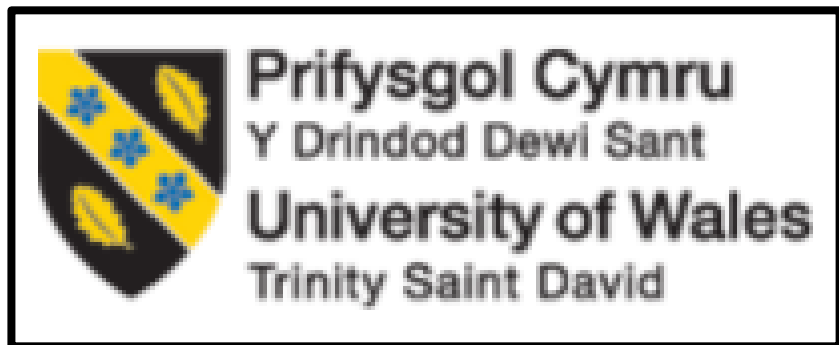




CertHE Computing Skills for Workplace

ACCA4023 Introduction to Web and Database

Concepts

Design and Implementation of a Relational Database

System

Student name: Mihai Cristin

Student ID: 2442385

Table of Contents

1. Introduction	3
2. Scenario Overview&Requirements	3
3. Database Design (ERD)	3
4. Data Structure Development	5
5. Normalisation (3NF)	7
6. Query Development	8
7. Validation and Testing	10
8. Design Decisions and Justification	10
9. Reflection and Future Improvements	11
10. Conclusion	11
References	12

1. Introduction

MediLink Pharmacy is an expanding local pharmacy that is still using manual means of running its business including dealing with customer records, prescriptions and sales transactions. Such a strategy tends to cause inefficiency and mistakes as well as problems in keeping proper records. This database system is supposed to automate these processes by offering a reliable and systematic means of storing, controlling, and accessing data. The automation of a pharmacy environment is critical since it can enhance the efficiency of the operations, minimise human error, and assist in making improved decisions due to the availability of correct and timely information.

2. Scenario Overview&Requirements

MediLink Pharmacy has been using manual processes to operate its business such as monitoring medicines, handling prescription orders and recordings of sales. This system causes hitches that include errors in data entry, lost records, and loss of time accessing valuable information. The available system will progressively become unreliable and hard to handle as the pharmacy continues to expand. A relational database system is needed to streamline the operations and automate them in order to overcome these problems. The system should effectively handle the major data entities such as medicines, customers, suppliers, prescriptions, and sales transactions. It must provide the right storage, easy access, and uniformity of information in all its operations. Moreover, the system is likely to produce valuable outputs like reports and queries. They involve tracking the availability of stocks, low stock medicines, extracting customer prescription records and reviewing sales performance. Generally, the database system is supposed to enhance efficiency, accuracy and decision-making in the pharmacy.

3. Database Design (ERD)

The database scheme of the MediLink Pharmacy is founded on the identification of key elements in the system. The main entities are Customer, Supplier, Medicine, Prescription, Sales, Prescription_Item and Sales_Item. Every object is a real-life object of the pharmacy system and holds pertinent attributes. As an example, the Customer entity is used to store personal information, and the Medicine entity contains such information as price and stock.

The interconnectioning of these entities is well organised to give real-life interaction. Customer to Prescription and Supplier to Medicine are one-to-many relationships whereby each Customer is allowed to have many prescriptions and Supplier is allowed to supply many medicines. Moreover, the many-to-many relationships are solved with the help of junction tables, including Prescription_Item and Sales_Item, connecting the medicines with the sales and prescriptions respectively [2]. Each record of a table is identified uniquely by the use of primary keys (PK), and tables have a relationship defined by the use of foreign keys (FK). ERD is also necessary because it gives a clear graphic view of the database structure that assists in the logical design, minimises redundancy, and assists in efficient data management [3].

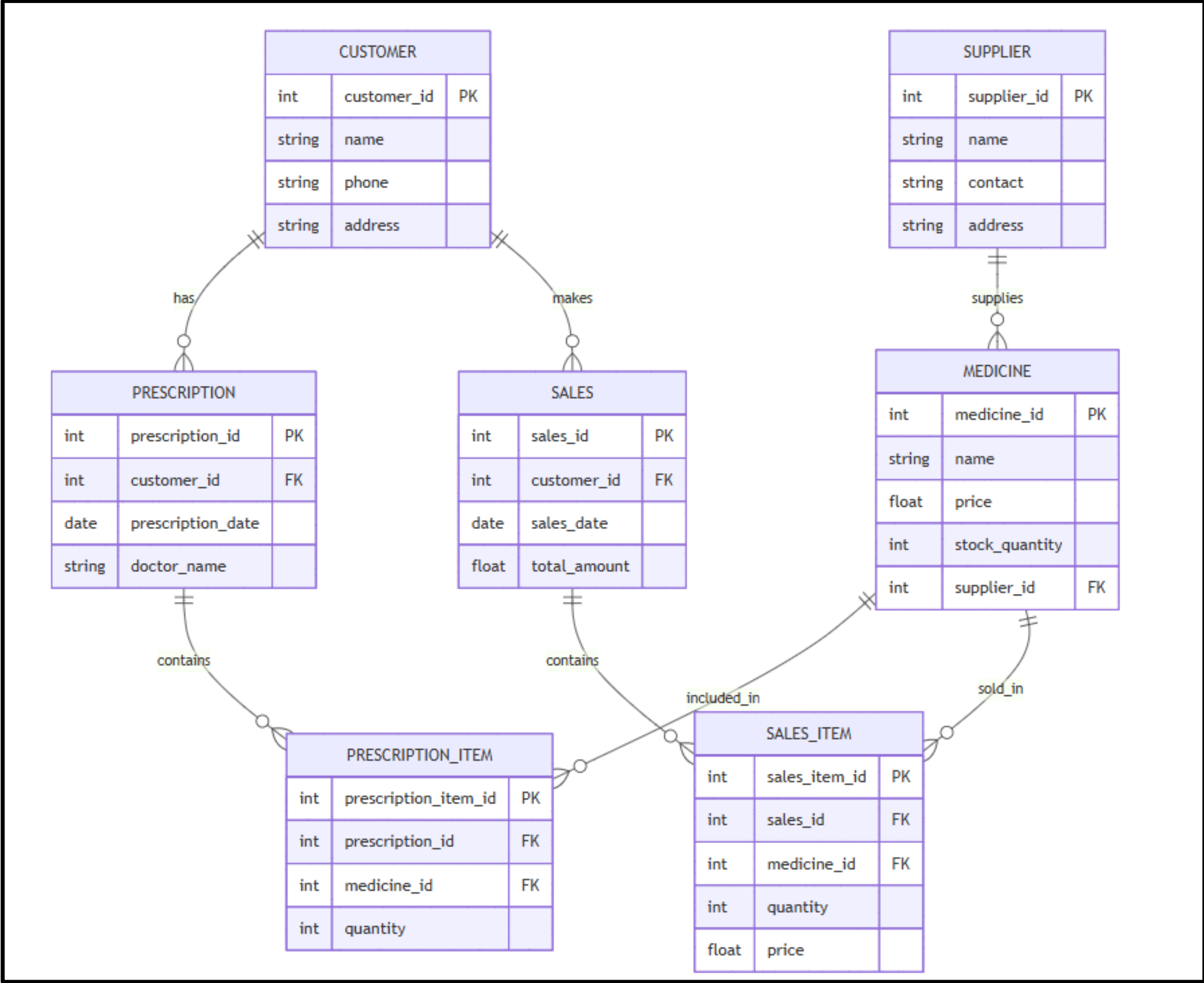


Figure 1: Entity Relationship Diagram (ERD)

(Created by Author)

4. Data Structure Development

The MediLink Pharmacy database data structure was designed based on the establishment of the actual entity-relationship Diagram (ERD), and thus all the entities identified were converted into clear relational tables. The key tables formed are CUSTOMER, SUPPLIER, MEDICINE, PRESCRIPTION, SALES, PRESCRIPTION_ITEM, and SALES_ITEM. All the tables reflect a particular element

of the pharmacy system and hold appropriate attributes needed to manage the data effectively.

Each table was assigned some primary keys (PK) to identify each record in a unique way. As an illustration, customer id in CUSTOMER table and medicine id in MEDICINE table are used to make sure that each record is unique and can be uniquely referenced. This will help in entity integrity because it will not allow duplicate or null records [4].

The concept of foreign keys (FK) was put in place to create relations among tables. To give an example, the customer_id in the PRESCRIPTION and SALES table are referenced to the CUSTOMER table, and supplier_id in the MEDICINE table is referenced to the SUPPLIER table. Such relationships make referential integrity to guarantee that there is related data that is present in the tables and is consistent [11]. Also constraints like NOT NULL and Check were used to ensure validity of data. The NOT NULL constraint was employed to be sure that such crucial fields as names and identifiers are not empty, but the CHECK constraint was employed to limit the values, such as the positivity of price and quantity. These structural components combined guarantee that there are accuracy, consistency and reliability of data throughout the database system.

```
11 ▾ CREATE TABLE CUSTOMER (  
12     customer_id INT PRIMARY KEY,  
13     name VARCHAR(100) NOT NULL,  
14     phone VARCHAR(15),  
15     address VARCHAR(255)  
16 );
```

Figure 2: CUSTOMER Table Structure

```
27 ▾ CREATE TABLE MEDICINE (  
28     medicine_id INT PRIMARY KEY,  
29     name VARCHAR(100) NOT NULL,  
30     price FLOAT CHECK (price > 0),  
31     stock_quantity INT CHECK (stock_quantity >= 0),  
32     supplier_id INT,  
33     FOREIGN KEY (supplier_id) REFERENCES SUPPLIER(supplier_id)  
34 );
```

Figure 3: MEDICINE Table Structure

```
37 CREATE TABLE PRESCRIPTION (  
38     prescription_id INT PRIMARY KEY,  
39     customer_id INT NOT NULL,  
40     prescription_date DATE,  
41     doctor_name VARCHAR(100),  
42     FOREIGN KEY (customer_id) REFERENCES CUSTOMER(customer_id)  
43 );
```

Figure 4: PRESCRIPTION Table Structure

Table 1: Summary of Tables and Relationships

Table Name	Primary Key	Foreign Key(s)	Description
CUSTOMER	customer_id	-	Stores customer details such as name, phone, and address.
SUPPLIER	supplier_id	-	Stores supplier information including contact and address.
MEDICINE	medicine_id	supplier_id → SUPPLIER(supplier_id)	Contains details of medicines, including price, stock, and supplier.
PRESCRIPTION	prescription_id	customer_id → CUSTOMER(customer_id)	Records prescriptions issued to customers by doctors.
PRESCRIPTION_ITEM	prescription_item_id	prescription_id → PRESCRIPTION(prescription_id), medicine_id → MEDICINE(medicine_id)	Links medicines to prescriptions and records quantity prescribed.
SALES	sales_id	customer_id → CUSTOMER(customer_id)	Stores sales transactions made by customers.

SALES_ITEM	sales_item_id	sales_id → SALES(sales_id), medicine_id → MEDICINE(medicine_id)	Links medicines to sales transactions and records quantity and price.
------------	---------------	--	---

5. Normalisation (3NF)

Table 2: Normalisation up to Third Normal Form (3NF)

Normal Form	Requirement	Application in MediLink Database
1NF	Atomic values, no repeating groups	Data stored in separate rows; no multiple values in a single field (e.g., medicines stored in PRESCRIPTION_ITEM)
2NF	Full dependency on primary key	All attributes depend on the whole primary key; junction tables ensure no partial dependency
3NF	No transitive dependency	Supplier details stored in separate SUPPLIER table, linked via foreign key

MediLink Pharmacy database design was brought to the Third Normal Form (3NF) to reduce redundancy and guarantee data integrity [6]. The First Normal Form (1NF) was used to normalise all data fields in such a way that all of them stored atomic values (there were no repeated sets of values). To illustrate, rather than having more than one medicine in a particular field, distinct records in the PRESCRIPTION_ITEM and the SALES_ITEM tables are held.

The Second Normal Form (2NF) was also managed by making sure that all the non-key attributes are completely dependent on the whole primary key. It is especially applicable in junction tables like PRESCRIPTION_ITEM and SALES_ITEM where quantity is reliant on the prescription or sales ID alongside the medicine ID removing partial dependence [9].

Lastly, the design meets the Third Normal Form (3NF) because transitive dependencies are eliminated. These attributes that were not directly related to the main key were stored in other tables. As an example, the details of suppliers are placed in the SUPPLIER table and not inserted in the MEDICINE table and a foreign key is used to create a relationship. This makes sure that every attribute is reliant on the primary key and this leads to a well-constructed and efficient database [3].

6. Query Development

The query development is important to illustrate the way, in which the MediLink Pharmacy database system can be employed to access, process and control information. The queries enable users to retrieve the meaningful information in the database that helps in decision-making procedures like stock control, customer management, and sales analysis [7]. The system addresses the needs of the users efficiently by developing a set of practical queries. Simple queries with SELECT statements were developed to present information of individual tables. As an illustration, a retrieval of all records of customers gives a total view of the registered customers. The use of conditional queries using the WHERE clause was also introduced to select certain pieces of information [5]. It is possible to take the example of determining the medicines that are low in stock, and this assists in holding inventory and preventing shortages.

The JOIN queries were created to join data in many tables, which represent relationships that the database defines. An example of a query between the CUSTOMER and PRESCRIPTION table would enable the system to show the name of customers and the details of their prescription. This shows how relational databases facilitate combined data retrieving among various entities [2]. Also, aggregation queries were used to analyse numerical data through functions such as SUM. As an illustration, the determination of the sum of the sales per client is an indication on buying behaviour. These queries improve the capacity of the system to produce helpful reports and promote business decisions [1][3].

customer_id	name	phone	address
1	John Doe	123456789	London
2	Jane Smith	987654321	Manchester
3	Ali Khan	111222333	Birmingham
4	Sara Ali	444555666	Leeds
5	Tom Hardy	777888999	Liverpool

Figure 5: All Customers Query Output

medicine_id	name	price	stock_quantity	supplier_id
3	Amoxicillin	5	30	3
4	Cough Syrup	4.5	20	4

Figure 6: Low Stock Medicines Query

name	prescription_date	doctor_name
John Doe	2026-03-20	Dr Smith
Jane Smith	2026-03-21	Dr John
Ali Khan	2026-03-22	Dr Lee
Sara Ali	2026-03-23	Dr Brown
Tom Hardy	2026-03-24	Dr White

Figure 7: Customer Prescriptions (JOIN Query)

sales_id	customer_id	sales_date	total_amount
201	1	2026-03-20	5
202	2	2026-03-21	3
203	3	2026-03-22	15
204	4	2026-03-23	4.5
205	5	2026-03-24	12

Figure 8: Sales Details Query

name	total_spent
John Doe	5
Jane Smith	3
Ali Khan	15
Sara Ali	4.5
Tom Hardy	12

Figure 9: Total Sales by Customer (GROUP BY)

7. Validation and Testing

The MediLink Pharmacy database system was extensively tested to fulfill the functional requirements and work well. A number of these factors of the system were tested, such as the ability to create a table, insert data, relations, and execute a query. All tables have been created successfully based on the required structure and appropriate keys, both primary and foreign, have been used to provide correct linkage among entities. The insertion of data was tested by inserting sample records to every table, and the accuracy and completeness of the stored data was confirmed with the help of SELECT queries [1]. The use of foreign key validated the relationships between tables and guaranteed referential

integrity [13]. As an example, the entries in the PRESCRIPTION table as well as SALES table refer to existing customers correctly and no invalid, and orphan records were identified.

The testing was done by running various SQL queries such as simple retrieval, conditional filtering, JOINS and aggregation functions. None of the queries gave faulty and unforeseen results. The deliverables proved that the database system is capable of accessing and manipulating information as per user request. Generally, the system was very reliable with no errors realized throughout the testing. The findings indicate that the database is properly designed, uniform, and can cater to the operational requirements of the MediLink Pharmacy.

8. Design Decisions and Justification

The MediLink Pharmacy database was developed based on individual tables to depict different entities that include; customers, medicines, suppliers, prescriptions and sales. Such separation maintains logical organisation of data and eliminates repetition, increasing the efficiency and maintainability [2]. All the tables are dedicated to a particular kind of information, which is easier to update and minimizes the chances of inconsistencies.

This was changed by introducing junction tables, i.e., PRESCRIPTION_ITEM and SALES_ITEM, to address many-to-many connection between prescriptions and medicines, and sales and medicines. Such tables enable different medicines to be associated with a single prescription or sale but with a structured and scalable format [3].

Data integrity and valid entry of data were enforced and implemented by constraints that included PRIMARY KEY, FOREIGN KEY, NOT NULL and CHECK. These limitations eliminate the chance of errors like inaccurate records or invalid

values. These design choices, in general, improve the integrity of data, minimise the redundancy and assist in the effective data management in the system.

9. Reflection and Future Improvements

In the process of creating the MediLink Pharmacy database, a number of issues were faced more so in the relational structure design where the various entities and relations should have been properly defined. The correct choice of primary and foreign keys and ensuring consistency between various related tables could not have been done without proper planning and repetitive testing [10]. Also, to generate more complicated queries, in particular, to use JOIN and aggregation queries, it was essential to understand the architecture of how data is related to each other in the system [12]. The project increased knowledge in the field of relational database concepts such as data modelling, normalisation and the need to ensure data integrity by using constraints. It as well increased the practical skills of structuring databases, and retrieving meaningful insights through SQL queries. In case of future upgrades, the system might be improved through coming up with user-friendly forms to make data entry and interaction easier. It could also use automated reports that offer real-time sales and stock data [8]. Also, the additional security measures that are to be added, including access control and user authentication, would enhance data security and general system reliability.

10. Conclusion

To sum up, the MediLink Pharmacy database system was effectively developed and installed to overcome the inefficiency in the manual data management. The system is efficient in controlling customers, medicines, prescriptions, as well as sales and guaranteeing the data integrity and consistency. The important goals such as the database design, normalisation, and query development were

attained. The project shows the significance of the database systems in enhancing the operational efficiency, aiding decision-making procedures, and facilitating proper data management in organisations.

References

- [1] Al Maruf, Abdullah, et al. "A Systematic Review of The Role Of SQL And Excel In Data-Driven Business Decision-Making For Aspiring Analysts." *American Journal of Scholarly Research and Innovation* 1.01 (2022): 249-269.
- [2] Connolly, Thomas M., and Carolyn E. Begg. *Database systems: a practical approach to design, implementation, and management*. 6th edn. Harlow: Pearson, 2015.
- [3] Elmasri, R. and Navathe, S.B. *Fundamentals of Database Systems*. 7th edn. Boston: Pearson, 2016.
- [4] Foster, Elvis, and Shripad Godbole. *Database systems: a pragmatic approach*. Auerbach Publications, 2022.
- [5] Gilad, Amir, et al. "Understanding queries by conditional instances." *Proceedings of the 2022 International Conference on Management of Data*. 2022.
- [6] Hardini, Marviola, et al. "Application of database normalization in increasing data storage efficiency." *International Transactions on Artificial Intelligence* 3.2 (2025): 201-211.
- [7] Hosen, Mohammed Shahadat, et al. "Data-driven decision making: Advanced database systems for business intelligence." *Nanotechnology Perceptions* 20.3 (2024): 687-704.
- [8] Iacob, Ionel, and Cezar Mihălcescu. "Triggers In Oracle Sql: Automation And Integrity In Economic Database Management." *Journal of Information Systems&Operations Management* 19.2 (2025): 246-259.
- [9] Imbulpitiya, Asanthika, Jacqueline Whalley, and Mali Senapathi. "A Qualitative Study of Novice Experiences with Database Normalisation." *Proceedings of the 26th Australasian Computing Education Conference*. 2024.
- [10] Karwin, Bill. *SQL Antipatterns, Volume 1: Avoiding the Pitfalls of Database Programming*. The Pragmatic Programmers LLC, 2022.

- [11] Mintrop, Rick. *Design-based school improvement: A practical guide for education leaders*. Harvard Education Press, 2020.
- [12] Özcan, Fatma, et al. "State of the art and open challenges in natural language interfaces to data." *Proceedings of the 2020 ACM SIGMOD international conference on management of data*. 2020.
- [13] Pandey, Piyush, et al. "Ensuring Data Accuracy in Text-to-SQL Systems: A Comprehensive Validation Framework." *INTERNATIONAL JOURNAL* 72.12 (2024): 17-24.