DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

University of Chittagong

Course Code	CSE 618
Course Title	Mobile Apps Development Lab
Report No	01
Report Name	Software Requirement Specification
Group	E

Submitted to:

Dr. Fahim Irfan Alam

Associate Professor

Department of Computer Science and Engineering

University of Chittagong.

Submitted by:

ID:18701032

ID:18701057

ID:18701071

ID:17701101

1. Preface

This is software requirement specification documents (SRS) of the Medical Management System. The purpose of an SRS is to list and prioritize all requirements set for the software being produced, and so work as an agreement between the project personnel and the customer. This document provides a basis for future enhancements and it also reduces development effort. This document describes the project's target audience and its user interface, hardware and software requirements. With version 1.0, the users will experience a completely stable release that includes high authentication of the users like patients, doctors, receptionist, nurse and pathologist. So, there is no worry about unauthenticated users. The version 1.0 will also provide a very secure and encrypted data submission and preview. Users will experience a very fast, smooth, and accurate and reliable health care services using this release.

2. Introduction

A software requirements specification (SRS) is a detailed description of a software system to be developed with its functional and non-functional requirements. The SRS is developed based on the agreement between customer and contractors. It may include the use cases of how the user is going to interact with the software system. The purpose of this document is to give a detailed description of the requirements for the "Medical Management System". It will provide a complete declaration for the development of the system in a precise and explicit manner. It will also explain system constraints, what system will do and how the system will react to external interactions.

"Medical Management System" is an online-based mobile app that helps patients to give necessary information. The system will manage a variety of users' information including patients, doctors, nurses, administrator profiles, and records of patients securely. It will be designed to access information from home and get updates quickly. Also, patients can make appointments to use this system. Doctors may create, edit, or delete patient charts. System will generate reminders for particular patients to schedule appointments. System will maintain the confidentiality, integrity, and availability of the information.

The SRS document includes ten sections. Section 1 describes the preface of this document while section 2 introduces the readers with the system as well as with this document. Section 3 familiarizes the readers with the technical terms used in this document. Section 4 narrates the user requirements definition while section 5 gives a brief description to the system architecture. The specific description of user requirement is described in section 6. Use case scenarios of the system are given in section 7 and section 8 bears the anticipated change or evolution of the system due to hardware changes. Section 9 and section 10 contains the appendices and index for this document respectively.

3. Glossary

Glossary is an alphabetical list of terms in a particular domain of knowledge with the definitions for those terms. It lists the technical terms used in the document. The glossary for this document is given in Table 1.

Table 1: Glossary

Technical Term	Description
Authentication	The process or action of verifying the identity of a user or process.
Backup	A copy of a file or other item of data made in case the original is lost or damaged.
Constraints	The limiting barrier of an action or a system.
Credentials	A group of information proving a user's identity or qualifications.
Database	A collection of information organized into rows, columns and tables, such a way that a computer program can quickly access, manage or update desired pieces of data.
Encryption	The process of converting information or data into a code, especially to prevent unauthorized access.
Login	The process by which an individual gains access to a computer system by identifying themselves.
Online	Operating being connected to a computer or telecommunication system such as internet.
Response Time	The length of time taken for a system to react to a given event.
Server	A computer or computer program which manages access to a centralized resource or service in a network.

4. User Requirements Definition

Requirements are physical or functional need that a particular design, product or process aims to satisfy. After meeting with the client and properly discussing with them, some requirements are discovered. The requirements are divided into two categories such as, *functional requirements*, which defines the functions of the system required by the client, and, *nonfunctional requirements*, which defines the characteristics as well as constraints of the system. The user requirements are defined in table 2.

Table 2 Definition of user requirements

Requirement Type	Definitions of Requirement
	1. Patients will make appointment using system
	2. Doctor will have option to use digital prescription for patients

Functional Requirement	3. Doctor and patient will be able to Communicate via telemeeting
	4. Patients will make online payment using system
	5. System will send reminder to schedule appointments for patient and corresponding doctor
	6. System will make plot to visualise the medical information.
Non-Functional Requirement	7. System must store the user medical information in an encrypted form
	8. System must ensure access control to access the medical information for the authorized users.
	9. System must response quickly
	10. System must require low hardware resources to operate
	11. System must be operable on android phone
	12. System must be user friendly

5. System Architecture

A system architecture is the conceptual model that defines the structure of a system. An architecture description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system. The system architecture of Medical Management System (MMS) is depicted in Figure 1.

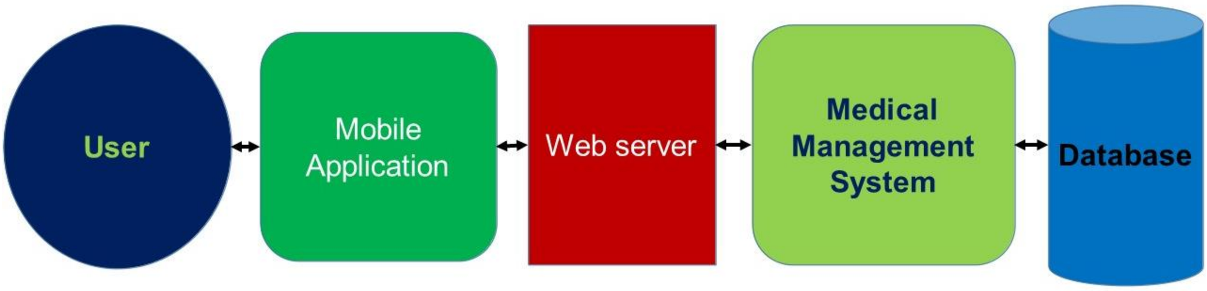

Figure 1: System architecture

6. User Requirements Specification

The software requirements specification document enlists necessary requirements that are required for the project development. To derive the requirements, the developer needs to have clear and thorough understanding of the products to be developed. Also, the requirements must be described at length for making the client clear and concise about what is going to be developed. The user requirements are described in great detail in table 3.

Table 3: Specification of user requirements

Requirement Type	Specifications of Requirements
Functional Requirement	1.1 System will show the appointment form 1.2 User will fill up the required information 1.3 User will make confirmation the appointment
	2.1 System will show the patient appointment list 2.2 Doctor will select prescription against the patient’s name 2.3 System will automatically fill up the patient information 2.4 Doctor will fill up the medicine information 2.5 Doctor will add feedback and suggestions 2.6 Doctor will make sure the submission
	3.1 Patient can show the list of doctors with whom he/she took appointment 3.2 Doctor can also show the list of patients who were taking appointment with him/her 3.3 Patient can contact with his/her doctors via video call and vice versa
	4.1 Accountant store patient information in system and determine how much money patient have to pay 4.2 Patient complete payment via online
	5.1 System will send message to patients for scheduling appointment 5.2 System will also send message to doctors to inform about their duty
	6.1 System analyse patients’ medical information and generate different graph 6.2 User can view that graph
	7.1 System will encrypt user personal information through highly secured RSA and PGP encryption technique

Non-functional Requirement	7.2 System will decrypt the encrypted data using private key
	8.1 Every actor must have a unique user-account. 8.2 Each user will have to login his own account submitting his credentials. 8.3 No operation will be allowed to unauthorized actors. 8.4 Only Administrator can modify and remove any other actors profile information 8.5 Only Doctor can view patients EHR report 8.6. Patient can view only his/her appointment information
	9.1 Overall response time will be less than 1 second. 9.2 System must be able to serve at least 300 users simultaneously. 9.3 System must be able to handle 150 appointment per second 9.4 System must be able to generate 20 EHR report per second
	10.1 System must be low power consuming. 10.2 The software must be operable with a minimum of 900MHz dual core CPU. 10.3 Software must be restricted within 512MB of RAM.
	11.1 The software must be able to run on android API 19 or higher. 11.2 User-interface must be nice-looking and user-friendly. 11.3 The software must be able to run in a device with 480dpi screen.
	12.1 The software UI must be design to keep mind all age people preference 12.2 The target user will be low level and high level both type of technical knowledgeable person

7. System Model

A. Use Cases

Use case is a list of actions or event steps typically defining the interactions between a role and a system to achieve a goal. Use cases are important to understand how the system interacts with the user or other systems. The use cases of the system are listed in table 4.

Table 4: List of Use Cases

Use Case	Title
UC1	Make Registration
UC2	Log In to the System
UC3	Make Appointment
UC4	Generate Digital Prescription
UC5	Communicate via telemeeting
UC6	Pay via Online Payment
UC7	Send Reminder
UC8	Generate Plot

The use cases are described in details in following:

UC1: Make Registration

Actors:

1. Administrator
2. Doctor
3. Nurse
4. Patient
5. Accountant
6. Receptionist

Preconditions:

1. Device must be powered on
2. Device is unlocked and in standby mode
3. Software must be installed in the device
4. User open the *medico* system application
5. Device is connected to the central server via LAN or Internet
6. User don’t have any previous account in the system

Main success scenario:

1. Click *Create Account* button to open registration window
2. Fill up required information for corresponding user
3. Click *Submit* button

4. Redirect to temporary page before email or phone credential verification
5. Verify email or phone successfully

Post condition:

1. Create your account profile successfully
2. Redirect to the login window

Alternative Course:

2.a Fill up the same email address or username which already exist an account
   2.a.1 Try to use unique email address or username
2.b Type incorrect format of email address
   2.b.1 Correct the format of the email address
4.a Verification credentials does not send
   4.a.1 Click *Resend* button to resend it
5.a Email or phone is not verified
   5.a.1 Repeat steps 2-5 again with valid information

UC2: Log into the System

Actors:

1. Administrator
2. Doctor
3. Nurse
4. Patient
5. Accountant
6. Receptionist

Preconditions:

1. Device must be powered on
2. Device is unlocked and in standby mode
3. User open the *medico* system application
4. Device is connected to the central server via LAN or Internet
5. User must have an account in the system

Main success scenario:

1. Click *Login* button to open login window
2. Fill up required information
3. Click *Log In* button

Post condition:

1. Your account is login successfully
2. Redirect to the corresponding user window

Alternative Course:

3.a Login not successful
   3.a.1 Fill up the correct email address or username, password and user role field
3.b Need new account
   3.b.1 Click *Create Account* button (UC2)

UC3: Make Appointment

Actors:

1. Administrator
2. Doctor
3. Receptionist
4. Patient

Precondition:

1. User open the Apps
2. User must be connected to the internet
3. User must have an account in the system
4. User must be logged in to the system
5. Phone is showing dashboard

Main success scenario:

1. Click the *Make Appointment* button
2. System shows the appointment form
3. User fill up the required information
4. Click the *Confirm Appointment* button

Post condition:

1. Patient will view his/her appointment
2. Other actors will view the updated appointment list

Alternative course:

2.a System doesn’t show the appointment form
2.a.1 Make sure you connected to the internet
2.a.2 Refresh the system
4.a User’s cannot view the newly added appointment
4.a.1 Make sure your internet connection is OK
4.a.2 Follow step 3-4

UC4: Generate Digital prescription

Actors:

1. Doctors
2. Patients
3. Nurses

Pre-condition:

1. Doctors must login into the system
2. System will connect to the server and internet
3. Patient successfully completed visiting fee transaction
4. Patient completed recommended diagnosis

Main success scenario:

1. Doctor click the *Appointment list* button
2. System shows the patient appointment list
3. Doctor click the patient’s name
4. System shows the *Prescription* button against patients name

5. Doctor click the *Prescription* button
6. System will show the prescription page and the page will automatically fill the patient's data
7. Doctor fill up the medicine information
8. Doctor add feedback and suggestion for patient
9. After filling all the information, doctor click the *Submit* button

Post condition:

1. After submission, doctor will show prescription from doctor’s dashboard
2. After submission, patient will show prescription from patient’s dashboard

Alternative course:

2.a. System is not showing any appointment list

- 2.a.1. Make sure your connection to server is OK
- 2.a.2. Tap the *Refresh* button

6.a. System is not automatically filling patients information

- 6.a.1. Make sure your connection to server is OK
- 6.a.2. Tap the *Refresh* button

9.a. System is not successfully submitted information

- 9.a.1. Make sure you fill up all must require information field
- 9.a.2. Fill up all must require field
- 9.a.3. Tap the *Refresh* button
  - 9.a.3.1 Follow steps 5-9 again

UC5: Communicate via Telemeeting

Actors:

1. Patient
2. Doctor

Preconditions:

1. User is logged in
2. The phone is unlocked
3. The device must be connected to the Internet
4. System must have permission to access camera and microphone
5. Patient must have a telemeeting appointment

Main Success Scenario:

1. Select receiver from contact list
2. Click the video icon of the device to start a video call
3. After clicking the icon start a call from caller side and show a message (Calling) in the screen
4. When receiver listening the incoming call, receiver side show a message (Incoming video call) and caller side show a message (Ringing) in the screen
5. If receiver accept the incoming call by clicking green button, both caller and receiver can communicate with each other via video call
6. Both caller and receiver can end call according to their wish by clicking red button

Post Condition: Phone shows a video calling interface

Alternative Course:

2.a Do Not start a video call

2.a.1 Make sure your device's internet connection is OK

2.a.2 Make sure your device is not in Aeroplane mode

5.a If receiver does not accept the incoming call

5.a.1 Repeat the steps 2-4 again

UC6: Pay via online payment

Actors:

1. Accountant
2. Patient
3. Administrator

Preconditions:

1. User must be logged in into the system
2. Device must be connected to a stable internet connection

Main success scenario:

1. Accountant click on *Generate Invoice* button to open invoice form
2. Accountant fill up the cart list information for corresponding patient
3. Click on *Generate* button to generate invoice and send to patient dashboard
4. Patient click on *Payment* button to pay bill according to invoice
5. System redirect patient to payment gateway
6. Patient successfully complete transaction procedure

Post condition:

1. Patient payment status will be updated
2. Accountant will view the transaction information
3. Administrator will view the overall updated transaction statistics

Alternative course:

3.a Invoice is not generating

3.a.1 Refresh the system and ensure that you fill up all must required field

3.b Does not send any invoice to the patient

3.b.1 Make sure you are connected to the internet

3.b.2 Repeat steps 2-3 again

6.a Payment is unsuccessful

6.a.1 Make sure you are connected to the internet

6.a.2 Make sure you have sufficient balance and use valid credentials

6.a.2 Repeat steps 4-6 again

UC7: Send Reminder

Actors:

1. System
2. Doctor

Preconditions:

1. Patient must have an account in system
2. Patient have previous medical history in system
2. Doctor have provided tentative date
3. The device must be connected to the Internet

Main Success Scenario:

1. System fetch patients data from the database
2. Make a sorted patient list according to time
3. Server will send appropriate reminder message to patient and corresponding doctor

Post Condition: Patients will receive a reminder to make appointment

Alternative Course:

3.a Don’t receive any reminder message

3.a.1 Make sure doctor provide tentative date correctly

3.a.2 Doctor will update the schedule

UC8: Generate Plot

Actors:

1. Administration
2. Doctor
3. Patient
4. System

Pre-condition:

1. Phone must installed the apps
2. User must be connected to the internet
3. User login to the system
4. System will show the dashboard

Main Success scenario:

1. User click the *Plot* icon
2. System will generate the plot details
3. User will view plot details

Alternative course:

2.a System won't generate plot automatically

2.a.1 Make sure you connected to the server

2.a.2 Refresh the system

B. Use Case Diagram

The use case diagram shows the interaction between user and the system graphically. The use case diagram is shown in Figure 1

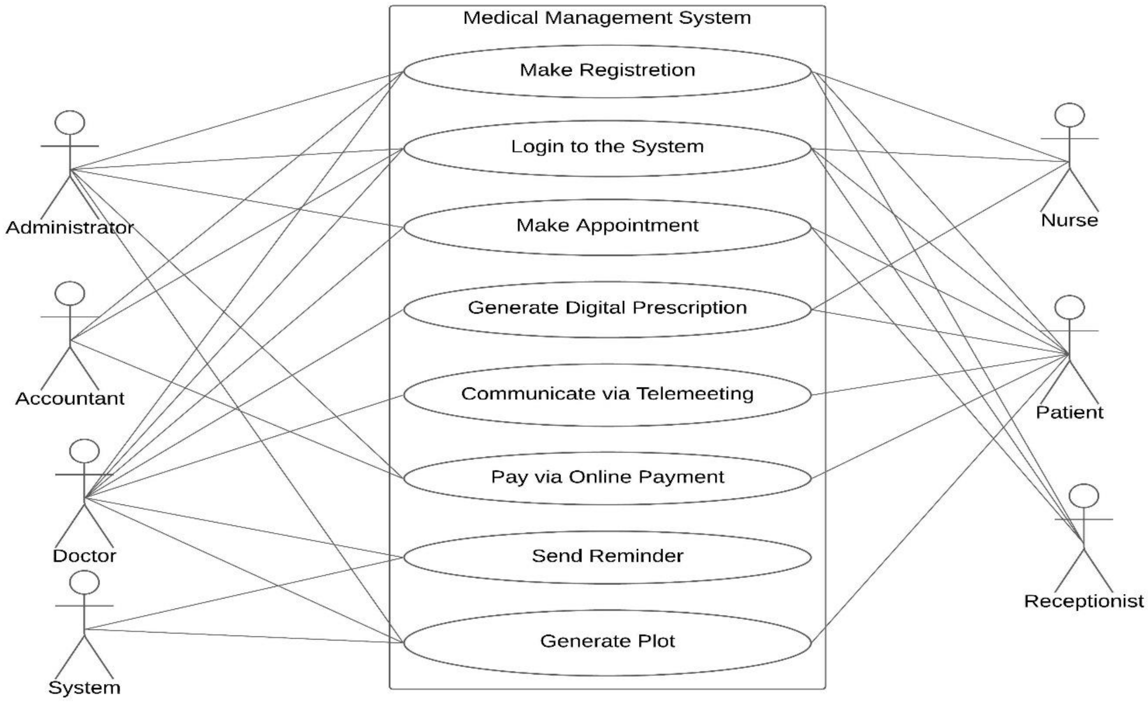

Figure 2: Use case diagram

8. System Evolution

Software Evolution is a term which refers to the process of developing software initially, then timely updating it for various reasons. The evolution process includes fundamental activities of change analysis, release planning, system implementation and releasing a system to customers.

The Medical Management System is developed to be an adaptive system. It is implemented in such a way that it adjusts its performance with respect to the specification of the hardware such as servers. With version 1.0, the system can process as much as 300 requests per second for a server with processor speed of 5 GHz, consisting of 4 logical cores and maximum memory of 1GB. Any change to the hardware would change the performance of the system in proportion to the change of processor speed, number of cores and maximum memory.

9. Appendices

Appendices contains the texts that is explanatory, statistical, or bibliographic in nature. The appendix for this document contains the hardware specification, database specification for the system.

Appendix A : Hardware Specification

The system is developed using the server “HPE ProLiant ML10 Gen9 Tower Server”. The specification of the server is given in Table :5

Table 5: Server specification

Processor	Intel® Xeon® E3-1225 v5
Number of Processors	1
Processor Core Available	4
Processor Cache	8MB (1 x 8MB) Level 3 cache
Processor Speed	3.3GHz
Chipset	Intel® C236 Chipset
Power Supply Type	300W Multi-Output Power Supply
Memory	4GB DDR4
Memory Slots	4 DIMM slots
Memory Type	1R x8 PC4-2133P-E-15
Memory Protection Features	Un-buffered ECC
Included Hard Drives	LFF SATA; 1TB
Maximum Internal Storage	24TB
Optical Drive Type	SATA 9.5mm DVD RW
System Fan Features	Non-Pluggable Fan
Network Controller	Intel® Ethernet Connection I219-LM
Storage Controller	Integrated SATA RAID
Infrastructure Management	Intel® Active Management Technology (Intel® AMT 11.0)

Appendix B: Database Specification

The system uses the “MySQL Enterprise Edition” as database management system. The technical specification of the database management system is given in Table 6.

Table 6: Database specification

Version	5.7
Data Type	Static
Architecture	Relational Model
Operating System	Linux, Solaris, FreeBSD, Mac OS, Windows
Software License	GNU General Public License
Security	• SSL Support • Built-in Data Encryption/Decryption • View Support • Triggers for auditing • Query Logs for auditing
Access Control	❖ Enterprise Directory Compatibility ❖ Native Network Encryption ❖ Run Privilege ❖ Security Certification

Indexes	• R-/R+ Tree • Hash • Full-text • Spatial
Partitioning	▪ Range, Hash, List, Key ▪ Composite ▪ 8k partitions per table ▪ Portable partitions between tables ▪ Explicit querying by partition ▪ Transparent Pruning
Max Database Size	Unlimited
Max Table Size	256 TB (MyISAM) 64 TB (Innodb)
Max Row Size	64 KB
Max Column Per Row	4096
Max Blob/Clob Size	4 GB
Max CHAR Size	64 KB
Max NUMBER Size	64 Bit

10. Index

A. List of Figures

Figures are graphical representation of information. The figures used in this document is listed in table 7

Table 7: List of Figures

B. Alphabetic