

DBMS Mini Project

Hospital Management System

Aarushi Saini- RA1811027010047
Shreya Kumari-RA1811027010053

Abstract

Our project Hospital Management system includes registration of patients, storing their details into the system, and also computerized billing in the pharmacy, and labs. Our software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. It includes a search facility to know the current status of each room. 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

Overview

Organisations such as hospitals have to deal with a lot of patients regularly and hence a lot of data. Hence it is very important for a hospital to have a DBMS with a frontend that easily allows patients to book appointments and allows doctors or administrators to manage patient data.

For this project I have chosen to build the frontend using ReactJS and JavaScript, backend in Node.js and the database used will be MySQL.

Front-end and Back-end

React

React is an open-source, front end, JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications.

Node JS

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser.

Express

Express.js, or simply Express, is a back end web application framework for Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs. It has been called the de facto standard server framework for Node.js.

MySQL

SQL is a domain-specific language used in programming and designed for managing data held in a relational database management system, or for stream processing in a relational data stream management system.

Functional Requirements

1. Separate interfaces for patients and doctors. Patients and doctors should have separate logins.
2. Allow patients to book appointments and give previous medical history.
3. Allow patients to view/update/cancel already booked appointments if necessary.
4. Allow doctors to cancel appointments.
5. Cancelled appointments should create free slots for other patients.
6. The system should avoid clash of appointments.
7. The system should take into consideration hospital and doctor schedules and allow appointments only when a doctor is not already busy or does not have a break.
8. Doctors should be able access patient history and profile, and add to patient history.
9. Doctors should be able to give diagnosis and prescriptions.
10. Patients should be able to see complete diagnosis, prescriptions and medical history.

Connectivity steps

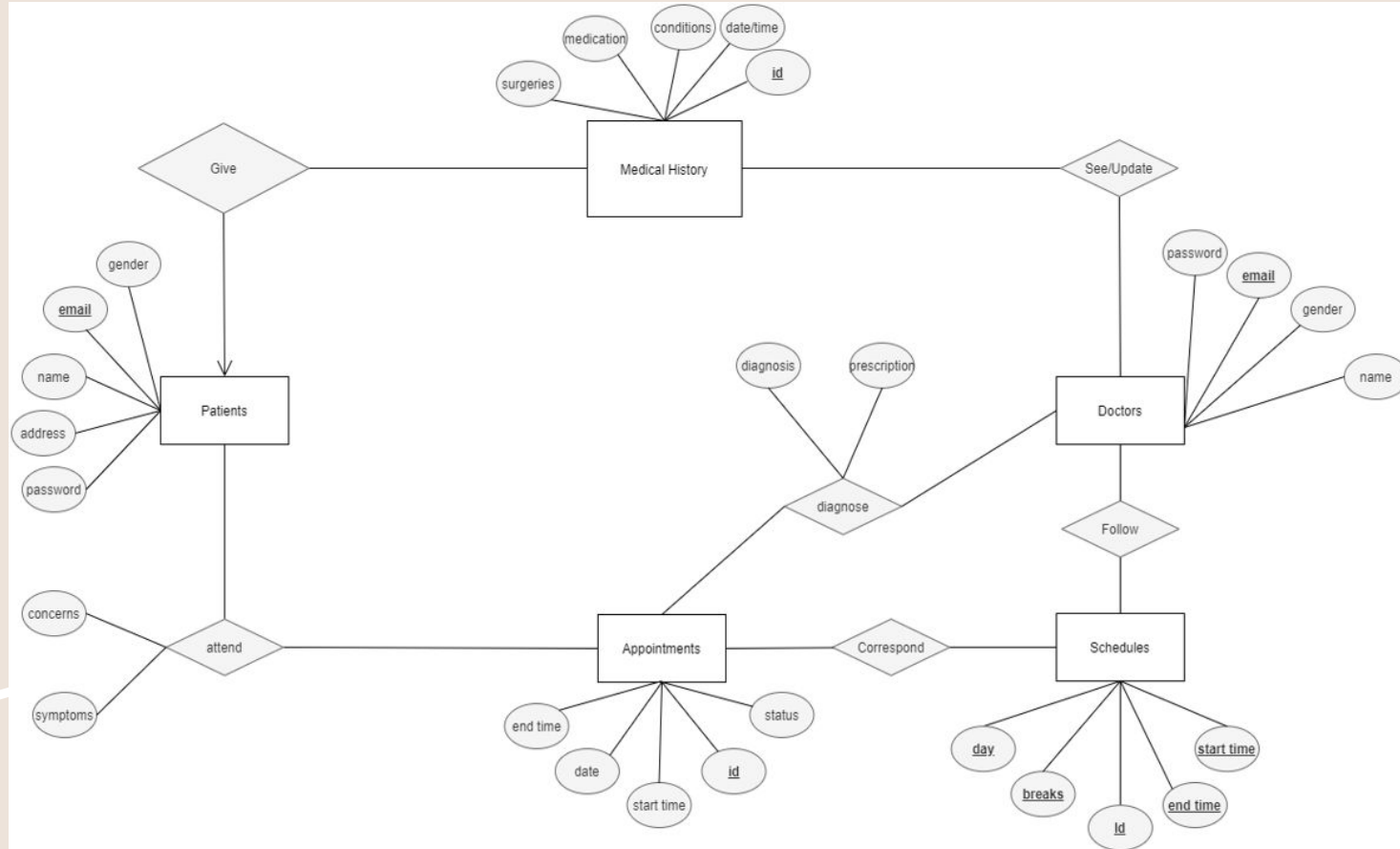
1. Set Up the API Server
2. Install required modules
3. Enable CORS
4. Configure Your Database Connection
5. Configure a User
6. Define schema and fire query

```
1 var createError = require('http-errors');
2 var express = require('express');
3 var path = require('path');
```

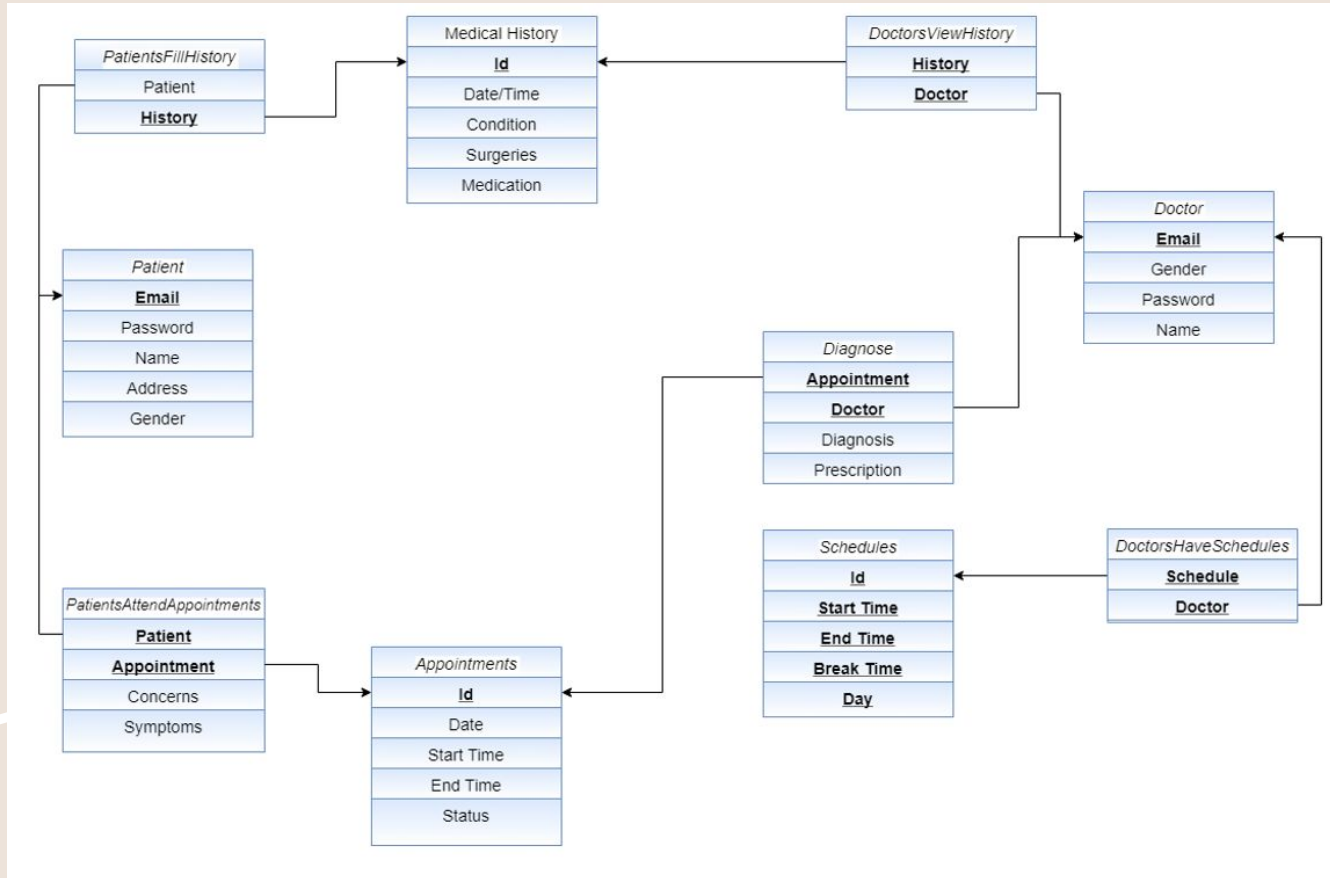
```
6 var mysql = require('mysql');
7 var cors = require('cors');
8 var port = 3001
9
10 //Connection Info
11 var con = mysql.createConnection({
12     host: 'localhost',
13     user: 'root',
14     password: '',
15     database: 'HMS',
16     multipleStatements: true
17 });
18
```

```
20 con.connect(function (err) {
21     if (err)
22         console.log(err.message);
23     else
24         console.log("Connected Successfullllyyyy to sql");
25 });
```

ER Diagram



Normalized Relational Schemas



Demo screenshots

HMS

SIGNUP
PAGE:

Doctor's registration form:

First Name

Please enter your first name.

Last Name

Please enter your last name.

Email

Please enter your email.

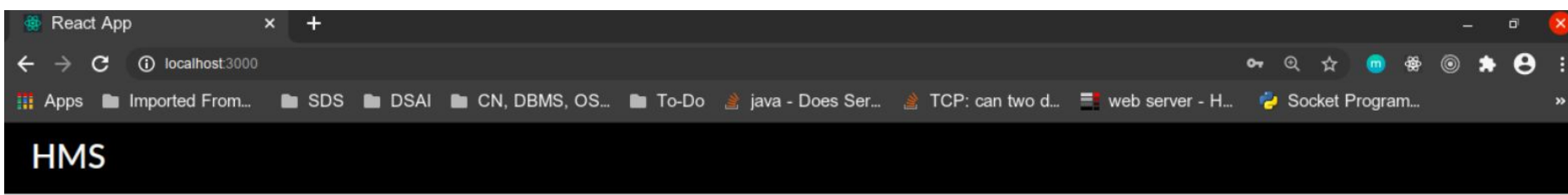
Schedule No

Please enter schedule number

Gender

Female or Male

Password



LOGIN Page:

Email

Password

●●●●●●●●●●

☒ I'm a doctor

Log In

Create Account

React App

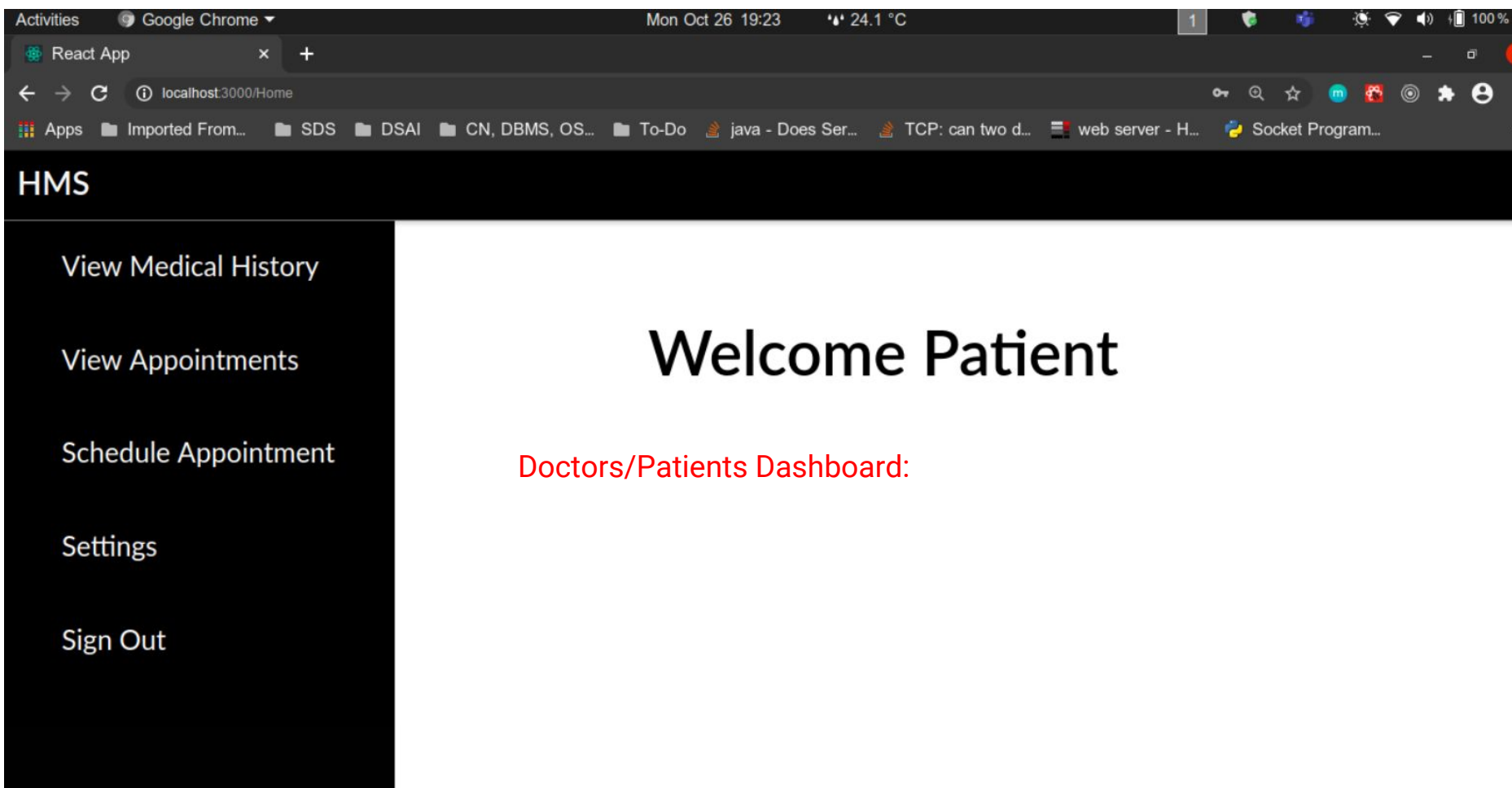
localhost:3000/AppList

Apps Imported From... SDS DSAI CN, DBMS, OS... To-Do java - Does Ser... TCP: can two d... web server - H... Socket Program...

HMS

ID	Name	Date	Start Time	Concerns	Symptoms	Status		
5	Ramesh	17/01/2021	11:00:00	Indigestion	Acidity	NotDone	Diagnose	Cancel
4	Ramesh	23/10/2020	15:00:00	Fever	Cough	NotDone	Diagnose	Cancel
1	Ramesh	15/01/2019	09:00:00	none	itchy throat	Done	Diagnose	
7	Suresh	10/07/2021	09:00:00	Myopia	Not able to see	NotDone	Diagnose	Cancel
6	Suresh	30/10/2020	16:00:00	Sprain	Muscle Pain	NotDone	Diagnose	Cancel

Client's health history and schedules appointments



SCHEDULE APPOINTMENTS
Hrishikesh Athaiye (Nathaiye) ▼

SCHEDULE TIME:

17/10/2020 15:00 

Enter your concerns...

Enter your symptoms...

Attempt To Schedule

HMS

Password Change

Change
password

Old Password

New Password

Change Password

Conclusion

We created a database that a market can use for keeping track on its patients and health. Patients are divided into are allowed to signup and login and doctors. This app help maintains the track of patients health history, issues and schedule appointments.

During our database management course we have learned about the basics of database design. This project gave us the opportunity to try our new skills in practice. While doing this project we also gained deeper understanding on database design and how it can be implemented in real life situations. We believe we can use our database designing skills also in other school projects.

Thank You!