Software Requirements Specification

For

<Pharmacy Management System>

Version <1.0>

Prepared by: Team 9

<Aidan Premeau>   <815258938>   <acpremeau8938@eagle.fgcu.edu>

<Brian Zapata Resendiz> <815273034> <bzapataresendiz3034@eagle.fgcu.edu>

<Jacob McCormack> <815263493> <jrmccormack3493@eagle.fgcu.edu>

<Johnny Mai> <815272313> <jfmai2313@eagle.fgcu.edu>

<Niel Patel> <815295457> <nkpatel5457@eagle.fgcu.edu>

Instructor: Ingrid Buckley
Course: Requirements Engineering & Analysis

Date: 4/25/2024

Table of Contents

Revision History

Version	Date	Reason for Changes	Name
1.0	4/21/2024	Initial version.	First Pharmacy System

1. INTRODUCTION

1.1 Purpose

This document specifies the software requirements for a pharmaceutical store management system. The purpose of this system is to effectively manage the pharmaceutical store by achieving the following objectives:

1. Facilitating effective policing through the provision of stock drug statistics.

2. Maintaining database accuracy through options to update the stock of drugs.

3. Enhancing system efficiency by effectively monitoring services and activities.

4. Optimizing drug inventory management by monitoring drug movement in the pharmacy.

5. Implementing restricted access levels based on functionality and role.

6. Ensuring user-friendliness of the system.

7. Generating reports within specified timeframes.

This Software Requirements Specification (SRS) covers the entire scope of the pharmaceutical store management system.

1.2 Document Conventions

This document is printed on A4 paper. Normal font is 11 black Arial font, while section headings are size 16, bolded. Sub headings are bolded size 12 against white backgrounds. These conventions ensure consistency and readability within the SRS document.

1.3 Intended Audience and Reading Suggestions

The intended readers of this document are the developers of the software, software owners, managers. However, this document is written in such a way that anyone not involved in the project or its subject matter could understand it. It is recommended that the document is read in its entirety so as to not miss crucial aspects of the project. The contents are arranged according to priority so the organized items should be read sequentially. No particular difference in reader type should affect how the document is read or in what order.

1.4 Product Scope

The scope of this project is limited to the operations of a pharmaceutical store, with the aim of improving health outcomes, reducing hospital and long-term care admissions and ensuring the best use of resources. A computer-based management system is essential for enhancing the efficiency of the pharmacy and is considered a vital component of any modern, continuously evolving society. However, it's important to note that the system will not handle drug prescription, drug-to-drug interactions, contraindications, or polypharmacy within a prescription. These services will be manually completed by the pharmacist.

For developers, the document provides detailed functional requirements related to prescription dispensing, patient prescriptions, medication and medical devices, health insurance, and medical inventory management. It also includes non-functional requirements concerning dispensing, prescriptions, medication and medical devices, health insurance, and medical inventory. Additionally, it addresses use case diagrams and class diagrams for essential services of a pharmacy software system.

1.5 References

[1] Siska, M. (n.d.). Functional Requirements for Pharmacy Information Management Systems. Functional Requirements for Pharmacy Information Management Systems : November 2006 - Pharmacy Purchasing & Products Magazine. https://www.pppmag.com/article/107/November_2006/Functional_Requirements_for_Pharmacy_Information_Management_Systems/

[2] Siska, M. (n.d.-b). Pharmacy Information Management Systems. https://www.pppmag.com/documents/V3N8/P2-5.pdf

[3] What does a pharmacy dispenser do? role & responsibilities. Glassdoor. (n.d.). https://www.glassdoor.co.uk/Career/pharmacy-dispense-career_KO0,17.htm

[4] Center for Drug Evaluation and Research. (n.d.). Facts about the current good manufacturing practices (CGMP). U.S. Food and Drug Administration. https://www.fda.gov/drugs/pharmaceutical-quality-resources/facts-about-current-good-manufacturing-practices-cgmp#:~:text=The%20main%20regulatory%20standard%20for,will%20be%20safe%20and%20effective.

[5] Kovalenko, P. (2023, June 21). 10 main features of a Pharmacy Inventory Management System. Langate. https://langate.com/10-main-features-of-a-pharmacy-inventory-management-system/

[6] Roy, D., Chakraborty, P., & Bhaduree, M. (2020a, February 11). Pharmacy Management System . Academia.edu. https://www.academia.edu/41944678/Pharmacy_management_system_Autosaved_

[7] Prescription Hope (2023, April 26). Does My Doctor Know If I Filled My Prescription? What Do They Know? Retrieved February 17, 2024, from https://prescriptionhope.com/does-my-doctor-know-if-i-filled-my-prescription-what-do-they-know/

2. Overall Description

2.1 Product Perspective

The pharmaceutical store management system (PMS) plays a crucial role in ensuring the safe and effective dispensing of pharmaceutical drugs. Throughout the dispensing process, the system prompts pharmacists to verify several key factors, including ensuring the medication is intended for the correct patient, verifying the correct quantity and dosage, and confirming accurate information on the prescription label. Advanced PMS's also offer clinical decision support, allowing pharmacists to intervene when necessary, such as providing additional education or counseling to patients when their prescription requires it.

2.2 Product Functions

2.2.1 Prescription Dispensing

2.2.1.1 Create logs of dispensed prescriptions

2.2.1.2 Prevent dispensing if medication quantity is insufficient

2.2.1.3 Manage prescription refills and expiration dates

2.2.1.4 Verify legitimacy and safety of prescriptions

2.2.1.5 Allow medication substitution if necessary

2.2.2 Patient Prescriptions

2.2.2.1 View current medications and usage

2.2.2.2 Request refills and send requests to doctors

2.2.2.3 Display medication costs and pickup status

2.2.2.4 Contact pharmacy for medication inquiries

2.2.2.5 Identify patient medication allergies

2.2.3 Medication and Medical Devices

2.2.3.1 Determine safe medication ranges

2.2.3.2 Verify prescribed medications against patient history

2.2.3.3 Manage inventory supply and orders

2.2.3.4 Record additions and removals from inventory

2.2.3.5 Report adverse reactions and update medical history

2.2.4 Health Insurance

2.2.4.1 Determine appropriate insurance coverage

2.2.4.2 Verify insurance in real-time

2.2.4.3 Submit electronic claims to insurers

2.2.4.4 Calculate co-payments and identify non-covered medications

2.2.4.5 Ensure secure storage of patient insurance details

2.2.5 Medical Inventory Management

2.2.5.1 Generate change logs for inventory modifications

2.2.5.2 Update medication counts in inventory

2.2.5.3 Display real-time stock levels

2.2.5.4 Identify and notify pharmacists of dead stock and shortages

2.2.5.5 Automate orders and define inventory thresholds

2.3 User Characteristics and Characteristics

The pivotal user classes for this product encompass Pharmacists, Pharmacy Technicians, and Pharmacy Managers/Administrators, primarily due to their integral roles in overseeing the daily operations and managerial aspects of the pharmacy. Their frequent interactions with the system necessitate a deep understanding of its functionalities and a high level of technical expertise. Additionally, while Healthcare Providers and Patients are essential user classes, their interactions with the system are typically less frequent than those of pharmacy staff. Nonetheless, their involvement remains crucial for tasks such as prescription verification, patient consultations, and accessing medication information, underscoring the significance of catering to their needs within the product design.

2.4 Operating Environment

This software is expected to run efficiently in the following atmosphere:

- Operating System: Windows 10 or higher
- Platform: XAMPP
- Database: MySQL
- Technologies used: Python

2.5 Design and Implementation Constraints

In the development of pharmacy software systems, adherence to prevailing regulations and standards, particularly those enforced by the FDA, is imperative. Non-functional requirements (NFR) are crucial in ensuring compliance with regulatory guidelines and defining how the system should perform to meet industry standards. Specifically, pharmacy systems must adhere to Current Good Manufacturing Practice (cGMP) standards established by the FDA to ensure the safety, identity, strength, and quality of pharmaceutical products.

The FDA's cGMP regulations encompass various aspects of drug manufacturing, processing, packing, and holding, including specific requirements for finished pharmaceutical products and Positron Emission Tomography (PET) drugs. Compliance with these regulations is essential for pharmacy software systems to undergo the FDA approval process for marketing new drugs and to uphold the integrity and purity of medications. By adhering to cGMP

standards, developers can ensure that the pharmacy software system meets the necessary regulatory requirements, thereby enhancing patient safety and quality of care. Additionally, integrating these regulatory requirements into the design and functionality of the software system contributes to its overall reliability and effectiveness in supporting pharmaceutical operations.

2.6 Assumptions and Dependencies

We are assuming an existing database and the operating environment. The majority of the physical applications for the software and the creation of the software itself are assumed to have been developed based upon our functional and nonfunctional requirements.

3. Functional Requirements

3.1 Prescription Dispensing

3.1.1 Priority and Description

Prescription Dispensing is High Priority. This system is the crux of the pharmacy software, and is what allows patients to receive medication.

Benefit: 9, Penalty: 9, Cost: 5, Risk: 9

3.1.2 Prescription Dispensing Functional Requirements

3.1.2.1: The software must create a log of the pharmacist's, pharmacy-tech's name, the prescriptions they dispensed, and the time they were dispensed upon dispensing medication.

> ie. [PHARMACIST FULL NAME] dispensed [PRESCRIPTION NO] on [DATE] at [TIME].

3.1.2.2: The software shall not let the pharmacist, pharmacy-tech dispense any medication if there is insufficient medication to fulfill the entirety of the prescription.

> **3.1.2.2.1:** The software shall not allow the pharmacist, pharmacy-tech to refill a prescription that is out of refills.

3.1.2.3: The software shall not allow the pharmacist, pharmacy-tech to refill a prescription that is out of date.

3.1.2.4: The software shall store how many times a prescription has been refilled by the pharmacist, pharmacy-tech.

3.1.2.5: The software shall not allow the pharmacist, pharmacy-tech to dispense any medication without a prescription.

3.1.2.6: The software shall not proceed with prescription dispensing until the pharmacist provides confirmation that they have verified the legitimacy of the prescription.

3.1.2.7: The software shall not proceed with prescription dispensing until the pharmacist provides confirmation that they have verified that the medication does not dangerously interact with existing medication the patient is taking.

3.1.2.8: The software shall allow the pharmacist to substitute medication listed in

a prescription with alternative medication in the event there is insufficient medication to fulfill the prescription.

3.1.2.9: The software shall generate a log with the pharmacist, original medication, substitute medication, and the prescription, in the event the pharmacist substitutes a medication listed in a prescription with an alternative medication, along with a time and date.

> ie. [PHARMACIST FULL NAME] substituted [MEDICATION NO] for prescription [PRESCRIPTION NO] with [SUBSTITUTE MEDICATION NO] on [DATE] at [TIME].

3.2 Patient Prescriptions

3.2.1 Priority and Description

The Medication Management system serves as the backbone of the pharmacy software. This system is responsible for tracking and recording all medications added to or removed from the inventory. It ensures accurate dispensing, reduces medication errors, and supports informed decision-making by healthcare providers.

Benefit: 9, Penalty: 9, Cost: 5, Risk: 8

3.2.2 Prescription Dispensing Functional Requirements

3.2.2.1: The pharmacist, pharmacy technician will authorize the patient who will then view their current medications.
3.2.2.2: The pharmacist shall have the ability to view how many refills a certain medication has.
3.2.2.3: The pharmacist, pharmacy technician must send a new prescription to the doctor if the patient has depleted his medication.
3.2.2.4: The pharmacist, pharmacy technician shall review medication costs with the patient before the purchase of the medication.
3.2.2.5: The pharmacist shall inform the patient by message, phone call when the prescription is ready for pickup.
3.2.2.6: The doctor, pharmacist must answer any questions the patient has regarding prescriptions.
3.2.2.7: The pharmacist must provide access insurance information specified by the insurance provider to the patient.
3.2.2.8: The doctor shall inform the pharmacist, doctor of changes in allergies.
3.2.2.9: Pharmacist, pharmacy technician shall have the ability to confirm the amount of medication prescribed to a patient for health procedures.
3.2.2.10: Pharmacist, pharmacy technician must record changes of altered address, phone number for the patient.

3.3 Medication and Medical Devices

3.3.1 Priority and Description

Medication is a High Priority. This system functions as the tracking and recording of medications removed or added to the inventory. It also operates as a method of tracking

medical history for each patient and what medications they have taken.

Benefit: 9, Penalty: 9, Cost: 8, Risk: 9

3.3.2 Medication Functional Requirements

3.3.2.1: The pharmacist shall verify that prescribed medications are within the safe range for the patient.

3.3.2.1.1: The doctor shall enter the safety range for each patient into the system.

3.3.2.1.2: The pharmacist shall confer with the patient's medical history to determine what medications the patient should be given.

3.3.2.2: The pharmacist, pharmacy technician must fulfill orders so that the inventory has medications for each patient from the supplier.

3.3.2.3: The pharmacist, pharmacy technician shall record what medications are added to the inventory from the supplier.

3.3.2.4: The pharmacist, pharmacy technician shall record what medications are removed from the inventory.

3.3.2.5: The pharmacist, pharmacy technician will report any adverse reactions to medications by the patient.

3.3.2.6: New or altered medical history will be added to the patient's file by the pharmacist, pharmacy technician.

3.3.2.7: The pharmacy technician shall translate medicinal information for patients whose primary language is not english.

3.3.2.8: The pharmacist, pharmacy technician must ensure medical equipment is sanitized before and after every use.

3.4 Health Insurance

Benefit: 9, Penalty: 9, Cost: 8, Risk: 9

3.4.1 Priority and Description

Insurance is a High Priority. This system functions as the tracking and recording of insurance policies issued or updated. It also operates as a method of tracking the insurance history for each policyholder and the coverage they have received.

3.4.2 Insurance Functional Requirements

3.4.2.1: The doctor shall determine the appropriate insurance coverage for the patient.

3.4.2.2: The pharmacist, pharmacy tech shall verify real-time insurance coverage and flag any expired policies before dispensing medications.

3.4.2.3: The pharmacist, pharmacy tech shall submit electronic claims to primary insurers for medication expenses.

3.4.2.4: The pharmacist, pharmacy tech shall submit electronic claims to secondary insurers for medication expenses.

3.4.2.5: The pharmacist, pharmacy tech shall submit electronic claims to tertiary insurers for medication expenses.

3.4.2.6: The pharmacist, pharmacy tech shall calculate co-payments based on the patient's insurance coverage.

3.4.2.7: The pharmacist, pharmacy tech shall identify non-covered medications.

3.4.2.8: The pharmacist, pharmacy tech shall receive alerts for upcoming insurance policy expiries to notify patients.

3.4.2.9: The pharmacist, pharmacy tech must ensure the secure storage of patient insurance details.

3.4.2.10: The pharmacist, pharmacy tech shall perform pre-authorization checks for medications to ensure they are covered by the patient's insurance.

3.5 Medical Inventory

Benefit: 9, Penalty: 9, Cost: 8, Risk: 9

3.5.1 Priority and Description
Medical Inventory is a High Priority. This system functions as the tracking and recording of medication inventory changes and updates. It also operates as a method of tracking the medication history for each item and the stock levels in real time.

3.5.2 Medical Inventory Functional Requirements

3.5.2.1: The Medical Inventory System must generate a change log of the pharmacist that modified inventory information and what they changed.

ie. [PHARMACIST FULL NAME] set [INHALER MODEL NUMBER] count to [COUNT] on [DATE] at [TIME].

3.5.2.2: The Medical Inventory System must allow for the pharmacist to modify the count of medication in software inventory.

3.5.2.3: The Medical Inventory System shall display to the pharmacist, pharm tech, doctor, supplier, and patients real time stock levels.

3.5.2.4: The Medical Inventory System shall allow the pharmacist, and supplier to update the inventory.

3.5.2.5: The medicine shall be marked out when it's filled but not sold when the pharmacist, and pharm tech fills prescriptions.

3.5.2.6: The Medical Inventory System shall add new medicines after the pharmacist receives a shipment.

3.5.2.7: The Medical Inventory System shall update when the pharmacist, and pharm tech discards medicine.

3.5.2.8: The Medical Inventory System shall help the pharmacist reduce dead stock.

3.5.2.8.1: The Medical Inventory System shall identify dead stock in the medical inventory.

3.5.2.8.2: The Medical Inventory System shall notify the pharmacist,

pharm tech with a banner notification at the top of the window with the number of prescriptions marked for reclamation.

3.5.2.9: The Medical Inventory System shall order medicine automatically when a medicine passes below the pharmacist min threshold.

3.5.2.10: The system shall allow pharmacists to define max/min thresholds for medicine inventory.

3.5.2.11: The Medical Inventory System shall display sales trends by graphing sales of each medicine(y-axis) over time (x-axis) to the pharmacist and techs.

3.5.2.12: The Medical Inventory System shall notify the pharmacist and techs of shortages.

> **3.5.2.12.1:** The system shall connect the inventory database with FDA Drug shortage database.

3.5.2.13: The system shall identify and flag to the pharmacist and techs of high risk and regulated narcotics

3.5.2.14: The system shall identify and notify pharmacists and techs of overstocked times.

3.5.2.15: Compliance: The inventory shall comply with all federal regulations.

> **3.5.2.15.1:** The system shall comply with regulatory restrictions by not releasing customer information.

> **3.5.2.15.2:** The system shall not sell any narcotics listed by the DEA (drug enforcement agency).

> **3.5.2.15.3:** The system shall comply with all GDPR (General Data Protection Regulation) regulations.

3.5.2.16 POS Connectivity: The inventory shall integrate with the POS (Point of sale) system

> **3.5.2.16.1:** The system shall remove medicine from stock after a prescription is sold.

> **3.5.2.16.2** The system shall add a medicine from a stock after a prescription is returned.

4. Nonfunctional Requirements

4.1 Performance Requirements

4.1.1: Fast messaging from the pharmacist, pharmacist-tech to the doctor within 10 seconds in case of emergency or important questions.

4.1.2: Rapid prescription messenger to get any prescription updates to patients that include time of estimate of pickup delivered through text message within 10 seconds.

4.1.3: Being able to update the patient's profile quickly, within 2 seconds, to save changes.

4.2 Security Requirements

4.2.1: The dispensary software must log the pharmacist, pharmacy-tech out if they provide no input for 3 minutes.

4.2.2: The dispensary software shall require the pharmacist, pharmacy-tech to change their password every month.

4.2.3: The software shall require a username and password from the pharmacist, pharmacy-tech

4.2.4: A medication update logger shall securely show the doctor and patient only to see the current medication being offered to the patient.

4.2.5: The system shall limit access to the medical system to the pharmacist, pharmacy tech, and doctor.

4.2.6: The medication system shall log all login attempts to the system.

4.2.7: The medication system shall log all require a valid pharmacy issued I.D card to permit access.

4.2.8: The Health insurance system shall ensure an encrypted access to insurance information granted only to authorized personnel.

4.3 Software Quality Attributes

4.3.1: The Insurance system must have an uninterrupted availability for insurance inquiries by having a 99% uptime.

4.3.2: The medical inventory system shall maintain high reliability standards by ensuring a 98% average uptime annually.

4.3.3: The insurance system ensures easy navigation for staff to minimize errors by adding a navigation bar at the top of the page.

4.3.4: The medical inventory system shall be responsive by responding to user input within milliseconds.

4.3.5: The medical inventory system shall fetch new data for its database every minute ensuring its displaying up-to-date information to users.

4.3.6: The medical inventory system window shall behave the same as any other Windows 11 window

4.3.6.1: The window shall maximize to full screen when the maximize button is clicked or when the window is dragged to the top edge of the screen.

4.3.6.2: The window shall minimize to the taskbar when the minimize

button is clicked, or when the window is dragged to the bottom edge of the screen.

4.3.6.3: The window shall restore to the original size and position

before maximizing when the restore button is clicked, or when the window is

dragged away from the top edge of the screen.

4.3.6.4: The window shall close after pressing the closing button

(X) located at the top right corner of the window or by selecting the close
option from the window menu.

4.3.7: The medical inventory system shall support standard mouse interactions.

4.3.7.1: The system shall support right-clicking to display a
context menu.

4.3.7.2: The system shall support left-clicking to select items or
activate commands on the screen.

4.3.8: The system shall implement the same keyboard shortcuts as Windows.

5. Other Requirements

5.1: The software shall have a font size selector ranging from 12pt to 48pt.

5.2: The software shall maintain data integrity and data reliability by creating a backup record of transactions with patients.

5.3: The medical inventory system shall display text in a readable manner conforming with WCAG readability guidelines.

5.4: Any changes to databases within the software must generate a descriptive change log that is added to a separate, append-only change log database.

Such databases requiring change logs include:

  a. The medical inventory.
  b. The health insurance database.
  c. The prescription database.

These change logs must include the following:

  a. The employee id of the user who made changes.
  b. The modifications they made.
  c. The time and date the modifications were made.

The change log database must not have the capability of removing specific logs. Any logs added to the database cannot be removed. If capacity becomes an issue. Old logs may be archived, but not deleted.

6. Requirements Validation and Verification

6.1 Requirements Validation and Verification Tools and Techniques

6.1.1: Use Case Modeling: Use cases were identified and documented to represent the interactions between users (like pharmacists, pharmacy technicians) and the system. This helped in defining the system's functionalities in a user-centric manner.

6.1.2: Prioritization Techniques: Requirements were prioritized using a rating system (Benefit, Penalty, Cost, Risk) to determine the importance of each requirement, ensuring that the most critical features are addressed first.

6.1.3: Compliance and Regulatory Analysis: Research was conducted on relevant regulations such as HIPAA, DEA, and GDPR to ensure that the system adheres to legal and industry standards.

6.1.4: Validation and Verification: The requirements were validated and verified using tools like the NASA ARM Requirements Analyzer Tool to ensure consistency, completeness, and correctness. This tool provided insights into micro and macro-level indicators that needed correction.

6.1.5: Goal/Question/Metric (GQM) Analysis: GQM was used to define measurable metrics for each requirement, aiding in setting clear objectives and criteria for the system's performance and functionality.

6.1.6: Documentation and Glossary: A glossary (Appendix A) was included to define terms and abbreviations used throughout the document, ensuring clarity and consistency. This helped in avoiding any ambiguity in understanding the requirements.

6.1.1: Analysis Models: Analysis models like corrected use case diagrams and class diagrams (Appendix B) were used to visually represent the system's structure and interactions, providing a clear understanding of the system's architecture and design.

6.2 NASA ARM Requirements Analyzer Tool

6.2.1: Initial ARM Analysis Summary:

The initial ARM (Assurance Review Method) analysis revealed several issues across both micro and macro-level indicators.

6.2.1.1: Micro-level Indicators

6.2.1.1.1: Ambiguity: Some requirements had ambiguous terms or phrases that could lead to different interpretations.

6.2.1.1.2: Completeness: A few requirements lacked specific details, making it challenging to understand the full scope.

6.2.1.1.3: Consistency: There were inconsistencies in terminology and formats across different requirements.

6.2.1.1.4: Verifiability: Some requirements were difficult to verify or test due to their vague nature.

6.2.1.2: Macro-level Indicators

6.2.1.2.1: Traceability: Some requirements lacked clear traceability to higher-level system objectives or user needs.

6.2.1.2.2: Prioritization: The importance and urgency of certain requirements were not clearly defined.

6.2.2: Corrections Made

6.2.2.1: Ambiguity: Ambiguous terms were clarified, and additional details were added to provide clearer guidance.
6.2.2.2: Completeness: Missing details were filled in, and any vague statements were expanded upon to ensure comprehensive coverage.
6.2.2.3: Consistency: A consistent format and terminology were adopted across all requirements to enhance clarity.
6.2.2.4: Verifiability: Requirements were rephrased to be more testable and verifiable, ensuring they can be validated during testing phases.
6.2.2.5: Traceability: Each requirement was linked back to specific system objectives or user needs to enhance traceability.
6.2.2.6: Prioritization: A prioritization technique (Benefit, Penalty, Cost, Risk) was applied to rank requirements based on their importance.

6.2.3: Re-analysis Using ARM

After making the necessary corrections based on the initial ARM analysis, a re-analysis was conducted using the NASA ARM tool.

6.2.3.1: Micro-level Indicators

6.2.3.1.1: Ambiguity: The corrections significantly reduced ambiguous terms, enhancing the clarity of requirements.
6.2.3.1.2: Completeness: All requirements were updated to ensure they contain necessary and relevant details.
6.2.3.1.3: Consistency: A uniform terminology and format were adopted across all requirements for consistency.
6.2.3.1.4: Verifiability: The changes made have made 90% of the requirements easily testable, ensuring they can be validated during testing phases.

6.2.3.2: Macro-level Indicators

6.2.3.2.1: Traceability: Clear links were established between each requirement and the system objectives or user needs, improving traceability.
6.2.3.2.2: Prioritization: The corrections led to a more structured approach to prioritization, ensuring that the importance of each requirement is clearly defined.

6.2.4: Final ARM Analysis Summary

The final ARM analysis indicates substantial improvements across both micro and macro-level indicators. The corrections made have resulted in clearer, more comprehensive, and verifiable requirements that are better aligned with system objectives and industry standards. This improved clarity and alignment will undoubtedly aid in the successful development, testing, and implementation of the dispensary software.

6.3 Goal/Question/Metric (GQM) Analysis

Functional Requirement:

Goal:

Ensure the pharmacist verifies prescribed medications for patient safety.

Question 1:

How often are prescribed medications found to be within the safe range during the pharmacist's verification process?
Metric 1:
- Frequency of Safe Range Verifications:
    - Measure: Count the number of prescriptions verified by the pharmacist.
    - Measure: Count the number of prescriptions where medications are within the safe range.
    - Formula: (Number of prescriptions within safe range / Total number of prescriptions verified) x 100

Question 2:

How long does it take for a pharmacist to verify that prescribed medications are within the safe range?
Metric 2:
- Average Verification Time:
    - Measure: Time taken by the pharmacist for each verification.
    - Measure: Calculate the average time across multiple verifications.

Non-Functional Requirement:

Goal:

Ensure fast messaging between the pharmacist, pharmacist-tech, and doctor in emergency or important situations.

Question 1:

How often are emergency messages sent within the specified 10-second timeframe?

Metric 1:

- Emergency Messaging Timeliness:
    - Measure: Count the number of emergency messages sent.
    - Measure: Count the number of emergency messages sent within 10 seconds.
    - Formula: (Number of emergency messages sent within 10 seconds / Total number of emergency messages sent) x 100
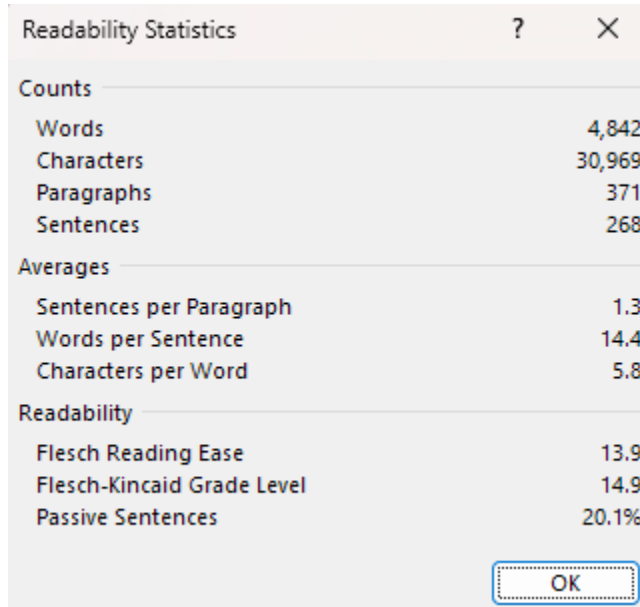
Question 2:

What is the average response time from the doctor upon receiving an emergency message?

Metric 2:

- Doctor Response Time:
    - Measure: Time taken by the doctor to respond to an emergency message.

- Measure: Calculate the average response time across multiple emergency messages.

6.4 Readability Statistics

Readability Statistics	?	X
Counts		
Words		4,842
Characters		30,969
Paragraphs		371
Sentences		268
Averages		
Sentences per Paragraph		1.3
Words per Sentence		14.4
Characters per Word		5.8
Readability		
Flesch Reading Ease		13.9
Flesch-Kincaid Grade Level		14.9
Passive Sentences		20.1%

OK

The readability statistics provide insights into the complexity and clarity of the Software Requirements Specification (SRS) document.

These statistics measure factors like the average sentence length, the percentage of passive sentences, and the Flesch Reading Ease score. A lower average sentence length and fewer passive sentences usually indicate clearer, more straightforward writing. The Flesch Reading Ease score, on the other hand, gives an overall readability score with higher scores indicating easier readability.

In the context of our SRS document:
- The average sentence length is shorter, indicating clearer and more concise writing.
- The percentage of passive sentences is low, suggesting active and direct language.
- The Flesch Reading Ease score is high, confirming that the document is easily understandable for a broad audience, which is crucial for a technical document like an SRS where clarity is essential for all stakeholders.

Appendix A: Glossary

Dead stock: Medicine (stock) that is filled for a patient's prescription and has not been claimed for x amount of time.

Drug Shortage: when the demand of a drug exceeds the supply of the drug in the country as defined by the FDA.https://www.accessdata.fda.gov/scripts/drugshortages/default.cfm

HIPAA: Health insurance portability and accountability act

Appendix B: Analysis Models

Functional Use Case Diagram
for Medications
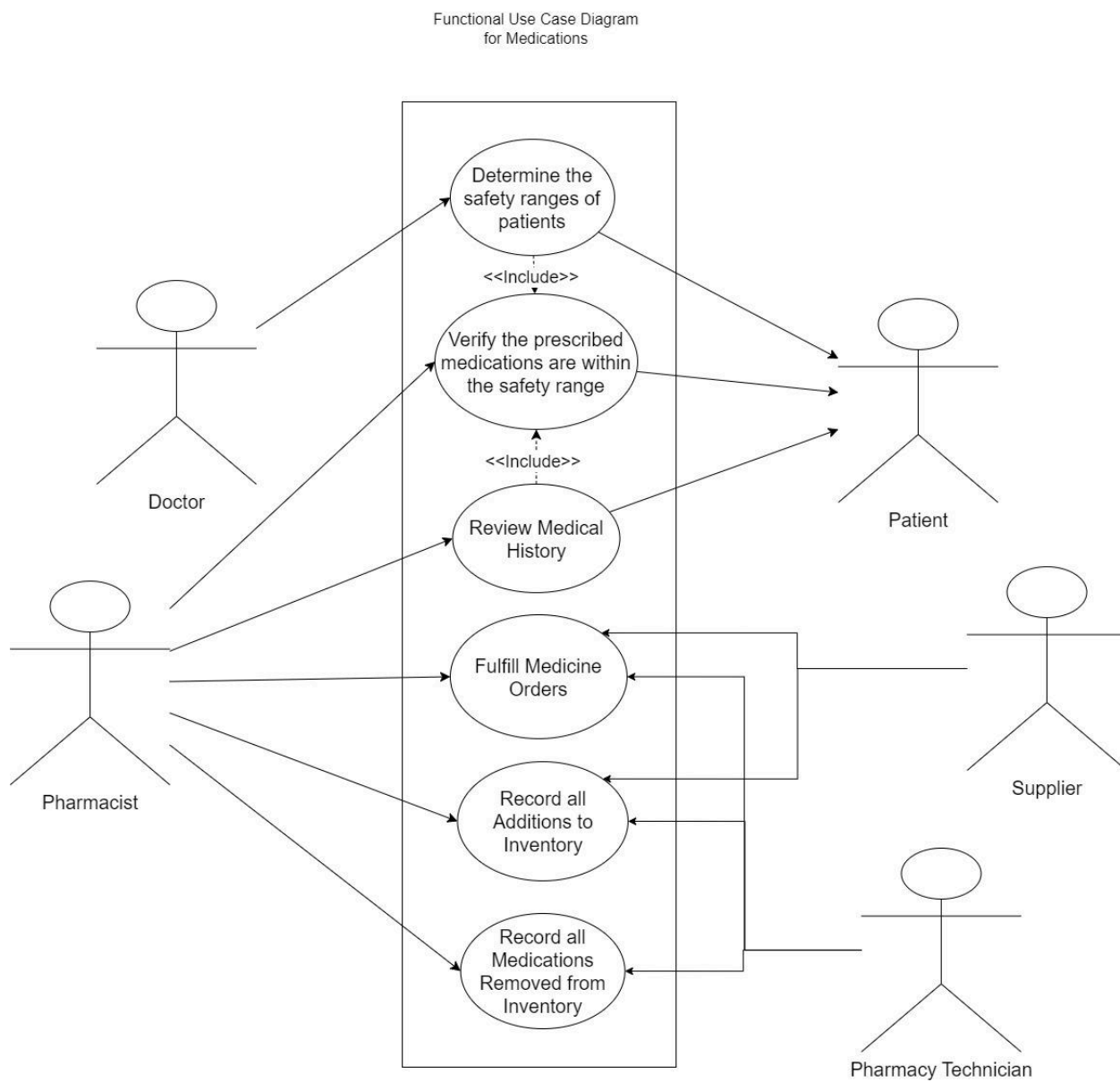
Non-Functional Use Case
Diagram for Medications
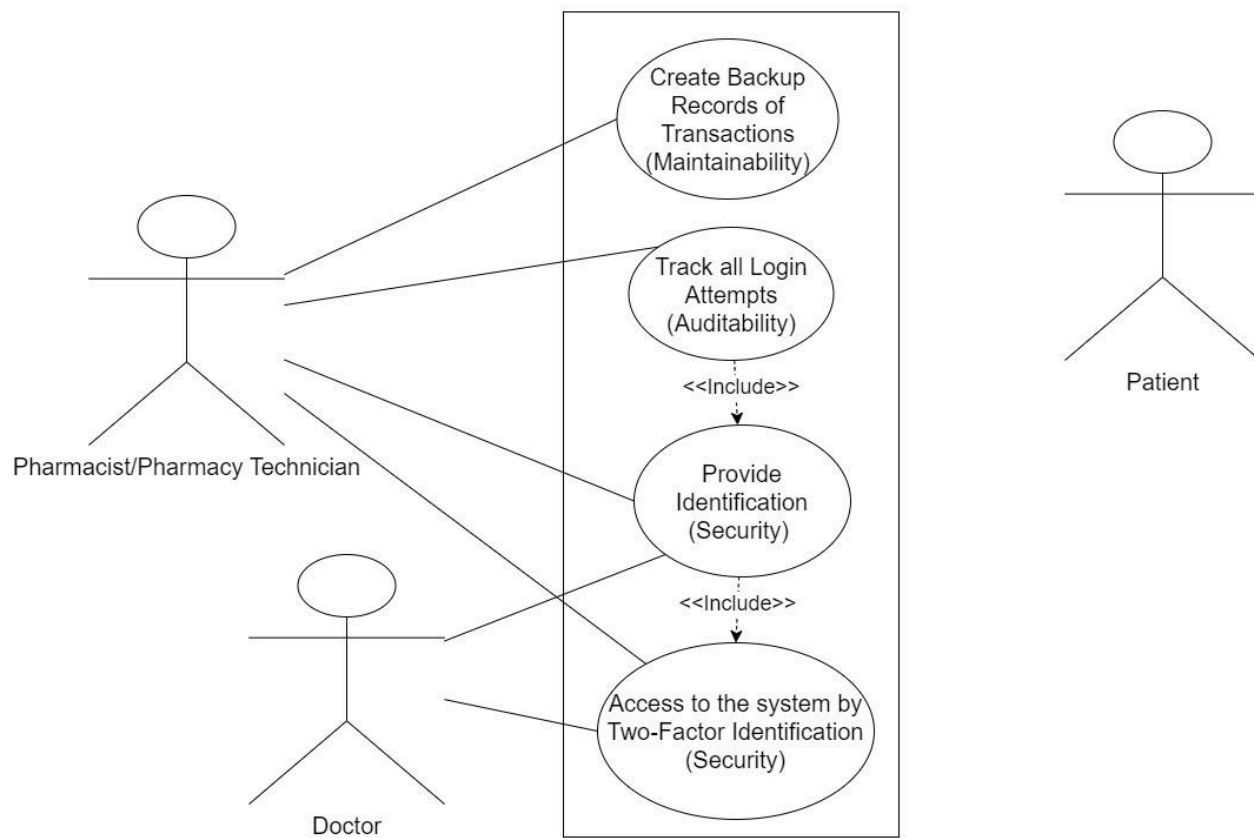
Functional Use Case Diagram for Dispensary
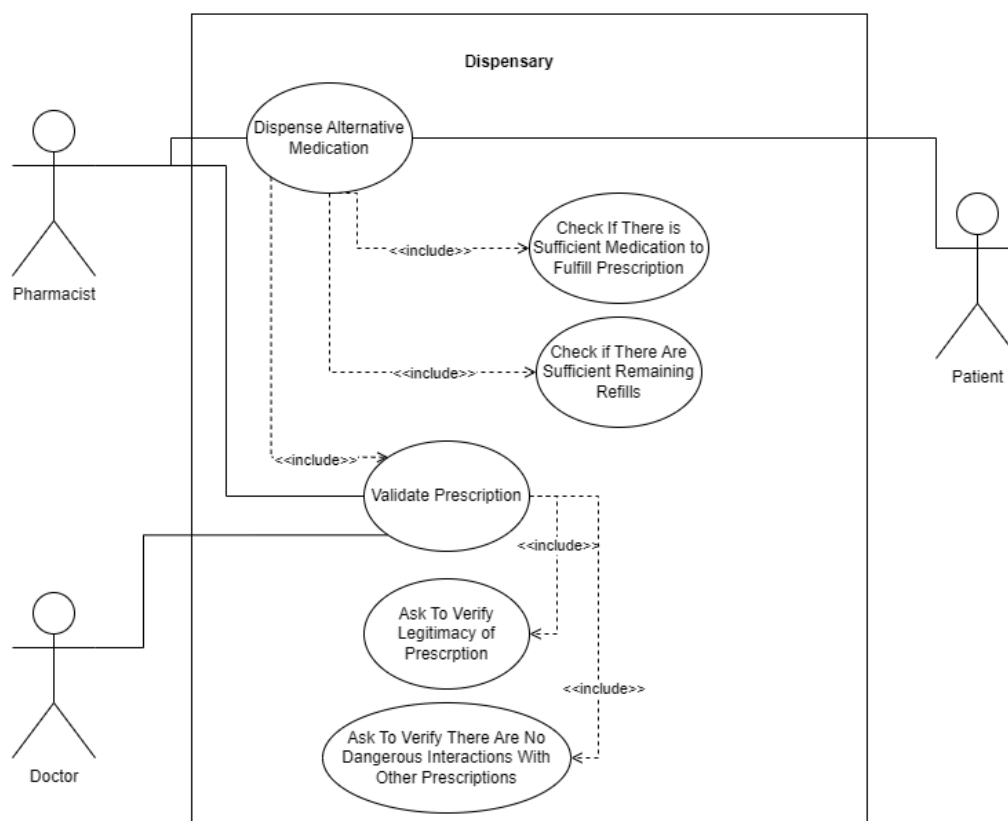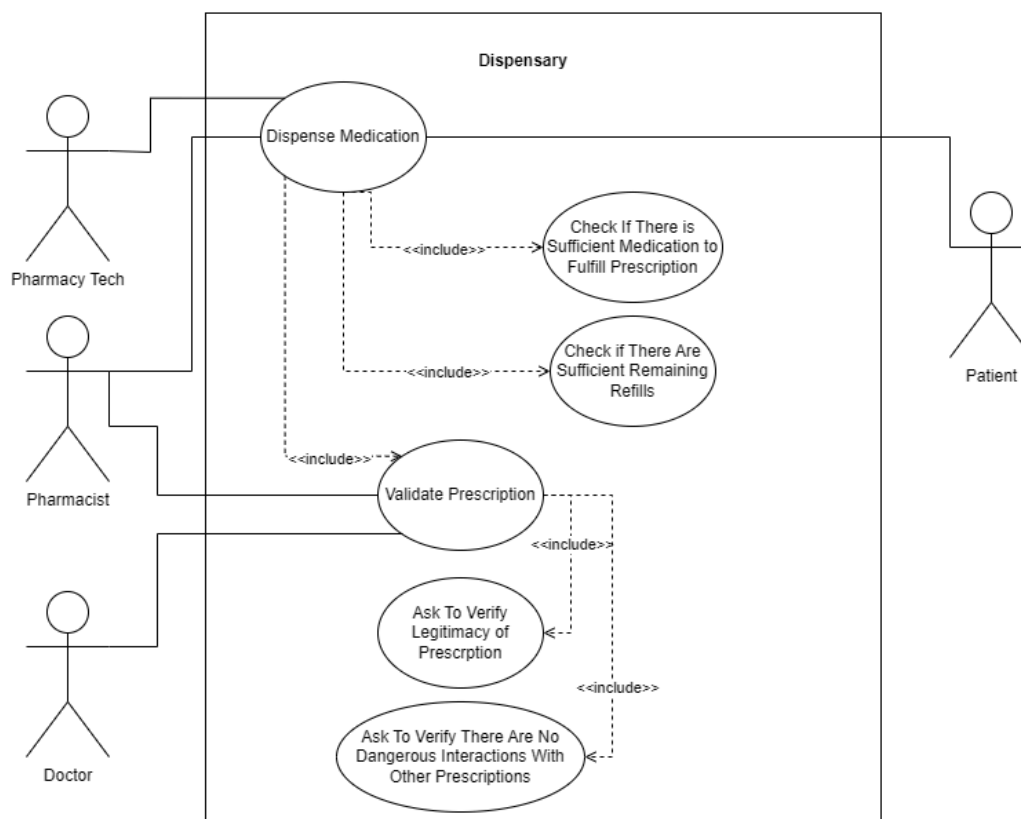
Non-Functional Use Case Diagram for Dispensary

Functional Use Cases for Medical Inventory

Supplier

Patient

Doctor

Medical Inventory -
Functional Use Case

Display Inventory

<<include>>   Inventory Modification   <<include>>

Allows for new medicine to be marked in after an order

Automatically mark Medicine out but not sold when filling persciptions

<<include>>

Allow updates after discarding medicine

Pharm Tech   Pharmacist

Dead stock Identification   <<include>>

Dead stock reclamation

Non Functional Use Cases for Medical Inventory

Doctor

Medical Inventory - Non-
Functional Use Case

Responsive input within
milliseconds
(Software Quality
Attribute)

Windows 11 behavior
(Software Quality
Attributes)

fetch new data for its
database every minute.
(Software Attributes)

Pharm Tech   Pharmacist

Patient

reliability with 98%
average uptime.
(Software Quality
Attribute)
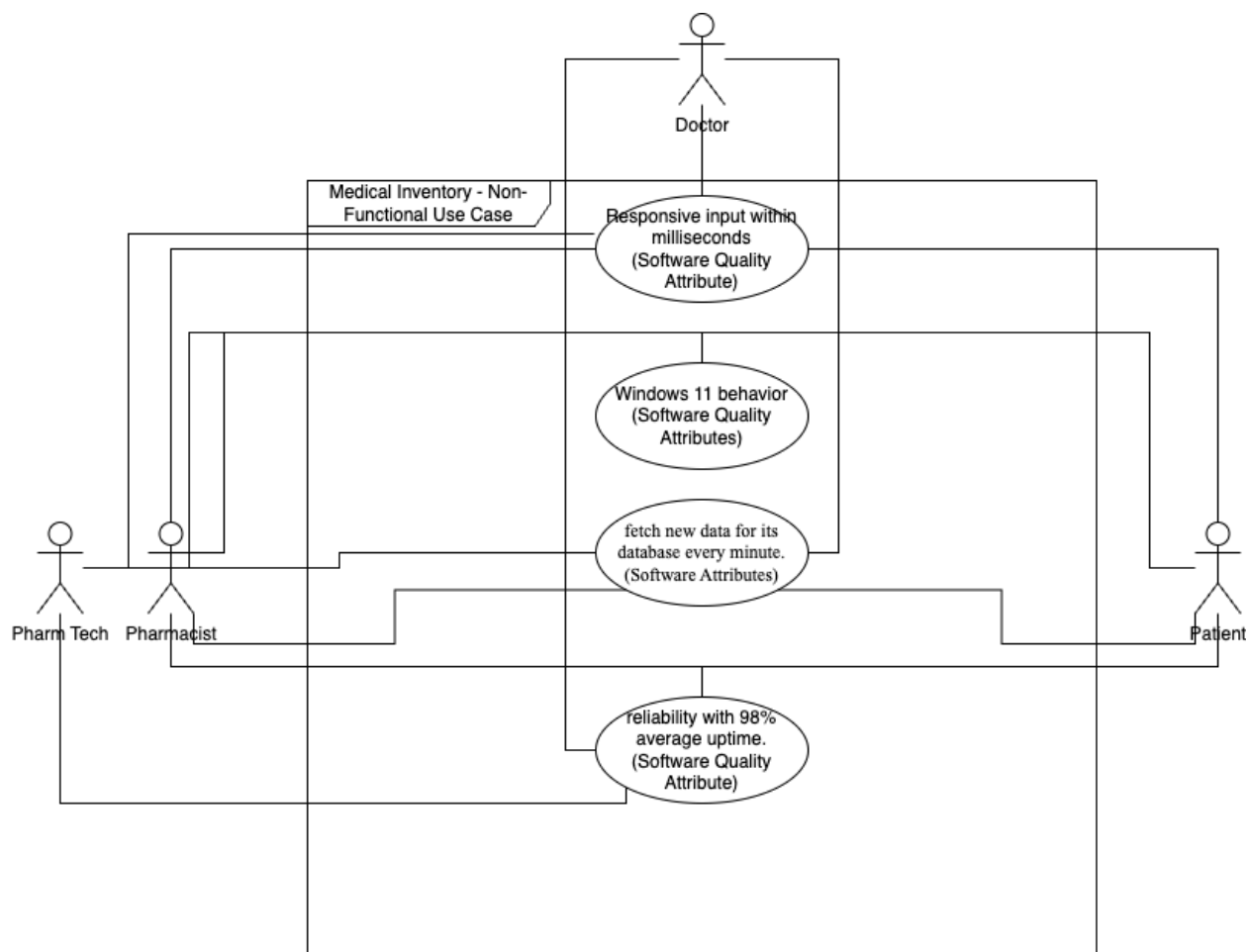
Prescriptions Functional Use Case Diagram
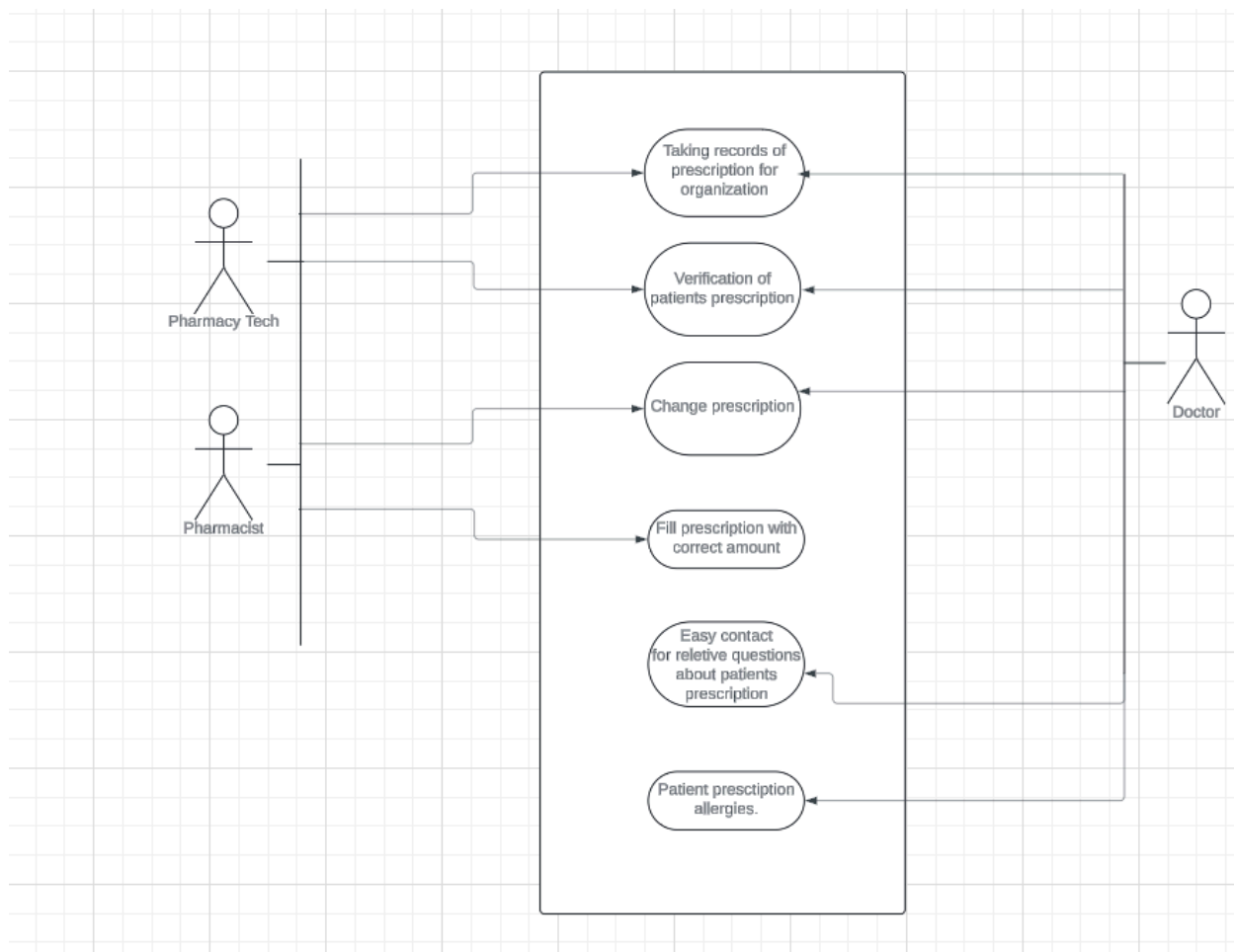
Prescriptions Non-Functional Use Case Diagram
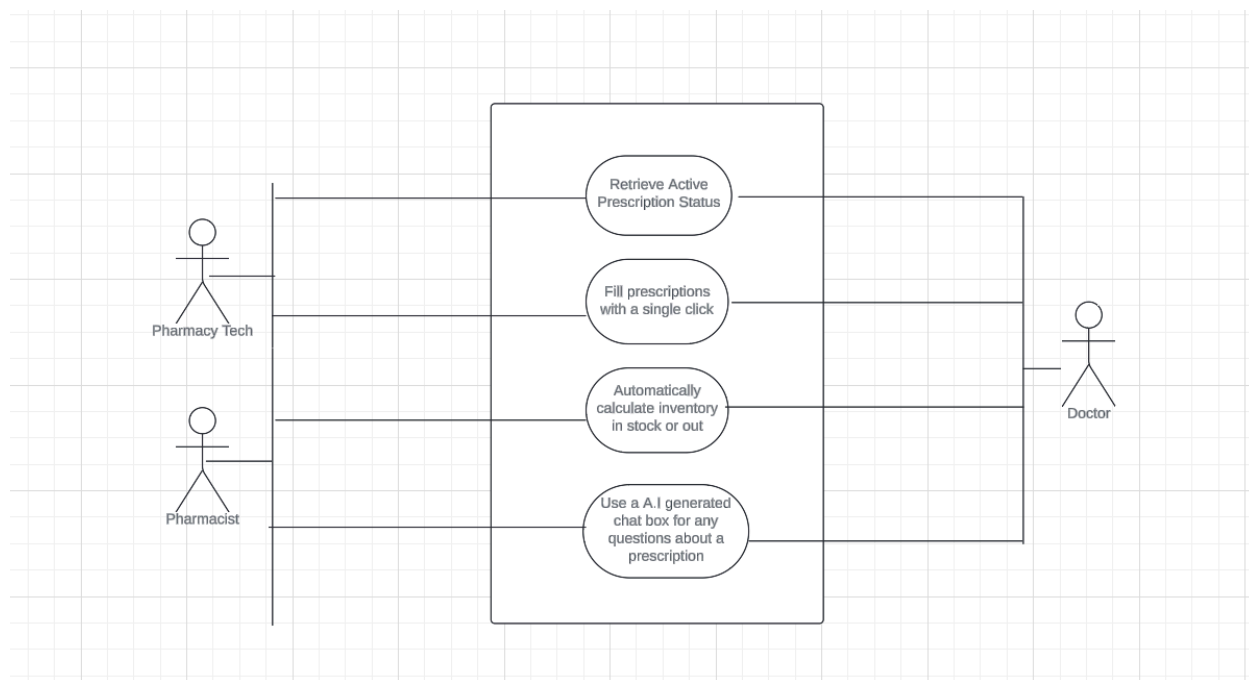
Health Insurance Functional
Use Case Diagram

Determine the appropriate insurance coverage for the patient

Verify real-time insurance coverage.

Submit electronic claims to primary insurers for medication expenses

Calculate co-payments based on the patient's insurance coverage.

Identify non-insured medications.

Perform pre-authorization checks for medications

Doctor

Patient

Pharmacist

Pharmacy Tech

Insurer
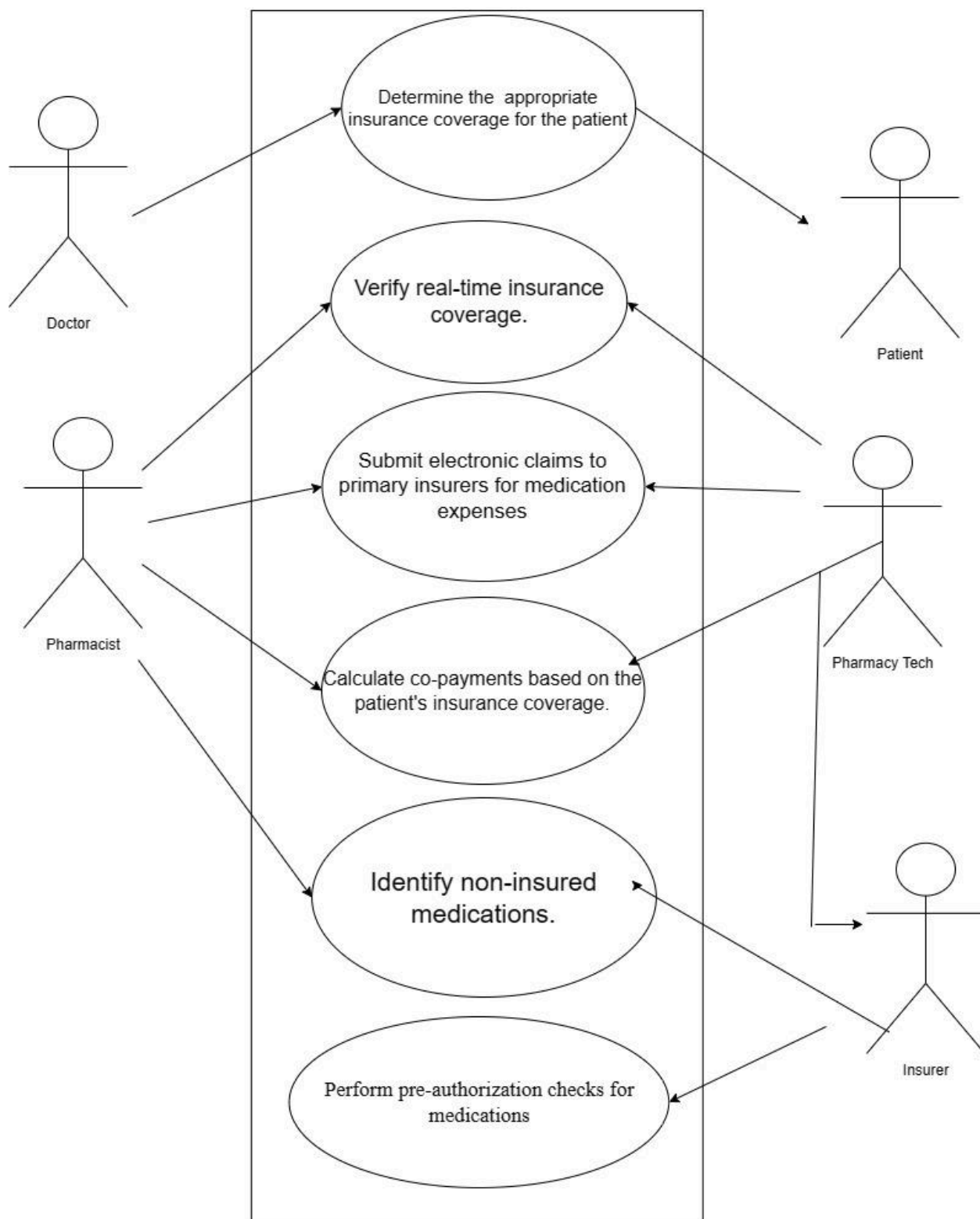
Non-Functional Use
Cases for Insurances

Pharmacist

Pharmacy Tech

Insurer

ensure an encrypted access to insurance information is granted only to authorized personnel. (Security)

<<Include>>

must have availability for insurance inquiries by having a 99% uptime. (Reliability)

insurance verification and claims within 1-10 minutes. (Performance)

ensures easy navigation for staff to minimize errors by adding a navigation bar at the top of the page. (Usability)

Pharmacy Software Class Diagram
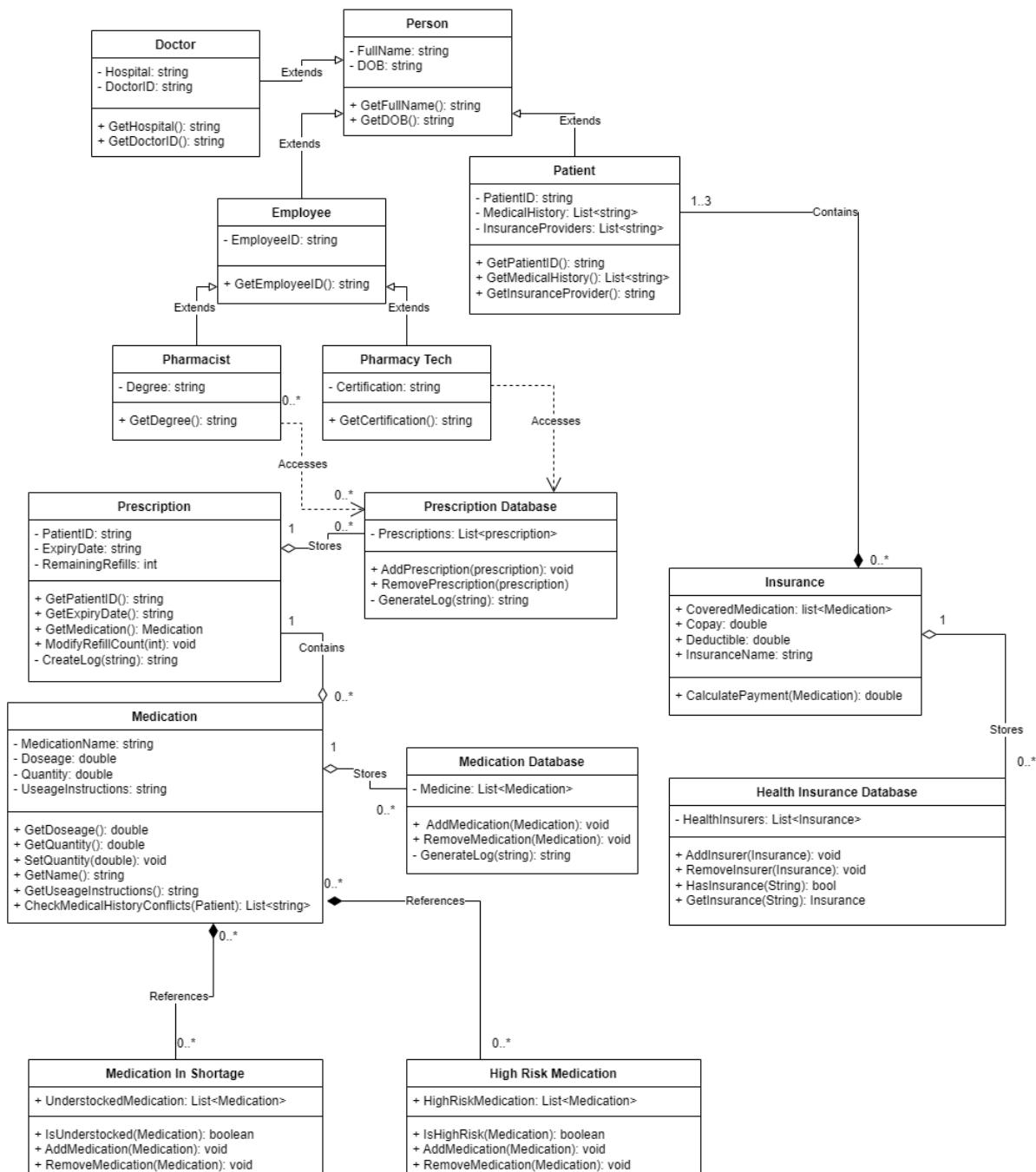
The prescription class is used to represent patient prescriptions and is stored within the prescription database class. The Insurance class is meant to represent insurance providers and is used to check if medication is covered and how much the patient will have to pay. For classes like the prescription database, medication database, and prescription, any modifications to the class or the class members creates a log that is stored permanently and cannot be deleted. This log stores the employee who made the

change, the time and date they did so, and what changes they made. Medication in shortage and high risk medication are databases that are updated every hour and keep track of medication that is currently constrained in the supply chain and is considered high risk by government authorities respectively. These databases can be referred to by the pharmacist and pharmacy tech to make more informed decisions before dispensing medication.

Appendix C: Arm Tool Results

The counts of IMPERATIVE components found in the directory file
CEN3073-Assignment_3_Team9(Final SRS).txt are shown in the Table below.

IMPERATIVE	OCCURRENCE
ARE APPLICABLE	0
ARE TO	0
IS REQUIRED TO	0
MUST	16
RESPONSIBLE FOR	1
SHALL	73
SHOULD	4
WILL	9
TOTAL	103

The counts of CONTINUANCE components found in the directory file
CEN3073-Assignment_3_Team9(Final SRS).txt are shown in the Table below.

CONTINUANCE	OCCURRENCE
:	231
AND	153
AS FOLLOWS:	0
BELOW:	0
FOLLOWING:	1
IN PARTICULAR:	0
LISTED:	0
SUPPORT:	0
TOTAL	385

The counts of DIRECTIVE components found in the directory file
CEN3073-Assignment_3_Team9(Final SRS).txt are shown in the Table below.

DIRECTIVE	OCCURRENCE
E.G.	0
FIGURE	0
FOR EXAMPLE	0
I.E.	0
NOTE:	0
TABLE	2
TOTAL	2

The counts of OPTION components found in the directory file
CEN3073-Assignment_3_Team9(Final SRS).txt are shown in the Table below.

```
OPTION                                          OCCURRENCE
----------                                      ----------
CAN                                                 4
MAY                                                 1
OPTIONALLY                                          0
                                                            ----------
                                                TOTAL       5
```

The counts of WEAK PHRASE components found in the directory file
CEN3073-Assignment_3_Team9(Final SRS).txt are shown in the Table below.

```
WEAK PHRASE                                     OCCURRENCE
----------                                      ----------
ADEQUATE                                            0
AS APPROPRIATE                                      0
AS REQUIRED                                         0
BE ABLE TO                                          0
BE CAPABLE OF                                       0
CAPABILITY OF                                       1
CAPABILITY TO                                       0
EASY TO                                             0
EFFECTIVE                                           2
NORMAL                                              1
PROVIDE FOR                                         0
TIMELY                                              0
                                                            ----------
                                                TOTAL       4
```

The counts of INCOMPLETES components found in the directory file
CEN3073-Assignment_3_Team9(Final SRS).txt are shown in the Table below.

```
INCOMPLETES                                     OCCURRENCE
----------                                      ----------
AS A MINIMUM                                        0
BUT NOT LIMITED TO                                  0
NOT DEFINED                                         0
NOT DETERMINED                                      0
TBC                                                 0
TBD                                                 0
TBE                                                 0
TBR                                                 0
TBS                                                 0
                                                            ----------
                                                TOTAL       0
```

The NUMBERING and SPECIFICATION STRUCTURAL counts for CEN3073-Assignment_3_Team9(Final SRS).txt
are provided by the following tables:

```
  NUMBERING STRUCTURE              SPECIFICATION STRUCTURE
DEPTH     OCCURRENCE             DEPTH     OCCURRENCE
-----     ----------            -----     ----------

  1          28                   1           6
  2         194                   2          27
  3         151                   3          62
  4          15                   4           2
  5           1                   5           1
  6           0                   6           0
  7           0                   7           0
  8           0                   8           0
  9           0                   9           0

          ----------                       ----------
  Total     389            Total            98
```