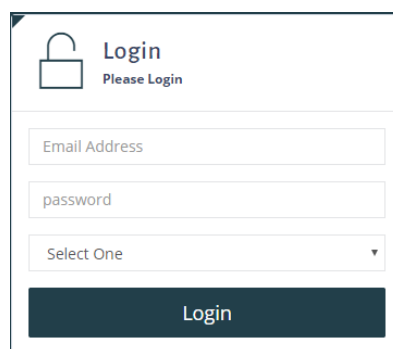
MySQL is one of the most distinguished open source database. It's a relational database. It provides us high quality of performance. We have used MySQL database in our project. Every data will be saved in the database.

4.3 Interaction Design & UX

User Interface (UI) design is the mode of making interfaces in software or computerized devices with a focus on looks or style. UI design usually refers to graphical user interfaces. It is the most important part for any site for ensuring better user experience.

4.3.1 Login Page

When the user of any type will browse the site, he or she will be redirected to this page for login.

The image shows a login form with a white background and a dark blue header. The header contains a lock icon, the text "Login", and "Please Login" below it. The form has three input fields: "Email Address", "password", and a dropdown menu labeled "Select One". At the bottom is a dark blue button with the text "Login".

Login	
Please Login	
Email Address	<input type="text"/>
password	<input type="password"/>
Select One	<input type="text"/>
Login	

Figure 4.3.1: Login UI

4.3.2 Dashboard for Admin

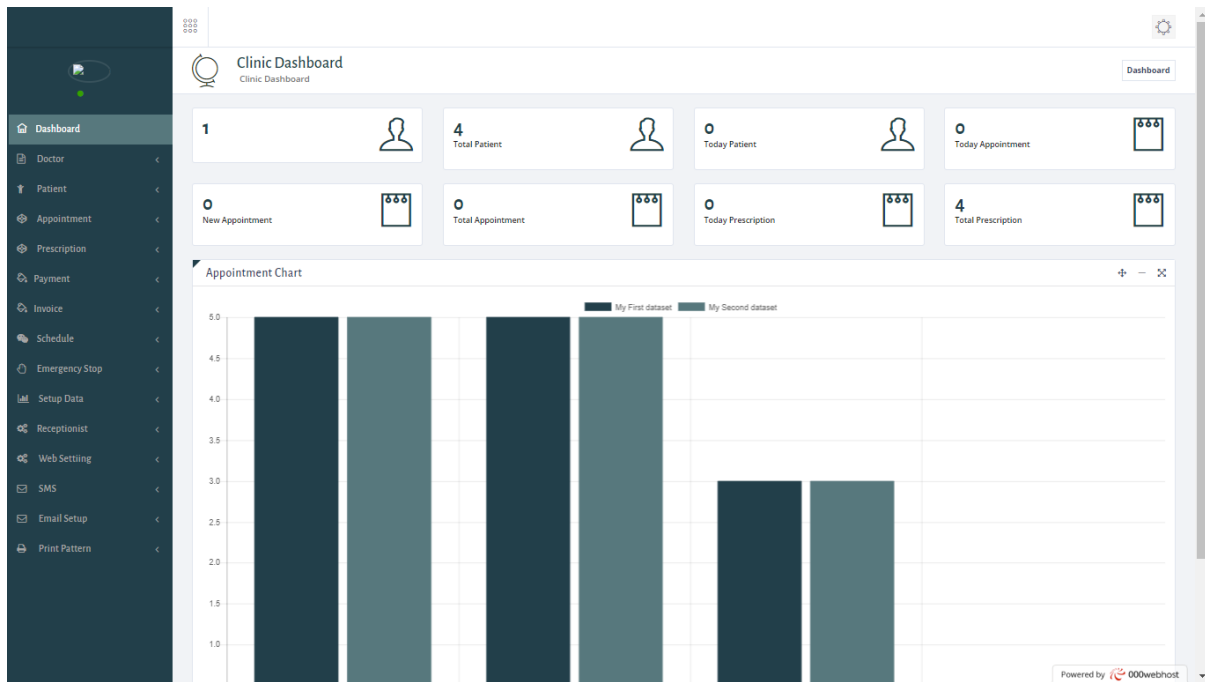


Figure 4.3.2: Dashboard for Admin

4.3.3 Dashboard for Receptionist

The Receptionist Dashboard, titled "Create Appointment", features a dark sidebar with navigation options: Patient, Appointment, and Invoice. The main content area displays a form for creating an appointment:

Add New Patient

Phone:

Date:

Doctor:

Choose Serial:

Note:

Developed By SPRS
/user/appointment_controller#

Powered by 000webhost

Figure 4.3.3: Dashboard of Receptionist

4.3.4 Dashboard for Doctor

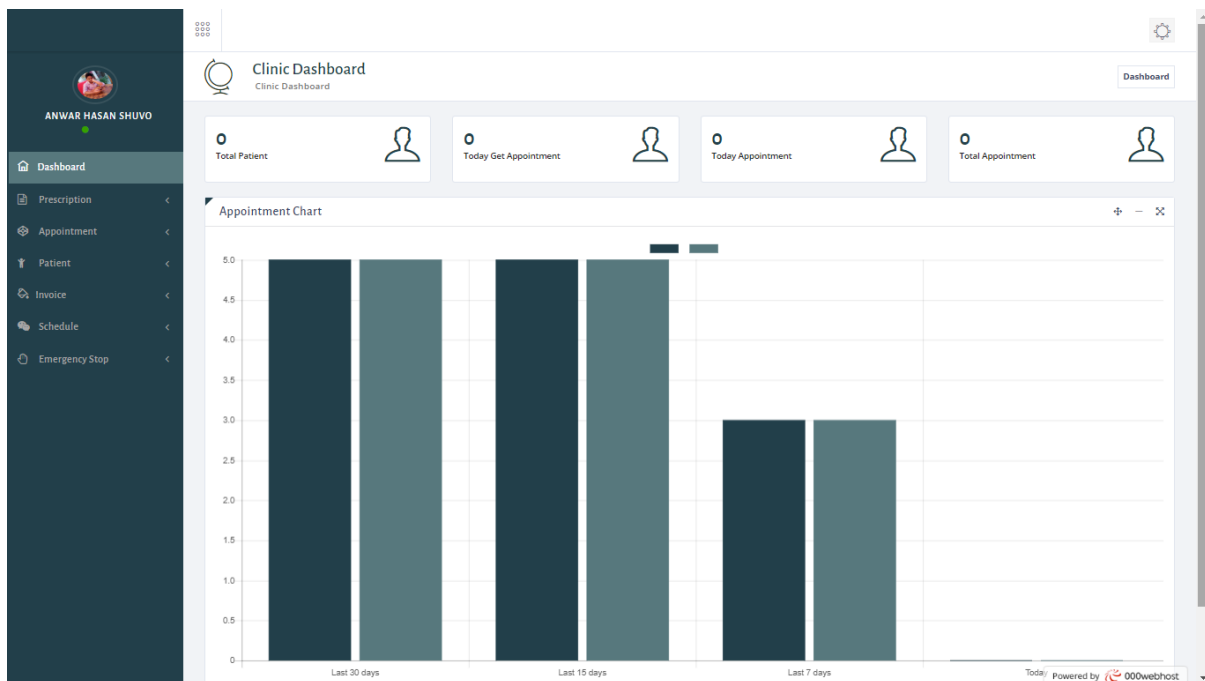


Figure 4.3.4: Dashboard of Doctor

4.3.5 Dashboard for Patient

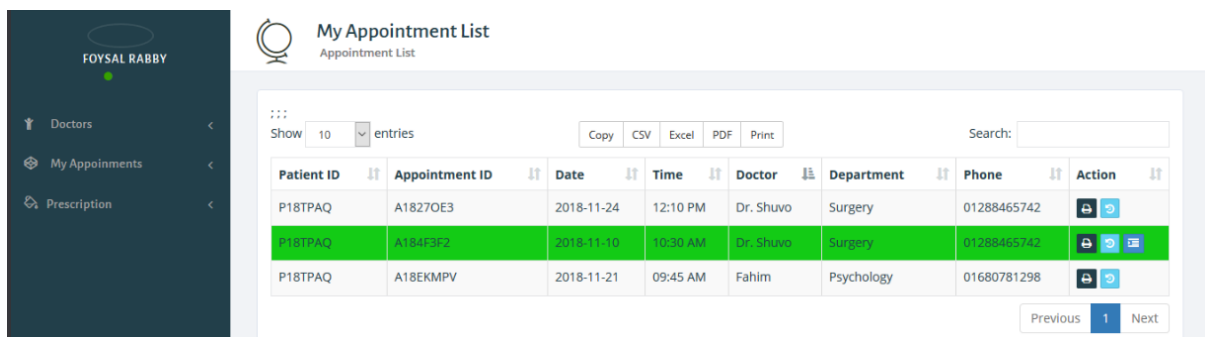


Figure 4.3.5: Creation of Doctor Profile

4.3.6 Doctor creation

The 'Create Doctor' form contains the following fields:

- Doctor Name
- Email Address
- password
- Designation
- Degrees
- Department
- Specialist
- Doctor Experience
- Service Place
- Birth date (format: yyyy-mm-dd)
- Phone Number
- Gender: Male, Female
- Blood Group (dropdown: --Select Blood Group--)
- Address
- About Me
- Picture: No file chosen

Figure 4.3.6: Creation of Doctor Profile

4.3.7 Appointment

The 'Add New Patient' form contains the following fields:

- Phone
- Date: yyyy-mm-dd
- Doctor: --Select doctor--
- Choose Serial
- Note

Figure 4.3.7: Appointment

4.3.8 Invoice

The 'New Invoice' form contains the following fields:

- Phone Number
- Patient Name
- Address
- Date: 2019-11-16
- Doctor: Select Doctor

Service Info	Quantity	Price	Discount	Total	Action
	1	0.00	0.00	0.00	<input type="button" value="Delete"/>
				Total Tax:	0.00
				Grand Total:	0.00
				Paid Amount:	0.00
				Due:	0.00

Figure 4.3.8: Invoice

4.3.9 Add New Schedule

Schedule

Select Day : Monday Tuesday Wednesday Thursday Friday Saturday Sunday

Set Time : Start Time to End Time

Per Patient Time :
You can set only minute

Visibility : Yes No

Figure: 4.3.9: Add New Schedule

4.3.10 Add New Service

Service List

Service Name

Price

Description

Tex Tax %

Service Model

Figure 4.3.10: Add New Service

4.3.11 Patient Add

Add New Patient

Add New Patient

Title Family Name Given name

Gender Male Female Others Birth date

Email Address Phone Number Mobile Number

Address Suburb

Post Code Blood Group Picture

Emergency Contact

Title Family Name Given name

Phone Number Mobile Number

Medical Information

Do you have allergies to any medicine or food? Do you have a tendency to bleed or bruise easily?

Please select illness, as following:

Heart Diseases High Blood Pressure Any Accidents

Diabetic Any Surgeries Others

Do you Consider yourself to be in a high risk group for infectious diseases?

Please list any relevant family medical history and social history

Please list your current medical conditions and medications

Figure 4.3.11: Patient Add

4.4 Implementation Requirements

Implementation means doing a work by a planning to give a visual look. The site which comes to the mind first is making it visible. The primary thought was to make the site easier for everyone.

We needed the following things to implement the application:

1. php
2. JavaScript
3. HTML & CSS
4. MySQL

The following software used to implement the application:

1. XAMPP
2. Notepad++
3. Php Storm (IDE)

CHAPTER 5

Implementation & Testing

5.1 Implementation of Database

The tables of database are shown below:

Column	Type
doctor_id	int(11)
log_id	int(11)
doctor_name	varchar(120)
department	text
designation	varchar(120)
degrees	text
specialist	varchar(250)
doctor_exp	text
birth_date	date
sex	varchar(50)
blood_group	varchar(50)
doctor_phone	varchar(15)
address	varchar(255)
service_place	text
about_me	text
picture	varchar(100)
doctor_status	int(11)

Figure 5.1.1: Doctor table

Column	Type
patient_id	varchar(250)
family_name	text
given_name	text
patient_email	varchar(120)
patient_phone	varchar(15)
mobile_number	varchar(222)
title	text
post_code	varchar(210)
suburb	text
address	text
sex	varchar(120)
birth_date	date
blood_group	varchar(250)
picture	varchar(250)
emg_title	text
emg_family_name	text
emg_given_name	text
emg_phone	varchar(210)
emg_mobile	varchar(210)
create_by	int(11)
create_date	datetime

Figure 5.1.2: Patient table

Column	Type
id	int(11)
appointment_id	varchar(250)
patient_id	varchar(250)
phone_number	varchar(120)
doctor_id	int(11)
schedul_id	int(11)
sequence	varchar(100)
problem	varchar(255)
get_date_time	datetime
get_by	int(11)
date	date
status	int(11)

Figure 5.1.3: Appointment table

Column	Type
doctor_details_id	int(11)
doctor_id	int(11)
doctor_short_bio	text
doctor_details_bio	text
academic_info	text
work_experience	text

Figure 5.1.4: Doctor Details

Column	Type
log_id	int(11)
email	varchar(100)
password	varchar(100)
user_type	int(11)
active_status	int(11)
logout_time	varchar(120)

Figure 5.1.5: Login table

Column	Type
id	int(11)
service_name	text
service_price	int(11)
description	text
tax	int(11)
model	varchar(122)

Figure 5.1.6: Service table

Column	Type
invoice_id	int(11)
doctor_id	int(11)
patient_id	varchar(111)
grand_total	int(11)
total_tax	int(11)
paid	int(11)
due	int(11)
payment_notes	text
payment_method	varchar(250)
payment_method_notes	text
date_time	datetime

Figure 5.1.7: Invoice table

Column	Type
pmi_id	int(11)
patient_id	text
dob	varchar(200)
food_allergies	text
bleed_tendency	text
heart_disease	text
diabetic	text
high_blood_pressure	text
surgeries	text
accidents	text
others	text
high_risk_diseases	text
family_history	text
current_medication	text
female_pregnant	text
female_breast_feeding	text
others_insurance	text
others_comcare	text
others_tac	text
others_low_income	text
others_reffer	text
subscription	text

Figure 5.1.8: Patient medical info table

5.2 Implementation of Front End Design

The implementations of the front end design are given below:

5.2.1 Doctor List

When the profile of doctors is created, the list of doctors can be shown.



Picture	Doctor Name	Email Address	Action
	Anwar Hasan Shuvo	shuvo@doctor.com	

Figure 5.2.1: Doctor List





















5.2.2 Patient List

Patient List

Show 10 entries

Copy CSV Excel PDF Print

Search:

Picture	Patient Id	Name(Family & Given)	Phone Number	Birth date	Action
	P187QXE	Fahim Foyzal	01776419791	1996-11-04	  
	P188C6F	Ahnaf Ibad Bhuiyan	01515604204	2015-02-21	  
	P18F74G	ff r	01776419790	2010-11-02	  
	P18OLR1	namewala asdf	01712234130	2005-11-02	  
	P18TPAQ	Foyzal Rabby name	01748070593	2015-11-03	  

Previous 1 Next

Figure 5.2.2: Patient List

5.2.3 Prescription List

Prescription List

Show 10 entries

Copy CSV Excel PDF Print

Search:





Picture	Patient Name	Patient Id	Phone Number	Gender	Action
	ff r	P18F74G	01776419790	Male	
	namewala asdf	P18OLR1	01712234130	Female	

Figure 5.2.3: Prescription List

5.2.4 Payment List

Payment List

Show 10 entries

Copy CSV Excel PDF Print

Search:

Patient Id	Appointment Id	Email Address	Amount	Date	Notes	Action
No data available in table						

Previous Next

Figure 5.2.4: Payment List

5.2.5 Receptionist List

All Receptionist

Show 10 entries Copy CSV Excel PDF Print Search:

Picture	Full Name	Phone Number	Blood Group	Address	Birth date	Gender	Action
	Fahim Foysal	01234567890	B-	31, B street	2010-11-02	Male	

Previous 1 Next

Figure 5.2.5: Receptionist List

5.2.6 Service List

+ Add Service

Show 10 entries Copy CSV Excel PDF Print Search:

SL NO	Service Name	Price	Tax	Description	Service Model	Action
1	Ambulance	1000	50	Not free ambulance available.	1	
2	X-ray	300	6	Digital Xray	Digital	
3	MRI	500	10	Digital MRI	Digital	
4	New AService	50	3	no description	3	

Previous 1 Next

Figure 5.2.6: Service List

5.2.7 Schedule List

Schedule List Dashboard

Show 10 entries Copy CSV Excel PDF Print Search:

#SL	Doctor Name	Start Time	End Time	Per Patient Time	Action
1	Fahim	08:00:00	12:00:00	15	
2	Fahim	08:00:00	12:00:00	15	
3	Fahim	08:00:00	12:00:00	15	
4	Fahim	08:00:00	12:00:00	15	

Figure 5.2.7: Schedule List

5.2.8 Appointment List

Patient Name	Phone Number	Appointment Id	Appointment Time	Date	Action
namewala asdf	01712234130	A1BRQ8L9	12:10 PM	2018-11-10	[Icons]
Foysal Rabby name	01748070593	A184F3F2	10:30 AM	2018-11-10	[Icons]
namewala asdf	01712234130	A180AFAE	02:10 PM	2018-11-24	[Icons]
Foysal Rabby name	01748070593	A18270E3	12:10 PM	2018-11-24	[Icons]

Figure 5.2.8: Appointment List

5.2.9 Invoice List

SL NO	Invoice Id	Patient Name	Grand Total	Due	Date	Action
1	1	ff r	1050	0 1050	2018-11-02 00:00:00	[Icons]
2	2	namewala asdf	110	0 110	2018-11-03 00:00:00	[Icons]

Figure 5.2.9: Invoice List

5.3 Test Results and Reports

There are some testing procedures to test the entire system whether it works correctly. Some tests are given below to manifests that the system is fully ready to be run.

5.3.1 System Test

Each and every page has been checked that it works correctly or not. It doesn't produce any error. The system has no bug at all.

5.3.2 Acceptance Testing

The acceptance test cases are executed against the test case data to check that we are getting the actual outcome or not.

5.3.3 Unit Test

Every single unit has been tested while implanting the system. It has helped us to complete the project with some comfort. It's a part of a software testing.

An example of unit test is mentioned below:

TABLE 5.3.3: Unit Testing

Test Case	Expected Result	Actual Result	Status
Correct Email & Correct Password	Login to the system	Login to the system	Pass
Incorrect Email & Incorrect Password	Incorrect email or password	Incorrect email or password	Pass
Correct Email & Incorrect Password	Incorrect email or password	Login to the system	Fail

5.3.4 Test Case

Test cases actually determines whether the system satisfies the requirements.

TABLE 5.3.4: Login Testing

Test Case ID: 1	Test Designed by: Shuvo, Fahim
Test Title: Signing to the system	Test Execution Date: 18th November, 2018
Pre-conditions: User must have correct email and password	

Step	Test Step	Test Data	Expected Result	Actual Result	Status
1	Go to the sign-in page				
2	Provide email	admin@gmail.com	Correct	Correct	Pass
3	Provide password	123456	Correct	Correct	Pass
4	Select user from combo box	Admin	Correct	Correct	Pass
5	Click on the login button	Clicked on the button	Successfully logged in	Successfully logged in	Pass

CHAPTER 6

Conclusion & Future Scope

6.1 Conclusion & Discussion

With all the limitations the features currently the project has adorned with the project will without a doubt help both users of the system to have their way around as necessity with the type of the work needs to be done.

Management systems are meant to be made human work easier. We hope the system will meet the requirement to ensure its productivity.

We have successfully completed the project. One of the most difficult parts was to make the site easy and eye catching for users.

At the beginning of the project we planned that how we will move forward to complete this project.

The project is a hearty work of all the members of the project team. We have come together and worked as a team to make our imagination into materialization. The project was developed part by part and at last we blended them together as a complete site which we are trying for.

We faced some difficulties while doing the project but we overcame all the hindrances by our hard work, mental strength, patience.

6.2 Scope for further developments

The project is complete but there can be some features which can be added.

The system can be enhanced by adding more features which we will implement in the future. Online payment system will be added to the system. Doctor & Patient can make appointment by using it.

We will develop a mobile application in the future so that users can find it more useful. At the very beginning we will develop the app for the patients so that they can view their prescription, search specialist doctors, blood donors etc. the features which are available.

APPENDIX

APPENDIX A: Project Reflection

The goal of the appendix is to give an introduction to the reflection of the project. We have enjoyed a lot while doing the project works. It has been an excellent journey. Without the help of each other we wouldn't be able to complete the project in time.

We met together on meeting and discussed about the project sitting together next to each other. We also discussed new ideas to make the system fantastic.

References:

- [1] UML Standards: <https://www.draw.io/> / Last Accessed on 10-September-2018
- [2] Scribd <https://www.scribd.com/document/324632985/Hospital-Management-System-Project-Report> / Last Accessed on 12-September-2018
- [3] BPMN <https://www.process.st/bpmn-tutorial/> / Last Accessed on 12-September-2018
- [4] UML Diagram <https://www.uml-diagrams.org/use-case-diagrams.html> / Last Accessed on 18-September-2018
- [5] Kiasap, Hospital Management System <https://www.scribd.com/doc/86464560/Hospital-Management-System> / Last Accessed on 23 -September-2018
- [6] Sumit Tembhare, Hospital Management System, <https://www.scribd.com/document/99440051/Final-Hospital-Management-System> / Last Accessed on 27 -September-2018

Plagiarism:

11/13/2018

Turnitin

Document Viewer

Turnitin Originality Report

Processed on: 13-Nov-2018 09:13 +06
ID: 1037881031
Word Count: 3746
Submitted: 1

151-15-5506 By Anwar Hasan
Shuvo

Similarity Index
19%

Similarity by Source

Internet Sources:	16%
Publications:	2%
Student Papers:	17%

[include quoted](#) [include bibliography](#) [excluding matches < 1%](#) [download](#)
[refresh](#) [print](#) mode: [quickview \(classic\) report](#)

4% match (student papers from 05-Apr-2018) ✕
Class: Article 2018
Assignment: Journal Article
Paper ID: [941584057](#)

4% match (student papers from 05-Apr-2018) ✕
Class: Article 2018
Assignment: Journal Article
Paper ID: [941491107](#)

3% match (student papers from 09-Apr-2018) ✕
Class: Article 2018
Assignment: Journal Article
Paper ID: [943495148](#)

1% match (Internet from 06-Sep-2018) ✕
<http://dspace.daffodilvarsity.edu.bd:8080>

1% match (student papers from 03-Apr-2018) ✕
Class: Article 2018
Assignment: Journal Article
Paper ID: [940334293](#)

1% match (student papers from 07-Apr-2018) ✕
Class: Article 2018
Assignment: Journal Article
Paper ID: [942529588](#)

1% match (student papers from 07-Apr-2018) ✕
Class: Article 2018
Assignment: Journal Article

https://www.turnitin.com/newreport_classic.asp?lang=en_us&oid=1037881031&ft=1&bypass_cv=1

1/8