HOSPITAL MANAGEMENT

Patrick George(111601015)
Roshin Raphel(111601018)
Vismay Raj(111601030)

This project aims to emulate the database of a hospital.
A hospital needs a lot of information to be stored for proper functioning and servicing of a patient.

We will keep records of namely:

- **Doctors**: The hospital will have names, ids, specialization, experience, the lab the doctor works in, work timing, salary etc.

- **Patients:** The hospital will have the records of the patients who have visited the hospital and will contain patient name, id, medical history, doctor treated, drugs prescribed, tests performed etc.

- **Drug store:** It will have the drug's name, the related disease, cost etc.

- **Labs:** It has the information of the tests that a lab can conduct, timings, doctors present etc.

User level views:

➔ **Patient:**
- ◆ **Online Booking:** Patient needs to see doctor availability, time slots and will be able to book an appointment.
- ◆ **Lab Report:** Patient needs to look at his/her lab reports.

➔ **Doctor:**

- ◆ **Drug availability:** Doctor needs to see the availability of drugs.
- ◆ **Patient Records and appointments:** Doctor needs to see the appointments and medical history of patient.

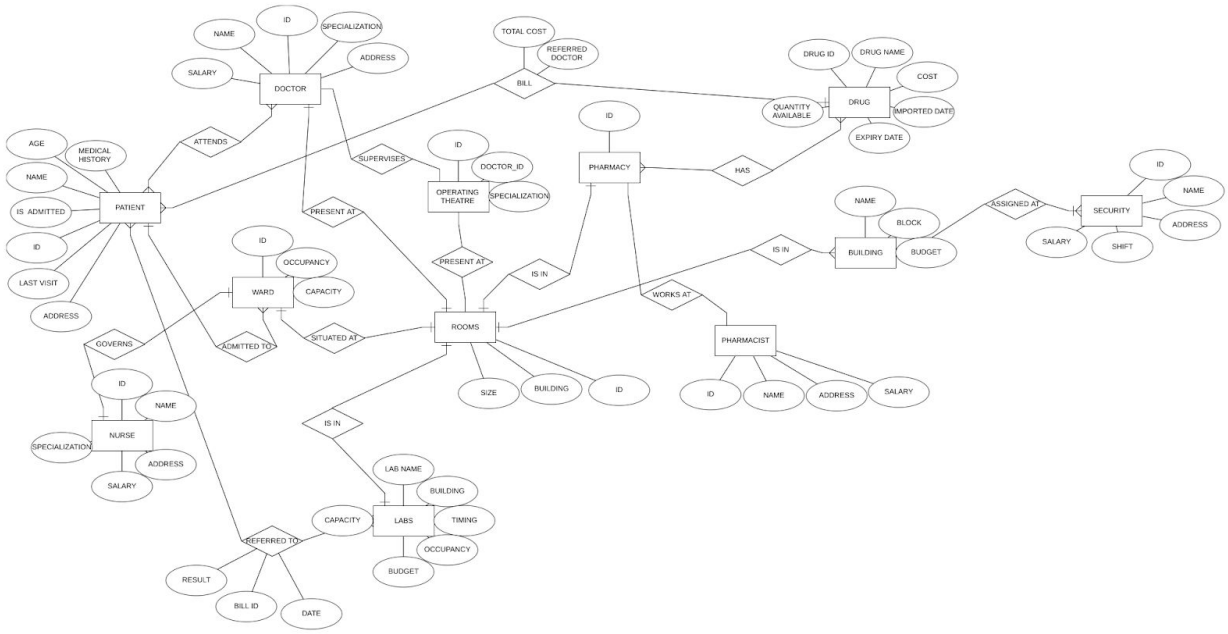
➔ **Nurse:**

- ◆ Should be able to see ward details, patient details, supervising doctor details.

➔ **Pharmacist:** Pharmacist needs to see the current drug stock, expiry dates etc.

➔ **Labs:** Lab operators need to see the patients' lab appointments, lab reports.

E-R diagram:



VIEWS, ROLES AND TRIGGERS:

VIEWS:

The views of this database are namely:

- **Drug list:** Contains information regarding the availability of drugs derived from the drug table.The attributes derived are drug ID, drug name, cost, imported date, expiry date, quantity available.
- **Patient list:** Contains the information regarding the patients who are currently admitted.The attributes would include patient ID, name, age and medical history.
- **Ward info:** The information regarding a ward are stored like ID, occupancy and capacity.

```
mysql> create view Ward_info as select * from Ward;
Query OK, 0 rows affected (0.01 sec)

mysql> create view Drug_list as select * from Drug;
Query OK, 0 rows affected (0.00 sec)

mysql> create view Patient_list as select ID, Name, Age, Medical_history from Patient;

Query OK, 0 rows affected (0.00 sec)
```

ROLES:

Roles are associated with the employees of the hospital. The roles are namely:

- **Doctor:** The doctor must be able to see list of patients who have currently made an appointment in the hospital, the availability of drugs and the ward details.The doctor has grant permission.

```
MariaDB [hospital]> grant all on TABLE Patient to doctor_role;
Query OK, 0 rows affected (0.01 sec)

MariaDB [hospital]> grant all on TABLE Ward to doctor_role;
Query OK, 0 rows affected (0.00 sec)

MariaDB [hospital]> grant all on TABLE Drug to doctor_role;
Query OK, 0 rows affected (0.00 sec)
```

- **Nurse:** The nurse has the permission to see the wards information, the list of patients who have made an appointment and the availability of drugs. The nurse has permission to update and add into the ward table.

```
MariaDB [hospital]> grant update,select,insert on TABLE Patient to nurse_role
Query OK, 0 rows affected (0.00 sec)

MariaDB [hospital]> grant update,select,insert on TABLE Drug to nurse_role;
Query OK, 0 rows affected (0.00 sec)
```

- **Pharmacist:** The pharmacist has information of the drugs and has the information to edit the drug table.

```
MariaDB [hospital]> create role pharmacist_role;
Query OK, 0 rows affected (0.00 sec)

MariaDB [hospital]> grant select,update,insert on table Drug to pharmacist_role;
Query OK, 0 rows affected (0.00 sec)
```

- **Security:** The security can see the shift details of the other security guards.

```
MariaDB [hospital]> create role security_role;
Query OK, 0 rows affected (0.00 sec)

MariaDB [hospital]> grant select on table Security to security_role;
Query OK, 0 rows affected (0.00 sec)
```

TRIGGERS:

- When a pharmacist is removed from the Pharmacist table, the corresponding Pharmacy's pharmacist count is reduced.
- When a new drug is added to the drug table, a new entry is added to the relation table Drug_in_pharmacy.
- When a patient is removed, the corresponding ward occupancy should be reduced.

```
MariaDB [hospital]> DELIMITER $$
MariaDB [hospital]> CREATE TRIGGER pat_remove_trigger AFTER DELETE ON Patient
    -> FOR EACH ROW
    -> BEGIN
    -> UPDATE Ward SET Occupancy=Occupancy-1 where ID = (select Ward_ID from Pat_admit_ward where Patient_ID=OLD.ID);
    -> DELETE FROM Pat_admit_ward where Patient_ID=OLD.ID;
    -> END$$
Query OK, 0 rows affected (0.23 sec)
```

GRAPHICAL USER INTERFACE:

An interactive form for adding a new patient:

Displaying patient records:

ID	Name	Age	LastVisit	Address	MedicalHistory
1	Ringer	24		223 E. Concord	First time visitor here
2	Arjun	21	2018-12-05	12529 State Road 535	Vistied for headache
3	Gokul Shaji	22	2019-01-01	Alappuzha	Previously visited for severe back pain, bedridden multiple times
4	Bhagya	26		2855 South Orange Ave	First time visitor here
5	Vivek	21		7822 W. Sand Lake Rd	First time visitor here

BCNF Form:

A relation schema R is in BCNF with respect to a set F of functional dependencies if for all functional dependencies in F+ of the form
$\alpha \rightarrow \beta$
where $\alpha \subseteq R$ and $\beta \subseteq R$, at least one of the following holds:
• $\alpha \rightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
• α is a superkey for R

- In our project, every table is made in such a way that given the primary key, every other attribute for that row can be derived. So, that functional dependency is in BCNF form.
- For any other pair of columns, our tables were made such that there is no dependence among them. That is given any attribute other than the primary key, we cannot determine another attribute.
- This means that all the dependencies of our tables are in BCNF form.

CONTRIBUTION:
Patrick George: Making ERD, tables, report.
Vismay Raj: Making ERD, tables, GUI.
Roshin Raphel: Making tables, adding values values to tables.