

Requirements Engineering and Analysis Assignment #2

Community Pharmacy UML Diagrams

[Team]

GMONCI [Names]

Department of Engineering, [School]

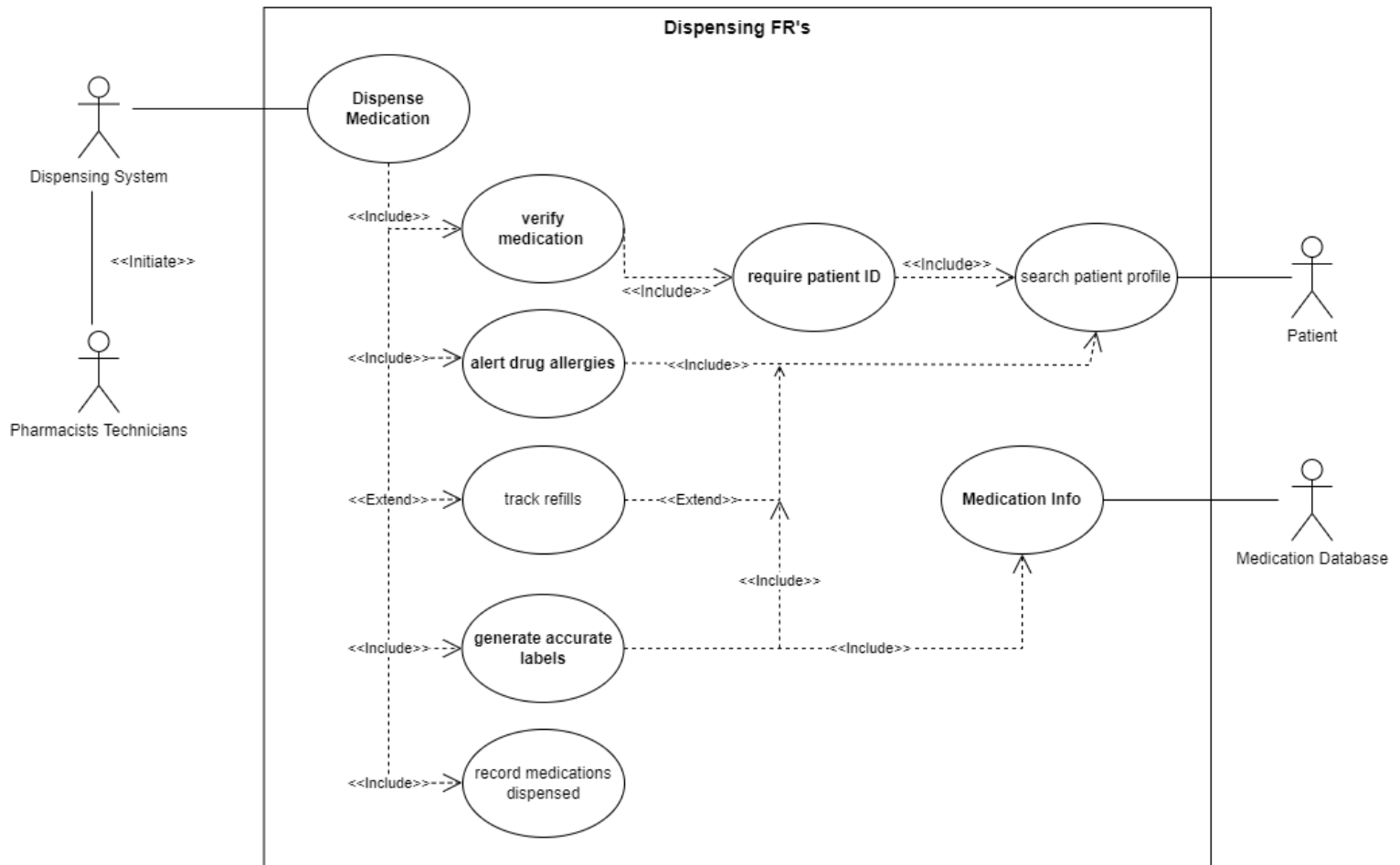
[Class]: Requirements Engineering & Analysis

[Professor]

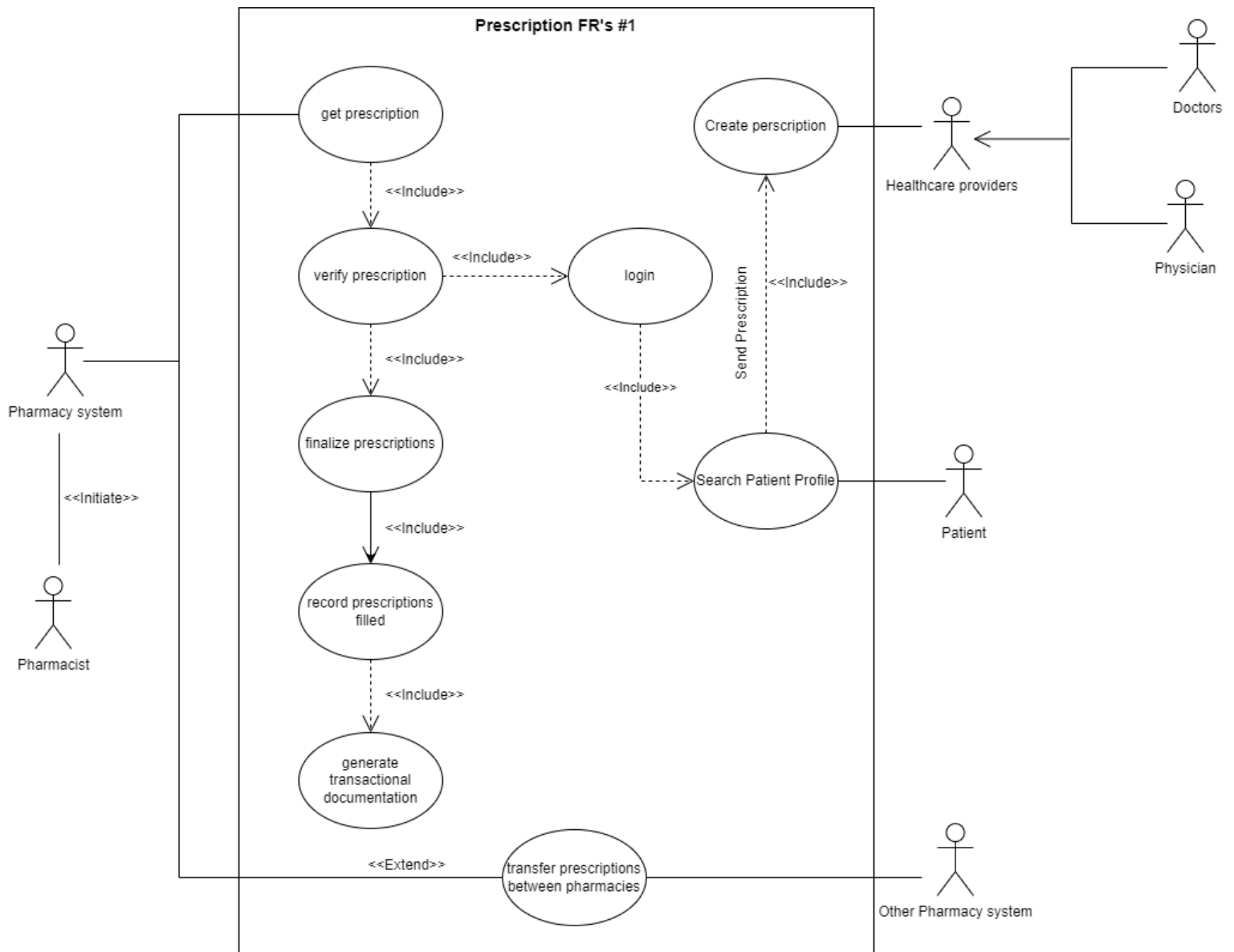
Requirements Engineering and Analysis Assignment #2

Question 1: Create Use Case diagrams based on the functional requirements you wrote in Assignment#1, for each of the following essential services of a pharmacy software system.

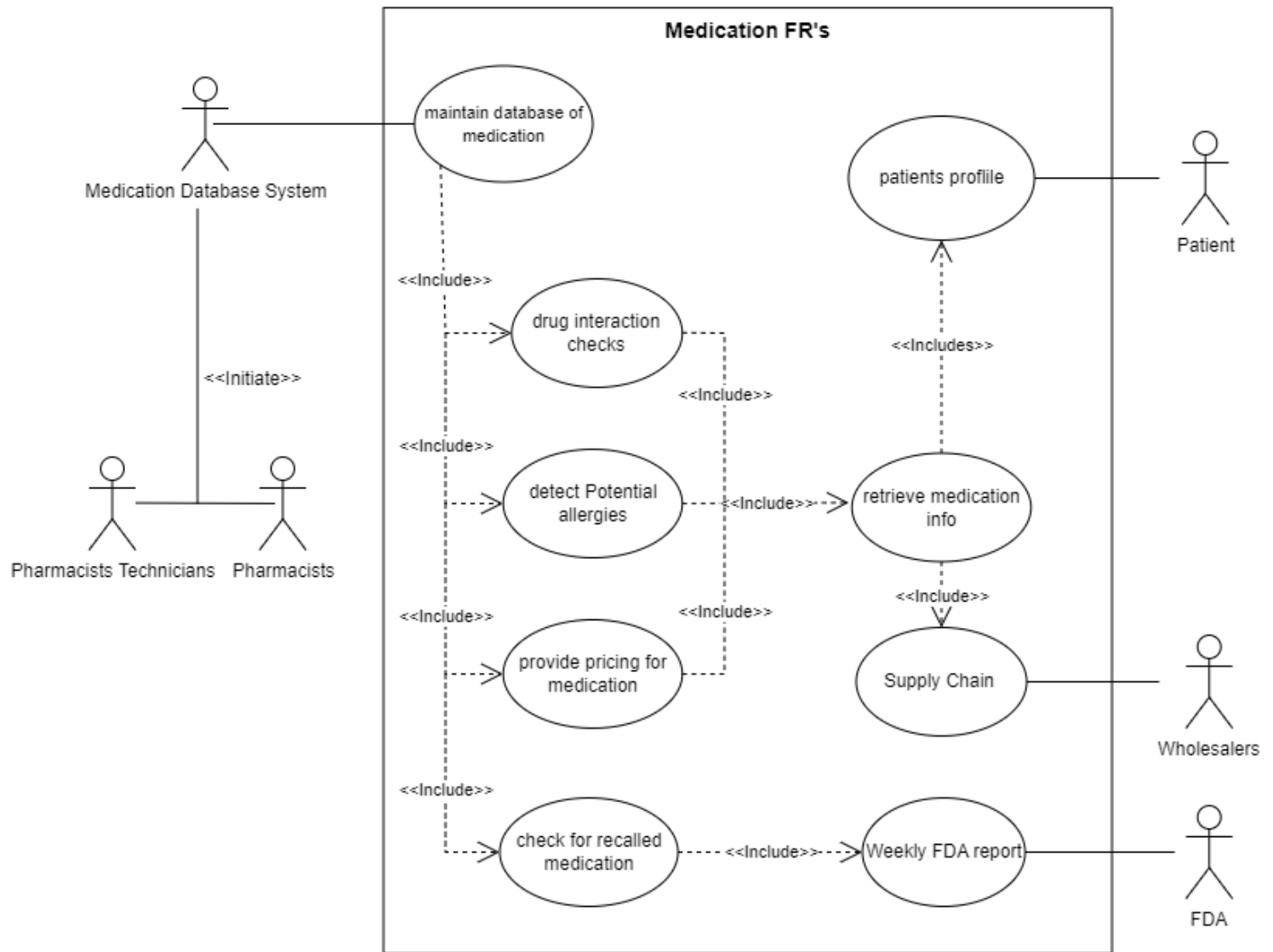
A. Dispensing:



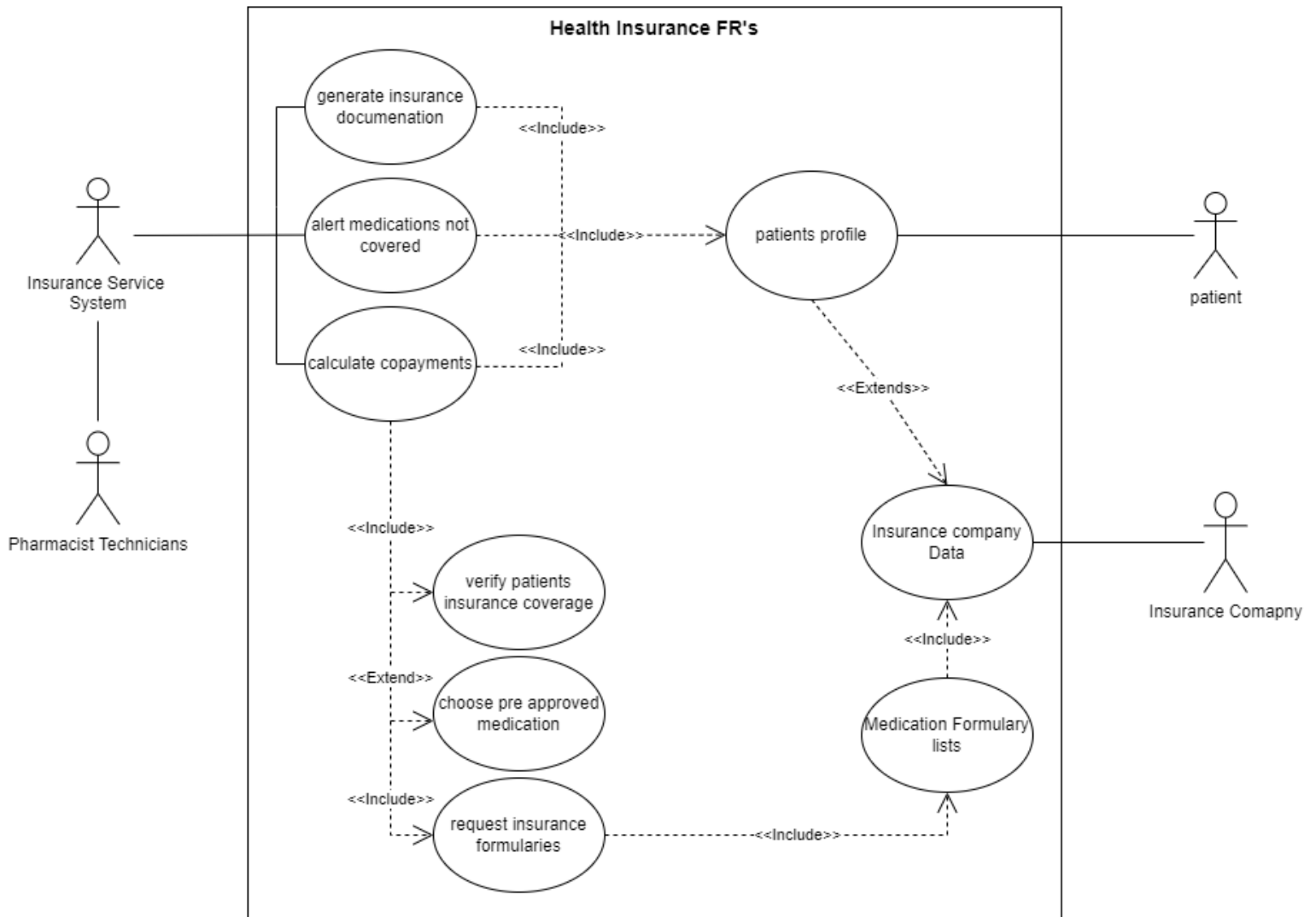
B. Prescriptions:



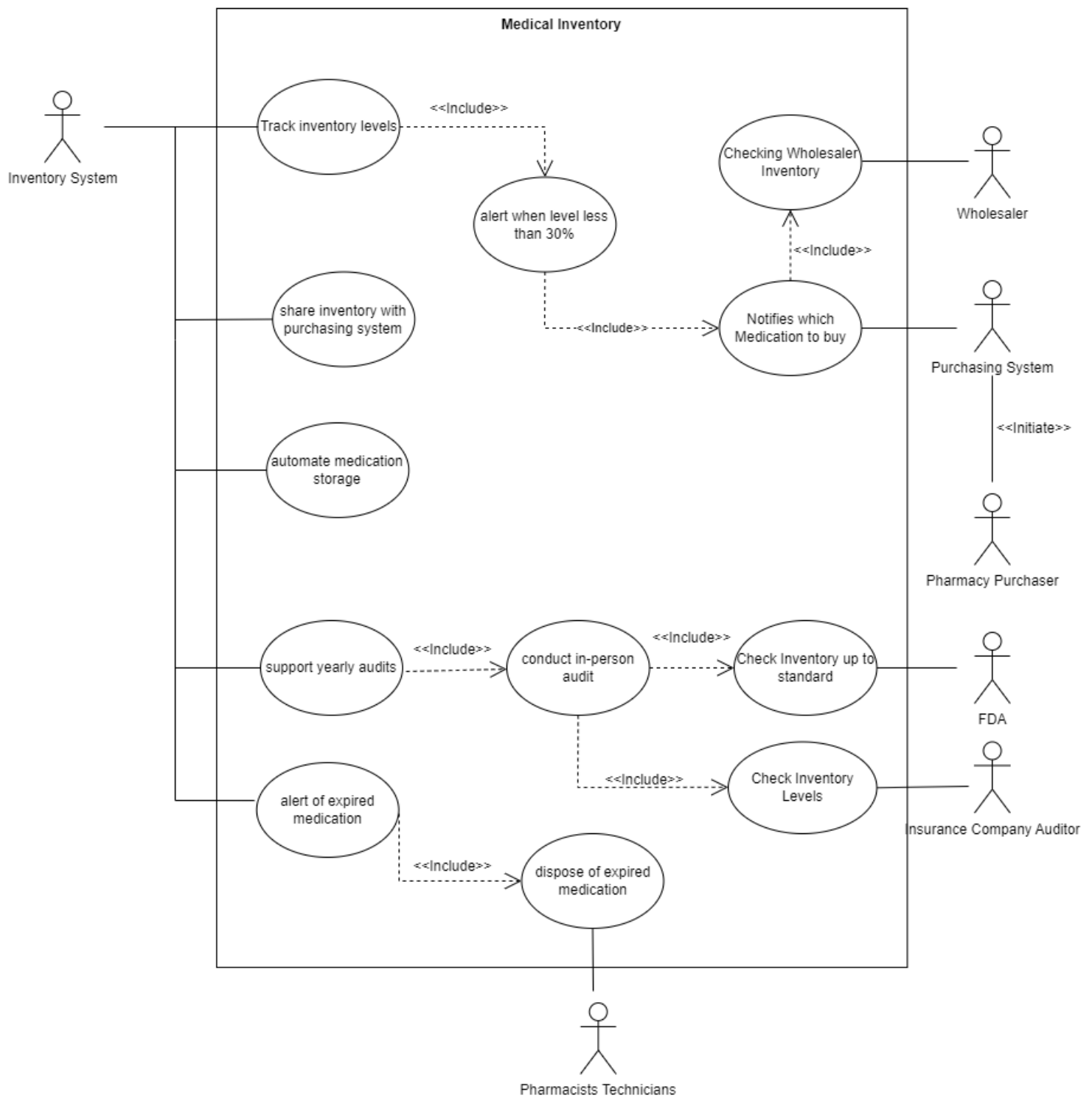
C. Medication:



D. Health Insurance:

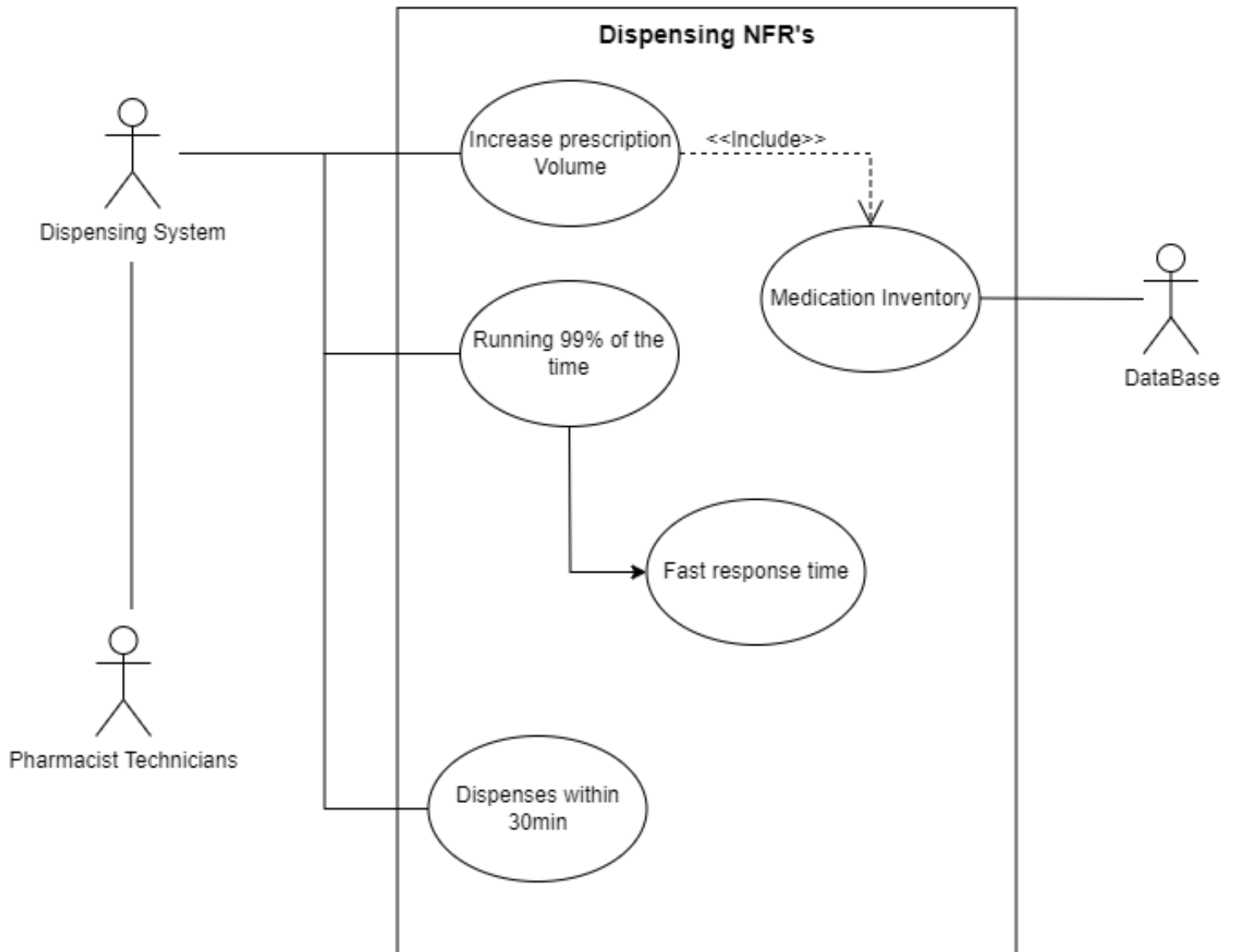


E. Medical Inventory:

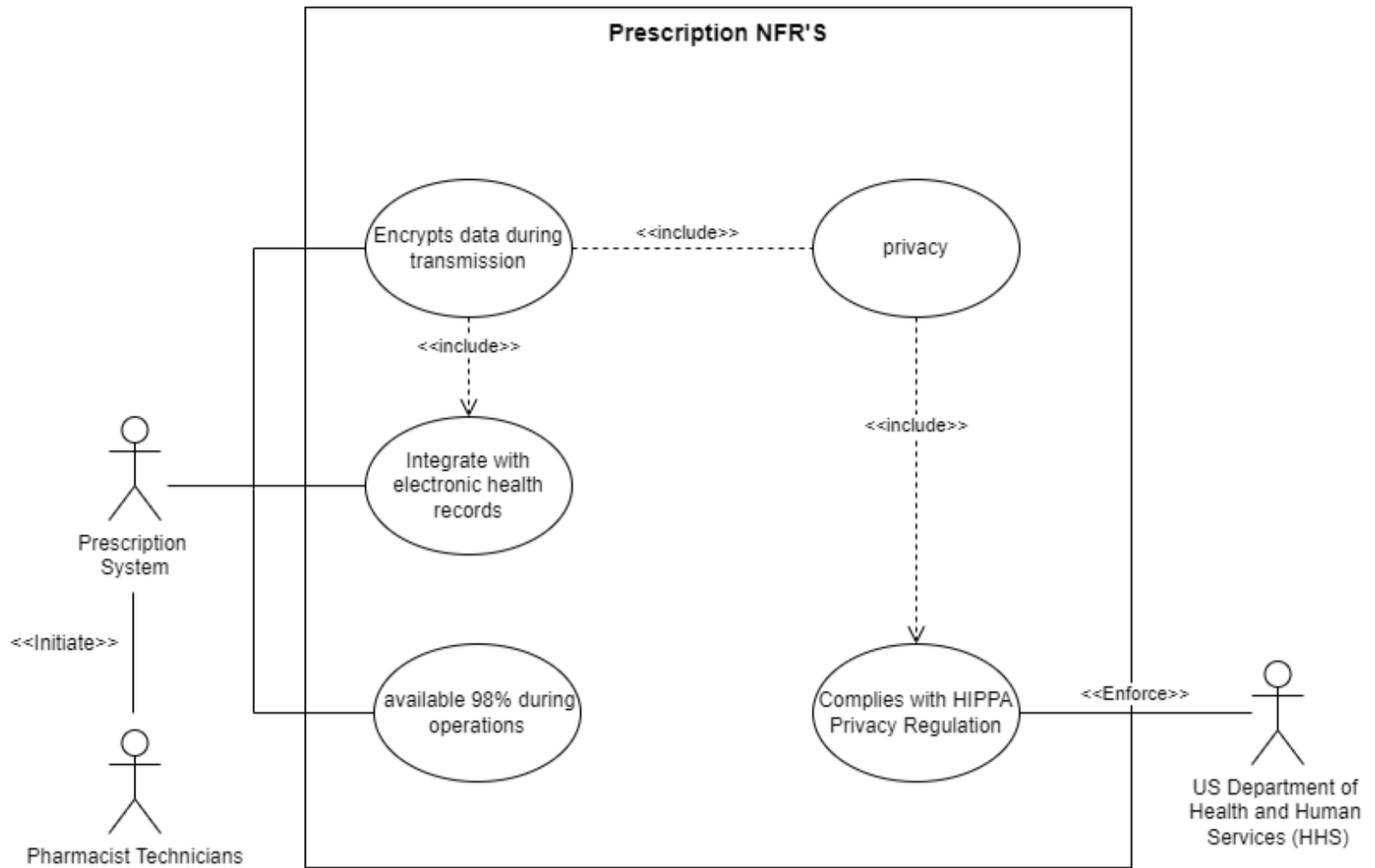


Question 2: Create Use Case diagrams based on the nonfunctional requirements you wrote in Assignment#1 for each of the following essential services of a pharmacy software system:

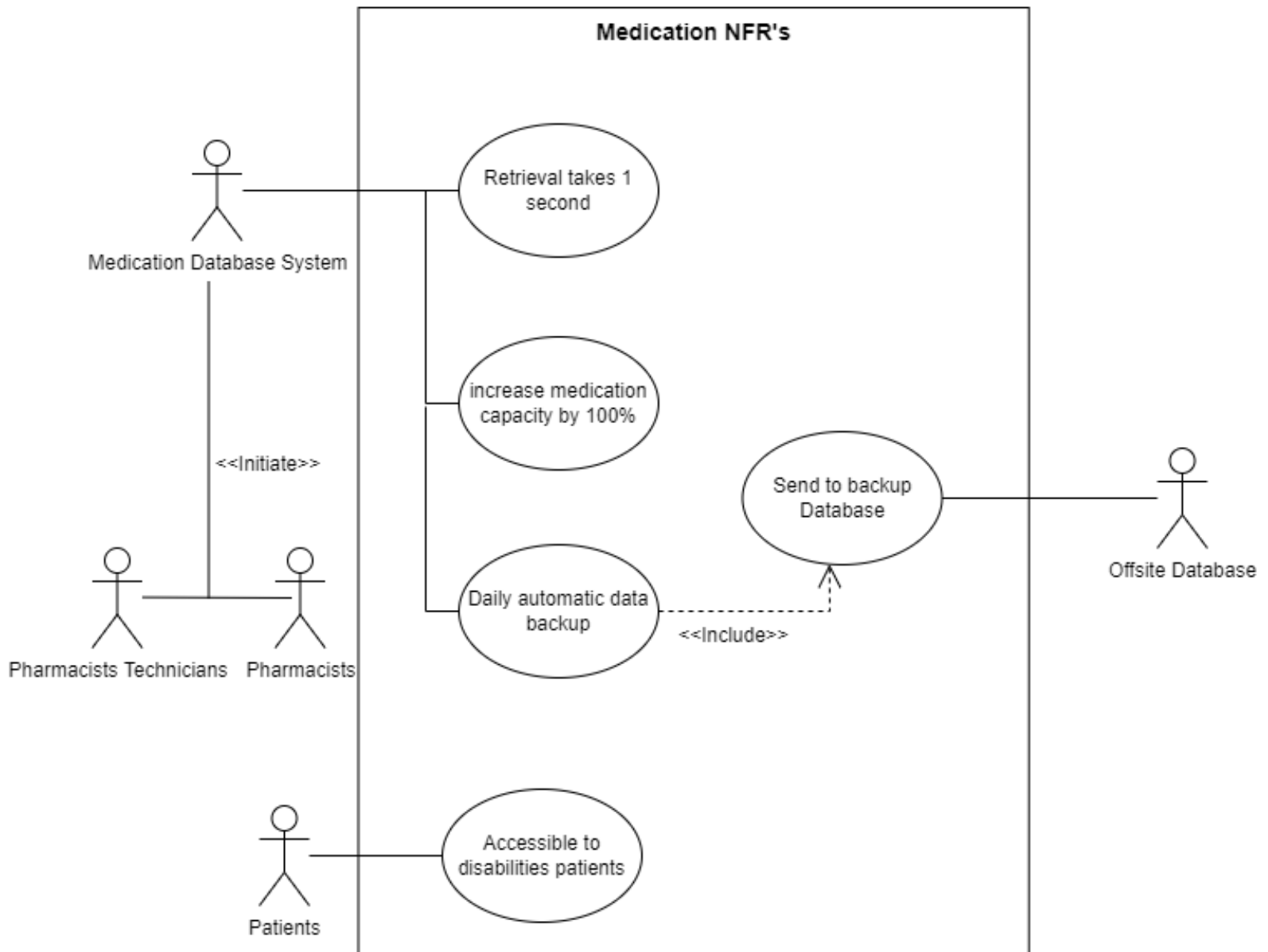
A. Dispensing:



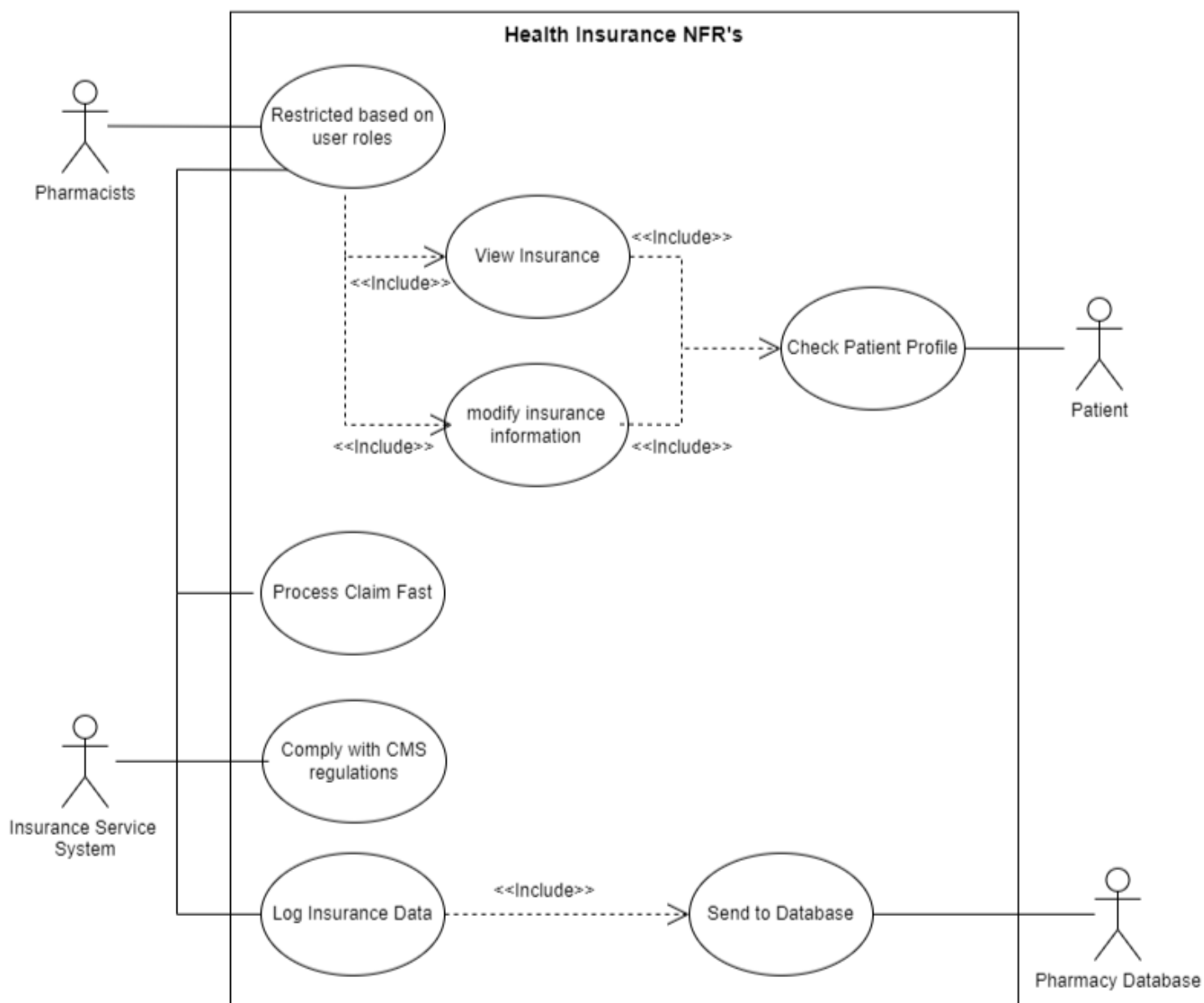
B. Prescriptions:



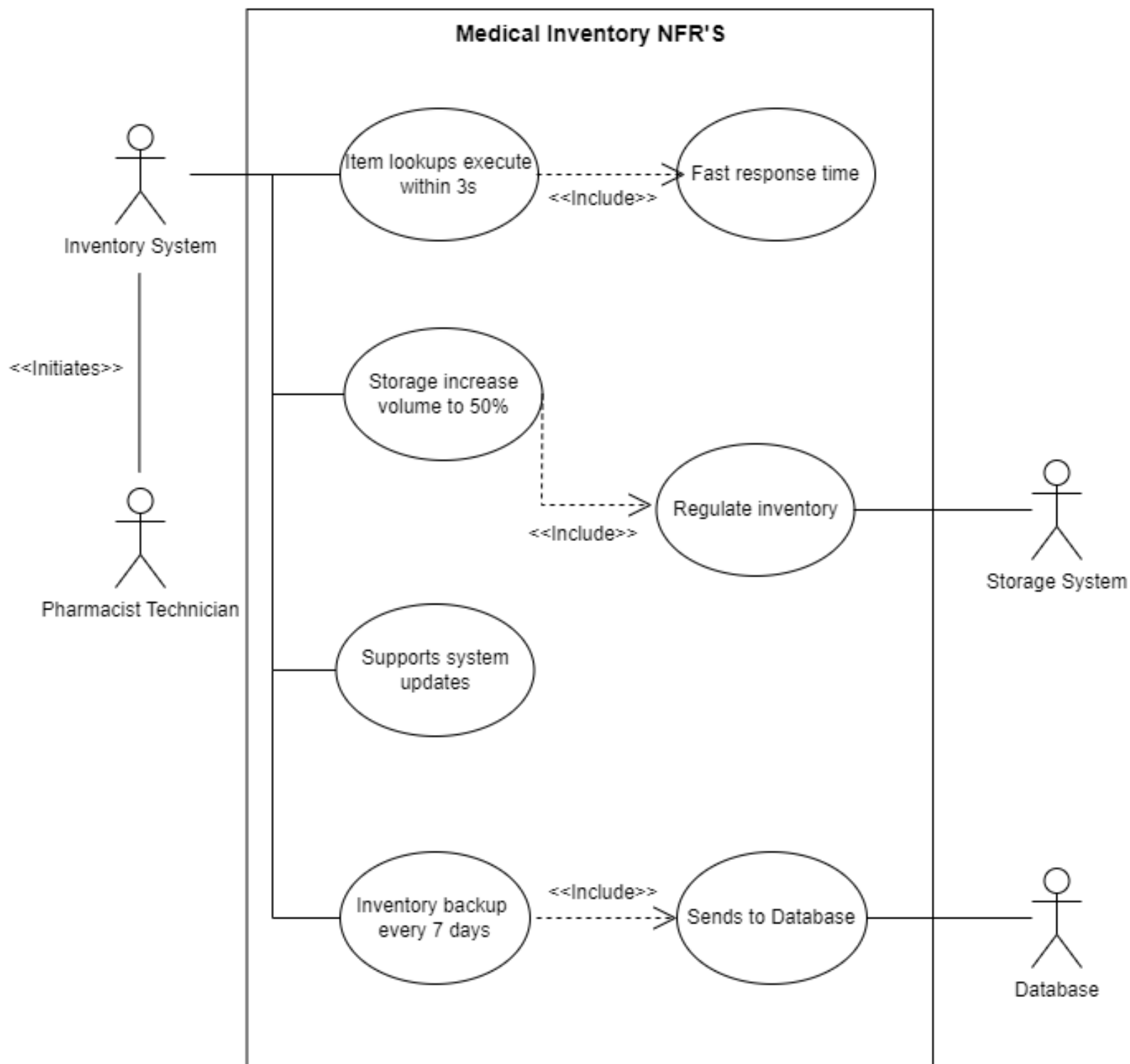
C. Medication:



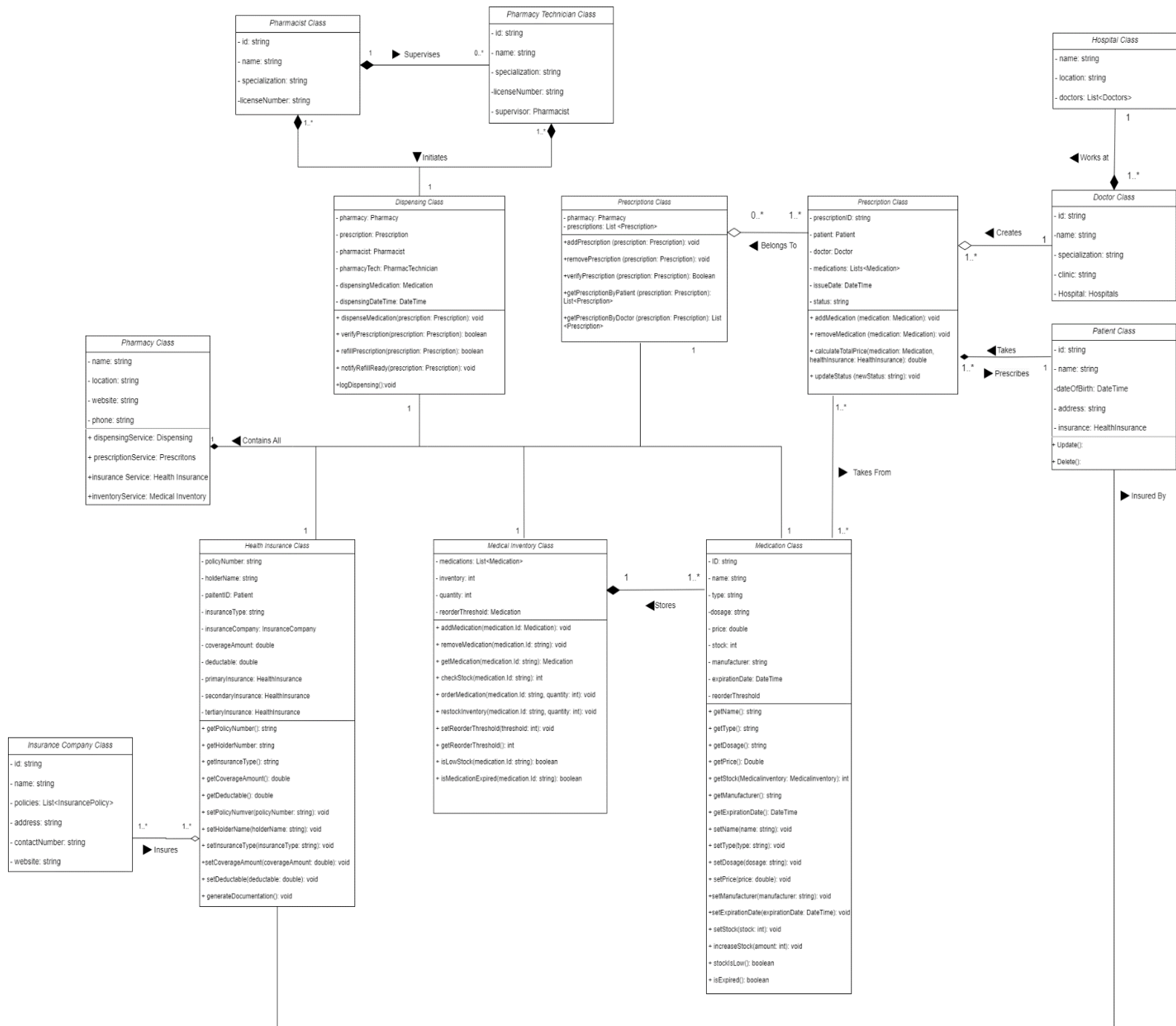
D. Health Insurance:



E. Medical Inventory:



Question 3: Create one class diagram that represents the each of the following essential services of a pharmacy software system:



Description of Class Diagram:

Pharmacy class: This class serves as the main pillar or the class diagram. It manages the main areas of insurance, dispensing, medications, inventory, and prescriptions. Has relationships with the key classes in the system.

Relations:

- Health Insurance class: *Relation: Composition; Multiplicity: 1 to 1*
- Dispensing class: *Relation: Composition; Multiplicity: 1 to 1*
- Medical Inventory class: *Relation: Composition; Multiplicity: 1 to 1*
- Prescriptions class: *Relation: Composition; Multiplicity: 1 to 1*
- Medication class: *Relation: Composition; Multiplicity: 1 to 1*

Pharmacist class: This is for an actor with the role of a pharmacist. Their information is provided by the pharmacist class. This information is used in the dispensing class. A pharmacist can act as a supervisor to none or more pharmacy technicians.

Relations:

- Pharmacy Technician class: *Relation: Composition; Multiplicity: 1 to 0...**
- Dispensing class: *Relation: Composition; Multiplicity: 1...* to 1*

Pharmacy Technician class: This is for an actor with the role of a pharmacy technician. Their information is provided by the pharmacy technician class. This information is used in the dispensing class. A pharmacy technician is supervised by a pharmacist. They are supervised by pharmacists and use the dispensing system.

Relations:

- Pharmacist class: *Relation: Composition; Multiplicity: 1 to 0...**
- Dispensing class: *Relation: Composition; Multiplicity: 1...* to 1*

Dispensing class: This class oversees dispensing medication for new prescriptions or refill prescriptions. It also verifies prescriptions, dispenses medications, and logs each dispensing action. These actions require a pharmacist or a pharmacy technician to occur. This information is retrieved from the pharmacist and pharmacy technician classes, respectively.

Relations:

- Pharmacist class: *Relation: Composition; Multiplicity: 1...* to 1*
- Pharmacy Technician class: *Relation: Composition; Multiplicity: 1...* to 1*
- Pharmacy class: *Relation: Composition; Multiplicity: 1 to 1*

Prescriptions class: This class manages prescriptions. It holds information about prescriptions such as the prescription id, patient, what medications they need, and the doctor who prescribed it. This class gets prescriptions from the prescription class by the doctor and patient. Its functionalities are to add, remove, and verify prescriptions.

Relations:

- Prescription class: *Relation: Aggregation; Multiplicity: 0... * to 1... **
- Pharmacy class: *Relation: Composition; Multiplicity: 1 to 1*

Prescription class: This class holds information about individual prescriptions, including details about the prescribing doctor and the patient. It can add and remove medications from a patient's prescription. It can also calculate the total price of the prescription. Gives prescription data to the prescriptions class. A patient can have many prescriptions. Gets Medication information from Medication class.

Relations:

- Doctor class: *Relation: Aggregation; Multiplicity: 1... * to 1*
- Patient class: *Relation: Composition; Multiplicity: 1... * to 1*
- Prescriptions class: *Relation: Aggregation; Multiplicity: 0... * to 1... **
- Medication class: *Relation: Association; Multiplicity: 0... * to 1... **

Hospital class: Represents hospitals where doctors from other hospitals can send prescriptions.

Relations:

- Doctor class: *Relation: Composition; Multiplicity: 1 to 1... **

Doctor class: Represents individual doctors who issue prescriptions. Relations with hospital class where they work. Also in relations with prescriptions class where they send patients prescriptions to.

Relations:

- Hospital class: *Relation: Composition; Multiplicity: 1 to 1... **
- Prescription class: *Relation: Aggregation; Multiplicity: 1... * to 1*

Patient class: Represents patients who receive prescriptions from the pharmacy. They are also customers of the pharmacy, who tell their primary health providers to send in their prescriptions to our pharmacy. They pay and aid in the growth of our pharmacy

Relations:

- Prescription class: *Relation: Composition; Multiplicity: : 1... * to 1*
- Health Insurance: *Relation: Association; Multiplicity: 1 to 1*

Insurance Company class: This class represents an insurance company that provides health insurance policies.

Relations:

- Health Insurance class: *Relation: Aggregation; Multiplicity: 1... * to 1... **

Health Insurance class: This class manages health insurance policies and their association with insurance companies. This class receives all insurance information such as policies from the insurance company class. Then it connects an insurance company to a patient. Generates legal documents.

Relations:

- Insurance Company class: *Relation: Aggregation; Multiplicity: 1... * to 1... **
- Pharmacy class: *Relation: Composition; Multiplicity: 1 to 1*
- Patient class : *Relation: Association; Multiplicity: 1 to 1*

Medical Inventory class: This class manages the inventory of medical supplies and medications within a pharmacy. Takes Medication information from the Medication class. It receives medication and stores it and its respective place. Monitors its expiration date and threshold. Orders more medication when its threshold is under 30%.

Relations:

- Medication class: *Relation: Composition; Multiplicity: 1 to 1... **
- Pharmacy class: *Relation: Composition; Multiplicity: 1 to 1*

Medication class: This class represents individual medications and holds information about them, including their association with medical inventory. Gets this information so that prescriptions can be accurately labeled by the dispensing system once its prescribed to a patient.

Relations:

- Medical Inventory class: *Relation: Composition; Multiplicity: 1 to 1... **
- Pharmacy class: *Relation: Composition; Multiplicity: 1 to 1*
- Prescription class: *Relation: Association; Multiplicity:*