

# **Introduction to Data Modelling and Databases Project March 2023**

MICHAL BABIK  
x22126180

CIAN O CONNOR X22109668

JOHN WARD  
X22109668

## Table of Contents

Overview .....	3
1. Introduction .....	3
Proposal .....	3
ERD Diagram Development Process .....	4
ERD Progress .....	5
2. Implementation.....	5
Tables Creation.....	5
Data Population .....	6
Queries.....	6
3. Downloading the Project Files .....	7
4. Conclusion .....	8

# Pharmacy Database

## Overview

This report presents the design, implementation, and administration of a comprehensive pharmacy database management system, focused on effectively managing data related to suppliers, medications, customers, staff, prescriptions, purchases, and activity logs. Developed using MySQL Workbench, the system employs advanced SQL programming to offer valuable reporting and insights.

Throughout the project's development, our team utilized GitHub for version control and collaboration. Each team member committed their work to a shared repository, allowing for seamless collaboration and tracking of changes made to the database and associated files. This approach ensured that the project maintained a consistent history of modifications, offering a platform for addressing potential issues efficiently. By combining a well-structured database design with effective collaboration, our team has delivered a pharmacy database management system that meets the organization's current and future needs.

To access the GitHub repository for this project, please click the following link: [Project Repository](#)

## 1. Introduction

In the fast-paced and highly regulated pharmaceutical industry, effective data management is essential for maintaining accuracy, compliance, and profitability. A pharmacy database management system is an invaluable tool for pharmacists, enabling them to track inventory, sales, and customer records with ease. By leveraging the latest technologies and best practices in database design, a well-structured pharmacy database management system can streamline workflows, reduce errors, and improve patient care. The aim of this project is to create a robust, scalable, and user-friendly pharmacy database management system that can meet the diverse needs of pharmacists and pharmacy owners. Whether it's tracking prescription orders, storing customer records, or allowing access to supplier information, this system will provide a comprehensive solution to optimize operations and drive success.

## Proposal

We identified the key features that a pharmacy database should have effective inventory management, user-friendly interaction, and enhanced customer care. We also brainstormed various queries that would enhance the functionality of the database, allowing pharmacists to manage stock, prescriptions, and customer details. Through effective communication and collaboration, we were able to pool our knowledge and ideas to come up with a comprehensive proposal that meets the requirements of the college project. The process was challenging, but also rewarding, as we developed our skills in teamwork, research, and critical thinking.

## **ERD Diagram Development Process**

The development of the ERD diagram for our pharmacy database management system involved a collaborative approach to ensure a simple yet effective design.

Below is a quick summary of the steps we took as a team to create the ERD diagram:

### **Idea Generation:**

- Before our scheduled meeting, each team member conducted research and brainstormed ideas for the tables and relationships they wanted to include in the database design.

### **Team Meeting:**

- During our meeting, we discussed the ideas each team member had gathered and collectively decided on the six initial tables and their relationships for the database.

### **ERD Diagram Creation:**

- One team member was appointed to create the ERD diagram, while the other members provided input and assistance via group chat.

### **Modifications:**

- After the initial ERD diagram was completed, the team reviewed the design and made minor changes to data types and relationships to refine the model.

### **Addition:**

- The team decided to include a 7th table, named Activity Log, to track staff activity and enhance the database's security features.\*

By following this collaborative process, we successfully designed an ERD diagram that captures the necessary entities, relationships, and attributes for our pharmacy database management system. The addition of the Activity Log table provides an extra layer of security, ensuring that the database meets the organization's requirements for data management and security.

## ERD Progress

To provide a visual representation of our progress in developing the ERD diagram, we have included links to the initial and final versions of the diagram. Click on the links below to view the evolution of our ERD diagram throughout the development process.

- [First ERD Diagram](#)
- [Final ERD Diagram](#)

The ERD defines relationships between the entities with cardinality and participation constraints. Primary keys and foreign keys are outlined to ensure data integrity and referential integrity.

## 2. Implementation

The pharmacy database was implemented using MySQL Workbench, with tables created based on the entities identified in the ERD. Each table includes primary and foreign key constraints to ensure data integrity. Throughout the implementation process, proper naming conventions and adherence to database normalization rules were followed.

### Tables Creation

The table creation process began with a scheduled team meeting to discuss the assignment of tables to each team member and determine the appropriate data types for the attributes. The tables were evenly distributed among the team:

- Cian: Medication and Supplier tables
- Michael: Purchase, Staff, and Activity Log tables
- John: Customer and Prescription tables

The database creation script source can be accessed through the following link:

- [Click here to view the dbp.tables.sql file](#)

## Data Population

After finalizing the database tables script, the team moved on to data population. Initially, we anticipated this step to be relatively simple, as it involved creating an INSERT statement for each table and adding values. The team agreed on 50 entries per table to ensure sufficient data for running queries later in the project.

However, when executing the data population script, we encountered various issues with the INSERT statements, such as misaligned foreign key constraints, primary key errors, and NOT NULL value errors. These challenges were compounded by the fact that each team member had their own style of inserting data, leading to inconsistencies.

To resolve these issues, we reduced the number of entries per table to 20 and then adjusted to 50 and focused on refining the INSERT statements. In hindsight, assigning the data population task to a single team member would have been more efficient, as it would have ensured a consistent approach throughout the process.

With the revised data population script, we successfully populated the tables, eliminating errors related to foreign key constraints, primary keys, and NOT NULL values. This experience highlighted the importance of clear communication and consistent formatting when working on a collaborative project.

The data population script can be accessed via the link below:

- [Click here to view the dbp.data.sql file](#)

## Queries

Our team developed a total of 12 complex queries using WHERE, GROUP BY, and JOIN clauses etc. as outlined in the project description. Each team member contributed four queries, providing valuable insights and information for the organization's current and future needs.

The development of these queries was discussed during a team meeting where all members shared their ideas and research from lecture slides. During the meeting, we decided to work on a few queries together and then evenly distribute the remaining queries among team members for further development and upload to our GitHub repository.

Here are two examples of the queries we created:

This query retrieves the date and time of each transaction by the staff name and the action.

```
select al.transaction_id, al.date_time, al.action, s.Name as Staff_Name from
activity_log al
inner join Staff s on al.user_id = s.Staff_ID where
al.date_time >= date_sub(now(), interval 6 month) order
by al.date_time asc;
```

This query retrieves the names and email addresses of the staff members who were involved in processing purchases on a specific date.

```
select distinct s.Name, s.Email from  
Staff s  
join Purchase p on s.Staff_ID = p.Staff_ID where  
p.Date_Purchased = '2023-04-15';
```

To run the queries, users must have access to MySQL Workbench or a compatible database management tool. The SQL queries provided can be executed to retrieve relevant information as needed.

The remaining queries source, along with descriptions of their purpose, can be accessed via the link below:

- [Click here to view the dbp.queries.sql file](#)

By carefully planning and collaborating on the development of these queries, our team was able to create a comprehensive set of tools for extracting valuable information from the pharmacy database.

### **3. Downloading the Project Files**

To download all the project files as a ZIP archive, click on the link below:

- [Download Project Files](#)

After downloading the ZIP file, extract its contents to a folder on your computer. You will find all the necessary SQL scripts and other project files within the extracted folder.

Now you have successfully downloaded, imported, and executed the database tables script, database data population script, and database queries script.

## 4. Conclusion

From the beginning, the team worked together to determine the objectives of this project, determine project scope, and set a timelines for completion. Each member of the team brought their own vision and skillset to the project, which allowed for a comprehensive and well-rounded approach to database development. As this was a first for us, each member was willing to accept ideas and work together with ease. Collaboration was key during the design and development phase, as the team members worked together to identify user requirements, design the database schema. Regular communication and feedback loops were established throughout the project to ensure that all team members were aligned and working towards the same objectives. Some of the main tools of communication included GitHub, Microsoft Teams and WhatsApp. Through effective collaboration, the team was able to successfully complete the project on time and to a high standard, resulting in a robust and user-friendly pharmacy database management system.

In conclusion, the pharmacy database management system developed for this project offers a powerful solution to the data management challenges facing pharmacists and pharmacy owners today. By providing a centralized and streamlined system for tracking inventory, sales, and customer records, this database can help pharmacies save time, reduce errors, and improve patient care. The system is designed with best practices in database design and is capable of meeting the complex needs of the pharmaceutical industry. Our aim was to give pharmacists a system which provides a comprehensive data management solution that can help pharmacies optimize operations and achieve their business goals. We are confident that this database will make a significant impact on the work life of its users, and we look forward to seeing its benefits in action.