Hospital Management Database

Database Specification:

Purpose, Creators, Business Problems Addressed, Business Rules, Design Requirements and Design Decisions.

Database Purpose:

This project aims to help a hospital franchise with a chain of outlets, better exercise its resources while administering treatments to patients. It also allows for a more adept handling of special pandemic situations by helping the franchise manage its entire array of clinical outlets. This enables efficient interconnectivity between all franchise branches by shifting the management system online.

Business Problems Addressed:

- Building a system for efficient functioning of a particular hospital franchise.
- Managing day to day activities of various labs, wards, hospital staff, ambulances.
- Managing the clinic, pharmacy, rooms, staff, medicines.
- Quickly fetching the relevant data pertaining to the situation at hand.
- To serve patients in a better way and improve efficiency.

Business Rules:

- Each hospital will maintain the records for resources, medical devices, ambulances and medicines to check availability.
- Each hospital will have multiple wards.
- Each staff member will be appointed to a department.
- Each hospital will have one lab where all tests are conducted.
- A doctor works in one particular hospital only and may have a specialty.
- A patient will be assigned one bed only, in the room of a particular

ward.

- The patient may have to undergo a few tests.
- Each staff member has an assigned department.
- A particular ward may have multiple rooms.
- A room can have one or more beds.
- A patient may have multiple case files.
- A case can have multiple bills.

- **Design Requirements:**

- In the ERD, the yellow key signifies a Primary Key, and the red key signifies a Foreign Key.
- Used the Crow's Foot Notation.
- Specified the primary key fields and the foreign key fields in each table
- by specifying PK beside the fields.
- Drawn a line between the fields of each table to show the
- relationships between each table. This line points directly to the fields
- in each table that are used to form the relationship.
- Specified which table is on the many side of the relationship by placing
- A crow's feet symbol next to the field where the line ends.

Design Decisions:

Entity name	Why this entity has been included	How is this Entity related to other Entities
Hospital	One of the primary purposes of this database is to collect identification information about each Hospital that falls under the franchise umbrella to help keep a record of each Hospital. The data to be collected from each Hospital includes: The Hospital ID, The Hospital's	As the core entity in the database, the Hospital entity's primary key Hosp_ID relates it to Doctor, Pharmacy, Resources, Ambulance, Medical Devices, Laboratory and Ward. A hospital has one to many relationship with doctors, resources, medical devices,

	name, Contact number, Address, City and a Zip Code.	ambulance and ward. A hospital can have many or one pharmacy. There's a one to one relationship between the hospital and laboratory. In the model, a particular hospital can have only one laboratory. They are all non-identifying relationships. The Hospital Entity's attributes include Address_Line1, State, City, ZipCode to make it more comprehensive.
Doctor	Hospitals cannot function without doctors as they are the backbone of this industry. This database collects and maintains records of the doctors working in each hospital. The information collected includes: The Doctor's ID, The Doctor's First and Last name, Contact number, Address, The Hospital ID of the hospital in which that Doctor works and The Doctor's Speciality ID.	There's many to one relationship between Doctor Entity and Hospital Entity. There are many doctors in a hospital and a doctor is working in a particular hospital. The Doctor is also related to the DoctorRef Entity in a "one to zero or many relationship". It is an identifying relationships A doctor sees many patients, and a patient may see several doctors. A doctor may or may not have a speciality or can have multiple specialties. The Doc_Address attribute in Doctor Entity was split into Doc_AdressLine1, City, State and Zipcode, because The Doc_Address is a composite attribute and we needed to split it into atomic attributes.
Patient	Hospitals are patient-centric. They strive to administer proper treatment and ensure patient safety. To do this, patient records must always be up-to-date and very accurate. The Patient identifying information collected includes: The Patient ID, The Patient's First, Middle and Last name, The Patient's	Patient Entity is related to the DoctorRef Entity in a "one to many relationship". Instead of an Age attribute we have a Date of Birth attribute, which makes calculating the patient's age more convenient. It is an identifying relationship. A patient may see several doctors

	Email ID, Contact Number, Age, Sex, Address, City, Zip Code, The date on which the patient visited the hospital and the ID of the Doctor under which the Patient is undergoing treatment.	and a doctor may see several patients.
Pharmacy	Hospitals generally have a pharmacy which sells medicines to the patients coming to the hospital, thus each Hospital must maintain a record of this. The information collected includes: The Medicine ID, The Medicine's Description, The Medicine's Type, The Medicine's current Quantity possessed by the Hospital and the Hospital ID to which that particular medicine belongs.	The Pharmacy Entity has many to one relationship with the Hospital Entity. A hospital can have one or many pharmacies and a pharmacy is related to that particular hospital. It is a non-identifying relationship.
Resources	Medical equipment such as surgical masks, gloves, syringes, PPE kits, scrubs etc are a staple of every Hospital. The Hospital needs to maintain a record of all its equipment/resources. The information collected includes: The Resource ID, The Category to which that Resource belongs, The Resource's Quantity and the Hospital ID of the hospital to which that Resource pertains.	Resources Entity is related to Hospital Entity in many to one relationship. A hospital can have many additional resources and a particular resource is associated with that particular hospital only. It is a non-identifying relationship.
Ambulance	Ambulances are a key factor in ferrying patients to and fro from hospitals which is why each Hospital must also maintain a record of each Ambulance it possesses. The identifying information collected for each Ambulance includes: The Ambulance ID, Type of Ambulance, Ambulance Availability and the Hospital's ID to which that Ambulance belongs.	We have created an Ambulance Entity which is related to the Hospital Entity in many to one relationship. A hospital can have many ambulances and a particular ambulance belongs to only one hospital. It is a non-identifying relationship.
Medical Devices	Medical Devices such as ECG machines, ventilators, MRI	The relationship between Medical Devices Entity and

	machines etc are heavily used while administering treatment to a patient. They are expensive devices and a record of them must be kept. The identifying information collected for each device includes: The Device ID, Name of the Device, The Device's current Quantity possessed by the Hospital and the Hospital's ID to which that Device belongs.	Hospital Entity is one to many. A hospital can have many Medical Devices and that respective set of Medical Devices belongs to that particular Hospital. It forms a non-identifying relationship.
Specialty	Doctors generally have a speciality (ex: cardiologist, neurologist etc) and based on their speciality, a Doctor treats the patient. A record of all types of speciality treatments available must be kept. The identifying information collected includes: The Speciality ID and the Speciality Description.	Specialty Entity is related to the DoctorRef Entity in a "one to zero or many" relationship. A doctor may or may not have a specialty or a doctor can have multiple specialties. It is an identifying relationship.
Ward	A hospital has several wards ex: the Casualty ward, The General ward, The emergency ward etc. A record of each ward the hospital possesses must be maintained. The identifying information for each ward includes: The Ward ID, The Ward's name, The Hospital ID of the hospital to which that ward belongs, the Floor number on which the ward is situated and the Case ID.	Ward Entity is related to Hospital Entity in many to one relationship. A hospital can have multiple wards and a ward belongs to a particular hospital. A ward can also have multiple rooms. They are all non-identifying relationships.
Room Details	Sometimes Patients need to be admitted so that their condition can be monitored and in case of deterioration, immediate care can be administered. To do this, Hospitals must have rooms in which the patients can stay, thus records for the same must be maintained. The identifying information for the Room records includes: The Room	Room Details Entity is related to Ward Entity and Bed_Details Entity. Room Details is in one Ward and a Ward can have multiple rooms. Bed_Details Entity will show all the bed details of a particular patient. And a bed is related to a single patient at a time. They are all non-identifying relationships.

	Number, The Ward ID of the ward to which the bed belongs, The Type of room and the availability of that room.	
Bed_Details	Sometimes Patients need to be admitted so that their condition can be monitored and in case of deterioration, immediate care can be administered. To do this, Hospitals must have rooms in which the patients can stay, thus records for the same must be maintained. The identifying information for the Bed's records includes: The Bed Number, The Room Number of the room to which that bed belongs, the Availability of the bed and the Case ID of the case with which that bed is associated.	Depending upon the type of case, a patient may or may not get admitted in the hospital. Bed_Details Entity will show all the bed details of a particular patient. And one bed is related to a single patient at a time. They are all non-identifying relationships.
Type of cases	A hospital must maintain patient case files for future reference. Each case file is unique and must be highly accurate. The identifying information for each Case includes: The Case ID, The Case Type, Case Description, The Patient ID of the patient to whom the file belongs, The Patient's Admittance Date and The Patient's Discharge Date.	The Type of cases Entity is related to Patient Entity. A patient can have multiple cases. The relationship between the patient and type of cases is one to many as a patient can visit a hospital multiple times to the hospital for the treatment and it has a non identifying relationship. Depending upon the case, if a patient is admitted he would be allotted a bed in a particular ward. A patient can have multiple bills. They are all non-identifying relationships.
Test	Sometimes to determine the cause or effect of a disease a patient has to undergo lab tests. A record of the tests administered by the hospital must be kept for future reference. The information collected includes: The Test ID, The Test's Name, Date of test administration, The Test's	Test Entity is related with Type of Cases Entity in zero or many to one relationship. The patient's doctor may or may not suggest the patient to take a test. And a test is conducted in a particular laboratory and that laboratory can have many tests. They are all non-identifying relationships. Lab_ID (PK from

	charges and the Case ID with which the test was associated.	Lab) has been added to the Test Entity.
Bill	Hospitals must keep records of the bills racked up by the patients. These bills include charges for the care administered, hospital stay, test charges etc. The identifying information collected for each bill includes: The Receipt ID, The Case ID, the Total Charge, Method of Payment and the Date on which the Payment was made.	Bill Entity is related to Type of Case Entity. Depending upon the case a patient may have one or many bills. And a particular bill is related to only one type of case. It is a non-identifying relationship.
Laboratory	Sometimes to determine the cause or effect of a disease a patient has to undergo lab tests. These tests are administered at the Hospital's laboratory. Each hospital's laboratory records must be maintained. The identifying information, for the laboratory, collected includes: The Lab ID, The Hospital ID of the hospital to which that lab belongs, The Test ID of the test administered and The Lab's description.	Laboratory Entity is related to Hospital Entity in one to one relationship. In the model, a hospital is associated with a single laboratory. And Laboratory Entity is related to Test Entity with one to many relationship. A laboratory can conduct multiple tests. They are all non-identifying relationships.
Staff	Apart from the Doctors there are a lot of other staff members like nurses and janitors,who are crucial for a hospital to function effectively. Records of each staff member must be meticulously maintained by the hospital. The identifying information for a staff member collected includes: The Staff ID, The Staff member's First and Last name, Contact Number and the Hospital ID of the Hospital in which that staff member works and The Department ID of the Department in which that staff member works.	The relationship between Hospital Entity and Staff Entity is one to many relationship. A hospital can have multiple staff and a staff member belongs to a particular hospital. Staff Entity is also related to Staff Department Entity which will indicate the department of that particular staff member. They are all non-identifying relationships.
Staff Department	It is necessary for the hospital to adequately allot staff to each	Staff Department Entity is related to Staff Entity with one-

	department. Thus, records of a staff member's appointment must be maintained. The information collected includes: The Staff member's Department ID and description of the department in which the staff member works.	to-many relationship. The Staff Department Entity will have the department_id and department description, which in turn will help to identify which staff member is working in which department. And a staff member can be in one department only. It is a non-identifying relationship.
DoctorRef	We created this associative entity as a patient can have multiple doctors, and a doctor can see many patients.	An associative entity "DoctorRef" is created between Doctor, Patient and Speciality, which has Doctor_ID (PK from Doctor), Patient_ID (PK from Patient) and Speciality_ID (PK from Speciality), and all fields are combinations of Primary Keys and Foreign Keys.