1. User Documentation

1.1 Application Overview

The **Medication Tracking System** is a Java-based application designed to manage patients, doctors, medications, and prescriptions in a pharmacy setting. It allows users to:

- Add, delete, and edit patients, doctors, and medications.
- Process new prescriptions.
- Check for expired medications.
- Generate system reports.
- Restock medications.

1.2 Classes and Their Working

Here’s an explanation of the classes in the application:

1. **Person**:
   - Base class for Patient and Doctor.
   - Contains common attributes like id, name, age, and phoneNumber.
2. **Patient**:
   - Extends Person.
   - Manages a list of medications and prescriptions for the patient.
3. **Doctor**:
   - Extends Person.
   - Manages a list of patients and prescriptions written by the doctor.
4. **Medication**:
   - Represents a medication with attributes like id, name, dose, quantityInStock, and expiryDate.
5. **Prescription**:
   - Represents a prescription with attributes like id, doctor, patient, medication, and prescriptionExpiry.
6. **MedicationTrackingSystem**:
   - Manages the core functionality of the system.
   - Contains lists of patients, doctors, medications, and prescriptions.

   - Provides methods for adding, deleting, and searching for entities.
7. **MainMenu**:
   - Provides a command-line interface for interacting with the system.
   - Handles user input and calls appropriate methods in MedicationTrackingSystem.

1.3 How to Start/Access the Application

1. **Prerequisites**:
   - Install Java Development Kit (JDK) 8 or higher.
   - Ensure the java and javac commands are available in your terminal.
2. **Steps to Run**:
   - Compile the project:

     ```
     javac MedicationTracking/*.java
     ```

   - Run the application:

     ```
     java MedicationTracking.MainMenu
     ```

3. **Using the Application**:
   - Follow the on-screen menu to perform operations like adding patients, doctors, medications, and prescriptions.

1.4 Class Diagram

```
+------------------+       +------------------+       +------------------+
|     Person       |       |    Medication    |       |  Prescription    |
+------------------+       +------------------+       +------------------+
| - id: String     |       | - id: String     |       | - id: String     |
| - name: String   |       | - name: String   |       | - doctor: Doctor|
| - age: int       |       | - dose: String   |       | - patient: Patient
| - phoneNumber:   |       | - quantityInStock: int | - medication: Medication
|   String         |       | - expiryDate: Date    | - prescriptionExpiry: Date
+------------------+       +------------------+       +------------------+
        ^                          ^                          ^
        |                          |                          |
```

```
       |                        |                        |
+-----------------+       +-----------------+       +-----------------+
|    Patient      |       |    Doctor       |       | MedicationTrackingSystem |
+-----------------+       +-----------------+       +-----------------+
| - medications:  |       | - specialization: String| - patients: List<Patient> |
|   List<Medication>|     | - patients: List<Patient>| - doctors: List<Doctor>  |
| - prescriptions: |    +-----------------+       | - medications: List<Medication> |
|   List<Prescription>|                       | - prescriptions: List<Prescription> |
+-----------------+                         +-----------------+
```

2. Development Documentation

2.1 Javadocs

Generate Javadocs for your project using the following command:

```
javadoc -d docs MedicationTracking/*.java
```

This will create HTML documentation in the docs folder. Include this folder in your project repository.

2.2 Source Code Directory Structure

```
MedicationTracking/
├── MainMenu.java
├── MedicationTrackingSystem.java
├── Person.java
├── Patient.java
├── Doctor.java
├── Medication.java
├── Prescription.java
└── docs/ (Javadocs)
```

2.3 Build Process

1. Compile the Project:

```
javac MedicationTracking/*.java
```

2. Run the Application:

```
java MedicationTracking.MainMenu
```

2.4 Compile-Time Dependencies

- **JDK 8 or higher**: Required to compile and run the application.
- **No external libraries**: The project uses only standard Java libraries.

2.5 Development Standards

- Follow Java naming conventions (e.g., camelCase for variables, PascalCase for classes).
- Use meaningful variable and method names.
- Include comments for complex logic.
- Write unit tests for critical functionality (optional but recommended).

2.6 Database Design (Theoretical)

If the application were to use a database, the **ERD (Entity-Relationship Diagram)** would look like this:

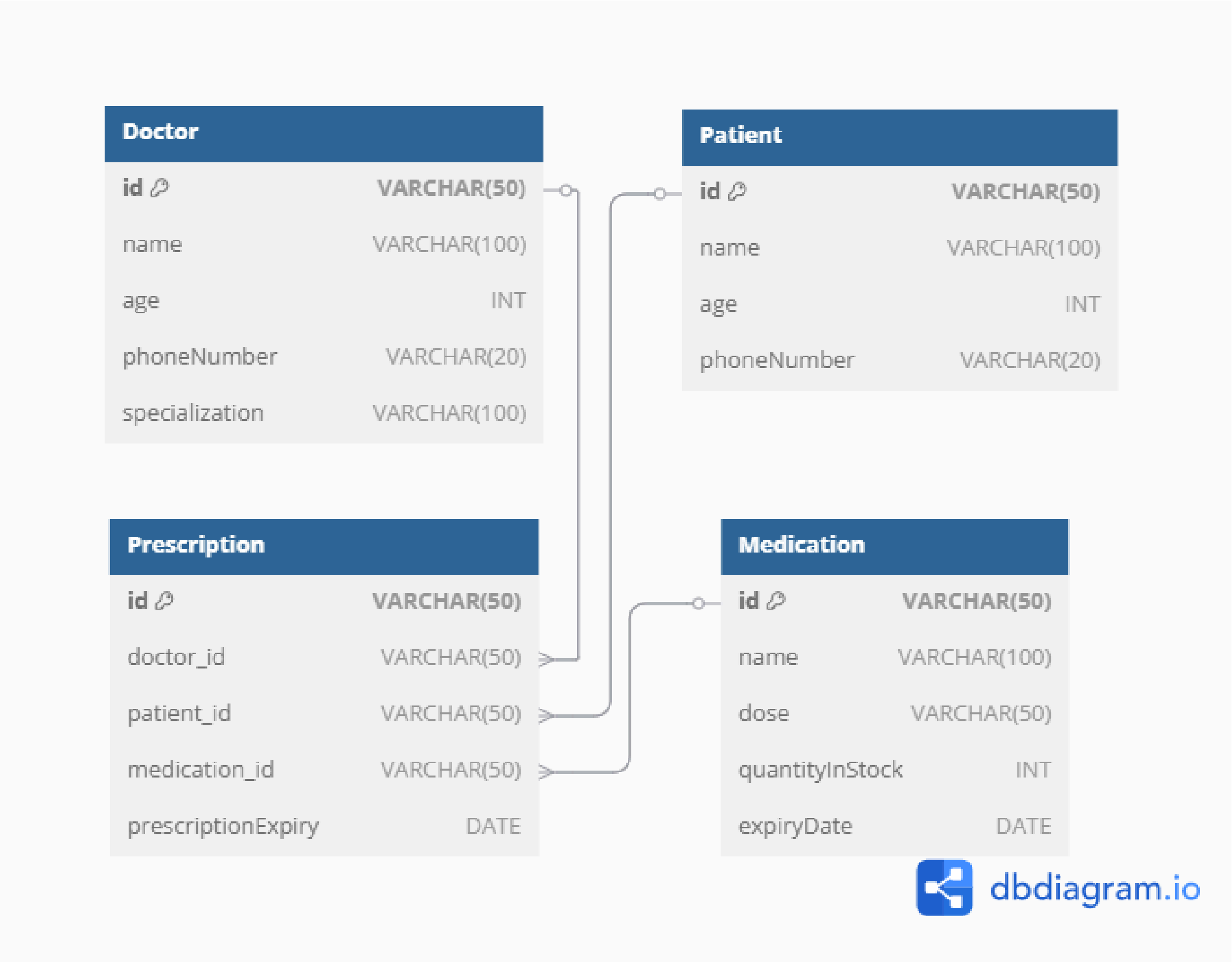
Entities:

1. **Patient**:
   - id (Primary Key)
   - name
   - age
   - phoneNumber
2. **Doctor**:
   - id (Primary Key)
   - name
   - age
   - phoneNumber
   - specialization
3. **Medication**:
   - id (Primary Key)
   - name

    - dose
    - quantityInStock
    - expiryDate

4. **Prescription**:
    - id (Primary Key)
    - doctor_id (Foreign Key)
    - patient_id (Foreign Key)
    - medication_id (Foreign Key)
    - prescriptionExpiry

ERD :



Doctor	
id	**VARCHAR(50)**
name	VARCHAR(100)
age	INT
phoneNumber	VARCHAR(20)
specialization	VARCHAR(100)

Patient	
id	**VARCHAR(50)**
name	VARCHAR(100)
age	INT
phoneNumber	VARCHAR(20)

Prescription	
id	**VARCHAR(50)**
doctor_id	VARCHAR(50)
patient_id	VARCHAR(50)
medication_id	VARCHAR(50)
prescriptionExpiry	DATE

Medication	
id	**VARCHAR(50)**
name	VARCHAR(100)
dose	VARCHAR(50)
quantityInStock	INT
expiryDate	DATE

2.7 GitHub Repository

- **Repository Structure:**

```
MedicationTracking/
├── src/
│   └── MedicationTracking/
│       ├── MainMenu.java
│       ├── MedicationTrackingSystem.java
│       ├── Person.java
│       ├── Patient.java
│       ├── Doctor.java
│       ├── Medication.java
│       └── Prescription.java
├── docs/ (Javadocs)
└── README.md
```

- **How to Clone:**

```
git clone <repository-url>
```

3. Deployment Documentation

3.1 Installation Manual

1. **Prerequisites:**
   - Install JDK 8 or higher.
   - Ensure `java` and `javac` are available in your terminal.
2. **Steps to Deploy:**
   - Clone the repository:

     ```
     git clone <repository-url>
     ```

   - Navigate to the project directory:

     ```
     cd MedicationTracking
     ```

   - Compile the project:

     ```
     javac MedicationTracking/*.java
     ```

   - Run the application:

```
java MedicationTracking.MainMenu
```

3. **Using the Application:**
    - **Follow the on-screen menu to perform operations.**