Final Report: Pharmacy Database

Course Title: CSC 33600

Instructor: Denis Khryashchev

Date: 05/17/23

Authors:

Markus Chmiel

Myesha Mahazabeen

Najia Jahan

Pharmacy Relation Database

a) Motivation. Why did you pick this database?

Motivation: We chose to create a relational database for a pharmacy because it is a common and practical scenario where a database can be utilized effectively. A pharmacy typically deals with patients, prescriptions, and medications, which are interconnected entities requiring proper management and organization. By designing a database for a pharmacy, it allows for efficient storage, retrieval, and manipulation of data related to patients, prescriptions, and medications.

b) List the authors' roles: who was responsible for what part of the project.

List of Authors:

Markus Chmiel: Code and functions implementation.

Myesha Mahazabeen: Code analysis, csv data generation, and report writing.

Najia Jahan: Code analysis, csv data generation, and report writing.

c) Logic of the database you created: relations, their attributes, constraints, etc. Provide some toy examples.

Logic of Database:

1. **Relations:**

- ***Patient:*** Stores information about patients, such as their unique identifier(patient_id: Primary key), first name, last name, patient number, and their insurances' information.
- ***Prescriptions:*** Stores information about prescriptions issued to patients. It includes attributes such as prescription ID (Primary Key), patient ID (foreign key referencing the Patient relation), medication ID (foreign key referencing the Medications relation), dosage, prescribed date and name of prescribing doctor.

- ***Medications*:** Stores information about the medications available at the pharmacy. It includes attributes like medication ID (Primary key), medicine name, medicine's unit price, medicine's expiration date.

2. Constraints:

- ***Primary Keys:*** Each relation should have a primary key that uniquely identifies each record. For example, Patient ID in the Patient relation, Prescription ID in the Prescriptions relation, and Medication ID in the Medications relation are the Primary Keys for our Pharmacy database.
- ***Foreign Keys:*** Appropriate foreign keys should be used to establish relationships between the tables. For example, the patient ID in the Prescriptions relation references the Patient ID in the Patient relation, and the medication ID in the Prescriptions relation references the Medication ID in the Medications relation.
- ***Other constraints:*** Additional constraints such as data types, nullability, unique constraints, etc. are also added as per the specific requirements of the pharmacy application.

Toy examples: Patient relation:

Patient ID	**First Name**	**Last Name**	**Phone Number**	**Insurance**
1	Shree	Philipp	215-456-7890	Aetna

Prescriptions relation:

Prescription	**Patient id**	**Medication**	**Dosage**	**Date**	**Doctor**

id		id		Prescribed	
1000	1	100	952	11/5/23	Hildegaard

Medications relation:

Medication id	Name	Unit Price	Expiration Date
321	Ibuprofen	30	9/21/2023

d) List the modules that the program includes (headers, helper functions, etc.) and explain what functions they have. Explain what tables / relations your database is built on.

List of Modules:

Helper Functions:

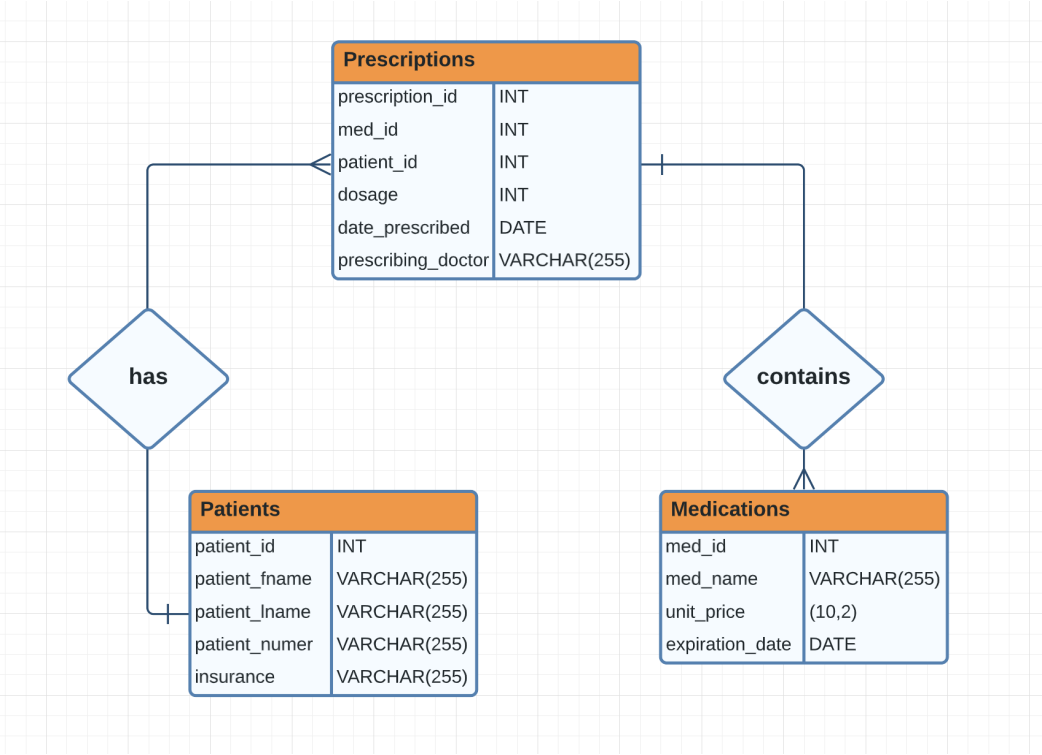
- def SelectQ: Takes a table name, an identifier column, and a value, and selects the row where the identifier column equals the value from the table. Returns the selected row.
- def DeleteQ: Takes a table name, an identifier column, and a value, and deletes the row where the identifier column equals the value from the table.
- def InsertQ: Takes a table name and a list of values to insert into the table. Inserts the values into the table and returns the row ID of the newly inserted row.
- def UpdateQ: Takes a table name, a column name, a new value, an identifier column name, and an old value. Updates the row in the table where the identifier column equals the old value with the new value.
- def executeQ: Takes a SQL query and executes and prints the result.

The database includes three tables: Medications, Patients, and Prescriptions.

- Medications table has columns for med_id, med_name, unit_price, and expiration_date.
- Patients table has columns for patient_id, patient_fname, patient_lname, phone_number, and insurance.

- The Prescriptions table has columns for prescription_id, med_id, patient_id, dosage, date_prescribed, and prescribing_doctor. It also has foreign key constraints that reference the med_id and patient_id columns in the Medications and Patients tables, respectively.

e) ER diagrams.



f) Conclude with what you successfully managed to implement, and what difficulties you faced working on the project. What would you change in your code / algorithm to improve the program?

We could include another table Inventory that would store information about the pharmacy's inventory of each medication, including the current quantity of each medication in stock and the date when the medication was last restocked. We could link it with the Medications table through "med_id" .