DBMS - Mini Project
Phamaceutical Database

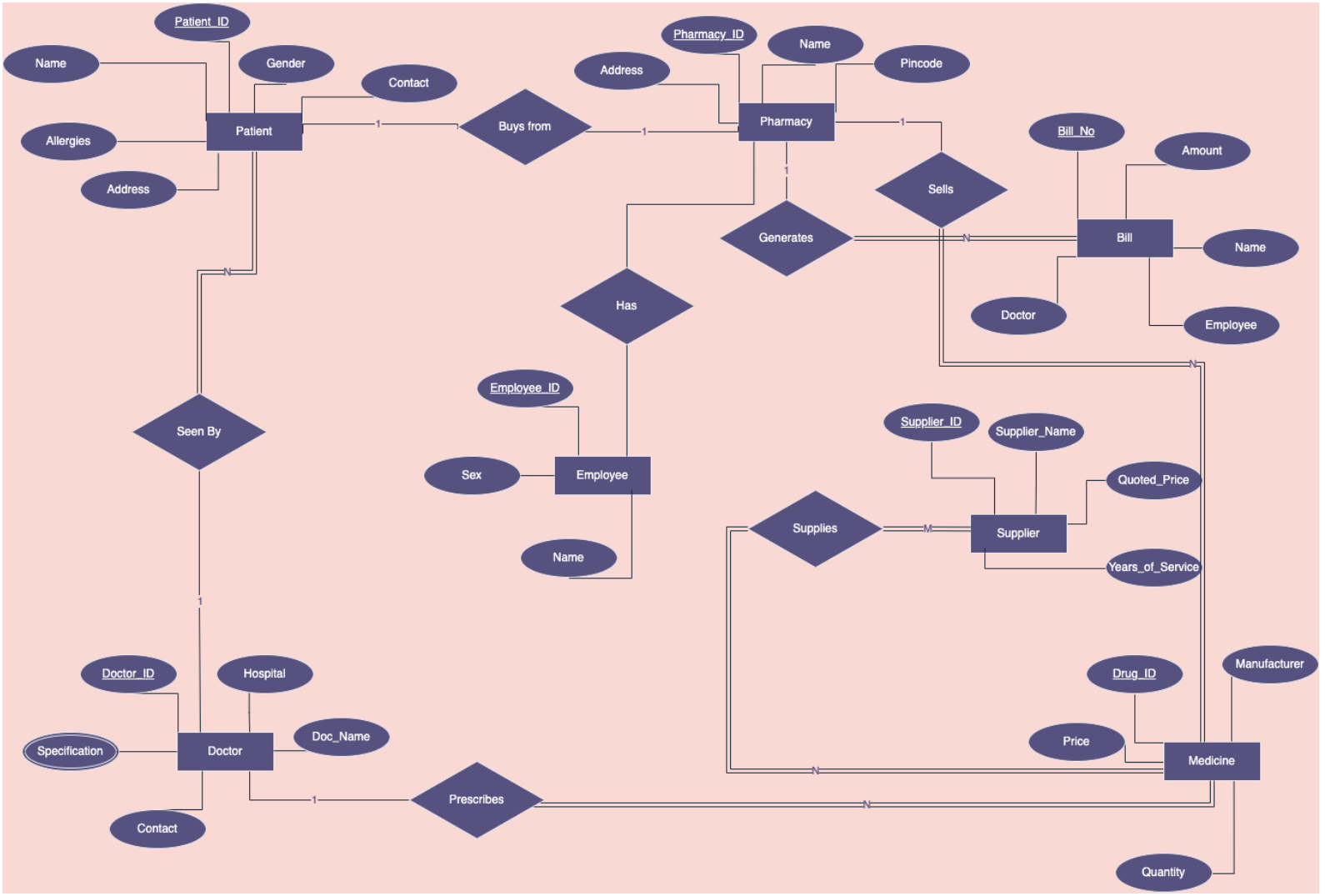
Submitted By:
Name Naman Choudhary
SRN PES2UG20CS209
V Semester Section D

Short Description and Scope of the Project

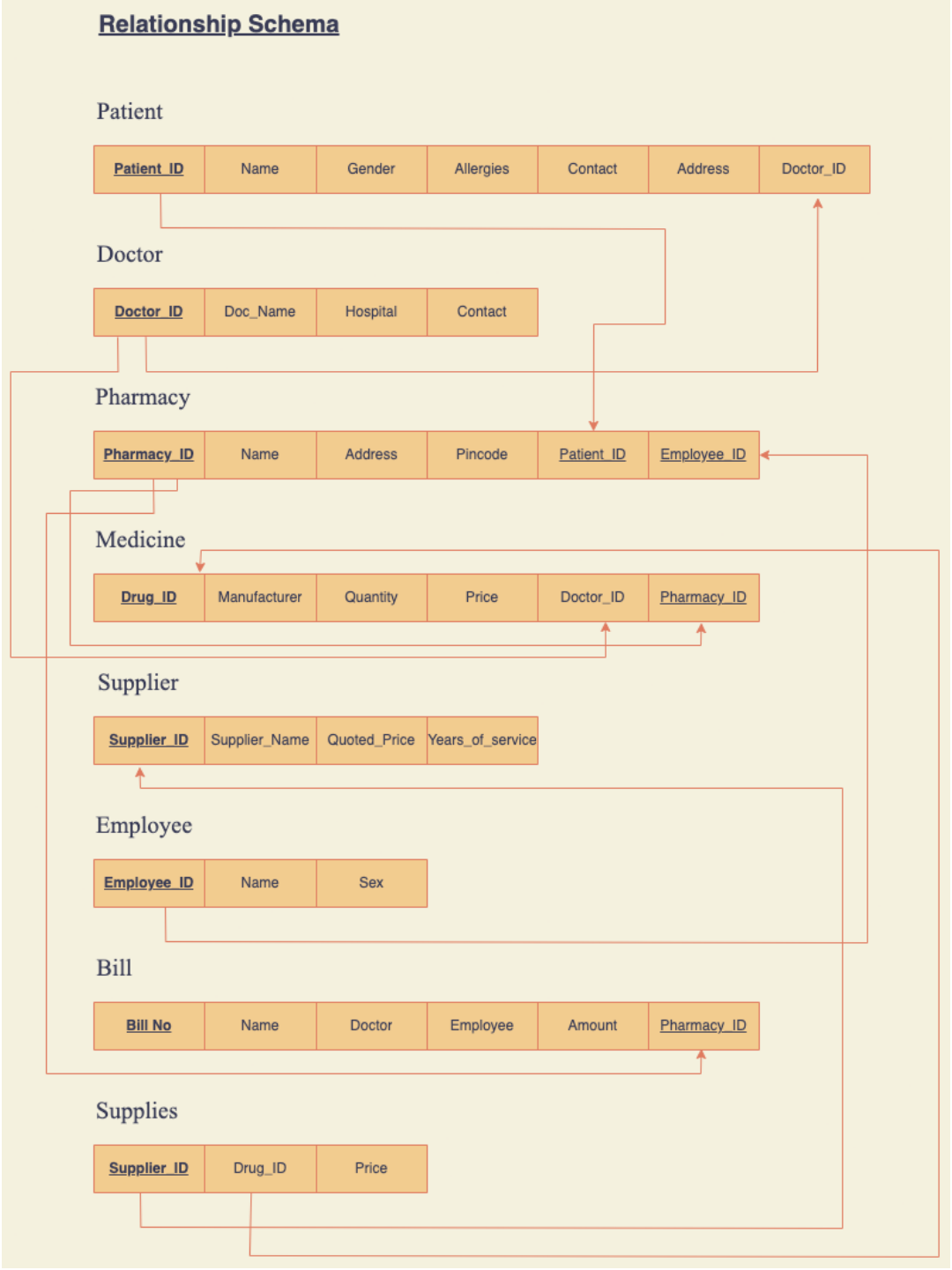
The Pharmaceutical management system will help Pharmacists across the country to manage, organise and keep track of medicine available in inventory and in the market. This will help Pharmacists to never be low on stock of medicine, as unavailability of medicine can put people suffering from long-term and short-term illnesses at risk. This system will also help in maintaining and tallying the expenditure as well as, keeping track of user information about required medicine of repeated customers

With this project, my aim was to develop a comprehensive system that could deal with challenges faced in day to day operation of a modern pharmacy.

ER Diagram


Patient_ID
Name
Gender
Contact
Allergies
Patient
Address
Buys from
1
1
Pharmacy_ID
Name
Address
Pincode
Pharmacy
1
Bill_No
Amount
Sells
Generates
Bill
Name
Doctor
Employee
N
N
Has
Employee_ID
Seen By
Supplier_ID
Supplier_Name
Sex
Employee
Quoted_Price
Supplies
M
Supplier
Name
Years_of_Service
1
Doctor_ID
Hospital
Drug_ID
Manufacturer
Specification
Doctor
Doc_Name
Price
Medicine
Prescribes
1
N
Contact
Quantity

Relational Schema



DDL statements - Building the database

SQL

```sql
-- Table creation with DDL commands

CREATE TABLE CUSTOMER (
Aadhaar BIGINT NOT NULL,
first_name CHAR(255) NOT NULL,
last_name CHAR(255) NOT NULL,
phone BIGINT NOT NULL UNIQUE,
gender CHAR(1) NOT NULL,
address CHAR(255) NOT NULL,
date_of_birth DATE NOT NULL,
insurance_id BIGINT NOT NULL,
PRIMARY KEY (Aadhaar)
);

ALTER TABLE Customer
ADD CONSTRAINT insures FOREIGN KEY (insurance_id) REFERENCES Insurance (
ON DELETE CASCADE;

,CREATE TABLE Prescription (
prescription_id BIGINT NOT NULL,
Aadhaar BIGINT NOT NULL,
doctor_id BIGINT NOT NULL,
prescribed_date DATE NOT NULL,
PRIMARY KEY (prescription_id)
);

ALTER TABLE Prescription
ADD CONSTRAINT holds FOREIGN KEY (Aadhaar) REFERENCES Customer (Aadhaar)
ON DELETE CASCADE;

CREATE TABLE PRESCRIBED_DRUGS (
prescription_id BIGINT NOT NULL,
drug_name CHAR(255) NOT NULL,
prescribed_quantity BIGINT NOT NULL,
refill_limit BIGINT NOT NULL,
PRIMARY KEY (prescription_id,drug_name)
);

ALTER TABLE PRESCRIBED_DRUGS
ADD CONSTRAINT consists_of FOREIGN KEY (prescription_id)
REFERENCES Prescription (prescription_id) ON DELETE CASCADE;

CREATE TABLE Orders (
```

```SQL
CREATE TABLE Orders (
order_id BIGINT NOT NULL,
prescription_id BIGINT NOT NULL,
EmployeeID BIGINT NOT NULL,
order_date DATE NOT NULL,
PRIMARY KEY (order_id)
);

ALTER TABLE Orders
ADD CONSTRAINT prepares FOREIGN KEY (EmployeeID) REFERENCES Employee (I

ALTER TABLE Orders
ADD CONSTRAINT uses FOREIGN KEY (prescription_id) REFERENCES
Prescription (prescription_id);

CREATE TABLE ORDERED_DRUGS (
order_id BIGINT NOT NULL,
drug_name CHAR(255) NOT NULL,
batch_number BIGINT NOT NULL,
ordered_quantity BIGINT NOT NULL,
Price BIGINT NOT NULL,
PRIMARY KEY (order_id, drug_name,batch_number)
);

ALTER TABLE ORDERED_DRUGS
ADD CONSTRAINT containss FOREIGN KEY (order_id) REFERENCES Orders (orde
ON DELETE CASCADE;

ALTER TABLE ORDERED_DRUGS
ADD CONSTRAINT Fulfilled_From FOREIGN KEY (drug_name, batch_number)
REFERENCES Medicine(drug_name, batch_number);

CREATE TABLE Insurance (
insurance_id BIGINT NOT NULL,
company_name CHAR(255) NOT NULL,
start_date DATE NOT NULL,
end_date DATE NOT NULL,
co_insurance BIGINT NOT NULL,
PRIMARY KEY (insurance_id)
);

CREATE INDEX Insurance_Company_Name ON Insurance (company_name);
```

```sql
CREATE TABLE Employee (
ID BIGINT NOT NULL,
Aadhaar BIGINT NOT NULL,
License BIGINT UNIQUE,
first_name CHAR(255) NOT NULL,
last_name CHAR(255) NOT NULL,
start_date DATE NOT NULL,
end_date DATE,
role CHAR(255) NOT NULL,
salary BIGINT NOT NULL,
phone_number BIGINT NOT NULL,
date_of_birth DATE NOT NULL,
PRIMARY KEY (ID)
);

CREATE TABLE Medicine (
drug_name CHAR(255) NOT NULL,
batch_number BIGINT NOT NULL,
MedicineType CHAR(255) NOT NULL,
Manufacturer CHAR(255) NOT NULL,
stock_quantity BIGINT NOT NULL,
expiry_date DATE NOT NULL,
Price BIGINT NOT NULL,
PRIMARY KEY (drug_name,batch_number)
);

CREATE TABLE Bill (
order_id BIGINT NOT NULL,
Customer_Aadhaar BIGINT NOT NULL,
total_amount BIGINT NOT NULL,
customer_payment BIGINT NOT NULL,
insurance_payment BIGINT NOT NULL,
PRIMARY KEY (order_id,Customer_Aadhaar)
);

ALTER TABLE Bill
ADD CONSTRAINT makes FOREIGN KEY (order_id) REFERENCES Orders (order_id

ALTER TABLE Bill
ADD CONSTRAINT pays FOREIGN KEY (Customer_Aadhaar)
REFERENCES Customer (Aadhaar);
```

Populating the Database

Join Queries

Showcase at least 4 join queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

Aggregate Functions

Showcase at least 4 Aggregate function queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

Set Operations

Showcase at least 4 Set Operations queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

Functions and Procedures

Create a Function and Procedure. State the objective of the function / Procedure. Run and display the results.

Triggers and Cursors

Create a Trigger and a Cursor. State the objective. Run and display the results.

Developing a Frontend

The frontend should support

1. Addition, Modification and Deletion of records from any chosen table
2. There should be an window to accept and run any SQL statement and display the result