PROJECT REPORT

ON

HOSPITAL MANAGEMENT SYSTEM

BY

SUBHAJIT KARMAKAR

(111801041)

COURSE NAME:-

Database Management System Lab

INDEX

ABSTRACT

In a hospital lots of things keep on happening on a day to day basis. Lots of doctors, nurses and other staff work in a hospital across various departments. A great number of patients visit the hospital for various purposes ranging from visiting a doctor to getting a surgery. A great amount of data is generated due to all these varied kinds of activities happening in a hospital. The hospital administration needs to make sure that they keep track of all the data, if some data is lost or is compromised then the effects can be very dangerous. The hospital administration needs to make sure they have an effective method to store the data, manage the data and manipulate the data.

Our hospital management system project has been built to provide an effective solution to this problem faced by the hospitals. The hospital management system project is aimed to automate the management of Hospital functioning by developing a software which is user friendly, fast, efficient, scalable, simple to use and cost effective. Our Hospital management system provides the hospital administration a safe and secure automated tool to store all the data and perform meaningful operations using the various features provided.

Many hospitals in different parts of the country still keep track of everything in log books by manually writing on them. It often leads to inconsistency when more than one person is maintaining the records and it is very difficult for anyone else to understand how the records are logged and managed, thus making it really inconvenient to read the data. This hospital management system application can potentially be used to assist numerous hospitals and medical centres across the country.

INTRODUCTION

Hospital Management System is aimed to facilitate efficient storage and proper management of data in a hospital. It is designed to enable the hospital management to function efficiently by automating their data management. It will help them to monitor all their key activities within a few clicks. Monitoring these key activities is imperative for proper functioning of the Hospital. It will help the Hospital administration be relieved of the stress of managing all their information and enable them to focus on improving the quality of clinical care provided by the Hospital.

The hospital management system application lets the admin of the Hospital securely log in the system using login credentials. The hospital management app stores the personal data and medical history of patients. It helps the Hospital admin to manage and manipulate all of the details of the patients. The hospital management system stores the services used by the patient, the medicines bought by the patient and the total cost of all the expenses incurred by the patient.

The hospital management system stores the information about the doctors who work in the hospital like the doctor's personal details and specialization etc. It helps the Hospital admin to manage and manipulate all of the details of the doctors.

The hospital management system stores data about the nurses who work in the hospital. It helps the Hospital admin to manage and manipulate all of the details of the nurses.

The hospital management system also stores data about rooms in the hospital and it also stores data about the room allocated to each patient admitted in the hospital and the data about nurses' in-charge of each room. It lets the admin add details of the appointment booked by a patient. It also lets the admin update the details about the current status of any appointment.

It helps the admin to keep a track of all the medicines stored in hospital inventory and automatically update the stock on any purchases made.

USER REQUIREMENT SPECIFICATIONS:-

Our user is expected to be the admin of the hospital. The admin will expect certain features in our application which will enable the admin to perform the below mentioned operations.

A patient's basic details and medical history can be stored in the database by the hospital admin and the hospital admin can update the details of the patients, view the details of the patients and remove a patient from the database if required.

A doctor's basic details along with the specialization and years of experience can be stored in the database by the hospital admin and the hospital admin can update the details of the doctors, view the details of the doctors and remove a doctor if required.

A nurse's basic details can be stored in the database by the hospital admin and the hospital admin can update the details of the nurse, view the details of the nurse and remove a nurse from the database if required.

Appointment can be booked by the admin for the patients. The details of the various previous appointments can be displayed. The appointment status of previously booked appointments can be updated by the admin and any appointment can be deleted when required.

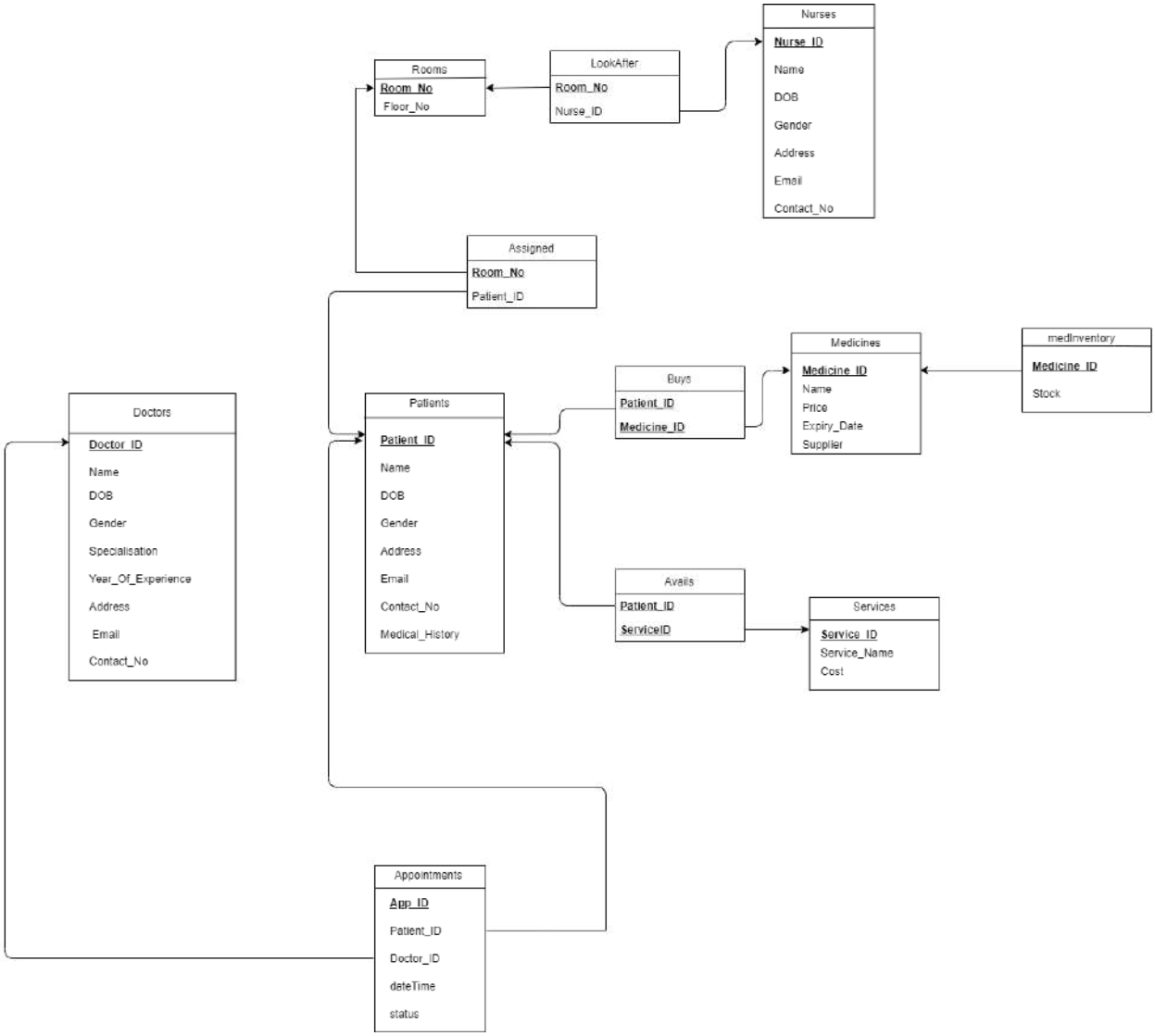
Details of various services provided by the hospital can be stored in the database by the hospital admin and the hospital admin can update details of the services provided, view the details of the existing services and remove a service if required. When a patient avails any service provided by the hospital, the hospital admin has the corresponding feature to update this in the database.

Details of various rooms present in the hospital can be stored in the database by the hospital admin. All the previously available rooms in the Hospital can be displayed. When a new patient is admitted in a room, then the admin can update the details accordingly. Similarly, if a nurse is assigned a duty in any room, the admin can update that details. The admin can also display the details of which room has which patients admitted in that, and also the details of the nurse assigned duty in that room.

Details of various medicines provided by the hospital can be stored in the database by the hospital admin and the hospital admin can update details of the medicines provided, view the details of the existing medicines and remove a medicine if required. When a patient avails buys any service provided by the hospital, the hospital admin has the corresponding feature to update this in the database. The admin can maintain the inventory of medicines available in hospital and can update those details after any patient buys any medicine.

Admin can also find details about the total revenue of the Hospital, and basic details about the no of patients admitted, nof of nurses and doctors working etc.

SCHEMA DIAGRAM:-



In our database, we have the following Entity tables Doctors, Patients, Nurses, Rooms, Services, Appointments, Medicines and medInventory and the following relationship tables Assigned, Avails, Buys, and LookAfter.

In our db schema we have primary keys for all of our entity tables as well as relationship tables. Our relationship tables also have foreign keys which refers to the primary keys of the Entity tables. A relationship table may have more than one foreign key which refers to more than one different table. If there is a one-one relationship between two entity tables then the relationship table may just need to make one foreign key as primary key and that would be enough to uniquely identify any relation in that relationship table. If there is a one-many relationship then the foreign key referring to the many tables is made the primary key of the relationship table. If there is a many-many relationship then more than one foreign key together needs to be made as the primary key.

SOFTWARE REQUIREMENT SPECIFICATIONS:-

Software Requirement specifications refers to defining the software resource requirements and the dependencies that are prerequisites for the application to be properly function in the system. These softwares and technologies and their dependencies must be installed on the system to ensure the expected behaviour of our application.

Software requirement specifications for our Hospital Management System applications are:-

- Php
- MariaDB
- Apache
- Linux operating system

All of these components need to be installed in the system on which our Hospital Management System application is going to be run.

SYSTEM DESIGN:-

The hospital management system application is made up of different components. Each serving a specific purpose. The front end of the application is built using HTML, CSS and JavaScript. For designing the backend of the application we have used MariaDB as the database server and we used php as the server-side scripting language.

The front end or the client side of the application is the part with which the user interacts. The design of the front end determines the experience that the user will have using our application. The front end has been designed using different combinations of text, images, colors, graphics, buttons, toolbars, etc.

Backend is the server-side part of the application. It is responsible for making sure that the user requests are redirected to the proper component which has the corresponding functionalities to serve that request. The backend determines if any data is to be written on to the database or any data is to be fetched from the database and perform those tasks accordingly and gives response back to the frontend, which then renders the response in a format which the user can understand.

In our application the user sends various requests through the forms, buttons etc (mostly implemented using HTML) which we have implemented in our frontend. This request then passes to the appropriate component of the backend. The functional components of the backend are mostly written in php.

These functional components then according to the request take various kinds of actions like sending the request to another component or setting up connection with the MariaDB server and ask for some operations to be done. The request for operations to the MariaDB server is mostly made using views, procedures and functions.

On successful execution of the request by MariaDB server, the functional components then take actions according to the written logic which is specific to the components. If the request was for displaying the data which are mostly implemented using views, then the component sends the data to the frontend to be rendered to the user, or if the request was for updating or inserting any data into the

database which are mostly implemented using procedures, then on successful insertion or update the appropriate message is send back to frontend.

For example, when inserting a doctor in a table, the data is taken input using a HTML form, then on clicking on an appropriate button the post request is sent to a php script. This php script is responsible for setting up connection with MariaDB server and calling an appropriate insert procedure to insert the data in the database. On successful completion of the insertion, some logic sends back a success message to the browser which is displayed to the user.

TECHNOLOGY STACK:-

Technology stack used

Our web application has been developed using LAMP stack.
LAMP stands for Linux, Apache, MariaDB, PHP.

- Linux is the operating system layer. All the other components of the stack run on top of Linux.

- Apache is a web server application that runs on the Linux Operating System. It's role is to establish connection between the web browser and the server. It receives requests from the user and fetches data corresponding to the request to the web browser.

- MariaDB is a relational database management system. It is used to store, manage and query data efficiently.

- PHP is a server side scripting language which combines all the elements of the lamp stack. PHP helps to build dynamic web pages easily. It easily integrates with HTML content.

Working of LAMP stack

LAMP stack works together to form the client-server connection and serves data in an reliable and organized way to be displayed to the user.

At first the user sends a request to the Apache web server application through a web browser. Apache directs this request to PHP. Then the file is loaded and executed. PHP communicates with MariaDB to fetch or store any data as per the code. After data fetching or storing is done, PHP using the data and logic in the file, sends the data back to Apache web server application, Apache then sends it to the web server, then to the browser. Linux OS acts as the backbone of all the above processes and makes all the above operations possible.

Benefits of LAMP Stack

LAMP stack was chosen because:-

- It is made of components which are free and open source.
- LAMP stack is secure and stable.
- It is easy to set up.
- It helps reduce development time.

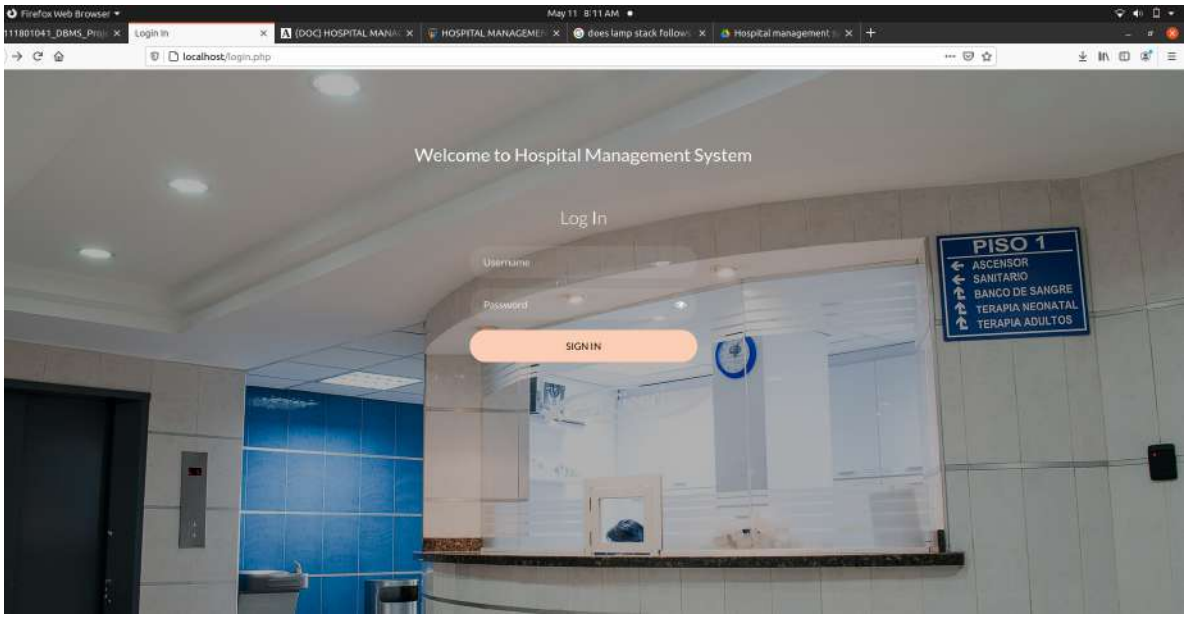
FEATURES PRESENT IN THE APPLICATION:-

- View Doctors :- It lets the user see all the doctors present in the database
- Add Doctors:- It lets the user add a new doctor to the database
- Update Doctors:- It lets the user update details of a doctor who is already present in the database.
- Delete Doctors:- It lets the user delete a doctor from the database
- View Patients :- It lets the user display all the patients present in the database
- Add Patients:- It lets the user add a new patient to the database
- Update Patients:- It lets the user update details of a patient who is already present in the database.
- Delete Patients:- It lets the user delete a patient from the database.
- View Nurses :- It lets the user display all the nurses present in the database
- Add Nurses:- It lets the user add a new nurse to the database
- Update Nurses:- It lets the user update details of a nurse who is already present in the database.
- Delete Nurses:- It lets the user delete a nurse from the database.
- View Appointments :- It lets the user display all the appointments previously in the database
- Book Appointments:- It lets the user book a new appointment and add it to the database
- Update Appointment status:- It lets the user update status of an appointment which is already present in the database.
- Delete Appointments:- It lets the user delete an appointment from the database.
- View Services :- It lets the user display all the services provided by the hospital
- Add Services:- It lets the user add a new service to the database
- Update Services:- It lets the user update details of a service which is already present in the database.
- Delete Services:- It lets the user delete a service from the database.
- Avail Service:- When a patient buys some service provided by the hospital, the admin can make a record of that purchase using this feature.
- View Rooms :- It lets the user display all the rooms present in the hospital
- Add Rooms:- It lets the user add a new room to the database

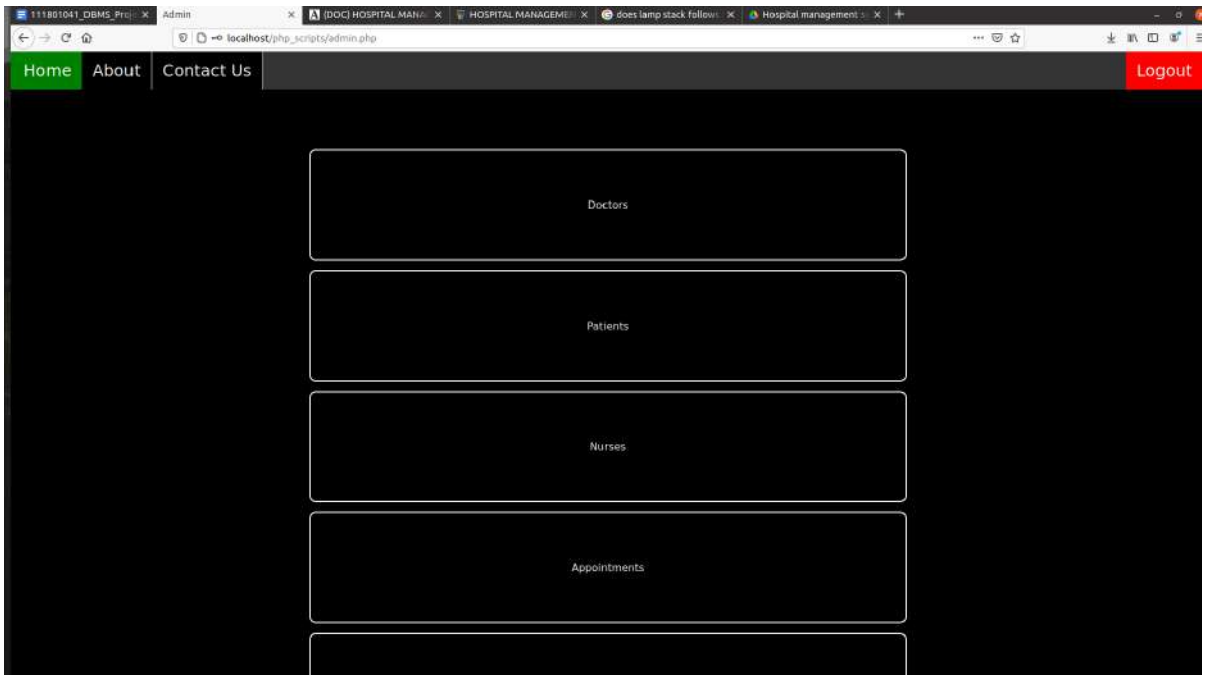
- Delete Rooms:- It lets the user delete a room from the database.
- Assign Patient:- When a patient is admitted to a room, the user can record that information in the database.
- Assign Nurse:- When a nurse is assigned duty to a room, the user can record that information in the database.
- View Nurse Duty:- It displays the room no of the rooms present in the database along with the details of the patient admitted in that room and details of the nurse assigned duty in that room.
- View Medicines :- It lets the user display all the medicines present in the database
- Add Medicines:- It lets the user add a new medicine to the database
- Update Medicines:- It lets the user update details of a medicine which is already present in the database.
- Delete Medicines:- It lets the user delete a medicine from the database.
- Avail Service:- When a patient buys some service provided by the hospital, the user can make a record of that purchase using this feature.
- View Inventory:- It lets the user see the current stock of each medicine in hospital inventory.
- Add medicine to Inventory:- It lets the user add a new medicine and its stock to the inventory.

WORKING OF THE APPLICATION:-

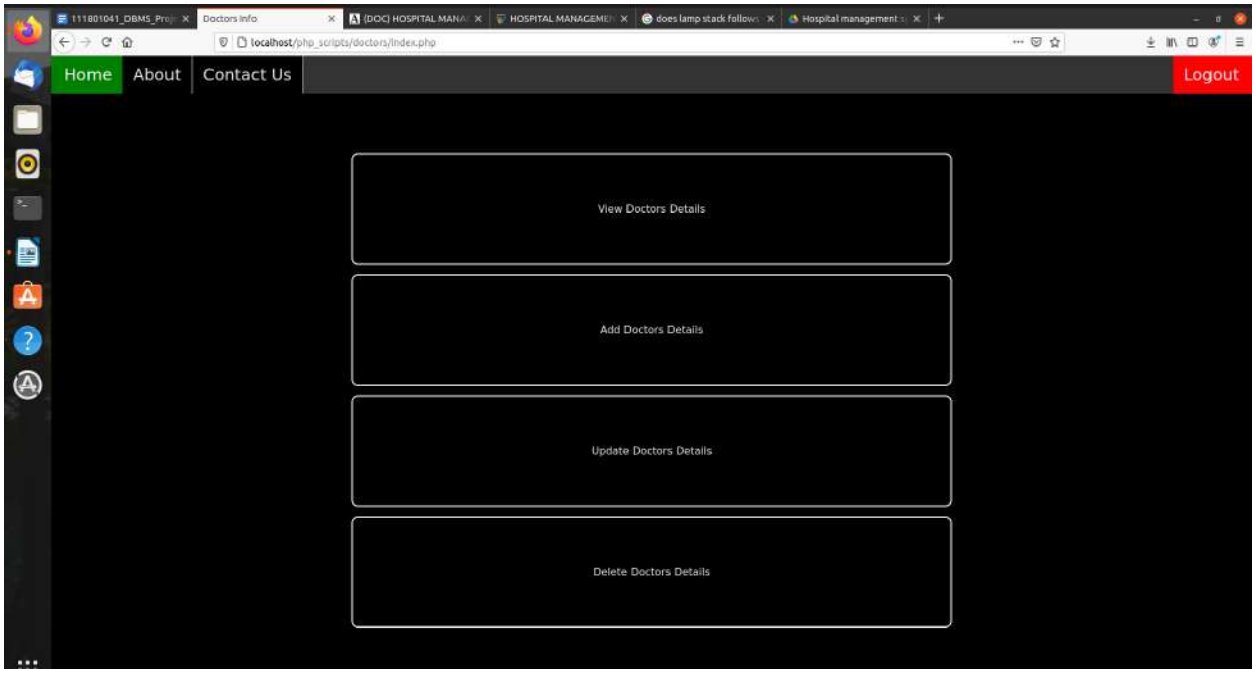
The hospital management system application works in multiple layers. The application starts with a login screen. This is the first layer of our application.



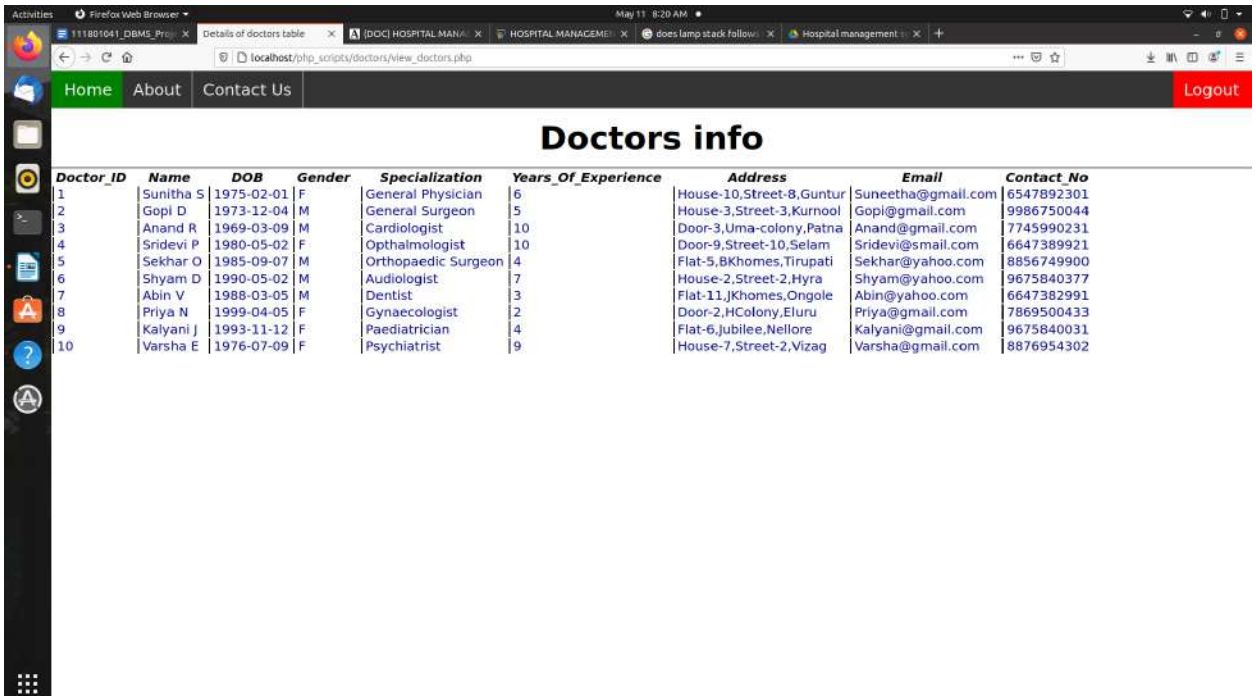
On successful login, it leads to the home page of our application. The home page basically consists of all the features provided by the application. It is the second layer.



On clicking on any of the buttons on the homescreen we go to the third layer which shows the features in that category. Suppose we click on the doctor button. Then we get the following screen showing all the features related to the Doctors table.

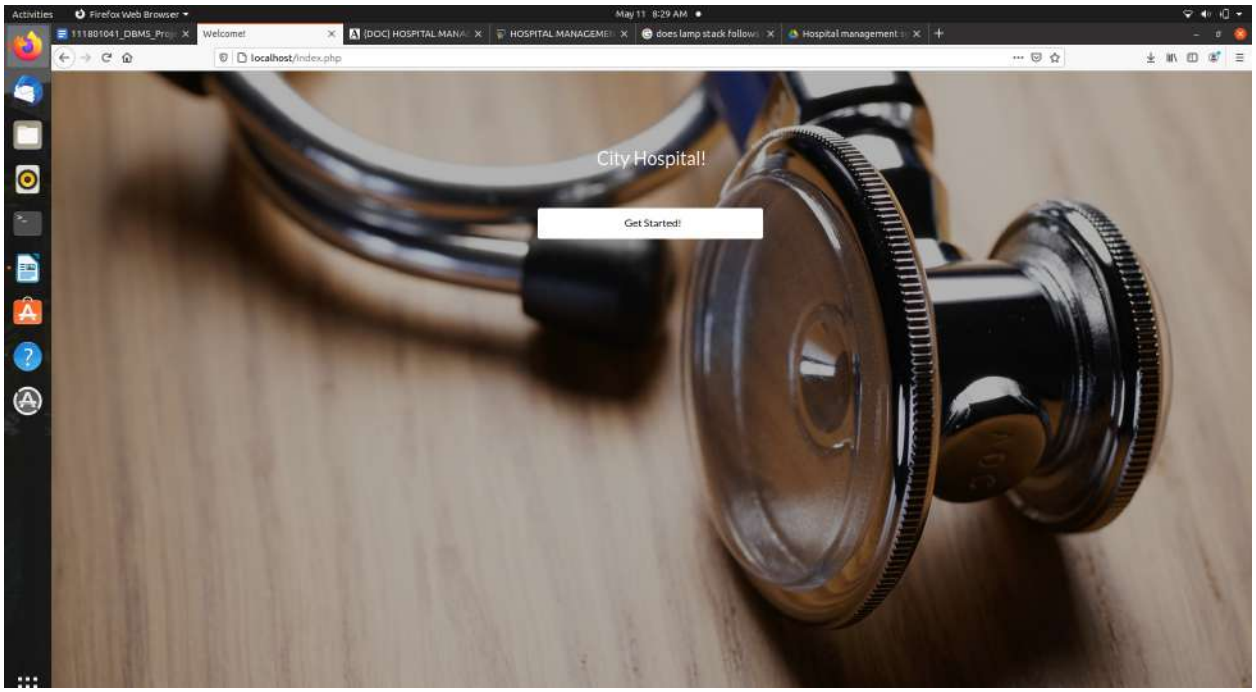


On clicking on any one of the buttons over here. We go to the fourth layer of our project which actually performs the desired operation. Suppose we click on the View Doctors Details button from the above window, then we see the below result.



Doctor_ID	Name	DOB	Gender	Specialization	Years_Of_Experience	Address	Email	Contact_No
1	Sunitha S	1975-02-01	F	General Physician	6	House-10,Street-8,Guntur	Suneetha@gmail.com	6547892301
2	Gopi D	1973-12-04	M	General Surgeon	5	House-3,Street-3,Kurnool	Gopi@gmail.com	9986750044
3	Anand R	1969-03-09	M	Cardiologist	10	Door-3,Uma-colony,Patna	Anand@gmail.com	7745990231
4	Sridevi P	1980-05-02	F	Opthalmologist	10	Door-9,Street-10,Selam	Sridevi@smail.com	6647389921
5	Sekhar O	1985-09-07	M	Orthopaedic Surgeon	4	Flat-5,BKhomes,Tirupati	Sekhar@yahoo.com	8856749900
6	Shyam D	1990-05-02	M	Audiologist	7	House-2,Street-2,Hyra	Shyam@yahoo.com	9675840377
7	Abin V	1988-03-05	M	Dentist	3	Flat-11,JKhomes,Ongole	Abin@yahoo.com	6647382991
8	Priya N	1999-04-05	F	Gynaecologist	2	Door-2,HColony,Eluru	Priya@gmail.com	7869500433
9	Kalyani J	1993-11-12	F	Paediatrician	4	Flat-6,Jubilee,Nellore	Kalyani@gmail.com	9675840031
10	Varsha E	1976-07-09	F	Psychiatrist	9	House-7,Street-2,Vizag	Varsha@gmail.com	8876954302

There are some buttons like Home, About and Contact Us, on clicking on any of those buttons takes us to the required page. If we click on the logout button it logs us out of the application and takes back to the starting page as given in the below result.



CONTRIBUTIONS:-

This project on the hospital management system was successfully completed by efforts put in by each member of the team. All the team members contributed equally in developing the application. Developing our application was done in different phases. In different phases we worked in different modes. Towards the beginning of the project we worked together to design the system of our application. Then as the project progressed to its implementation phase we sorted out the work among us and each worked on a different component.

I worked along with the other team members on designing the db schema, creating the ER diagram, and testing out our database tables by inserting dummy data in the initial phase. As we progressed towards the implementation phase, I worked on designing the various functional components of our application.

I worked on creating some of the views and procedures for our backend.
I designed the control flow in our application, how the different components are interacting with each other. I designed the application architecture in our application involving how the requests from the user are redirected to the required functional components in the application and how the required calls to the database are being made.

I designed the login feature in our application to ensure that the application is secure and data is safe from unauthorized access.

I created some of the frontend components of the application using HTML. The various functional components were written in PHP.

CONCLUSION:-

The Hospital Management System project is for automating the data management task in a hospital. This application solves the day to day needs of the hospital admin by providing a user friendly, easy to use application which stores the hospitals data securely and lets them perform various operations using the data.

Hospital Management System is important for maintaining the details of various patients admitted in the Hospital and various people working in the Hospital.It lets the Hospital administration improve their supervision on different operational units of the Hospital thus helping them improve the operational control and synchronise working of various departments. This would let the hospital administration improve the pace at which operation management is done in the hospital thus helping them to improve the quality of service they provide patients.

On the technical side this application has been built successfully after lots of ups and downs over the past few months. There were lots of brainstorming sessions on the design of the project right from the beginning when we designed the db schema to the end when we planned on how the various functional components should interact with each other. The project journey was filled with lots of learning about the technical intricacies of building a real life application. Apart from the technical aspects, we learnt a lot about working in a team. It is really important to collaborate and coordinate while working in a team and to keep on learning from each other's mistakes and feedback. Sometimes differences in opinion will arise over some potential solution to a problem, but it is essential to reach an agreeable solution quickly.

FUTURE SCOPE:-

- This hospital management system application works only for a single user, who is going to be the admin of the hospital. There is a lot of scope of improvement in this regard. This project can be modified to include multiple logins from different systems for the same user and even logins by different categories of users like login by doctors or nurses can be implemented as well.

- As a real world application works on large amounts of data. Some of the queries used to build this project may not give optimal performance on large amounts of data, those queries need to be optimised for better performance .
- Features like generating bills for patients, maintaining check in and checkout logs of doctors, nurses and other staff can be added as well.

- The authentication system for login can be made more secure.

ACKNOWLEDGEMENT:-

I would like to express my deep sense of gratitude towards my team members Vishal Rao, Aswin S Vijay and Duddu Sritha Hadassah for their continuous cooperation and support right throughout the entire duration of the project. Their sincere efforts and the team spirit were imperative to the successful completion of the project.

I would like to thank our mentor Sneha Mohan for her valuable guidance and feedback throughout the project which helped us improve our work and ultimately led to successful completion of the project.

I would like to extend my sincere gratitude to Dr. Mrinal Kanti Das, our course instructor, for giving us this wonderful opportunity to do this project on such an interesting topic. It helped us apply the knowledge gained in class to build a real life application and learn so many new things.

I also would like to acknowledge the efforts of my friends who have helped me to understand various concepts which led to successful completion of this project.