

LAB REPORT

Submitted by

DISHITA SIBAL [RA2011003011162]

Under the Guidance of

Dr. M Aruna

Assistant Professor, Computing Technologies

In partial satisfaction of the requirements for the degree of

**BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING**



SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

JUNE 2022



**SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203**

BONAFIDE CERTIFICATE

Certified that this lab report titled **“Hospital Management System ”** is the bonafide work done by Dishita Sibal (RA2011003011162) who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Dr. M Aruna

SEPM-Course Faculty

Assistant professor,

Computing Technologies

ABSTRACT

Our project Hospital Management system includes registration of patients, storing their details into the system, and also booking their appointments with doctors. Our software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. User can search availability of a doctor and the details of a patient using the id. The Hospital Management System can be entered using a username and password. It is accessible either by an administrator or receptionist. 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. It is having mainly two modules. One is at Administration Level and other one is of user I.e. of patients and doctors. The Application maintains authentication in order to access the application. Administrator task includes managing doctors information, patient's information. To achieve this aim a database was designed one for the patient and other for the doctors which the admin can access. The complaints which are given by user will be referred by authorities. The Patient modules include checking appointments, prescription. User can also pay doctor's Fee online.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	
	LIST OF FIGURES	
	LIST OF ABBREVIATIONS	
1	PROBLEM STATEMENT	
2	STAKEHOLDERS & PROCESS MODELS	
3	IDENTIFYING REQUIREMENTS	
4	PROJECT PLAN & EFFORT	
5	WORK BREAKDOWN STRUCTURE & RISK ANALYSIS	
6	SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM	
7	ENTITY RELATIONSHIP DIAGRAM	
8	DATA FLOW DIAGRAM	
9	SEQUENCE & COLLABORATION DIAGRAM	
10	DEVELOPMENT OF TESTING FRAMEWORK/USER INTERFACE	
11	TEST CASES & REPORTING	
12	ARCHITECTURE/DESIGN/Framework/IMPLE- -MENTATION	
	CONCLUSION	
	REFERENCES	
	APPENDIX (CODE)	

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
2.1	AGILE MODEL	
5.1	GANTT CHART	
5.2	SWOT	
6.1	SYSTEM ARCHITECTURE	
6.2	USE CASE DIAGRAM	
6.3	CLASS DIAGRAM	
7.1	ENTITY RELATION	
8.1	DFD DIAGRAM LEVEL 0	
8.2	DFD LEVEL 1	
8.3.1	DFD_LEVEL 2(REGISTRATION)	
8.3.2	DFD LEVEL 2 (LOGIN)	
8.3.3	DFD_LEVEL 2(APPOINTMENT)	
8.3.4	DFD_LEVEL_2(ADD PRESCRIPTION)	
8.3.5	DFD_LEVEL_2(DOCTOR MODULE)	
8.3.6	DFD LEVEL 2(PAYMENT)	

8.3.7	DFD_LEVEL_2(CANCEL APPOINTMENT)
8.3.8	DFD_LEVEL_2(PATIENT MODULE)
9.1	SEQUENCE DIAGRAM
9.2	COLLABORATION DIAGRAM
12.1	MODULE IMPLEMENTATION

LIST OF TABLES

TABLE NO.	TOPIC NAME	PAGE NO.
2.1	STAKEHOLDER	
4.1	PROJECT	
4.2	EFFORT AND COST ESTIMATION	
4.3	INFRASTRUCTURE/ RESOURCE COST	
4.4	MAINTAINENCE AND SUPPORT COST	
4.5	PROJECT TEAM FORMATION	
4.6	RESPONSIBILITY ASSIGNMENT MATRIX	
5.1	WORK BREAKDOWN STRUCTURE	
5.2	RISK ID REGISTER	
10.1	TYPES OF TESTING	
11.1	FUNCTIONAL TEST CASE	
11.2	NON-FUNCTIONAL TEST CASE	
11.3	DEFECT LOG	

CHAPTER 1

PROBLEM STATEMENT

In this busy world, we don't have the time to wait in infamously long hospital queues. The problem is, that queuing at hospitals 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 and 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 centres, 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

1. Helps patients cut the long queue and saves their time
2. Is equipped with features like automated email and text message

reminders

- Role-Based Access Control

1. Allows employees to access only the necessary information to effectively
2. perform their job duties

- Overall cost reduction

1. Cuts down paper costs as all the data are computerized
2. No separate costs for setting up physical servers

- Data accuracy

1. Removes human errors
 2. Alerts when there's a shortage of stock
- Data security
 1. Helps to keep patients records private
 2. Restricts access through role-based access control
 - Revenue management
 1. Makes daily auditing simple
 2. Helps with statistics and other financial aspects

THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

- The project, Hospital Management System is a web-based application that allows the Hospital in-charge to handle all hospital activities online.
- The mission is to facilitate easy management and administration of a hospital with capabilities to do Booking or reservations of the doctors, Cancellation of the doctors, Cash billing, Bill service, virtual and real service, Total Billing etc. using the automated hospital management software.
- One can Keep detailed records or info on an unlimited amount of customers.
- The system lets the user Know which all doctors are available at any point of time.
- This makes the Booking considerably faster. And thus helps the hospital in better management and reduce a lot of paper work as well as manpower.

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.

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.

THE HISTORY

In bullet points, describe the current situation.

- In the existing system, only provides the information about particular doctor and only some of the doctors has possibility to reserve the rooms.
- The previous system was failure to publish the doctor's availability of multiple hotels.
- it is failure to provide defence to admin and user information.

LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

- Human challenges:
- Awareness of HMS advantages & importance.
- In general, Experience, and knowledge of using computer applications.
- Impressions and Beliefs regarding HMS and making use of them efficiently.
- Shortage of professional healthcare faculty who have in-depth knowledge of HMS and other similar technologies.
- Poor acceptance of HMS Software.
- Shortage of health informatics professionals who are well capable of establishing and implementing the techniques.
- Technical Challenges:

- Other few technical challenges that fail the implementation of HMS in the healthcare industry includes Networks and computer have different maintenance problems, lack of no standards for Data entry and data retrieval, difficulties in training users technically to use HMS.

APPROACH

List what is needed to complete the project.

- Software Requirements:
- It's better to choose a software that is easy to upgrade, scalable and is not hardware intensive. Cloud-based online software is better than offline software in these aspects. The providers could also easily fix any errors or bugs in case of online software.
- Having Trusted Hospital

OVERVIEW

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.

BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

- An accurate Hospital Management System ensures that decision making in clinics is made precise, fast, and useful. With a single view and easy availability of data points, medical support staff and doctors get facilitated.

- tracking the accurate details of staff availability, operational information, and room occupancy can also be readily available at ones fingerprints by using the automated Hospital Information System.
- Installing a complete and automated HMS indicated that hospital's pieces of the information stay safe and protected from any unauthorized sources and accesses.
- A manual system can ensure 100% accurate processing and full proof. There are higher chances for mistakes and errors in this case. Here, hospital management system nullifies every mistake and also you can avoid lawsuits and compliance issues, which is considered to be the two most significant drawbacks of hospitals and medical centers.

CHAPTER 2

STAKEHOLDERS & PROCESS MODELS

Software Development Life Cycle Model

A software development life cycle (SDLC) model is a conceptual framework describing all activities in a software development project from planning to maintenance. This process is associated with several models, each including a variety of tasks and activities.

AGILE MODEL

The Agile model was selected as the SDLC model due to the following reasons:

Advantages of Agile model:

- Customer satisfaction by rapid, continuous delivery of useful software.
- People and interactions are emphasized rather than process and tools. Customers, developers and testers constantly interact with each other.
- Working software is delivered frequently (weeks rather than months).
- Face-to-face conversation is the best form of communication.
- Close, daily cooperation between business people and developers.
- Continuous attention to technical excellence and good design.
- Regular adaptation to changing circumstances.
- Even late changes in requirements are welcomed

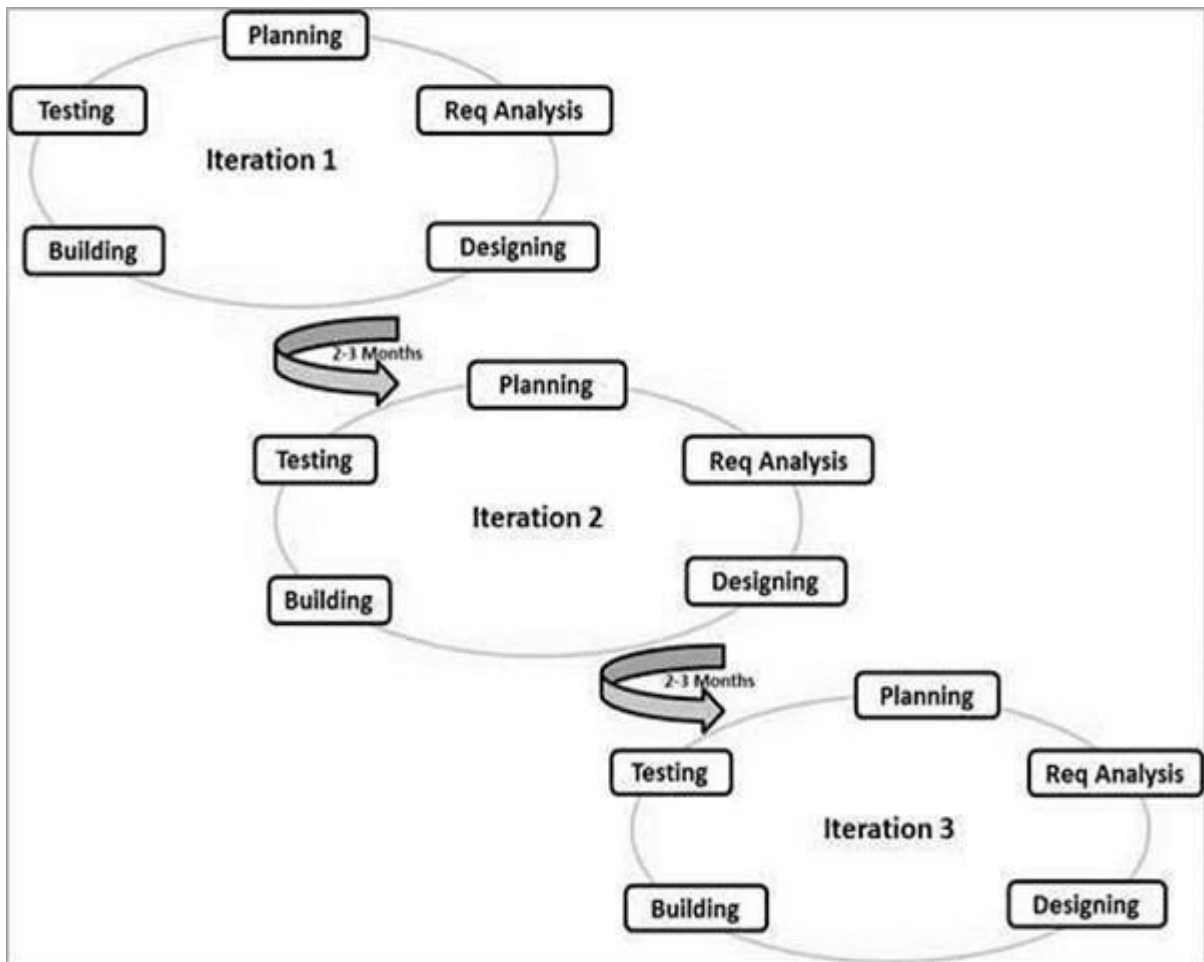


Fig 2.1 AGILE MODEL

2.1) STAKEHOLDER

Stakeholder Name	Activity/ Area /Phase	Interest	Influence	Priority (High/ Medium/ Low)
Owner	Minimizing cost and increasing profits, Ensuring compliance with local regulations	High	High	1
Nurses	Access to EMR in timely manner, Easy input of patient data and documentation, compatible with	High	Med	4
	other hospital systems			

Doctors	Providing exact information, Acquiring information swiftly	High	High	2
Patients	Provides Feedback for the Hospital, Appointment scheduling, Charting, Information Communication, Prescribing, Billing and Insurance claims	High	High	2
Office managers	Managing appointments and patient flow, Billing	Med	Low	7

CHAPTER 3

IDENTIFYING REQUIREMENTS

Functional Requirements:

There are a lot of software requirements specifications included in the functional requirements of the Hospital Management System, which contains various process, namely Registration, Check out, Report Generation, and Database.

Registration Process of SRS (Software Requirements Specification)

- Adding Patients: The Hospital Management enables the staff in the front desk to include new patients to the system.
- Assigning an ID to the patients: The HMS enables the staff in the front desk to provide a unique ID for each patient and then add them to the record sheet of the patient. The patients can utilize the ID throughout their hospital stay.

Check Out of SRS:

- Deleting Patient ID: The staff in the administration section of the ward can delete the patient ID from the system when the patient's checkout from the hospital.
- Adding to beds available list: The Staff in the administration section of the ward can put the bed empty in the list of beds-available.

Report Generation of SRS:

- Information of the Patient: The Hospital Management System generates a report on every patient regarding various information like patients name, Phone number, bed number, the doctor's name whom its assigns, ward name, and more.
- Availability of the Bed: The Hospital Management system also helps in generating reports on the availability of the bed regarding the information like bed number unoccupied or occupied, ward name, and more.

Database of SRS:

- Mandatory Patient Information: Every patient has some necessary data like phone number,

their first and last name, personal health number, postal code, country, address, city, 'patient's ID number, etc.

- Updating information of the Patient: The hospital management system enables users to update the information of the patient as described in the mandatory information included.

Non Functional Requirements

There are a lot of software requirements specifications included in the non-functional requirements of the Hospital Management System, which contains various process, namely Security, Performance, Maintainability, and Reliability.

Security:

- Patient Identification: The system needs the patient to recognize herself or himself using the phone.
- Logon ID: Any users who make use of the system need to hold a Logon ID and password.
- Modifications: Any modifications like insert, delete, update, etc. for the database can be synchronized quickly and executed only by the ward administrator.
- Front Desk Staff Rights: The staff in the front desk can view any data in the Hospital Management system, add new patients record to the HMS but they don't have any rights alter any data in it.
- Administrator rights: The administrator can view as well as alter any information in the Hospital Management System.

Performance:

- Response Time: The system provides acknowledgment in just one second once the 'patient's information is checked.
- Capacity: The system needs to support at least 1000 people at once.
- User-Interface: The user interface acknowledges within five seconds.
- Conformity: The system needs to ensure that the guidelines of the Microsoft accessibilities are

followed.

Maintainability:

- Back-Up: The system offers the efficiency for data back up.
- Errors: The system will track every mistake as well as keep a log of it.

Reliability:

- Availability: The system is available all the time.

CHAPTER 4

PROJECT PLAN AND EFFORT

4.1)Project Plan

Focus Area	Details
Integration Management	<p>Project Team Structure :</p> <p>Team leader – Dishita Sibal</p> <p>Team members – Dishita Sibal, Anushree Upadhyaya</p> <p>Roles & Responsibilities of Team :</p> <ol style="list-style-type: none">1) User/Admin Interface Module – Dishita Sibal2) User Authentication Module – Dishita Sibal3) Database Management Module – Anushree Upadhyaya4) Website Component Module – Anushree Upadhyaya5) Integration Module –Anushree Upadhyaya, DIshita Sibal <p>Change Management : Team Leader</p> <p>Project Closure : Ms. Aruna</p>
Scope Management	<p>Scope Statement</p> <p>Requirement Management (Gathering, Control, Assumption, Constraint Stakeholder)</p> <p>Define Deliverable</p> <p>Requirement Change Control</p> <p>Activities and Sub-Tasks</p>
Schedule Management	<p>Milestones :</p> <ul style="list-style-type: none">• PLANNING – April 2022• CODING – May 2022• WEBSITE FOR ONLINE REG. - June 2022• PRESENTATION – July 2022• TESTING- July 2022 <p>Schedule Control :</p> <p>April 2022 to July 2022</p>
Cost Management	<p>Estimate Effort</p> <p>Assign Team</p> <p>Budget Control</p>

Quality Management	Improve budgeting, as strong project management directly aligns resources with important work. Increase staff productivity. Proper booking management. Proper Customer Service
Resource Management	Estimate and Manage the need People: People & Skills Required Finance: Budget Required Physical: Facilities, IT Infrastructure
Stakeholder	Identifying, Analyzing, Engaging Stakeholders
Communication Management	Determine communication requirements, roles and responsibilities, tools and techniques. [Type of Communication, Schedule, Mechanism Recipient]
Risk Management	Identifying, analysing, and prioritizing project risks
Procurement Management	Adhering to organization procurement process

Estimation

4.2)Effort and Cost Estimation

Activity Description	Sub-Task Description	Effort (in hours)	Cost in INR
Design the user screen	Confirm the user requirements (acceptance criteria)	3	1500
	Customer Service	2	1000
User/Admin Interface Module	Allow admin/user to access the data	2	1000
User Authentication Module	Checks the authenticity of the user	2	1000
Database Management Module	Stores all the user information and related data.	4	2000

Website Component Module & Integration		All the features of the managing system	5	2500
Effort (hr)	Cost (INR)			
1	500			

4.3)Infrastructure/Resource Cost [CapEx]

< OneTime Infra requirements >

Infrastructure Requirement	Qty	Cost per qty	Cost per item
PC	3	1,00,000	3,00,000
Chair	3	5,000	15,000
Tables	3	10,000	30,000

4.4)Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin Developer , Support Consultant	3	2,000,000	6,000,000
License	Operating System Database Middleware IDE	10	10000	100,000
Infrastructures	Server, Storage and Network	20	20000	400,000

4.5)Project Team Formation

Identification Team members

Name	Role	Responsibilities
Dishita Sibal	Key Business User (Product Owner)	Provide clear business and user requirements
Dishita Sibal	Project Manager	Manage the project
Anushree Upadhyaya, Dishita Sibal	Business Analyst	Discuss and Document Requirements
Dishita Sibal	Technical Lead	Design the end-to-end architecture

Dishita Sibal	UX Designer	Design the user experience
Anushree Upadhyaya	Frontend Developer	Develop user interface
Anushree Upadhyaya, Dishita Sibal	Backend Developer	Design, Develop and Unit Test Services/API/DB
Anushree Upadhyaya	Cloud Architect	Design the cost effective, highly available and scalable architecture
Anushree Upadhyaya	Cloud Operations	Provision required Services
Anushree Upadhyaya, Dishita Sibal	Tester	Define Test Cases and Perform Testing

4.6)Responsibility Assignment Matrix

RACI Matrix	Team Members			
Activity	Dishita Sibal , Anushree Upadhyaya(Developer & BA)	Anushree Upadhyaya(Developer)	Anushree Upadhyaya,Dishita Sibal (Project Manager)	Dishita Sibal Key Business User
User Requirement Documentation	C/A	C/I	I	R
Authentication	C	I	A&R	I
Graphics and Visual	I	A&R	C	C
Database Management	A&R	I	A	I
Project Management	C	C	A	I
Testing	R	I	A	C

A= Accountable	R= Responsible	I= Inform	C= Consult
----------------	----------------	-----------	------------

Chapter 5

Work breakdown Structure and Risk Analysis

Timeline Gantt Chart



5.1 GANTT CHART

5.1) Work breakdown

Work Breakdown Structure With Project Schedule

< Assign team members for sub-tasks based on RACI and skill requirement >

Module (#)	Activity (#)	Sub Task(#)	Assignee(s)	Planned Start Date	Planned End Date	Actual Start Date	Actual End Date	Status
1	Specification	Feasibility study	Dishita Sibal & Anushree Upadhyaya	9/03/22	11/03/22	9/03/22	11/03/22	Completed
2	Planning	Requirement analysis	Dishita Sibal & Anushree Upadhyaya	17/03/22	21/03/22	17/03/22	22/03/22	Completed

3	Planning	Methodology Identification	Dishita Sibal & Anushree Upadhyaya	31/03/22	7/04/22	31/04/22	7/04/22	Completed
4	Planning	Process Model Identification	Dishita Sibal & Anushree Upadhyaya	8/04/22	15/04/22	8/04/22	20/04/22	Completed
5	Analysis	Risk management	Dishita Sibal & Anushree Upadhyaya	2/05/22	10/05/22	2/05/22	10/05/22	Completed
6	Design	Architecture & abstract design	Dishita Sibal & Anushree Upadhyaya	2/05/22	10/05/22	Not yet started	-	To be done
7	Implementation	Interface design(Front end)	Dishita Sibal & Anushree Upadhyaya	15/05/22	23/05/22	Not yet started	-	To be done
8	Implementation	Component design	Dishita Sibal & Anushree Upadhyaya	1/06/22	10/06/22	Not yet started	-	To be done
9	Development	Back end development	Dishita Sibal & Anushree Upadhyaya	10/06/22	16/06/22	Not yet started	-	To be done
10	Validation	Testing the software	Dishita Sibal & Anushree Upadhyaya	17/06/22	24/06/22	Not yet started	-	To be done

5.2)Risk Id Register

Risk Id (#)	Risk Description	Impact Description
R01	Strategic	A competitor coming on to the market
R02	Financial	increased interest charges on a business loan
R03	Operational	The breakdown or theft of key equipment
R04	IT System	Hardware and software failure
R05	Security	Loss of confidential information
R06	Server Down	Loss of customers
R07	Complexity	Section will break down
R08	Schedule slippage	Outburst among customer due to delay in scheduled updates

R03	Close	Avoid	Completed	Dishita & Anushree	15/06/2022
R04	Open	Accept	Onprocess	Anushree & Dishita	15/06/2022
R05	Close	Avoid	Completed	Dishita & Anushree	15/06/2022
R06	Close	Accept	Completed	Anushree & Dishita	15/06/2022
R07	Open	Accept	Onprocess	Dishita & Anushree	15/06/2022
R08	Close	Accept	Completed	Anushree & Dishita	15/06/2022
R02	Close	Accept	Completed	Dishita & Anushree	15/06/2022

Swot Analysis

SWOT Analysis



STRENGTHS:

- High success rate
- Continuous investment in clinical equipment
- Personalized care
- A strong brand name
- Niche patient base
- Excellent location



WEAKNESSES:

- Limited number of physicians
- Longer patient wait times
- Bad financial management
- High staff turnover
- Inability to attract new patients



OPPORTUNITIES:

- Provide diagnostic and lab services to other medical practices
- Attract quality physicians and staff
- Expansion to underserved areas
- Convenient parking
- Cost-effectiveness
- Successful referral programs with other physicians



THREATS:

- With expansion, you may face challenges with patient experience and efficient communication
- Stiff competition
- Staff dissatisfaction
- Security breaches
- Dissatisfied patients

Fig 5.2 SWOT

Chapter 6

System Architecture Use Case and Class Diagram

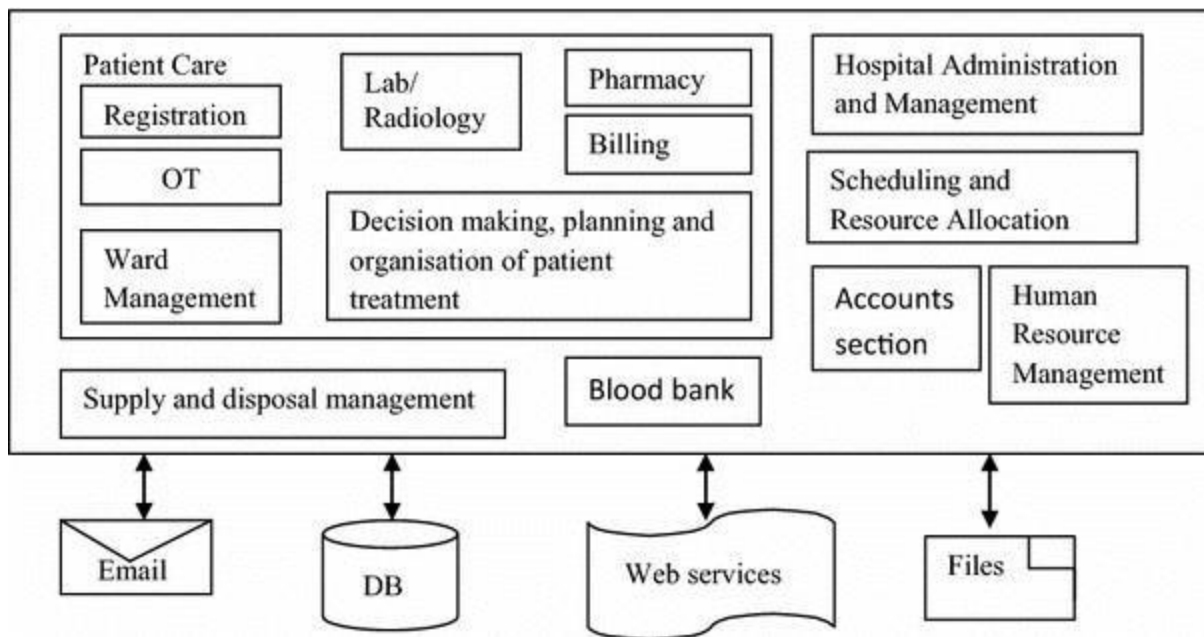


Fig 6.1 System Architecture

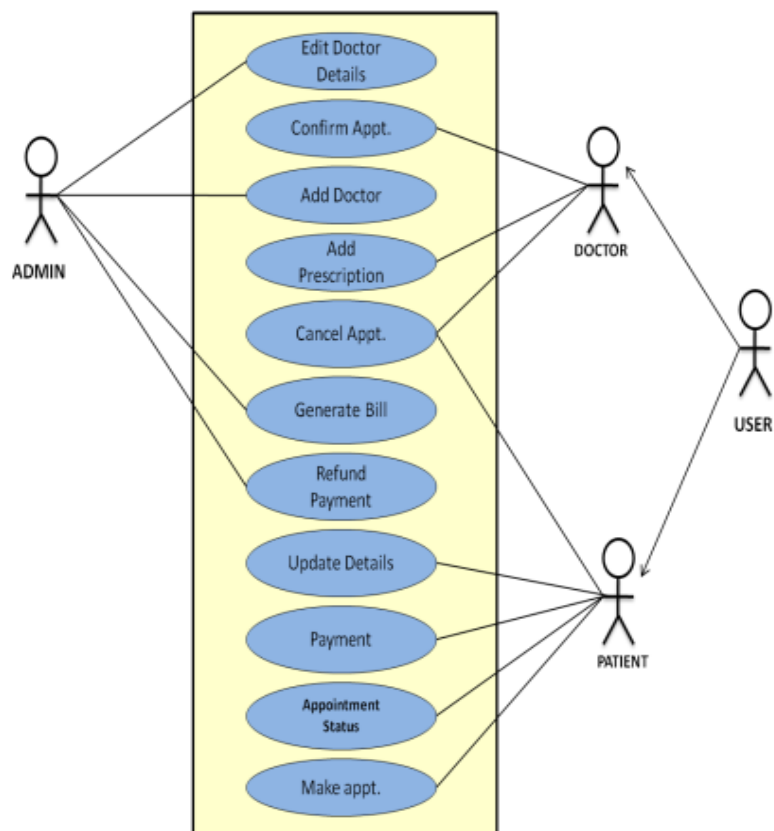


Fig 6.2 USE CASE

HOSPITAL MANAGEMENT SOFTWARE - CLASS
DIAGRAM

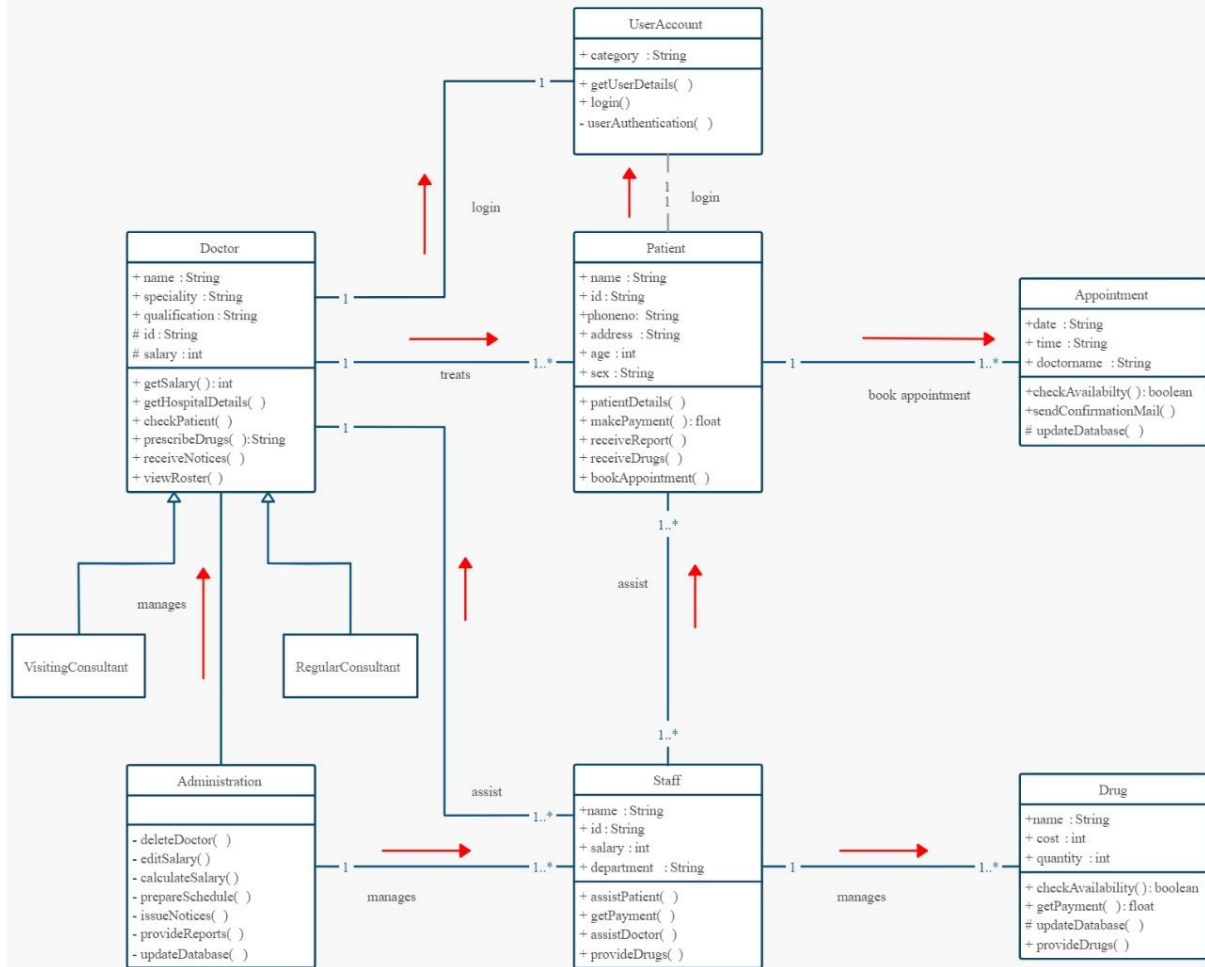


Fig 6.3 CLASS DIAGRAM

Chapter 7

Entity Relationship Diagram

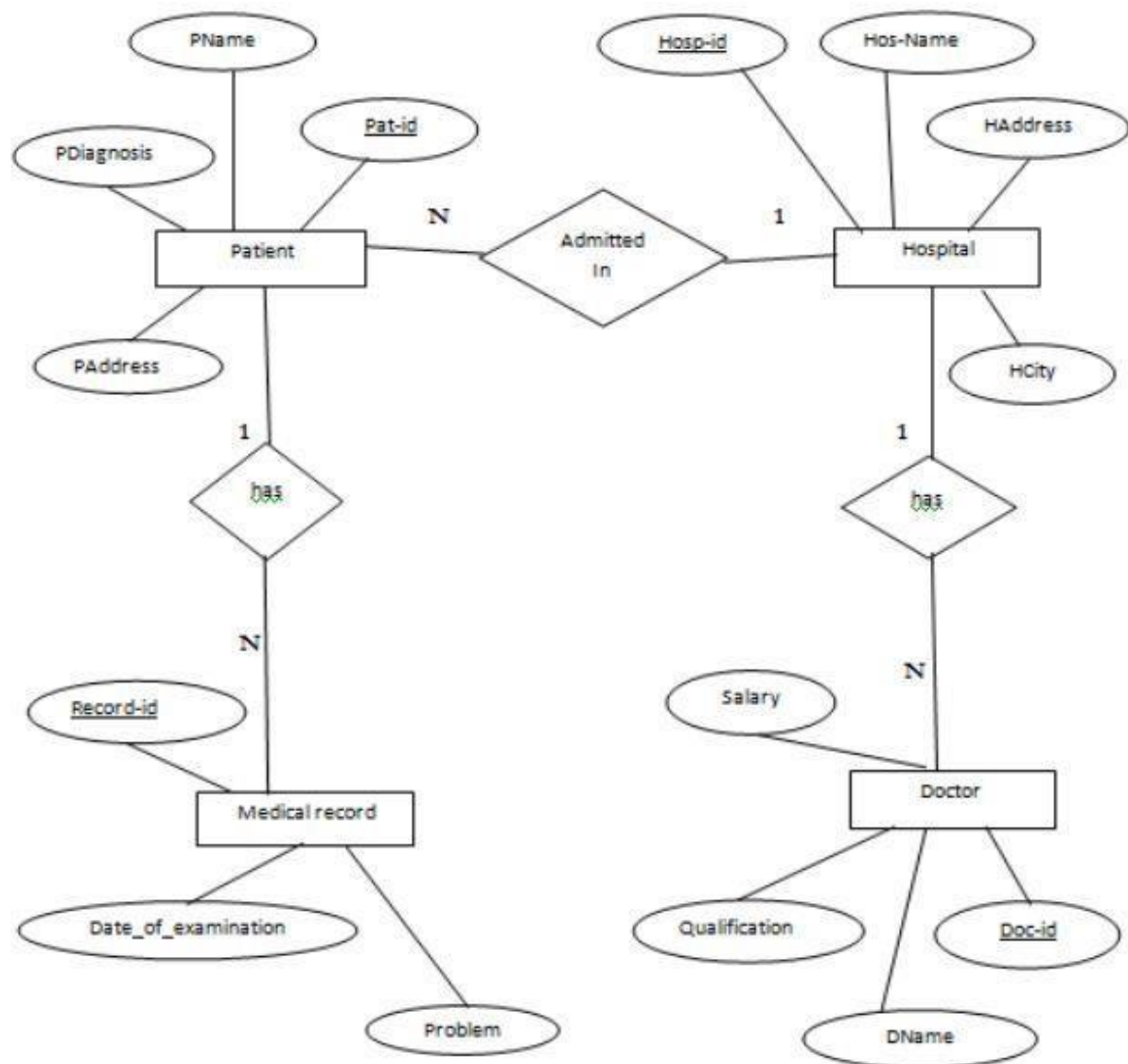


Fig 7.1 ENTITY RELATIONSHIP DIAGRAM

Chapter 8

Data Flow Diagram

Context Level

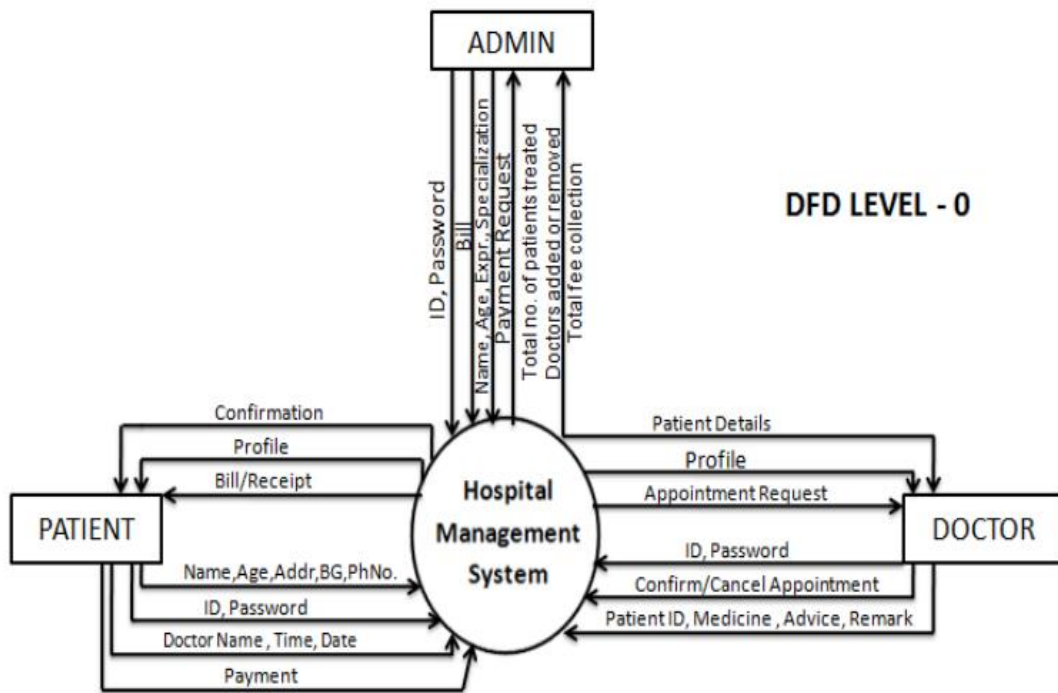


Fig 8.1 DFD LEVEL 0

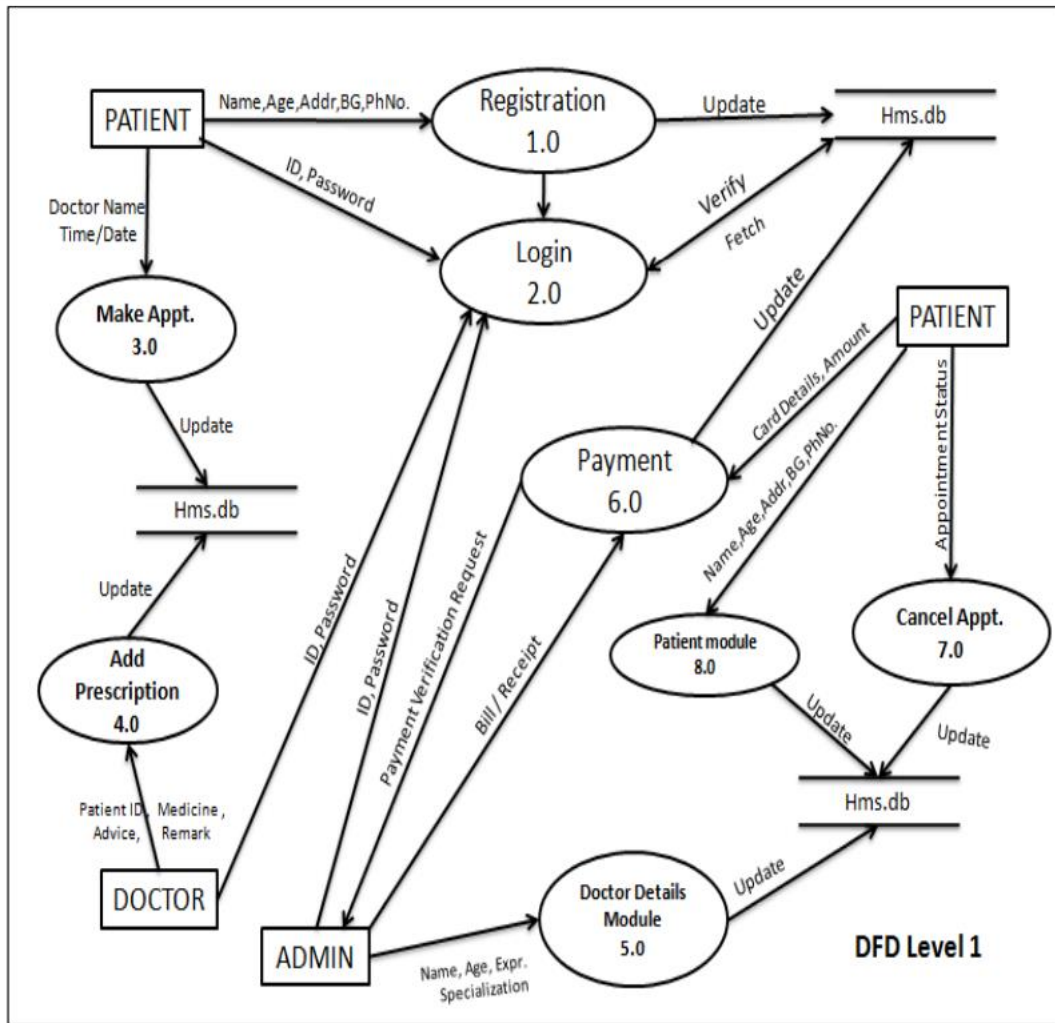


Fig 8.2 DFD LEVEL 1

DFD level 2(Registration)

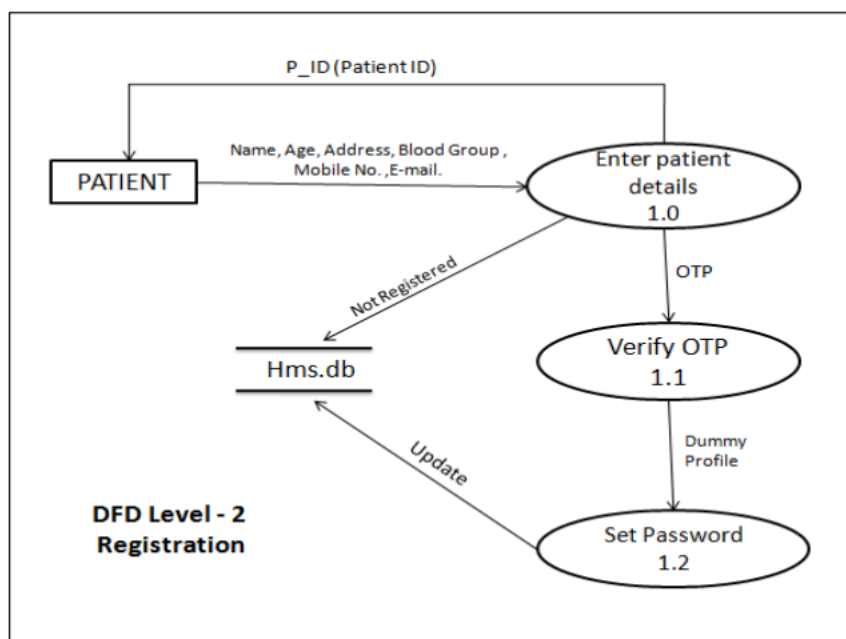


Fig 8.3.1)LEVEL 2 REGISTRATION

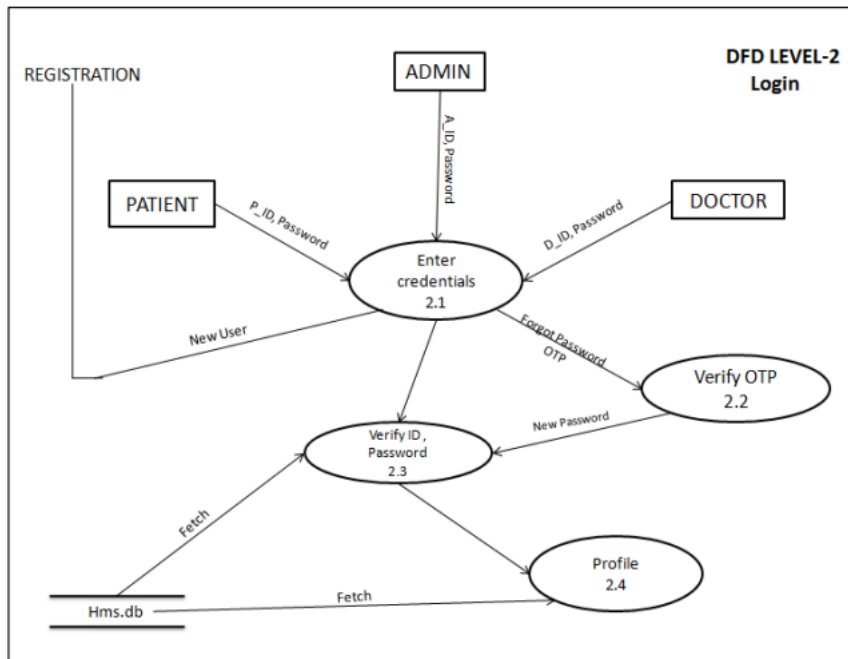


Fig 8.3.2)DFD level 2(LOGIN)

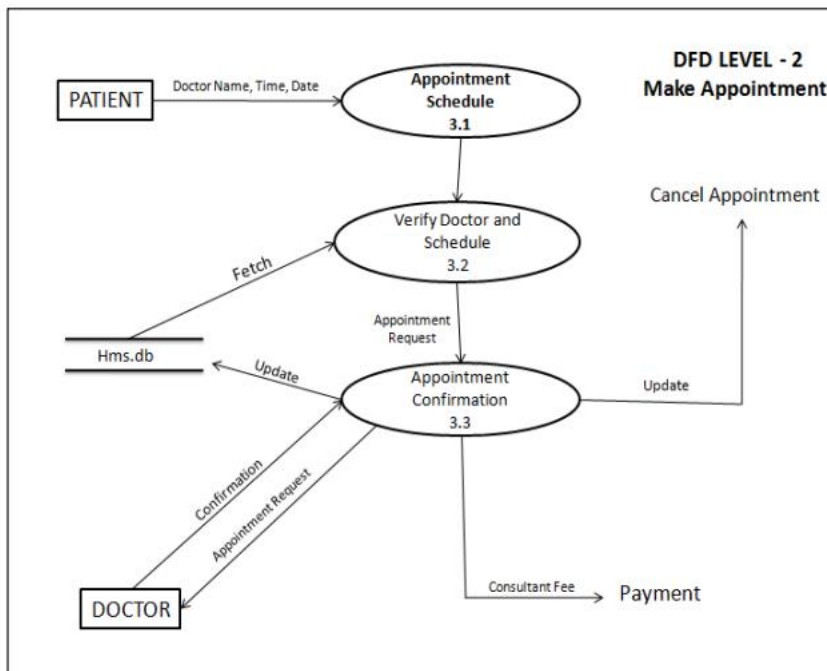


Fig 8.3.3)DFD I(APPOINTMENT)

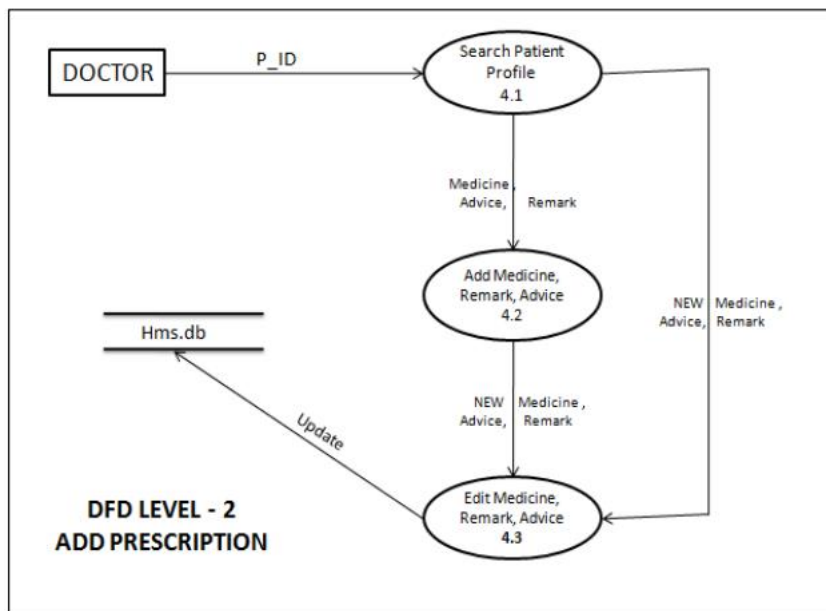


Fig 8.3.4)DFD level 2(ADD PRESCRIPTION)

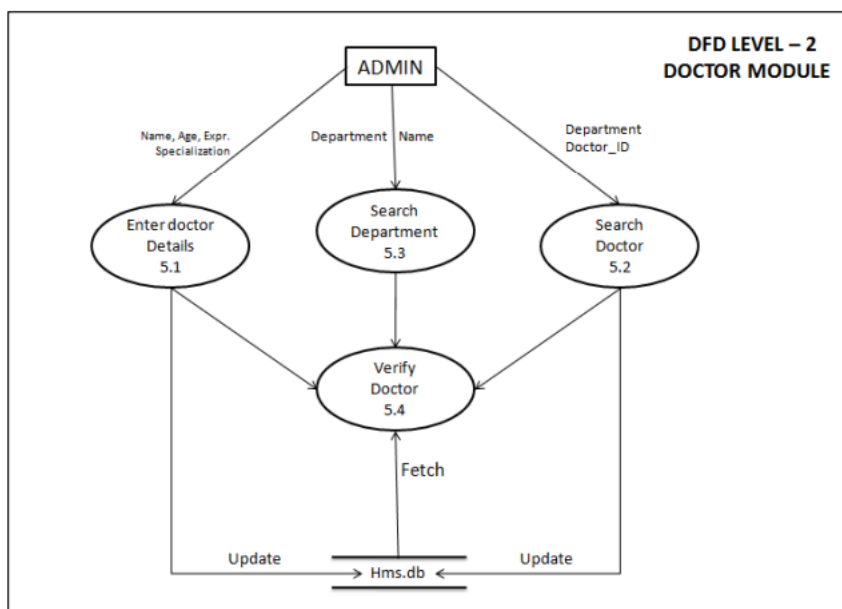


Fig 8.3.5)DFD level 2 (DOCTOR MODULE)

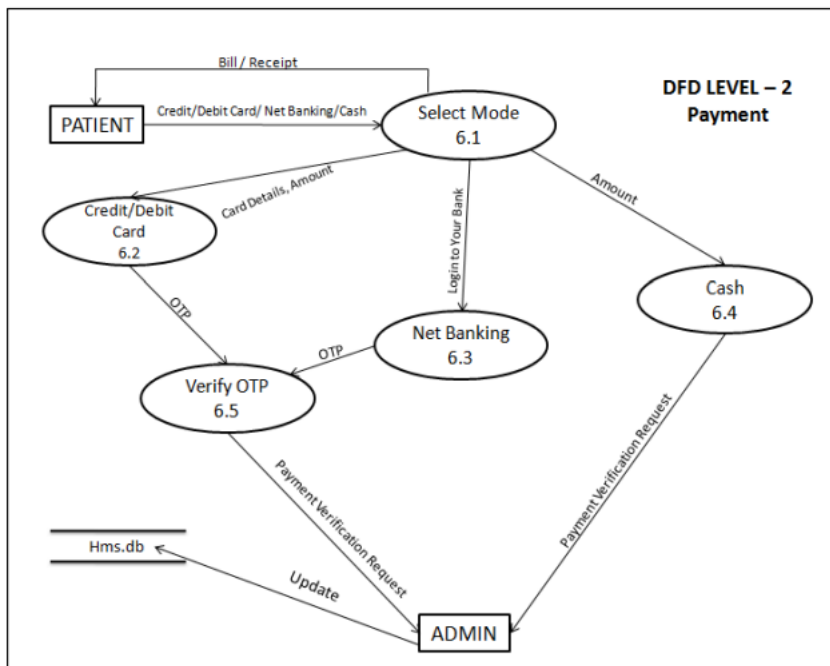


Fig 8.3.6)DFD level 2(PAYMENT)

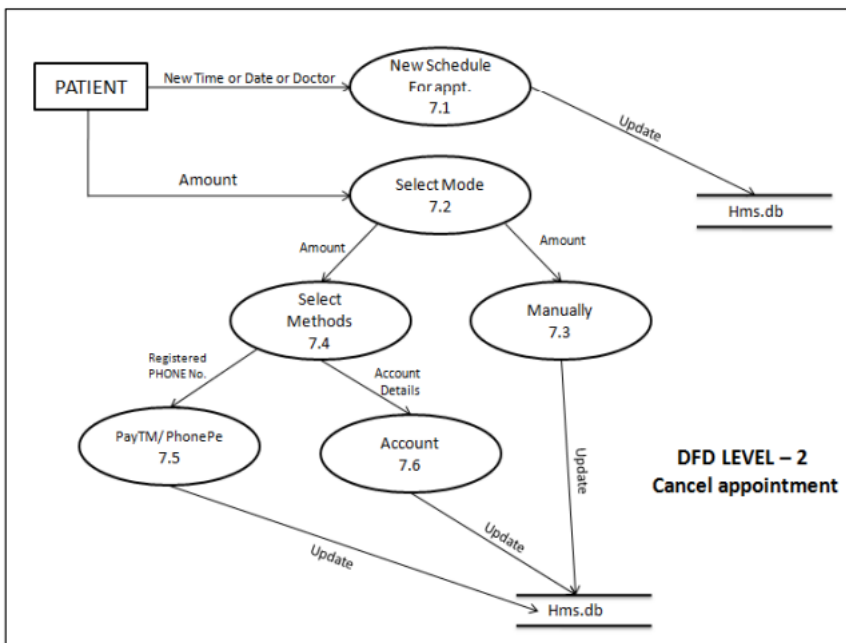


Fig 8.3.7)DFD level 2 (CANCEL APPOINTMENT)

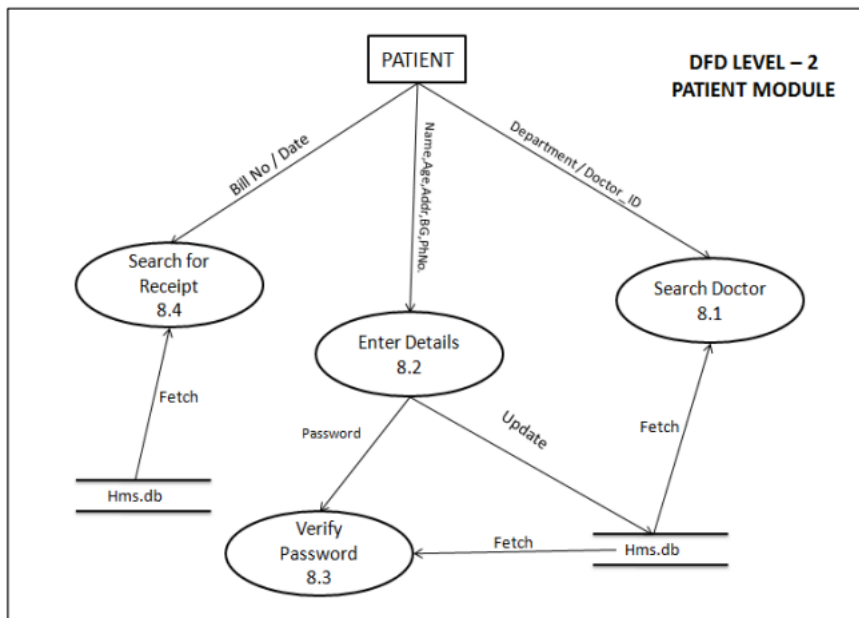


Fig 8.3.8) DFD LEVEL2(PATIENT MODULE)

Chapter 9

Sequence and Collaboration diagram

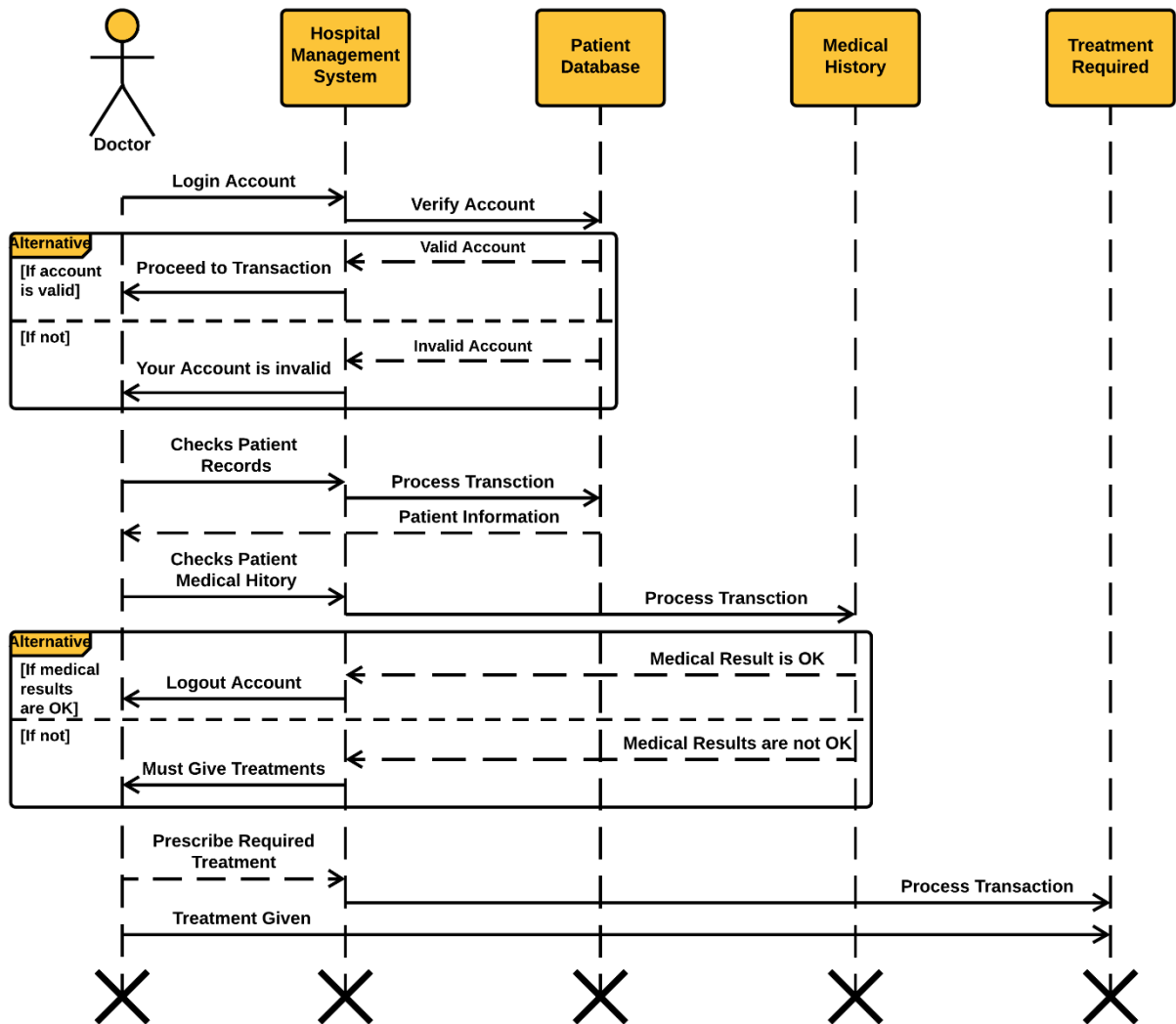


Fig 9.1 Sequence Diagram

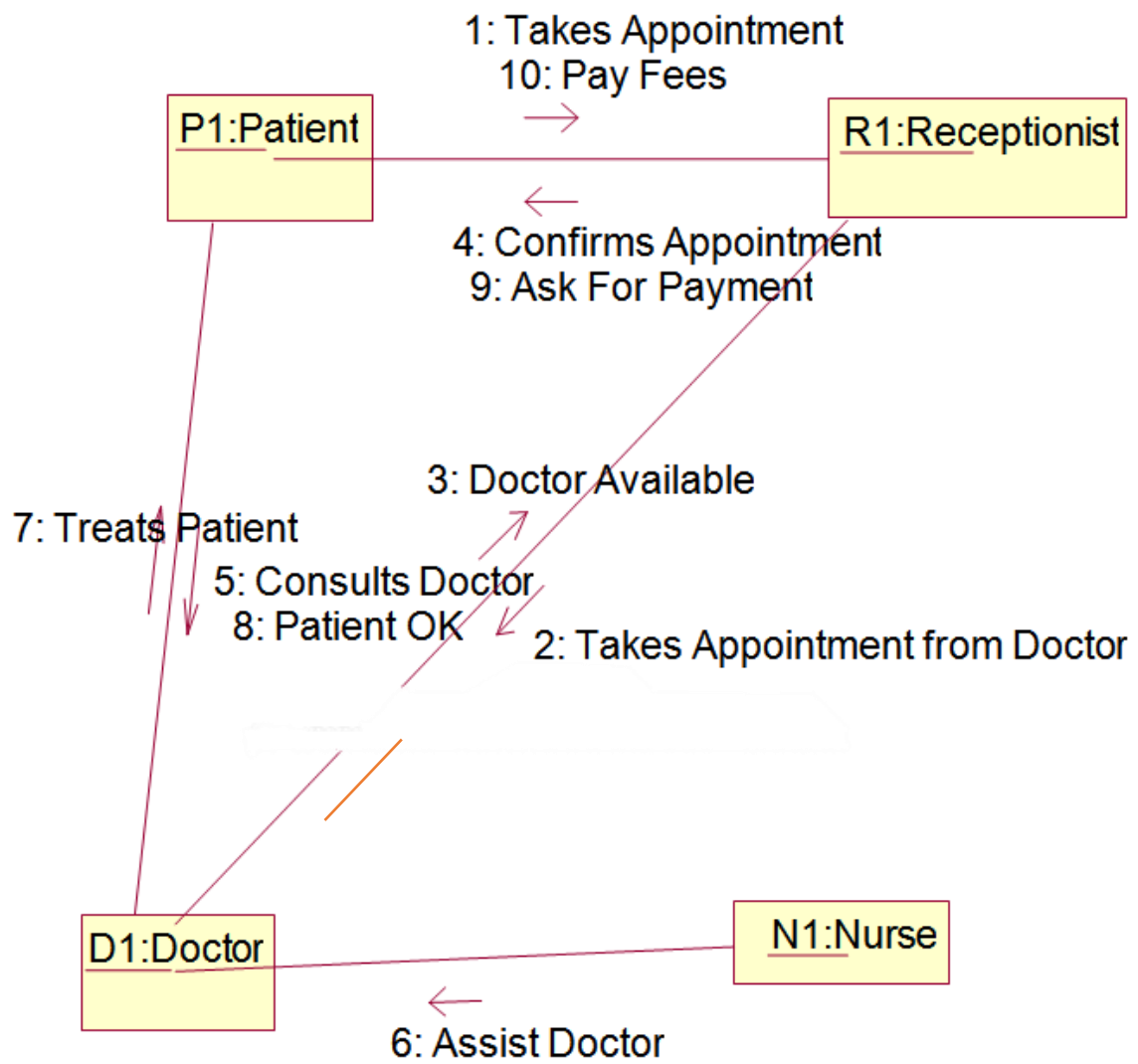


Fig 9.2 Collaboration Diagram

Chapter 10

Development of Testing Framework/ User Interface

Executive Summary

Scope of testing

Technically scope is that Software Testing is an investigation conducted to provide shareholders with information about the quality of a particular product or service under test. In other words, software testing is a process of verification and validation.

To test software, a tester is expected to know the software that is to be tested, plus the required to test it. Often, it can be very specialized, and your employer company may you. To be a good tester, you need to be very organized, logical, meticulous, thorough, and patient.

The objective of testing

to evaluate the work products such as requirements, design

1. To validate if the test object is complete and works as per the expectation of the users and the stakeholders
2. To find and prevent defects in the software product
3. To reduce the level of risk of insufficient software quality
4. To provide sufficient information to stakeholders to allow them to make informed decisions, especially regarding the level of quality of the test object

Test Plan

10 A)Types of Testing , Methodology , Tools

Category	Methodology	Tools Required
Functional Requirements	Manual	Word Template
Non Functional Requirements	Manual	Word Template

Chapter11

Test Cases

Test Case

11 A)Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
FT1	Verify User Login	Username-Valid, Password-Valid	<ol style="list-style-type: none"> 1. User clicks on User login link 2. Enter the email id and password on the text box 3. Click Login button 	Gives user authentication and should be taken to the next page	The outcome is same as the expected outcome	Pass	success
		Either the username or the password or both are invalid or blank	<ol style="list-style-type: none"> 1. User clicks on User login link 2. Enter the email id and password on the text box 3 Click Login button 	Displays error message	Actual outcome same as expected outcome	pass	success
FT2	Registration	User gives required user name and the details asked for	<ol style="list-style-type: none"> 1. Click on register 2. Creation of account 	Allow access to the user to the program manager	Actual outcome same as expected outcome	pass	success
FT3	Book appointment	Booking of the appointment by selecting the desired choice	<ol style="list-style-type: none"> 1. Click on the first option 2. Select required doctor disease and consultancy fees 3. Create entry 	Allow user to book an appointment for himself	Actual outcome same as expected outcome	pass	success

FT3.1	Cancel appointment	The user can cancel or keep the appointment	1 Click on cancel button	Should lead to cancellation of the booked appointment	Actual outcome same as expected outcome	pass	success
FT4	Payment of bill	User can pay the bill of the required amount of consultant	Go back to the homescreen Click on prescriptions	On selecting the pay bill the payment of the bill will be done	Actual outcome same as expected outcome	pass	success
FT5	Adding doctors or removing doctors	Helps the admin to remove or add the doctors	1. Click on the add doctors or remove doctors.	Based on the pre entered details	Actual outcome same as expected outcome	pass	success
FT6	Log out	logging out of the patient, doctor, admin account	Top of the screen has the option of logging out	On selecting the log out option the user is redirected to the login page	Same as expected	pass	success

11 B)Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
NFT 1	A help option(faq) shall be built, to help user get access to more detailed explanation and help user get a quick start to the		1. Access to a help button in	A FAQ page will appear with the required details for the issue	Data to be updated	In progress	In progress

	portal and how to be familiar to it.						
NFT 2	A attractive home page with clear icon for user and more dynamic view point		1. Data to appear on opening the home page	A mutliclo ur attractiv e homempa ge with distinct options will appear	Same as expecte d	pass	success
NFT 3	System shall be cut into several small independe nt part, in order to make modificatio ns easy	Different module for different functions		Separate code blocks for the coder to edit in fututre	Same output as expecte d	pass	success
NFT 4	System shall track users progress and calculate all useful data in time, in order to keep the informatio n valuable	Calculate lp address and other informati on		All the required data to be appeare d in the required module	Same output as expecte d	pass	success

11 C)Defect Log

Requirement #	Defect ID #	Defect Description	Assignee	Status
M1R1	DEF1	The data generated is computerized and the results are taken into consideration ingnoring payment of the bill option	<ul style="list-style-type: none"> Anushree Upadhyaya Dishita Sibal 	It is a app so the defect will remain until and unless we make a huge data base for thousands of individual

11 D)Test Report

Summary

The functional testing has been completed 100% within all the modules and the non functional testing is at a 70% completion with a few module left to be updated which will be completed with a few days and will be updated accordingly.

Category	Progress Against Plan	Status
Functional Testing	Green	Completed
Non-Functional Testing	Amber	In progress

Functional	Test Case Coverage (%)	Status
TEST ID FT1	100%	Completed
TEST ID FT2	100%	Completed
TEST ID FT3	100%	Completed
TEST ID FT3.1	100%	Completed
TEST ID FT4	100%	Completed
TEST ID FT5	100%	Completed
TEST ID FT6	100%	Completed

Non Functional	Test Case Coverage (%)	Status
TEST ID NFT1	30%	IN PROGRESS
TEST ID NFT2	100%	Completed
TEST ID NFT3	100%	Completed
TEST ID NFT4	100%	Completed

Chapter 12

Architecture/Design/Implementation

PROJECT-HOSPITAL MANAGEMENT SYSTEM

INTRODUCTION:

The IT system has revolutionised the field of medicine. In this fast-paced world of medicine, it is a daunting task to manage a multi-speciality hospital. A hospital management system (HMS) is a computer or web based system that facilitates managing the functioning of the hospital or any medical set up¹. This system or software will help in making the whole functioning paperless. It integrates all the information regarding patients, doctors, staff, hospital administrative details etc. into one software⁴. It has sections for various professionals that make up a hospital.

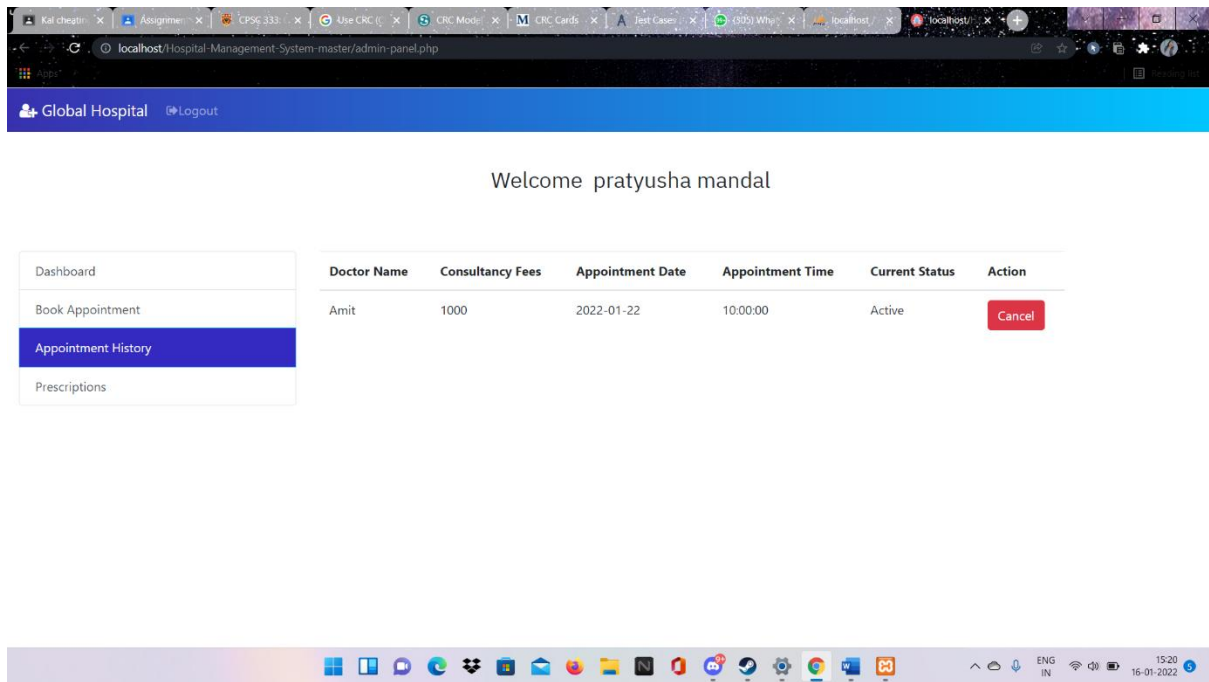
The screenshot displays the 'Create an appointment' form in the HMS. The sidebar on the left contains the following links: Dashboard, Book Appointment (highlighted), Appointment History, and Prescriptions. The main form area has the following fields:

- Specialization: Select Specialization (dropdown)
- Doctors: Select Doctor (dropdown)
- Consultancy Fees: (text input field)
- Appointment Date: dd-mm-yyyy (calendar icon)
- Appointment Time: Select Time (dropdown)
- Create new entry (button)

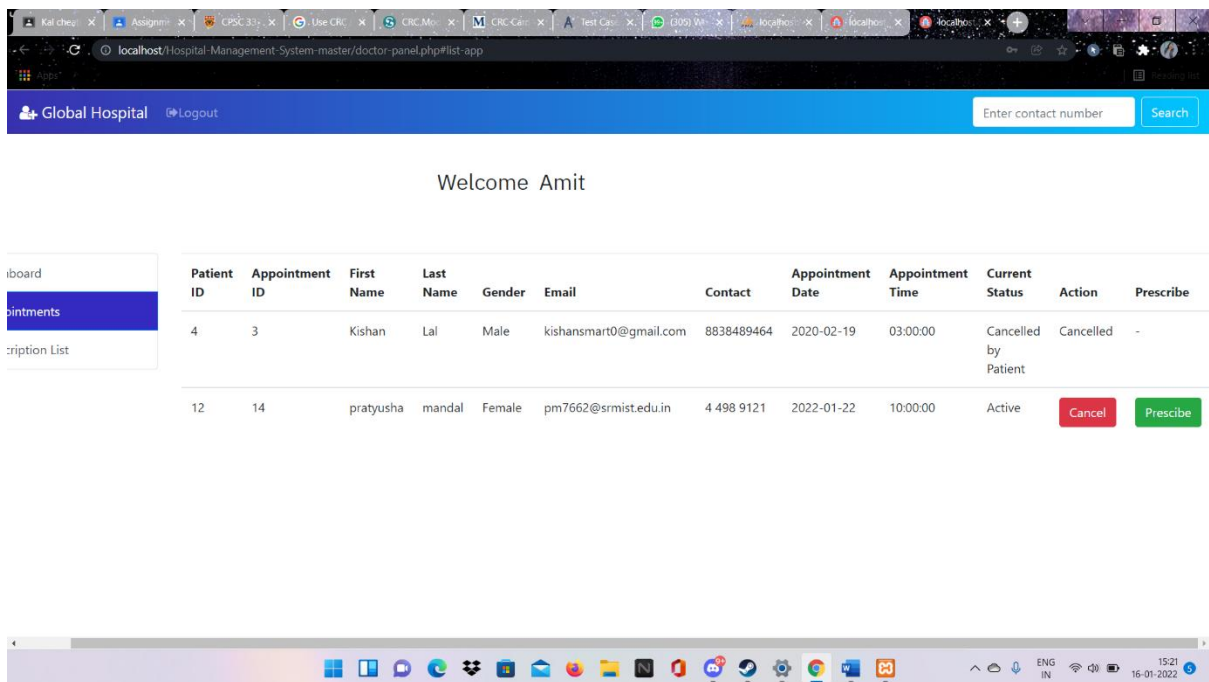
The browser's address bar shows the URL: localhost/Hospital-Management-System-master/admin-panel.php#list-home. The system's name 'Global Hospital' and a 'Logout' link are visible in the top navigation bar.

Fig 12.1 MODULE IMPLEMENTATION

This is the appointment booking page. This helps the patients to book the appointments that is needed by them.



This is the display of the appointment fixed



This is the doctor login page where appointments will be shown for him. He can either cancel or prescribe it

Global Hospital Logout

Enter contact number Search

Welcome Amit

Dashboard
Appointments
Prescription List

Patient ID	First Name	Last Name	Appointment ID	Appointment Date	Appointment Time	Disease	Allergy	Prescribe
12	pratyusha	mandal	14	2022-01-22	10:00:00	hello	cucumber,tomato	do not take these things

This is the prescribed medicines that is shown from the point of view of the doctor.

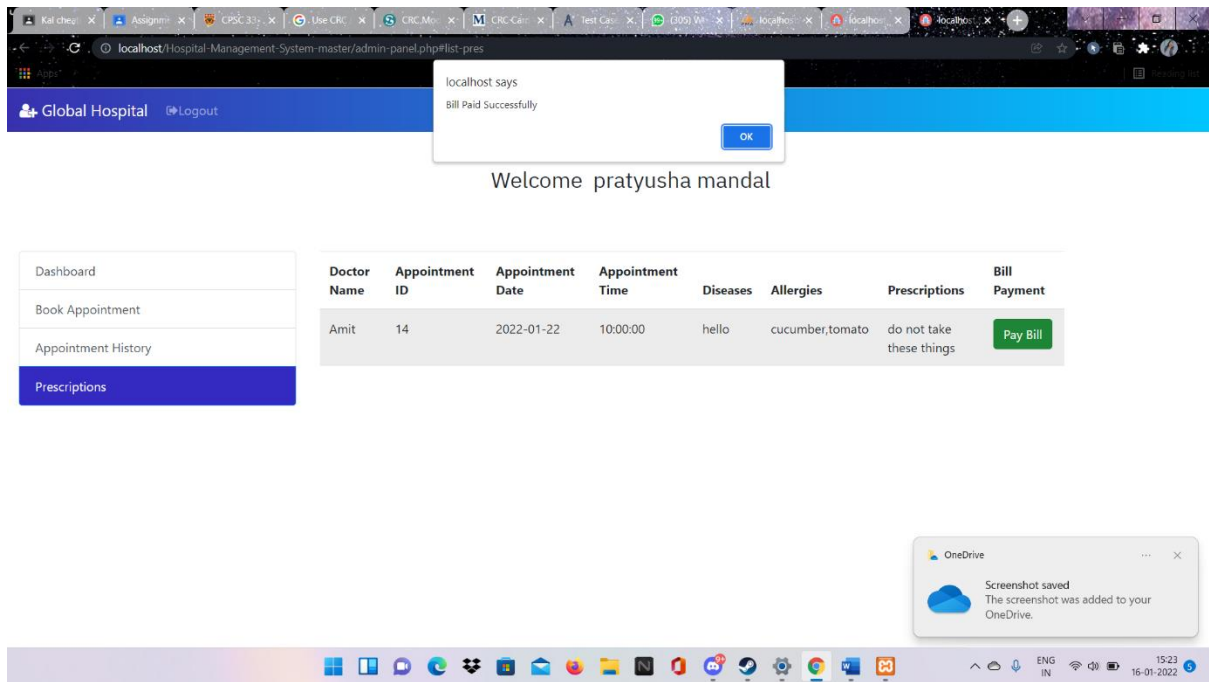
Global Hospital Logout

Welcome pratyusha mandal

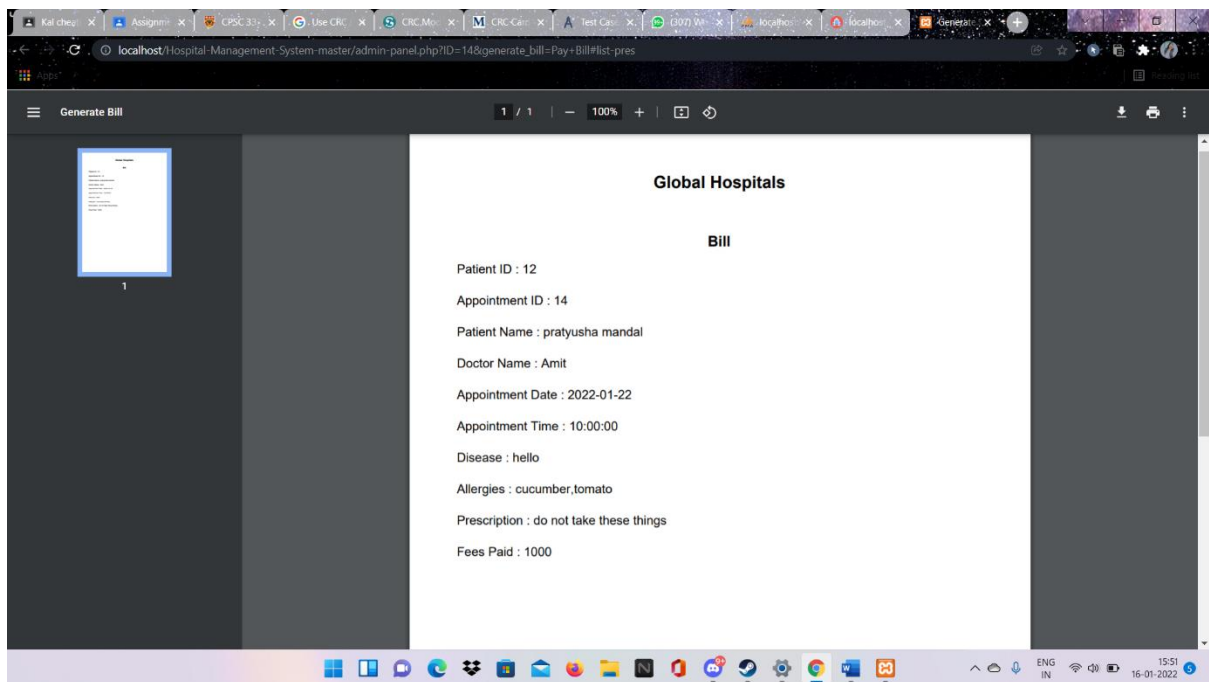
Dashboard
Book Appointment
Appointment History
Prescriptions

Doctor Name	Appointment ID	Appointment Date	Appointment Time	Diseases	Allergies	Prescriptions	Bill Payment
Amit	14	2022-01-22	10:00:00	hello	cucumber,tomato	do not take these things	Pay Bill

Once the doctor prescribes the medicines and puts instructions the patient needs to pay the bill



It shows its paid and a copy of the bill is generated automatically



NOTE:-

This module contains multiple modules as it is a module which is completely based on user preference and user requirement so there can be N number of permutations that a user can perform based on his interest and preferences

IMPLEMENTATION DETAILS

In this Section we will do Analysis of Technologies to use for implementing the project.

FRONT END



Hypertext Markup Language (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 web page 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 `` 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.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Contact Us</title>
5      <link rel="shortcut icon" type="image/x-icon" href="images/favicon.png" />
6      <link href="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">
7      <link rel="stylesheet" href="vendor/fontawesome/css/font-awesome.min.css">
8      <link href="https://fonts.googleapis.com/css?family=IBM+Plex+Sans&display=swap" rel="stylesheet">
9      <link rel="stylesheet" type="text/css" href="contact.css">
10     <script src="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/js/bootstrap.min.js"></script>
11     <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
12     <script >
13         function alphaOnly(event) {
14             var key = event.keyCode;
15             return ((key >= 65 && key <= 90) || key == 8 || key == 32);
16         };
17     </script>
18
19     <style >
20         body{
21             background: -webkit-linear-gradient(left, #3931af, #00c6ff);
22         }
23         .contact-form {
24             background: #f8f9fa;

```

```

25             border-top-left-radius: 10% 50%;
26             border-bottom-left-radius: 10% 50%;
27             border-top-right-radius: 10% 50%;
28             border-bottom-right-radius: 10% 50%;
29         }
30     </style>
31 </head>
32 <body>
33     <nav class="navbar navbar-expand-lg navbar-dark fixed-top" id="mainNav" >
34         <div class="container">
35
36             <a class="navbar-brand js-scroll-trigger" href="index.php" style="margin-top: 10px;margin-left:-65px;font-family: 'IBM Plex Sans',
37             <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarResponsive" aria-controls="navbarResponsive
38                 <span class="navbar-toggler-icon"></span>
39             </button>
40             <div class="collapse navbar-collapse" id="navbarResponsive">
41                 <ul class="navbar-nav ml-auto">
42                     <li class="nav-item" style="margin-right: 40px;">
43                         <a class="nav-link js-scroll-trigger" href="index.php" style="color: white;font-family: 'IBM Plex Sans', sans-serif;"><h6>HOME
44                     </li>
45
46                     <li class="nav-item" style="margin-right: 40px;">
47                         <a class="nav-link js-scroll-trigger" href="services.html" style="color: white;font-family: 'IBM Plex Sans', sans-serif;"><h6>SERVICES
48                     </li>

```

```

41                 <ul class="navbar-nav ml-auto">
42                     <li class="nav-item" style="margin-right: 40px;">
43                         <a class="nav-link js-scroll-trigger" href="index.php" style="color: white;font-family: 'IBM Plex Sans', sans-serif;"><h6>HOME
44                     </li>
45
46                     <li class="nav-item" style="margin-right: 40px;">
47                         <a class="nav-link js-scroll-trigger" href="services.html" style="color: white;font-family: 'IBM Plex Sans', sans-serif;"><h6>SERVICES
48                     </li>
49
50                     <li class="nav-item">
51                         <a class="nav-link js-scroll-trigger" href="contact.html" style="color: white;font-family: 'IBM Plex Sans', sans-serif;"><h6>CONTACT
52                     </li>
53                 </ul>
54             </div>
55         </div>
56     </nav>
57

```



```

65     <div class="container contact-form" style="font-family: 'IBM Plex Sans', sans-serif;">
66         <div class="contact-image">
67             
68         </div>
69         <form method="post" action="contact.php">
70             <h3>Drop Us a Message</h3>
71             <div class="row">
72                 <div class="col-md-6">
73                     <div class="form-group">
74                         <input type="text" name="txtName" class="form-control" placeholder="Your Name *" value="" onkeydown="return a
75                     </div>
76                     <div class="form-group">
77                         <input type="email" name="txtEmail" class="form-control" placeholder="Your Email *" value="" required />
78                     </div>
79                     <div class="form-group">
80                         <input type="tel" name="txtPhone" class="form-control" placeholder="Your Phone Number *" value="" minlength="
81                     </div>
82                     <div class="form-group" >
83                         <input type="submit" name="btnSubmit" class="btnContact" value="Send Message" />
84                     </div>
85                 </div>
86                 <div class="col-md-6">
87                     <div class="form-group" >

```

```

88                         <textarea name="txtMsg" class="form-control" placeholder="Your Message *" style="width: 100%; height: 150px;"
89                     </div>
90                 </div>
91             </div>
92         </form>
93     </div>
94
95 </body>
96 </html>

```

Css



Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like 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, and reduce complexity and repetition in the structural content.

CSS information can be provided from various sources. These sources can be the web browser, the user and the author. The information from the author can be further classified into inline, media type, importance, selector specificity, rule order, inheritance and property definition. CSS style information can be in a separate document or it can be embedded into an HTML document. Multiple style sheets can be imported. Different styles can be applied depending on the output device being used; for example, the screen version can be quite different from the printed version, so that authors can tailor the presentation appropriately for each medium. The style sheet with the highest priority controls the content display. Declarations not set in the highest priority source are passed on to a source of lower priority, such as the user agent style. The process is called cascading.

One of the goals of CSS is to allow users greater control over presentation. Someone who finds red italic headings difficult to read may apply a different style sheet. Depending on the browser and the web site, a user may choose from various style sheets provided by the designers, or may remove all added styles and view the site using the browser's default styling, or may override just the red italic heading style without altering other attributes.

```
1  .symbol {
2      display: inline-block;
3      border-radius: 50%;
4      border: 5px double white;
5      width: 30px;
6      height: 30px;
7  }
8
9  .symbol-empty {
10     background-color: #ccc;
11 }
12
13 .symbol-filled {
14     background-color: black;
15 }
```

```

1      /*
2      #1 color = #9b59b6;
3      #2 color = #8e4fa7;
4      */
5
6      /*
7      #1 color = #9b59b6;
8      #2 color = #8e4fa7;
9      */
10
11     a:hover,
12     a:focus {
13     |   color: #9b59b6;
14     }
15
16     .circled {
17     |   background: #9b59b6;
18     }
19
20     .con1 > div, .con2 > div, .con3 > div {
21     |   background-color: #9b59b6;
22     }
23
24     .scrollup:hover {

```

```

24     .scrollup:hover {
25     |   background: #9b59b6;
26     }
27
28
29     /* service */
30
31     .service-icon span {
32     |   color: #9b59b6;
33     }
34
35
36
37     /* button */
38     .btn-skin {
39     |   background-color: #9b59b6;
40     |   border-color: #9b59b6;
41     }
42
43
44     /* top */
45     .top-area {
46     |   background: #8e4fa7;
47     }

```

```

47     }
48
49     /* menu */
50
51     .navbar-custom ul.nav ul.dropdown-menu li a:hover {
52     |   color: #8e4fa7;
53     }
54
55     .navbar-custom.top-nav-collapse .nav li a:hover,
56     .navbar-custom .nav li a:hover,
57     .navbar-custom .nav li.active a, .navbar-custom.top-nav-collapse .nav li.active a {
58     |   color: #8e4fa7;
59     }
60
61
62     .owl-theme .owl-controls .owl-page.active span,
63     .owl-theme .owl-controls.clickable .owl-page:hover span {
64     |   background-color: #9b59b6;
65     }
66
67     .bg-skin {
68     |   background: #9b59b6;
69     }
70

```

```

70
71 .highlight-color {
72   background-color: #9b59b6;
73 }
74 .color {
75   color: #9b59b6;
76 }
77
78 /* pricing box */
79 .pricing-content.featured h2,.pricing-content.featured .price-bottom {
80   background-color: #8e4fa7;
81 }
82 .pricing-content.featured h3 {
83   background: #9b59b6;
84 }
85
86
87
88 /* team */
89 .cbp-l-grid-team-name {
90   color: #8e4fa7;
91 }
92
93 .cbp-l-filters-button .cbp-filter-counter {

```

```

94   background-color: #9b59b6;
95 }
96
97 .cbp-l-filters-alignLeft .cbp-filter-item-active {
98   /* @editable properties */
99   background-color: #9b59b6;
100   border: 1px solid #9b59b6;
101 }
102
103 .cbp-l-filters-dropdownWrap {
104   /* @editable properties */
105   width: 200px;
106   position: absolute;
107   right: 0;
108   background: #9b59b6;
109 }
110 .cbp-l-filters-button .cbp-filter-item.cbp-filter-item-active {
111   /* @editable properties */
112   background-color: #9b59b6;
113   border-color: #9b59b6;
114 }
115 .cbp-l-caption-buttonLeft,
116 .cbp-l-caption-buttonRight {
117   /* @editable properties */

```

```

116 .cbp-l-caption-buttonRight {
117   /* @editable properties */
118   background-color: #9b59b6;
119 }
120
121 .cbp-l-caption-buttonLeft:hover,
122 .cbp-l-caption-buttonRight:hover {
123   /* @editable properties */
124   background-color: #8e4fa7;
125 }
126
127 .form-control:focus {
128   border-color: #8e4fa7;
129 }

```

JavaScript



JavaScript is a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities. It relies upon the host environment in which it is embedded to provide these features.

Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

The terms Vanilla JavaScript and Vanilla JS refer to JavaScript not extended by any frameworks or additional libraries. Scripts written in Vanilla JS are plain JavaScript code. Google's Chrome extensions, Opera's extensions, Apple's Safari 5 extensions, Apple's Dashboard Widgets, Microsoft's Gadgets, Yahoo! Widgets, Google Desktop Gadgets, and SerenceKlipfolio are implemented using JavaScript.

```
1 (function($) {
2     var selectors = [];
3
4     var check_binded = false;
5     var check_lock = false;
6     var defaults = {
7         interval: 250,
8         force_process: false
9     }
10    var $window = $(window);
11
12    var $prior_appeared;
13
14    function process() {
15        check_lock = false;
16        for (var index = 0; index < selectors.length; index++) {
17            var $appeared = $(selectors[index]).filter(function() {
18                return $(this).is(':appeared');
19            });
20
21            $appeared.trigger('appear', [$appeared]);
22
23            if ($prior_appeared) {
24                var $disappeared = $prior_appeared.not($appeared);
```

```

25     $disappeared.trigger('disappear', [$disappeared]);
26   }
27   $prior_appeared = $appeared;
28 }
29 }
30
31 // "appeared" custom filter
32 $.expr[':']['appeared'] = function(element) {
33   var $element = $(element);
34   if (!$element.is(':visible')) {
35     return false;
36   }
37
38   var window_left = $window.scrollLeft();
39   var window_top = $window.scrollTop();
40   var offset = $element.offset();
41   var left = offset.left;
42   var top = offset.top;
43
44   if (top + $element.height() >= window_top &&
45       top - ($element.data('appear-top-offset') || 0) <= window_top + $window.height() &&
46       left + $element.width() >= window_left &&
47       left - ($element.data('appear-left-offset') || 0) <= window_left + $window.width()) {
48     return true;

```

```

48     return true;
49   } else {
50     return false;
51   }
52 }
53
54 $.fn.extend({
55   // watching for element's appearance in browser viewport
56   appear: function(options) {
57     var opts = $.extend({}, defaults, options || {});
58     var selector = this.selector || this;
59     if (!check_binded) {
60       var on_check = function() {
61         if (check_lock) {
62           return;
63         }
64         check_lock = true;
65
66         setTimeout(process, opts.interval);
67       };
68
69       $(window).scroll(on_check).resize(on_check);
70       check_binded = true;
71     }

```

```

71   }
72
73   if (opts.force_process) {
74     setTimeout(process, opts.interval);
75   }
76   selectors.push(selector);
77   return $(selector);
78 }
79 });
80
81 $.extend({
82   // force elements's appearance check
83   force_appear: function() {
84     if (check_binded) {
85       process();
86       return true;
87     }
88     return false;
89   }
90 });
91 })(jQuery);

```

BACK END



PHP is a server side scripting language that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages. PHP scripts can only be interpreted on a server that has PHP installed. The client computers accessing the PHP scripts require a web browser only. A PHP file contains PHP tags and ends with the extension ".php".

The term PHP is an acronym for PHP: Hypertext Preprocessor. PHP is a server-side scripting language designed specifically for web development. PHP can be easily embedded in HTML files and HTML codes can also be written in a PHP file. The thing that differentiates PHP with client-side language like HTML is, PHP codes are executed on the server whereas HTML codes are directly rendered on the browser.

PHP: Hypertext Preprocessor (or simply PHP) is a general-purpose programming language originally designed for web development. It was originally created by RasmusLerdorf in 1994. PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control.

```
1  <?php
2  $type = 'cidfont0';
3  $name = 'STSongStd-Light-Acro';
4  $displayname = 'STSong Light (Simp. Chinese)';
5  $desc = array(
6      'Ascent' => 752,
7      'Descent' => -271,
8      'CapHeight' => 737,
9      'Flags' => 6,
10     'FontBBox' => '[-25 -254 1000 880]',
11     'ItalicAngle' => 0,
12     'StemV' => 58,
13     'Style' => '<< /Panose <000000000400000000000000> >>',
14 );
15 $cidinfo = array(
16     'Registry' => 'Adobe',
17     'Ordering' => 'GB1',
18     'Supplement' => '2',
19 );
20 $enc = 'UniGB-UCS2-H';
21
22 // underline position, needs checking:
23 $up = -130;
24 $ut = 40;
```

```

22 // underline position, needs checking;
23 $sup = -130;
24 $ut = 40;
25
26 $dw = 1000;
27 $cw = array(
28 32 => 207, 33 => 270, 34 => 342, 35 => 467, 36 => 462, 37 => 797, 38 => 710, 39 => 239, 40 => 374, 41 => 374,
29 42 => 423, 43 => 605, 44 => 238, 45 => 375, 46 => 238, 47 => 334, 48 => 462, 49 => 462, 50 => 462, 51 => 462,
30 52 => 462, 53 => 462, 54 => 462, 55 => 462, 56 => 462, 57 => 462, 58 => 238, 59 => 238, 60 => 605, 61 => 605,
31 62 => 605, 63 => 344, 64 => 748, 65 => 684, 66 => 560, 67 => 695, 68 => 739, 69 => 563, 70 => 511, 71 => 729,
32 72 => 793, 73 => 318, 74 => 312, 75 => 666, 76 => 526, 77 => 896, 78 => 758, 79 => 772, 80 => 544, 81 => 772,
33 82 => 628, 83 => 465, 84 => 607, 85 => 753, 86 => 711, 87 => 972, 88 => 647, 89 => 620, 90 => 607, 91 => 374,
34 92 => 333, 93 => 374, 94 => 606, 95 => 500, 96 => 239, 97 => 417, 98 => 503, 99 => 427, 100 => 529, 101 => 415,
35 102 => 264, 103 => 444, 104 => 518, 105 => 241, 106 => 230, 107 => 495, 108 => 228, 109 => 793, 110 => 527, 111 => 524,
36 112 => 524, 113 => 504, 114 => 338, 115 => 336, 116 => 277, 117 => 517, 118 => 450, 119 => 652, 120 => 466, 121 => 452,
37 122 => 407, 123 => 370, 124 => 258, 125 => 370, 126 => 605
38 );
39 // --- EOF ---|

```

```

1 <?php
2 // TCPDF FONT FILE DESCRIPTION
3 $type='core';
4 $name='Symbol';
5 $sup=-100;
6 $ut=50;
7 $dw=587;
8 $diff='';
9 $enc='';
10 $desc=array('Flags'=>4,'FontBBox'=>'[-180 -293 1090 1010]', 'ItalicAngle'=>0, 'Ascent'=>1010, 'Descent'=>-293, 'Leading'=>0, 'CapHeight'=>1010
11 $cw=array(0=>587,1=>587,2=>587,3=>587,4=>587,5=>587,6=>587,7=>587,8=>587,9=>587,10=>587,11=>587,12=>587,13=>587,14=>587,15=>587,16=>587,1
12 // --- EOF ---

```

MySQL



MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). It is one part of the very popular LAMP platform consisting of Linux, Apache, My SQL, and PHP. Currently My SQL is owned by Oracle. My SQL database is available on most important OS platforms. It runs on BSD Unix, Linux, Windows, or Mac OS. Wikipedia and YouTube use My SQL. These sites manage millions of queries each day. My SQL comes in two versions: My SQL server system and My SQL embedded system.

RDBMS TERMINOLOGY

Before we proceed to explain MySQL database system, let's revise few definitions related to database.

- **Database:** A database is a collection of tables, with related data.
- **Table:** A table is a matrix with data. A table in a database looks like a simple spreadsheet.
- **Column:** One column (data element) contains data of one and the same kind, for example, the column postcode.
- **Row:** A row (= tuple, entry or record) is a group of related data, for example, the data of one subscription.
- **Redundancy:** Storing data twice, redundantly to make the system faster.
- **Primary Key:** A primary key is unique. A key value cannot occur twice in one table. With a key, you can find at most one row.
- **Foreign Key:** A foreign key is a linking pin between two tables.
- **Compound Key:** A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
- **Index:** An index in a database resembles an index at the back of a book.
- **Referential Integrity:** Referential Integrity makes sure that a foreign key value always points to an existing row.

CONCLUSION AND FUTURE WORK

The conclusion of this project is A Hospital management system is a computerized management system. This system keeps the records of hardware assets besides software of this organization. The proposed system will keep a track of Doctors, Patients, Accounts and generation of report regarding the present status. This project has GUI based software that will help in storing, updating and retrieving the information through various user-friendly menu-driven modules. The project “Hospital Management System” is aimed to develop to maintain the day-to-day state of admission of Patients, List of Doctors, payment details etc. Main objective of this project is to provide solution for Hospital to manage most there work using computerized process. This software application will help admin to handle Patients information, slot allocation details, payment details, billing information, etc. Detailed explanation about modules and design are provided in project documentation. The existing system is a manually maintained system. All the Hospital records are to be maintained for the details of each patients, Fee details, slot Allocation, etc. All these details are entered and retrieved manually, because of this there are many disadvantages like Time Consuming ,updating process, inaccuracy of data. For avoiding this we introduced or proposed a new system in proposed system the computerized version of the existing system which provides easy and quick access over the data.

REFERENCE

- 1) The Pragmatic Programmer-Andrew Hunt and David Thomas
- 2) Code Complete written by Steve McConnell
- 3) Patterns of Enterprise Application Architecture-Martin Fowler
- 4) Design Patterns: Elements of Reusable Object-Oriented Software was written by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, with a foreword by Grady Booch.
- 5) www.github.com
- 6) www.projectz.com
- 7) www.uml diagrams.com
- 8) Refactoring: Improving the Design of Existing Code- Martin Fowler