

# **EHR for Emergency Care**

## **Group 4:**

Shiny Pidugu - 801320154

Meghna Reddy Aleti - 801326493

Lakshmi Sahithi Talluri - 801324888

**Due Date:** 10/18/2023

## **Table of Contents**

### Project Overview

- Background Study 1
- Scope 1
- Motivation 1

### Database Design 2

### Functional Requirements 2

### Entity Relationship

- ER Diagram 4
- UML Diagram 6

### Proof of BCNF 8

### Table Information 9

### Tools Used 11

## **Project Overview:**

### **Background Study:**

The proposed project involves the development of an Electronic Health Records (EHR) database tailored for an emergency room environment. The primary objective of this database is to efficiently store and manage crucial patient data, provider details, clinical records, appointment scheduling, orders, billing information, and other pertinent aspects related to emergency patient care. The goal is to optimize the patient care process, enhance patient outcomes, and enable seamless coordination among healthcare providers.

### **Scope:**

The project's scope centers on the creation of a comprehensive database system serving as the backend for a simplified EMR system, focusing specifically on the emergency department. This initiative entails the construction of a UML diagram and a well-structured database schema consisting of various interconnected tables, encompassing key components such as addresses, insurance details, provider information, facility records, patient profiles, visit logs, clinical data, appointment schedules, orders, bed management, supply inventory, and billing data. Each table will be equipped with unique attributes and primary keys, establishing associations through foreign keys to facilitate data relationships. The database will enable users to input and manage essential patient demographic information, provider details, visit logs, clinical data, and other relevant data based on specific emergency scenarios. Additionally, it will support data editing for rectifying entry errors or updating legitimate information, along with robust search functionalities based on patient names, IDs, and visit dates. The system will also be equipped with reporting capabilities, facilitating the identification of patients meeting specific criteria.

### **Motivation:**

The driving force behind this undertaking is to enhance the quality of patient care in emergency room settings by ensuring prompt access to critical patient data for healthcare providers. By implementing an EHR database, healthcare professionals can swiftly retrieve relevant patient information, thereby minimizing the occurrence of medical errors and bolstering overall patient outcomes. Moreover, the comprehensive EHR database will foster effective communication among healthcare providers, ensuring timely and appropriate care delivery for patients. Ultimately, the project endeavors to elevate the standard of patient care within emergency room settings, providing crucial support to healthcare providers in delivering top-notch healthcare services.

## **Database Design:**

Presented is a database design tailored for an emergency Electronic Health Records (EHR) system, encompassing 12 distinct tables, each representing a discrete entity within the emergency EHR framework. Each table is equipped with a primary key, ensuring the unique identification of every entry, while inter-table relationships are established through the use of foreign keys. For instance, the "ADDRESSES" and "INSURANCES" tables incorporate foreign keys referencing the "PATIENTS" table, whereas the "VISITS" table integrates foreign keys linked to the "PATIENTS," "PROVIDERS," and "FACILITIES" tables.

The "CLINICAL INFORMATION" table captures and logs pertinent clinical data derived from patient visits, including symptoms, discharge diagnoses, and administered medications. The "ORDERS" table documents the tests prescribed for patients during their visits, while the "BEDS" table catalogs information concerning hospital beds, detailing their respective types.

Furthermore, the "BILLING" table houses essential information pertaining to the billing process of patient visits, recording details such as the visit ID and corresponding billing amount. This table is intricately linked to the "VISITS" table through a foreign key association, facilitating seamless tracking of billing information for each individual visit.

In essence, this database design furnishes a comprehensive and methodical approach to securely store and manage patient and clinical data within an emergency EHR system. It enables the efficient monitoring of patient visits, clinical specifics, appointment scheduling, and billing processes, while facilitating convenient access to pertinent data regarding healthcare providers, facilities, bed allocations, and inventory supplies.

## **Functional Requirements:**

**Patients:** This section encompasses a table holding essential demographic data for each patient, including details such as their name, date of birth, and contact particulars.

**Providers:** This category involves a table containing pertinent information about each healthcare provider, including their name, area of specialization, and contact information.

**Addresses:** This section comprises a table dedicated to storing address information for patients, including details such as city, state, and postal code.

**Insurances:** This category incorporates a table responsible for managing insurance information, including data on the insurance company's name and policy numbers.

**Visits:** This section entails a table capturing information about each visit to the emergency department, encompassing data such as the patient ID, provider ID, timestamp, and facility details.

**Clinical Information:** This category involves a table focused on documenting the clinical care administered during each patient visit, encompassing data related to symptoms, diagnoses, prescriptions, and the outcomes of various tests and procedures.

**Facilities:** This section encompasses a table dedicated to recording information pertaining to the various facilities, including their names and types.

**Appointments:** This category involves a table dedicated to managing information about scheduled appointments for patients, detailing data such as patient IDs, provider IDs, appointment dates and times, and the allocated examination rooms.

**Beds:** This section comprises a table focused on storing information about the beds within the emergency department, including specific details such as bed IDs, locations, and bed types.

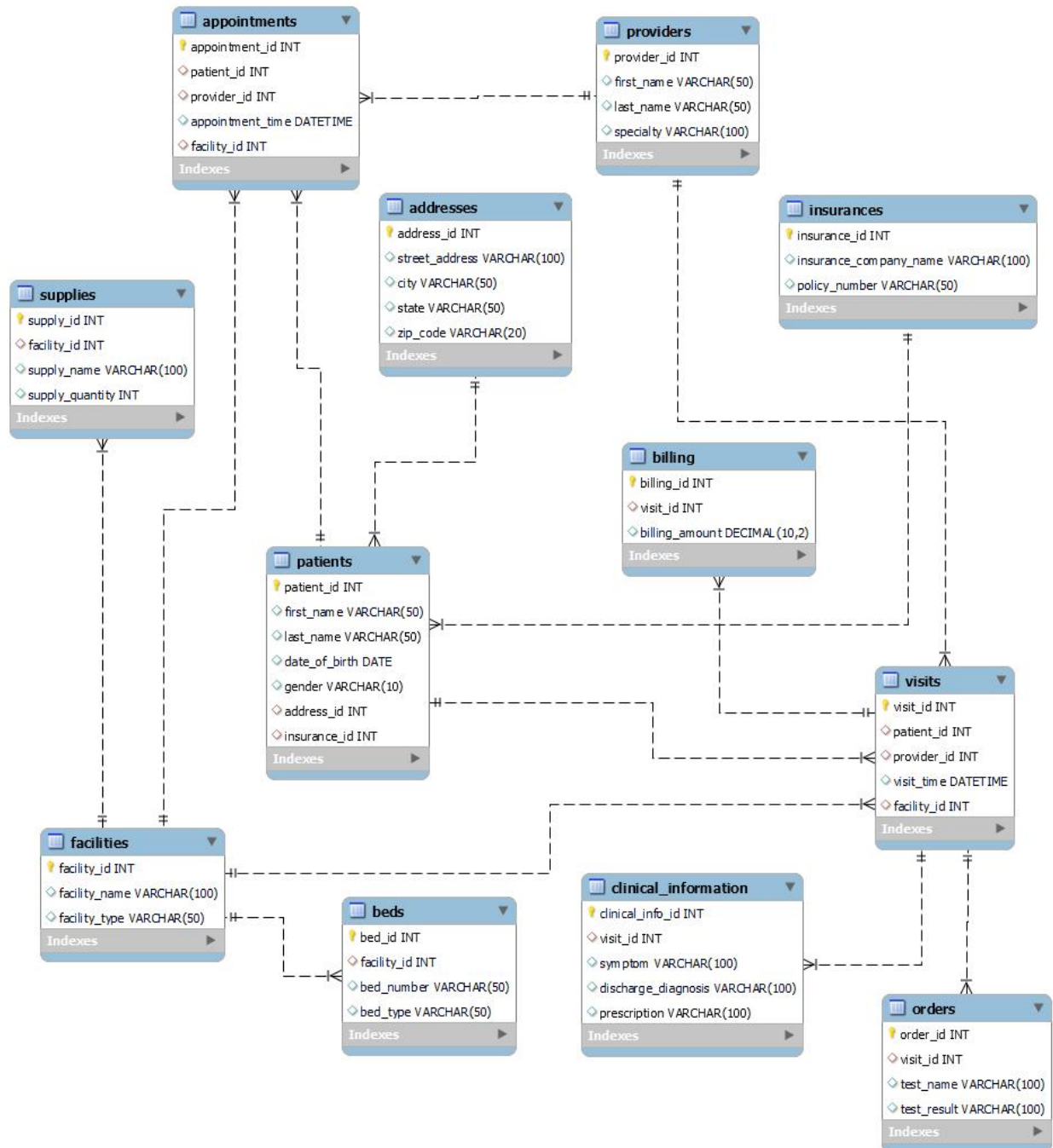
**Supplies:** This category involves a table dedicated to tracking the available supplies within the emergency department, detailing data such as supply IDs, names, quantities, and respective locations.

**Orders:** This section encompasses a table dedicated to recording information about laboratory tests, including the names of tests and the corresponding test results.

**Billing:** This category comprises a table dedicated to managing information regarding the billing process for each visit, including details such as patient IDs, provider IDs, visit IDs, and associated charges.

## Entity Relationship Diagram:

The ER diagram shows a simplified database schema for an Emergency room Electronic Health Record (EHR) system. The system is designed to manage patient information, including their addresses, insurance details, clinical information, and visits to healthcare providers.



The ER diagram includes the following entities and their relationships:

**Patients:** This entity represents patients' personal information, including their first and last names, date of birth, gender, and their address and insurance IDs. A patient can have multiple visits to healthcare providers, and each visit is associated with a unique visit ID.

**Addresses:** This entity represents the addresses of patients. Each address is identified by a unique address ID and includes the street address, city, state, and zip code.

**Insurances:** This entity represents the insurance details of patients, including the name of the insurance company and the policy number. Each insurance is identified by a unique insurance ID.

**Providers:** This entity represents healthcare providers' information, including their first and last names and specialty. Each provider is identified by a unique provider ID.

**Facilities:** This entity represents the healthcare facilities' information, including their name and type. Each facility is identified by a unique facility ID.

**Visits:** This entity represents patients' visits to healthcare providers, including the visit ID, patient ID, provider ID, visit time, and facility ID. Each visit can have multiple clinical information, orders, and billing associated with it.

**Clinical\_Information:** This entity stores clinical information associated with a patient's visit, including the visit ID, symptoms, discharge diagnosis, and prescription. Each clinical information has a unique clinical info ID.

**Appointments:** This entity represents appointments made by patients with healthcare providers, including the patient ID, provider ID, facility ID, and the date and time of the appointment. Each appointment has a unique appointment ID.

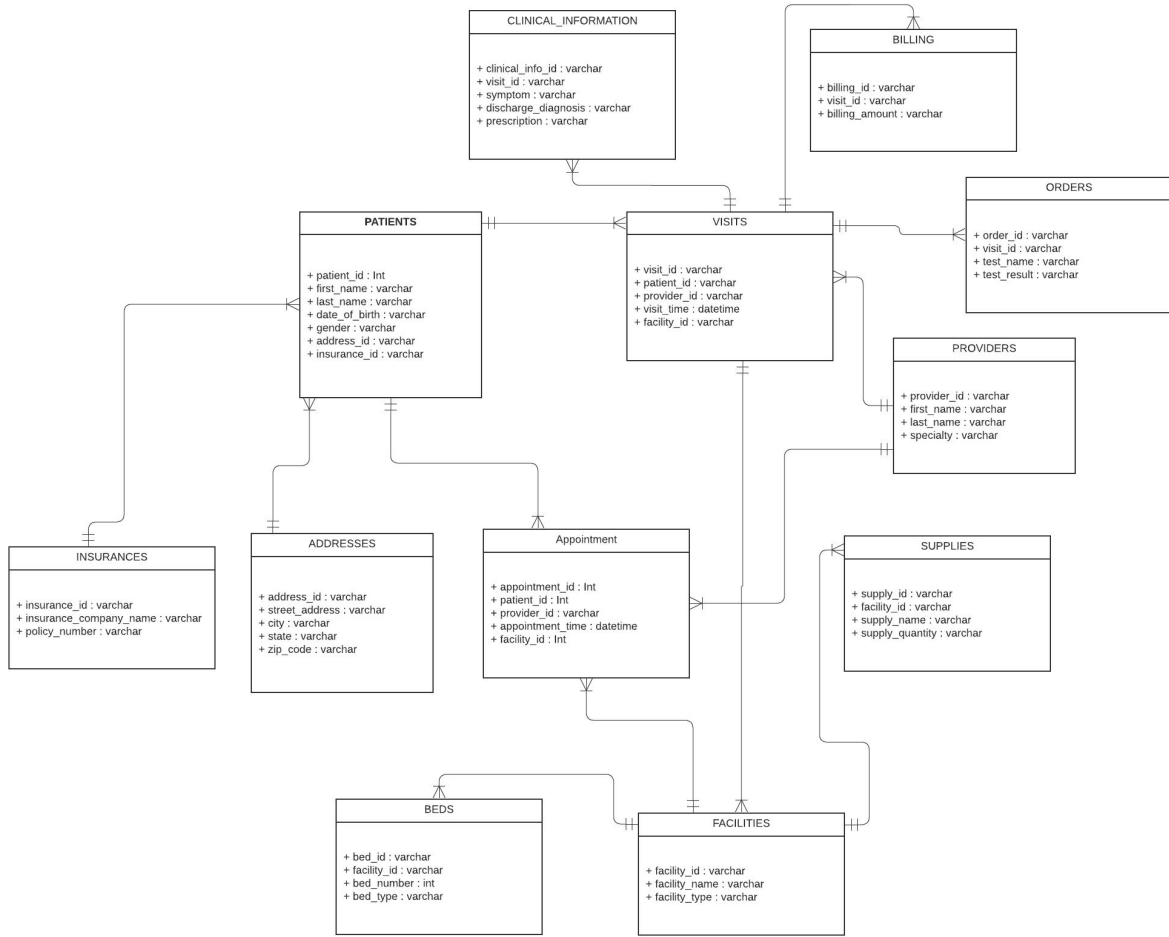
**Orders:** This entity stores orders made by healthcare providers during a patient's visit, including the visit ID, test name, and test result. Each order has a unique order ID.

**Beds:** This entity stores information about beds in healthcare facilities, including the facility ID, bed number, and bed type. Each bed has a unique bed ID.

**Supplies:** This entity stores information about supplies in healthcare facilities, including the facility ID, supply name, and supply quantity. Each supply has a unique supply ID.

**Billing:** This entity stores information about the billing associated with a patient's visit, including the visit ID and billing amount. Each billing has a unique billing ID.

### UML Diagram:



The table relationships for the above UML diagram can be described as follows:

**PATIENTS** table has a many-to-one relationship with **ADDRESSES** and **INSURANCES** tables, as each patient can have only one address and one insurance, but each address and insurance can be associated with multiple patients.

**VISITS** table has a many-to-one relationship with **PATIENTS**, **PROVIDERS**, and **FACILITIES** tables, as each visit can have only one patient, one provider, and one facility, but each patient, provider, and facility can have multiple visits.



**CLINICAL\_INFORMATION** table has a many-to-one relationship with **VISITS** table, as each visit can have only one set of clinical information.

**APPOINTMENTS** table has a many-to-one relationship with **PATIENTS**, **PROVIDERS**, and **FACILITIES** tables, as each appointment can have only one patient, one provider, and one facility, but each patient, provider, and facility can have multiple appointments.

**ORDERS** table has a many-to-one relationship with **VISITS** table, as each order can be associated with only one visit, but each visit can have multiple orders.

**BEDS** and **SUPPLIES** tables have a many-to-one relationship with **FACILITIES** table, as each bed and supply can be associated with only one facility, but each facility can have multiple beds and supplies.

**ADDRESSES** table has a one-to-many relationship with the **PATIENTS** table. This means that a patient can have only one address, but an address can be associated with multiple patients.

**INSURANCES** table has a one-to-many relationship with the **PATIENTS** table. This means that a patient can have only one insurance, but an insurance company can be associated with multiple patients.

**PROVIDERS** table has a many-to-many relationship with the **VISITS** table. This means that multiple providers can be associated with multiple visits, and vice versa.

**FACILITIES** table has a many-to-one relationship with the **BEDS**, **SUPPLIES**, **VISITS**, and **APPOINTMENTS** tables. This means that a facility can have multiple beds, supplies, visits, and appointments, but each bed, supply, visit, and appointment can only be associated with one facility.

**SUPPLIES** table has a many-to-one relationship with the **FACILITIES** table. This means that a facility can have multiple supplies, but each supply can only be associated with one facility.

**BILLING** table has a many-to-one relationship with the **VISITS** table. This means that a visit can have only one billing, but each billing can only be associated with one visit.

## **Proof of BCNF:**

Proving that a set of tables are in Boyce-Codd Normal Form (BCNF) involves verifying that each table meets the following criteria:

- Each non-trivial functional dependency in the table is a dependency on a super key.
- There are no non-trivial dependencies between candidate keys.
- All attributes are functionally dependent on the primary key.

Based on our schema, we can verify that the tables are in BCNF by examining each table and its relationships to other tables in the schema:

- **ADDRESSES** table has a primary key `address_id`, and there are no non-trivial functional dependencies in the table.
- **INSURANCES** table has a primary key `insurance_id`, and there are no non-trivial functional dependencies in the table.
- **PROVIDERS** table has a primary key `provider_id`, and there are no non-trivial functional dependencies in the table.
- **FACILITIES** table has a primary key `facility_id`, and there are no non-trivial functional dependencies in the table.
- **PATIENTS** table has a primary key `patient_id`. The `address_id` and `insurance_id` columns are foreign keys that reference the **ADDRESSES** and **INSURANCES** tables, respectively. All non-key attributes in the table are functionally dependent on the primary key.
- **VISITS** table has a primary key `visit_id`. The `patient_id`, `provider_id`, and `facility_id` columns are foreign keys that reference the **PATIENTS**, **PROVIDERS**, and **FACILITIES** tables, respectively. All non-key attributes in the table are functionally dependent on the primary key.
- **CLINICAL\_INFORMATION** table has a primary key `clinical_info_id`. The `visit_id` column is a foreign key that references the **VISITS** table. All non-key attributes in the table are functionally dependent on the primary key.
- **APPOINTMENTS** table has a primary key `appointment_id`. The `patient_id`, `provider_id`, and `facility_id` columns are foreign keys that reference the **PATIENTS**, **PROVIDERS**, and **FACILITIES** tables, respectively. All non-key attributes in the table are functionally dependent on the primary key.

- **ORDERS** table has a primary key `order_id`. The `visit_id` column is a foreign key that references the **VISITS** table. All non-key attributes in the table are functionally dependent on the primary key.
- **BEDS** table has a primary key `bed_id`. The `facility_id` column is a foreign key that references the **FACILITIES** table. All non-key attributes in the table are functionally dependent on the primary key.
- **SUPPLIES** table has a primary key `supply_id`. The `facility_id` column is a foreign key that references the **FACILITIES** table. All non-key attributes in the table are functionally dependent on the primary key.
- **BILLING** table has a primary key `billing_id`. The `visit_id` column is a foreign key that references the **VISITS** table. All non-key attributes in the table are functionally dependent on the primary key.

Therefore, we can conclude that all tables in the database are in BCNF.

## **Table Information:**

The database of Electronic Health Records (EHRs) in emergency rooms contains digital records of patient health information collected during emergency room visits. These records include details such as patient demographics, medical history, medications, allergies, vital signs, lab results, imaging studies, diagnoses, and treatment plans.

### **Table 1: Patients**

- `patient_id`
- `first_name`
- `last_name`
- `date_of_birth`
- `gender`
- `address_id`
- `insurance_id`

### **Table 2: Providers**

- `provider_id`
- `first_name`
- `last_name`
- `specialty`

**Table 3: Addresses**

- address\_id
- street\_address
- city
- state
- zip\_code

**Table 4: Insurances**

- insurance\_id
- insurance\_company\_name
- policy\_number

**Table 5: Visits**

- visit\_id
- patient\_id
- provider\_id
- visit\_time
- facility\_id

**Table 6: Clinical Information**

- clinical\_info\_id
- visit\_id
- symptom
- discharge\_diagnosis
- prescription

**Table 7: Facilities**

- facility\_id
- facility\_name
- facility\_type

**Table 8: Appointments**

- appointment\_id

- patient\_id
- provider\_id
- appointment\_time
- facility\_id

#### **Table 9: Beds**

- bed\_id
- facility\_id
- bed\_number
- bed\_type

#### **Table 10: Supplies**

- supply\_id
- facility\_id
- supply\_name
- supply\_quantity

#### **Table 11: Orders**

- order\_id
- visit\_id
- test\_name
- test\_result

#### **Table 12: Billing:**

- billing\_id
- visit\_id
- billing\_amount

#### **Tools Used:**

- MySql Workbench
- Lucid Chart