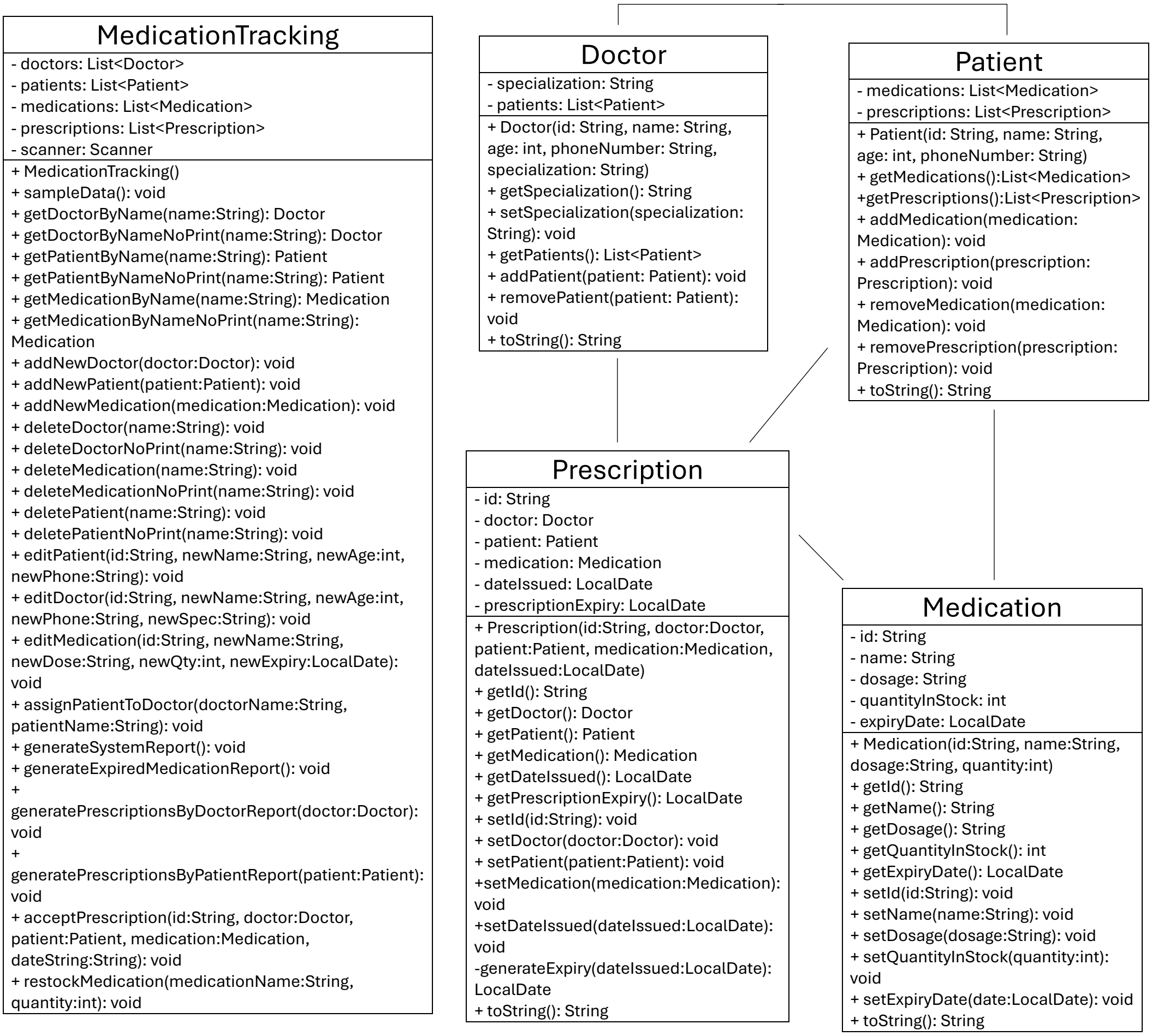
2025

User Documentation

JAVA

CAMERON BOYER & KASSAUNDRA FEQUET

Class Diagram for the Medication Tracking System



Class Relationships

Inheritance:

- Doctor → inherits from Person
- Patient → inherits from Person

Associations:

Person
└── (abstract superclass of Doctor and Patient)

Doctor
├── Patients (1 Doctor → Many Patients)
└── Prescriptions (1 Doctor → Many Prescriptions)

Patient
├── Medications (1 Patient → Many Medications)
├── Prescriptions (1 Patient → Many Prescriptions)
└── Doctor (1 Patient → 1 Doctor)

Medication
├── Prescriptions (1 Medication → Many Prescriptions)
└── Patients (1 Medication → Many Patients)

Prescription
├── references 1 Doctor
├── references 1 Patient
└── references 1 Medication

MedicationTracking
├── contains Doctor
├── contains Patient
├── contains Medication
└── contains Prescription

Program Overview

The Medication Tracking System is a Java-based console application that simulates a small healthcare management system. It keeps track of doctors, patients, medications, and prescriptions using object-oriented programming principles. The system allows doctors to manage their patients and issue prescriptions, while also monitoring medication stock levels and expiry dates.

The program automatically loads sample data and provides options to:

- Add, edit, or remove doctors, patients, and medications
- Assign patients to doctors
- Create and view prescriptions
- Generate system reports and lists of expired medications

It demonstrates key OOP features such as:

Inheritance: Shared attributes and behavior defined in the `Person` superclass

Encapsulation: Private fields with controlled access through getters and setters

Association and Composition: Classes interact and contain references to each other to model real-world relationships

How to Run

1. Open a Java IDE (IntelliJ IDEA, VS Code, or Eclipse).
2. Clone the git repository
   - git clone https://github.com/KassFequet/Sem3-Java-Sprint
3. Navigate to project directory
   - cd Java-Sem3-Sprint
4. Compile and run
   - Java Menu
5. The console menu will display available options for interacting with the system.

Purpose

This program demonstrates the design and implementation of an object-oriented system for managing healthcare data. It serves as a tool for understanding relationships between classes and the use of inheritance, associations, and aggregation in Java.