**Abstract**

Health care system in any country is pivotal to the country’s populace. This project involves the automation of all the activities and systems present in a hospital in Algeria. Maintaining the records of patient history is a cumbersome process and the presence of ambiguity and redundancy in the data further aggravates the entire process. Hence, automating the entire system makes data maintenance a less difficult job. This will be an appropriate solution to the problems currently faced by the hospital, which manually enters the patient details. Looking to widen its scope in areas like inclusion of transfer patients and visiting doctors from various other hospitals, it cannot thrive with a manual data storage practice in place.

The main objective is to develop a database that would maintain records of various components in the hospital. It is only logical if system is defined in terms of relationships rather than a mere representation of the components. The various components would include Providers, nurses, Patients, Rooms, IT management, Visitors, Finance, and Pharmacy among others .Each component has a distinct functionality and is inter-related. The categories under a Provider would include an In-house doctor, a Primary Care Physician, and a visiting doctor, who also would be accessing the patient records, history of illness and other relevant details. So, having a system that supports concurrent updating and minimizes duplicity is absolutely essential. Moreover, since, the hospital has planned to expand its coverage to target areas that have been recently infected by a flu .To accommodate the throngs of infected patients, the hospital has to have a robust system. The categories under patient include patients who have undergone surgery, transfer patients and patients who have been quarantined. There are different types of rooms which include general rooms, private rooms, and quarantined areas. IT management sector, a specialty addition to the hospital’s workforce, should ensure that the system is secure and is functioning accurately. Finally, we intend to improve upon the proposed model as Healthcare system is more evolutionary than revolutionary.

**Problem Statement:**

The DNEXUS General Hospital was established a few years ago. Being in the third world, the number of patients visiting the clinic kept increasing by the day and neither the staff nor the doctors were equipped to handle the surge in patients. They were relying on a paper-based system to handle and maintain the patient details and records. Thus they were not able to back track the patient medical history. Lot of issues such as missing data and data replacement popped in, which in turn led to lot of complications. The area being beset by a flu also poses a stiff challenge to the hospital.

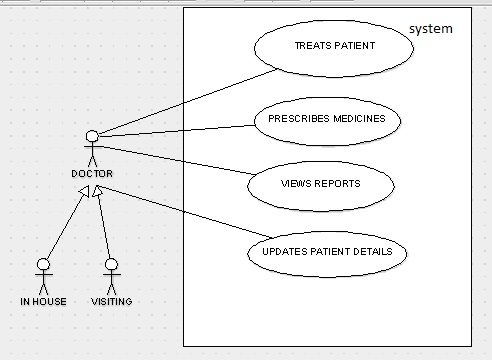
The hospital management is determined to revamp its process of data storage, inventory and put in a system that is capable and robust. To make it robust, we have suggested to the hospital to include an IT section that would handle any system related issues and ensure Business Continuity during emergencies.

To sort out this issue and to amplify the performance of the hospital we are implementing this Health care Management System that aims to:

1. Build a robust system that can store and retrieve data effectively and conveniently.
2. Create “constraints” to minimize data duplications.
3. Provide Improved data security,
4. Incorporate a designated IT section to enforce Business Continuity.
5. Put in place an Emergency Department to attend to emergencies.

So here we are aiming to automate the system and thereby aiding the expansion of the hospital. This automation system involves the usage of a database management system. The following section goes on to explain the roles and responsibilities of each actor in the hospital.

**Use Cases:**



Fig, 1.1 Provider use case diagram

Here the provider is the actor and the various roles are

* Treating patients
* Prescribing medicines
* Views Reports
* Updating patient details

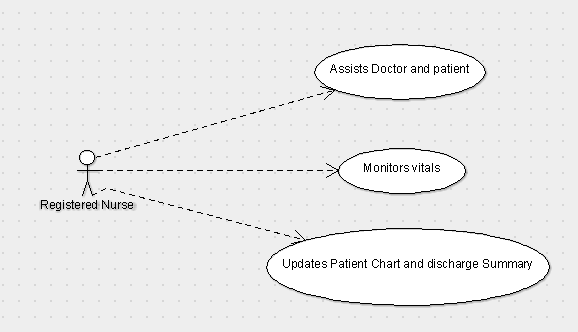


Fig.1.2. Nurse Use case diagram

Here Nurse is the actor and the various roles are:

* Assists doctor and patient
* Monitors vitals
* Updates Patient chart and discharge summary

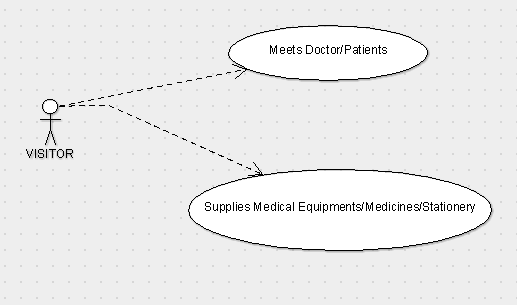


Fig.1.3 Visitor Use case diagram

Here the visitor is the actor and his roles include

* Meets the doctor/Patients
* Supplies medical equipment

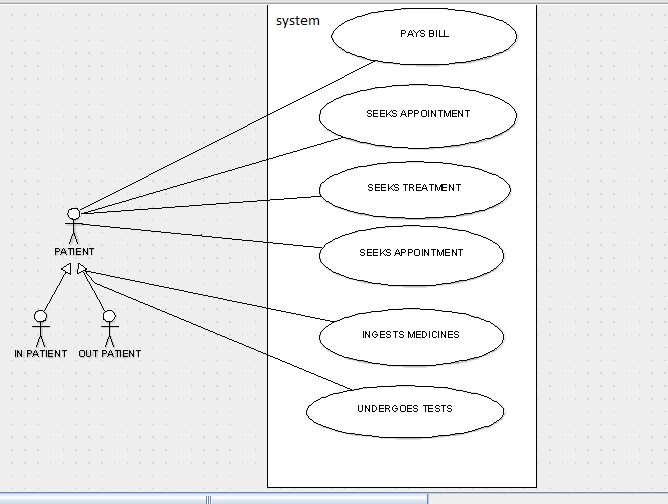


Fig.1.4 Patient Use case diagram

Here the patient is the actor and his roles include

* Schedules appointments
* Pays bills
* Buys medicines
* Undergoes Tests

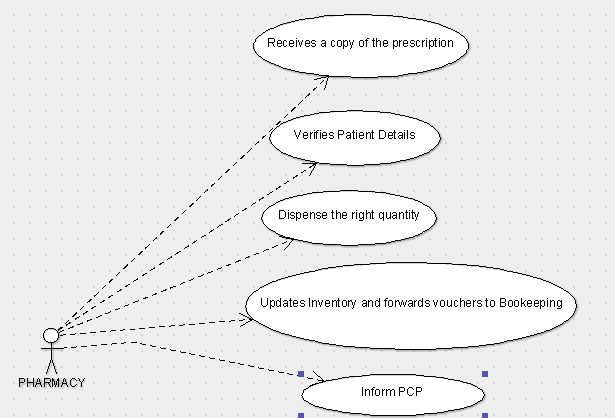


Fig.1.5 Pharmacy Use case diagram

Pharmacy is an actor here and various roles of the pharmacy would include

* Receives a copy of prescription
* Verifies patient details
* Dispensaries
* Maintains inventory
* Informs PCP

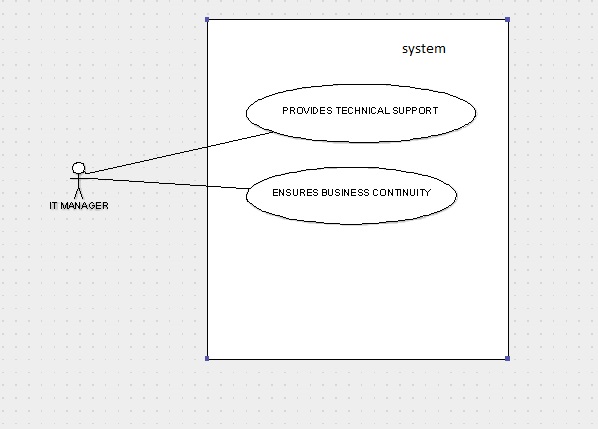


Fig.1.6 IT Team Use case diagram

Here the IT team is an actor and the roles of this team would include

* Ensure business continuity plan
* Provide technical support

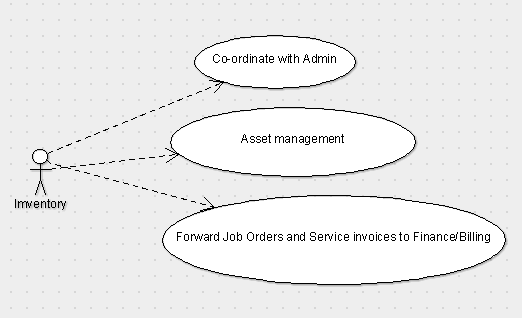


Fig.1.8 Inventory Use case diagram

Here the Inventory is an actor and the roles include

* Co-ordination with admin
* Asset Management
* Forward Job Orders and Service Invoices to Finance/Billing

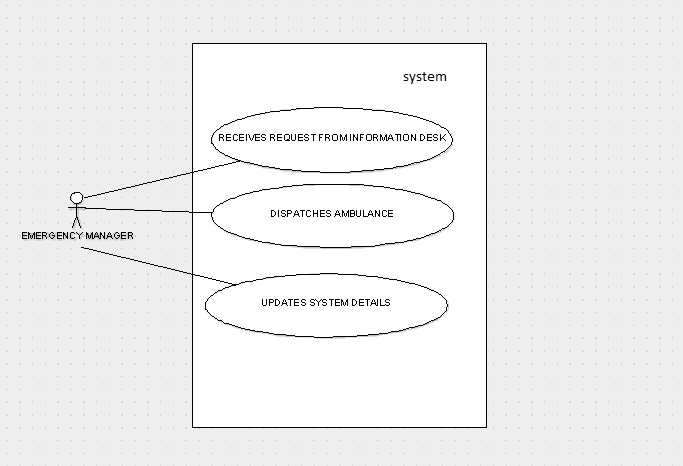


Fig.1.9 Emergency Response Team Use case diagram

Here the Emergency Response Team is an actor and the roles include

* Receives Request from Information Desk
* Dispatch Ambulance
* Update status for other actors

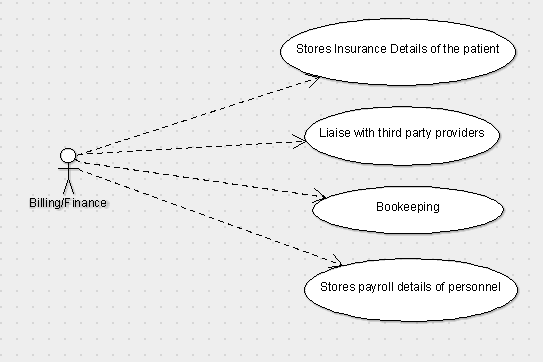


Fig.1.10 Billing/Finance Use case diagram

Here the Billing/Finance is an actor and the roles include

* Stores Insurance Detail of patient
* Liaise with third party providers
* Bookkeeping
* Store payroll details

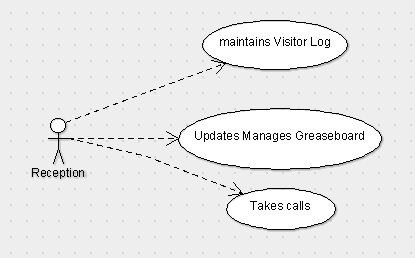
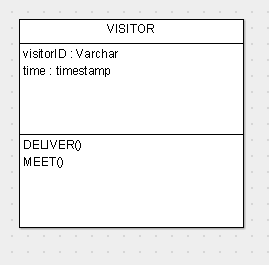
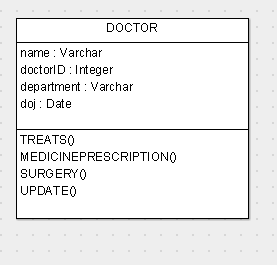
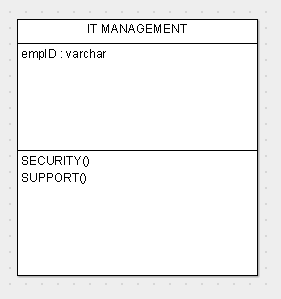


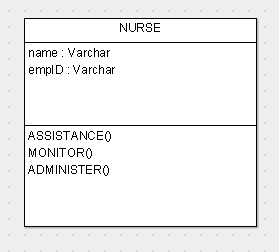
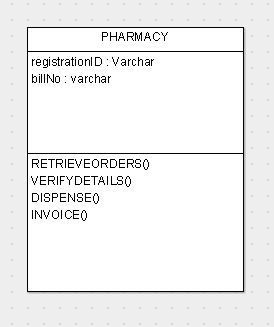
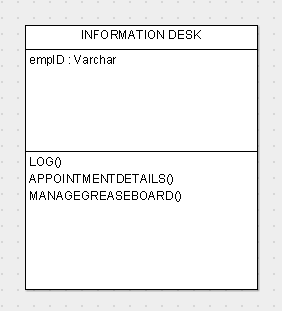
Fig.1.11 Reception Use case diagram

Here the reception is an actor and the roles include

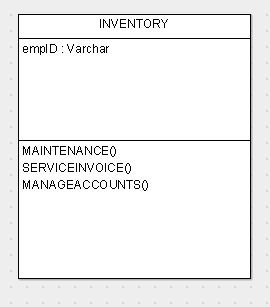
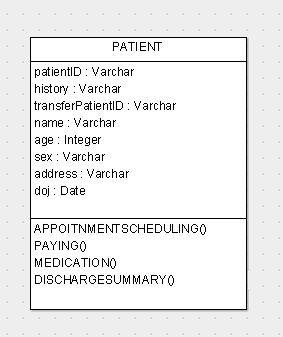
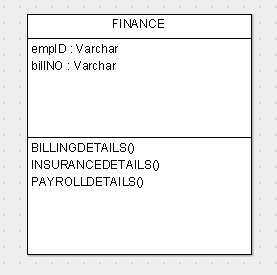
* Maintenance of Visitor Log
* Updates/manages Grease board
* Takes calls

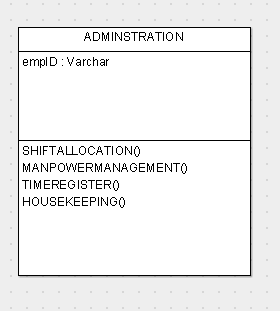
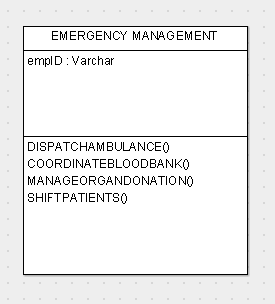
**Class Diagram:**

 1…\*  

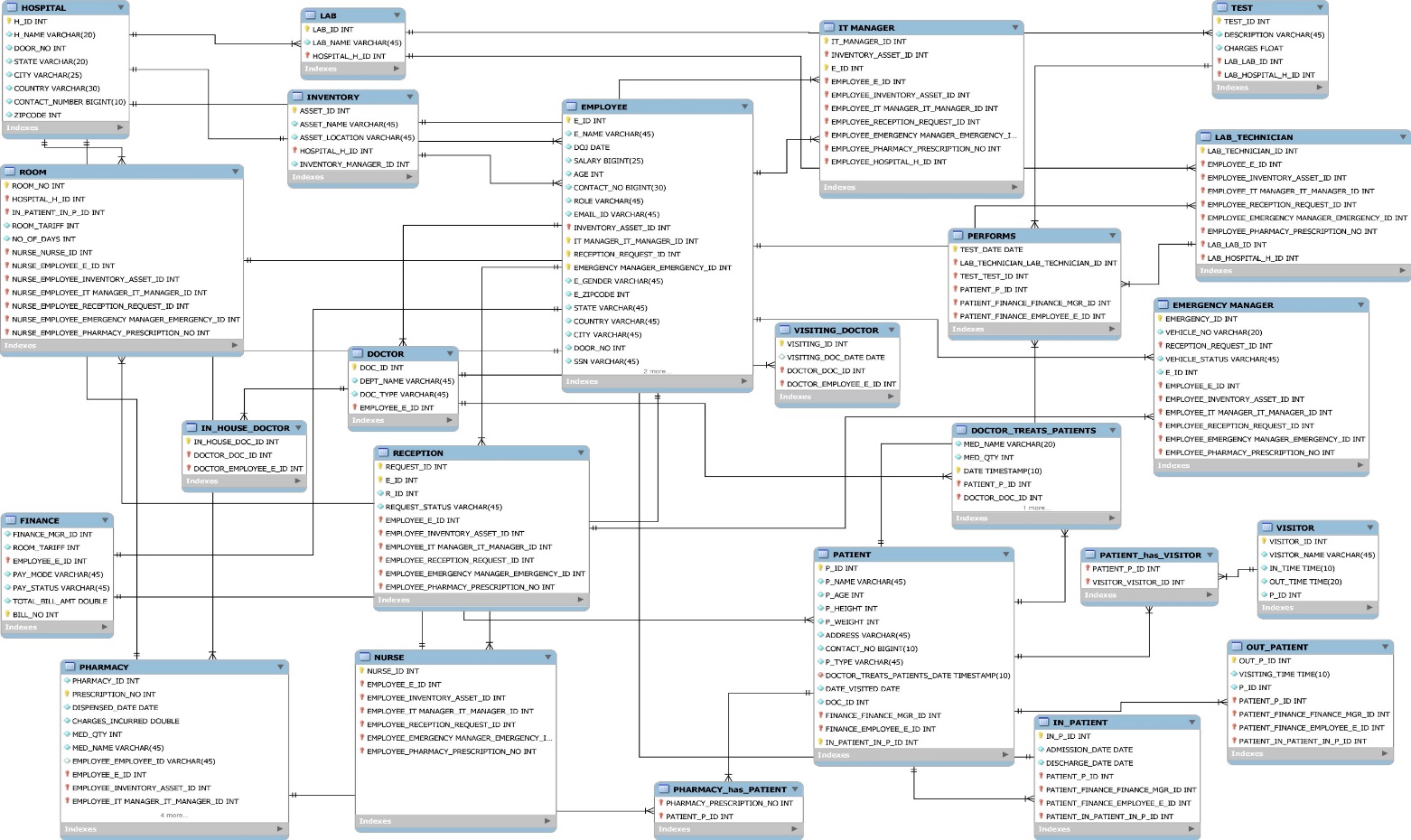
  

1…\*

**EER Diagram:**



**User Privileges:**

We have created the below users in our model and their corresponding privileges have been mentioned in the below tabular column:

Admin

Doctor

Emergency Manager

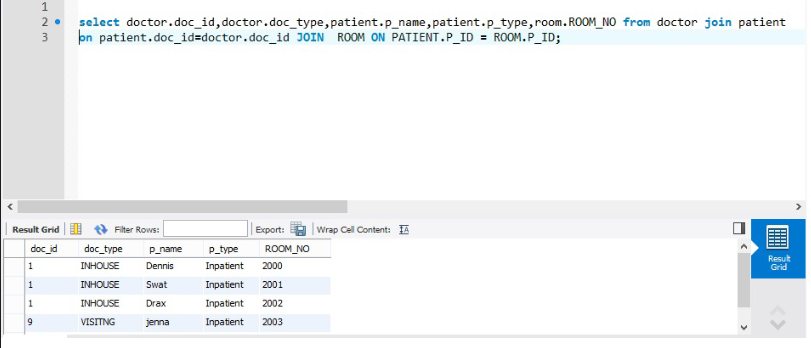
Finance Manager

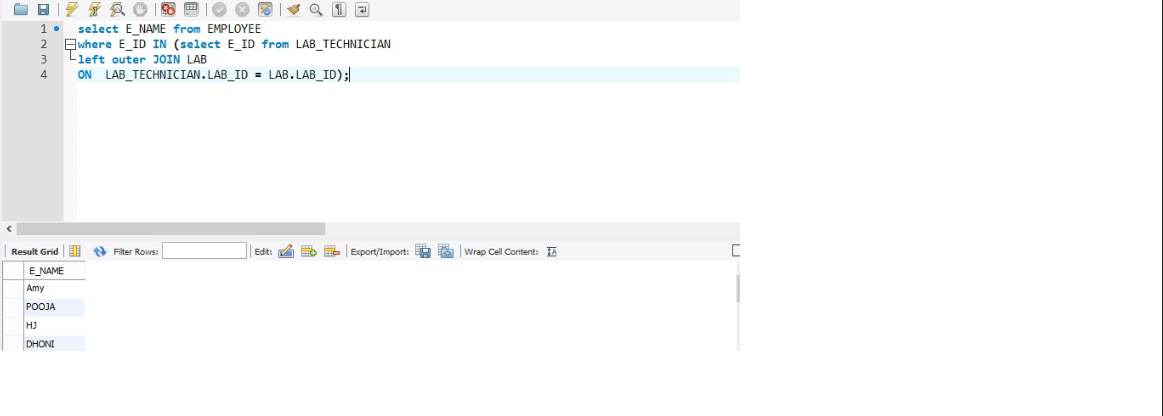
|  |  |
| --- | --- |
| **User** | **Privileges** |
| ADMIN | Has all privileges over the database including GRANT option. |
| DOCTOR | Has all privileges over Patients Table and SELECT over DOCTOR Table |
| EMERGENCY MANAGER | Has Update Privileges over Emergency\_Mgr Table |
| RECEPTIONIST | Has SELECT, UPDATE, INSERT privileges on RECEPTIONIST Table |
| FINANCE MANAGER | Has ALL Privileges over FINANCE, ROOM, and TEST tables |

**JOINS AND SUBQUERIES:**

In our system, there were scenarios to fetch details from multiple tables and to accomplish them, we have used joins and sub queries. Few of the scenarios are mentioned below:

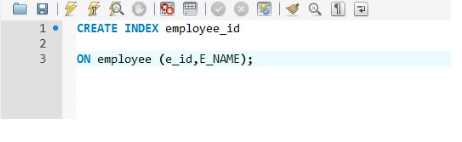
* To display hospital details
* To view the status of emergency requests and their owners.
* To Track which Doctor is treating which Patient in which room.
* Fetch the Employee Name, who is in-charge of the LAB.



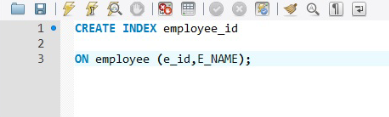


**INDEX:**

Indexes are used for faster retrieval of data from any table. In our model, there are large number of employees, and hence in-order to retrieve the employee name, we have used index.



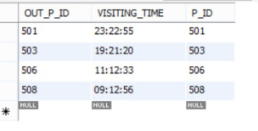
We have also created index for retrieving patient ID from patient table.

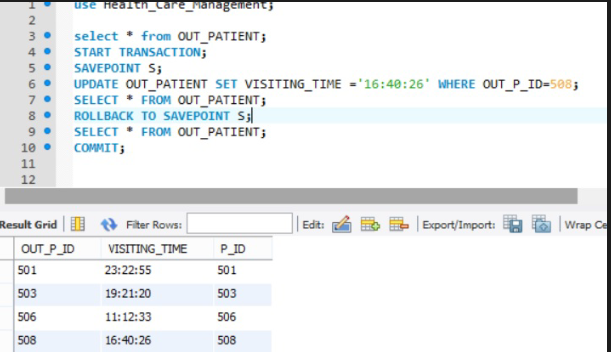


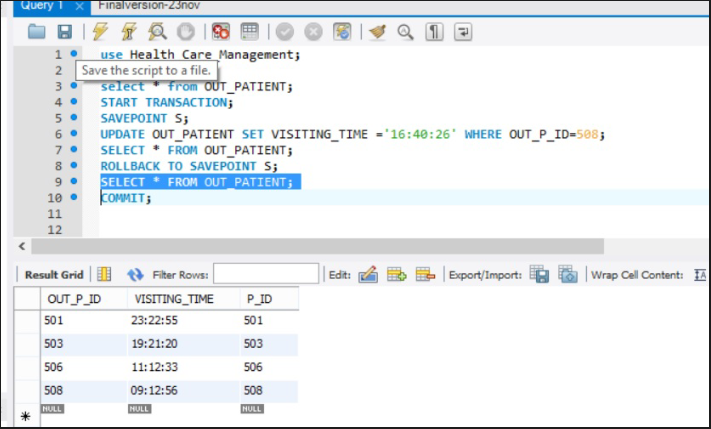
**TRANSACTIONS:**

We have used transaction in the OUTPATIENT table to update the appointment timings of the patient. We have a save point, before changing his appointment.

If the patient wishes to revert to the old timing then a rollback to the save point action is performed.

****

****

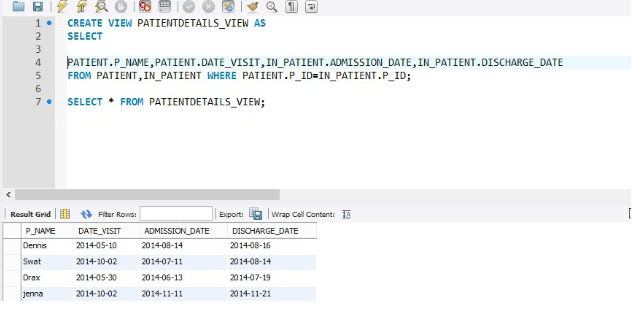
****

**VIEWS:**

We have used views in our system for the below two scenarios:

* To view Inpatient details.
* To view medicine details.

Below is one of the scenarios and its corresponding output:



**STORED PROCEDURES:**

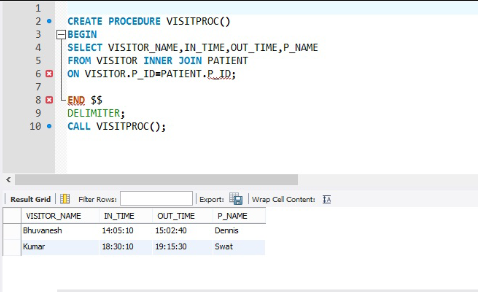
Certain queries need to be called repeatedly and in order to accomplish them, we have used stored procedures. The scenarios are:

* To display visitor details for patients
* To calculate the total bill amount in finance table.

The total bill amount needs to be calculated from different tables.

For an inpatient, the total charges should be calculated from test table, which includes test charges and from room table, which includes room charges for number of days, a particular patient stays. The total needs to be updated in finance table. Whereas, for an outpatient, the total needs to be updated by considering only the test charges from test table.

In order to accomplish the above criteria, we have used cursors and exception handling in our procedure.



**TRIGGERS:**

In order to execute the procedure for calculating total bill amount, we have used six triggers:

1. Insert and update triggers on Outpatient table.
2. Insert and update triggers on Room table
3. Insert and update triggers on Test table.

We have also used triggers to change the request status in Receptionist table when a status is updated in Emergency manager table.

Please find the sample code and output for the triggers used by us:

