Table of Contents

LIST OF FIGURES USED IN THE PROJECT

LIST OF TABLES USED IN THE PROJECT

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.

HMS 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.

HMS 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.

Benefits of implementing a hospital management 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

PROCESS MODEL

Hospital Management System follows <u>**INCREMENTAL MODEL**</u> because initially software requirements are reasonably well defined but the overall scope of development effort is a purely linear process. There may be other requirements of the user which will be known later. So, those requirements can the implemented and delivered in the following next increments. Our project is a short term project of 3 months and 3 weeks only and staffing available is also low (3 persons).

CHAPTER 1
INTRODUCTION

1.1 PURPOSE
1.2 SCOPE
1.3 DEFINITIONS, ACRONYMS, and ABBREVIATIONS
1.4 REFERENCES
1.5 OVERVIEW

1.1 PURPOSE

This software will help the company to be more efficient in registration of their patients and manage appointments, records of patients. It enables doctors and admin to view and modify appointments schedules if required. The purpose of this project is to computerize all details regarding patient details and hospital details.

1.2 SCOPE

The system will be used as the application that serves hospitals, clinic, dispensaries or other health institutions. The intention of the system is to increase the number of patients that can be treated and managed properly.

If the hospital management system is file based, management of the hospital has to put much effort on securing the files. They can be easily damaged by fire, insects and natural disasters. Also could be misplaced by losing data and information.

1.3 DEFINITIONS, ACRONYMS, and ABBREVIATIONS

1. **Cardiologist** - treats heart disease.
2. **Pediatrician** - treats infants, toddlers, children and teenagers.
3. **Plastic Surgeon** - restores, reconstructs, corrects or improves in the shape and appearance of damaged body structures, especially the face.
4. **Psychiatrist** - treats patients with mental and emotional disorders.
5. Ophthalmologist - treats eye defects, injuries, and diseases
6. **ENT**- Ear, Nose and Throat Specialist.

- **SRS**: Software Requirement Specification.
- **DFD:** Data Flow Diagram.
- **ENT**- Ear, Nose and Throat Specialist.
- **BG** - Blood group

- ✓ **Appt –** Appointment.
- ✓ **Sign up -** Creating New User.
- ✓ **Log in -** Logging in Existing User.
- ✓ **PhNo** - Mobile number.
- ✓ **Addr** – Address.
- ✓ **Expr** – Experience.

1.4 <u>REFERENCES</u>

- https://www.officetimeline.com/make-gantt-chart/excel
- https://medium.com/@datamateuaecrescent/hospital-management-system-features-objectives-62eeb13f4fc4
- R.S Pressman, Software Engineering: A Practitioner's Approach, Mc-Graw-Hill, Edition-7 (2010).
- P. Jalote, an Integrated Approach to Software Engineering, Narosa publication house, Edition -3 (2011).

1.5 <u>OVERVIEW</u>

Our application contains two modules – the admin module and the user module. Our application will not only help the admin to preview the monthly and/or yearly data but it will also allow them to edit, add or update records. The software will also help the admin to monitor the transactions made by the patients and generate confirmations for the same. The admin will be able to manage and update information about doctors.

The user module can be accessed by both the doctors and the patients. The doctor can confirm and/or cancel appointments. The doctors can even add prescriptions for their patients using our application. The patients will be able to apply for the appointment and make transaction for the same, and can even cancel appointments with the doctors. They can track details about the previous transactions made by them.

Advantages

- The system automates the manual procedure of managing hospital activities.
- Doctors can view their patients' treatment records and details easily.
- It even generates an instant bill.
- 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.

CHAPTER 2
SOFTWARE REQUIREMENT SPECIFICATION

2.1 Product Perspective

      2.1.1 System Interfaces

      2.1.2 System Specifications

          2.1.2.1 H/W Requirement

          2.1.2.2 S/W Requirement

          2.1.3 Communication Interfaces

2.2 Product functions

2.3 Data Flow Diagram (DFD)

      2.3.1 Context Level Diagram

      2.3.2 DFD Level – 1

      2.3.3 DFD Level – 2

2.4 Use Case Diagram

2.5 Use Case Description

2.6 User characteristics

2.7 Constraints

2.8 Assumptions and dependencies

2.1 Product Perspective

This Hospital Patient Info Management System is a self-contained system that manages activities of the hospital.

Due to improperly managed details medical center faces quite a lot of difficulties in accessing past data as well as managing present data. The fully functional automated hospital management system which will be developed through this project will eliminate the disadvantages caused by the manual system by improving the reliability, efficiency and performance. The usage of a database to store patient, employee, stock details etc. will accommodate easy access, retrieval, and search and manipulation of data. The access limitations provided through access privilege levels will enhance the security of the system. The system will facilitate concurrent access and convenient management of activities of the medical center.

2.1.1 System Interfaces

❖ *User Interfaces*
- This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.
- The **protocol used** shall be **HTTP**.
- The Port number used will be 80.
- There shall be logical address of the system in IPv4 format.

❖ *Hardware Interfaces*
- **Laptop/Desktop PC**-Purpose of this is to give information when Patients ask information about doctors, medicine available lab tests etc. To perform such Action it need very efficient computer otherwise due to that reason patients have to wait for a long time to get what they ask for.
- **Laser Printer (B/W)** - This device is for printing patients' info etc.
- **Wi-Fi router** - Wi-Fi router is used to for internetwork operations inside of a hospital and simply data transmission from pc's to sever.

❖ *Software Interfaces*
- **JDK 1.8** - Java is fast, secure, and reliable. From laptops to data centers, game consoles to scientific supercomputers, cell phones to the Internet,
- **Mysql server** - Database connectivity and management
- **OS Windows 7/8/8.1**- Very user friendly and common OS
- **JRE 1.8** - JAVA Runtime Environment for run Java Application and System

2.1.2 System Specifications

2.1.2.1 H/W Requirement
- ☞ Core i5 processor
- ☞ 2GB Ram.
- ☞ 20GB of hard disk space in terminal machines
- ☞ 1TB hard disk space in Server Machine

2.1.2.2 S/W Requirement
- ☞ Windows 7 or above operating system
- ☞ JRE 1.8
- ☞ Mysql server

### 2.1.3 Communication Interfaces

- ⌃ **NIC (Network Interface Card)** – It is a computer hardware component that allows a computer to connect to a network. NICs may be used for both wired and wireless connections.
- ⌃ **CAT 5 network cable**- for high signal integrity
- ⌃ **TCP/IP protocol**- Internet service provider to access and share information over the Internet
- ⌃ **Ethernet Communications Interface**- Ethernet is a frame-based computer network technology for local area networks (LANs)
- ⌃ Ubiquitous, easy to set up and easy to use. Low cost and high data transmission rate.

2.2 Product functions

- o Provide access to registered users only.
- o Registration of new patients.
- o Enable patient to view their record.
- o Enable patient to update their record.
- o Generate appointment date and timing.
- o Confirmation by doctor.
- o Patients can do Payment.
- o Modification in schedule by patient.
- o Admin access to patient's record.
- o Admin Verify Payment and Generate Bill/Receipt.
- o Admin can view monthly/yearly records.

2.3 DATA FLOW DIAGRAM (DFD)

CONTEXT LEVEL DIAGRAM
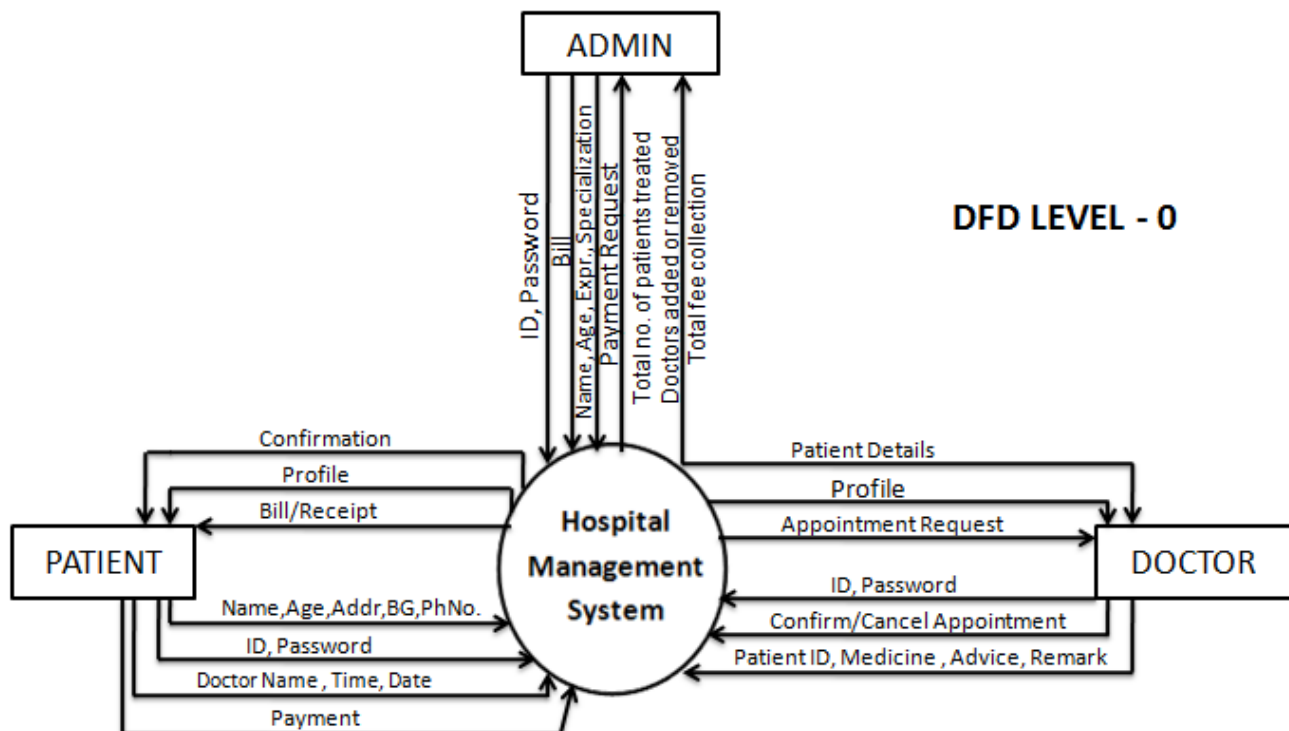
FIGURE 2.1 CONTEXT LEVEL DFD

DFD LEVEL – 1
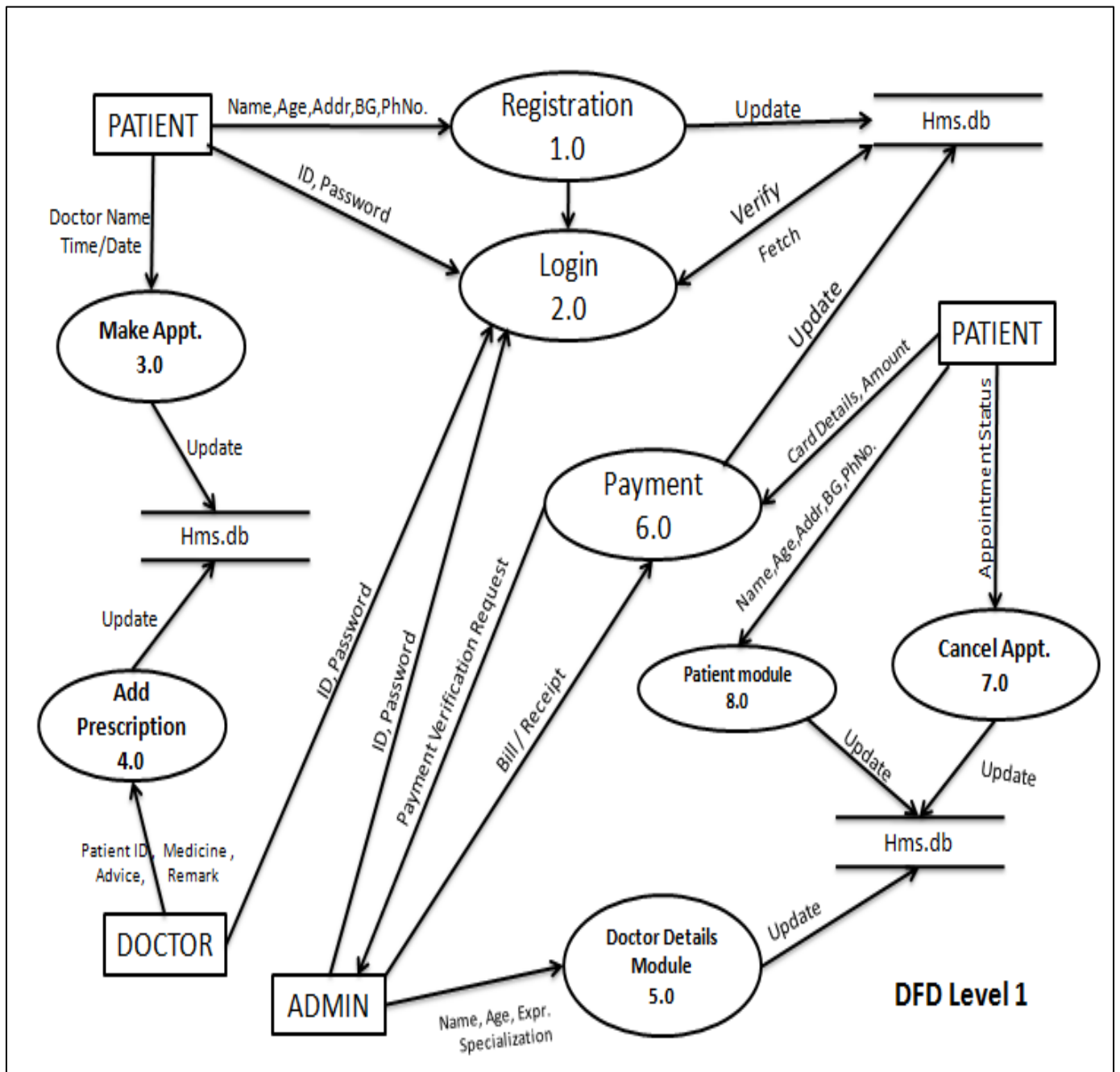
FIGURE 2.2 LEVEL – 1 DFD
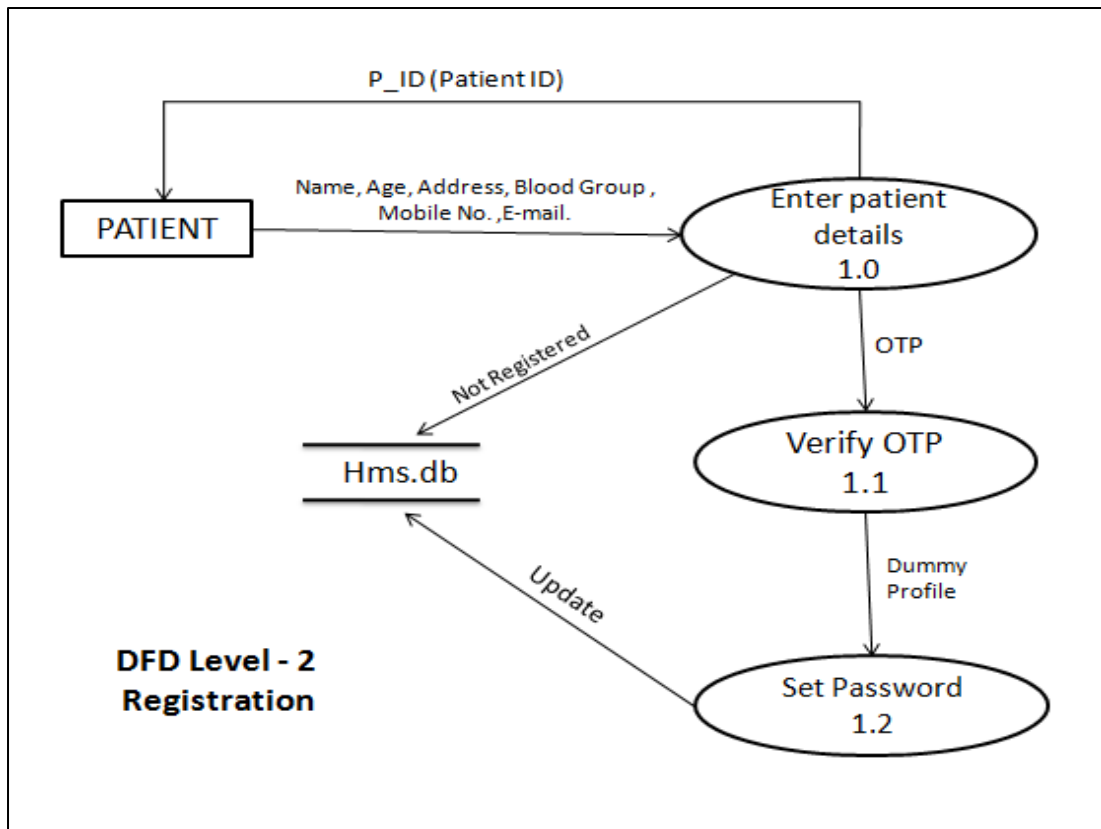
FIGURE 2.3 LEVEL – 2 Registration

FIGURE 2.4 LEVEL – 2 Login

FIGURE 2.5 LEVEL – 2 Make Appointment

FIGURE 2.6 LEVEL – 2 Add Description
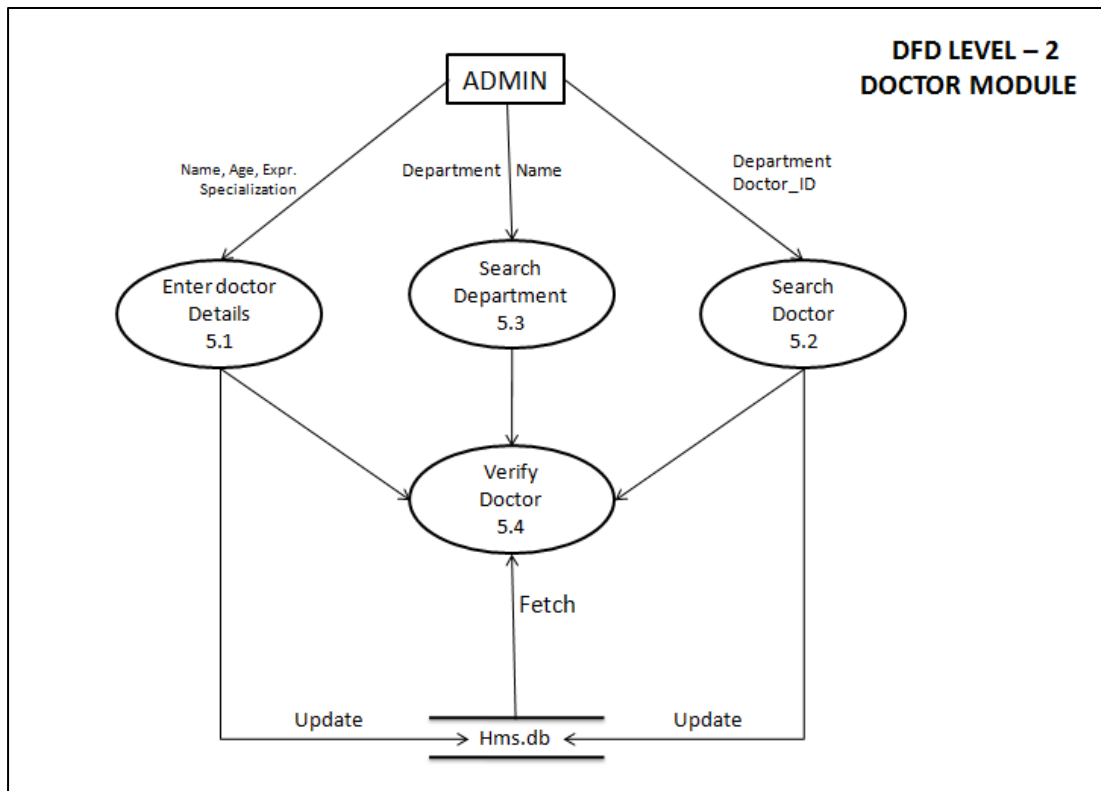
FIGURE 2.7 LEVEL – 2 Doctor Module
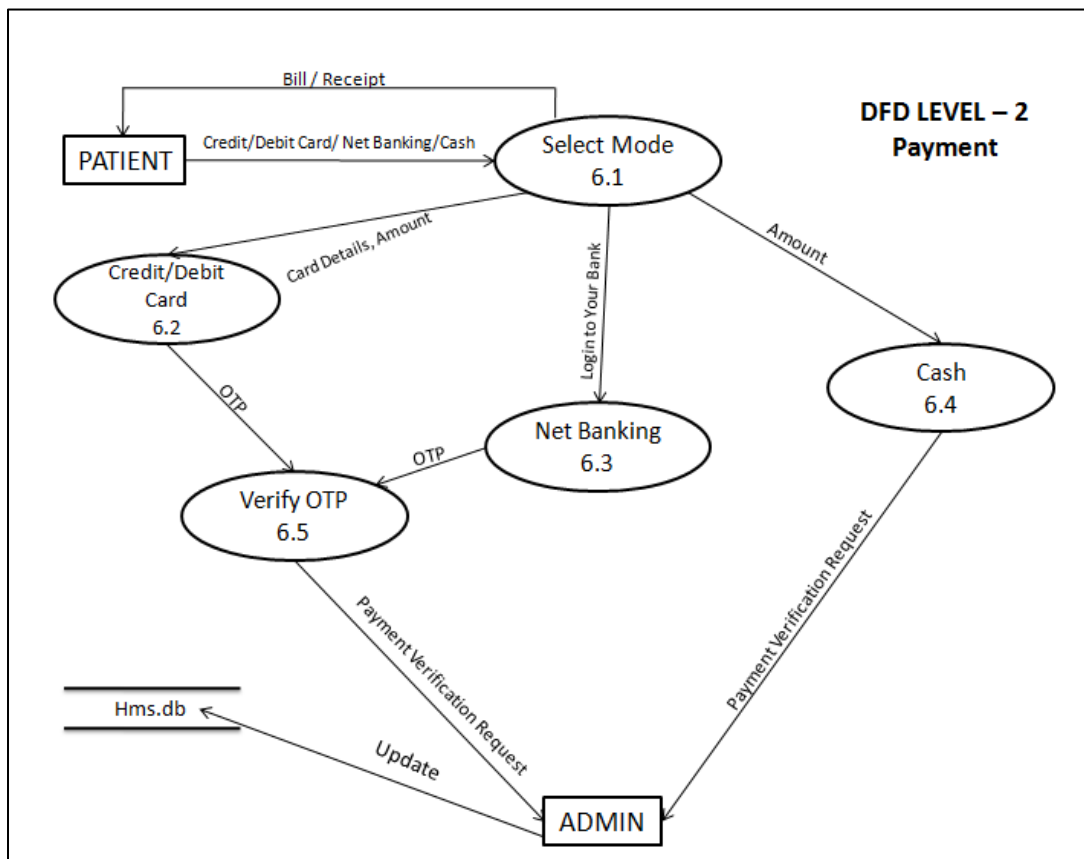
FIGURE 2.8 LEVEL – 2 Payment
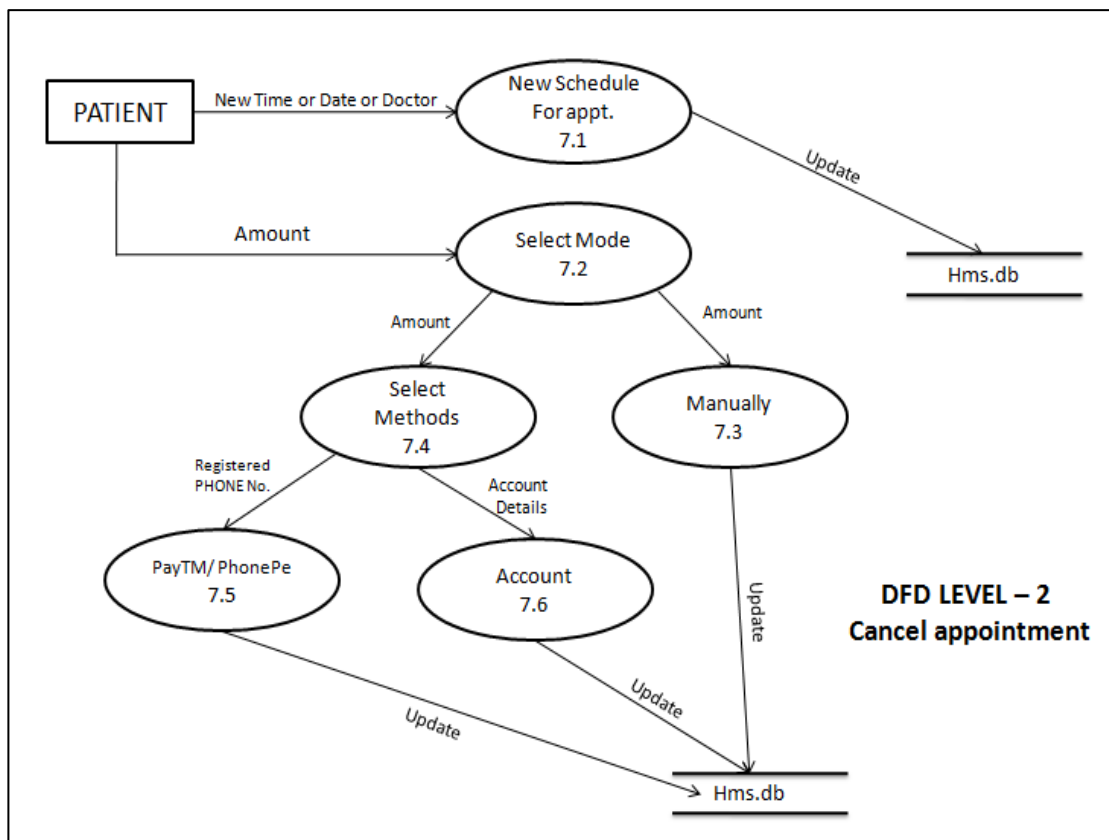
FIGURE 2.9 LEVEL – 2 Cancel Appointment
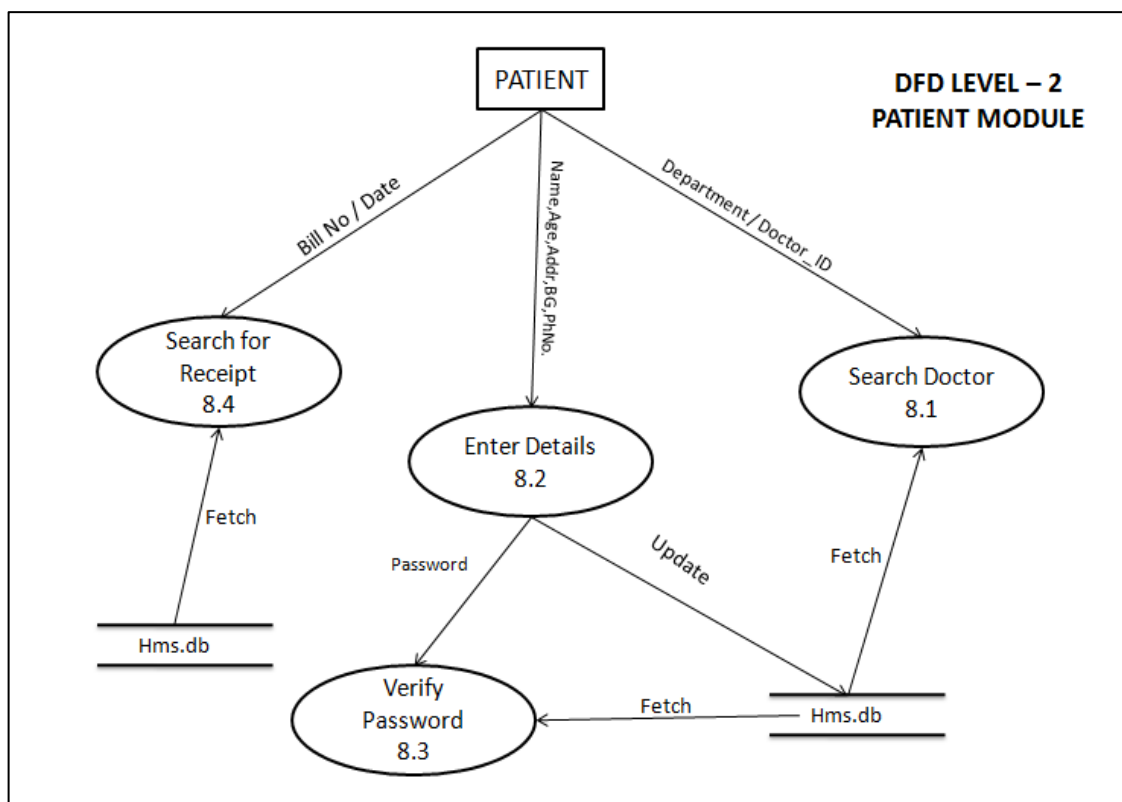
FIGURE 2.10 LEVEL – 2 Patient Module

2.4 USE CASE DIAGRAM
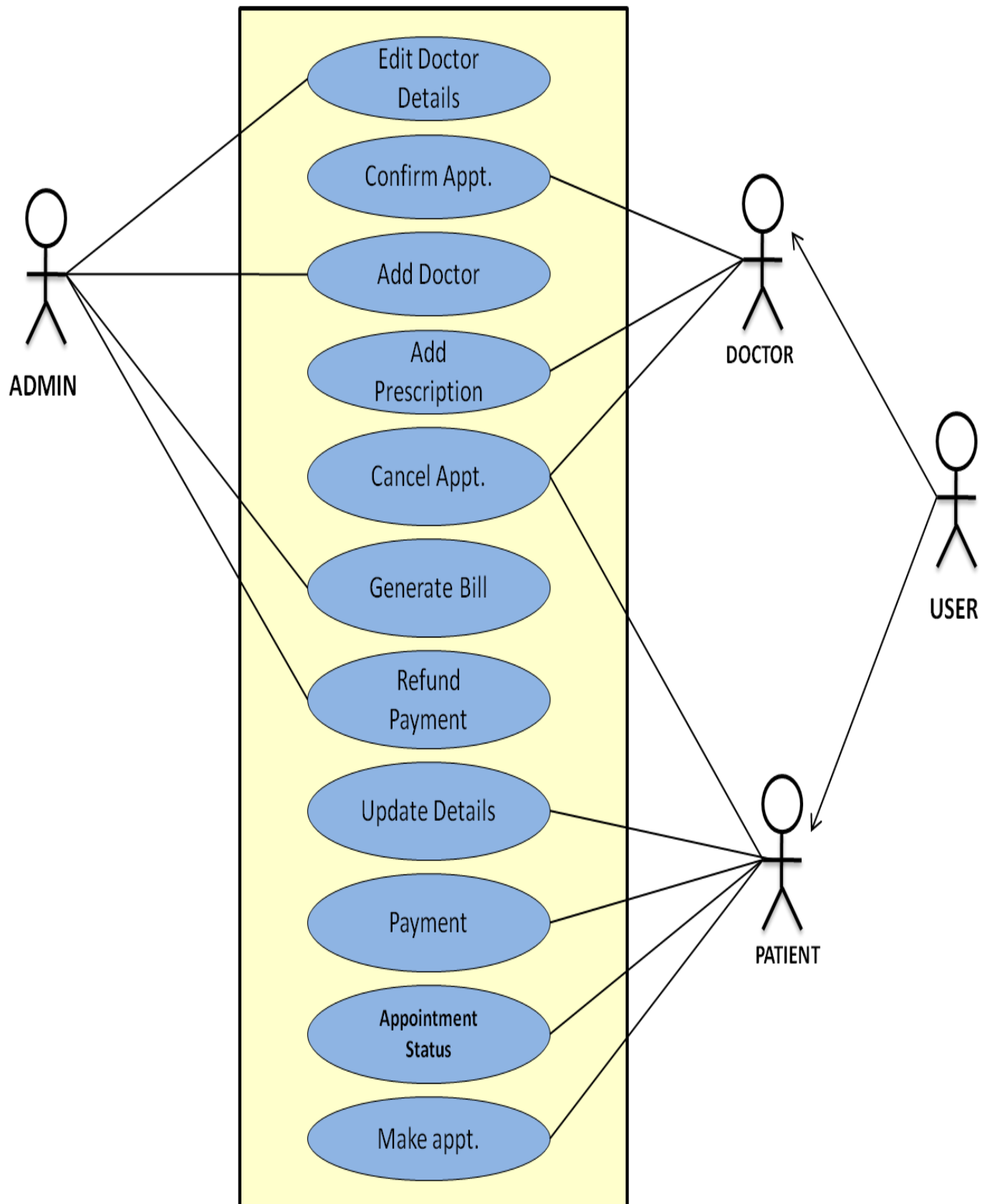
2.5 USE CASE DESCRIPTION

(1) PATIENT

* REGISTRATION

DESCRIPTION - The new patient can register themselves and add their details like name, age , gender, blood group etc. The patient entry will be made in the hms database.

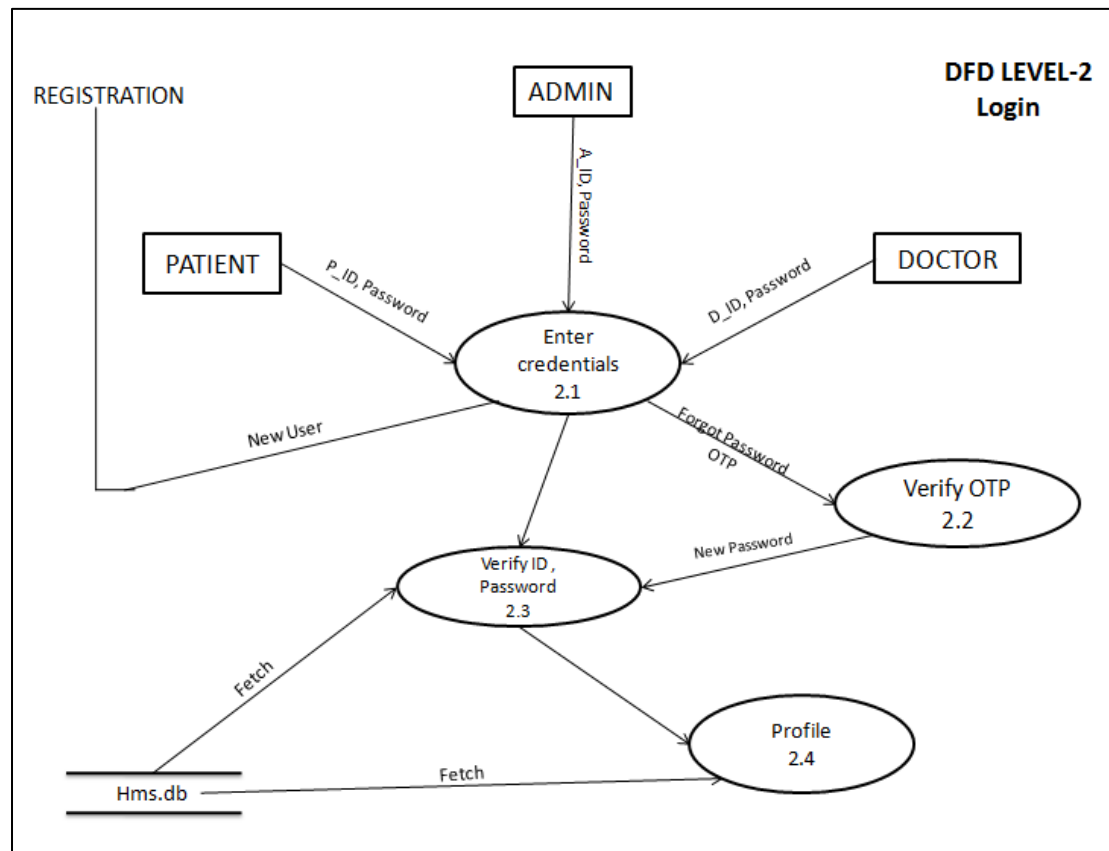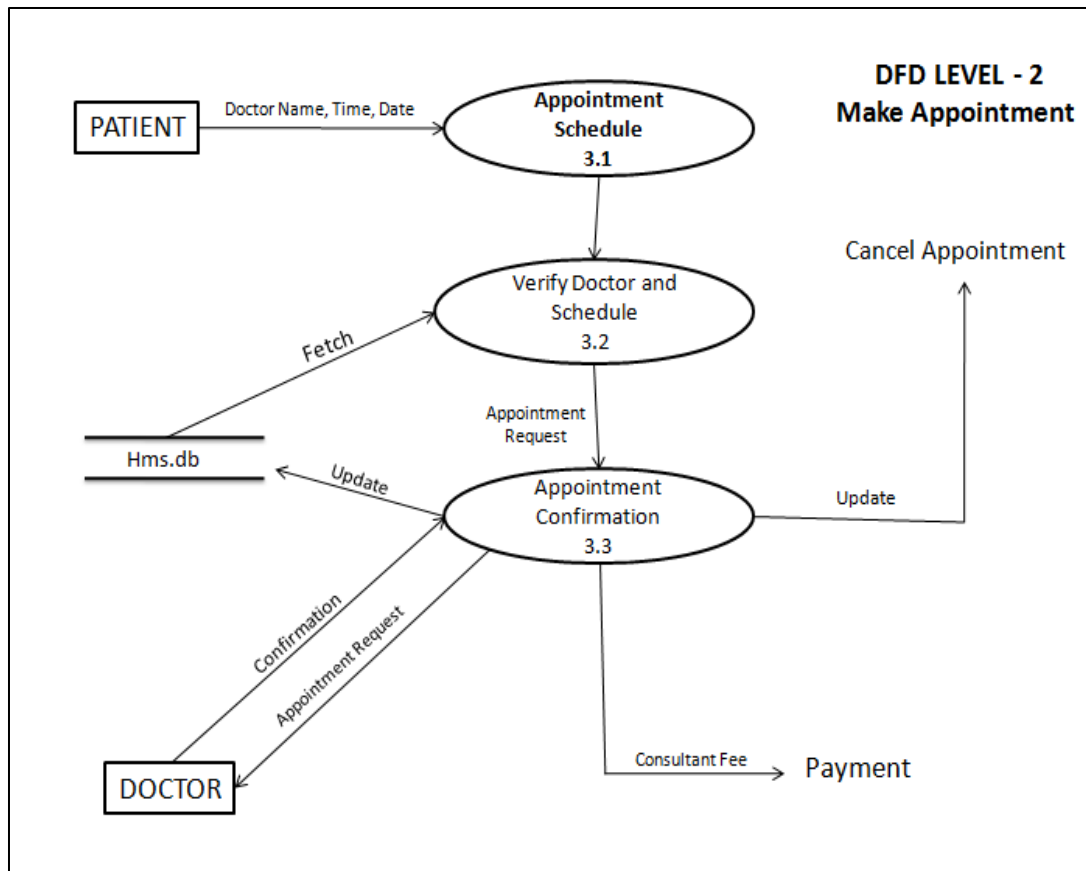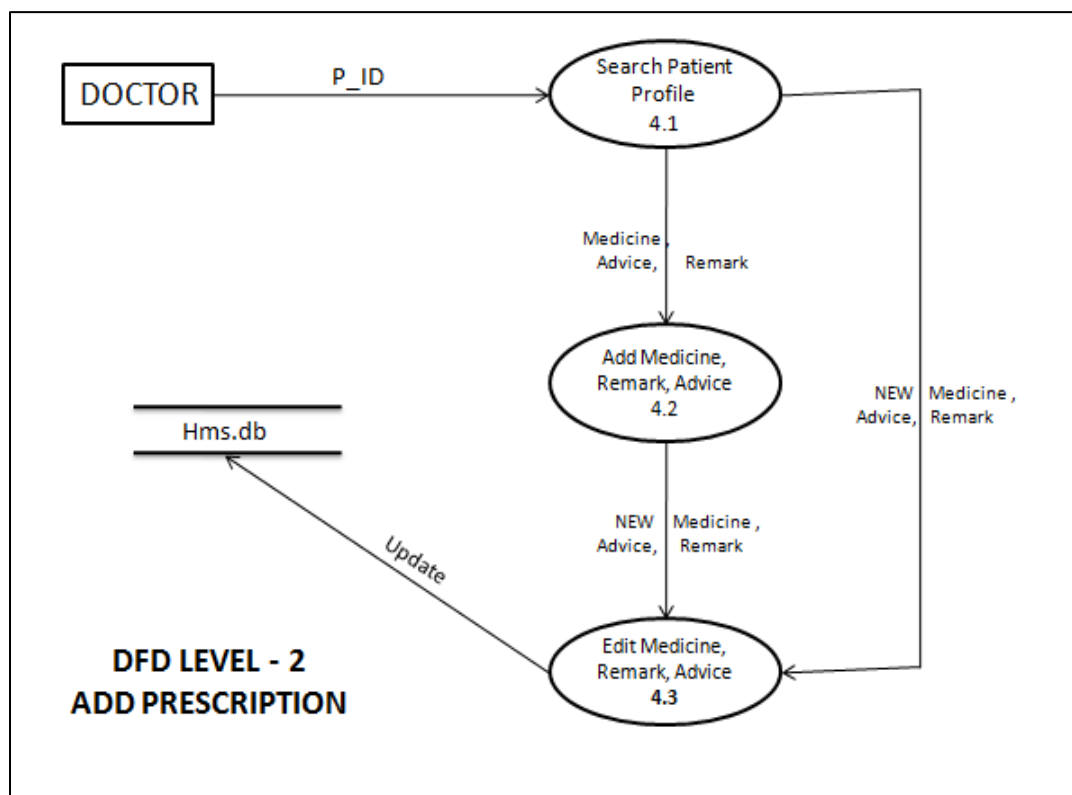PRE -CONDITION – The patient must be a new patient, If necessary fields left by user then prompt user to fill the necessary fields.

MAIN FLOW OF EVENTS
1. Patient selects sign up in login module.
2. A registration form get displayed
3. Patient fills the required details.

POST CONDITIONS - Patient record is added to hms database.

* UPDATION

DESCRIPTION-The patient should be enabled to update his/her details and the changes should reflect in hms database.

PRE-CONDITION – The patient must be a registered patient, The patient cannot update details after treatment starts.

MAIN FLOW OF EVENTS
1. Patient logs in to the system.
2. Patient view his record
3. Patient selects update details.
4. Now patient may change the necessary fields.
5. Pop of update details.

POST CONDITION - The record of patient is updated in hms database.

*APPOINTMENT

DESCRIPTION - It shows users a list of available doctors, timings, dates and enables patients to select the most suitable appointment date and doctor. The patient may also the cancel the appointment.

PRE-CONDITION - The patient must be a registered patient, Patient can fix only one appointment for a particular department.

MAIN FLOW OF EVENT
1. Patient first logs in to system.
2. View his/her record.
3. Create a new appointment or cancel the appointment..

POST CONDITIONS - patient details are displayed and a new appointment is fix or a existing appointment is cancelled. The hms database is updated.

*PAYMENT

DESCRIPTION – It enables user to pay the consultant fee of Doctor online.

PRE-CONDITION - The patient must be a registered patient, If Patient don't wants to pay online he/she can pay by cash also.

MAIN FLOW OF EVENT
1. Patient first logs in to system.
2. View his/her record.
3. Appointment confirmed by the Doctor then go for Payment.

POST CONDITIONS – A Reciept will be displayed. The hms database is updated

(2) DOCTOR

DESCRIPTION- The doctor view patient record/ update his details and add description of the treatment given to patient.

PRE-CONDITION – The doctor must be a registered doctor, System does not allow the doctor to modify the qualification, hospital managed details.

MAIN FLOW OF EVENTS
1.Doctor logs in to the system.
2. Doctor may select view patient.
   2.1 Patient record is displayed with treatment history.
3. Doctor add description of patient treatment.
4. Doctor may select appointment details
   4.1 Appointment Requests is displayed with schedule.
5. Doctor confirm or cancel appointment.

POST CONDITION – The patient and doctor 's database are updated.

(3) ADMIN

DESCRIPTION - The admin add doctor, update docotr details and verify payment and generate Bill/Reciept for the same.

MAIN FLOW OF EVENTS
1. Admin logs in the system.
2. Admin may add doctor new doctor.
   2.1 admin fills the doctor's details.
3. Admin view Doctor record.
   3.1 Admin enters the doctor id in the system.
   3.2 Doctor details are displayed, Admin can update details.
4. Admin Verify the payment submited by the Patient.
   4.1 Generate Bill/Reciept and confirmation message for the same.

PRE –CONDITION - Admin must first log in with his/her credentials.

POST CONDITION - The hms database is updated.

2.6 <u>User characteristics</u>

<u>ADMIN</u>
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.

Key functions:
- Access patient record, doctor Record.
- Add new doctor entry in system database.
- Confirm Payment and Generate Bill.
- View Records.(Total no of patients treated, doctor added/remove, consultant fee).

<u>PATIENT</u>
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.

Key functions:
- Make appointment.
- Cancel appointment.
- Update Details.
- Payment.
- View Payment History.

<u>DOCTOR</u>
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.

Key functions:
- Confirmation of appointment.
- Cancellation of appointment.
- Modification of appointment list.
- Add Prescription.

2.7 <u>Constraints</u>

- System is wirelessly networked with an encryption.
- System is only accessible within the hospital's website only.
- Database is password protected.
- Should use less RAM and processing power.
- Each user should have individual ID and password.
- Only administrator can access the whole system.

2.8 <u>Assumptions and dependencies</u>

- Each user must have a valid user id and password
- Server must be running for the system to function
- Users must log in to the system to access any record.
- Only the Administrator can delete records.

CHAPTER 3
SPECIFIC REQUIREMENTS

3.1 Performance requirements

3.2 Safety requirements

3.3 Security constraints

3.4 Software system attributes

      3.4.1 Usability

      3.4.2 Availability

      3.4.3 Correctness

      3.4.4 Maintainability

      3.4.5 Accessibility

3.5 Functional Requirements

3.1 <u>PERFORMANCE REQUIREMENTS</u>

- o **Response time**- The system will give responses within 1 second after checking the patient information and other information.
- o **Capacity**-The system must support 1000 people at a time
- o **User interface**- User interface screen will response within 5 seconds

3.2 <u>SAFETY REQUIREMENTS</u>

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure. All the administrative and data entry operators have unique logins so system can understand who is login in to system right now no intruders allowed except system administrative nobody cannot change record and valuable data.

3.3 <u>SECURITY REQUIREMENTS</u>

1. Want take the responsibility of failures due to hardware malfunctioning.
2. Warranty period of maintaining the software would be one year.
3. Additional payments will be analyzed and charged for further maintenance.
4. If any error occur due to a user's improper use. Warranty will not be allocated to it.
5. No money back returns for the software.

3.4 <u>SOFTWARE SYSTEM ATTRIBUTES</u>

3.4.1 <u>Usability:</u> Software can be used again and again without distortion.

3.4.2 <u>Availability:</u> The system shall be available all the time.

3.4.3 <u>Correctness:</u> Bug free software which fulfills the correct need/requirements of the client.

3.4.4 <u>Maintainability:</u> The ability to maintain, modify information and update fix problems of the system.

3.4.5 <u>Accessibility:</u> Administrator and many other users can access the system but the access level is controlled for each user according to their work scope.

3.5 <u>FUNCTIONAL REQUIREMENTS</u>

S.No.	MODULE NAME	APPLICABLE ROLES	DESCRIPTION
1.	LOGIN	PATIENT DOCTOR ADMIN	**PATIENT:** Can login using unique Id and Password after this system shall show his/her profile. **DOCTOR:** Can login using unique Id and Password after this system shall show his/her profile. **ADMIN:** Can login using unique Id and Password after this system shall show a profile with links to maintain the website.
2.	REGISTRATION	PATIENT	**PATIENT:** Can Register by filling all the required details, after this the system will verify the details and check if already registered or not.
3.	MAKE APPT.	PATIENT	**PATIENT:** Can Select doctor, date time and make an appointment request after this system shall show a confirmation for appointment request.
4.	CANCL APPT.	PATIENT DOCTOR	**PATIENT :** Can Cancel appointment if want to by just one click after this system shall ask for re-schedule or refund of payment. **DOCTOR :** Can Cancel appointment if want to by just one click after this system shall send a message to the patient.
5.	PAYMENT	PATIENT	**PATIENT :** Enter payment details and make payment after this system shall show the generated bill by the hospital.
6.	DOCTOR MODULE	ADMIN	**ADMIN :** Can add a new doctor by filling all the details after this system shall show a confirmation message. Can Remove a doctor by just one click after this system shall show confirmation message.

7.	PATIENT MODULE	PATIENT	**PATIENT :** Can view payment history or can search for a particular bill also after this system shall show a bill or history. Can also See or search for a doctor by entering dept. name or doctor id if known after this system will check for the doctor if found shall show doctor's profile. Can also update details after this system shall ask for re-enter password and after verifying password shall update details.
8.	ADD PRESCRIPTION	DOCTOR	**DOCTOR :** Enter Patient Id and after this all the treatment details and medicine, remark and advice for the patient after this system shall show a message for update.

CHAPTER 4
DESIGN

4.1 Data Dictionary

4.2 ER Diagram

4.3 Data Design

4.4 Component Level Diagram

4.1 DATA DICTIONARY

1.	legal_character	[a-z| A-Z]
2.	Dig it	[0-9]
3.	special_ch	[@|$|#|+|-]
4.	Blood	[A|B|AB|O]

1.	Name	first_name + (middle_name) + last_name
2.	first_name	{legal_character}*
3.	middle_name	{legal_character}*
4.	last_name	{legal_character}*
5.	P_ID	{legal_character + digit}*
6.	D_ID	{legal_character + digit}*
7.	A_ID	{legal_character + digit}*
8.	Password	{legal_character + digit + special_ch}*
9.	Address	House_no + (Street) + City
10.	House_no	{legal_character + digit}*
11.	Street	{legal_character}*
12.	City	{legal_character}*
13.	Mobile No.	{ digit }*
14.	Blood_Group	{Blood + special_ch}*
15.	Specialization	{legal_character}*
16.	Consultant Fee	{ digit }*
17.	Medicine	{legal_character + digit}*
18.	Advice	{legal_character + digit}*
19.	Remark	{legal_character + digit}*

Table 4.1 Data Dictionary

4.2 ER DIAGRAM
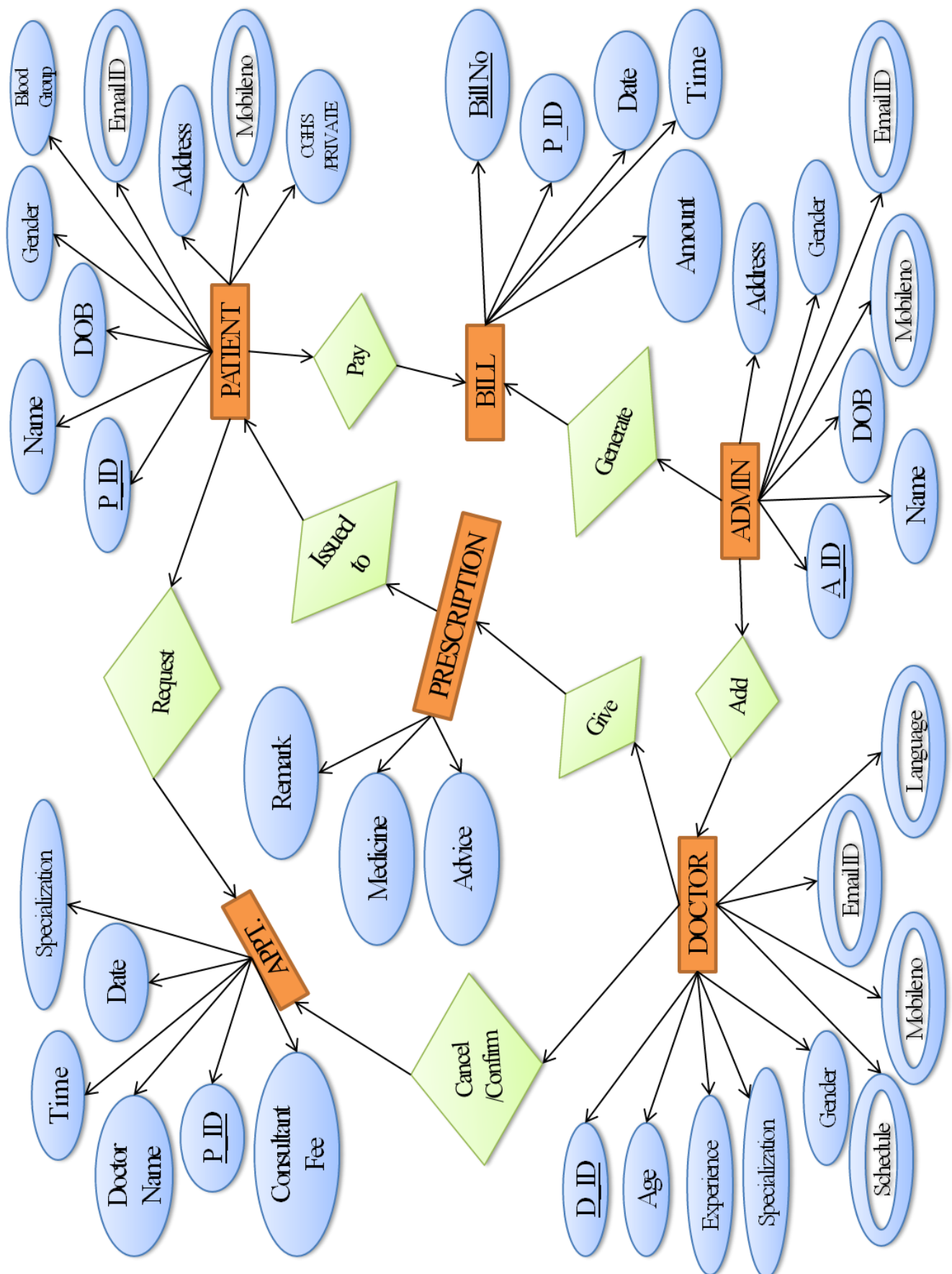
4.3 <u>DATA DESIGN</u>

S NO.	COLUMN NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1.	P_ID	Varchar(50)	Primary Key	Contains Unique Id
2.	Name	Varchar(50)	-	Contains Name
3.	DOB	Varchar(50)	-	Contains Date Of Birth
4.	Gender	Varchar(50)	-	Contains Gender
5.	Blood Group	Varchar(50)	-	Contains Blood Group
6.	Email ID	Varchar(50)	-	Contains Email Id
7.	Address	Varchar(50)	-	Contains Address
8.	Mobile No.	Integer	-	Contains Mobile No.
9.	CGHS/Private	Varchar(50)	-	Contains Category

Table 4.2 Patient

S NO.	COLUMN NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1.	P_ID	Varchar(50)	Primary Key	Contains Unique Id Patient
2.	Specialization	Varchar(50)	-	Contains Name of the Department in which Patient wants to visit
3.	Doctor's Name	Varchar(50)	-	Contains Doctor Name Patient Wants To Visit
4.	Consultant Fee	Integer	-	Contains Consultant Fee Of Doctor
5.	Date	Date	-	Contains Date For The Appointment
6.	Time	Time	-	Contains Time For The Appointment

Table 4.3 Appointment

S NO.	COLUMN NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1.	D_ID	Varchar(50)	Primary Key	Contains unique ID
2.	Age	Integer	-	Contains age
3.	Gender	Varchar(50)	-	Contains gender
4.	Specialization	Varchar(50)	-	Contains specialization
5.	Experience	Varchar(50)	-	Contains experience of the doctor (In months)
6.	Language	Varchar(50)	-	Contains in how many languages doctor can speak.
7.	Mobile No.	Integer	-	Contains mobile number
8.	Email ID	Varchar(50)	-	Contains Email Id
9.	Schedule	Varchar(50)	-	Contains day and time for which the doctor is available

Table 4.4 Doctor

S NO.	COLUMN NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1.	D_ID	Varchar(50)	-	Contains unique ID
2.	P_ID	Varchar(50)	Primary Key	Contains unique ID
3.	Medicine	Varchar(50)		Contains name of the medicine.
4.	Remark	Varchar(50)		Contains Remark given by the doctor for the patient.
5.	Advice	Varchar(50)		Contains any advice for the patient.

Table 4.5 Prescription

S NO.	COLUMN NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1.	A_ID	Varchar(50)	Primary Key	Contains unique ID.
2.	Name	Varchar(50)	-	Contains Name
3.	DOB	Varchar(50)	-	Contains Date Of Birth
4.	Gender	Varchar(50)	-	Contains Gender
5.	Email ID	Varchar(50)	-	Contains Email Id
6.	Mobile No.	Integer	-	Contains Mobile No.
7.	Address	Varchar(50)	-	Contains Address

Table 4.6 Admin

S NO.	COLUMN NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1.	P_ID	Varchar(50)	-	Contains unique ID.
2.	Bill No.	Varchar(50	Primary Key	Contains number of the bill.
3.	Date	Varchar(50	-	Contains Date of The bill.
4.	Time	Varchar(50	-	Contains Time of the bill generated.
5.	Amount	Int	-	Contains amount of the bill.

Table 4.7 Bill

4.4 COMPONENT LEVEL DIAGRAM

Book Appointment Module

enum Status { confirm , cancel} ;

int Department, Date , Time, mode, ch;
char Dr_Name(50);

cout<< Enter The Information :
cin>> Department;
cin>>Dr_Name;
cin>> Date;
cin>> Time;
bool Appointment = cancel;

cout<<Mode;
cout<<1.Cash;
cout<<2.Debit Card/Credit Card
cout<<3.Net Banking
cout<<Enter mode of payment;
cin>>mode;

if(mode==1)
{
        Generate a Receipt and send confirmation message;
}
else if(mode == 2)
{
      Enter Card Details
      Make Payment
      Send confirmation message
}
else
{
      Enter Account Details

Make Payment
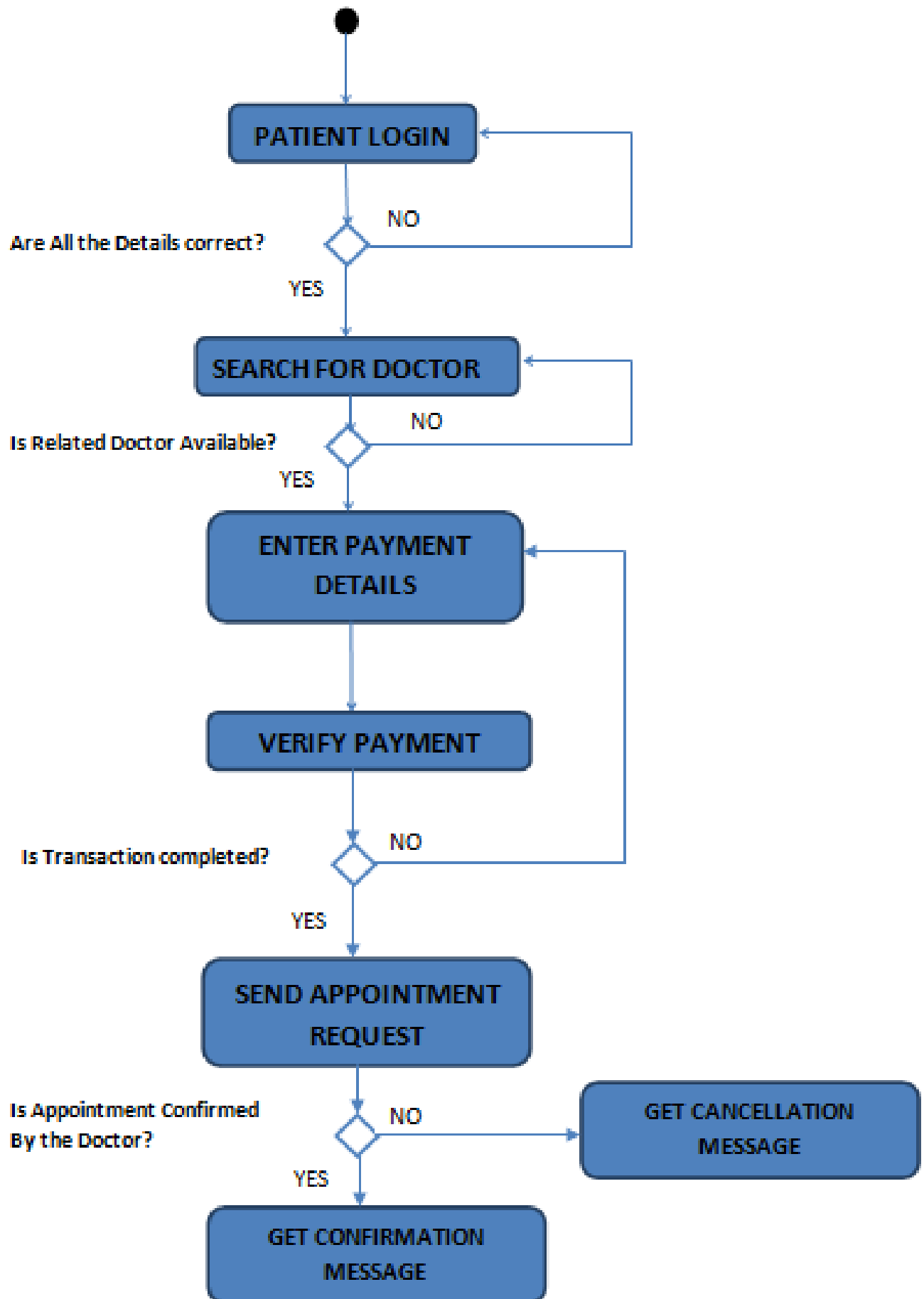   Send confirmation message
} //end if

Send appointment Request to the doctor

Doctor will check the Appointment Requests;
 cout<<Mode;
 cout<<1.Confirm;
 cout<<2.Cancel;
 cout<<Enter Your choice;
 cin>>ch;

if(ch==1)
{
        Appointment = Confirm;
        Send a Confirm Message to the patient.
 }
else
{
        Send a Cancel Message to the patient.
}//end if

CHAPTER 5
ESTIMATION AND SCHEDULING

5.1 <u>Project Scheduling</u>

TASK ~>	Planned Start	Actual Start	Planned complete	Actual complete	Assigned person
PROBLEM STATEMENT	Jan W1	Jan W1	Jan W1	Jan W1	Esha, Akansha, Monica
SOFTWARE MODEL	Jan W2	Jan W2	Jan W2	Jan W3	Akansha, Monica
PROJECT SCHEDULING	Jan W2	Jan W2	Jan W3	Jan W3	Esha, Akansha
SRS	Jan W3	Jan W3	Feb W1	Feb W1	Esha, Monica
CONTEXT LEVEL DIAGRAM	Feb W1	Feb W1	Feb W1	Feb W2	Esha, Monica
DFD LEVEL - 1	Feb W2	Feb W2	Feb W2	Feb W3	Akansha, Monica
DFD LEVEL - 2	Feb W3	Feb W3	Feb W3	Feb W4	Esha, Akansha, Monica
DATA DICTIONARY	Mar W1	Mar W1	Mar W1	Mar W1	Esha, Akansha
ER DIAGRAM	Mar W1	Mar W1	Mar W2	Mar W2	Esha
DATA DESIGN	Mar W2	Mar W2	Mar W2	Mar W2	Akansha
USE CASE DIAGRAM	Mar W3	Mar W3	Mar W3	Mar W3	Akansha
USE CASE DISCRIPTION	Mar W4	Mar W4	Mar W4	Mar W4	Akansha, Esha
FUNCTION POINT MATRIX	Mar W3	Mar W3	Mar W3	Mar W4	Esha, Akansha, Monica
COCOMO MODEL	Apr W1	Apr W1	Apr W1	Apr W1	Esha, Akansha, Monica
RISK ANALYSIS	Apr W2	Apr W2	Apr W2	Apr W2	Esha, Akansha
TESTING	Apr W2	Apr W2	Apr W2	Apr W2	Esha, Akansha

Table 5.1 Project Scheduling

Jan – January   Feb – February   Mar – March   Apr – April   W - Week
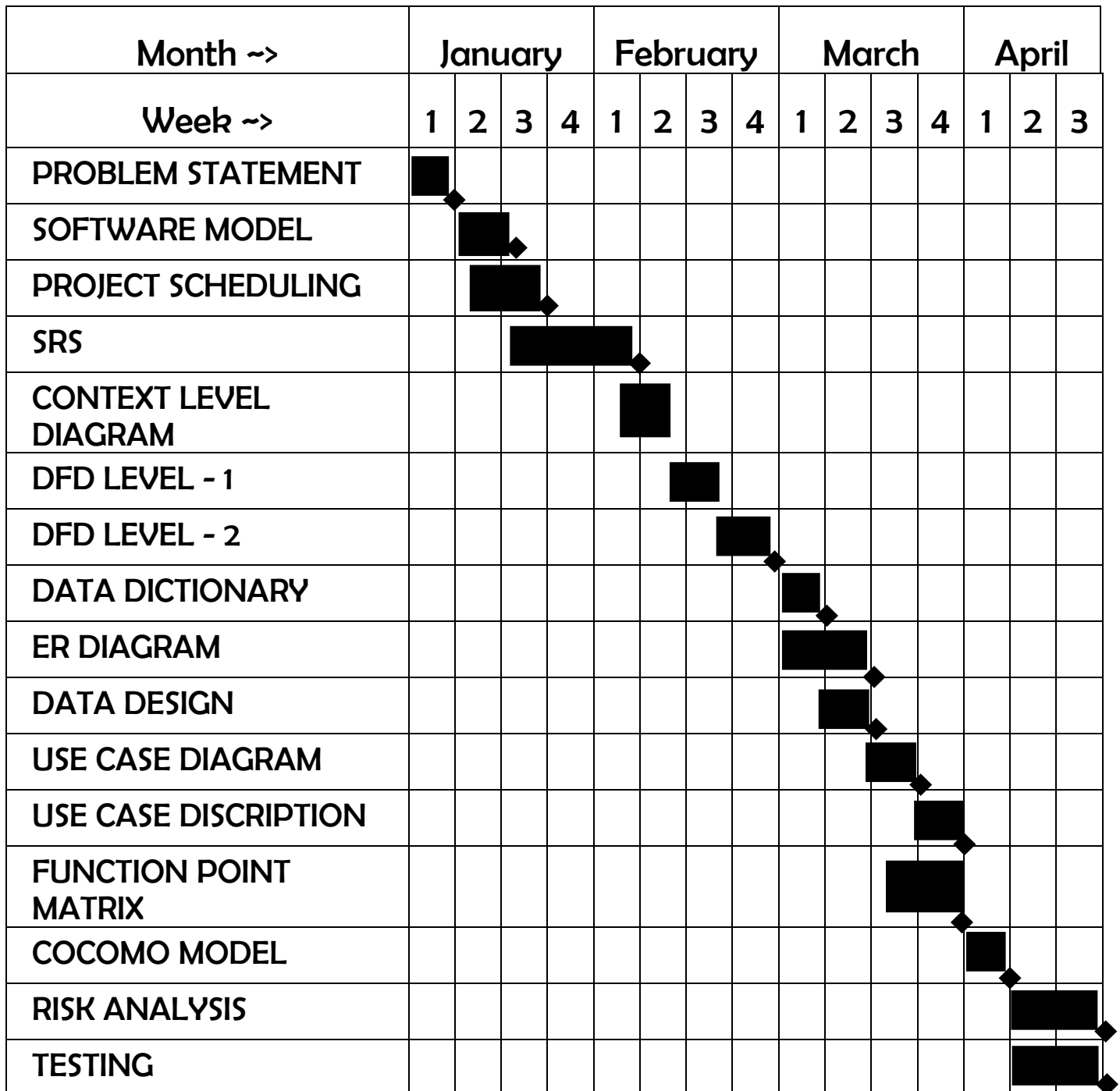
5.2 Timeline chart

Month ~>	January				February				March				April		
Week ~>	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3
PROBLEM STATEMENT	■														
SOFTWARE MODEL		■													
PROJECT SCHEDULING		■	■												
SRS			■	■											
CONTEXT LEVEL DIAGRAM					■										
DFD LEVEL - 1						■									
DFD LEVEL - 2							■								
DATA DICTIONARY								■							
ER DIAGRAM									■	■					
DATA DESIGN										■					
USE CASE DIAGRAM											■				
USE CASE DISCRIPTION												■			
FUNCTION POINT MATRIX												■			
COCOMO MODEL													■		
RISK ANALYSIS														■	
TESTING														■	

Table 5.2 Timeline Chart

5.3 <u>Size Estimation</u> (FUNCTION BASED METRICS)

Information domain values are defined in the following manner:

- **Number of external inputs (EIs)** - Each external input originates from a user or is transmitted from another application and provides distinct application-oriented data or control information. Inputs are often used to update internal logical files (ILFs). Inputs should be distinguished from inquiries, which are counted separately.

- **Number of external outputs (EOs) -** Each external output is derived data within the application that provides information to the user. In this context external output refers to reports, screens, error messages, etc. Individual data items within a report are not counted separately.

- **Number of external inquiries (EQs) -** An external inquiry is defined as an online input that results in the generation of some immediate software response in the form of an online output (often retrieved from an ILF).

- **Number of internal logical files (ILFs) -** Each internal logical file is a logical grouping of data that resides within the application's boundary and is maintained via external inputs.

- **Number of external interface files (EIFs)**. - Each external interface file is a logical grouping of data that resides external to the application but provides information that may be of use to the application.

SIZE ESTIMATION FOR THIS PROJECT

Screen No	EIs	EOs	EQs	ILFs	EIFs
1.	1.Select Language	-	1.Doctor's On Leave 2.Visitors on Website	Hospital File	-
2.	-	-	-	-	-
3.	1.Username 2. Password	-	-	Hospital File	-
4.	1 .Name 2 .Dob 3. Gender 4 .Email 5. Blood Group 6 .Mobile No 7 .Address 8 .CGHS / Private 9.Card Picture	-	-	Hospital File	-
5.	-	1.Profile	-	HF	-
6.	1. Department 2 .Date 3 .Time 4 .Doctor Name	-	-	Hospital File	-
7.			1.Appointment Status	Hospital File	-
8.	1 .Card Holder Name 2. Card number 3. Expire Date 4. CVC Number	-	-	Hospital File	-
9.	1.Registered Mobile No. 2.Edit Appt. Schedule	-	-	Hospital File	-
10.	-	-	1.Payment History	Hospital File	-
11.	-	1.Profile	-	HF	-

12.	1.Doctor ID	1.Doctor Details		Hospital File	-
13.	-	1.Bill	-	Hospital File	-
14.	1.Username 2. Password	-	-	Hospital File	-
15.	-	1. Profile		Hospital File	-
16.	-	-	1.appt. Details	Hospital File	-
17.	1. Treatment Name 2 .Medicine 3 .Advice 4 .Remark 5.Patient ID	1.Patient Profile	-	Hospital File	-
18.	1.Username 2. Password	-	-	Hospital File	-
19.	1.Payment Verify	-	-	Hospital File	-
20.	1 Name 2 Age 3 Gender 4 Specialization 5 Experience 6 Language 7 Mobile No 8 Email Id 9 Schedule	-	-	Hospital File	-
21.	1.Doctor Id	1.Doctor Profile	-	Hospital File	-
22.	1.Select Monthly/Yearly 2.Select Year 3.Select Month	-	1.Records	Hospital File	-

TABLE 5.3 Function Point Complexity Weights

Measurement parameter	Weighting factor		
	Simple	Average	Complex
Number of user inputs	3	4	6
Number of user outputs	4	5	7
Number of user inquiries	3	4	6
Number of files	7	10	15
Number of external interfaces	5	7	10

TOTAL EXTERNAL INUPUTS = 41

TOTAL EXTERNAL OUTPUTS = 7

TOTAL LOGICAL INTERNAL FILES = 1

TOTAL EXTERNAL INQUIRIES = 6

TOTAL EXTERNAL INTERFACE FILES = 0

Function point = FP = UFP x CAF = Count Total * (0.65 + (0.01 *$\sum fi$)

UFP (Count Total) = Sum of all the complexities i.e. the 5 parameters provided in the question,

CAF = Complexity Adjustment Factor i.e. 0.65 + (0.01 * $\sum$fi),

CALCULATING ($\sum fi$)

Total Degree of Influence of the 14 General System Characteristics

1. How many communication facilities are there to aid in the transfer or exchange of information with the application or system?
2. How are distributed data and processing functions handled?
3. Did the user require response time or throughput?
4. How heavily used is the current hardware platform where the application will be executed?
5. How frequently are transactions executed daily, weekly, monthly, etc.?
6. What percentage of the information is entered online?
7. Was the application designed for end-user efficiency?
8. How many ILFs are updated by online transaction?
9. Does the application have extensive logical or mathematical processing?
10. Was the application developed to meet one or many user's needs?
11. How difficult is conversion and installation?
12. How effective and/or automated are start-up, back-up, and recovery procedures?
13. Was the application specifically designed, developed, and supported to be installed at multiple sites for multiple organizations?
14. Was the application specifically designed, developed, and supported to facilitate change?

Considering all adjustment factors of average influence $\sum fi$ = 14 * 3 = 42

TOTAL EXTERNAL INUPUTS = 41

TOTAL EXTERNAL OUTPUTS = 7

TOTAL LOGICAL INTERNAL FILES = 1

TOTAL EXTERNAL INQUIRIES = 6

TOTAL EXTERNAL INTERFACE FILES = 0

Assuming all the parameters are of <u>SIMPLE COMPLEXITY</u>.

UFP (Count Total) = {41 * 3} + {7 * 4} + {1 * 3} + {6 * 7} + {0 * 5} = 196

Considering all adjustment factors of average influence $\sum fi$ = 14 * 3 = 42

Function point = FP = Count Total + (0.65 + (0.01 * $\sum fi$)

$$= 196 * (0.65 + (0.01 * 42)$$
$$= 196 * (0.65 + (0.42)$$
$$= 196 * (1.07)$$
$$= 209.72$$

FUNCTION POINT = 209.72

5.4 Cost Estimation (COCOMO II MODEL)

The original COCOMO model became one of the most widely used and discussed software cost estimation models in the industry. It has evolved into a more comprehensive estimation model, called COCOMO II.

COCOMO II models require sizing information. Three different sizing options are available as part of the model hierarchy:-
o Object Points
o Function Points
o Lines Of Source Code

The COCOMO II application composition model uses object points.

Like function point, the **object point is an indirect software measure** that is computed using counts of the number of
(1) screens (at the user interface),
(2) reports,
(3) components likely to be required to build the application.
Each object instance (e.g., a screen or report) is classified into one of three complexity levels (i.e. ,simple ,medium, or difficult).
Once complexity is determined, the number of screens, reports, and components are weighted according to the table illustrated in Table 5.4 .

TABLE 5.4 COCOMO II Complexity Weights

OBJECT TYPE	COMPLEXITIY WEIGHT		
	SIMPLE	MEDIUM	DIFFICULT
SCREEN	1	2	3
REPORT	2	5	8
3GL COMPONENT			10

The object point count is then determined by multiplying the original number of object instances by the weighting factor in the figure and summing to obtain a total object point count.

When component-based development or general software reuse is to be applied, the percent of reuse (%reuse) is estimated and the object point count is adjusted:

NOP = (Object Point) * [(100 - %reuse) / 100]

where NOP = defined as new object points.

To derive an estimate of effort based on the computed NOP value, a **"productivity rate"** must be derived.

$$\text{PROD} = \frac{NOP}{Person-Month}$$

Table 5.5 presents the productivity rate for different levels of developer experience and development environment maturity. Once the productivity rate has been determined, an estimate of project effort is computed using

$$\text{ESTIMATED EFFORT} = \frac{NOP}{PROD}$$

TABLE 5.5 Productivity Rate For Object Point Counts

Developer's experience/capability	Very Low	Low	Normal	High	Very high
Environment maturity/capability	Very Low	Low	Normal	High	Very high
PROD	4	7	13	25	50

COST ESTIMATION FOR THIS PROJECT

(1) SCREENS

1. Home Page.
2. Select Login.
3. Login Page For Patient .
4. Registration For Patient.
5. Patient Profile.
6. Patient Update Details.
7. Book Appointment .
8. View Appointment Status .
9. Cancel Appointment.
10. Payment By Patient.
11. Receipt Of Payment .
12. Login Page For Doctor.
13. Doctor Profile.
14. Appointment Details.
15. View Patient by Doctor.
16. Add Prescription.
17. Login Page For Admin.
18. Generate Bill.
19. Update Doctor Details.
20. Add Doctor.
21. View doctor By Admin.
22. View Records.

(2) REPORTS

1. Total Visitors on Website.
2. Total Patients Treated.
3. Total Appointments Taken.
4. Total Appointments Cancelled.
5. Total Doctors on Leave.
6. Total Doctors Added.
7. Total Doctors Removed.
8. Total Consultant Fee Collected .

TOTAL SCREENS = 22

TOTAL 3GL MODULES = 0

TOTAL REPORTS = 8

CONSIDERING ALL OF THE ABOVE HAVE MEIDEM COMPEXITY, 0% OF COMPONENTS ARE REUSED AND TAKING THE DEVELOPER EXPERIENCE AND ENVIRONMENT MATURITY AS LOW.

PRODUCTIVITY RATE $= \dfrac{7+7}{2} = 7.$

OBJECT POINT = {22 * 2} + {8 * 5} = 84.

ESTIMATED EFFORT $= \dfrac{\text{NOP}}{\text{PROD}} = \dfrac{84}{7} =$ **12 Person-Months.**

CHAPTER 6
SAMPLE SCREENSHOTS

FIGURE 6.1 HOME PAGE

FIGURE 6.2 SELECT LOGIN

FIGURE 6.3 PATIEN LOGIN PAGE

FIGURE 6.4 REGISTRATION

FIGURE 6.5 PATIENT PROFILE

MOEKA HOSPITAL

Patient ID : Esha09

CGHS PATIENT

Patient Name	: Esha Bisht
D.O.B	: 09/10/2000
Age/Gender	: 18 years 5 months/F
Blood Group	: A+
Email	: esha013@gmail.com
Address	: ITO , New Delhi
Mobile no.	: 9987654321

CGHS CARD PHOTO

CENTRAL GOVERNMENT HEALTH SCHEME

Director

Recent Treatment Details

Treatment Name : Nose Problem
Doctor Name : Dr. Akansha Rathi, ENT Specialist

Prescription: Fluticasone / Mometasone Nose Spray

Remark : Use at Bedtime.

Advice: You don't get the full effect for several days or even a week, But if you use it daily, it'll be incredibly effective.

- ➤ Profile
- Book Appointment
- Appointment Status
- Update Details
- View Doctor
- Receipt of Payment
- Logout

FIGURE 6.6 PATIENT UPDATE DETAILS

MOEKA HOSPITAL

Patient ID : Esha09

NAME	:	Esha Bisht
DOB	:	09/10/2000
GENDER	:	FEMALE
BLOOD GROUP	:	A+
EMAIL	:	esha013@gmail.com
MOBILE NO	:	ITO , New Delhi
ADDRESS	:	9987654321
PRIVATE/CGHS	:	CGHS

CGHS CARD PHOTO

CENTRAL GOVERNMENT HEALTH SCHEME

Director

Browser

- Profile
- Book Appointment
- Appointment Status
- ➤ Update Details
- View Doctor
- Receipt of Payment
- Logout

SUBMIT

FIGURE 6.7 PATIENT BOOK APPOINTMENT

MOEKA HOSPITAL		Profile
		➤**Book Appointment**
		Appointment Status
		Update Details
Select Department : Select ▼		View Doctor
Select Doctor Name* : Select ▼		Receipt of Payment
Consultant Fee :		Logout
Date* : Select ▼		
Time* : Select ▼		
Book Appointment **Edit Appointment**		
		*Mandatory

FIGURE 6.8 PATIENT APPOINTMENT STATUS

	Profile
MOEKA HOSPITAL	Book Appointment
	➤**Appointment Status**
	Update Details
	View Doctor
	Receipt of Payment
	Logout

Patient ID	Doctor Name	Dept	Consultant Fee	Status	Cancel	Payment
Esha09	Akansha Rathi	ENT	150.00	Confirmed	🚫	**Paid**

Note:-
1. Appointment will be confirmed by the doctor .

FIGURE 6.9 PATIENT CANCEL APPOINTMENT

CANCEL APPOINTMENT

Profile

Book Appointment

➤ Appointment Status

Update Details

View Doctor

Receipt of Payment

Logout

Do You Want To Edit Schedule?   YES   NO

SELECT METHOD

| PhonePe ← |
| PayTM |
| CASH |

ENTER REGISTERED MOBILE NO. [   ]

SUBMIT

FIGURE 6.10 PAYMENT

PAYMENT DETAILS

Accepted Card   VISA   MasterCard   Maestro   AMERICAN EXPRESS

Card Holder's Name [   ]

Card Number [   ][   ][   ][   ]

Expire Date [   ]   CVC [   ]

Cancel   CONTINUE

This payment will be billed to the card you specified. The name on the statement will be company.com

FIGURE 6.11 PATIENT PAYMENT RECIPET

BILL OF SUPPLY

Mobile no. : 9987654321
Patient Name : Esha Bisht
Age/Sex : 18 years 5 months / F
Address : ITO , New Delhi

Patient ID : Esha09
Bill No. : C9456

Bill Date : 23/02/18   04:38PM

Referred By : SELF

CGHS/PRIVATE : CGHS

MOEKA HOSPITAL

S.No	Services	Qty	Base Price(Rs.)	Tariff Price(Rs.)	Discount	Net Amount(Rs.)	Tax Amt (Tax %)	Bill Amount(Rs.)
1	Akansha Rathi (ENT) (Consultation)	1	600.00	150.00	0.00	150.00		150.00
			Total :	0.00	150.00	0.00		150.00

Paid by Patient :   150.00

Signature
Monica

Print   Download

FIGURE 6.12 PATIENT VIEW PAYMENT HISTORY

MOEKA HOSPITAL

PAYMENT HISTORY

DATE	BILL NO.	DOCTOR'S NAME	AMOUNT	STATUS	
23/2/2018	C9456	Akansha Rathi	150.00	Paid	Open
11/1/2018	D521	Ashi Singh	500.00	Refund	Open

Profile

Book Appointment

Appointment Status

Update Details

View Doctor

➢ Receipt of Payment

Logout

FIGURE 6.13 PATIENT VIEW DOCTORS

		Profile
		Book Appointment
		Appointment Status
		Update Details
		➤View Doctor
		Receipt of Payment
		Logout

LIST OF DOCTORS AND THEIR DEPARTMENT

Search By Department or Doctor's Name

Name of Doctor	Department	Consultant Fee
Akansha Rathi	ENT	600.00
Shikha Bisht	Plastic Surgeon	1200.00
Reha Karmakar	Pediatrician	500.00
Muskan	Ophthalmologist	700.00
Kavita Pandey	Psychiatrist	900.00
Ashi Singh	Cardiologist	1500.00

FIGURE 6.14 DOCTOR LOGIN PAGE

DOCTOR LOGIN

USERNAME :

PASSWORD :

LOGIN

FIGURE 6.15 DOCTOR PROFILE

MOEKA
HOSPITAL

Doctor ID : ARathi12

Name: Akansha Rathi
Age : 35
Gender : Female
Specialization : Nasal
Experience : 18 years
Language Spoke : English, Hindi, Urdu

Contact Details :-
Mobile No. : +91-8945639342
Email: Akansha12@gmail.com

Schedule :-

Monday	Wednesday	Friday
9:00AM-12:00PM	2:00PM-4:00PM	12:00PM-3:00PM

FIGURE 6.16 DOCTOR VIEW APPOINTMENT

MOEKA
HOSPITAL

Today : 25/02/18, Wednesday

P_ID	Name	Schedule	Status	Cancel
Esha09	Esha Bisht	Wednesday(2:30PM)	Confirmed	🚫

23/02/18, Monday

P_ID	Name	Schedule	Status	Cancel
Esha09	Esha Bisht	Monday(10:30AM)	Cancelled	🚫

FIGURE 6.17 DOCTOR ADD DESCRIPTION

MOEKA HOSPITAL	Profile
	My Appointments
	➤ Add Prescription
	Logout

Enter Patient ID : Search by P_ID

P_ID : Esha09

Patient Name : Esha Bisht
Treatment Name : Nose Problem

Prescription: Fluticasone / Mometasone Nose Spray
Remark : Use at Bedtime.

Advice: You don't get the full effect for several days or even a week, But if you use it daily, it'll be incredibly effective.

Edit Prescription

FIGURE 6.18 ADMIN LOGIN PAGE

ADMIN LOGIN

USERNAME :

PASSWORD :

LOGIN

NEW USER? REGISTER...

FIGURE 6.19 ADMIN ADD DOCTOR

MOEKA HOSPITAL

ADD NEW DOCTOR

NAME	:	
AGE	:	
GENDER	:	
SPECIALIZATION	:	▼
EXPERIENCE	:	
LANGUAGE SPOKEN	:	
MOBILE NO.	:	
EMAIL	:	
SCHEDULE	:	

Profile

Payment Request

➤ Add Doctor

Update Details of Doctor

Records

Logout

Add Doctor

FIGURE 6.20 ADMIN UPDATE DOCTOR DETAILS

MOEKA HOSPITAL

Enter Doctor ID : Search by D_ID

Profile

Payment Request

Add Doctor

➤ Update Details of Doctor

Records

Logout

Name: Akansha Rathi
Age : 35
Gender : Female
Specialization : Nasal
Experience : 18 years
Language Spoke : English, Hindi, Urdu

Contact Details :-
Mobile No. : +91-8945639342
Email: Akansha12@gmail.com

Schedule :-

Monday	Wednesday	Friday
9:00AM-12:00PM	2:00PM-4:00PM	12:00PM-3:00PM

Edit Details   Remove Doctor

FIGURE 6.21 ADMIN PAYMENT REQUEST

MOEKA HOSPITAL

PAYMENT REQUESTS

S.No	Patient ID	Doctor Name	Consultant Fee	Status	Generate Receipt and Upload
1	Esha09	Akansha Rathi	600.00	Verified	Done

VERIFY ALL REQUESTS GENERATE RECEIPT AND UPLOAD

Profile

➤Payment Request

Add Doctor

Update Details of Dr

Records

Logout

FIGURE 6.22 ADMIN VIEW RECORDS

MOEKA HOSPITAL

RECORD : Monthly ⬇

SELECT YEAR : 2018 ⬇
SELECT MONTH : January ⬇

Total Patient Treated	:	10,123
Total Doctor Added	:	10
Total Doctor's Removed	:	5
Total Appointments Taken	:	10,123
Total Appointments Cancelled	:	456
Total Fee Collected	:	Rs. 10,10,700

Profile

Payment Request

Add Doctor

Update Details of Doctor

➤Records

Logout

CHAPTER 7
RISK ANALYSIS

S.No	RISK	CATEGORY	PROBABLITY (P)	IMPACT (I)	RMMM PLAN
1.	SOME TEAM MEMBER BECOME SICK IN BETWEEN	TECHNICAL RISK	20%	2	OTHER TEAM MEMBERS DISTRIBUTE THE WORK IN BETWEEN THEM
2.	DELIVERY DEADLINE TIGHTENED	PROJECT RISK	30%	1	TEAM MAY USE EXTRA MEMBERS TO DO JOB ON SCHEDULED TIME
3.	LOSING OF ALL PROJECT DATA THIS MAY HAPPEN DUE TO HARD DISK FAILURE	PROJECT RISK	20%	2	BACK UP THE PROJECT ONLINE OR IN EVERY SYSTEM OF EVERY MEMBER
4.	TEAM DISTENTION / LACK OF COHESION	PROJECT RISK	10%	3	WE MAKE SOME RULES HOW WE CONSULT EACH OTHER

TABLE 7.1

CHAPTER 8

TESTING

8.1 WHITE BOX TESTING
    8.1.1 Basic Path (Pseudo code)
    8.1.2 Flow Graph
    8.1.3 Cyclomatic Complexity
    8.1.4 Independent Paths

BASIS PATH TESTING FOR BOOK APPOINTMENT MODULE

```
enum Status { confirm , cancel} ;

int Department, Date , Time, mode, ch;
char Dr_Name(50);

cout<< Enter The Information :
cin>> Department;
cin>>Dr_Name;
cin>> Date;
cin>> Time;
bool Appointment = cancel;                                    1

cout<<Mode;
cout<<1.Cash;
cout<<2.Debit Card/Credit Card
cout<<3.Net Banking
cout<<Enter mode of payment;
cin>>mode;

if(mode==1)    ──────────────────────    2
{
        Generate a Receipt and send confirmation message;   3
}
else if(mode == 2)──────────────────    4
{
        Enter Card Details
        Make Payment                                 5
        Send confirmation message
}
else
{
        Enter Account Details                        6
          Make Payment
        Send confirmation message
```

} //end if   ———————————————— 7

Send appointment Request to the doctor   8

   Doctor will check the Appointment Requests;
   cout<<Mode;
   cout<<1.Confirm;
   cout<<2.Cancel;   9
   cout<<Enter Your choice;
   cin>>ch;

if(ch==1)   ———————————————— 10
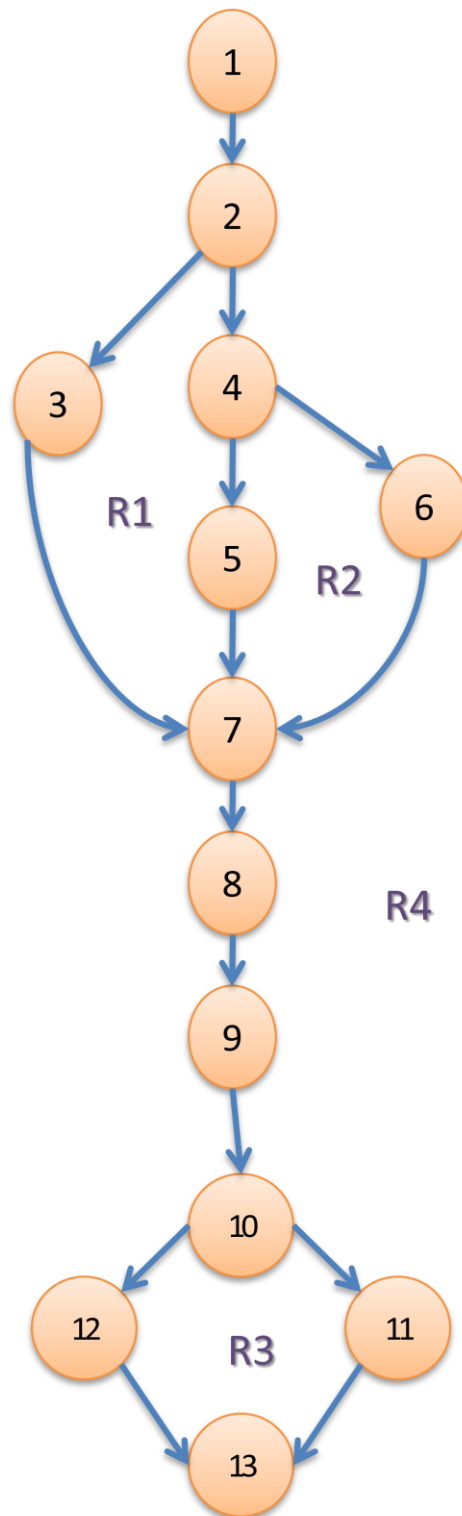{
     Appointment = Confirm;
     Send a Confirm Message to the patient.   11
}
else
{
     Send a Cancel Message to the patient.   12

}//end if   ———————————————— 13

FLOW GRAPH NOTATION

2) CYCLOMATIC COMPLEXITY V(G)

1. Cyclomatic complexity V(G) = Total number of Regions.

V(G) = 4.

2. Cyclomatic complexity V(G) = (E − N) + 2
where E = the number of flow graph edges. i.e. 15
     N = the number of flow graph nodes. i.e. 13

V(G) = (15 − 13) + 2 = 4.

3. Cyclomatic complexity V(G) = P + 1
where P = the number of predicate nodes contained in the flow graph G.

V(G) = 3 + 1 = 4.

There will be 4 independent Paths.

3) INDEPENDENT PATHS

Path A : 1 − 2 − 3 − 7 − 8 − 9 − 10 − 11 − 13

Path B : 1 − 2 − 4 − 5 − 7 − 8 − 9 − 10 − 12 − 13

Path C : 1 − 2 − 4 − 6 − 7 − 8 − 9 − 10 − 11 − 13

Path D : 1 − 2 − 3 − 7 − 8 − 9 − 10 − 12 − 13

CHAPTER – 9
CONCLUSION

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 Hospital Management System and interactivity making it more reliable and useful.

This project focused that scheduling a project and adhering to that schedule creates a hard sense of time- management. It has also let us known that co-operative teamwork always produce effective results.

The entire project has been developed and deployed as per the requirements stated by the user. It is found to be bug free as per the testing standards that are implemented.

The estimated cost of the project is (efforts) 12 and the estimated size of the project is (FP) 209.72.

There are also few features which can be integrated with this system to make it more flexible. Below list shows the future points to be consider :
- Getting the current status of patient.
- Including a different module for pharmacy, LAB, Bed Allotment and many more.
- Including a Frequently Asked Questions Section.

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.