Hospital Database Management System

Presented by -
Kazi Shadman Sakib
Roll – 097
Registration Number : 2017-614-973

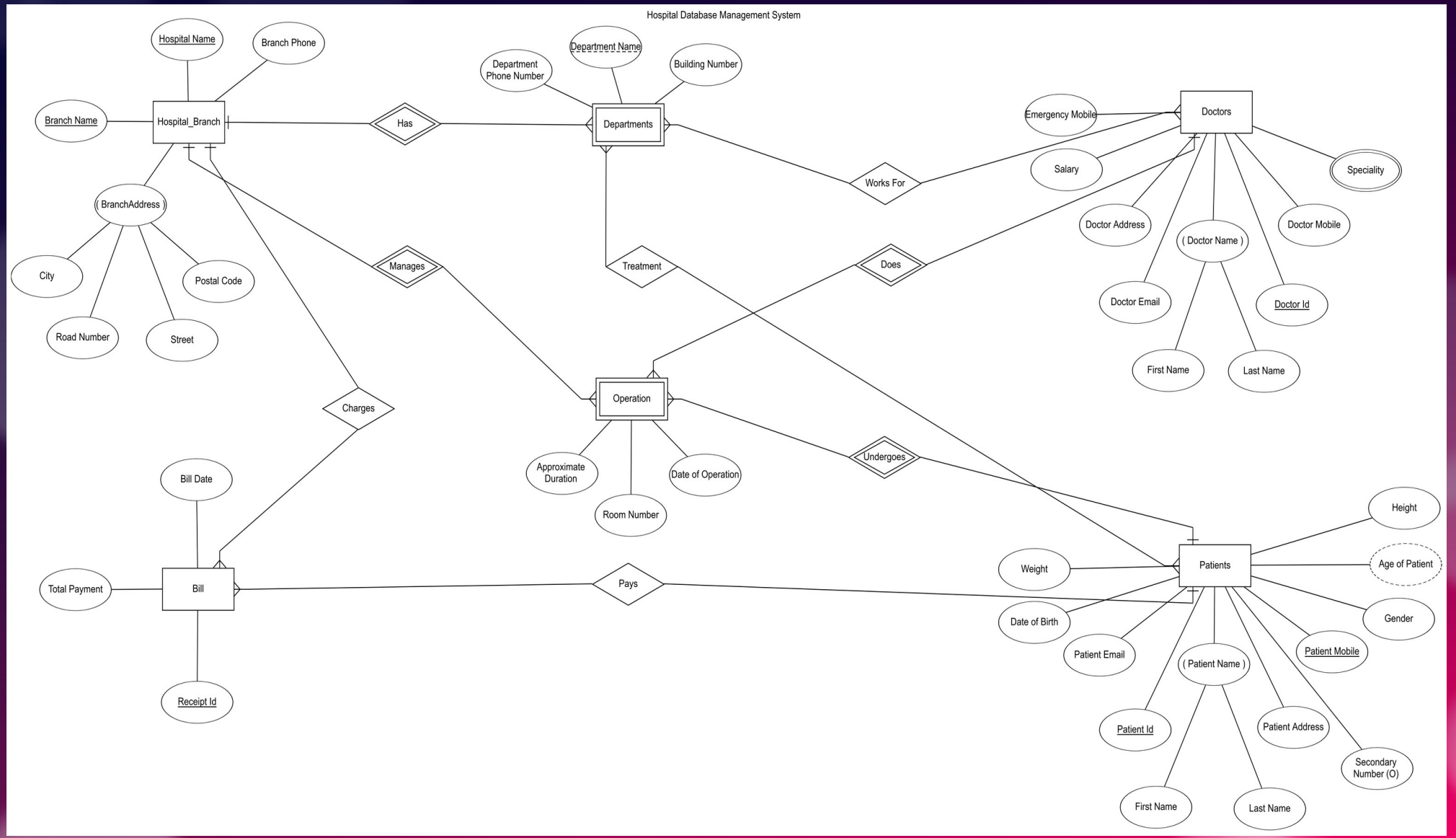
Introduction

- Data management of Hospital, Branch, Departments, Doctors, Patients.
- Stores information like :
  - Operations done by a certain Hospital
  - Treatment of a certain Patient in a certain Department of a Hospital
  - Currently which Doctor servers in which Hospital
  - Bills that a certain Hospital charges a certain Patient
  - Records of Hospitals and their Branches, Departments of a certain Hospital, Doctors, Patients.

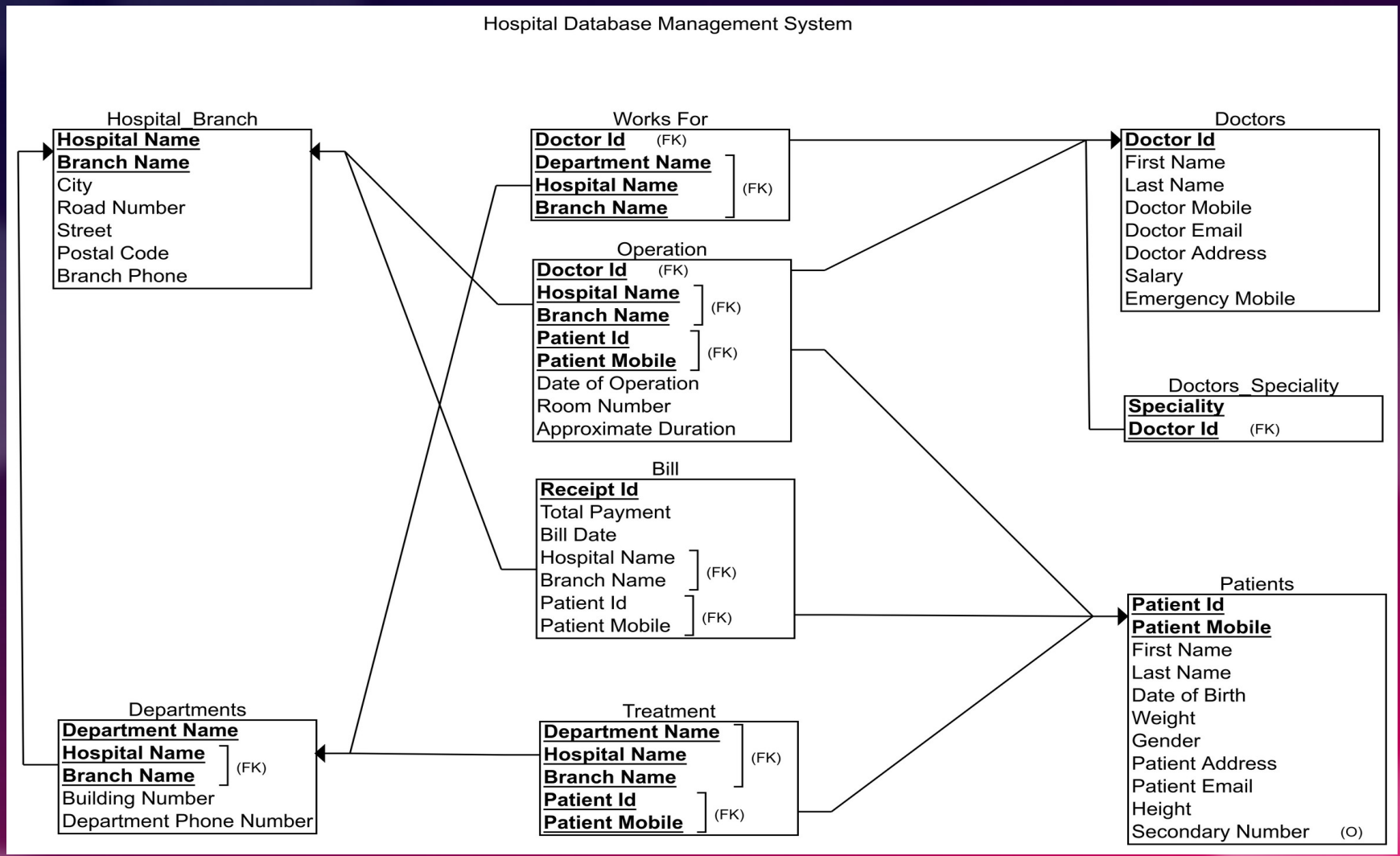
Design Phases

- Initial Phase :
  - Characterizing fully the data needs of the prospective database users. In this case, Hospital.
- Second Phase :
  - An ER Diagram is created which resembles the actual Hospital. This diagram indicates the functional requirements of the Hospital enterprise.
- Final Phase :
  - Then moving from an abstract data model to implementation of the database.
  - Here the Entity Relationship Model (ER Diagram) is converted into an Relational Model and corresponding tables are created.

Entity Relationship Model (ER Diagram)



Relational Schema Diagram



What queries can be done?

- List of patients who had an operation on a particular date / by a particular Doctor.
- List of all the Patients who have no Bills yet.
- List all the Doctors who have the max salary.
- Show all the Hospitals whose average payment is greater than 55000.
- List all the total operations done by all the doctors in decreasing order.
- List all the Hospitals which have more payment amount than a certain Hospital and its given branch.

List of Schemas with Primary and Foreign keys

- Hospital_Branch (Hospital Name, Branch Name, City, Road Number, Street, Postal Code, Branch Phone).
- Departments (Department Name, Hospital Name (FK) , Branch Name (FK), Building Number, Department Phone Number).
- Doctors (Doctor Id, First Name, Last Name, Doctor Mobile, Doctor Email, Doctor Address, Salary, Emergency Mobile).
- Doctors Speciality (Speciality, Doctor Id (FK)).
- Patients (Patient Id, Patient Mobile, First Name, Last Name, Date of Birth, Weight, Gender, Patient Address, Patient Email, Height, Secondary Number).
- Works For (Doctor Id (FK), Department Name (FK), Hospital Name (FK), Branch Name (FK)).
- Operation (Doctor Id (FK), Hospital Name (FK), Branch Name (FK), Patient Id (FK), Patient Mobile (FK), Date of Operation, Room Number, Approximate Duration).
- Bill (Receipt Id, Total Payment, Bill Date, Hospital Name (FK), Branch Name (FK), Patient Id (FK), Patient Mobile) (FK).
- Treatment (Department Name (FK), Hospital Name (FK), Branch Name (FK), Patient Id (FK), Patient Mobile (FK)).

List of Non-Trivial Functional Dependencies

- Hospital_Branch :
  - F = { (Hospital_Name, Branch_Name) → (City, Road_Number, Street, Postal_Code, Branch_Phone) }
- Departments :
  - F = { (Department_Name, Hospital_Name, Branch_Name) → (Building Number, Department_Phone_Number) }
- Doctors :
  - F = { Doctor_Id → (First_Name, Last_Name, Doctor_Mobile, Doctor_Email, Doctor_Address, Salary, Emergency Number) }
- Doctors_Speciality :
  - No Non-Trivial Functional Dependency is found here.
- Patients :
  - F = { (Patient_Id, Patient_Mobile) → (First_Name, Last_Name, Date_of_Birth, Weight, Gender, Patient_Address, Patient_Email, Height, Secondary_Number) }

- Works For :
  - No Non-Trivial Functional Dependency is found here.
- Operation :
  - F = { (Doctor_Id, Hospital_Name, Branch_Name, Patient_Id, Patient_Mobile) → (Date_of_Operation, Room_Number, Approximate_Duration) }
- Bill :
  - F = { Receipt_Id → (Total_Payment, Bill_Date, Hospital_Name, Branch_Name, Patient_Id, Patient_Mobile) }
- Treatment :
  - No Non-Trivial Functional Dependency is found here.

Are my schemas in Good Normal Forms?

- A relational schema R is in BCNF if atleast one of the following holds :
  - $\alpha \rightarrow \beta$ is trivial Functional Dependency
    - where $\beta \subseteq \alpha$.
  - α is a Super key for schema R
- Thus my schemas are in BCNF as there is no other Non-Trivial Functional Dependency where α is not a super key.

Learning Outcome

- Avoiding Redundancy.
- Design Phases while making a Database Management System.
- Trivial and Non-Trivial Functional Dependency.
- Learnt whether a particular schema is in BCNF.
- Patience while inserting Data into Schemas / Tables.
- Learnt to make a whole Database Management System (Loved it).

Conclusion

Thank you very much to our Dear Sir for spending your valuable time teaching me Database Management System and to build my project.