**American University of Beirut**

**Department of Computer Science**

**CMPS 277 Database Systems – Spring 21-22**

**Hospital Database Management System – Proposal**

**Team:** Runtime Terror

**Members:** Malak Mehrez, Samer AbdulBaki, Sergio Khalil, Tamer AlBouz

**Project Description:**

This project aims at building a database system to manage information about the records of the patients and doctors, rooms, medications, payments, etc.

Entities:

1. Staff: staffID, supervisorID, jobType
   1. Nurse: nurseID
   2. Janitor: janitorID
   3. Cashier: cashierID
   4. Doctor: DoctorID, specialty
2. Room: roomID, roomType
3. Bill: billID, billStatus, field,charge
4. Patient: patientID, diagnosis
5. Medicine: medicineID, medicineName, price, quantity
6. patientRecord: recordID, firstName, lastName, gender, address,phoneNumber,admissionDate, dischargeDate
7. staffRecord: : recordID, firstName, lastName, gender, address,phoneNumber,startDate, endDate

**Relationships:**

The following relationships will hold between our entities

1. Each Doctor can have exactly one record that is updated in the case of discharge and re-admission
2. Each Patient can have exactly one record that is updated in the case of multiple admissions to the hospital
3. Each doctor can treat multiple patients and each patient can be treated by multiple doctors
4. Each nurse is responsible for multiple rooms and each room is the responsibility of exactly one nurse
5. Upon entrance, each patient is assigned to specific room type, and a record for his/her info like bill is issued.
6. Upon the completion of diagnosis, the doctor responsible for the patient provide him/her with the appropriate medicine.

**Constraints:**

1. a patient cannot be assigned to n-doctors with the same specialty
2. a patient cannot check-out unless he/she checked-in for at least 24hrs ago
3. a patient cannot check-out unless the status of his/her bill is "fully paid"
4. a doctor cannot prescribe a medication that does not exist in the hospital
5. a doctor cannot prescribe a medication unless the diagnosis of patient is completed
6. Exactly one patient cannot be assigned to two different rooms

**Sample Queries:**

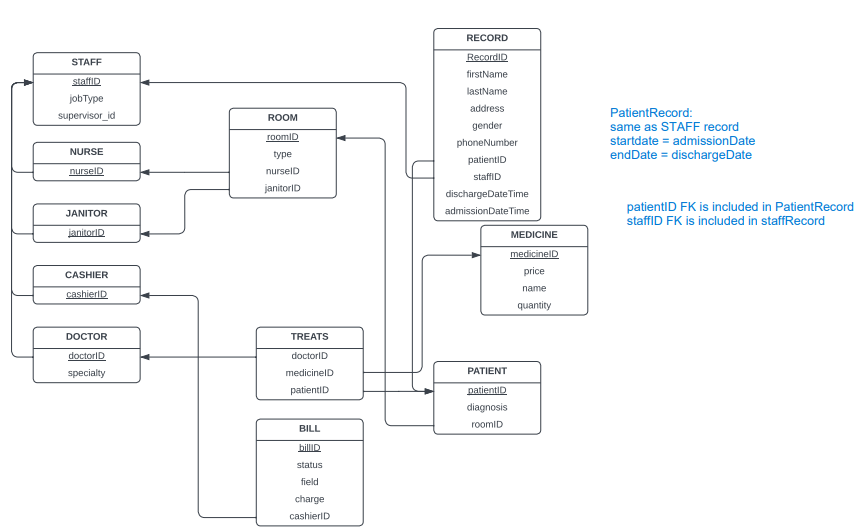
1. List all the patients that checked-in on specific date
2. List all the doctors of a specific specialty
3. List all the patients of a specific doctor
4. list all the patients that a nurse is responsible for
5. list all the medications that a patient took
6. create a new record for a patient upon check-in

**Enhanced Entity Relationship Model:**

Diagram

Description automatically generated

**Relational Model:**



**Views:**

* The staffInfo view combines the STAFF table that includes job type and supervisor with the personal information from staffRecord Table
* The doctorInfo view selects only the doctors among the staff
* The patientInfo combines patient that includes diagnosis and room that the patient is assigned to with the personal information of the patient in patientRecord

**GUI Description:**

LogIn.java is the entry point of the application. As such, you will first be asked to "log in", i.e. entering the credentials to your local database. On successful entry, you will be redirected to a page containing buttons that navigate you to different CRUD operations, one for each of insert, delete, update, and retrieve (query).

In the insertion screen, you can add new staff (nurses, janitors, cashiers, or doctors), rooms, bills, medications, and records (personal information about the staff or patients such as phone number and address).

The deletions screen is split into staff deletions and other deletions (for rooms, bills, and medications).

The updates screen is similar to the insertions screen but updates existing records instead of adding new ones.

Finally, there are 6 different queries you can perform: viewing a patient's existence and their information, viewing doctors with a specific specialty, viewing a certain patient's medications, viewing a certain doctor's patients, viewing a certain nurse's patient, and viewing patients checked in on a specific day.

If you perform an operation (insertion, deletion, update, or query/retrieval) successfully, a message will pop up saying so, and, if performed unsuccessfully, a different message will appear. Finally, an error message will also pop up upon failure to log in (incorrect credentials for example).