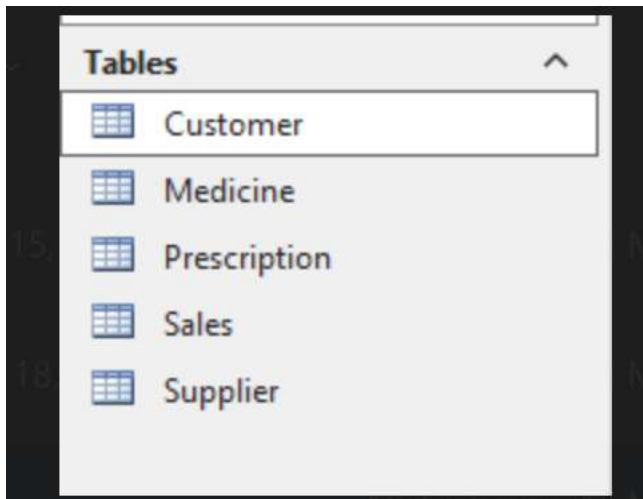


Database Coursework Submission Template

1. Student Details

- Name: Abdirahman Mahdi
- Student ID: 2429936
- Module: Introduction to Web and Database Concepts
- Lecturer:

2. Table of Contents



3. Introduction

MediLink Pharmacy is a database management application which has been developed to manage all the operations of the pharmacy using storing and organization of information regarding customers, medicines, prescriptions, suppliers, and sales transactions. Customers are able to buy medicine through this database with or without prescription, and all this will be recorded in the most accurate manner possible. An efficient database helps in saving time, reducing the amount of paperwork, maintaining accuracy of record keeping, and providing quick services to the customers. This report will discuss the design and implementation of this database.

The purpose of the database system

MediLink Pharmacy Database serves the aim of organizing pharmacy information in an effective manner. The database enables the staff to have accurate records of clients, medicines, prescriptions, suppliers, and sales transactions. Through related tables in the database, data duplication is eliminated and therefore makes the retrieval of data easy through queries. This database assists the pharmacy in providing good customer service, keeping track of the quantity of medicines, and recording sale transactions effectively and accurately.

4. Scenario Analysis

- Identify entities:

Customer

Medicine

Prescription

Sales

Supplier

- Attributes of each entity

Customer: CustomerID, Phone, Address, Name

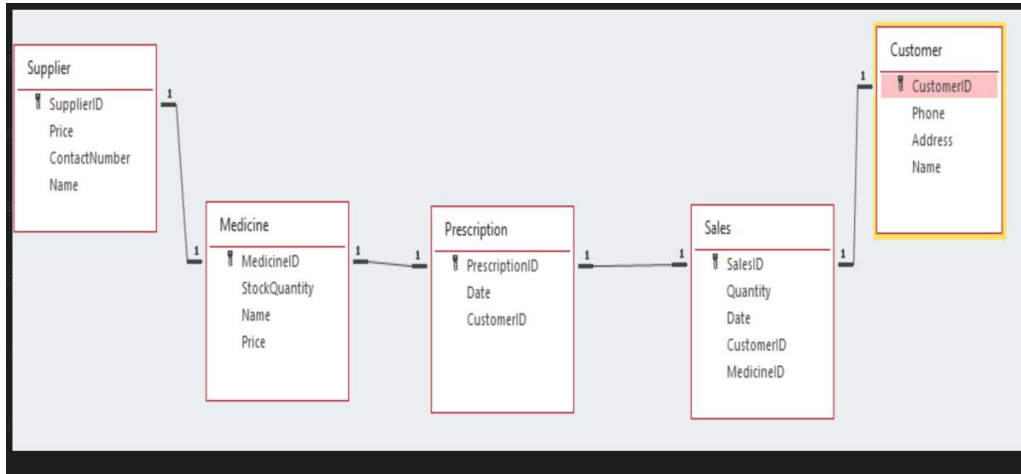
Supplier: SupplierID, Price, Contact Number, Name

Medicine: MedicineID, Stock quantity, Name, Price

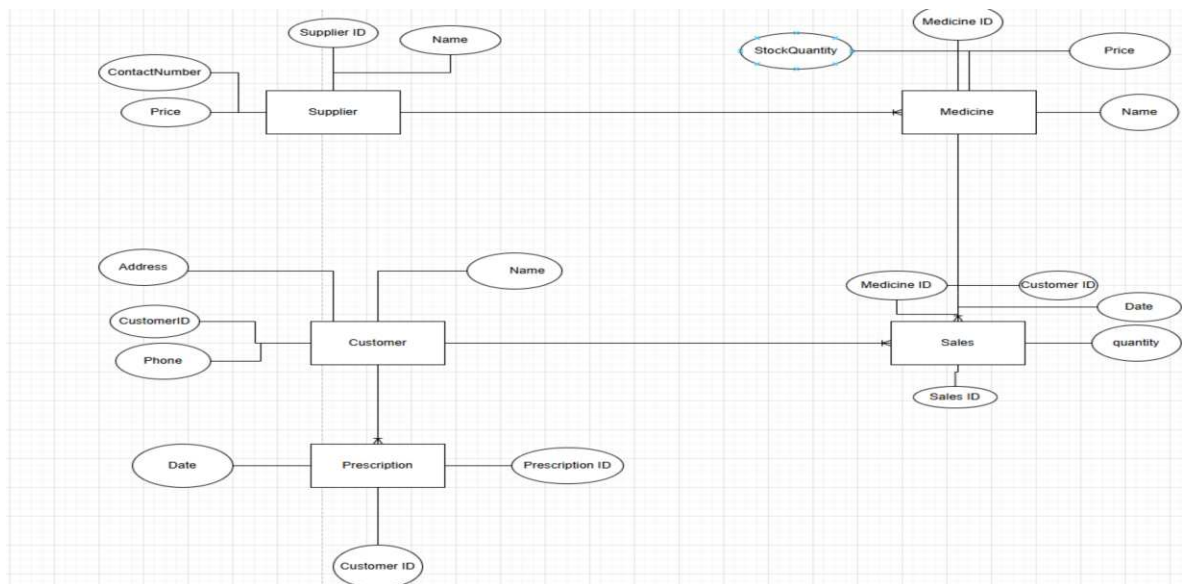
Sales: SalesID, CustomerID, MedicineID, Date, Quantity

Prescription: PrescriptionID, Date, CustomerID

- Relationships between entities



5. Entity Relationship Diagram (ERD)

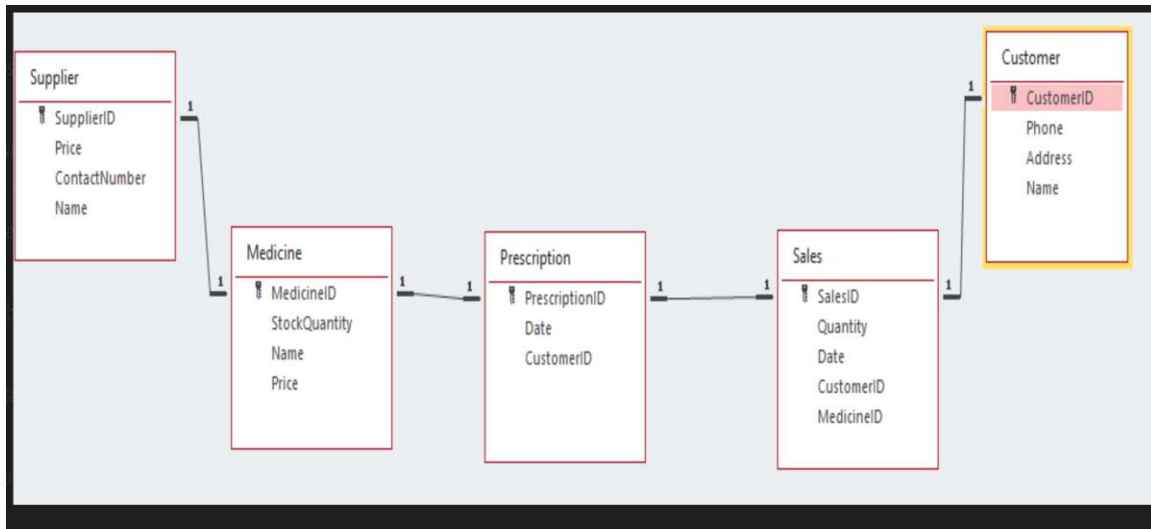


6. Data Structure Design

CustomerID	Phone	Address	Name		
CUST-1001	7700900001	LS1 4BJ,	James		
CUST-1002	7700900002	BS8 1QU,	Olivia		
CUST-1003	7700900003	B4 7SL	Noah		
CUST-1004	7700900004	M1 3AN,	Amelia		
CUST-1005	7700900005	NW1 6XE	Ethan		
SupplierID	Price	ContactNumb	Name		
SUP-001	£3.50	1618502101	Liam		
SUP-002	£5.99	1217603402	Sophia		
SUP-003	£7.25	1139205503	Daniel		
SUP-004	£10.00	2079467804	Ava		
SUP-005	£14.75	1516309905	Lucas		
MedicineID	StockQuantit	Price	Name		
3001	50	£2.99	James Wilson		
3002	25	£4.50	Olivia Taylor		
3003	100	£12.30	Daniel Harris		
3004	75	£9.99	Sophia Clark		
3005	200	£21.89	Ethan Walker		
SaleID	Quantity	Date	CustomerID	MedicineID	
SALE-001	34	15/01/2026	1001	4002	
SALE-002	54	18/01/2026	1003	4005	
SALE-003	50	20/01/2026	1002	4001	
SALE-004	100	27/01/2026	1005	4004	
SALE-005	75	19/01/2026	1004	4003	
PrescriptionID	Date	CustomerID			
PRES-001	15/01/2026	6001			
PRES-007	18/02/2026	9002			
PRES-002	15/05/2026	9006			
PRES-004	22/01/2026	4002			
PRES-006	20/01/2026	5001			

The database model was created through the formation of five tables, which include the following: Customer, Supplier, Medicine, Sales, and Prescription. Every table has its own primary keys like CustomerID, SupplierID, MedicineID, SaleID and PrescriptionID respectively to distinguish each record. The sales table connects customers and medicines through the use of their IDs while the prescription table connects prescription with customers. This kind of structure helps to organize the data efficiently and avoid duplication of the information, as well as accessing related records quickly and easily. This kind of

database will be helpful in managing the activities of the pharmacy on a daily basis.



The ERD provides important entities for the database design of the MediLink Pharmacy database system. These entities include customers, medicines, suppliers, sales, and prescriptions. Each entity has attributes and a primary key to help easily locate the records. The relationship is connected to show the link between these entities.

For example: One customer can have many prescriptions

Linking data through relationships secures the overall database design.

7. Normalisation (Up to 3NF)

PrescriptionID		Date	CustomerID
PRES-001		15/01/2026	6001
PRES-007		18/02/2026	9002
PRES-002		15/05/2026	9006
PRES-004		22/01/2026	4002
PRES-006		20/01/2026	5001
SupplierID	Price	ContactNumber	Name
SUP-001	£3.50	1618502101	Liam
SUP-002	£5.99	1217603402	Sophia
SUP-003	£7.25	1139205503	Daniel
SUP-004	£10.00	2079467804	Ava
SUP-005	£14.75	1516309905	Lucas
MedicineID	StockQuantity	Price	Name
3001	50	£2.99	James Wilson
3002	25	£4.50	Olivia Taylor
3003	100	£12.30	Daniel Harris
3004	75	£9.99	Sophia Clark
3005	200	£21.89	Ethan Walker

SaleID	Date	CustomerID	MedicineID	Quantity
SALE-001	15/01/2026	1001	4002	34
SALE-002	18/01/2026	1003	4005	54
SALE-003	20/01/2026	1002	4001	50
SALE-004	27/01/2026	1005	4004	100
SALE-005	19/01/2026	1004	4003	75

- Explain decisions

The data was converted to First Normal Form (1NF) by ensuring that there are no repeating groups in any table, and each record in a table is unique with a primary key, with all fields having a single value. The unstructured data was split into different tables, e.g., Sales, Medicine, Supplier, Prescription, etc., to reduce ambiguity and redundancy.⁸

Sample Data

SaleID	Date	CustomerID
SALE-001	15/01/2026	1001
SALE-002	18/01/2026	1003
SALE-003	20/01/2026	1002
SALE-004	27/01/2026	1005
SALE-005	19/01/2026	1004

MedicineID	StockQuantity	Price	Name
3001	50	£2.99	James Wilson
3002	25	£4.50	Olivia Taylor
3003	100	£12.30	Daniel Harris
3004	75	£9.99	Sophia Clark
3005	200	£21.89	Ethan Walker

SupplierID	Price	ContactNumber	Name
SUP-001	£3.50	1618502101	Liam
SUP-002	£5.99	1217603402	Sophia
SUP-003	£7.25	1139205503	Daniel
SUP-004	£10.00	2079467804	Ava
SUP-005	£14.75	1516309905	Lucas

PrescriptionID	Date	CustomerID
PRES-001	15/01/2026	6001
PRES-007	18/02/2026	9002
PRES-002	15/05/2026	9006
PRES-004	22/01/2026	4002
PRES-006	20/01/2026	5001

The database is transformed into a Second Normal Form (2NF) by removing partial dependencies. The Sales table is split into two tables, Sales, and Sales Details. In the Sales table, the attributes dependent only on the SaleID are stored. Similarly, the attributes dependent on both SaleID and MedicineID are stored in the Sales Details table.

CustomerID	Name
1001	James Wilson
1002	Olivia Taylor
1003	Daniel Harris
1004	Sophia Clark
1005	Ethan Walker

SaleID	Date	CustomerID
SALE-001	15/01/2026	1001
SALE-002	18/01/2026	1003
SALE-003	20/01/2026	1002
SALE-004	27/01/2026	1005
SALE-005	19/01/2026	1004

MedicineID	Name	Price	StockQuantity	SupplierID
3001	Paracetamol	£2.99	50	SUP-001
3002	Ibuprofen	£4.50	25	SUP-002
3003	Amoxicillin	£12.30	100	SUP-003
3004	Cough Syrup	£9.99	75	SUP-004
3005	Vitamin C	£21.89	200	SUP-005

PrescriptionID	Date	CustomerID
PRES-001	15/01/2026	1001
PRES-007	18/02/2026	1003
PRES-002	15/05/2026	1002
PRES-004	22/01/2026	1005
PRES-006	20/01/2026	1004

SupplierID	Name	ContactNumber
SUP-001	Liam	1618502101
SUP-002	Sophia	1217603402
SUP-003	Daniel	1139205503
SUP-004	Ava	2079467804
SUP-005	Lucas	1516309905

The database is normalized to Third Normal Form (3NF) by removing the Transitive Dependency. Attributes such as the supplier's information are moved to another table called Supplier, and the Medicine table is related to the Supplier table through the foreign key. In addition, the wrong placement of attributes is corrected so that all non-key attributes are dependent on the primary key only.

8. Query development

These queries can be used in an actual pharmacy because they assist in making decisions. For example, the stock-quantity query will help employees in the pharmacy to determine which stock-quantity should be restocked. In this way, low stock is avoided.

Query 1.

As demonstrated by the query results, the specified medication has fallen below its designated minimum threshold, indicating an immediate requirement for replenishment.

MedicineID	StockQuantity
3001	50

Query 2.

This query gives you an instant look at the exact price tag for restocking each medication. Instead of pulling out a calculator and wasting time on manual math, the pharmacy team can see the total costs automatically.

MedicineID	StockQuantity	Price
3001	50	£2.99

Query 3.

This query will present the prescription ID along with its due date. It is an indication that the required medicine should be kept ready by the hospital personnel before that particular day.

PrescriptionID	Date
PRES-001	15/01/2026

Query 4.

This will display the prescription ID number, due date, and the customer ID, which shows the whole transaction. It is very important for these records to remain linked to one another in the database because without them it would be impossible to track the important details which might cause some administrative problems for the patient.

PrescriptionID	Date	CustomerID
PRES-001	15/01/2026	1001
PRES-007	18/02/2026	1003

9. Validation and Testing

The testing of my database was conducted by entering a variety of valid and invalid data into it. The goal was to verify whether the data types and constraints worked as required. The queries were used to check if the correct output was obtained from the database. The relationships were tested through the introduction of invalid foreign keys which I cannot provided currently this is due to not having to access MS access.

Test	Expected Result	Actual Result	Status
Search for medicine	Customer record is successfully added	Medicine found and displayed	PASS
Low stock query	Medicines with stock below 70 are displayed	Correct medicines returned	PASS
Display Prescription	Customer prescription history is displayed	Correct prescription information shown	PASS
Add a new customer	Customer record is successfully added	Customer record added successfully	PASS
Vaid customer data	Enter correct name, phone	Data successfully found	PASS

Validation is crucial in the pharmacy system due to the possibility of invalid data causing dire effects, such as prescription errors and out of stock medicines. Data validation within the pharmacy system is done by ensuring that data type enforcement and referential integrity are upheld in the database.

10. Issues and fixes

While creating my database, I encountered a few problems that I needed to solve. The problem of incorrect data types resulted in query errors. To resolve the problem of the wrong data type, I changed the data types of the fields. Furthermore, I eliminated duplicate fields in tables like the Stock table.

11. Reflection

- One of the major challenges in this case was how to organize the data correctly during the normalization process. It was also hard to ensure that each attribute belonged to the correct entity. It was hard to ensure that relationships between tables were correctly defined.
- The system can be enhanced in the future by adding more features such as stock updates, user interfaces, and validation rules. Realistic data can also be used in addition to more tests.

12. Conclusion

- The MediLink Pharmacy database system effectively manages data on customers, medicines, suppliers, prescriptions, and sales. Using normalization up to 3NF, the database minimizes data redundancy. It is an efficient way to manage the activities of the pharmacy.

13. References

- T. Connolly and C. Begg, *Database Systems: A Practical Approach to Design, Implementation, and Management*, 6th ed. Harlow, UK: Pearson, 2015.

-A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill, 2019.