DESIGN OF A HOSPITAL-BASED DATABASE SYSTEM

Case study

Under the Guidance of

Mr. Ram Murti Rawat

Submitted By:

PRAVESH(2K19/EE/193)

YATHARTH SHARMA(2K19/AE/073)

Design of a Hospital-Based Database System
(A Case Study of DTU Health Care)

INTRODUCTION

Delhi Technological University Health Care, abbreviated as DTU HC, is a hospital situated inside DTU campus. There are doctors, patients, and admin which are considered as entities in the designed system. It becomes a tough, tedious and comparatively slow process to store the information manually. But having all the necessary information stored in one database, it not only helps in orderly maintain but very speed retrieval of data

OBJECTIVES

- The system automates the processes that were previously done manually for managing hospital activities.
    - Doctors can retrieve and manage their patients' treatment records easily in no time.
    - By the assistance of this technique receptionist can easily see doctors availability and also the resources available and immediately schedule the appointments whenever any patient arrives without going personally to the doctors.
    - The system is convenient and versatile to be used.
    - It saves a lot of time and equally the efforts expended, money and even many resources.

METHODOLOGY OF DATA COLLECTION

- Raw data (also known as primary data) is a term for data collected from a source. Raw data has not been subjected to processing or any other manipulation, and are also referred to as primary data.

- Primary data is a type of information that is obtained directly from first-hand sources by means of surveys, observation or experimentation. It is data that has not been previously published and is derived from a new or original research study and collected at the source such as in marketing.

- Primary data collection are observed and recorded directly from respondents. The information collected is directly related to the specific research problem identified. All the questions that one asks the respondents must be totally unbiased and formulated so that all the different respondents understand it.

EXTRACTION OF INFORMATION FROM HOSPITAL MANAGEMENT (DTU HC)

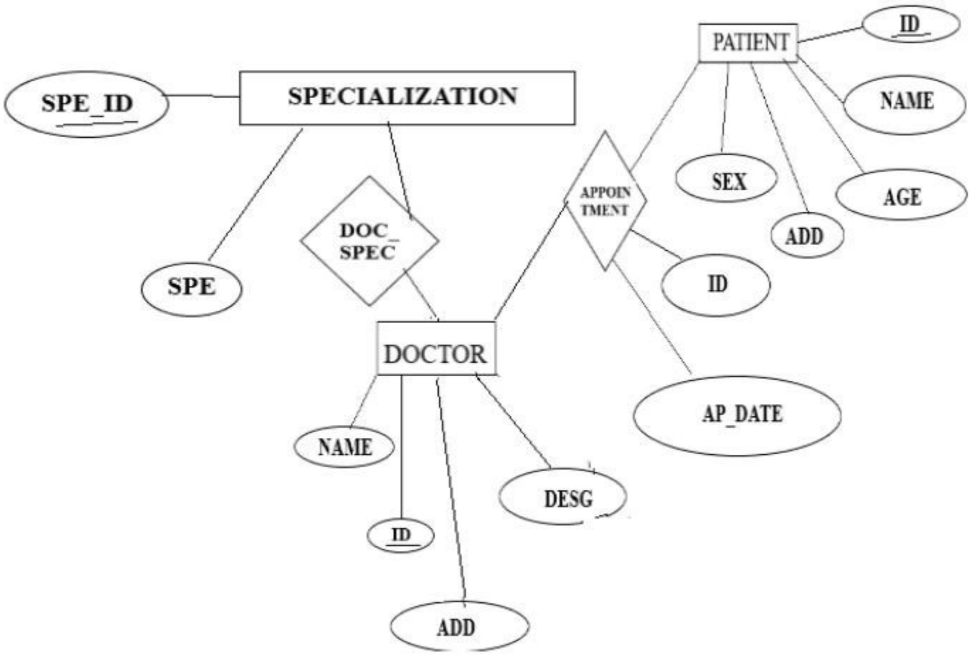
DTU Health Care was visited and information was gathered; demos were seen.
A database system was designed based on a case study of DTU Health Care via Entity Relationship Diagram (ERD), Relational Model , Normalization of tables and Implementation in SQL server. The ERD is outlined below

DATABASE DESIGN

A. Entity Relationship Model (ERM)

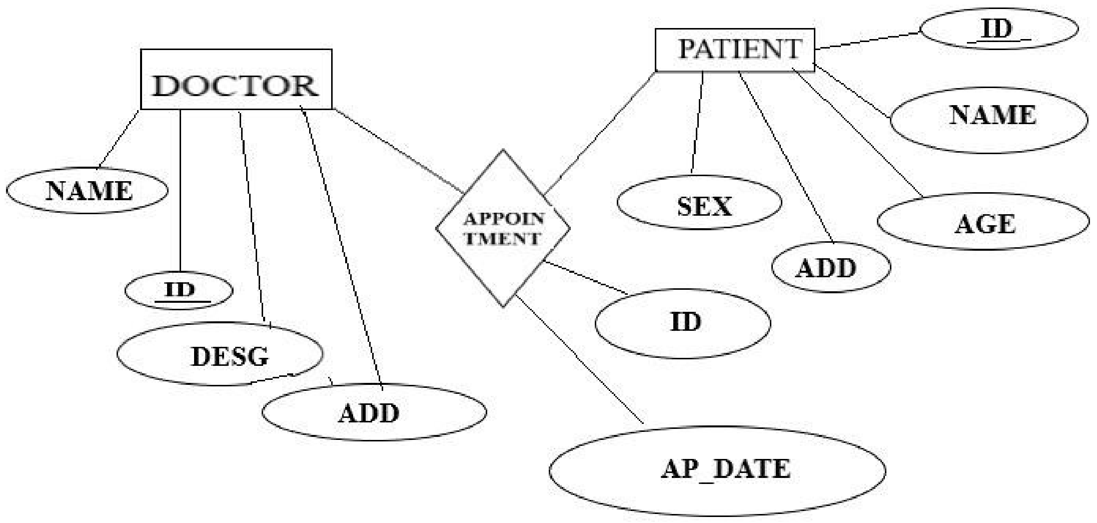
1) **ER Diagram**

In the ER diagram, we can view the entities- Patient, Doctor, Admin, User, Specialization, and Appointment. Among these entities, relationship exist which connect all the entities in the diagram. For example, Admin and Specialization are connected via the relationship Manages. In other words, an admin can manage (add/ delete) specialization. Similarly Doctor and Patient are connected via the relationship Treats. Here, a doctor may treat one or more patients. In a similar way, other entities are connected via relationships in a meaningful way. Cardinality ratios for the entities connected to a relationship are explained in the next section.



2) **Cardinalities**

As can be seen in Figure 2, 1 Doctor can have 1 or more patients. 1 patient may have 1 or more Doctors. So it is a many to many relationship named appointment (in the diamond).



Similarly there can be one to one, one to many or/and many to one relationships.

B. ***Relationship Model***

The entire ER diagram can be converted to a Relational Model (relational tables) as shown below. The attribute(s) of a relation which serve as a primary key of the table are underlined. Those attribute(s) which represent foreign keys are indicated as fk underneath .

i) ADMIN (ID, USERNAME, PASSWORD) ii) DOCTOR (DOC_ID, NAME, ADDR, SPEC_ID) iii) APPOINTMENT (DOC_ID,PAT _ID, AP_ID, AP_DATE)

This is a junction table among Patient and Doctor tables. Primary key of the Patient table goes to Appointment table as foreign key. Primary key of the Doctor table goes to Appointment table as foreign key.App_id is a primary key in the Appointment table. Since the cardinality ratio is many to many the foreign key becomes part of primary key. iv) PATIENT (PAT_ID, NAME, SEX, AGE, ADDr) v) SPECIALIZATION(SPEC_ID, SPE)

This is a junction table between Doctor & Specialization tables.. Primary key of the Patient table goes to Doc_spec table as foreign key. Primary key of the Doctor table goes to Doc_spec table as foreign key.Since the cardinality ratio is many to one, Primary key of doctor table becomes part of primary key.

C. Functional Dependencies and Normalization

1) ***Fulfillment of all normal forms***

Example:

Doctor table :doctors (id, name, address, desg) {id} => {name}

Here id corresponds to serial numbers like 1, 2, 3, 4 etc. Name corresponds to name of the doctor. A Functional Dependency exists from Doctor to id because two different doctors cannot correspond to the same id.

Similarly, the following functional dependencies exist:
{id} => {address}
{id} => {desg}

Considering the above full functional dependencies, we now verify whether the relation fulfills rules of normalization.
The relation is in 1NF because the attributes of the relation do not have sub attributes. The relation is in 2NF because non-primary keys are fully functionally dependent on primary key. The relation is in 3NF because no transitive dependency exists from non-primary key to primary key.
The relation is in BCNF because there is no part of primary key that is fully functionally dependent on non primary key.

Similarly, all other relations of the system follow rules of normalization.

*D. **Implementation in SQL Server***

SQL server is a modern software where we can store huge amount of information via a database. In fact we have implemented our database system in SQL server. In this server we can store data easily, retrieve data speedily and execute the queries conveniently by SQL query language. We can create relations easily by code and also manually (drag and drop method like MS.Access).

FEW SQL QUERIES USED IN DATABASE CREATION

Tables were created as follows:

- `CREATE TABLE doctor(doc_id int , name varchar(20), addr varchar(30),spec_id int);`
- `CREATE TABLE appointment(doc_id int ,pat_id int ,ap_id int ,ap_date varchar(10));`
- `CREATE TABLE patient (pat_id int,name varchar(10),sex varchar(8), age int , addr varchar(30));`
- `CREATE TABLE specialization(spec_id int, spe varchar(20));`

Primary Keys were added in each table as follows:

- `ALTER TABLE doctor ADD PRIMARY KEY (doc_id);`

- ALTER TABLE appointment ADD PRIMARY KEY (doc_id,pat_id);
- ALTER TABLE patient   ADD PRIMARY KEY (pat_id);
- ALTER TABLE specialization ADD PRIMARY KEY (spec_id);

  Values were inserted in each of the table as follows:

- INSERT INTO doctor VALUES (1, 'Aarjav',’Delhi’,1);
- INSERT INTO patient VALUES (1, 'Aditya', 'male',18, 'Balaji Pg');
- INSERT INTO specialization VALUES(1,’General Physician’);

  INSERT INTO appointment VALUES (1,6,1, '27/04/2020');

  In similar fashion mutiple rows were inserted into each table

To add new data(row) im any table for eg, a new patient , queries same as initial insert one should be written. Besides data can also be modified (updated, deleted) depending upon the need.

Eg,

- Suppose a doctor retires from the hospital. To delete his record

  ```
  DELETE FROM doctor
  WHERE doc_id = x;
  ```

- We can also update records of any table UPDATE table_name

  ```
  SET col1=val1, col2=val2……
  WHERE condition;
  ```

- We can even combine two or more tables to get desired results. For that we can use any of the following joins:
- CROSS JOIN
  - INNER JOIN
  - LEFT JOIN
  - RIGHT JOIN
  - FULL OUTER JOIN

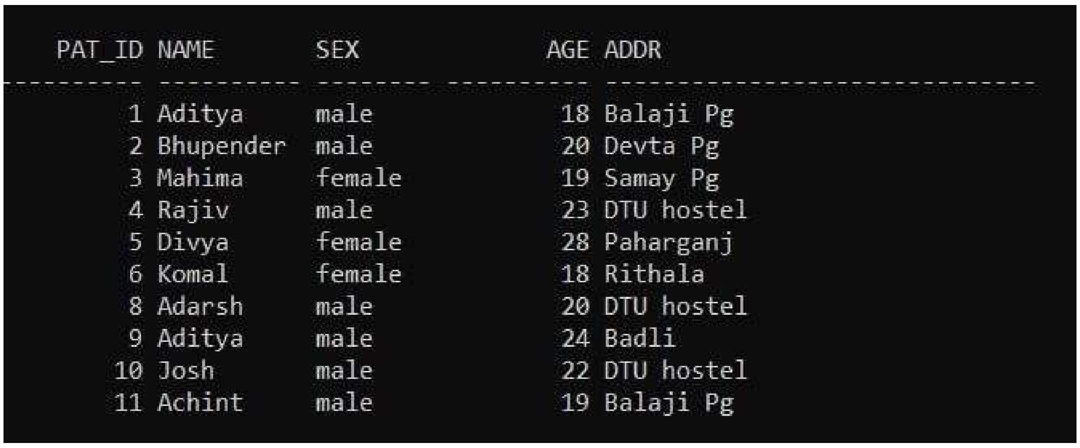
USAGE OF THE SYSTEM

We can easily retrieve various data based on our demands using SQL Queries. Sample data has been entered in the relational tables. Few Queries are enlisted below

Question 1 Give the list of all the patients registered in DTU Health Care Query 1:

SELECT * FROM

patient; Output 1:

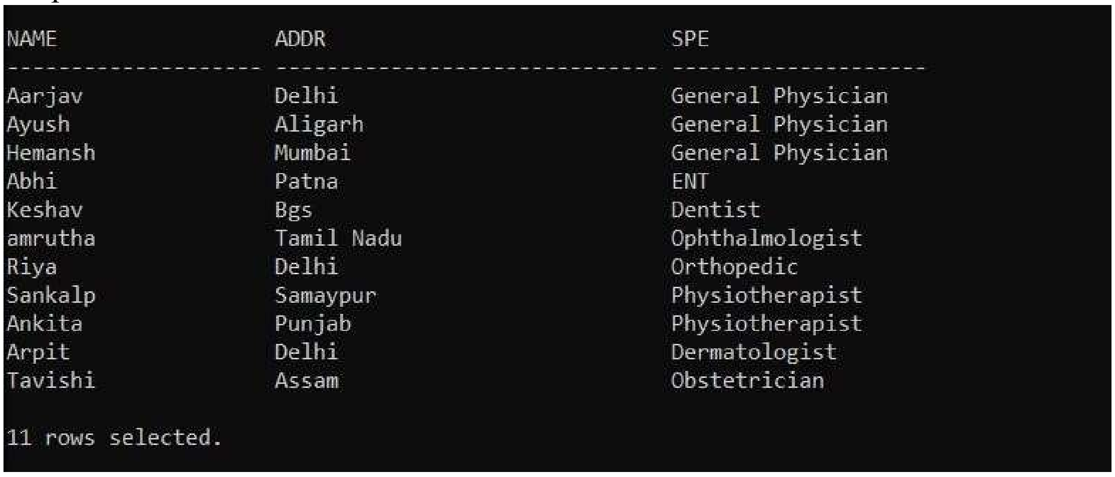


```
   PAT_ID NAME       SEX              AGE ADDR
---------- ---------- -------- ---------- ------------------------------
         1 Aditya     male             18 Balaji Pg
         2 Bhupender  male             20 Devta Pg
         3 Mahima     female           19 Samay Pg
         4 Rajiv      male             23 DTU hostel
         5 Divya      female           28 Paharganj
         6 Komal      female           18 Rithala
         8 Adarsh     male             20 DTU hostel
         9 Aditya     male             24 Badli
        10 Josh       male             22 DTU hostel
        11 Achint     male             19 Balaji Pg
```

Question 2 Give the list of all the doctors currently working in DTU Health Care along with their address and specialization Query 2:

SELECT name, addr, spe
FROM doctor d join specialization s
ON d.spec_id = s.spec_id;

Output 2:



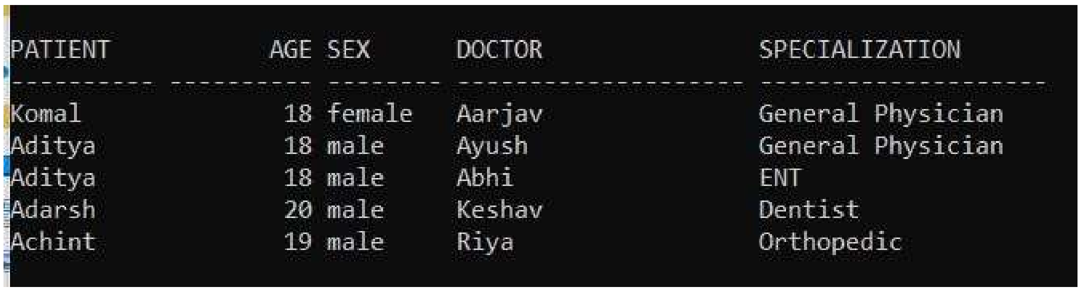
```
NAME                 ADDR                           SPE
-------------------- ------------------------------ --------------------
Aarjav               Delhi                          General Physician
Ayush                Aligarh                        General Physician
Hemansh              Mumbai                         General Physician
Abhi                 Patna                          ENT
Keshav               Bgs                            Dentist
amrutha              Tamil Nadu                     Ophthalmologist
Riya                 Delhi                          Orthopedic
Sankalp              Samaypur                       Physiotherapist
Ankita               Punjab                         Physiotherapist
Arpit                Delhi                          Dermatologist
Tavishi              Assam                          Obstetrician

11 rows selected.
```

Question 3 Give the list of all the patients having appointment on 27/04/2020

Query 3:

SELECT p.name AS patient, age, sex, d.name AS doctor, s.spe AS

Specialization
FROM ((patient p join appointment a ON p.pat_id = a.pat_id) join
doctor d ON a.doc_id=d.doc_id) join specialization s
ON d.spec_id = s.spec_id
WHERE ap_date ='27/04/2020'; Output 3:



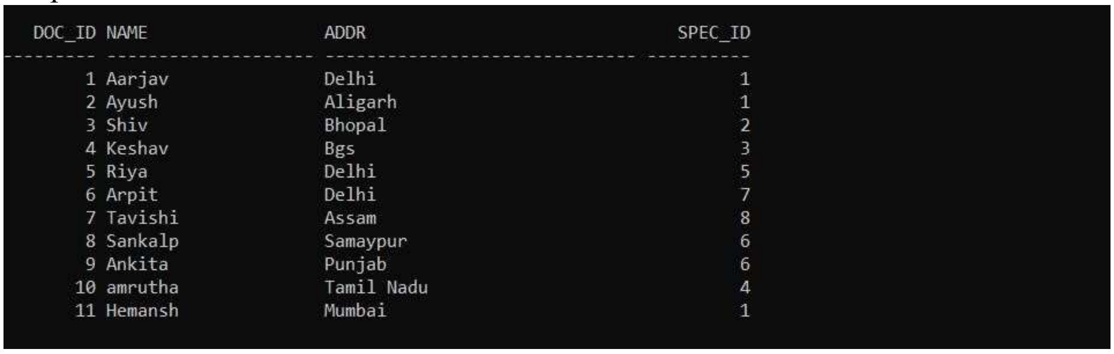
```
PATIENT           AGE SEX      DOCTOR               SPECIALIZATION
---------- ---------- -------- -------------------- --------------------
Komal              18 female   Aarjav               General Physician
Aditya             18 male     Ayush                General Physician
Aditya             18 male     Abhi                 ENT
Adarsh             20 male     Keshav               Dentist
Achint             19 male     Riya                 Orthopedic
```

Question 4 Delete record of Dr. Abhi, ENT, who retired. Add record of new joine Dr. Shiv ,whose address is Bhopal. Query 4:

DELETE from doctor
WHERE doc_id = 3;
-- doc_id of Dr. Abhi is 3 insert into doctor VALUES
(3, 'Shiv','Bhopal',2);

Output 4:

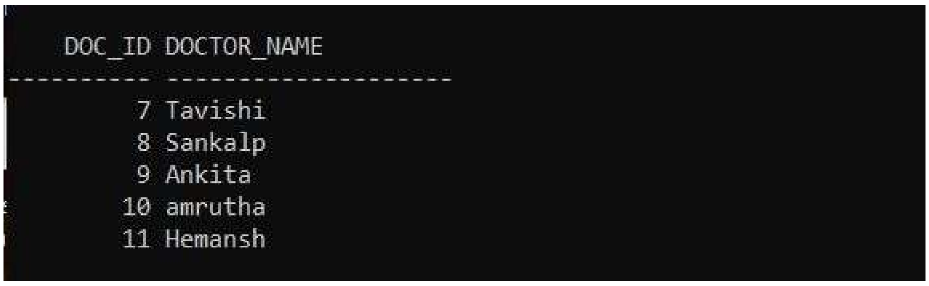
// Update doctor table



```
  DOC_ID NAME                 ADDR                             SPEC_ID
-------- -------------------- ------------------------------ ----------
       1 Aarjav               Delhi                                   1
       2 Ayush                Aligarh                                 1
       3 Shiv                 Bhopal                                  2
       4 Keshav               Bgs                                     3
       5 Riya                 Delhi                                   5
       6 Arpit                Delhi                                   7
       7 Tavishi              Assam                                   8
       8 Sankalp              Samaypur                                6
       9 Ankita               Punjab                                  6
      10 amrutha              Tamil Nadu                              4
      11 Hemansh              Mumbai                                  1
```

Question 5 Give the list of doctors having no appointment on 27/04/2020 Query 5:

SELECT d.doc_id,name AS doctor_name
FROM doctor d left join appointment a
ON d.doc_id=a.doc_id
WHERE ap_id IS NULL AND ap_date='27/04/2020'; Output 5:



```
    DOC_ID DOCTOR_NAME
---------- --------------------
         7 Tavishi
         8 Sankalp
         9 Ankita
        10 amrutha
        11 Hemansh
```

CONCLUSION

Since we are entering details of the patients electronically in the "Hospital Management System", data will be secured. Using this application we can retrieve patient's history with a single click. Thus processing information will be faster. It guarantees accurate maintenance of Patient details. It easily reduces the book keeping task and thus reduces the human effort and increases accuracy. Taking into account all the mentioned details, we can make the conclusion that the hospital management system is the inevitable part of the lifecycle of the modern medical institution.

BIBLIOGRAPHY

1. Software Engineering by K.K. Aggarwal.
2. https://www.slideshare.net/NausheenRizvi/hospital-mangement-system-report-file-73285573
3. https://www.slideshare.net/HimaniChopra/hospital-management-system-project
4. https://www.lovelycoding.org/hospital-management-system-project-for-final-year/
5. https://www.adroitinfosystems.com/products/ehospital-systems
6. https://www2.slideshare.net/iffi910/hospital-management-systemdatabase
7. https://www.studentprojectguide.com/projectreport/hospital-management-system-project-report