SUBJECT NAME: DATABASE MANAGEMENT SYSTEM

SUBJECT CODE: 18CSC303J

TITLE : HOSPITAL MANAGEMENT SYSTEM

MINI PROJECT REVIEW 2 REPORT

TEAM MEMBERS AND REGISTRATION NUMBER:

1. NANDINI (RA2011031010003)
2. SHIVENDRA BHARUKA (RA2011031010020)
3. STUTI JAIN (RA2011031010031)

DATE OF REVIEW: 16/03/2023

FACULTY NAME : DR. S. THENMALAR

DESIGNATION: ASSOCIATE PROFESSOR

ABSTRACT :

Hospital Management System (HMS) is a crucial component of the healthcare industry, as it streamlines various processes and enhances the overall efficiency of healthcare organizations. This system integrates various functions, such as patient management, medical records, appointment scheduling, billing, and inventory management, into a single platform. The use of HMS leads to improved patient care, reduced costs, and increased productivity. This project focuses on the design and implementation of a comprehensive HMS, with a user-friendly interface and advanced features to meet the needs of healthcare organizations. This project Hospital Management system includes registration of patients, storing their details into the system, and also computerized updating of diagnosis and prescriptions on the patient's page. The software has the facility to create a unique id for every patient and store the entire medical history. It includes a search facility for the doctor to know the current status of each patient.The Hospital Management System can be entered using a username and password. It is accessible either by an administrator, doctor or a patient. 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. Hospital Management System is an organized computerized system designed and programmed to deal with day to day operations and management of the hospital activities. The program can look after inpatients, outpatients, records, database treatments, status illness, billings in the pharmacy and labs. It also maintains hospital information such as ward id, doctors in charge and department administering. The major problem for the patient nowadays is to get a report after consultation. Many hospitals manage reports in their system but it's not available to the patient when he / she is outside. In this project we are going to provide the extra facility to store the report in the database and make it available from anywhere in the world. The project Hospital Management system includes registration of patients, storing their details into the system, and also computerized billing in the pharmacy, and labs. The 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. Users can search for the 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 is well protected for personal use and makes the data processing very fast. The Hospital Management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals. Hospital Management System is designed for multispeciality hospitals, to cover a wide range of hospital administration and management processes. It is an integrated end-to-end Hospital Management System that provides relevant information across the hospital to support effective decision making for patient care, hospital administration and critical financial accounting, in a seamless flow. Hospital Management System is a software product suite designed to improve the quality and management of hospital management in the areas of clinical process analysis and activity-based costing. Hospital Management System enables you to develop your organization and improve its effectiveness and quality of work. Managing the key processes efficiently is critical to the success of the hospital and helps you manage your processes.

LIST OF GENERAL AND UNIQUE SERVICES IN THE DATABASE APPLICATION :

The Hospital Management System has always been one of the most critical solutions to be implemented in a hospital. An advanced cloud based HMS can significantly boost the productivity of a hospital and save up to 50% of operational costs.

A reliable hospital management system can give you more detailed insights and empower you to make better decisions in the future. But it won't be easy to get one until you become aware of a few features of HIMS that are necessary for any size hospital.

During the investment decision on HMS, pick several issues that you think features of the hospital management system can fix. The decision you take here will help you not only achieve cost-saving but also save you from the false promises of several HMS providers.

GENERAL FEATURES :

Patient side features:

- There is a separate interface for patients. Patients have a separate login.
- Patients can book appointments. Patients can view/update/cancel already booked appointments if necessary.
- Patients are able to see complete diagnoses, prescriptions, and medical histories.

Doctor side features:

- There is a separate interface for doctors. Doctors have a separate login.
- Doctors are able to access patient history and profile, and add to patient history.
- Doctors are able to give diagnosis and prescriptions.

UNIQUE FEATURES:

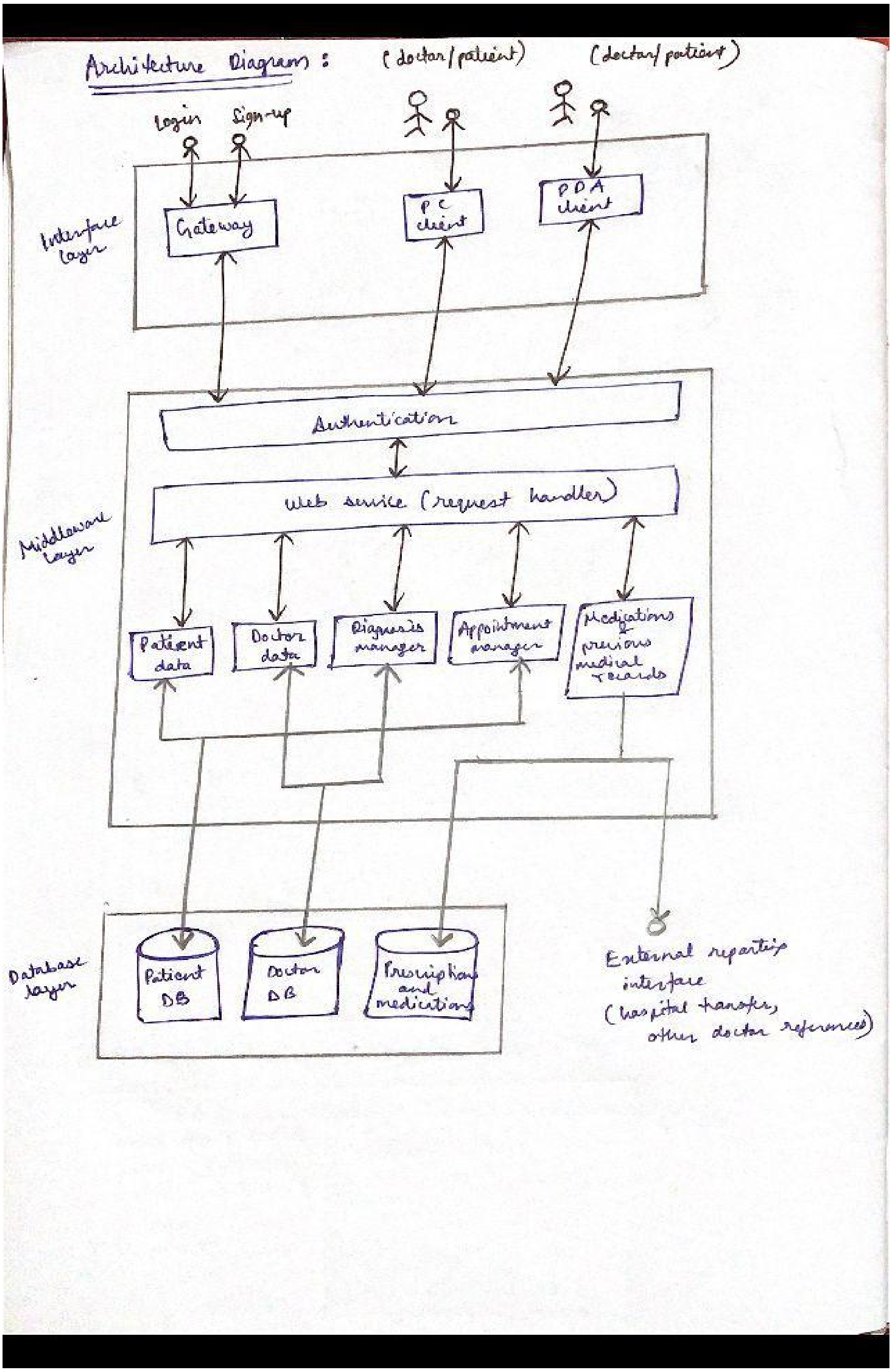
Patient side features:

- Patients can enter their previous medical history.
- Canceled appointments create free slots for other patients.
- The system avoids clashes of appointments with other patients. Each patient is therefore ensured his/her slot.
- Patient medical history is only available to the doctor with whom the appointment is booked to ensure privacy.

Doctor side features:

- The system takes into consideration doctor schedules and does not allow appointments when a doctor is already busy or has a break.
- Doctors are able to modify diagnosis and prescriptions.

ARCHITECTURE DIAGRAM AND EXPLANATION :



The Interface layer provides the application with a program with which it can access, communicate and configure the middleware component.

- In the gateway we have a login and sign up feature where the users can sign up with existing credentials or sign up with completely new credentials by creating them first.
- The PC Client and the PDA Client are present, that is both mobile, web and other devices like tablets and iPads will be supported.
- All clients will have to go through the authentication, be it a doctor or a patient. Only upon successful authentication, the patient or the doctor can gain access to the Hospital Management System interface.

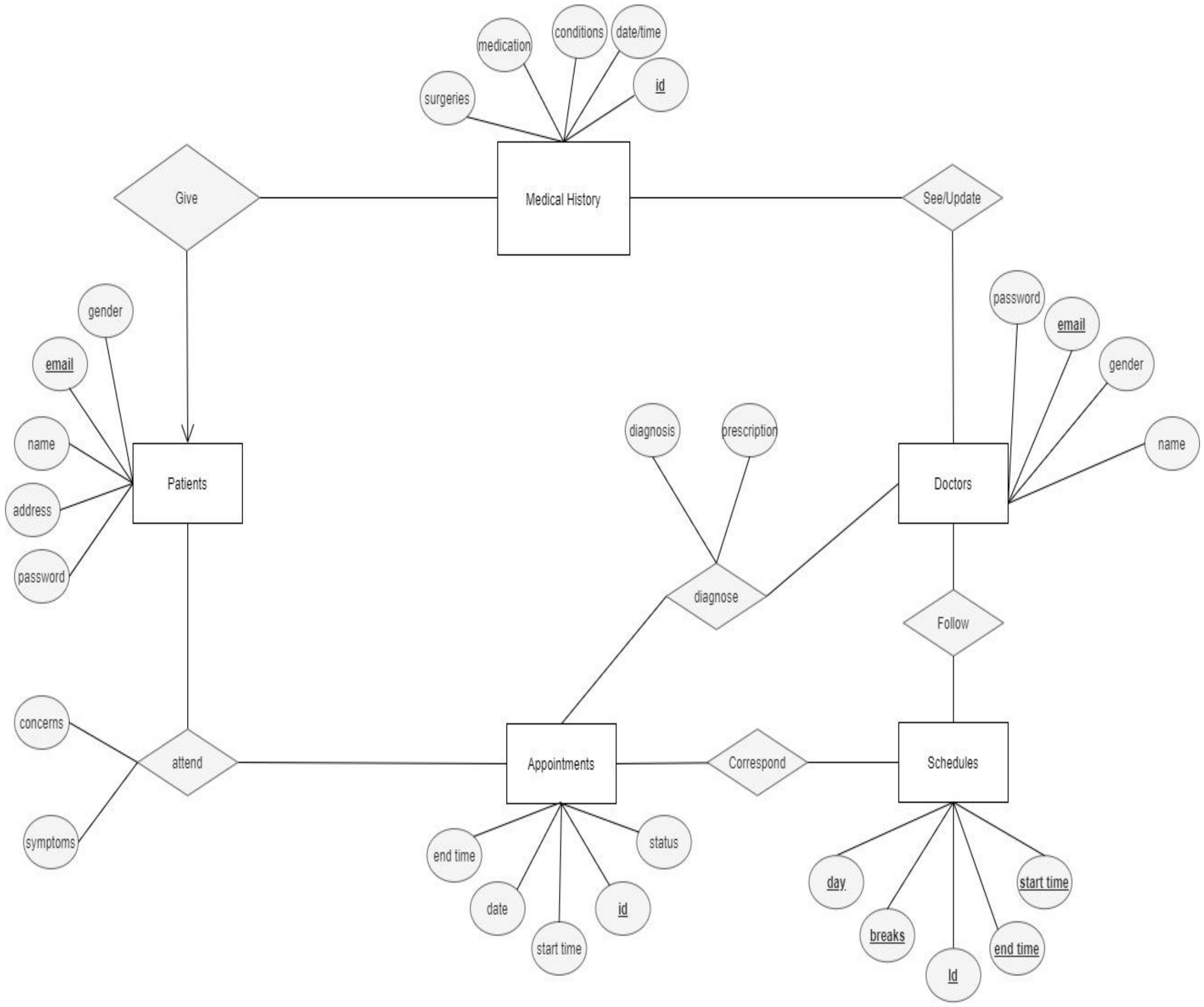
The role of the middleware layer is to enable and ease access to those back-end resources. Middleware programs will typically provide a messaging service for applications to transfer data.

- The Middleware layer has the authentication feature.
- The web service is also known as the request handler which displays the array of available services once the authentication is successful.
- Further we have the Patient Data and the Doctor data which is maintained according to the data entered, modified and retrieved by the users.
- It also works for the process of Diagnosis Manager.
- To book and manage appointments of patients and doctors, where the patients can easily book, modify and cancel their appointments
- Manages medications and previous medical records.

The database layer consists of technologies that give your Web and mobile applications the ability to store and retrieve data. Databases organize data into tables and create relationships between the data in those tables. It consists:

- The Patient Database
- The Doctor Database
- Prescription and Medication
- Patient data and Appointment manager is included under the patient database.
- Doctor data and Diagnosis manager comes under the doctor database.
- Prescription and medication includes medication and other previous medical records.
- Here, it also contains External Reporting Interface which includes any hospital transfer or other doctor references as per the patient's health and choice.

ENTITY-RELATIONSHIP DIAGRAM:



The above shown ER diagram has the following entities:

- Medical History: Id, date/time, conditions, medication and surgeries.
- Doctors: Email, password, gender and name.
- Schedules: Start time, end time, Id, breaks, and day.
- Appointments: Id, status, start time, date and end time.
- Patients: Email, gender, name, address and password.

The above ER diagram has the following relationships:

- Give: Patients give the medical history.
- See/Update: Doctors see or update the medical history of the patient.
- Follow: Doctors follow schedules.
- Correspond: Appointments correspond to schedules.
- Diagnose: Doctors diagnose patients in the appointments.
- Attend: Patients attend the appointments

The first step is to identify the entity sets. As per the requirements, we will have some main entities. For example, if we are required to observe which patient goes to which hospital or whether they have a previous record or not, or if you simply want to analyze the number of patients a doctor treats.

The second step is to map out the attributes of the entities, (including the identification of key attributes). The key attributes for each entity have been listed below:

Medical History: **Id**, date/time, conditions, medication, and surgeries.
Doctors: **email**, password, gender, and name.
Schedules: **start time**, **end time**, **Id**, **breaks**, and **day**.
Appointments: **ID,** status, start time, date, and end time.
Patients: **email**, gender, name, address, and password.

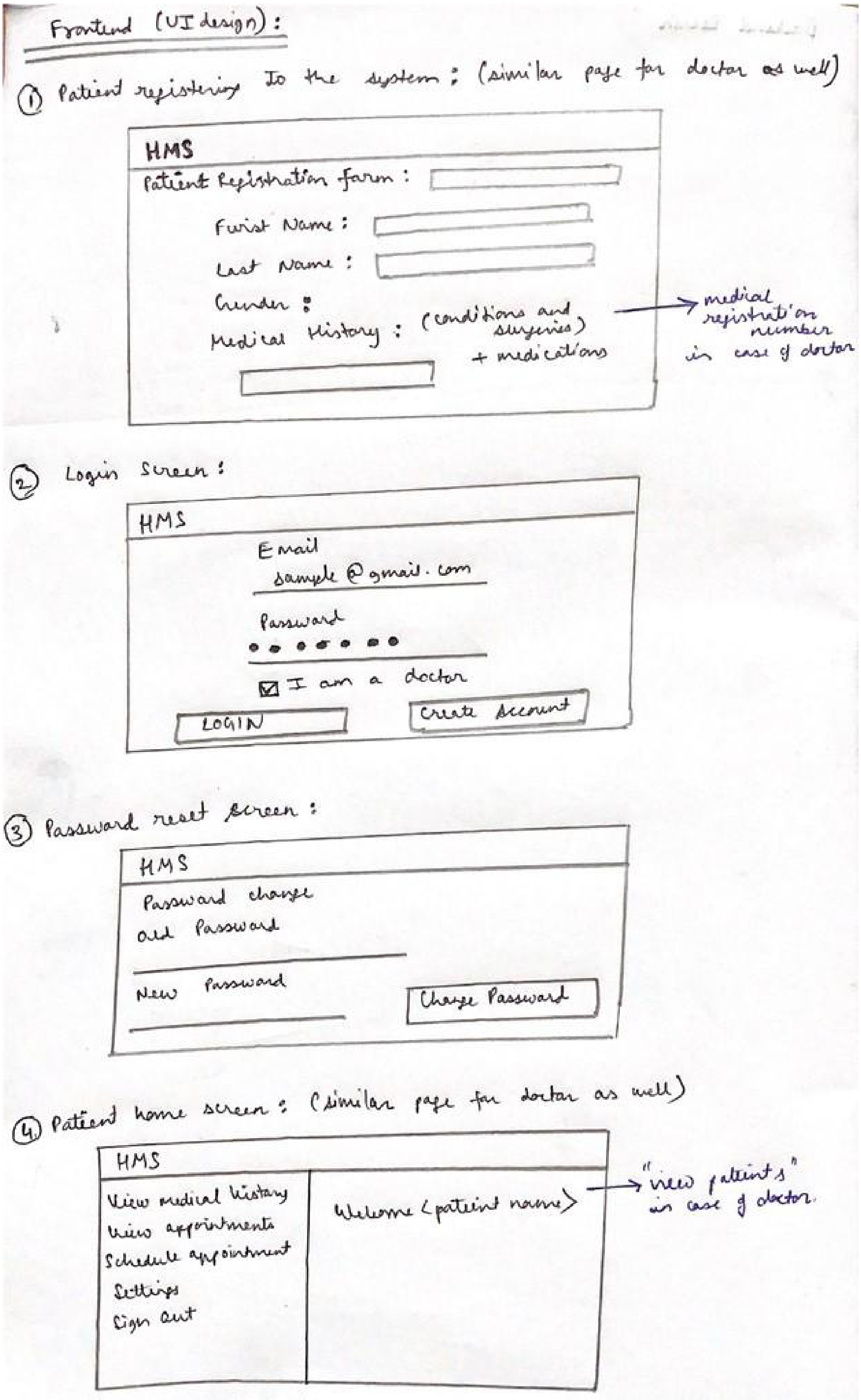
Identify the type of relationship that exists between the entities. This can be done by identifying primary and foreign keys. For example, if the hospital table makes a foreign key reference to the patient ID of the patient table, then both of them will be joined together. Based on the cardinalities, you have to place the appropriate sign.

Once we have identified all the relationships, it is time to map out the lines.

- Since a hospital has multiple patients, it is a one-to-many relationship.
- Since a single hospital has many doctors, the relationship that exists is one-to-many.
- Since a doctor is associated with many patients, it is a one-to-many relationship.
- Since a single patient has multiple medical records, it is a one-to-many relationship.

The last step is to combine all the relationships and draw a complete ER diagram.

FRONT-END UI DESIGN:



Frontend (UI design):

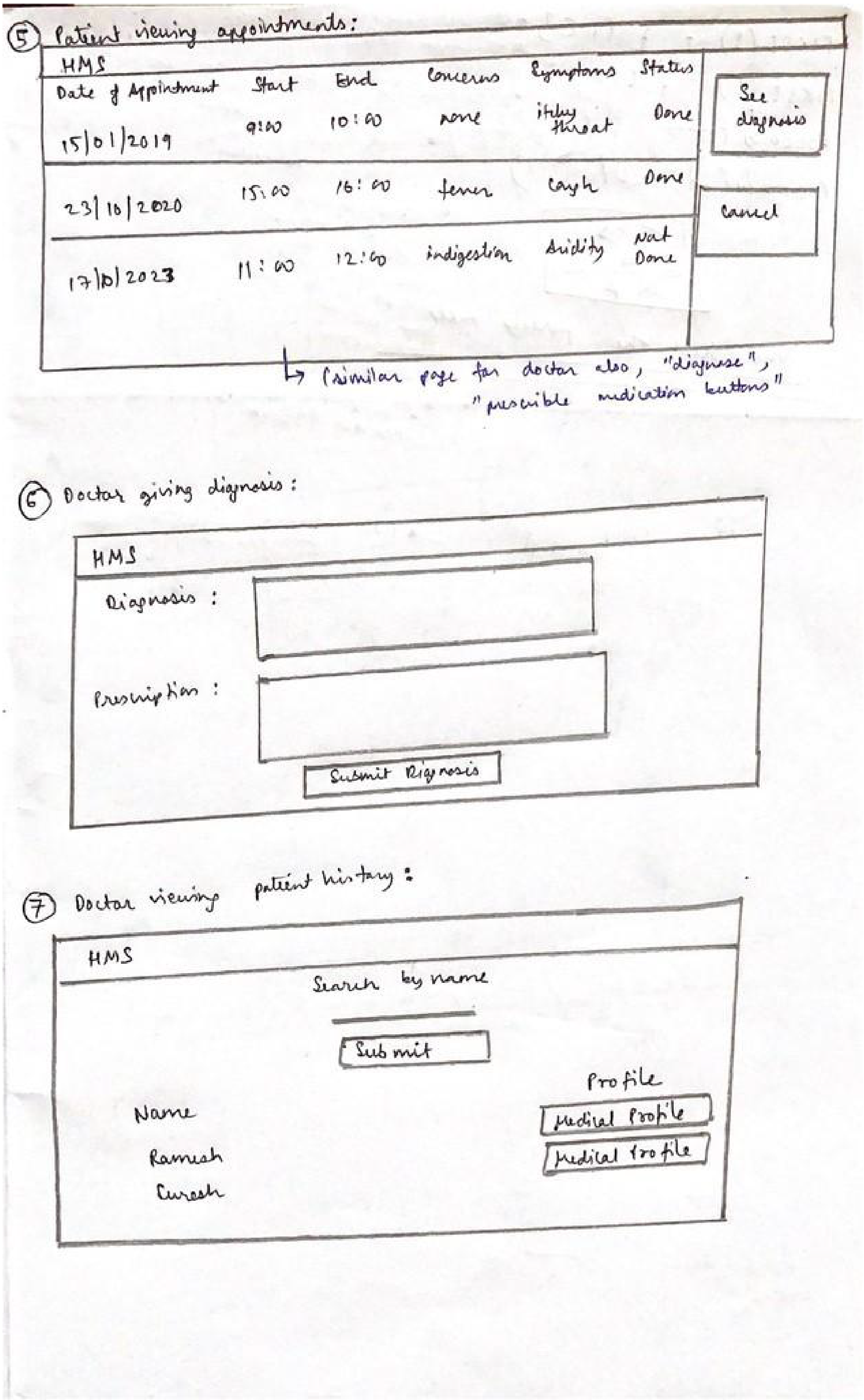
① Patient registering To the system: (similar page for doctor as well)

HMS
Patient Registration form :
First Name :
Last Name :
Gender :
Medical History : (conditions and surgeries)
+ medications
→ medical registration number in case of doctor

② Login Screen :

③ Password reset screen :

④ Patient home screen : (similar page for doctor as well)



5) Patient viewing appointments:

HMS

Date of Appointment	Start	End	Concerns	Symptoms	Status
15/01/2019	9:00	10:00	none	itchy throat	Done
23/10/2020	15:00	16:00	fever	cough	Done
17/10/2023	11:00	12:00	indigestion	Acidity	Not Done

↳ (similar page for doctor also, "diagnose", "prescrible medication buttons"

6) Doctor giving diagnosis:

HMS

Diagnosis :

Prescription :

7) Doctor viewing patient history:

HMS

Search by name

Name	Profile
Ramesh	Medical Profile
Curesh	Medical Profile

SOFTWARE USED:

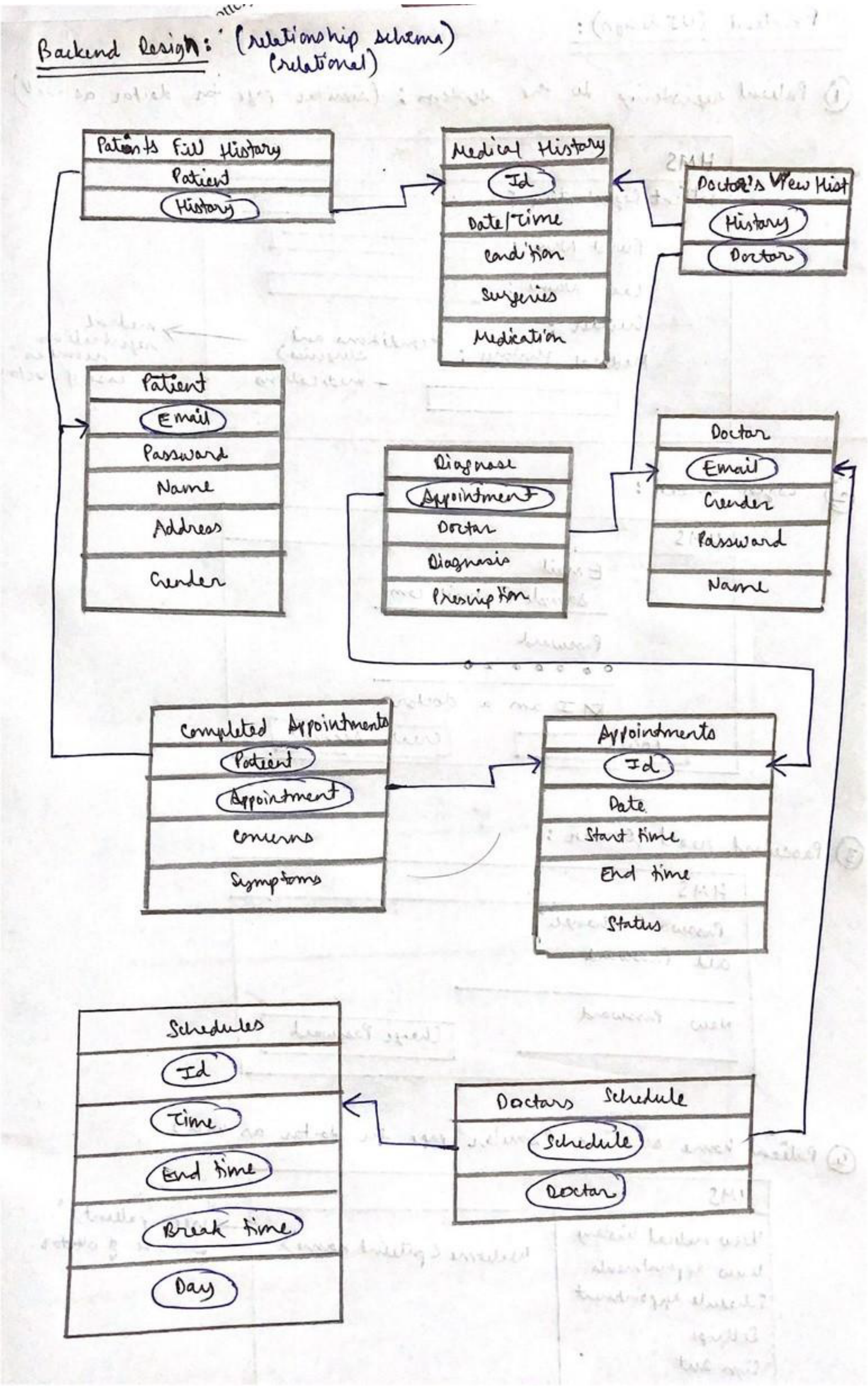
- React.JS

EXPLANATION:

Since we have not yet started implementing the project, we have sketched the UI designs in with a pencil to show an estimated look of the final web application.

- Patient/Doctor registering for the system: Includes the fields of patient registration number provided by the hospital itself, first name, last name, gender and an option to enter the medical history.
- Login screen: Includes the fields of email and password, but also includes two checkboxes having the options of “I am a doctor” and “I am a student”. Simultaneously, two buttons will be displayed, login and create an account.
- Password reset screen: Prompts the user to reset the password by entering the new password and then entering the new password again to make sure there are no typing errors.
- Patient or doctor home screen: Displays a welcome message for the user and shows an array of available web services on a left panel, the services being:
    1. View medical history
    2. View appointments
    3. Schedule appointments
    4. Settings
    5. Sign out
- Patient viewing appointments: Shows a tabular view of the previous and upcoming appointments and finally, two buttons of “see diagnosis” and “cancel” will also be provided for convenience. The table includes:
    1. Date of Appointment
    2. Start time
    3. End time
    4. Concerns
    5. Symptoms
    6. Status
- Doctor giving diagnosis: Two separate text fields will be prompted for the doctor to enter the diagnosis and the prescription medications, and then a submit diagnosis button to save the diagnosis to the database.
- Doctor viewing patient history: Here the doctor can search the patient history by name by submitting the full name. On a successful search, the name and hyperlinked medical profile will be displayed.

BACK-END DESIGN:



SOFTWARE USED:

- Node.JS, Express

TYPE OF CONNECTIVITY USED FOR DATABASE ACCESS:

Database connectivity refers to the ability of an application or program to interact with a database in order to perform operations such as querying, inserting, updating, or deleting data. Database connectivity typically involves the use of a database driver or connector, which is a software component that enables communication between the application and the database.

The database driver acts as an intermediary between the application and the database, translating the application's requests into the specific commands and protocols understood by the database. Different database management systems (DBMS) require different drivers or connectors, as well as different programming interfaces and protocols.

Any of the following connections can be used:

Microsoft SQL Server: Connects to databases through ODBC or the native database driver. It is a powerful and widely-used database platform that is designed to manage and store large amounts of structured data. SQL Server supports a wide range of features and capabilities, including support for high availability, scalability, and security. It also includes tools for data warehousing, business intelligence, and analytics, making it a popular choice for organizations that need to manage and analyze large amounts of data.

Oracle: Connects to databases through the native database driver. (most probably we will use Oracle server connection as it is being implemented in the lab as well)

Oracle supports a wide range of features and capabilities, including support for high availability, scalability, and security. It also includes tools for data warehousing, business intelligence, and analytics, making it a popular choice for organizations that need to manage and analyze large amounts of data.

Oracle can be accessed and managed through various tools and technologies, including Oracle SQL Developer, Oracle Data Integrator, and Oracle Enterprise Manager. It can also be accessed through various programming languages and technologies, including Java,.NET, Python, and others, using database connectivity methods such as ODBC, JDBC, ADO.NET, and OLE DB.

IBM DB2: It connects to databases through the native database driver.

DB2 can be accessed and managed through various tools and technologies, including IBM Data Studio, IBM DB2 Control Center, and IBM Data Server Manager. It can also be accessed through various programming languages and technologies, including Java,.NET, Python, and others, using database connectivity methods such as ODBC, JDBC, ADO.NET, and OLE DB.

LIST OF MODULES AND FUNCTIONALITIES:

1)Module: Doctor Management System

Functionalities:

- Doctor Scheduling: The system should allow scheduling appointments for patients with doctors. It should also provide an easy way to manage doctor schedules, including time off, holidays, and availability.
- Patient Record Management: The system should be able to maintain detailed records of patients, including medical history, diagnosis, and treatment plan. It should also allow for easy access and updating of patient records.
- Billing and Payment: The system should be able to handle billing and payment processes, including insurance claims, co-payments, and billing statements.
- Prescription Management: The system should provide a way to manage and track prescription orders, including tracking medication orders, dosage information, and refills.
- Communication: The system should enable easy communication between doctors, nurses, and other healthcare providers. It should also allow for secure messaging and file sharing.
- Reporting and Analytics: The system should be able to generate reports and analytics on various aspects of doctor management, such as patient volume, appointment scheduling, and financial performance.

2)Module: Patient Management System

Functionalities:

- Electronic Medical Records (EMR): A central repository for storing patient medical records, including demographics, medical history, allergies, medications, test results, and imaging reports.
- Appointment scheduling: A module that enables patients to schedule appointments online, or for healthcare providers to schedule appointments for their patients.
- Billing and invoicing: A module that generates invoices and tracks payments for healthcare services rendered.
- Prescription management: A module that manages medication orders, renewals, and refills for patients.
- Clinical decision support: A module that provides healthcare providers with evidence-based guidelines and alerts to help them make informed decisions about patient care.
- Reporting and analytics: A module that generates reports on patient outcomes, quality of care, and operational efficiency.
- Patient portal: A secure online platform that allows patients to access their medical records, schedule appointments, request prescription refills, and communicate with their healthcare providers.
- Health information exchange (HIE): A system that enables the secure exchange of patient medical records between different healthcare providers.

3)Module: Online Management System

Functionalities:

- Appointment scheduling: Patients can view available appointment slots and book an appointment online. Healthcare providers can also schedule appointments on behalf of their patients.
- Calendar management: The system should include a calendar view for healthcare providers to see their schedule, appointments, and availability at a glance.
- Automated reminders: The system should automatically send appointment reminders to patients via email or SMS.
- Waiting list management: If there are no available slots for a specific appointment, the system can add the patient to a waiting list and notify them when an appointment becomes available.
- Cancellation and rescheduling: Patients can easily cancel or reschedule their appointments online, without needing to contact the healthcare provider directly.
- Integration with other systems: The system should integrate with the healthcare provider's electronic medical record system to ensure that patient records are up to date.
- Patient registration and verification: The system should allow new patients to register online and verify their identity to ensure the accuracy of their records.

COMMENTS GIVEN BY FACULTY IN SECOND REVIEW

1)To show the relationship between different entities.

2) Whether the software used for the development is open source?

3) Whether Microsoft SQL Server or Oracle will be used for the project?

4) To make changes in the backend design.

5) To show how data will be secure in the application.

ANSWERS GIVEN FOR QUESTIONS IN FIRST REVIEW:

Comment: The abstract should be in points.

Answer: Changes in the abstract were made.

Comment: Define the scope of the project.

Answer: The goals and boundaries of the project were set.

Comment: Decide for what department you will be handling the whole department, and based on that, create your own system architecture.

Answer: The system's architecture was rebuilt in accordance with its needs.

Comment: ER diagram should be redesigned with real time use cases.

Answer: The ER diagram was designed with real time use cases, and the relationship was also established between different entities.

REFERENCES:

- https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=56446b88f60668a39dd19da3c47b760b8eb8dfe2 (for architecture diagram)
- https://www.edrawsoft.com/article/er-diagrams-for-hospital-management-system.html (for ER diagram)
- https://www.indiatoday.in/india/story/tamil-nadu-hospital-patients-data-sold-online-dark-web-aiims-server-issue-2304872-2022-12-03 (Data of 1.5L patients sold online)
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC80879/ (obtaining proper medical history of the patient to make informed clinical decisions)
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2779965/