

CHAPTER 1

INTRODUCTION

1.1 Overview of DBMS

A Database management system, or DBMS is software designed to assist in managing and utilizing large collection in data, and the need of such system, as well as their use, is growing rapidly. The Alternative to using a DBMS is used to hoc approaches that do not carry over from one application to another.

The area of the Database Management system is microcosm of computer science in general. The issues addressed and the technique used to span a wide spectrum, including languages, object orientation and other programming paradigm, compilation, operating system, concurrent programming, data structures, algorithms, theory, parallel and distributed systems user interface, expert systems and artificial intelligence, statistical techniques, and dynamic programming.

1.2 History

From the earliest days of computers, storing and manipulating data have been a major application focus. The first general purpose DBMS was designed by Charles Bachman at General Electric in the early 1960s was called The Integrated Data Store. It formed the basis for the network data model, which was standardized by the Conferences on Data Systems Languages (CODASYL) and strongly influenced database systems through the 1960s. Bachman was the first recipient of ACM's Turing Award (the computer science equivalent of a Nobel prize) for work in the database Area; he receives the award in 1973.

In the late 1960's IBM developed the Information Management System (IMS)

DBMS, used even today in many major installations. IMS form, the basis for an alternative data representation framework called the hierarchical data model. The SABRE system for making airlines reservation was jointly developed by American Airlines and IBM around the same time, and it allowed several people to access the same data through.

An interesting phenomenon is the emergence of several enterprise resource planning (ERP) and management resources planning (MRP) packages, which add a substantial layer of application-oriented features on top of a DBMS. widely used packages, include systems from Bann, Oracle, PeopleSoft, SAP and Siebel.

1.3 Applications of DBMS

Nowadays DBMS are used in almost all the areas ranges from science, engineering, medicine, business, industry, government, art, entertainment, education and training.

- **DBMS in the field of Library Management System**

There are thousands of books in the library so it is very difficult to keep records of all the books in a copy or register. DBMS is used to maintain all the information related to book issue date, name of book, author and availability of book.

- **DBMS in the field of Banking**

Another major application is the banks. Thousands of transactions through daily can do this without going bank. To manage such huge transactions is just because of DBMS that manages bank transactions.

- **DBMS in the field of Universities and colleges**

Examinations are done online today and universities and colleges maintain all these records through DBMS. Student's registrations details, results, courses and grades all the information are stored in database.

- **DBMS in the field of Telecommunications**

Any telecommunication company cannot even think about their business without DBMS. DBMS is required for these companies to store the call details and monthly postpaid bills.

- **DBMS in the field of Online Shopping**

Online shopping has become a big trend of these days. No one wants to go to shops and to waste his time. Everyone wants to shop from home. So all these products are added and sold only with the help of DBMS. Purchase information, invoice and payment all these are done with the help of DB.

- **DBMS in the field of Military**

Military keeps records of millions of soldiers and it has millions of files that should be kept secured and safe. As DBMS provides a big security assurance to military information so it is widely used in militaries. One can easily search for all the information about anyone within seconds with the help of DBMS.

1.4 Problem Statement

This project is website to manage all the information in the hospital. Hospital is built with rooms and beds to cure the patients with illness. Hospital Management System is a website application to maintain all the appointments and details and the patient information. This project helps the admin to book an appointment, show the patient details, doctor's details etc. It is a big part of hospital management as it helps the admin to collect data and store information in a proper way. So, our main objective is to make a website that will help the storing of data and appointment more easily.

Objectives of the Project:

The main objective of this application is to make it simpler for the management. The need for maintaining hundreds and thousands of records of all appointments and patient details. Also

searching should be very faster so they can find required details instantly. The objective of this project is to develop a computerized project through which we can reduce human efforts.

1.5 Organization of the report

This section deals with the Introduction and organization of the project report. Chapter 2 discusses the Specific to the problem-Requirement Analysis-Design. Chapter 3 discusses the Design and Implementation Chapter 4 gives information about the snapshot and results Chapter 5 include conclusion and future scope. Chapter 6 gives the references of the project.

Summary

In this chapter we have learnt the overview of what is DBMS and it's history and application on various fields, then we learnt about the problem statement taken for the project and it's objectives, also organization of this report.

Chapter 2

REQUIREMENT ANALYSIS AND DESIGN

In this chapter you have to provide the requirements for the development of the project, from the requirements you should give a high-level system design, software requirements, etc.

2.1 Project Definition

Hospital Management System is mainly designed to maintain details of the patient and patient medical record and also the information of doctors and staff working there.

2.2 Advantages

- This system decreases the chance of error.
- This system requires less time for completion of any work.
- It is used to maintain the information such as medicines and patient information.
- Work load and man power is very fast.

2.3 Requirement Analysis

SOFTWARE

Windows 10

Windows10 is a personal computer operating system developed and released by Microsoft as part of the Windows NT family of operating systems. It was released on July 29, 2015. It is the first version of windows that receives ongoing feature updates. Devices in enterprise environments can receive these updates at a slower pace, or use long-term support milestones that only receive critical updates, such as security patches etc.

BACK END

SQL Workbench

MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. It is developed by Oracle Corporations and Sun Microsystems.

PHP

PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites. It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server. PHP also helps to connect to these databases and do various operations on databases.

FRONT END

HTML

Hypertext Mark-up Language is the standard mark-up language for documents designed to be displayed in a web browser [4].

CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a mark-up language such as HTML.

BOOTSTRAP

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development [5].

Software requirements

- Operating system - Windows 10

- Backend - SQL Workbench and PHP
- Front end - Html/CSS/bootstrap
- Platform - Wamp Server

Hardware components

- Processor - Intel core i5
- Processor speed - 2.4 GHz
- Ram - 8 GB
- Hard disk - 1 TB

Summary

Here we discussed about the requirement analysis and its design on project definition and its advantages and which software is used, what languages used for frontend and the database software used for the backend, also we have mentioned the software and hardware components of the system used for this project.

Chapter 3

DESIGN AND IMPLEMENTATION

In this chapter there you should clearly specify the entities, the attributes selected and from that how the design has been achieved to provide the ER diagram and how the schema diagram is evolved.

SCHMA DESCRIPTION

PATIENT: It contains the attributes pat_id, fname, lname, age, weight, gender, phoneno, disease, address, medicine.

DOCTOR: It contains the attributes doc_id, doc_name, phoneno, qualification, area_of_exp, availability.

APPOINTMENT: It contains the attributes app_id, pat_id, doc_id, admDate.

MEDICAL RECORD: It contains the attributes record_id, pat_id, doc_id, admDate, disDate, roomNo, bedNo, infoillness, medicine.

STAFF: It contains the attributes nur_id, nur_name, nur_phone, nur_quali.

3.1 ER DIAGRAM

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

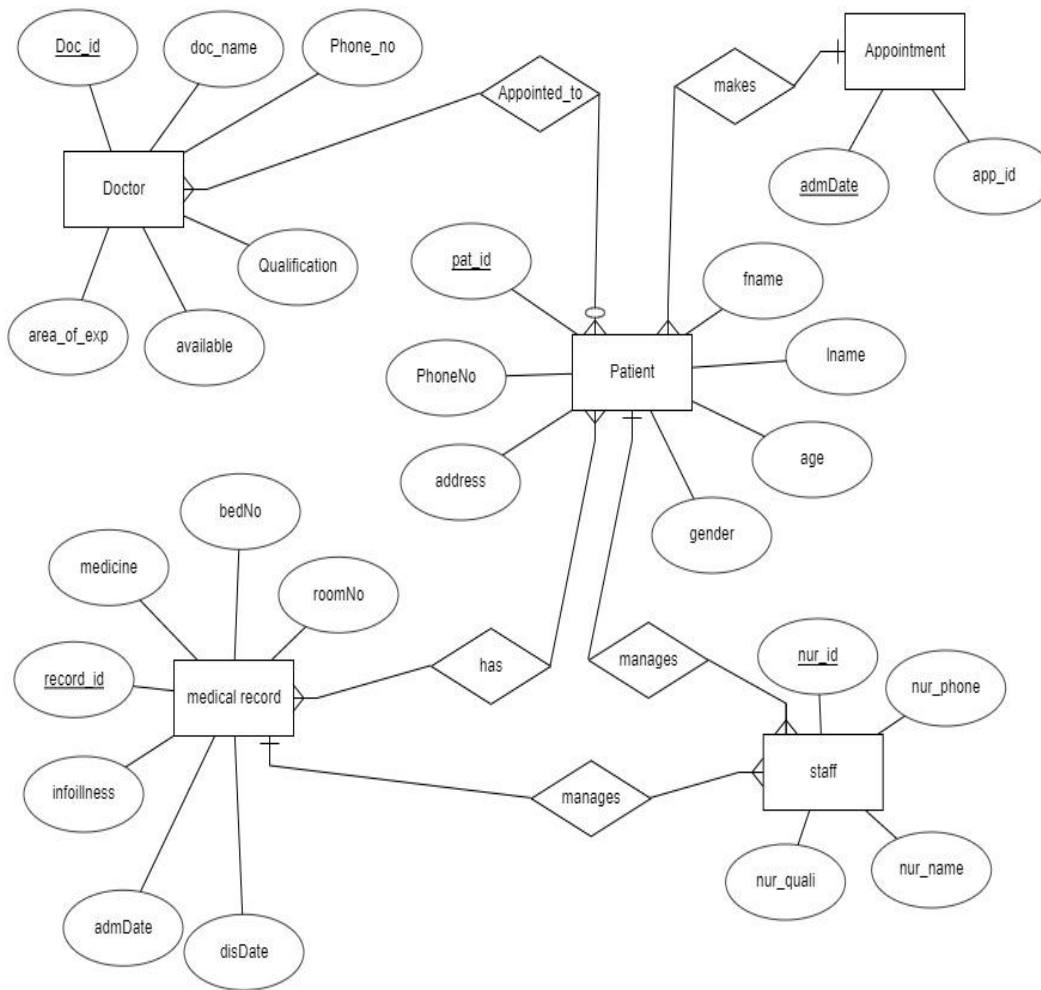


Fig 3.1: Entity-Relationship diagram of HMS

An entity-relationship model describes interrelated things of interest in a specific domain of knowledge. In the ER-model depicted in Fig 3.1 Hospital Management System entities and attributes and their inter-relationships and cardinality ratio are shown. Each entity contains primary and unique keys.

3.2 SCHEMA DIAGRAM

A database **schema** is the skeleton structure that represents the logical view of the entire database.

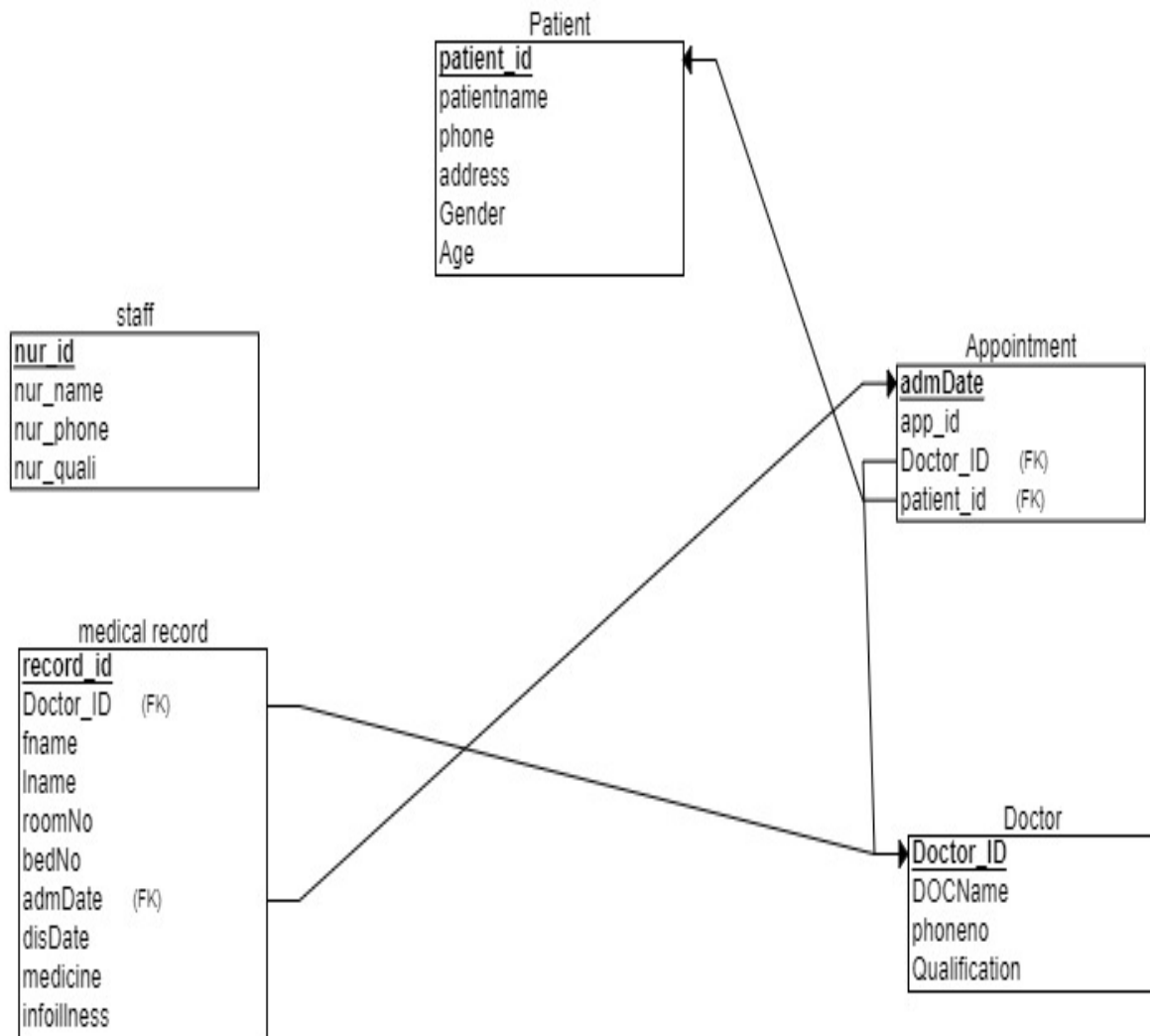


Fig 3.2. Schema Diagram of the HMS

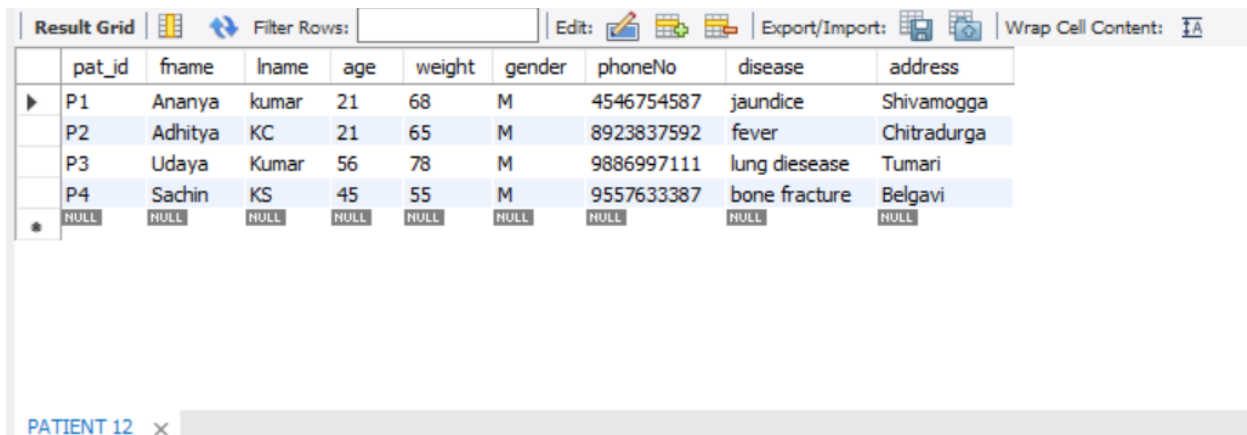
A database schema is the skeleton structure that represents the logical view of the entire database. As shown in Fig 3.1 we have 5 tables, and their primary keys are highlighted.

3.3 IMPLEMENTATION

INSERT

To Insert data the following queries are used.

"INSERT INTO PATIENT (pat_id, fname, lname, age, weight, gender, phoneNo, disease, address) VALUES ('P1', 'Ananya', 'kumar', 21, 68, 'M', 4546754587, 'jaundice', 'Shivamogga')";



pat_id	fname	lname	age	weight	gender	phoneNo	disease	address
P1	Ananya	kumar	21	68	M	4546754587	jaundice	Shivamogga
P2	Adhitya	KC	21	65	M	8923837592	fever	Chitradurga
P3	Udaya	Kumar	56	78	M	9886997111	lung disease	Tumari
P4	Sachin	KS	45	55	M	9557633387	bone fracture	Belgavi
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Fig 3.3. PATIENT TABLE

In Fig 3.3 it is explained that we have 4 patients in a hospital, in which we store their information, who have given the appointment.

INSERT INTO doctor (doc_id, doc_name, phoneno, qualification, area_of_exp, available) VALUES ('D1','Madhava M', '93849837849', 'MBBS','Physicist', '8pm to 12pm')";

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	doc_id	doc_name	phoneno	qualification	area_of_exp	available
▶	D1	Madhava M	93849837849	MBBS	Physicist	8am to 12pm
	D2	Deepak D	98843375643	BAMS	Surgeon	2pm to 4pm
	D3	Shivananda	98374329434	MBBS	Physicist	2pm to 4pm
	D4	Shashwathi	98784832244	BAMS	general practitioner	6pm to 8pm
✱	NULL	NULL	NULL	NULL	NULL	NULL

DOCTOR 14 ✕

Fig 3.4. DOCTOR TABLE

In Fig 3.4 it is explained that we have 4 doctors are in a hospital, in which we store their information, who would treat the patients in the hospital.

"INSERT INTO NURSE VALUES('N1','Sheetal','8634783748','Bsc');"

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	nur_id	nur_name	nur_phone	nur_quali
▶	N1	Sheetal	8634783748	Bsc
	N2	Soujanya	98774367467	Bs
	N3	Suraksha	98774367467	Bs
✱	NULL	NULL	NULL	NULL

NURSE 16

×

Fig 3.5. NURSE TABLE

In Fig 3.5 it is explained that we have 3 nurses are in a hospital, in which we store their information, who would treat and look after the patients in the hospital.

INSERT INTO APPOINTMENT VALUES ('A1', P1,'D1','2021-01-01')





Result Grid	Filter Rows:	Edit:	Export/Import:
app_id	pat_id	doc_id	admDate
A1	P1	D1	2021-01-01
A2	P2	D3	2020-12-02
A3	P3	D2	2020-11-30
A4	P4	D4	2020-11-14
NULL	NULL	NULL	NULL

appointment 19 x

Fig 3.6. Appointment Table

In Fig 3.6 it is explained that we have 4 patients are in a hospital, in which we store their appointment details with the doctor in which date, so that the doctors can see their schedule and manage the patients accordingly.

"INSERT INTO MEDICALRECORD VALUES ('R1','Ananya','kumar','D1','2021-01-01','2021-01-09',1,1,'Fever, Chills, Abdominal pain ','Hepasafe Capsule')";

Result Grid		 Filter Rows:		 Edit:		 Export/Import:		 Wrap Cell Content:		
	record_id	fname	lname	doc_id	admDate	disDate	roomNo	bedNo	infoillness	medicine
▶	R1	Ananya	kumar	D1	2021-01-01	2021-01-09	1	1	Fever, Chills, Abdo...	Hepasafe Capsule
	R2	Adhitya	KC	D1	2021-01-08	2021-01-09	2	2	fagaef	fafafa
	R3	Udaya	kumar	D2	2020-11-30	2021-01-26	2	1	Trouble breathing. ...	Bronchodilators Inhale...
	R4	Sachin	KS	D4	2020-11-14	2021-01-01	2	3	Stable fracture Ope...	Abaloparotide Aspirin ...
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

medicalrecord 31 x

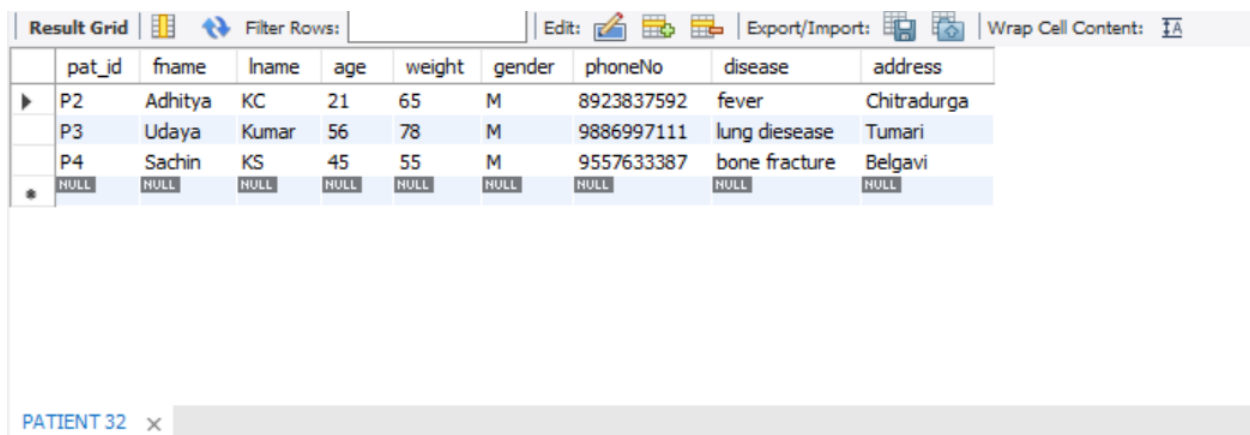
Fig 3.7. Medical Record Table

In Fig 3.7 it is explained that we have 4 patients are in a hospital, in which we store their medical record details with the doctor in which date, discharge date room number, bed number illness and medication information of a particular patient are stored here.

DELETE:

To delete data from the table, following query is used:

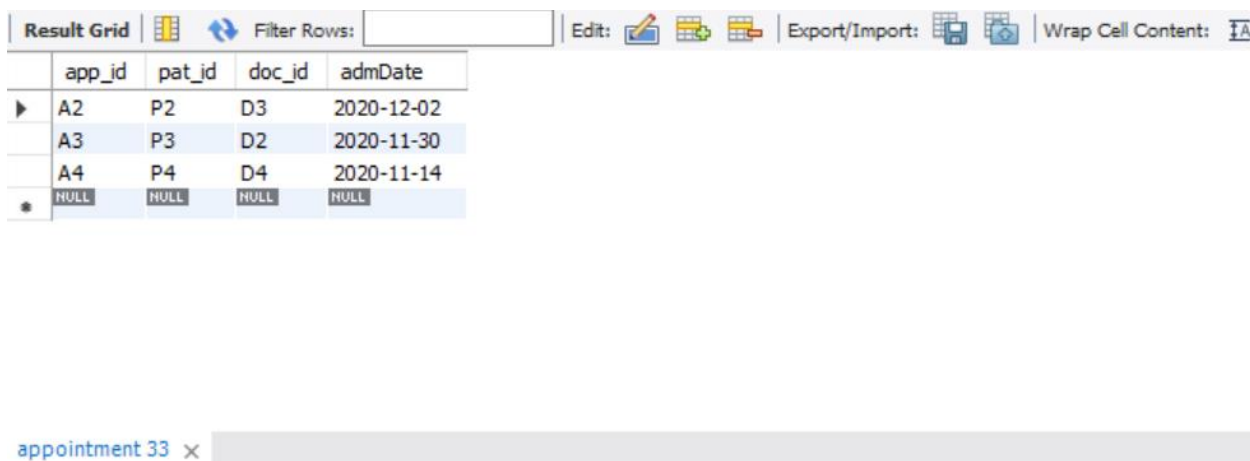
"DELETE FROM PATIENT WHERE pat_id='P1'";



	pat_id	fname	lname	age	weight	gender	phoneNo	disease	address
▶	P2	Adhitya	KC	21	65	M	8923837592	fever	Chitradurga
	P3	Udaya	Kumar	56	78	M	9886997111	lung disease	Tumari
	P4	Sachin	KS	45	55	M	9557633387	bone fracture	Belgavi
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

PATIENT 32 x

Fig 3.8. PATIENT TABLE (after deletion)



	app_id	pat_id	doc_id	admDate
▶	A2	P2	D3	2020-12-02
	A3	P3	D2	2020-11-30
	A4	P4	D4	2020-11-14
*	NULL	NULL	NULL	NULL

appointment 33 x

Fig 3.9. APPOINTMENT TABLE (after deletion)

record_id	fname	lname	doc_id	admDate	disDate	roomNo	bedNo	infoillness	medicine
R1	Ananya	kumar	D1	2021-01-01	2021-01-09	1	1	Fever, Chills, Abdo...	Hepasafe Capsule
R2	Adhitya	KC	D1	2021-01-08	2021-01-09	2	2	fagaef	fafafa
R3	Udaya	kumar	D2	2020-11-30	2021-01-26	2	1	Trouble breathing. ...	Bronchodilators Inhale...
R4	Sachin	KS	D4	2020-11-14	2021-01-01	2	3	Stable fracture Ope...	Abaloparotide Aspirin ...
* NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Fig 3.10. MEDICAL RECORD TABLE (after deletion)

As you can see in Fig 3.8,3.9,3.10 in the patient database the details of patient id 'P1' is deleted the moment he is discharged from the hospital, the appointment also gets deleted but the patient record remains in the database for future use in medication if the patient once again admits into the hospital.

STORED PROCEDURE AND UPDATE

```
CREATE STORED PROCEDURE updateRecord (recordID varchar (5), fn varchar (40), ln
varchar (40), admdate date, disdate date, roomno int, bedno int, medi varchar (100), ill varchar
(200))
```

```
BEGIN
```

```
UPDATE medicalrecord SET fname=fn, lname=ln, admDate = admdate, disDate = disdate,
roomNo = roomno, bedNo = bedno, medicine = medi, infoillness = ill WHERE
record_id=recordID;
```

```
END
```

record_id	fname	lname	doc_id	admDate	disDate	roomNo	bedNo	infoillness	medicine
R1	Ananya	kumar	D1	2021-01-01	2021-01-09	1	1	Fever, Chills, Abdo...	Hepasafe Capsule
R2	Adhitya	KC	D1	2021-01-08	2021-01-09	2	2	fagaagef	fafafa
R3	Udaya	kumar	D2	2020-11-30	2021-01-26	2	1	Trouble breathing, ...	Bronchodilators Inhale...
R4	Sachin	KS	D4	2020-11-14	2021-01-01	2	3	Stable fracture Ope...	Abaloparatide Aspirin ...
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Fig 3.11. MEDICAL RECORD (before update)

record_id	fname	lname	doc_id	admDate	disDate	roomNo	bedNo	infoillness	medicine
R1	Ananya	kumar	D1	2021-01-01	2021-01-09	1	1	Fever, Chills, Abdominal pain.	Hepasafe Capsule
R2	Adhitya	KC	D1	2021-01-08	2021-01-09	2	2	fever of 44-degree celsius along with cough an...	Paracetamol Antibiotic...
R3	Udaya	kumar	D2	2020-11-30	2021-01-26	2	1	Trouble breathing, Shortness of breath.	Bronchodilators Inhale...
R4	Sachin	KS	D4	2020-11-14	2021-01-01	2	3	Stable fracture Open fracture	Abaloparatide Aspirin ...
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Fig 3.12. MEDICAL RECORD (after update)

Here in this figure Fig 3.11, 3.12 you can see that from the insertion of medical record, the medical record Id 'R2' was having some gibberish information during insertion after updating using stored procedure the 'R2' id patient information is updated.

DISPLAY

To display data from the table, following query is used:

"SELECT * FROM PATIENT"

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	pat_id	fname	lname	age	weight	gender	phoneNo	disease	address
▶	P1	Ananya	kumar	21	68	M	4546754587	jaundice	Shivamogga
	P2	Adhitya	KC	21	65	M	8923837592	fever	Chitradurga
	P3	Udaya	Kumar	56	78	M	9886997111	lung disease	Tumari
	P4	Sachin	KS	45	55	M	9557633387	bone fracture	Belgavi
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

PATIENT 12

×

Fig 3.13.TO DISPLAY THE PATIENT FROM PATIENT TABLE

In Fig 3.13 we are displaying the list of patients. There is an option to view the details of a patient in the front-end, by clicking the submit button or the patient details bar on the right it displays the name, phone no, address, disease age weight gender of the patient who taken appointment in the hospital.

Summary

In this chapter we have covered the design and implementation of the backend of the project where we have showed the ER diagram and Schema diagram along with its definition and explanation, then we also covered on how to implement it using the SQL queries like insert delete and update using stored procedure on tables created for the project.

Chapter 4

RESULT AND SNAPSHOTS

Figure 4.1 Login page

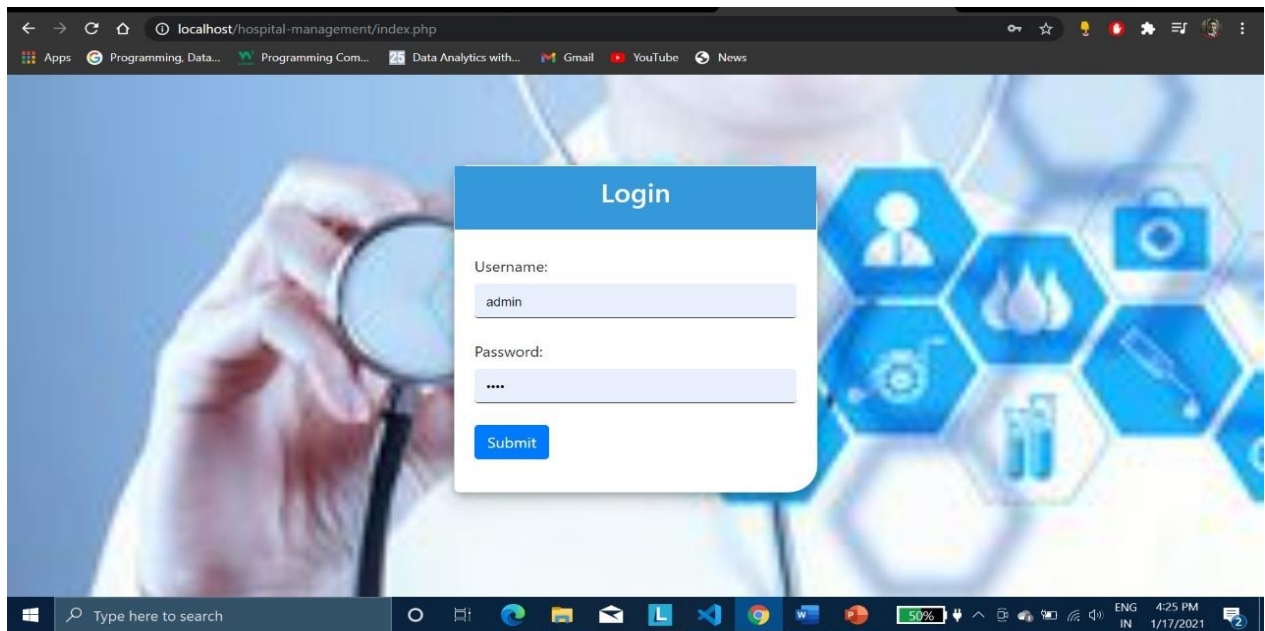


Figure 4.1 Login Page

The fig 4.1 is the login page where the user has to insert username and password and once it matches with the username and password in the database, it takes you to the next page by clicking the submit button.

Figure 4.2 patient registration

Welcome
This is the hospital management system.

Patient

Patient details:
Appointment:
Appointment list:

Staff

Add New Staff:
Staff Details:

Doctor

Add Doctor:
Doctor details:

Records

Add record:
Medical Records:

Patient Registration

Patient Id: _____

First Name: Enter your first Name

Last Name: Enter your last Name

Age: Enter your age

Weight: Enter your weight

Gender: ☒ Male ☐ Female ☐ Other

Phone No: number

Disease: give disease

Address: Enter your Address

Figure 4.2 patient registration

In this fig 4.2 it is the home page of the HMS where the user can register the patient by inserting the information of the patient by clicking the submit button and can do various operations on other pages which is directed by clicking the links on the left side of the web page, which contains the list.

list of patients

localhost/hospital-management/listofpatients.php?by=0

Apps Programming, Data... Programming Com... Data Analytics with... Gmail YouTube News

Patient Doctor Appointment Staff

List of patient :

SL.No	PID	Patient Name	Age	Weight	Gender	Address	Phone No	Disease	Action
1	P1	Ananya kumar	21	68	M	Jammu	4546754587	jaundice	<input type="button" value="Edit"/>
2	P2	Adhitya KC	20	70	M	chitradurga	8923837592	dysentery	<input type="button" value="Edit"/>
3	P3	Anubhav Khajuria	22	70	M	Jammu and Kashmir	8923837592	covid-19	<input type="button" value="Edit"/>

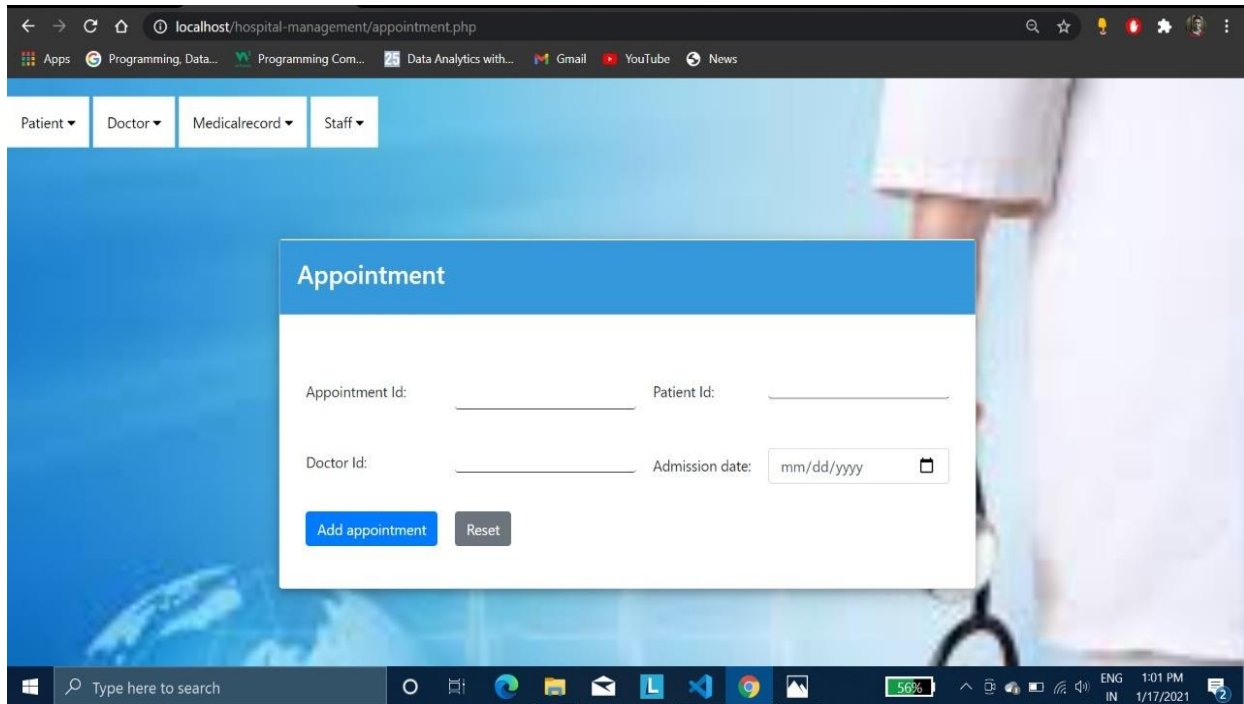
Type here to search

58% 12:56 PM 1/17/2021

Figure 4.3 patient details

The above figure 4.3 gives the details of the patient taken during the appointment where the details are stored in the database where the information is retrieved from it and displayed here.

Figure 4.4 Appointment form

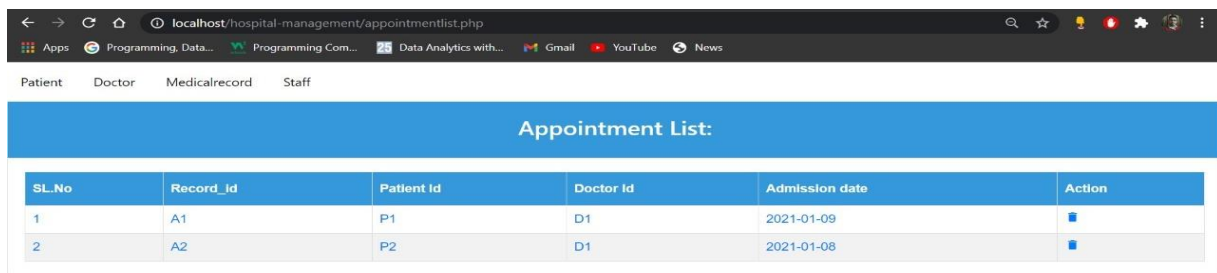


The screenshot shows a web browser window with the URL `localhost/hospital-management/appointment.php`. The browser's address bar and tabs are visible at the top. Below the browser window, there is a navigation bar with four dropdown menus: "Patient", "Doctor", "Medicalrecord", and "Staff". The main content area features a blue header with the word "Appointment". Below the header, there are four input fields: "Appointment Id:", "Patient Id:", "Doctor Id:", and "Admission date:". The "Admission date:" field includes a date picker icon. At the bottom of the form, there are two buttons: "Add appointment" (in blue) and "Reset" (in grey). The background of the form is a blurred image of a person in a white lab coat.

Figure 4.4 Appointment form

In the above figure 4.4 is the appointment form where the user can fill the details of appointment and store in the database for later retrieval to see the list.

Figure 4.5 Appointment list





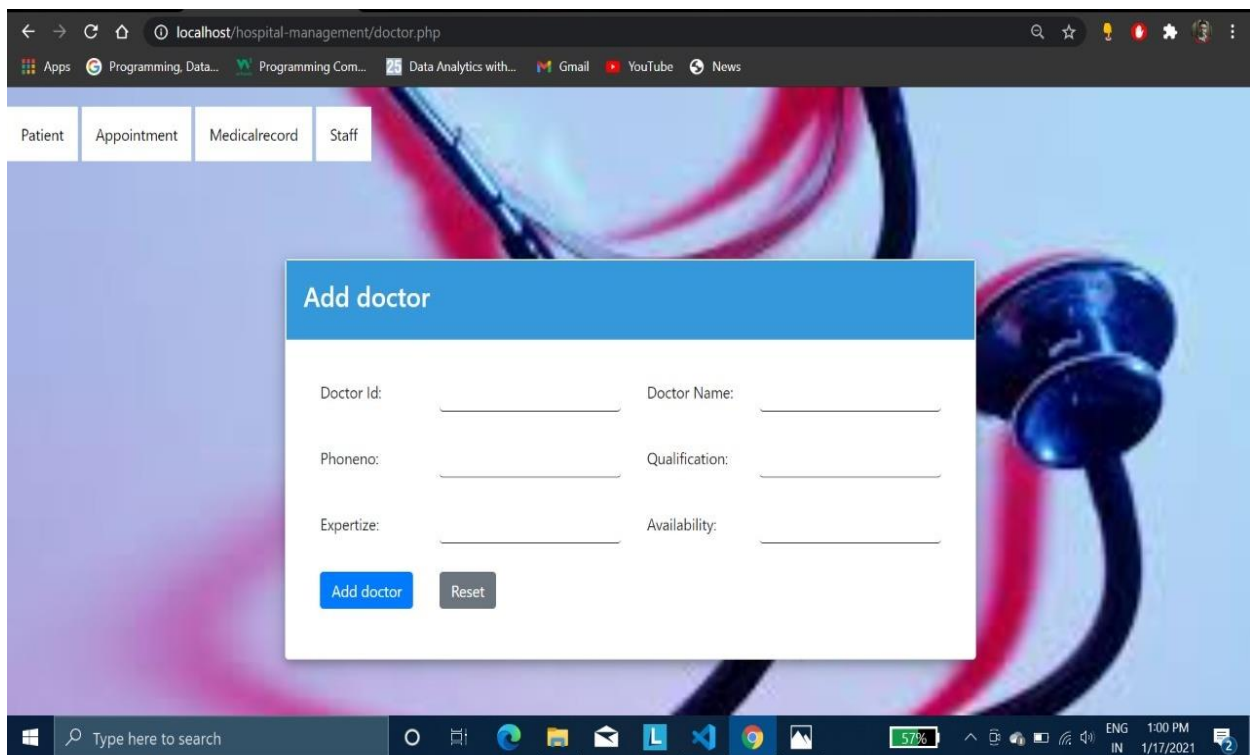
SL.No	Record_Id	Patient Id	Doctor Id	Admission date	Action
1	A1	P1	D1	2021-01-09	
2	A2	P2	D1	2021-01-08	

Figure 4.5 Appointment list

In the above figure 4.5 gives the details of the appointment in the table form where the user can use it to display the appointment list.

Figure 4.6 Doctor form



Add doctor

Doctor Id: Doctor Name:

Phoneno: Qualification:

Expertize: Availability:

Figure 4.6 Doctor form

In the above figure 4.6 is the doctor registration form, the user can register the doctor in the hospital by adding the doctor's details and store it in the database.

Figure 4.7 List of doctor

SL.No	doctor_ID	Doctor_Name	Phone no	Qualification	Expertize	Available	Action
1	D1	Madhava M	93849837849	MBBS	Physicist	8am to 12pm	
2	D3	Shivananda	98374329434	MBBS	Physicist	2pm to 4pm	

Figure 4.7 Doctor details

In the above figure 4.7 is the webpage of doctor list, where the user can view the details of doctors working in the hospital.

Figure 4.8 Nurse Form

Figure 4.8 Nurse Form

The figure 4.8 is the Nurse or Staff registration form, where the user can register the nurses who would want to work in the hospital by storing their information in their database.

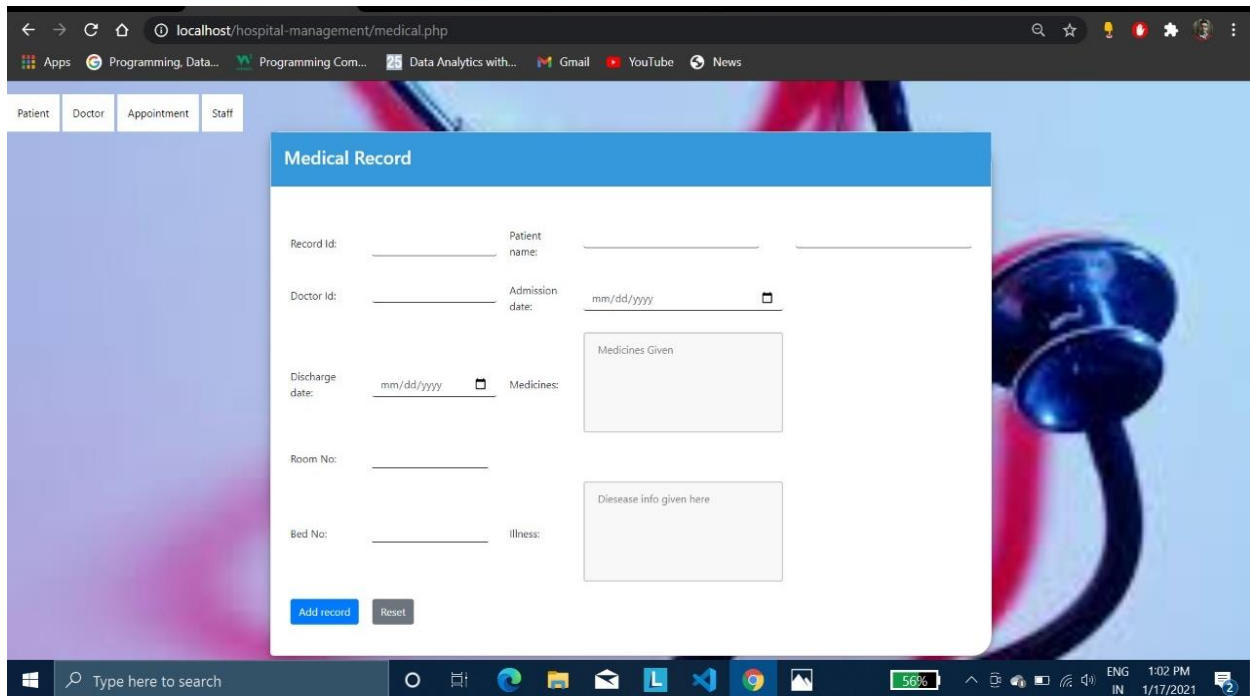
Figure 4.9 List of staff

Nurse_Id	Nurse Name	Nurse Phone	Qualification	Action
N1	Sheetal	8634783748	Bsc	
N2	Soujanya	98774367467	Bs	

Figure 4.9 staff details

The above figure 4.9 contains the information of the nurses working in the hospital, where the user can view the details of the nurses which is in the table form.

Figure 4.10 Medical record form



The screenshot shows a web browser window with the URL `localhost/hospital-management/medical.php`. The browser's address bar and tabs are visible. The web application has a navigation bar with tabs: Patient, Doctor, Appointment, and Staff. The main content area displays a 'Medical Record' form. The form includes the following fields and sections:

- Record Id:
- Patient name:
- Doctor id:
- Admission date:
- Discharge date:
- Medicines:

Medicines Given
- Room No:
- Bed No:
- Illness:

Disease info given here

At the bottom of the form, there are two buttons: 'Add record' (in blue) and 'Reset' (in grey). The background of the form is a light blue gradient with a stethoscope image on the right side. The Windows taskbar is visible at the bottom of the browser window.

Figure 4.10 Medical Record form

The above figure 4.10 is the medical record storing form of the patient, where the user can store the information of the patient admitted in the hospital for a longer period.

Figure 4.11 Medical records

SL.No	Record_id	Patient Name	Doctor Id	Admission date	Discharge date	Room No	Bed No	Medicines given	Illness Report	
1	R1	Ananya kumar	D1	2021-01-08	2021-01-08	1	1	gkhchkcyc	vjykdhcgchgc	
2	R2	Adhitya KC	D1	2021-01-08	2021-01-09	2	2	fafafa	fagaef	
3	R3	Anubhav Khajuria	D1	2021-01-12	2021-01-12	2	1	vagkbikafajsbf	afkvjffgasgasgasga	

Figure 4.11 Record list

The figure 4.11 is the list of medical record stored in the database , where the user can view the records of the patient stored here and can also delete and update the record from here by clicking the dustbin and pencil symbol.

Summery

In this chapter we have covered the result of the project and the snapshots of the project.

CONCLUSION

Implementation of **hospital management system project** helps to store all the kinds of records, provide coordination and user communication, implement policies, improve day-to-day operations, arrange the supply chain, manage financial and human resources, and market **hospital** services.

Future Scope

The proposed software product is the Hospital Management System(HMS).The system can be used in any Hospital,Clinic to get the information from the patients and then storing that data for future usage.

The current system in use is a paper-based system.It is too slow and cannot provide accurate list of patients and also updated medical records of patients for future use and to track the improving of the admitted patients within the reasonable timeframe.The intensions of the system are to reduce the time and to increase the number of patiencss that can be treated accurately.

Future Enhancement

The proposed system is Hospital Management System. We can enhance the project my adding more facilities like pharmacy system for the stock details of medicine and also the total payment of the appointment from hospital management and medical bills in the pharmacy.Providing such features enable the users to include more comments into the system.

REFERENCES

- [1] Fundamentals of database systems, Ramez Elmasri and Shamkant B. Navathe, 7th edition, 2017, Pearson.
- [2] Database management systems, Ramakrishnan, and Gehrke, 3rd edition, 2014, McGraw Hill.
- [3] www.W3schools.com = W3schools Online web tutorials.
- [4] <https://www.php.net/docs.php> = php documentation
- [5] Bootstrap 4 documentation.
- [6]. <https://www.youtube.com/c/webslesson/videos> = videos

