New York University Abu Dhabi
CS-UH 2012: Software Engineering

(Group 1)

rya2005, mn3001, fya210, maa1150

Software Requirements Specification
for
<*MediConnection* >

Version: (2) **Date: (26/02/2023)**

Table of Contents

1. Introduction

1.1 Purpose

1.1.1 The main goal of this System Software Requirements Specification for MediConnection is to succinctly and clearly describe MediConnection's primary functionality to stakeholders (professionals and experts) involved in its development process. This document also provides thorough documentation for the software requirements that will help managers, professionals, and experts develop an effective digital health aid for the subject hospital.

1.1.2 This document also serves as a resource for MediConnection software maintenance, updates, and testing. To have an accurate record of the system requirements at any time during the implementation and development of MediConnection, this document must be regularly updated whenever requirement changes are made.

1.1.3 Finally, this document can be used as a checklist after implementation to make sure that all requirements have been correctly integrated into the system. It serves as the fundamental guide for validation and testing after implementation.

This SRS is intended for the following people:

- Project managers are in charge of all matters of the software development process.
- Programmers: Individuals with a strong background in coding and program development.
- Testers: Individuals who run test cases, estimate performance, and analyze the effectiveness and efficiency of the system.
- Writers of the software's manuals and other documentation are known as documenters.

1.2 Scope

1.2.1 The software to be developed is a mobile application named “MediConnection". MediConnection is a virtual health assistant that connects patients with licensed healthcare providers in real time. The software should allow patients to access medical advice, recommendations, and prescriptions without the need for in-person visits. The software should also allow healthcare providers to access and share patient information seamlessly. MediConnection aims to automate and ease medical procedures mainly through facilitated medication intake, improved scheduling services, simple messaging, and many others. The software makes it an objective to ease communication between doctors and their patients through simple messaging, thereby limiting in-person walk-ins and consultations where applicable. MediConnection also aims to reduce traffic at the hospital by implementing an appointment booking system, rather than conventional unprompted walk-ins. MediConnenction will also provide patients with medication tracking alerts and doctors with information on how their patients are coping with medications.

1.3 Definitions, Acronyms, and Abbreviations.

Definitions:

User: A user may describe any one of the following people in the system: doctors, patients, and other healthcare professionals.

Appointment: A scheduled meeting between doctors and their assigned patients.

Medication: A set of drugs prescribed to patients by their doctors.

Symptom: Signs or descriptions a patient gives to their medical/diseases condition.

Doctor/Practitioner: Registered skilled personnel who provides medical treatment and services.
Patient: This refers to any person in the system who requires medical attention from doctors.
Dosage: Prescribed amount/dose of drugs for treatment of medical complications.
Biometric: Biological/physical data about people.
Illness/disease: A disorder or complication in one's health.

Abbreviations and Acronyms:

HTTP: Hypertext Transfer Protocol

IP: Internet Protocol

API: Application Programming Interface

SCORM: Shareable Content Object Reference Model

HIE: Health Information Exchange

EHR: Electronic Health Record

FHIR: Fast Healthcare Interoperability Resources

SOAP: Simple Object Access Protocol

REST: Representational State Transfer

HIPAA: Health Insurance Portability and Accountability Act of 1996

1.4 References

[1] B. Bruegge and A.H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns, and Java, Pearson, 3rd edition (International), Publication date: 2014. ISBN 978-1292024011.
[2] Software Engineering A Practioner's Approach, Roger S Pressmen
[3] I. Sommerville, Software Engineering, Pearson, 10th edition (Global), Publication date: March 24, 2015. ISBN 978-1292096131.

1.5 Overview

Customers or prospective users of this system are welcome to read the entire document on choice, but they should turn specifically to section 2 for the section of the document that is intended with them in view.
Developers can read the entire document, but for information on implementation that is specific to them, they should specifically consult the documentation in section 3.

2. The Overall Description

2.1 Product Perspective

MediConnection is a standalone software product that provides patients with a virtual platform to connect with healthcare providers and manage their healthcare needs remotely. It is designed to streamline the healthcare delivery process and provide patients with personalized and accurate health information.

Similar products in the marketplace include telemedicine platforms such as Teladoc, Amwell, and Doctor on Demand, which also connect patients with licensed healthcare providers remotely. However, MediConnection is tailored specifically to the needs and requirements of patients at XYZ Hospital and is fully integrated with the hospital's electronic health records and other healthcare technologies. This makes it easier for healthcare providers to access and share patient information and provide accurate diagnoses and medical advice.

MediConnection also offers patients the ability to track their symptoms and biometric data, communicate with healthcare providers via video or text chat, and access a comprehensive database of health information. Additionally, the feature allowing patients to write their symptoms and concerns and receive different suggestions and advice from healthcare advisors within a specific condition provides a unique and valuable feature.

2.1.1 System Interfaces

1. Electronic Health Records (EHRs) System Interface: MediConnection needs to interface with the hospital's EHR system to access and update patient information, including medical history, test results, and prescriptions.
2. Payment System Interface: MediConnection requires integration with a payment gateway or billing system to facilitate transactions for telemedicine consultations or prescription delivery.
3. Pharmacy System Interface: MediConnection interfaces with a pharmacy system to enable patients to order and refill their prescriptions online.
4. Medical Devices Interface: MediConnection needs to interface with various medical devices, such as blood glucose monitors, blood pressure monitors, and digital scales, to enable patients to track their health data and transmit it to healthcare providers.
5. Video Conferencing Interface: MediConnection requires integration with a video conferencing platform to enable patients and healthcare providers to communicate remotely through video or audio calls.

6. Laboratory Information System Interface: MediConnection interfaces with the laboratory information system to allow healthcare providers to access patient lab results and make informed decisions about diagnosis and treatment.
7. Medical Image Management System Interface: MediConnection needs to interface with a medical image management system to enable healthcare providers to view and analyze medical images such as X-rays, MRIs, and CT scans.
8. Appointment Scheduling System Interface: MediConnection interfaces with an appointment scheduling system to enable patients to schedule telemedicine appointments with healthcare providers.
9. Electronic Prescription System Interface: MediConnection requires integration with an electronic prescription system to enable healthcare providers to prescribe medications electronically and facilitate the delivery of prescriptions to patients.
10. Health Information Exchange (HIE) Interface: MediConnection interfaces with a Health Information Exchange (HIE) to allow the exchange of health information between healthcare providers, patients, and other stakeholders, ensuring that patient information is available across different healthcare settings and systems.

2.1.2 Interfaces

MediConnection has a user-friendly graphical user interface (GUI) that patients can access from the hospital's website or mobile app. designed for patients to easily access and manage their healthcare needs. The software's interface is optimized to provide a seamless user experience, enabling patients to enter their medical history, track their symptoms and biometric data, communicate with healthcare providers via video or text chat, and access a comprehensive database of health information. They can also write their symptoms and concerns, and receive different suggestions and advice from healthcare advisors within a specific condition.

MediConnection also offers a comprehensive database of health information, covering a wide range of topics from nutrition and fitness to common illnesses and medical procedures. Healthcare providers from XYZ Hospital can access patients' medical records and provide accurate diagnoses and medical advice without the need for in-person visits. MediConnection is fully integrated with XYZ Hospital's electronic health records (EHRs) and other healthcare technologies, making it easier for healthcare providers to access and share patient information. This streamlines the medical care delivery process, improving the overall patient experience and enhancing the efficiency of healthcare providers. In addition to the patient interface, the software also has a separate interface designed specifically for doctors and other licensed healthcare providers at XYZ Hospital. This interface provides access to patients' medical records, enabling healthcare providers to provide accurate diagnoses and medical advice without the need for in-person visits.

To ensure accessibility for all patients, MediConnection is designed with consideration for patients with disabilities. The app is compatible with assistive technologies such as screen readers, voice recognition software, and alternative input devices. Additionally, the app's user interface prioritizes clear and concise language and should avoid using small or low-contrast text, as these can be difficult for patients with visual impairments to read.

To maximize user experience, MediConnection interfaces are intuitive and easy to navigate. It provides clear instructions and feedback to guide users through the process of entering their medical history, tracking their symptoms, and communicating with healthcare providers. The app should also minimize the learning curve for new users, providing simple and straightforward tutorials and help documentation as needed.

2.1.3 Hardware Interfaces

MediConnection has specific hardware interface requirements to ensure proper functionality. The application is designed to work on mobile devices. It also requires an internet connection to access the virtual platform. The hardware devices the application will interact with and control include webcams, microphones, and speakers for video and audio communication between patients and healthcare providers. It also needs to be compatible with medical devices such as blood pressure monitors, thermometers, and glucometers for patients to monitor and track their health data.

MediConnection is fully integrated with the hospital's electronic health records system, and as such, it requires a compatible interface to access and retrieve patient information from the hospital's database. This interface must be secure and adhere to HIPAA regulations to ensure patient confidentiality and privacy. In addition, designers and developers should ensure that the hardware interface of MediConnection is optimized for the user's ease of use and accessibility. The application should be compatible with assistive technologies and follow accessibility standards to ensure that patients with disabilities can use the platform with ease.

2.1.4 Software Interfaces

The Virtual Health Assistant for XYZ Hospital requires the use of the following software products and interfaces with other application systems:

1. Operating System
   Name: Android or iOS

Mnemonic: N/A
Specification number: N/A
Version number: Latest stable version
Source: Google Play Store or Apple App Store

2. Electronic Health Record System
   Name: Epic Systems
   Mnemonic: EHR
   Specification number: N/A
   Version number: Latest stable version
   Source: On-premises installation at XYZ Hospital

3. Video Chat API
   Name: Twilio
   Mnemonic: TWILIO
   Specification number: N/A
   Version number: Latest stable version
   Source: Online developer portal

4. Symptom Checker API
   Name: Infermedica
   Mnemonic: INFERMEDICA
   Specification number: N/A
   Version number: Latest stable version
   Source: Online developer portal

5. Biometric Tracking API
   Name: Google Fit or Apple HealthKit
   Mnemonic: FITNESS
   Specification number: N/A
   Version number: Latest stable version
   Source: Google Play Store or Apple App Store

6. Payment Gateway API
   Name: Stripe
   Mnemonic: STRIPE
   Specification number: N/A
   Version number: Latest stable version
   Source: Online developer portal

Interfaces:

1. The Electronic Health Record System (EHR) interface is necessary to allow the Virtual Health Assistant to access patient records and medical history. The interface is able to receive and send data to and from the EHR system in a secure and standardized format. The interface is also able to integrate with the EHR system's authentication and authorization mechanisms to ensure only authorized personnel can access patient information.
2. The Video Chat API (TWILIO) interface allows patients to connect with healthcare providers via video chat. The interface is able to establish and maintain a secure video connection between the patient and the healthcare provider. The interface is also able to support both one-on-one and group video chats, as well as screen sharing and file sharing functionalities.
3. The Symptom Checker API (INFERMEDICA) interface allows patients to enter their symptoms and receive a list of possible conditions and recommended actions. The interface is also able to receive patient data and provide accurate and relevant information in real-time. The interface also is able to integrate with the app's database of healthcare information to provide personalized recommendations and next steps.
4. The Biometric Tracking API (FITNESS) interface allows patients to track their biometric data, such as heart rate, steps taken, and calories burned. The interface is also able to receive and transmit data to and from the app in a secure and standardized format. The interface is able to integrate with the app's data visualization tools to provide patients with a clear and intuitive way to track their progress.
5. The Payment Gateway API (STRIPE) interface allows patients to make payments for services rendered via the app. The interface is able to securely process credit card transactions and integrate with the app's user account system to ensure accurate billing and payment records.

2.1.5 Communications Interfaces

MediConnection requires the following interfaces to support communication:

1. Internet Protocol (IP) - This is the primary protocol that will be used for communication with other systems over the internet.
2. Bluetooth - This protocol will be used for communication with other devices in close proximity such as wearable fitness devices or other medical monitoring devices.
3. Wi-Fi - This protocol will be used for communication with other devices over a local wireless network. This will be useful for situations where internet connectivity may not be available or reliable.

4. Health Level 7 (HL7) - This is a standard protocol for exchanging healthcare information between medical systems. The virtual health assistant will need to interface with medical systems that support HL7 to retrieve patient health information.
5. Fast Healthcare Interoperability Resources (FHIR) - This is a newer standard protocol for exchanging healthcare information, similar to HL7. The virtual health assistant will need to interface with medical systems that support FHIR to retrieve patient health information.
6. Simple Object Access Protocol (SOAP) - This protocol will be used for communication with web services that are not transparently integrated into the application.
7. Representational State Transfer (REST) - This protocol will be used for communication with web services that are not transparently integrated into the application.

2.1.6 Memory Constraints

2.1.6.1 According to research, the minimum amount of RAM needed for the majority of mobile applications is 1GB. MediConnection is the same weight as other applications, so 1GB RAM is enough to run it without any glitches. Furthermore, MediConnection download needs 50MB of internal memory.

2.1.7 Operations

MediConnection will have various modes of operation within the user organization, including but not limited to: patient registration, appointment scheduling, symptom checking, medication reminders, and virtual consultations. The user will be able to interact with the application during periods of interactive operations such as scheduling appointments or consulting with a healthcare professional, and during periods of unattended operations such as receiving medication reminders.

Data processing support functions will include the ability to store and retrieve patient information, such as medical history, medication list, allergies, and test results. The system will also need to be able to process and analyze patient symptoms and provide appropriate recommendations based on the analysis. Additionally, the system will need to support secure communication between patients and healthcare professionals, as well as between different healthcare professionals.

Backup and recovery operations will be crucial for ensuring the continuity and integrity of patient data. The system will need to regularly back up patient information, and provide mechanisms for restoring data in the event of a system failure or other data loss event. The system will also need to have appropriate security measures in place to prevent unauthorized access or modification of patient data.

2.1.8 Site Adaptation Requirements

2.1.8.1 To be suitable for People of Determination, the system must adhere to the accessibility standards outlined in the ADA.
2.1.8.2 If the institution decides to switch up its branding themes, the system should incorporate customizable branding in its front-end development.
2.1.8.3 The system ought to support an accessible system for creating content.
2.1.8.4 The system ought to let users have access to various levels of permission.
2.18.5 The system should be able to support multiple languages so that users can choose the one they prefer.
2.1.8.6 The system interface should be designed to be compatible with a variety of devices, i.e. mobile, desktop, etc.

2.2 Product Functions

2.2.1. Patient Profiles:

- Patients will be able to create and manage their profiles, including medical records, personal information, and medical history.
- Patients will be able to view their medication schedules and history
- Patients will be able to add and remove medications from their profile through a calendar system
- Patients will be able to share their medication records and history with their doctors
- Patients will be able to send messages and video call with their doctors through the app

2.2.2. Doctor Profiles:

- Doctors will be able to create and manage their profiles including their information, specialties, and contact information
- Doctors will be able to view patient profiles, medication usage, and medication histories

- Doctors will be able to prescribe medications for patients and update patient medication schedules through a calendar system.
- Doctors can send messages to and video call with their patients through the app

2.2.3. Consulting and Appointment:

- Patients can set virtual appointments with their doctors through the app
- Patients can connect with their doctors through video conferencing and messaging
- Patients can receive reminders about their appointments and medications through the app.
- Patients can receive consultation regarding medication dosage and usage from healthcare professionals
- Patients can upload images and documents of medication history, lab reports, and anything health related to their profile
- Patients go rate and review appointments with doctors to provide feedback to administrators

2.2.4. Medication Tracking:

- Patients can add their medication dosage usages and frequency to a calendar system that can be viewed and edited by their doctors
- Patients and Doctors can view a timeline of past medication records.

2.2.5. Medication Schedule Management:

- Patients can set reminders for medication doses
- Patients can add and edit medication schedules
- Patients and doctors can receive notifications for patient missed doses.
- Patients can tick off medication doses as complete after taking their medications

2.2.6. Pill Identification:

- Patients can save pill identification numbers for reference or doctor verification
- Patients can take photos of pills for identification by the system or doctor

2.2.7. Refill Reminders:

- Patients can set reminders for when to refill their medications
- Patients can receive notifications of when to refill their medications

2.2.8. Communication System:

- Patients can send and receive messages from doctors through the app
- Patients can send automatic refill messages to their pharmacies
- Patients can receive notifications through text messages or email
- Patients can video chat with doctors and pharmacists

2.2.9. Symptom Tracking:

- Patients can log in their symptoms to document to be viewed by their doctors
- Patients can share their symptom history with their healthcare professionals

2.2.10. Integration with Existing Systems:

- MediConnection can be integrated with the hospital's electronic health record to allow healthcare professionals to view patient treatment and medication history
- Patients can access their health records from the app due to integration with the hospital patient records
- Patients can schedule doctor appointments directly from the app due to integration with hospital's scheduling system

2.2.11. Biometric Data Registration:

- Patients can register their biometric data on the app such as sugar level, blood pressure, weight, and height.
- Doctors can view patients' biometric data history.

2.2.12. Secure Storage and Data Transmission:

- The app uses secure transmission protocols and standard data encryption to secure sensitive patient information and patient-doctor communications
- MediConnection follows secure HIPAA data privacy regulations and multi-factor authentication
- The app uses secure hospital servers to store data and uses a strong data backup and recovery system

2.2.12. Analytics and Data Visualization:

- Patients and doctors can access personalized analytics and visualization of medication history
- Patients can track visualized progress towards health goals and receive a recommendation from health professionals
- Patients can access visualized data of trends and patterns in their biometric data, symptoms, and diagnosis

2.2.13. Practitioner Referral System:

- Patients can search for a hospital's healthcare practitioners based on specialty and availability through the app
- Patients can schedule appointments with health professionals other than their domestic doctors through the app

2.3 User Characteristics

2.3.1 Since MediConnection is an institutional hospital management application system, all of its users must be associated with the hospital being considered. Patients, doctors, or other professionals working for the institution can all be users of the system. Anyone who requires the services of the subject hospital is eligible to register as a patient. To restrict who can register as a doctor or practitioner in the system, access to doctor or practitioner accounts must be validated or authenticated by the system's database, that is the system must ensure that only qualified personnel can be doctors or practitioners.

2.4 Constraints

2.4.1 Java will be the primary backend language to ensure platform independence.
2.4.2 For smooth and uninterrupted performance, a strong and stable internet connection is a necessity.
2.4.3 Delay in system functions should be set to a maximum of four (4) seconds.
2.4.4 For every page in frontend development, there should be an element of the hospital's watermark present on-screen.
2.4.5 Safety and security considerations: Implementation of encryption systems and security systems to protect university data.
2.4.6 Control functions: control of user access to the system

2.5 Assumptions and Dependencies

- Assumptions: The SRS may have been based on certain assumptions, such as the availability of specific hardware or software, or the behavior of users. If these assumptions turn out to be incorrect, the requirements may need to be revised accordingly.
- External interfaces: If the mobile app needs to interact with other systems or devices, changes to those external interfaces could impact the requirements of the software.
- Regulatory or legal requirements: Any changes to regulations or laws that apply to the mobile app, such as data privacy or accessibility requirements, could affect the requirements stated in the SRS.
- Budget or resource constraints: If the budget or resources available for developing the software product change, this could impact the requirements in terms of functionality, performance, or other aspects.

- Stakeholder feedback: As stakeholders provide feedback on the mobile app, their input could lead to changes in the requirements. For example, a user may request a new feature that was not originally included in the SRS.
- Technical advancements: Changes in technology, such as new programming languages or development tools, could impact the requirements by enabling new functionality or changing the performance requirements.
- Business goals: If the business goals or objectives for the mobile app change, the requirements may need to be revised to ensure they align with these goals.
- Market trends: Changes in the market, such as the emergence of new competitors or changing customer demands, could impact the requirements by requiring new features or changes to existing ones.
- Risk factors: If new risks emerge that could impact themobile app, the requirements may need to be revised to address these risks. For example, if there is a new security vulnerability that could impact the product, the requirements may need to include additional security measures.

2.6 Apportioning of Requirements.

1. Integration with external systems and services that are not critical for the initial functionality of the system. For example, integration with electronic health records (EHRs) may be important, but integration with wearable devices or other health apps may be postponed to a later version.
2. User interface enhancements that are desirable but not essential for the initial release of the system. For example, adding animations or personalized avatars to the mobile app may improve user engagement, but can be delayed until the basic functionality is working well.
3. Localization and internationalization features that are needed to support users from different countries and cultures. These features can be added in future versions of the system as the user base expands.
4. Non-functional requirements that are important but not critical for the initial release of the system. For example, performance, scalability, and security requirements can be improved over time as the system is used and tested in real-world scenarios.

3. Specific Requirements

3.1 External Interfaces

3.1.1 User Interface

The user interface will be split into two parts depending on the profile that is accessing the application. Thus, the patient's profile will be faced with a different UI whilst the doctor's profile will be faced by a UI that is more tailored to the functional requirements of the doctors.

3.1.1.1 Patient Profile Registration Page

The patient will be able to register for their profile. The patient will input their emails, usernames, and password to create a login profile. While registering, the patient will also enter their medical record and history, and their personal information. Patients will also upload images and documents of medication history, lab reports, and anything health related to their profile, including biometrics. After successful registration, the user is sent to the login page.

3.1.1.2 Patient Profile Login Page

The patient will be able to enter their email/username and password to login into the system. If successful, the page will transition into the "home" page which is nothing but the "Patient's Scheduling and History" page described in the next section. If a username/email or password is, they will be prompted to try again or answer questions from the personal information section that was filled by the user when they first registered to reset their password.

3.1.1.3 Patient's Medication Schedule

This page will by default be set by the doctor who has prescribed the medication. The patient will then have the ability to view it and mark a medicine as taken or not (i.e., if they have taken the pill they were supposed to take at breakfast). The doctor will also have the ability to track this dosage. This page will also send notifications to the patient when the time to take medicine is nearing.

3.1.1.4 Patient's Scheduling and History

This is also the home page of the patient profile. The patient will be able to view their schedule, including not only current and upcoming appointments, but also the history of their previous visit. This page will also allow the user to schedule another appointment. Patients can search for a hospital's healthcare practitioners based on specialty and availability through this page.

3.1.1.5 Patient’s Video Call and Messaging Interface

This page will allow the user to message the doctors they have already had an appointment with or are scheduled to have one. Given that it’s time for the patient’s scheduled appointment, the patient will also be able to video call the doctor. The same page will also allow patients to rate and give feedback on the doctor after they are done with the scheduled appointment with the doctor.

3.1.1.6 Patient’s Symptom Tracking

This page will allow the patients to log in their symptoms to be documented and be viewed by their doctors.

3.1.1.7 Doctor Profile Registration Page

The doctor will be able to register for their profile. The patient will input their emails, usernames, and password to create a login profile. While registering, the doctor will also list their specialty, languages they can speak, their history of where they have worked and the education they have received. After successful registration, the user is sent to the login page.

3.1.1.8 Doctor Profile Login Page

The doctor will be able to enter their email/username and password to login into the system. If successful, the page will transition into the “home” page which is nothing but the “Doctor’s Schedule and History” page described in the next section. If a username/email or password is, they will be prompted to try again or answer questions from the personal information section that was filled by the user when they first registered to reset their password.

3.1.1.9 Doctor’s Prescribed Medication

The doctor will be able to prescribe and edit the “Medication Schedule” for the patient they have an appointment with. They can also use the same page to track the medication usage of their patient. They can access this page again by going to their history on the scheduling page (described in the next section), clicking on the the patient they want to access this page for and accessing this page.

3.1.1.10 Doctor's Scheduling and History

This is also the homepage of the doctor's profile. This will display the current, upcoming, and previous they have had. The doctor will be able to schedule "busy/off" times on this page which will prevent the patient from scheduling an appointment at that time.

3.1.1.11 Doctor's Video Call and Messaging Interface

This page will allow the doctor to message the patients they have already had an appointment with or are scheduled to have one. The will also be able to accept/call the patients they have an appointment with, given that the appointment time is active.

3.1.1.12 Patient's Symptom Tracking (From Doctor's Side)

This page will allow the doctor's to access this page from the patient's profile and check for the symptoms they have been updating. This is necessary so the doctor can alter the dosage, change or stop the medication for a patient that might be facing side-effects.

3.2 Functions

3.2.1. User Registration and Profile Management:

- The system shall validate user information when registering to the app
- The system shall use multi-factor authentication for validating user login to the app
- The system shall have a username and password as login to the system
- The system shall allow users to update their profile with personal information, a photo of themselves and health records

3.2.2. Appointment and Consulting

- The system shall allow patients to schedule appointments with their doctors and other healthcare professionals at the hospital
- The system shall generate automatic reminder notifications to patients for appointments made
- The system shall provide a platform for virtual appointment meetings via video conferencing and messaging systems
- The system shall provide a shared platform for doctors and patients to view appointment records, diagnosis and prescribed medications

3.2.3. Medication Tracking and Reminders:

- The system shall provide a calendar system for patients and doctors to schedule medication timings and dosage.
- The system shall notifications at the scheduled times to remind patients to take their medications
- The system shall provide a checkmark system for patients to tick or sign after taking their medications
- The system shall use alert systems to prevent patients from not taking their medications
- The system shall allow doctors to view patient medication schedules and the check mark system

3.2.4. Symptom Tracking:

- The system shall allow patients to record and track their systems
- The system shall provide visualizations and reports of symptoms
- The system shall enable a validation system for doctors to validate and review patient symptoms
- The system shall allow for doctors to adjust patient medications and dosages per the tracking of symptoms

3.2.5. Biometric Data Registration:

- The system shall allow patients to enter their biometric data
- The system shall validate the biometric data entered by patients to ensure they are between accepted ranges and formats
- The system provide reports and visualizations of biometric data over time to show trends and patterns

3.2.6. Integration with Existing Healthcare Systems:

- The system shall integrate with existing hospital healthcare systems
- The system shall comply with standards for data exchange between systems
- The system shall allow the smooth exchange of data between existing hospital systems
- The system shall be able to synchronize appointment and medical records between the app and the hospital healthcare systems

3.2.7. Communication System:

- The system shall allow patients and doctors to communicate via messaging systems and video conferencing

- The system sends notifications and alerts to patients about medication schedules, appointments, and messages received
- The system shall allow patients to upload documents and images about medical records and lab results
- The system shall use data encryption standards to ensure secure communication between patients and doctors

3.2.8. Secure Data Transmission and Storage:

- The system shall standard encryption and data transmission protocols to secure user data and communication
- The system shall comply with HIPAA standards for data privacy and storage
- The system shall provide access controls for users to access and manage their data

3.2.9. Data Visualization and Analytics:

- The system shall provide data visualization and analysis of patient medication history and records
- The system shall provide visualizations to help track patient progress in medications and symptoms

3.2.10. Doctor and Practitioner Referral System:

- The system shall allow users to search for profiles of registered healthcare professionals by specialty on the app
- The system shall allow patients to request appointments from healthcare professionals other than their domestic doctor
- The system shall allow healthcare professionals to view and access patient medication history to facilitate diagnosis and treatment
- The system to provide secure messaging between patients and healthcare professionals

3.3 Performance Requirements

3.3.1 The system shall support up to 10000 simultaneous users

3.3.2 The system shall support up to 600 terminals simultaneou

3.3.3 All system operations should have a maximum delay of eight seconds.

3.3.4 The system should allow up to 9000 active users simultaneously.

3.3.5 A single user should be allowed to do at most two concurrent sign-ins.

3.3.6 The system shall process 10000 user logins per hour during peak workload conditions.

3.4 Logical Database Requirements

3.4.1.Types of information used by various functions

3.4.1.1. Patient Information: Patient name, email address, Patient ID, medication records, diagnosis history.

3.4.1.2. Doctor Information: Doctor name, Doctor Speciality, contact details, availability schedules, diagnosis and treatment records

3.4.1.2. Medication Information: drug name, dosage, timings, schedule, frequency, route of administration

3.4.1.3. Appointment information: date, time, location, healthcare professional involved, and reason for the appointment.

3.4.1.4, Communication data: messages, files, and documents exchanged between healthcare professionals and patients.

3.4.2. Frequency of use:

3.4.2.1. Patient profiles and medication information will be frequently accessed by both patients and healthcare professionals.

3.4.2.2. Symptom and biometric data will be updated periodically based on the patient's condition.

3.4.2.3. Appointment and communication data will be accessed as needed by both patients and healthcare professionals.

3.4.3. Accessing capabilities:

3.4.3.1. Patients will have read and write access to their own profile and medication information.

3.4.3.2. Healthcare professionals will have read and write access to patient profiles, medication information, appointment information, and communication data.

3.4.3.3. Access controls will be implemented to ensure privacy and security of patient data

3.4.4. Data entities and their relationships:

3.4.4.1. Patient profiles will be related to medication information, symptom data, and biometric data.

3.4.4.2. Healthcare professional profiles will be related to appointment information and communication data.

3.4.4.3. Medication information will be related to prescription data and treatment plans.

3.4.5. Integrity constraints:

3.4.5.1. The system will enforce data validation and ensure that only valid information is stored in the database.

3.4.6. Data retention requirements:

3.4.6.1. The system will store patient data for a specified period based on regulatory requirements and data retention policies.

3.4.6.2. Data that is no longer needed will be securely and permanently deleted from the database.

3.5 Design Constraints

3.5.1 Standards Compliance

3.5.1.1 The system must adhere to SCORM requirements.

3.5.1.2 To protect user data, the system must employ data encryption.

3.5.1.3 To avoid vulnerabilities, the system must undergo routine security audits and updates.

3.5.1.4 The system ought to support captions and alternative text for in-application images.

3.5.1.5 The system's design must be flexible enough to accommodate various screen sizes and orientations.

3.5.1.6 To reduce device data usage, the system must adopt quick loading times.

3.5.1.7 The system must support screen readers and allow keyboard navigation.

3.6 Software System Attributes

3.6.1 Reliability

3.6.1.1 The system must accurately record and report exceptions so that programmers can quickly identify problems and fix bugs.

3.6.1.2 To make sure that the system's application interface satisfies user expectations, Selenium testing should be used.

3.6.1.3 To ensure that each component of the system is operating correctly, extensive unit and integration tests should be run before deployment.

3.6.1.4 The system should have backup servers to prevent instances of data loss in case of active server failure.

3.6.1.5 The try-catch block of Java should be used by the system to handle exceptions and avoid crashes when unforeseen errors occur.

3.6.1.6 For a scalable architecture, the system should make use of Java's multi-threading and distributed computing capabilities to manage user traffic.

3.6.2 Availability

3.6.2.1 A performance monitoring system for the system is required, notifying developers of problems before they become serious enough to require downtime.
3.6.2.2 The system needs a standby database as a failover component to ensure availability in the event of a systemic failure.
3.6.2.3 The system should use load balancers to distribute traffic across servers and prevent overloading.
3.6.2.4 To make changes easy to track and undo as necessary, the system should implement a strong version control system.
3.6.2.5 To increase availability, the system should use a cloud-based server system design.
3.6.2.6 The system should be accessible 24/7 for convenience except during update hours.

3.6.3 Security

3.6.3.1 Spring Security must be integrated with the system as its security framework.
3.6.3.2 Secure coding techniques must be used in the system's code implementation to thwart attacks like SQL injection.
3.6.3.3 To fend off attacks, the system needs to be equipped with a firewall and an intrusion detection system.
3.6.3.4 To ensure the secure transmission of sensitive data, the system must use Java's cryptography APIs.
3.6.3.5 The system should maintain an application history log to troubleshoot bugs, investigate outages, and analyze security concerns.

3.6.4 Maintainability

3.6.4.1 The system must use patterns like Mode-View-Controller (MVC) for easy design management.
3.6.4.2 The system needs thorough documentation for its codebase and must be testable.
3.6.4.3 For the system to enhance loose coupling between components, dependency injection frameworks like Spring should be used.
3.6.4.4 The system should use Java’s package and class structure to break the backend components of the system into small manageable components.

3.6.4.5 The system must have a robust version control system for smooth updates.
3.6.4.6 To prevent data loss, the system should regularly be backed up.

3.6.5 Portability

3.6.5.1 The system shall use little to no host-dependent code in its codebase to ensure platform independence and easier porting.
3.6.5.2 The system shall be designed to be platform-agnostic and use standard APIs and libraries to make porting across platforms easier.
3.6.5.3 The system should isolate system dependencies to achieve portability.

3.8. Additional Information

Questionnaires used in the requirements gathering process:

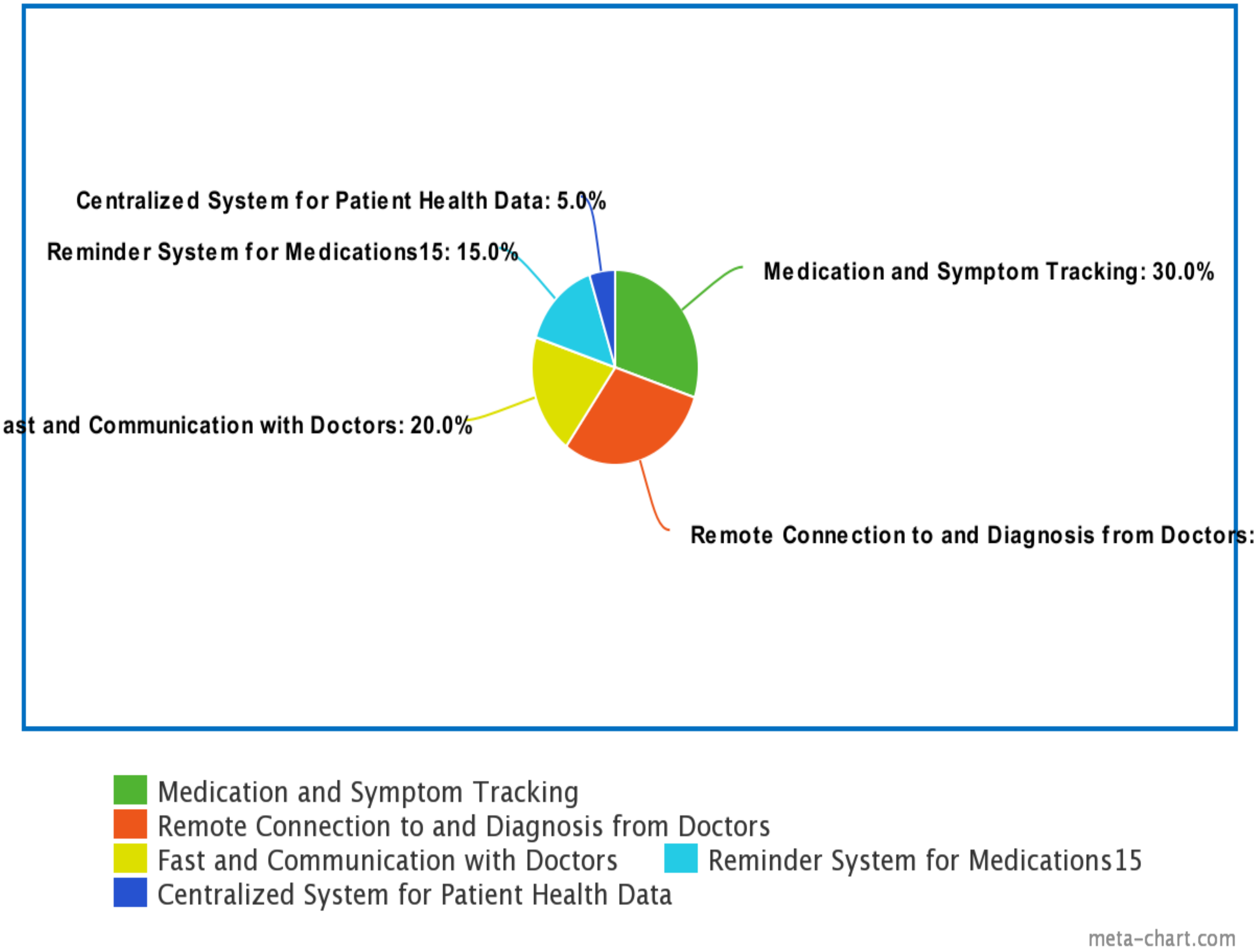
1. What is the primary goal of the medication app? Is it for symptom tracking, medication scheduling and dosage tracking or connecting with doctors?
2. Who will be the main users of the app? Patients, healthcare professionals or both?
3. How will medication dosages and scheduling be tracked on the app?
4. How will patient health information and diagnosis be managed on the app and who has access to the information?
5. How will patients connect with doctors through the app? Through simple messaging or video conferencing?
6. Which existing hospital healthcare systems will be integrated with the app?
7. How important will security and data privacy be for the app? Will the app follow standard security and data transfer protocols?
8. What type of medication will be tracked on the app? Prescription medication or over-the-counter medication?
9. How important is virtual consultation functionality in the app development process? Is fast-track consultation with medical professionals important?
10. How will the app handle unintended cases such as missed appointments or medications and medication overdose?

Information gathered from randomly sampled patients at the hospital:

- 75% of patients reported that they forget to take their medications on time

- 85% of patients reported that getting in touch with healthcare professionals remotely is a major challenge
- 70% of patients complained of long waiting times for consultation at the hospital

App Functionality Priority Distribution:



4. Change Management Process

4.1 The agile model is used in MediConnection's system engineering. In this, the development team effectively embraces change while collaborating closely with customers and working incrementally and iteratively. Customer requirements should be given priority by the team, and any requirement changes or specifications must be submitted to the team in writing via user stories. Customers and the development team will participate in sprint reviews, where the development team shows the customer the work they have finished, in order to assess the progress of the project. The client can then comment on the work that has been delivered and make suggestions for amendments or new specifications for the following iteration.

5. Document Approvals

Name :
Signature:
Date:

6. Supporting Information

- Table of Contents

- 2.2 Product Functions
- 2.3 User Characteristics
- 2.4 Constraints
- 2.5 Assumptions and Dependencies
- 2.6 Apportioning of Requirements

3. Specific Requirements
    - 3.1 External interfaces
    - 3.2 Functions
    - 3.3 Performance Requirements
    - 3.4 Logical Database Requirements
    - 3.5 Design Constraints
        - 3.5.1 Standards Compliance
    - 3.6 Software System Attributes
        - 3.6.1 Reliability
        - 3.6.2 Availability
        - 3.6.3 Security
        - 3.6.4 Maintainability
        - 3.6.5 Portability
    - 3.7 Organizing the Specific Requirements
    - 3.7.1 System Mode
    - 3.7.2 User Class
    - 3.7.3 Objects
    - 3.7.4 Feature
    - 3.7.5 Stimulus
    - 3.7.6 Response
    - 3.7.7 Functional Hierarchy
    - 3.8 Additional Comments
4. Change Management Process
5. Document Approvals
6. Supporting Information