Introduction to Database Management Systems

PROJECT FINAL REPORT

HOSPITAL MANAGEMENT SYSTEM

HARSHA SARANAM-16230289

NAGARAJ-16234997

AKSHAY-16231179

RAMANA-16231459

Introduction:

It is hard for any person to maintain hand copies for the things they perform. When it comes to maintaining a hospital, it makes difficult for the management to keep track their hand copies about their patients which needs huge warehouses to keep the documents. So, we decided to develop an application for easy access for the patients to track their bills, knowing available timings of the doctors, to get their medication bills.

Objective:

It is aimed to develop to maintain day-to-day state of patients, doctors, reports. It is designed to achieve the following objectives:

- To computerize all details regarding and doctors.
- Fixing the appointment timings.
- The patients can view their billing info from their respective authorized login.
- The info. of the patients are updated and stored in the database for future purposes.

Information Collection & requirements:

In this hospital management system, patient, doctor and receptionist are the departments we have. We are developing web application for this.

Patients department have the values such as patient name, address (city, state, zip), mobile number, birthdate, date they are consulting the doctor and a unique id or email to identify the person. Patient access the system through their login details which are loaded into the database during registration.
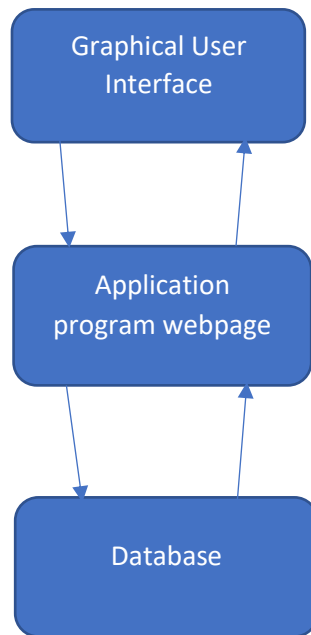
Doctors treat the patients and can view their patient's timings. Doctors have their name, designation, experience and identified uniquely by their doctor id.

Admin generates the bills and sends them to the patient which including the treatment the patients have taken and billed amounts.

Architecture:

We are going to use three tier architecture which comprises of

- Database layer
- Business layer
- Graphical user interface

```
        ┌──────────────────┐
        │  Graphical User  │
        │    Interface     │
        └──────────────────┘
           │           ▲
           ▼           │
        ┌──────────────────┐
        │   Application    │
        │  program webpage │
        └──────────────────┘
           │           ▲
           ▼           │
        ┌──────────────────┐
        │     Database     │
        └──────────────────┘
```

Functional Requirements:

We have different functionalities included in our hospital management system. We have registration facility for patients. And we have different logins for patients, doctor and admin. Each have different functionalities based on their account.

Admin:

Admin can add the details of employees (doctors, nurses) and the bills for the patients that can be viewed in the patient login.

Patient:

Patients can register and login into the system and can view the doctors they need to consult and view their bills.

Working Model:

- **Client application**
- **Data source**

- Working model of the present project includes a regular login page for patient, doctor, admin.
- Patient, doctor, admin enters his customer id and password to able to log in into his own homepage.
- The database verifies the login id with password to retrieve the user's home page.
- Patients page contains doctor's details, appointment timings.
- Patients can opt their required doctor.
- Doctors can view their patients list.
- Admin adds the employees' details into the database.

ER-DIAGRAM:
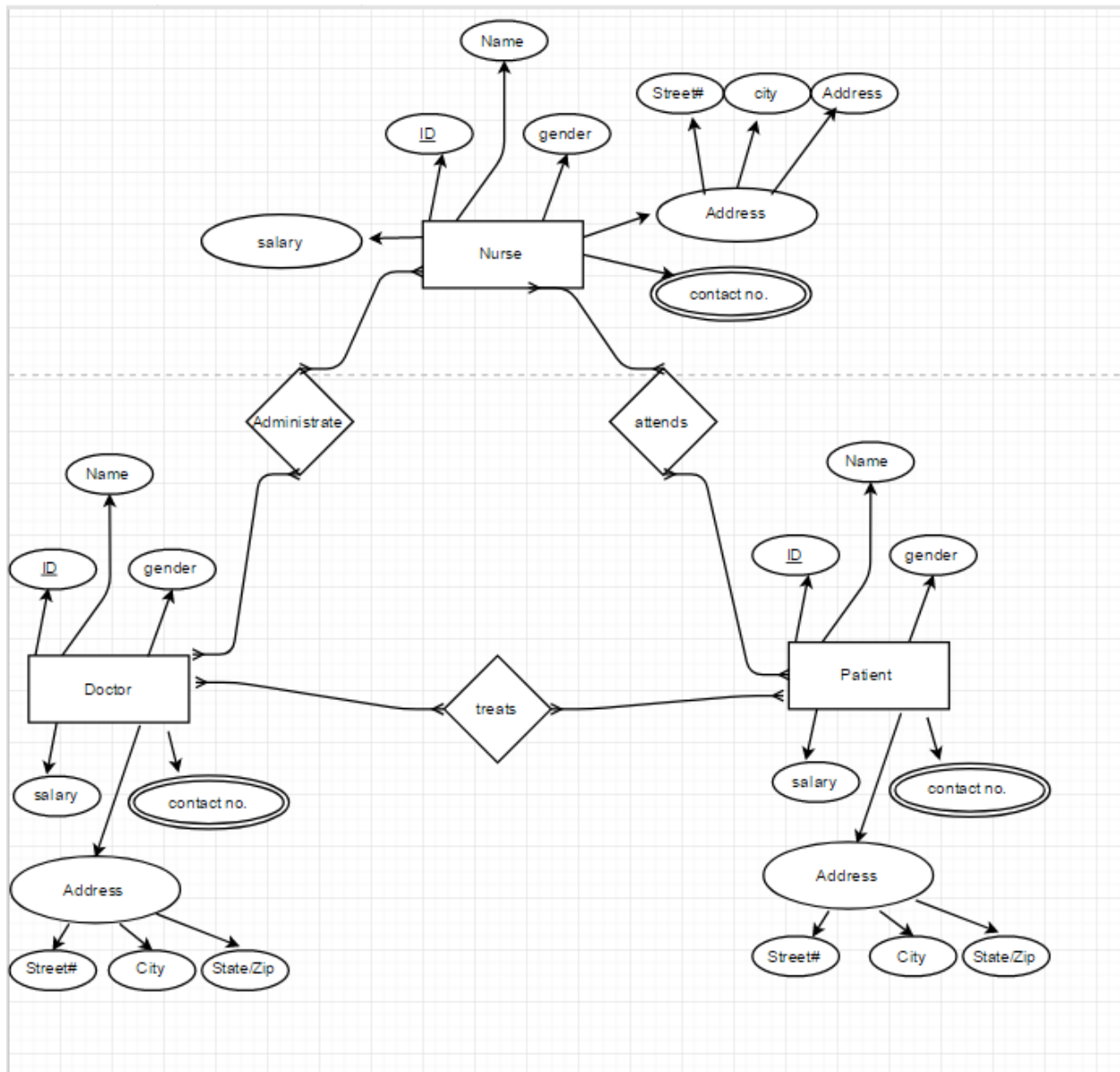
Tools we Used:

MySql:

MySQL is a relational database management system. MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Widenius' daughter,[7] and "SQL", the abbreviation for Structured Query Language. MySQL provides operations such as insert record, update record, delete record, select record, create table, drop table etc.

XAMPP:

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

Platforms used:

Database: MySQL

Web Interface: HTML, CSS, JAVASCRIPT, JQuery.

Backend: PHP

Interface, we designed:

Signup:

Login:

Admin page:

Where he can add the employee details.

List of doctors in database:

Tables creation:

CREATE TABLE SIGNUP

(name VARCHAR(50),

Email VARCHAR(50) primary,

Age INT(20),

Account VARCHAR(20),

Gender VARCHAR(20),

Blood Group VARCHAR(20),

Homeadd VARCHAR(20),

City VARCHAR(20),

State VARCHAR(20),

Zip INT(20),

Phonenumber VARCHAR(20),

Password VARCHAR(20)

);

After values are inserted into SIGNUP:

name	email	age	account	gender	bloodgroup	homeadd	city	state	zip	phonenumber	password
harish	harish@gmail.com	23	Nurse	Male	O+	5008,Rockhill	Kansas City	MO	64110	9849976999	Harish@0032
harsha	harshasaranam@gmail.com	22	Patient	Male	O+	5442 Harrison Street, Apt 2	Kansas City	MO	64110	6692548527	Fuckdude.0032
nagarajureddy myaka	nagarajureddymyaka@gmail.com	23	Doctor	Male	O-	5442 Harrison Street, Apt 2	Kansas City	MO	64110	8168247044	Myaka@445
Nihar Dudam	nihardnp04@gmail.com	23	Doctor	Male	A-	5303 Charlotte St Apt B	Kansas City	Missouri	64110	8165175910	Reddy@4445
ramana	ramana@gmail.com	23	Doctor	Male	A-	5442 Harrison Street Apt 2	Kansas City	MO	64110	8162995454	Ramana@445

You can see at the table with different accounts.

CREATE TABLE Doctor

(name VARCHAR(50),

         Email VARCHAR(50),

         Phonenumber VARCHAR(20),

         Age INT(20),

         SSN VARCHAR(20) primary,

         Department VARCHAR(40),

         Gender VARCHAR(20),

         Blood Group VARCHAR(20),

         Homeadd VARCHAR(20),

         City VARCHAR(20),

         State VARCHAR(20),

         Zip INT(20),

);

After values are inserted into doctor:

name	email	phonenumber	age	ssn	department	gender	bloodgroup	homeadd	city	state	zip
Hann	younghan@gmail.com	678543678	43	566778876	CARDIOLOGY	Male	O+		Kansas City	MO	64110
young zen	youngzen@gmail.com	8168245698	33	679056738	GENERAL SURGERY	Male	B-	5303 Charlotte St Apt B	Kansas City	Missouri	64110
young lee	lee@gmail.com	818245678	44	876486590	NEUROLOGY	Female	A-		Kansas City	MO	64110
buttler zeng	zeng@gmail.com	8168242431	34	945734590	GENERAL SURGERY	Male	A-	5001,Rockhill	Kansas City	MO	64110

```
CREATE TABLE employee

(       name VARCHAR(50),

        Email VARCHAR(50),

        Phonenumber VARCHAR(20),

        SSN VARCHAR(20) primary,

        Bdonate VARCHAR(20),

        Emptype VARCHAR(20),

        Gender VARCHAR(20),

        Blood Group VARCHAR(20),

        Age INT(20),

        City VARCHAR(20),

        State VARCHAR(20),

        Zip INT(20),

);
```

After the values inserted into employee table:

name	email	phonenumber	ssn	bdonate	emptype	gender	bloodgroup	age	city	state	zip
bubby	bubby@gmail.com	2147483647	394628453	Yes	Receptionist	Female	AB-	18	Kansas City	MO	64110
kajal	kajal@gmail.com	2147483647	987234512	Yes	General Duty Assistant	Female	O-	27	kansas	Kansas	64112

Query Implementation:

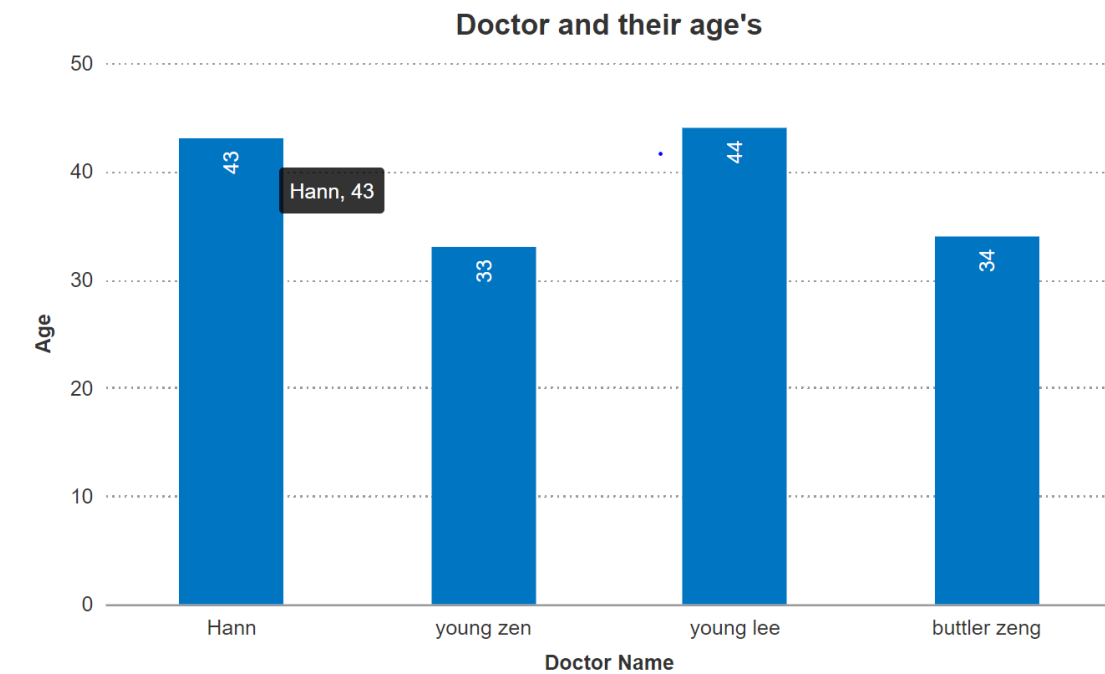The chart below displays the age for the doctors in the form of bar charts.

<?php

        //the SQL query to be executed

        $query = "SELECT name, age FROM doctor";

        //storing the result of the executed query

        $result = $conn->query($query);

?>

The chart below shows the count by gender among the employees.

```php
 $query = "SELECT gender, count(*) as number FROM doctor GROUP BY gender";

$result = $conn->query($query);

while($row = mysqli_fetch_array($result))
            {
                    echo "['".$row["gender"]."', ".$row["number"]."],";
            }
?>

$query = "SELECT gender, count(*) as number FROM doctor GROUP BY gender";

$result = $conn->query($query);

while($row = mysqli_fetch_array($result))
            {
                    echo "['".$row["gender"]."', ".$row["number"]."],";
            }
?>
```
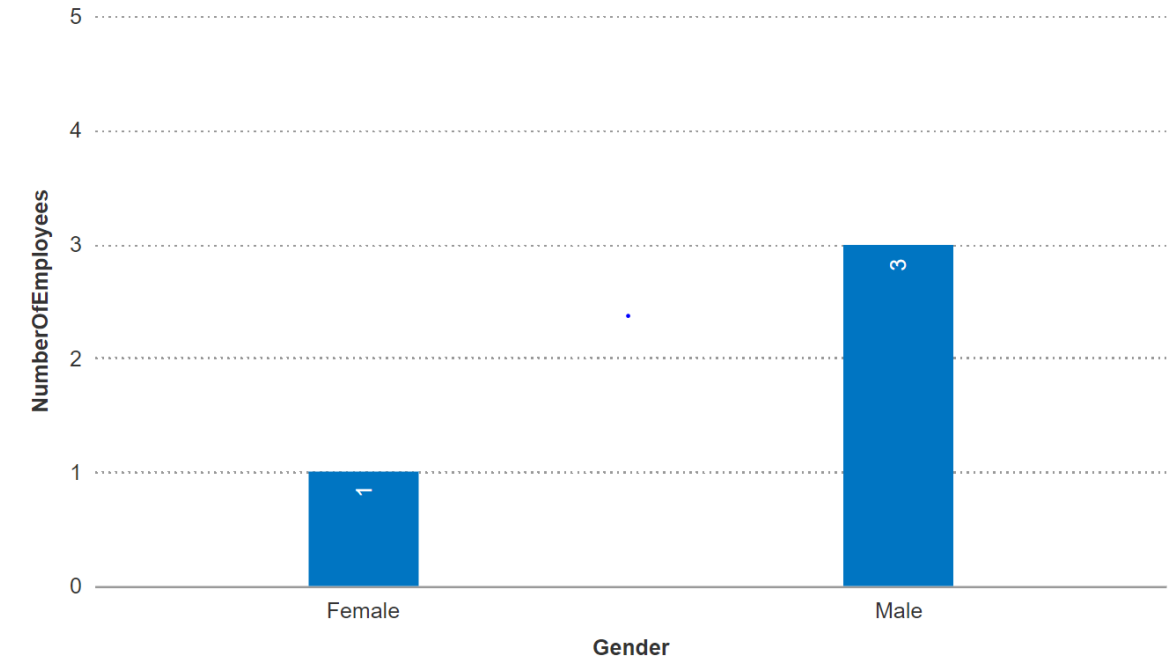
Number of Male to Female Employees

Deficiencies:

- We can provide more user-friendly interface and can include API such as google maps, email for user notifications and API access for the payment.
- We can also provide a database providence for patients where they can store medical bills, X-rays etc.. and can retrieved for future uses.
- We developed this application with local server so it has limited access from outside.
- We didn't provide providence for user to update their details (phone number, SSN etc..).

FUTURE SCOPE & ENHANCEMENT

This project traverses a lot of areas ranging from business concept to computing field and required to perform several researches to be able to achieve the project objectives. There are many features that can be included in future to improve the system.

The area covers include:
- Based on the database of patients, their diseases and treatment analysis can be performed and results can be drawn on most dangerous and spreading diseases so that government can take preventive steps in controlling the diseases.
- A providence can be made such that patients track their health regularly so that doctors can monitor patient's health status and determine required medications.
- Online fee payments can have done by patients which reduces human efforts.
- Patients can order their medicines at online Medical stores only with conformation from doctor.
- PHP Technology used for the development of the application
- General employees such as doctors, nurses and admin as well as the staff and patients will be able to use the system effectively.