

# Defined System Requirements for the Medication Adherence System

Tyler Sapp      Wyatt Henderson      Ronaldo Menendez  
Gustavo Naveto      Nick Larson

## Table of contents

<b>1</b>	<b>Defined System Requirements</b>	<b>1</b>
1.1	Functional Requirements . . . . .	1
1.2	Non-Functional Requirements . . . . .	2
1.2.1	How to Use These Requirements . . . . .	3
1.3	Use Case Diagrams . . . . .	3
1.3.1	Functional Requirements Use Case Diagram . . . . .	4
1.3.2	Non-Functional Use Case Diagram . . . . .	5

## 1 Defined System Requirements

### 1.1 Functional Requirements

Req ID	Title	Description	Priority
FR1	Medication Schedule Retrieval	The system shall retrieve upcoming medication schedules from the Medication Schedule Store for each patient.	High
FR2	Medication Reminder Dispatch	The system shall automatically send medication reminders to patients when the scheduled time is reached.	High

<b>Req ID</b>	<b>Title</b>	<b>Description</b>	<b>Priority</b>
<b>FR3</b>	Adherence Confirmation Input	The system shall receive adherence confirmation from patients, including relevant details such as timestamp and status.	High
<b>FR4</b>	Adherence Record Logging	The system shall record each adherence confirmation in the Adherence Records Store to track when medications are taken or missed.	High
<b>FR5</b>	Report Generation	The system shall generate adherence reports by compiling data from the Adherence Records Store (and optionally the Patient Data Store), store the reports, and send them to the Healthcare Provider.	Medium
<b>FR6</b>	Patient Data Submission	The system shall allow patients to submit their personal and medical data, which will be validated and stored in the Patient Data Store.	High
<b>FR7</b>	Healthcare Provider Data Access	The system shall allow healthcare providers to securely retrieve patient data from the Patient Data Store after verifying their credentials.	High
<b>FR8</b>	Reminder Logging	The system shall log all reminder dispatch events for auditing and troubleshooting purposes.	Medium

## 1.2 Non-Functional Requirements

<b>Req ID</b>	<b>Title</b>	<b>Description</b>	<b>Priority</b>
<b>NFR1</b>	Security & Privacy	All data transactions must be secured via encryption and proper authentication. The system should comply with relevant regulations (e.g., HIPAA) where applicable.	High
<b>NFR2</b>	Performance	The system shall deliver medication reminders and process adherence confirmations within 2 seconds of the scheduled time.	Medium
<b>NFR3</b>	Availability	The system shall maintain an uptime of at least 99.9%, ensuring reliable access for patients and healthcare providers.	High

Req ID	Title	Description	Priority
NFR4	Usability	The system shall provide an intuitive, user-friendly interface that requires minimal training for both patients and healthcare providers.	Medium
NFR5	Scalability	The system shall be designed to scale efficiently, accommodating an increasing number of patients and healthcare providers without performance degradation.	Medium
NFR6	Maintainability	The system shall be developed using modular and well-documented code to facilitate ease of maintenance and future enhancements.	Medium

### 1.2.1 How to Use These Requirements

- **Functional Requirements (FRs):**

These describe the specific behaviors and functions of the system. For example, FR2 ensures that reminders are sent on time, while FR4 tracks adherence events.

- **Non-Functional Requirements (NFRs):**

These outline the quality attributes of the system. They address security, performance, availability, usability, scalability, and maintainability, which are critical for a system that handles sensitive healthcare data.

## 1.3 Use Case Diagrams

Below are two use case diagrams. One for the functional requirements and one for the non-functional requirements.

### 1.3.1 Functional Requirements Use Case Diagram



Figure 1: Use case diagram for functional requirements using Visual Paradigm

### 1.3.2 Non-Functional Use Case Diagram

Non-Functional Requirements Use Case Diagram

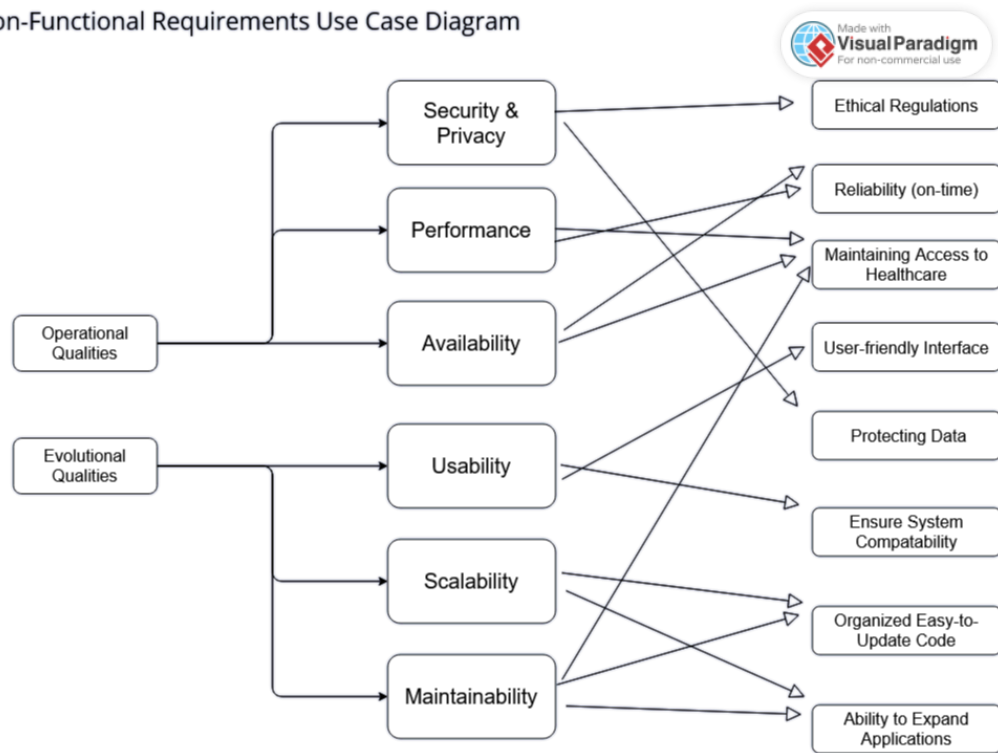


Figure 2: Use case diagram for non-functional requirements using Visual Paradigm