**CEBU INSTITUTE OF TECHNOLOGY**

**UNIVERSITY**

COLLEGE OF COMPUTER STUDIES

Software Requirements Specifications

for

Hospital Management System

Developers:

**Flores, Adrian Ash D.**

**Calzada, Earl Owen V.**

**Rivera, Benjie Jr.**

Submitted to:

**Mr. Frederick L. Revilleza Jr.**

**Professor**

Change History

Table of Contents

Change History 2

Table of Contents 3

1. Introduction 4

1.1. Purpose 4

1.2. Scope 4

1.3. Definitions, Acronyms and Abbreviations 4

1.4. References 4

2. Overall Description 5

2.1. Product perspective 5

2.2. User characteristics 5

2.4. Constraints 5

2.5. Assumptions and dependencies 6

3. Specific Requirements 7

3.1. External interface requirements 7

3.1.1. Hardware interfaces 7

3.1.2. Software interfaces 7

3.1.3. Communications interfaces 7

3.2. Functional requirements 7

Module 1 7

Module 2 8

3.4 Non-functional requirements 8

Performance 8

Security 8

Reliability 8

# Introduction

The **Hospital Management System (HMS)** is a comprehensive digital platform designed to streamline and enhance the operational processes within healthcare facilities. It offers both **web** and **mobile** applications that aim to improve the efficiency of patient management, doctor-patient communication, inventory control, billing, and telemedicine services.

## Purpose

* *Describe the purpose of the SRS;*
* *Specify the intended audience for the SRS.*

## Scope

* *Identify the software product(s) to be produced by name (e.g., Host DBMS, Report Generator, etc.);*
* *Explain what the software product(s) will, and, if necessary, will not do;*
* *Describe the application of the software being specified, including relevant benefits, objectives, and goals;*
* *Be consistent with similar statements in higher-level specifications (e.g., the system requirements specification), if they exist.*

## Definitions, Acronyms and Abbreviations

* *provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS*

## References

* *Provide a complete list of all documents referenced elsewhere in the SRS;*
* *Identify each document by title, report number (if applicable), date, and publishing organization;*
* *Specify the sources from which the references can be obtained.*

# Overall Description

## Product perspective

* *Put software product into perspective with other related products. If the product is independent and totally self-contained, it should be so stated here. If the SRS defines a product that is a component of a larger system, as frequently occurs, then this subsection should relate the requirements of that larger system to functionality of the software and should identify interfaces between that system and the software.*
* *A block diagram showing the major components of the larger system, interconnections, and external inter- faces can be helpful.*
* *Describe the modular decomposition of the components using the format below:*

*Module 1*

*Transaction 1.1*

*Transaction 1.2*

*Module 2*

*Transaction 2.1*

*Transaction 2.2*

*. . .*

## User characteristics

* *Describe all user types and their roles and privileges in the system*

## 2.4. Constraints

* *Provide a general description of any other items that will limit the developer’s options.*
* *Regulatory policies;*
* *Hardware limitations (e.g., signal timing requirements);*
* *Interfaces to other applications;*
* *Parallel operation;*
* *Audit functions;*
* *Control functions;*
* *Reliability requirements;*
* *Criticality of the application;*
* *Safety and security considerations.*

## 2.5. Assumptions and dependencies

*This subsection of the SRS should list each of the factors that affect the requirements stated in the SRS. These factors are not design constraints on the software but are, rather, any changes to them that can affect the requirements in the SRS. For example, an assumption may be that a specific operating system will be available on the hardware designated for the software product. If, in fact, the operating system is not avail- able, the SRS would then have to change accordingly.*

# Specific Requirements

## External interface requirements

### 3.1.1. Hardware interfaces

*This should specify the logical characteristics of each interface between the software product and the hard- ware components of the system. This includes configuration characteristics (number of ports, instruction sets, etc.). It also covers such matters as what devices are to be supported, how they are to be supported, and protocols. For example, terminal support may specify full-screen support as opposed to line-by-line support.*

### 3.1.2. Software interfaces

*This should specify the use of other required software products (e.g., a data management system, an operating system, or a mathematical package), and interfaces with other application systems (e.g., the linkage between an accounts receivable system and a general ledger system).*

### 3.1.3. Communications interfaces

*This should specify the various interfaces to communications such as local network protocols, etc.*

## Functional requirements

### Web

#### 1.1 AI Chatbot for Consultations and Scheduling

##### Use Case Diagram

##### The AI chatbot will converse with the patient, getting their general information (name, symptoms, consultation type) so it can find a suitable doctor in the database it’s connected to. Once found, it will first check if there is free time, then it would notify the doctor about this. If the doctor accepts, the consultation will be scheduled. If not, the doctor or AI chatbot can suggest another doctor in the same department.

##### Activity Diagram

##### Wireframe

#### 1.2 Inventory, Room and Billing Management

##### Use Case Diagram

##### Hospital technicians and staff will assign a patient to a room (basic, ICU, check-ins) and for possible uses in the department’s inventory, which is automatically tracked by usage (needles, pillows, blankets and towels, dextrose, medicine, etc.), and would then be automatically summed up for their total bill, including possible discounts for PWD or Senior Citizens.

##### Activity Diagram

##### Wireframe

#### 1.3 Dashboard

##### Use Case Diagram

##### Doctors, Patients, and Admins will be able to access their dashboards that contain relevant information, which varies based on their user privileges. For users, they will be able to see their medical records, billing information, past consultations, and future schedules. For doctors, they will be able to see the assigned patients’ medical records, blah blah blah. For admins, they will be able to see relevant information about the inventory, rooms, and billing. Only admins can create accounts for both doctors and patients.

##### Activity Diagram

##### Wireframe

### Mobile

#### 2.1 Patient Data and Record Management

##### Use Case Diagram

##### This function will grant the doctor access to the patient’s data and records when signing up for the consultation of the doctor.

##### Activity Diagram

##### Wireframe

#### 2.2 Medical Second-Opinion Telemedicine Platform

##### Use Case Diagram

##### This function enables patients to seek a second opinion on their medical condition from other qualified doctors. This feature facilitates remote consultations, allowing patients to share their symptoms and receive professional feedback from other healthcare experts, without the need for in-person visits.

##### Activity Diagram

##### Wireframe

#### 2.3 Prescription Information and Order-Management

##### Use Case Diagram

##### This function will allow the doctors to input the prescription details based on the symptoms presented by the patient. After assessing the patient’s condition, the doctor will provide the necessary prescription information, which includes medication names, dosages, and usage instructions. This prescription is directly linked to the patient’s health records to ensure accuracy and personalized care. Once the prescription is done created by doctor, the prescribed medications can be ordered through the system. The patient or authorized healthcare staff can place an order with the linked pharmacy based on the prescription.

##### Activity Diagram

##### Wireframe

## Non-functional requirements

### Performance

##### Details

### Security

##### Details

### Reliability

##### Details