OptiMed Pharma

Project Report

Group 15
Student1: Gnanasudharsan Ashokumar
Student2: Meena Periasamy

(857)-540-9994
(959)-237-0006

ashokumar.g@northeastern.edu
periasamy.m@northeastern.edu

Percentage of Effort Contributed by Student1: 50

Percentage of Effort Contributed by Student2: 50

Signature of Student 1: Gnanasudharsan

Signature of Student 2: Meena

Submission Date: 12/08/2024

OptiMed Pharma

Meena Periasamy and **Gnanasudharsan Ashokumar**

I. PROBLEM STATEMENT:

In today's pharmaceutical industry, efficient business management is critical for stores to meet the needs of consumers. Pharmacies must manage inventory, prescriptions, customer relationships, and regulatory compliance to ensure a consistent supply of medications, timely delivery, and excellent service. Utilizing digital health solutions can help pharmacies with smooth-running of operations, increase customer engagement, and optimize supply chain management. It is essential for the pharmacies to digitalize their data for overall efficiency and increase customer satisfaction.

THEORY FOR PHARMACEUTICAL STORE BUSINESS REQUIREMENTS:

A pharmaceutical store has a complex and detailed supply chain of various medical products, to ensure regulatory compliance and deliver high-quality service to customers. The database must maintain accurate records of inventory, manage prescriptions records, and provide reliable service to everyone. Profit of the pharma comes from the difference between the selling price and their wholesale cost of products. Success depends on efficient inventory management, pricing strategies, and customer relationship management (CRM).

THE COMPANY MUST RECORD THE FOLLOWING:

Inventory Management: Each product in stock will be assigned a unique product ID, name, and quantity. It is essential to record the wholesale price, retail price, expiration dates, purchase order ID and supplier details for each product to ensure accurate tracking and regulatory compliance.

Supplier Management: For suppliers, the company will store the supplier's ID, name, address, contact information, purchase order ID to ensure efficient supply chain operations. Each product must be linked to its supplier through purchase ID.

Customer Management: The store will monitor customer information including customer ID, name, age, contact details along with the prescription history.

Prescription Management: The store will keep track of every prescription filled, including prescription ID, doctor's ID, drug name, and dosage. Prescription verification by the employees of the pharma and compliance with local regulation are crucial for each transaction.

Sales Tracking: Every sale transaction will be recorded with a sales order ID, product name, quantity, and selling price.

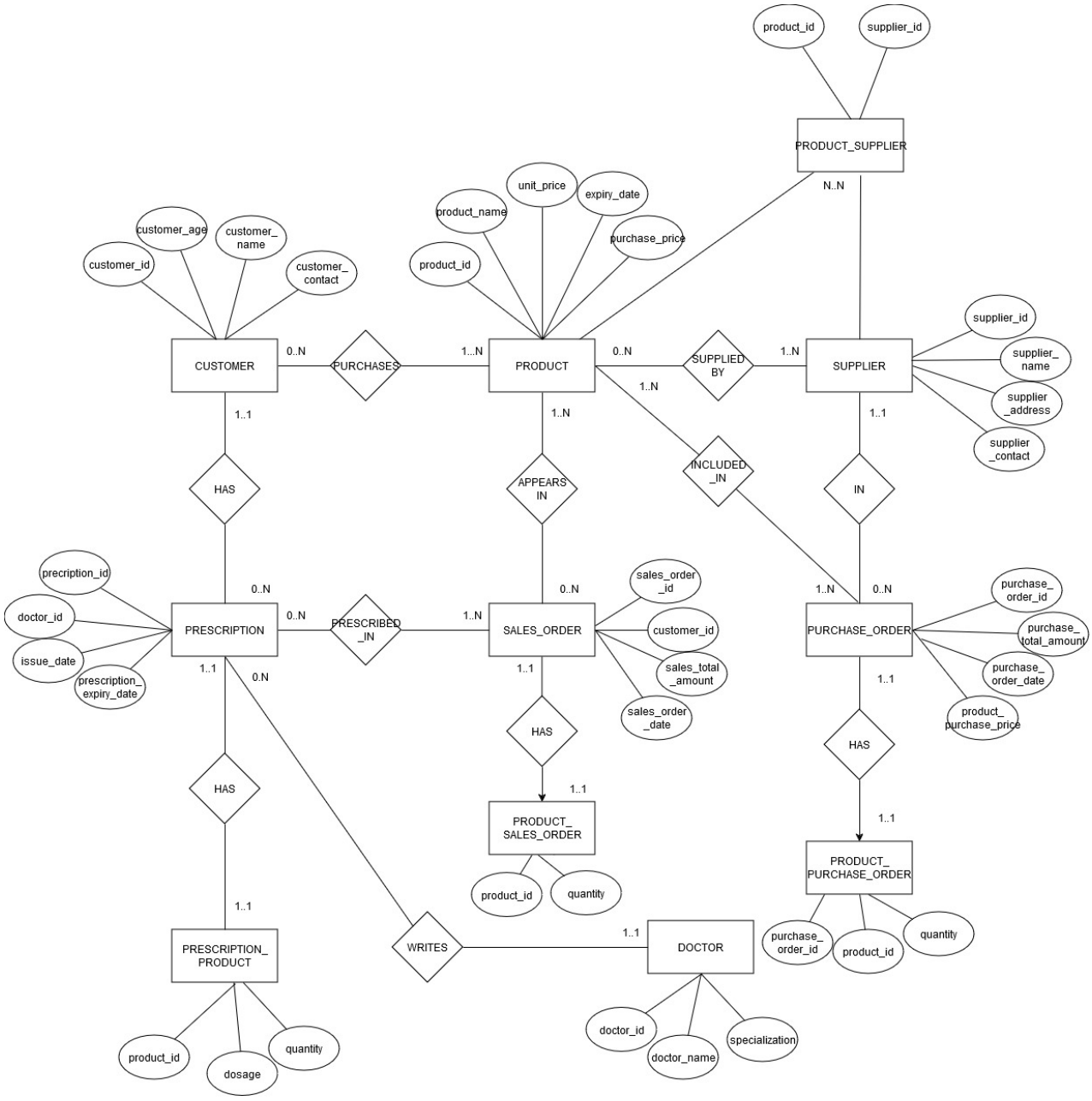
Compliance and Regulatory Management: The store should ensure that all transaction, especially prescription drug sales, comply with healthcare regulations. This protocol requires logging and reporting on the sale of all controlled substance through accurate record-keeping and managing customer prescription refills according to legal standards.

BUSINESS REQUIREMENTS:

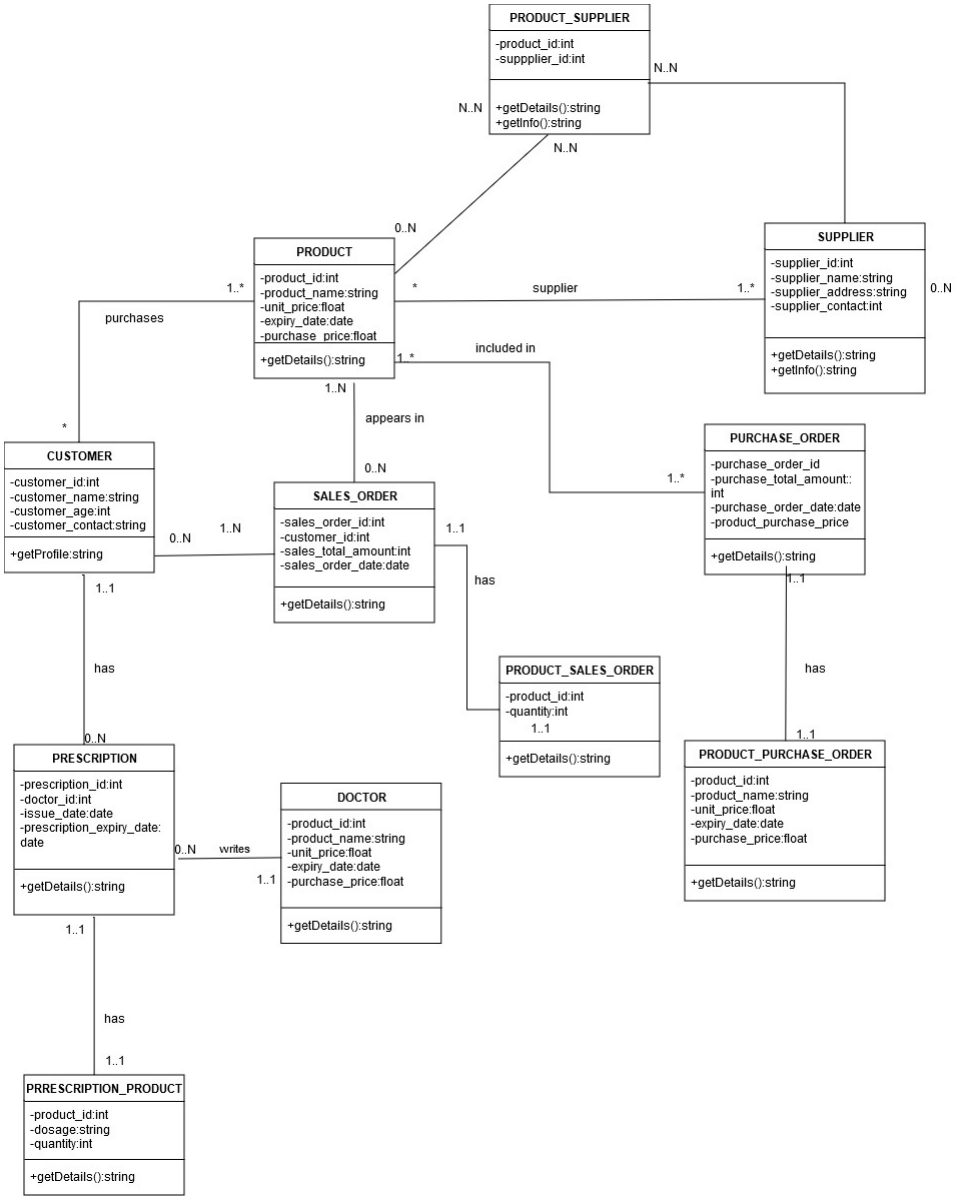
1) A product can have zero to infinite prescriptions associated with it; A prescription links to at least one product.
2) A customer can have zero to infinite prescriptions; a prescription must belong to exactly one customer.
3) A customer can purchase one to infinite products; a product can be purchased by multiple customers.
4) A product can have multiple but at least one supplier; a supplier must supply at least one but can supply multiple products.
5) An order can contain multiple but at least one product; a product can appear in multiple order
6) A prescription can be linked to multiple products but at least one (e.g., in combination therapies); a product can belong to multiple prescriptions.
7) Each prescription must be linked to exactly one doctor; a doctor can write zero to infinite prescriptions for various customers.
8) Each product can have multiple purchase orders. A purchase order must belong to at least one product.
9) Each sales order must contain at least one product; a product can be included in multiple sales orders.

II. CONCEPTUAL MODELING

1. EER DIAGRAM



2. UML CLASS DIAGRAM



III. MAPPING CONCEPTUAL MODEL TO RELATIONAL MODEL:

1. PRODUCT

- **PRODUCT**(product_id, product_name, purchase_price, unit_price, expiry_date)
- **PK**: product_id, **NOT NULL**

2.SUPPLIER

- **SUPPLIER**(supplier_id, supplier_name, supplier_address, supplier_contact)
- **PK**: supplier_id, **NOT NULL**

3.CUSTOMER

- **CUSTOMER**(customer_id, customer_name, customer_age, customer_contact)
- **PK**: customer_id, **NOT NULL**

4.DOCTOR

- **DOCTOR**(doctor_id, doctor_name, specialization)
- **PK**: doctor_id, **NOT NULL**

5.PRESCRIPTION

- **PRESCRIPTION**(prescription_id, customer_id, doctor_id, issue_date, prescription_expiry_date)
- **PK**: prescription_id, **NOT NULL**
- **FK**: customer_id, references **CUSTOMER**(customer_id), **NOT NULL**
- **FK**: doctor_id, references **DOCTOR**(doctor_id), **NOT NULL**

6.PRESCRIPTION_PRODUCT

- **PRESCRIPTION_PRODUCT**(prescription_id, product_id, dosage, quantity)
- **PK:** (prescription_id, product_id), **NOT NULL**
- **FK:** prescription_id, references PRESCRIPTION(prescription_id), **NOT NULL**
- **FK:** product_id, references PRODUCT(product_id), **NOT NULL**

7.SALES_ORDER

- **SALES_ORDER**(sales_order_id, customer_id, sales_total_amount, sales_order_date)
- **PK**: sales_order_id, **NOT NULL**
- **FK**: customer_id, references **CUSTOMER**(customer_id), **NOT NULL**

8.PRODUCT_SALESORDER

- **PRODUCT_SALESORDER**(sales_order_id, product_id, quantity)
- **PK**: (sales_order_id, product_id), **NOT NULL**
- **FK**: sales_order_id, references **SALES_ORDER**(sales_order_id), **NOT NULL**
- **FK**: product_id, references **PRODUCT**(product_id), **NOT NULL**

9.PURCHASE_ORDER

- **PURCHASE_ORDER**(purchase_order_id, supplier_id, purchase_total_amount, purchase_order_date,)
- **PK**: purchase_order_id, **NOT NULL**
- **FK**: supplier_id, references **SUPPLIER**(supplier_id), **NOT NULL**

10.PRODUCT_PURCHASE_ORDER

- **PRODUCT_PURCHASE_ORDER**(purchase_order_id, product_id, quantity, product_purchase_price)
- **PK**: (purchase_order_id, product_id), **NOT NULL**
- **FK**: purchase_order_id, references **PURCHASE_ORDER**(purchase_order_id), **NOT NULL**
- **FK**: product_id, references **PRODUCT**(product_id), **NOT NULL**

11.PRODUCT_SUPPLIER

- **PRODUCT_SUPPLIER** (product_id, supplier_id)
- **PK**: (product_id, supplier_id), **NOT NULL**
- **FK**: product_id, references **PRODUCT**(product_id), **NOT NULL**
- **FK**: supplier_id, references **SUPPLIER**(supplier_id), **NOT NULL**

IV. IMPLEMENTATAION OF RELATION MODEL VIA MYSQL AND NOSQL

SQL QUERIES:

1. Simple Query - Retrieve the names of all customers who are older than 30 years.

```
 SELECT
customer_name,
    customer_age
FROM
    opti_med.customer
WHERE
    customer_age > 30;
```

customer_name	customer_age
John Smith	32
Robert Brown	45
Patricia Davis	53
Linda Martinez	40
William Anderson	36
David Taylor	47
Barbara Harris	34
Jennifer Lewis	30

2. Aggregate Query - Calculate the average age of customers.

```
SELECT
    AVG(customer_age) AS average_age
FROM
    opti_med.customer;
```

average_age
38.4875

3. Inner Join - List all prescriptions with customer names and doctor names.

```
SELECT
    p.prescription_id,
    c.customer_name,
    d.doctor_name,
    p.issue_date,
    p.prescription_expiry_date
FROM
    opti_med.prescription p
INNER JOIN
    opti_med.customer c
ON
    p.customer_id = c.customer_id
INNER JOIN
    opti_med.doctor d
ON
    p.doctor_id = d.doctor_id;
```

prescription_id	customer_name	doctor_name	issue_date	prescription_expiry_date
1	John Smith	Dr. Michael Wilson	2024-01-10	2024-07-10
2	Mary Johnson	Dr. Barbara Harris	2024-02-14	2024-08-14
3	Robert Brown	Dr. Joseph Hall	2024-03-01	2024-09-01
4	Patricia Davis	Dr. Karen Wright	2024-03-15	2024-09-15
5	Michael Wilson	Dr. Anthony Adams	2024-04-05	2024-10-05
6	Linda Martinez	Dr. Carol Roberts	2024-04-20	2024-10-20
7	William Anderson	Dr. Joshua Evans	2024-05-01	2024-11-01
8	Elizabeth Thomas	Dr. Michelle Morris	2024-05-10	2024-11-10

4. Nested Query - Find the names of customers who have a prescription issued by a doctor specializing in 'Cardiology'.

```
SELECT
    customer_name
FROM
    opti_med.customer
WHERE
    customer_id IN (
        SELECT
            p.customer_id
```

customer_name
James Clark

```
    FROM
        opti_med.prescription p
    JOIN
        opti_med.doctor d
    ON
        p.doctor_id = d.doctor_id
    WHERE
        d.specialization = 'Cardiology'
);
```

5. Correlated Query - Find the products whose unit price is higher than the average unit price.

```
SELECT
    product_name,
    unit_price
FROM
    opti_med.product p1
WHERE
    unit_price > (
        SELECT
            AVG(p2.unit_price)
        FROM
            opti_med.product p2
        WHERE
            p2.product_id != p1.product_id
    );
```

product_name	unit_price
Amoxicillin	12.00
Insulin	50.00
Albuterol Inhaler	35.00
Warfarin	15.00
Azithromycin	18.00
Codeine	14.00
Fluoxetine	13.00

6. >=ALL/ >ANY/ EXISTS - List the names of doctors who have issued at least one prescription.

```
SELECT
    doctor_name
FROM
    opti_med.doctor d
WHERE
    EXISTS (
        SELECT
            1
        FROM
            opti_med.prescription p
        WHERE
            d.doctor_id = p.doctor_id
    );
```

doctor_name
Dr. John Smith
Dr. Mary Johnson
Dr. Robert Brown
Dr. Patricia Davis
Dr. Michael Wilson
Dr. Linda Martinez
Dr. William Anderson

7. Set Operations (UNION) - List all product names that are either purchased or prescribed (no duplicates).

```
SELECT
    p.product_name
FROM
    opti_med.product p
JOIN
    opti_med.product_purchase_order ppo
```

```
ON
   p.product_id = ppo.product_id
UNION
SELECT
   p.product_name
FROM
   opti_med.product p
JOIN
   opti_med.prescription_product pp
ON
   p.product_id = pp.product_id;
```

product_name
Aspirin
Ibuprofen
Amoxicillin
Cough Syrup
Metformin
Vitamin C
Paracetamol
Antibiotic Cream

8. Subqueries in SELECT Clause - Display the total quantity of each product in stock along with its name.

```
SELECT
   p.product_name,
   (SELECT
      SUM(quantity)
    FROM
      opti_med.product_purchase_order ppo
    WHERE
      ppo.product_id = p.product_id
   ) AS total_quantity
FROM
   opti_med.product p;
```

product_name	total_quantity
Aspirin	47
Ibuprofen	72
Amoxicillin	74
Cough Syrup	65
Metformin	54
Vitamin C	74
Paracetamol	76
Antibiotic Cream	73

9. Subqueries in FROM Clause - Find the average quantity of products sold per sales order.

```
SELECT
   AVG(quantity_per_order) AS average_quantity
FROM
   (
      SELECT
         so.sales_order_id,
         SUM(pso.quantity) AS quantity_per_order
      FROM
         opti_med.sales_order so
      JOIN
         opti_med.product_sales_order pso
      ON
         so.sales_order_id = pso.sales_order_id
      GROUP BY
         so.sales_order_id
   ) AS subquery;
```

average_quantity
6.6600

10. Outer Join - List all doctors along with the number of prescriptions they have issued (include doctors with no prescriptions).

```
SELECT
   d.doctor_name,
   d.specialization,
   COUNT(p.prescription_id) AS total_prescriptions
FROM
```

```
    opti_med.doctor d
LEFT OUTER JOIN
    opti_med.prescription p
ON
    d.doctor_id = p.doctor_id
GROUP BY
    d.doctor_id;
```

	doctor_name	specialization	total_prescriptions
▶	Dr. John Smith	Cardiology	1
	Dr. Mary Johnson	Pediatrics	1
	Dr. Robert Brown	Orthopedics	1
	Dr. Patricia Davis	Dermatology	1
	Dr. Michael Wilson	Neurology	1
	Dr. Linda Martinez	Gynecology	1
	Dr. William Anderson	Ophthalmology	1
	Dr. Elizabeth Thomas	Oncology	1

NOSQL IMPLEMENTATION(MONGODB):

1.Query to retrieve all supplier

db.suppliers.find({});


```
{
  _id: ObjectId('674cc6f330da26fdf07b2ac5'),
  supplier_id: 15,
  supplier_name: 'Prime Care Supply',
  supplier_address: '72 Pine Dr, Detroit, MI',
  supplier_contact: '313-555-1401'
}
{
  _id: ObjectId('674cc6f330da26fdf07b2ac6'),
  supplier_id: 16,
  supplier_name: 'WellCare Pharma',
  supplier_address: '29 Beech Rd, Nashville, TN',
  supplier_contact: '615-555-1501'
}
{
  _id: ObjectId('674cc6f330da26fdf07b2ac7'),
  supplier_id: 17,
  supplier_name: 'Unity Medical Supply',
  supplier_address: '56 Cypress Ln, San Diego, CA',
  supplier_contact: '619-555-1601'
}
{
  _id: ObjectId('674cc6f330da26fdf07b2ac8'),
  supplier_id: 18,
  supplier_name: 'Bright Life Distributors',
  supplier_address: '80 Poplar St, San Francisco, CA',
  supplier_contact: '415-555-1701'
}
{
  _id: ObjectId('674cc6f330da26fdf07b2ac9'),
  supplier_id: 19,
  supplier_name: 'MediLink Pharma',
  supplier_address: '34 Willow Ave, Austin, TX',
  supplier_contact: '512-555-1801'
}
{
  _id: ObjectId('674cc6f330da26fdf07b2aca'),
  supplier_id: 20,
  supplier_name: 'QuickMed Supplies',
  supplier_address: '75 Cherry St, Columbus, OH',
  supplier_contact: '614-555-1901'
}
Type "it" for more
```

2.Inventory Status Report

"Retrieve all products along with their stock quantity, total sales volume, calculated reorder point (30% of stock quantity), and a flag indicating if the product needs to be reordered (if stock is below the reorder point)."

```
db.products.aggregate([
{
$lookup: {
from: "product_sales_order",
localField: "_id",
foreignField: "product_id",
as: "sales_data"
}
},
{
$project: {
product_name: 1,
stock_quantity: 1,
sales_volume: { $sum: "$sales_data.quantity" },
reorder_point: { $multiply: ["$stock_quantity", 0.3] }
}
},
{
$addFields: {
needs_reordering: { $lt: ["$stock_quantity", "$reorder_point"]
}
}
}
]);
```

3.Identify all doctors who have issued prescriptions that include both "Aspirin" and "Ibuprofen" to any customer.

```
db.prescriptions.aggregate([
{
$lookup: {
from: "prescription_product",
localField: "prescription_id",
foreignField: "prescription_id",
as: "prescription_details"
}
},
{
$unwind: "$prescription_details"
},
{
$lookup: {
from: "products",
localField: "prescription_details.product_id",
foreignField: "product_id",
as: "product_details"
}
},
{
$unwind: "$product_details"
},
{
$match: {
"product_details.product_name": { $in: ["Aspirin", "Ibuprofen"] }
}
},
{
$group: {
_id: "$doctor_id",
products: { $addToSet: "$product_details.product_name" }
}
},
{
$match: {
products: { $all: ["Aspirin", "Ibuprofen"] }
}
},
{
$lookup: {
from: "doctors",
localField: "_id",
foreignField: "doctor_id",
as: "doctor_details"
```

```
}
},
{
$project: {
doctor_name: "$doctor_details.doctor_name",
products: 1
}
}
])
```

V. IMPLEMENTATION OF APPLICATION USING PYTHON:

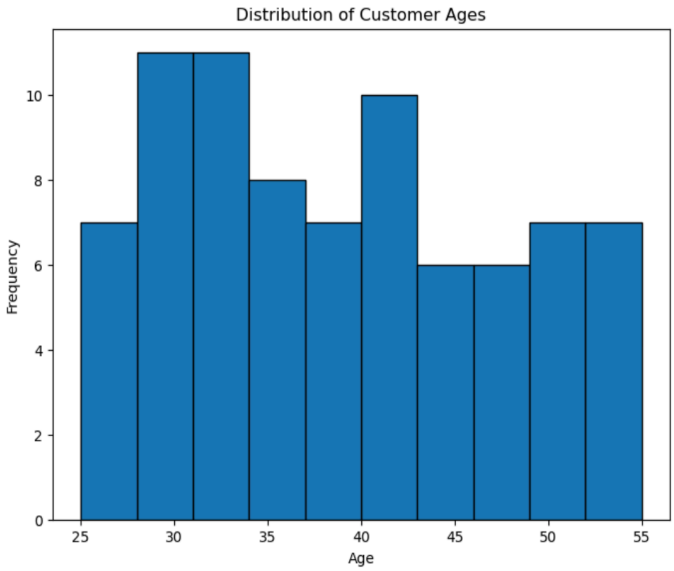


Fig. Histogram of Customer Ages

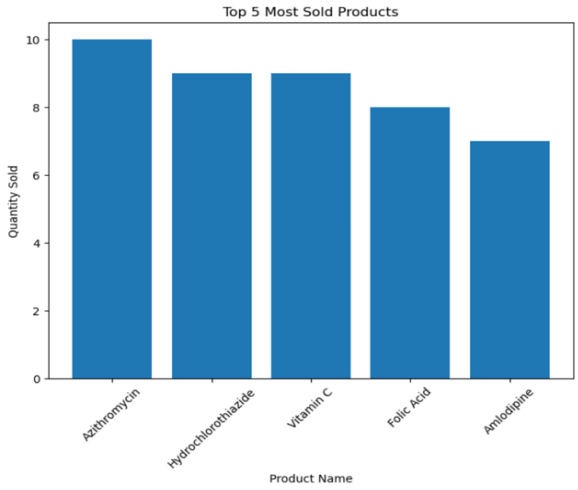


Fig. Bar Chart of Top 5 Most Sold Products

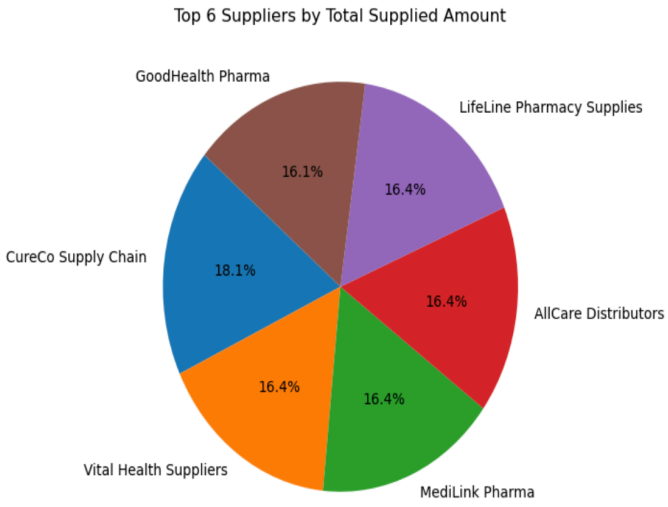


Fig. Pie Chart of Top 6 Suppliers

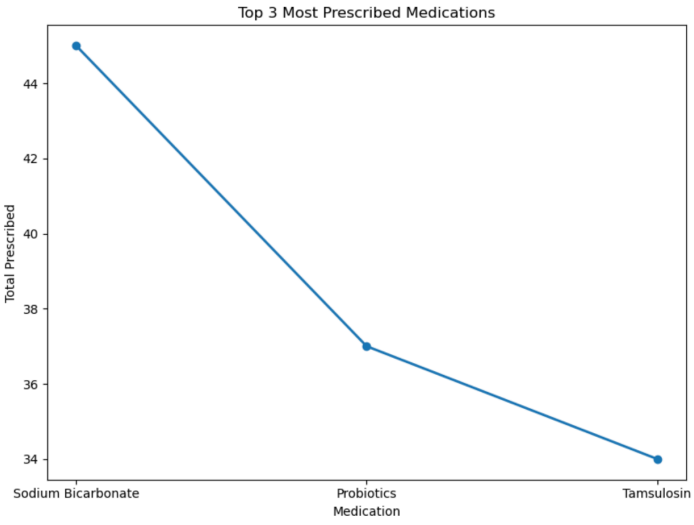


Fig. Line Chart of Top 3 Prescribed Medications

USERINTERFACE CREATED USING PYTHON:

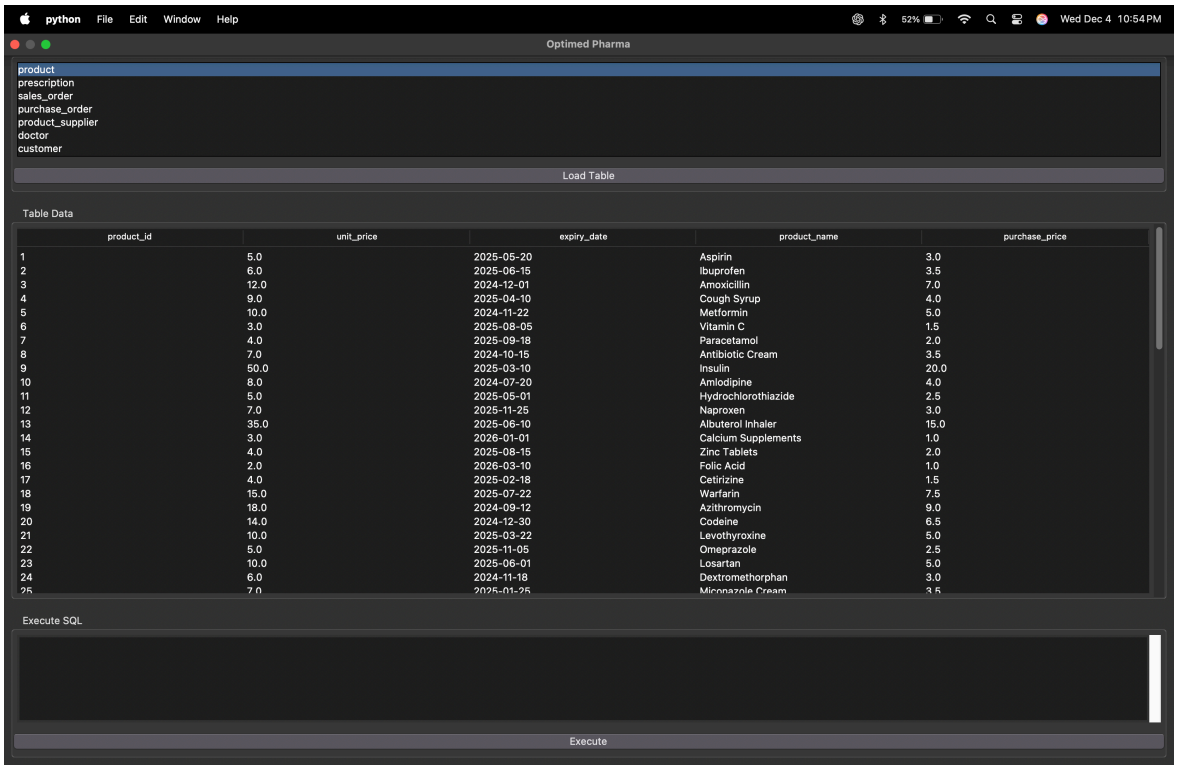


Fig. Application Interface using Python showing Table.

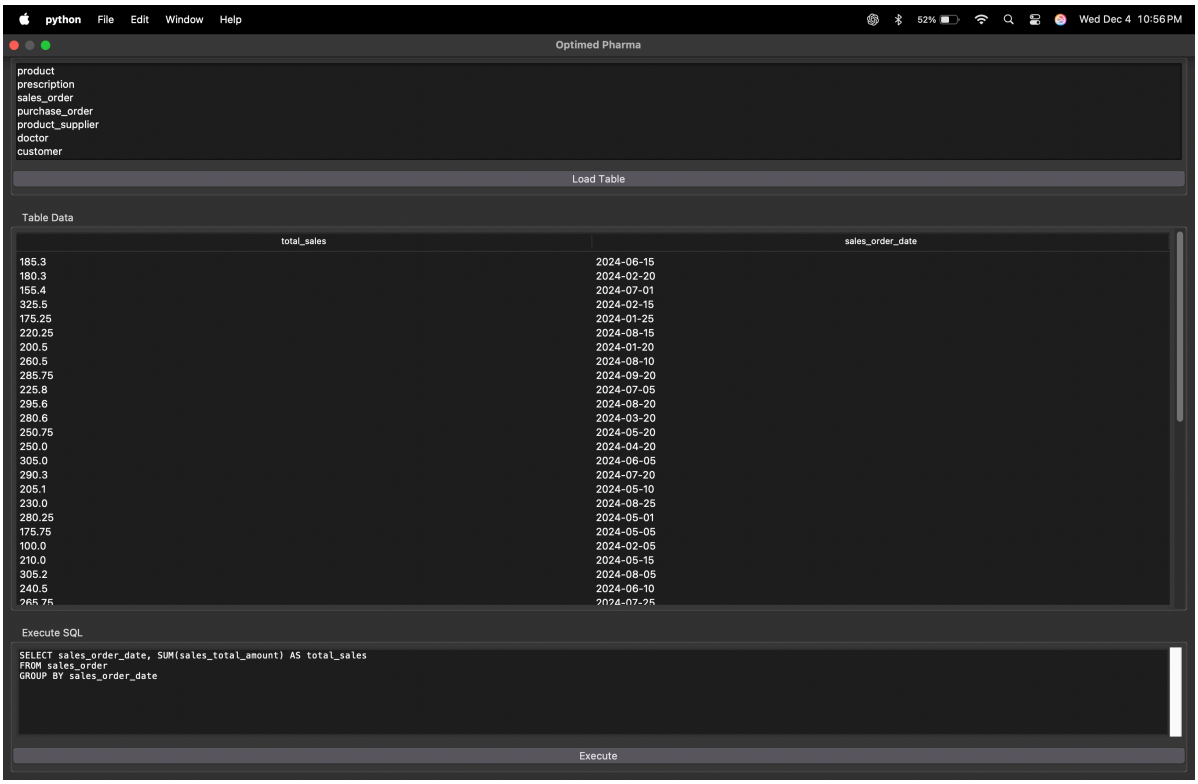


Fig. Executing SQL queries via created python application

SUMMARY:

The OptiMed Pharma Project is a comprehensive data management system tailored for the pharmaceutical industry, designed to streamline operations, enhance efficiency, and improve customer satisfaction. This hybrid database solution integrates relational data management via MySQL and NoSQL capabilities via MongoDB to address the multifaceted challenges faced by modern pharmacies. By digitizing key processes, such as inventory management, prescription tracking, customer relationships, and regulatory compliance, the project ensures seamless operations across the supply chain.

Key features of the system include real-time inventory tracking, supplier and sales management, customer engagement strategies, and detailed compliance tracking for prescription medications. Analytical capabilities were demonstrated through Python-based data visualizations, including customer demographic trends, product performance, and supplier contributions, alongside MongoDB aggregation queries for advanced inventory analysis and prescription insights.
Recommendations for Enhancement:

To further elevate the system, several enhancements can be implemented:

- Advanced Predictive Analytics: Integrate machine learning models to optimize inventory levels, predict sales trends, and improve customer retention.
- Improved Data Governance: Employ automated tools and APIs to validate supplier and customer information for enhanced data quality and consistency.
- Scalability Enhancements: Expand NoSQL capabilities to handle larger datasets, including real-time product demand forecasting and multimedia prescription records.
- User Experience Refinement: Develop an intuitive front-end interface for pharmacy staff to improve accessibility and reduce operational complexity.
- Global Standards Compliance: Align the system with international healthcare and pharmaceutical standards to support broader market adaptability.

The OptiMed Pharma Project has demonstrated its potential in transforming pharmacy operations, providing a scalable and efficient solution for the modern pharmaceutical industry.