

USE CASE STUDY REPORT

Group No: Group 7

Student Names: Deepshikha Soni and Chandini Nekkanti

Executive Summary:

The objective of this project is to enhance the functionality of the Medimart Database, specifically focusing on optimizing the overall management system for Medimart, a pharmaceutical and healthcare product provider. The primary aim is to develop an advanced system that seamlessly integrates data from various facets of the business, streamlining operations and ensuring a more efficient workflow.

The project encompasses the enhancement of Medimart's database to address key functionalities, including inventory management, order processing, customer data management, supplier relationships, sales tracking, employee management, and the integration of consultant doctors. The data modeling process involves establishing robust relationships between products, suppliers, customers, orders, employees, and medical professionals within the Medimart ecosystem.

The implementation phase will involve the creation of a user-friendly web-based application that empowers both internal stakeholders and external customers. Users will be able to effortlessly navigate through functionalities such as placing and tracking orders, managing inventory, accessing customer information, handling supplier interactions, and consulting with available medical professionals. Additionally, administrative tools will be integrated to facilitate efficient inventory control, order fulfillment, customer relationship management, and employee supervision.

The anticipated outcomes of this project include a significant improvement in operational efficiency, heightened customer satisfaction, increased accuracy in inventory management, streamlined order processing, enhanced decision-making capabilities for Medimart, and improved access to healthcare advice through the integration of consultant doctors. By automating manual processes, reducing errors, and providing comprehensive data management tools, the enhanced system aims to elevate the overall performance of Medimart's operations, ultimately contributing to the company's success in the pharmaceutical and healthcare product industry.

I. Introduction

The project centers around the development of an advanced database management system tailored for Medimart, a leading provider in the pharmaceutical and healthcare product industry. Medimart's operations encompass a diverse range of functions, including inventory management, order processing, customer relations, supplier interactions, sales tracking, employee management, and the integration of consultant doctors.

In this comprehensive database enhancement, the goal is to seamlessly integrate and optimize data from various dimensions of Medimart's business, fostering a cohesive and efficient workflow. The system's overarching objectives include improving operational efficiency, enhancing customer satisfaction, increasing accuracy in inventory management, streamlining order processing, and providing robust decision-making capabilities for Medimart.

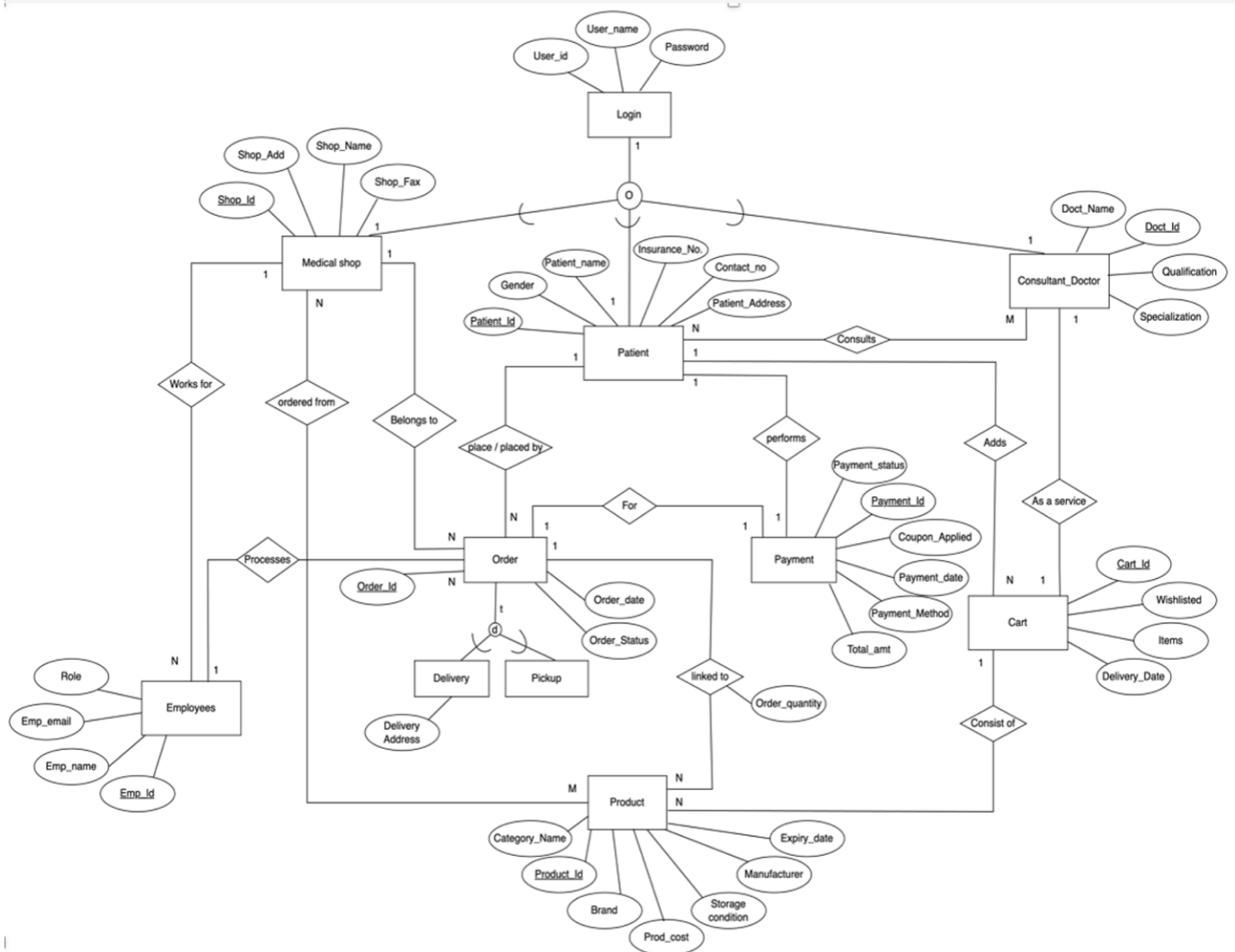
The scope of the project extends beyond traditional pharmaceutical management, acknowledging the role of consultant doctors within the system. The integration of medical professionals allows customers to access healthcare advice and consultations, creating a holistic platform for both product provision and medical guidance. This addition enhances the overall value proposition of Medimart, aligning the company with contemporary healthcare needs.

By leveraging modern technologies and database management practices, the project aims to automate manual processes, reduce errors, and offer comprehensive data management tools. The anticipated outcomes include a notable uplift in operational efficiency, improved customer experience, precise inventory control, seamless order fulfillment, and elevated performance across various facets of Medimart's operations.

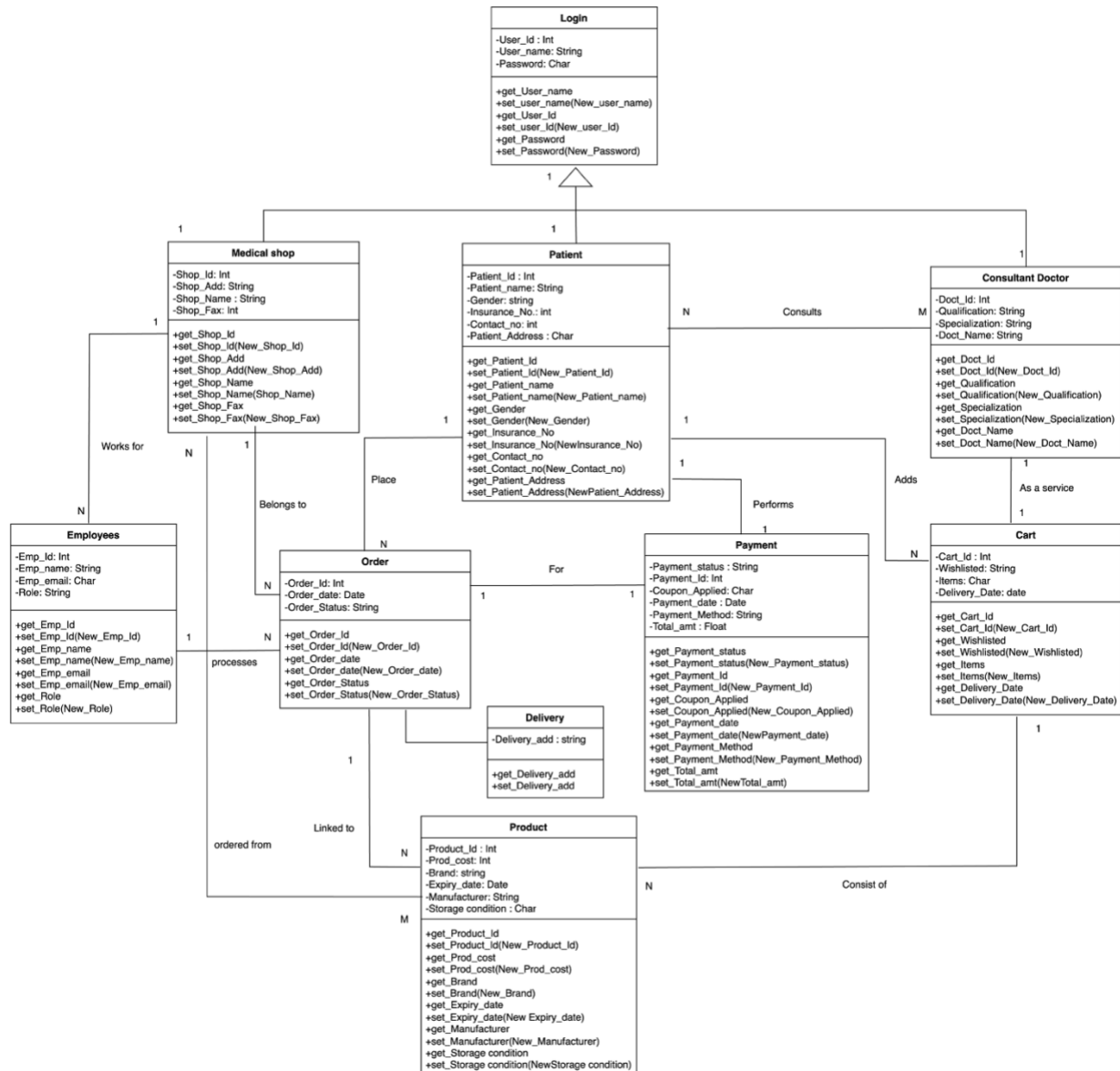
Through this project, we endeavor to not only meet but exceed the evolving expectations of the pharmaceutical and healthcare product industry, positioning Medimart as a frontrunner in delivering both quality products and valuable healthcare insights.

II. Conceptual Data Modeling

1. EER Diagram



2. UML Diagram



III. Mapping Conceptual Model to Relational Model

1. Login (User_ID, User_Name, Password)
Primary Key: User_ID , Null not allowed
2. Medical Shop (Shop_ID, Shop_add, shop_name, shop_fax, login_ID)
Primary Key: Shop_ID, Foreign Key: login_ID refers to primary key of relation login, Null not allowed.
3. Consultant_Doctor (Doc_ID, Doc_Name, Qualification, Specialization, login_ID)

- Primary Key: Doc_ID, Foreign Key: login_ID refers to primary key of relation login, Null not allowed
4. Employees (Emp_ID, Emp_Name, Emp_Email, Role, Shop_ID)
Primary Key: Emp_ID , Foreign Key: Shop_ID refers to primary key of relation Medical Shop, Null not allowed
 5. Delivery (Delivery Address, Order_ID, order_date, order_status, shop_ID, Patient_ID)
Foreign Key: Order_ID refers to primary key of relation Order, Shop_ID refers to primary key of relation Medical Shop, Pateint_ID refers to primary key of relation Patient, Null not allowed
 6. Pickup (Order_ID, order_date, order_status, shop_ID, Patient_ID)
Foreign Key: Order_ID refers to primary key of relation Order, Shop_ID refers to primary key of relation Medical Shop, Pateint_ID refers to primary key of relation Patient, Null not allowed
 7. Patient (Patient_ID, gender, Patient_Name, Insurance_No, Contact_No, Patient_address, shop_ID, login_ID, Doc_ID)
Primary Key: Patient_ID
Foreign Key: login_ID refers to primary key of relation Login, Shop_ID refers to primary key of relation Medical Shop, Doc_ID refers to primary key of relation Consultant Doctor, Null not allowed
 8. Payment (Payment_ID, payment_status, coupon_applied, payment_date, payment_metod, total_amt, order_id, Patient_ID)
Primary Key: Payment_ID
Foreign Key: Order_ID refers to primary key of relation Order, Pateint_ID refers to primary key of relation Patient, Null not allowed
 9. Cart (cart_ID, whislisted, items, Delivery_date, Patient_ID, Doc_ID)
Primary Key: cart_ID
Foreign Key: Pateint_ID refers to primary key of relation Patient, Doc_ID refers to primary key of relation Consultant Doctor, Null not allowed
 10. Product (product_ID, category name, brand, prod_cost, storage condition, manufacturer, Expiry_date, order quantity, Cart_ID, Order_ID)
Primary Key: Product_ID
Foreign Key: cart_ID refers to primary key of relation cart, Order_ID refers to primary key of relation Order, Null not allowed
 11. Ordered_from (Shop_ID, Product_ID)
Foreign Key: Shop_ID refers to primary key of relation Medical Shop, Product_ID refers to primary key of relation Product, Null not allowed
 12. Consults (patient_ID, Doc_ID)
Foreign Key: Pateint_ID refers to primary key of relation Patient, Doc_ID refers to primary key of relation Consultant Doctor , Null not allowed

IV. Implementation of Relation Model via MySQL and NoSQL

MySQL Implementation:

The database was created in MySQL and the following queries were performed:

1. Retrieve the count of products in each category:

```
SELECT p.Category_Name,
COUNT(p.Product_ID) AS ProductCount
FROM Product p GROUP BY
p.Category_Name;
```

Category_Name	ProductCount	
Antibiotics	27	
Vitamins	19	
Painkillers	14	

2. Retrieve the top 10 best-selling products:

```
SELECT p.Product_ID, p.Category_Name, p.manufacturer,
SUM(p.Order_Quantity) AS TotalQuantitySold
FROM Product p
JOIN Ordered_from od ON p.Product_ID = od.Product_ID
GROUP BY p.Product_ID, p.Category_Name, p.manufacturer
ORDER BY TotalQuantitySold DESC
LIMIT 10;
```

Product_ID	Category_Name	manufacturer	TotalQuantitySold
T82858S	Antibiotics	Uriel Pharmacy Inc.	564
B509	Antibiotics	Physicians Total Care, Inc.	285
S72366C	Antibiotics	A&Z Pharmaceutical, Inc.	120
V4311XS	Antibiotics	Walgreen Company	93
S52325Q	Antibiotics	REMEDYREPACK INC.	93
M9722	Painkillers	Nelco Laboratories, Inc.	92
S02411G	Antibiotics	Wegman Food Markets Inc	90
T17910	Vitamins	Cantrell Drug Company	83
T24599	Antibiotics	Otsuka America Pharmaceutical, Inc.	83
S233XXS	Antibiotics	Dolgenercorp, LLC	82

3. Retrieve orders with product details and corresponding patient information:

```
SELECT d.Order_ID, d.Order_Date, p.Product_ID, p.Order_Quantity,
p.Prod_cost, pa.Patient_ID, pa.Patient_Name, pa.Insurance_No
FROM delivery d
JOIN Patient pa ON d.Patient_ID = pa.Patient_ID
JOIN Product p ON d.order_ID = p.order_ID;
```

Order_ID	Order_Date	Product_ID	Order_Quantity	Prod_cost	Patient_ID	Patient_Name	Insurance_No
H7101	8/16/23	S3217XB	49	\$2,231.66	S82854J	Kipp	M80062D
M84759S	5/31/23	T17910	83	\$2,420.70	A59	Duncan	S72101B
H7101	8/16/23	S01101A	25	\$2,254.35	S82854J	Kipp	M80062D
H7101	8/16/23	S53441A	80	\$2,969.92	S82854J	Kipp	M80062D
S82822K	11/1/23	S72136B	73	\$3,723.32	S79002G	Shelba	M216X9
V55	3/23/23	Q6111	25	\$2,730.83	Q30	Win	Y00XXXS
V272	6/15/23	O6013X1	78	\$1,162.66	T23221D	Rani	S62347B
M21961	8/19/23	S59002A	10	\$1,670.21	S5609	Tove	S72462H
O9984	6/17/23	T485X5D	26	\$3,135.25	S86011D	Caleb	T22099
S92065A	8/10/23	S63112	76	\$3,009.72	S96802D	Elysia	T23102A
M84759S	5/31/23	I83218	73	\$4,266.27	A59	Duncan	S72101B
T20419A	1/30/23	V404	77	\$692.98	Z818	Roch	T3130
H7101	8/16/23	S52325Q	93	\$2,696.04	S82854J	Kipp	M80062D
S12650G	5/6/23	V5959	26	\$409.54	S52309	Darleen	S83401
V9226XS	1/29/23	V712	38	\$4,191.92	S96802D	Elysia	T23102A
M84759S	5/31/23	K958	72	\$4,943.42	A59	Duncan	S72101B
S93504S	8/10/23	Q671	79	\$458.84	H26049	Bradford	S42447D
T24402	3/23/23	T85310A	62	\$3,999.07	Q30	Win	Y00XXXS
Z053	7/13/23	T82858S	94	\$2,800.36	T471X	Murvyn	V169
S638X2	3/19/23	S233XXS	82	\$1,480.67	W69XXD	Robinson	T438X1A
V0210XA	5/26/23	T56813D	56	\$1,868.76	Q30	Win	Y00XXXS
S63423A	11/17/22	S82012J	32	\$2,960.49	V7988XD	Trudi	O3101X2
S52242M	12/23/22	V0402	91	\$3,758.53	S96802D	Elysia	T23102A

4. Retrieve orders placed by patients whose payment status is failed or pending:

```

SELECT d.Order_ID, d.Order_Date,
d.Patient_ID, pa.Patient_name,
p.Payment_Status
FROM delivery d
JOIN patient pa ON d.Patient_ID =
pa.Patient_ID
JOIN Payment p ON d.Order_ID = p.Order_ID
WHERE EXISTS (
    SELECT 1
    FROM Payment pay
    WHERE d.Order_ID = pay.Order_ID
    AND pay.Payment_Status IN ('Failed',
'Pending')
);

```

Order_ID	Order_Date	Patient_ID	Patient_name	Payment_Status
S82822K	11/1/23	S79002G	Shelba	Pending
V55	3/23/23	Q30	Win	Failed
V272	6/15/23	T23221D	Rani	Failed
M21961	8/19/23	S5609	Tove	Pending
O9984	6/17/23	S86011D	Caleb	Pending
T881	3/8/23	T382X1	Papageno	Failed
O411425	1/3/23	M85479	Wallas	Failed
T20419A	1/30/23	Z818	Roch	Failed
H7101	8/16/23	S82854J	Kipp	Pending
S51002D	8/12/23	T2176XS	Doll	Pending
S12650G	5/6/23	S52309	Darleen	Pending
V9226XS	1/29/23	S96802D	Elysia	Failed
T24402	3/23/23	Q30	Win	Failed
T474X5A	4/3/23	T4392	Constance	Failed
Z053	7/13/23	T471X	Murvyn	Pending
S9402	3/28/23	T23139	Pen	Failed
V0210XA	5/26/23	Q30	Win	Failed
S63423A	11/17/22	V7988XD	Trudi	Failed
S52242M	12/23/22	S96802D	Elysia	Pending

5. Retrieve Top 10 Most Recent Orders:

```

SELECT Order_ID, Order_Date, Patient_ID,
shop_id
FROM delivery
ORDER BY Order_Date DESC
LIMIT 10;

```

Order_ID	Order_Date	Patient_ID	shop_id
M21961	8/19/23	S5609	S66517D
H7101	8/16/23	S82854J	H1022
S51002D	8/12/23	T2176XS	S82011P
S92065A	8/10/23	S96802D	S72309F
S93504S	8/10/23	H26049	S63043S
Z053	7/13/23	T471X	T2019XA
O9984	6/17/23	S86011D	S06827S
V272	6/15/23	T23221D	S72019
S12650G	5/6/23	S52309	S66517D
M84759S	5/31/23	A59	S72309F

6. Retrieve the Employee who have processed the Most Orders:

```

SELECT e.Emp_ID, e.Emp_Name, e.Shop_Id,
COUNT(d.Order_ID) AS NumberOfOrders
FROM Employees e
JOIN delivery d ON e.Emp_ID = d.Emp_ID
GROUP BY e.Emp_ID, e.Emp_Name, e.shop_Id
HAVING COUNT(d.Order_ID) = (
    SELECT COUNT(Order_ID) AS MaxOrders
    FROM delivery
    GROUP BY Emp_ID
    ORDER BY MaxOrders DESC
    LIMIT 1);

```

Emp_ID	Emp_Name	Shop_Id	NumberOfOrde...
S0027	Dawna	T83128A	3
S4900	Toinette	O7189	3
S42294	Beverie	V245XXD	3

7. Retrieve the Latest Order for Each Patient:

```

SELECT distinct(d.Patient_ID), d.Shop_ID, p.Product_Id, p.Category_name,
MAX(d.Order_Date) AS LatestOrderDate
FROM delivery d
JOIN Product p ON d.order_id = p.order_ID
GROUP BY Patient_ID, d.Shop_ID, p.Product_Id, p.Category_name;

```

Patient_ID	Shop_ID	Product_Id	Category_name	LatestOrderDate
S82854J	H1022	S3217XB	Antibiotics	8/16/23
A59	S72309F	T17910	Vitamins	5/31/23
S82854J	H1022	S01101A	Painkillers	8/16/23
S82854J	H1022	S53441A	Painkillers	8/16/23
S79002G	S66517D	S72136B	Antibiotics	11/1/23
Q30	S76012A	Q6111	Antibiotics	3/23/23
T23221D	S72019	O6013X1	Vitamins	6/15/23
S5609	S66517D	S59002A	Vitamins	8/19/23
S86011D	S06827S	T485X5D	Antibiotics	6/17/23
S96802D	S72309F	S63112	Antibiotics	8/10/23
A59	S72309F	I83218	Painkillers	5/31/23
Z818	S66517D	V404	Antibiotics	1/30/23
S82854J	H1022	S52325Q	Antibiotics	8/16/23
S52309	S66517D	V5959	Painkillers	5/6/23
S96802D	S72036S	V712	Antibiotics	1/29/23
A59	S72309F	K958	Antibiotics	5/31/23
H26049	S63043S	Q671	Antibiotics	8/10/23
Q30	T486X3	T85310A	Antibiotics	3/23/23
T471X	T2019XA	T82858S	Antibiotics	7/13/23
W69XXD	S66517D	S233XXS	Antibiotics	3/19/23
Q30	H25	T56813D	Vitamins	5/26/23
V7988XD	S66517D	S82012J	Antibiotics	11/17/22
S96802D	S86811D	V0402	Painkillers	12/23/22

8. Retrieve Total Sales Amount for Each Category:

```

SELECT p.Category_Name,
SUM(py.Total_Amt) AS TotalSales
FROM product p
JOIN delivery d ON p.order_ID = d.order_ID
JOIN Payment py ON d.Order_ID = py.Order_ID
GROUP BY p.Category_Name
ORDER BY TotalSales DESC;

```

Category_Name	TotalSales
Antibiotics	47096.189999999995
Painkillers	19223.43
Vitamins	12221.380000000001

9. Retrieve patients that appear in either for pickup or delivery or in both.

```
SELECT patient_id
FROM pickup
UNION
SELECT patient_id
FROM delivery
Limit 10;
```

patient_id
S96802D
Q30
T23221D
S5609
S86011D
T4392
S82202K
G803
M85479
Z818

10. select patient names along with the count of orders they have placed.

```
SELECT
  p.Patient_name,
  (SELECT COUNT(*)
   FROM delivery d
   WHERE d.Patient_Id = p.Patient_Id
  ) AS OrderCount
FROM
  Patient p
LIMIT 10;
```

Patient_name	OrderCount
Shelba	1
Win	3
Rani	1
Tove	1
Caleb	1
Papageno	1
Cindee	0
Jobye	1
Wallas	1
Roch	1

NoSQL Implementation:

- Retrieve the count of products in each category:

```
db.product.aggregate([ {
  $group: {
    _id: "$Category_name", ProductCount: {
      $sum: 1 }
    }
  }
]);
```

```
{
  _id: 'Antibiotics',
  ProductCount: 27
}
{
  _id: 'Vitamins',
  ProductCount: 19
}
{
  _id: 'Painkillers',
  ProductCount: 14
}
```

2. Retrieve Top 10 Most Recent Orders:

```
db.delivery.aggregate([
  {
    $sort: { Order_date: -1 }
  },
  { $limit: 10 },
  {
    $project: {
      _id: 0,
      Order_ID: "$Order_ID",
      Order_date: "$Order_date",
      Patient_ID: "$Patient_ID",
      Shop_ID: "$Shop_ID" }
    }
]);
```

```
{
  Order_ID: 'Z053',
  Order_date: '7/13/23',
  Patient_ID: 'T471X',
  Shop_ID: 'T2019XA'
}
{
  Order_ID: '09984',
  Order_date: '6/17/23',
  Patient_ID: 'S86011D',
  Shop_ID: 'S06827S'
}
{
  Order_ID: 'V272',
  Order_date: '6/15/23',
  Patient_ID: 'T23221D',
  Shop_ID: 'S72019'
}
{
  Order_ID: 'S12650G',
  Order_date: '5/6/23',
  Patient_ID: 'S52309',
  Shop_ID: 'S66517D'
}
{
  Order_ID: 'M84759S',
  Order_date: '5/31/23',
  Patient_ID: 'A59',
  Shop_ID: 'S72309F'
}
```

```
< {
  Order_ID: 'M21961',
  Order_date: '8/19/23',
  Patient_ID: 'S5609',
  Shop_ID: 'S66517D'
}
{
  Order_ID: 'H7101',
  Order_date: '8/16/23',
  Patient_ID: 'S82854J',
  Shop_ID: 'H1022'
}
{
  Order_ID: 'S51002D',
  Order_date: '8/12/23',
  Patient_ID: 'T2176XS',
  Shop_ID: 'S82011P'
}
{
  Order_ID: 'S92065A',
  Order_date: '8/10/23',
  Patient_ID: 'S96802D',
  Shop_ID: 'S72309F'
}
{
  Order_ID: 'S93504S',
  Order_date: '8/10/23',
  Patient_ID: 'H26049',
  Shop_ID: 'S63043S'
}
```

3. Retrieve 10 results for patients who have placed any orders:

```

db.patient.find(
  { _id: {
    $nin: db.ordered_from.distinct("patient_id")
  }},
  {
    Patient_ID: 1,
    Gender: 1,
    Patient_Name: 1,
    Insurance_No: 1,
    _id: 0
  }).limit(10);

```

```

{
  Patient_ID: 'T382X1',
  Gender: 'Male',
  Patient_Name: 'Papageno',
  Insurance_No: 'R701'
}
{
  Patient_ID: 'S82202K',
  Gender: 'Female',
  Patient_Name: 'Cindee',
  Insurance_No: 'Y9375'
}
{
  Patient_ID: 'G803',
  Gender: 'Female',
  Patient_Name: 'Jobye',
  Insurance_No: 'V00211A'
}
{
  Patient_ID: 'M85479',
  Gender: 'Male',
  Patient_Name: 'Wallas',
  Insurance_No: 'I6523'
}

```

```

< {
  Patient_ID: 'S79002G',
  Gender: 'Female',
  Patient_Name: 'Shelba',
  Insurance_No: 'M216X9'
}
{
  Patient_ID: 'Q30',
  Gender: 'Male',
  Patient_Name: 'Win',
  Insurance_No: 'Y00XXXS'
}
{
  Patient_ID: 'T23221D',
  Gender: 'Female',
  Patient_Name: 'Rani',
  Insurance_No: 'S62347B'
}
{
  Patient_ID: 'S5609',
  Gender: 'Female',
  Patient_Name: 'Tove',
  Insurance_No: 'S72462H'
}
{
  Patient_ID: 'S86011D',
  Gender: 'Male',
  Patient_Name: 'Caleb',
  Insurance_No: 'T22099'
}

```

4. Retrieve the Latest Order for Each Patient:

```
db.delivery.aggregate([
  {
    $sort: { Order_date: -1 }
  },
  { $group: {
    _id: "$Patient_ID",
    LatestOrder: { $first: "$$ROOT" } }},
  {
    $lookup: {
      from: "product",
      localField: "LatestOrder.Order_ID",
      foreignField: "Order_ID",
      as: "ProductDetails"
    },
  },
  {
    $unwind: "$ProductDetails" },
  { $project: {
    _id: 0,
    Patient_ID: "$_id",
    Shop_ID: "$LatestOrder.Shop_ID",
    Product_Id: "$ProductDetails.Product_ID",
    Category_name:
"$ProductDetails.Category_name",
    LatestOrderDate: "$LatestOrder.Order_date"
  }},
  {
    $limit: 10}]);
```

```
Patient_ID: 'T471X',
Shop_ID: 'T2019XA',
Product_Id: 'T82858S',
Category_name: 'Antibiotics',
LatestOrderDate: '7/13/23'
}
{
  Patient_ID: 'S5609',
  Shop_ID: 'S66517D',
  Product_Id: 'S59002A',
  Category_name: 'Vitamins',
  LatestOrderDate: '8/19/23'
}
{
  Patient_ID: 'H26049',
  Shop_ID: 'S63043S',
  Product_Id: 'Q671',
  Category_name: 'Antibiotics',
  LatestOrderDate: '8/10/23'
}
{
  Patient_ID: 'S86011D',
  Shop_ID: 'S06827S',
  Product_Id: 'T485X5D',
  Category_name: 'Antibiotics',
  LatestOrderDate: '6/17/23'
}
{
  Patient_ID: 'S52309',
  Shop_ID: 'S66517D',
  Product_Id: 'V5959',
  Category_name: 'Painkillers',
  LatestOrderDate: '5/6/23'
```

```
Patient_ID: 'Q30',
Shop_ID: 'H25',
Product_Id: 'T56813D',
Category_name: 'Vitamins',
LatestOrderDate: '5/26/23'
}
{
  Patient_ID: 'S82854J',
  Shop_ID: 'H1022',
  Product_Id: 'S3217XB',
  Category_name: 'Antibiotics',
  LatestOrderDate: '8/16/23'
}
{
  Patient_ID: 'S82854J',
  Shop_ID: 'H1022',
  Product_Id: 'S01101A',
  Category_name: 'Painkillers',
  LatestOrderDate: '8/16/23'
}
{
  Patient_ID: 'S82854J',
  Shop_ID: 'H1022',
  Product_Id: 'S53441A',
  Category_name: 'Painkillers',
  LatestOrderDate: '8/16/23'
}
{
  Patient_ID: 'S82854J',
  Shop_ID: 'H1022',
  Product_Id: 'S52325Q',
  Category_name: 'Antibiotics',
  LatestOrderDate: '8/16/23'
```

5. Retrieve the top 10 best-selling products:

```
db.ordered_from.aggregate([
  {$group: {
    _id: "$Product_ID",
    TotalQuantitySold: { $sum: 1 }
  }},
  { $lookup: {
    from: "product",
    localField: "_id",
    foreignField: "Product_ID",
    as: "productDetails"
  } },
  {
    $unwind: "$productDetails" },
  {
    $project: {
      _id: 0,
      Product_ID: "$_id",
      Category_Name:
        "$productDetails.Category_name",
      Manufacturer:
        "$productDetails.Manufacturer",
      TotalQuantitySold: 1
    } },
  { $sort: { TotalQuantitySold: -1 }
},
{ $limit: 10
}]);
```

```
{
  TotalQuantitySold: 1,
  Product_ID: 'V5959',
  Category_Name: 'Painkillers',
  Manufacturer: 'Talmocom Co., Ltd'
}
{
  TotalQuantitySold: 1,
  Product_ID: 'S01101A',
  Category_Name: 'Painkillers',
  Manufacturer: 'Aaron Industries, Inc.'
}
{
  TotalQuantitySold: 1,
  Product_ID: 'X008XXD',
  Category_Name: 'Vitamins',
  Manufacturer: 'Budpak Inc.'
}
{
  TotalQuantitySold: 1,
  Product_ID: 'S3217XB',
  Category_Name: 'Antibiotics',
  Manufacturer: 'Melaleuca Inc.'
}
{
  TotalQuantitySold: 1,
  Product_ID: 'T17910',
  Category_Name: 'Vitamins',
  Manufacturer: 'Cantrell Drug Company'
}
```

```
{
  TotalQuantitySold: 6,
  Product_ID: 'T82858S',
  Category_Name: 'Antibiotics',
  Manufacturer: 'Uriel Pharmacy Inc.'
}
{
  TotalQuantitySold: 5,
  Product_ID: 'O328XX1',
  Category_Name: 'Antibiotics',
  Manufacturer: 'Nelco Laboratories, Inc.'
}
{
  TotalQuantitySold: 5,
  Product_ID: 'B509',
  Category_Name: 'Antibiotics',
  Manufacturer: 'Physicians Total Care, Inc.'
}
{
  TotalQuantitySold: 4,
  Product_ID: 'S72366C',
  Category_Name: 'Antibiotics',
  Manufacturer: 'A&Z Pharmaceutical, Inc.'
}
{
  TotalQuantitySold: 1,
  Product_ID: 'K958',
  Category_Name: 'Antibiotics',
  Manufacturer: 'STAT Rx USA LLC'
}
```

6. Retrieve orders placed by patients whose payment status is failed or pending:

```
db.delivery.aggregate([
  {
    $lookup: {
      from: "patient",
      localField: "Patient_ID",
      foreignField: "Patient_ID",
      as: "patientInfo"
    } },
  {
    $lookup: {
      from: "payment",
      localField: "Order_ID",
      foreignField: "Order_ID",
      as: "paymentInfo"
    } },
  {
    $match: {
      "paymentInfo.Payment_status": { $in: ["Failed", "Pending"] }
    } },
  {
    $project: {
      _id: 0, // Exclude the _id field
      Order_ID: "$Order_ID",
      Order_Date: "$Order_date",
      Patient_ID: "$Patient_ID",
      Patient_name: { $arrayElemAt: ["$patientInfo.Patient_Name", 0] },
      Payment_Status: "$paymentInfo.Payment_status"
    } },
  {
    $limit: 10
  }
])
```

```

    Order_ID: 'S82822K',
    Order_Date: '11/1/23',
    Patient_ID: 'S79002G',
    Patient_name: 'Shelba',
    Payment_Status: [
      'Pending'
    ]
  }
}
{
  Order_ID: 'V55',
  Order_Date: '3/23/23',
  Patient_ID: 'Q30',
  Patient_name: 'Win',
  Payment_Status: [
    'Failed'
  ]
}
}
{
  Order_ID: 'V272',
  Order_Date: '6/15/23',
  Patient_ID: 'T23221D',
  Patient_name: 'Rani',
  Payment_Status: [
    'Failed'
  ]
}
}
{
  Order_ID: 'M21961',
  Order_Date: '8/19/23',
  Patient_ID: 'S5609',
  Patient_name: 'Tove',
  Payment_Status: [
    'Pending'
  ]
}
}

```

```

    Order_ID: 'O9984',
    Order_Date: '6/17/23',
    Patient_ID: 'S86011D',
    Patient_name: 'Caleb',
    Payment_Status: [
      'Pending'
    ]
  }
}
{
  Order_ID: 'T881',
  Order_Date: '3/8/23',
  Patient_ID: 'T382X1',
  Patient_name: 'Papageno',
  Payment_Status: [
    'Failed'
  ]
}
}
{
  Order_ID: 'O411425',
  Order_Date: '1/3/23',
  Patient_ID: 'M85479',
  Patient_name: 'Wallas',
  Payment_Status: [
    'Failed'
  ]
}
}
{
  Order_ID: 'T20419A',
  Order_Date: '1/30/23',
  Patient_ID: 'Z818',
  Patient_name: 'Roch',
  Payment_Status: [
    'Failed'
  ]
}
}

```

7. Query to Retrieve Top 5 Patients with the Highest Total Amount

```

db.payment.aggregate([
  {
    $group: {
      _id: "$Patient_ID",
      TotalAmountSpent:
        { $sum: "$Total_amt"
        }
    },
  },
  {
    $sort: { TotalAmountSpent: -1 }
  },
  {
    $limit: 5
  }
]);

```

```

{
  _id: 'S92213S',
  TotalAmountSpent: 9788.32
}
{
  _id: 'W458',
  TotalAmountSpent: 9780
}
{
  _id: 'T23139',
  TotalAmountSpent: 9494.09
}
{
  _id: 'S82854J',
  TotalAmountSpent: 9455.23
}
{
  _id: 'Z4901',
  TotalAmountSpent: 9334.94
}

```


8. Query to Retrieve Orders Delivered in a Specific Location:

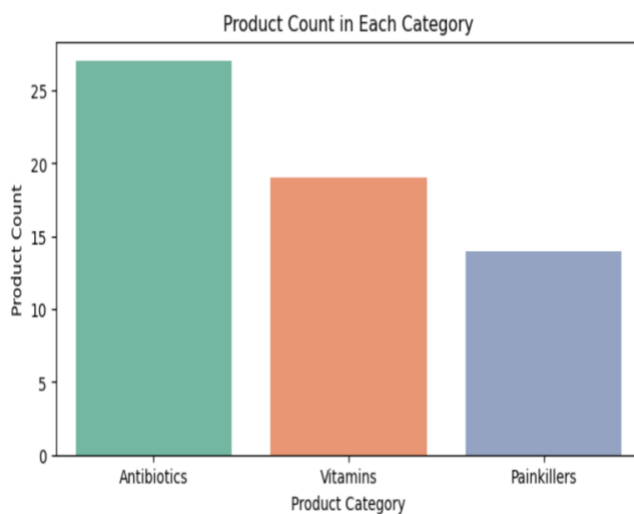
```
db.delivery.find (
{
  "Delivery_Address": "R802",
  "Order_status": "Delivered"
})
```

```
{
  "_id": ObjectId("656a7e9f3a43003b64f88360"),
  "Delivery_Address": 'R802',
  "Order_ID": 'S82822K',
  "Order_date": '11/1/23',
  "Order_status": 'Delivered',
  "Shop_ID": 'S66517D',
  "Patient_ID": 'S79002G',
  "Emp_ID": 'T38811D'
}
```

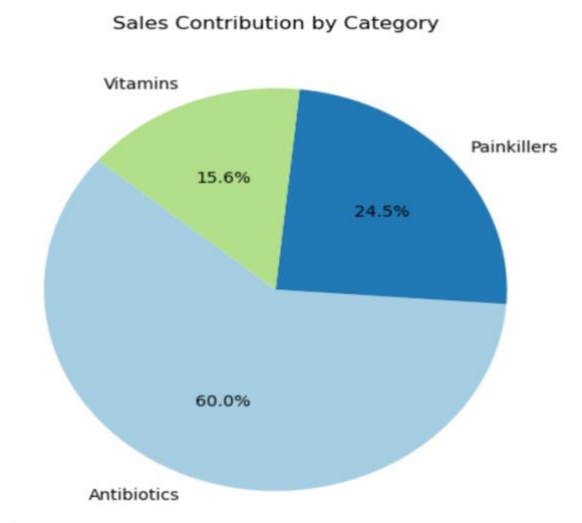
V. Database Access via Python

The database is accessed using Python and visualization of analyzed data is shown below. The connection of MySQL to Python is done using mysql.connector, followed by cursor.execute to run and fetchall from query, followed by converting the list into a dataframe using pandas library and using matplotlib to plot the graphs for the analytics.

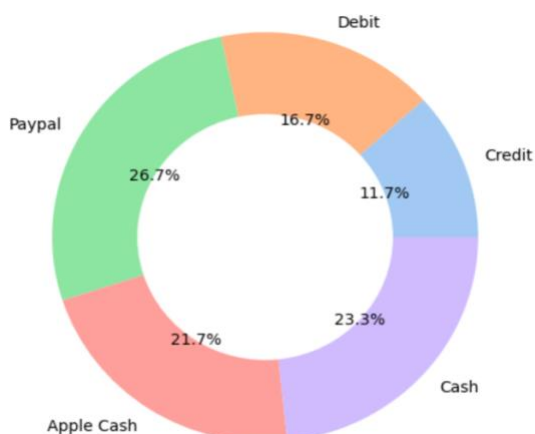
Graph1: Products in each category



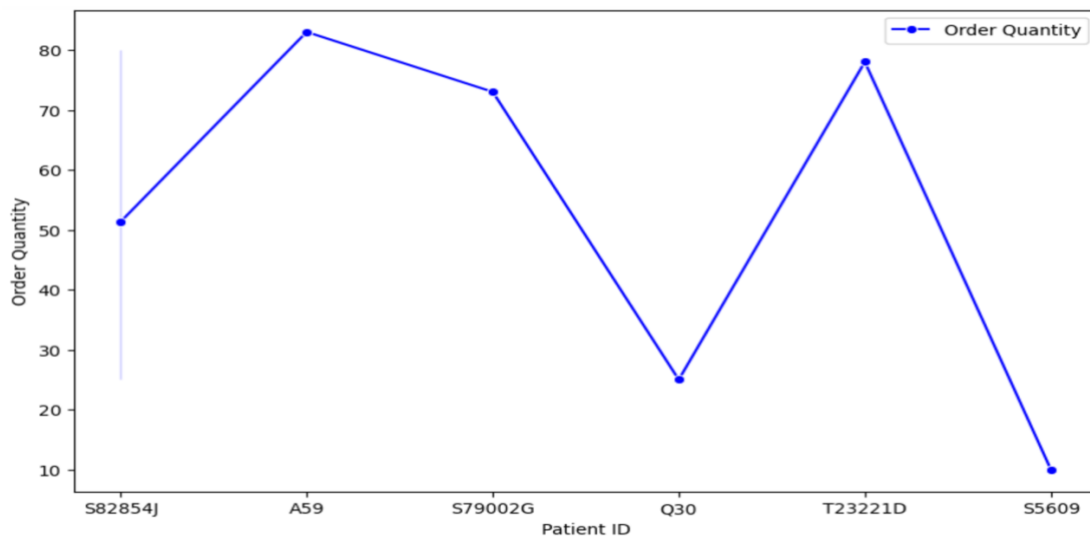
Graph 2: Total sales contribution by Category



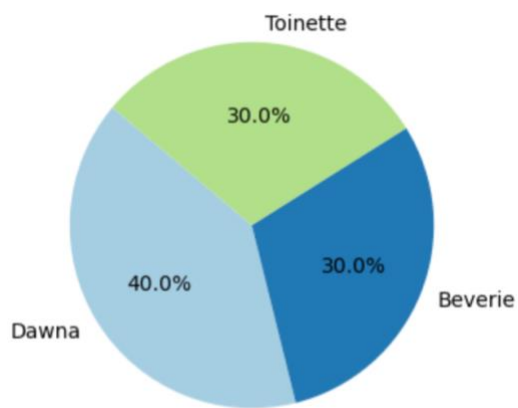
Graph 3: Distribution of Payment Methods.



Graph 4: Distribution of Order quantity placed by each patient:



Graph 5: Distribution of Orders Processed by Employees



Graph 6: Distribution of Pickup and Delivery Orders



VI. Summary and Recommendation

In response to the challenges faced by traditional pharmacies, the Medimart database designed in MYSQL is an Online Pharmacy Inventory and Order Management System. The project addresses the inefficiencies and limitations prevalent in traditional pharmacy models by leveraging advanced technology and innovative solutions. By providing a secure online platform, Medimart aims to enhance the efficiency of pharmacy operations, minimize errors in inventory management and order processing, and elevate patient engagement.

The Medimart database is positioned to revolutionize the pharmacy industry by addressing critical challenges through technology-driven solutions. By focusing on optimized inventory management, streamlined order processing, enhanced patient engagement, operational efficiency, and maintaining a customer-centric approach, Medimart can establish itself as a leader in the online pharmacy space. Continuous innovation and a commitment to meeting customer expectations will be key to the success of Medimart in the evolving landscape of digital healthcare services.

Operational efficiency will be enhanced by integrating the Medimart database with existing healthcare information systems, implementing workflow automation to reduce manual processes, and providing comprehensive training programs for pharmacy staff. To maintain a competitive edge and ensure customer satisfaction, Medimart commits to monitoring industry trends, gathering feedback for continuous improvement, and implementing loyalty programs and personalized discounts to enhance customer retention.