**Aim:**

Prepare a Software Requirement Specification (SRS) document for a given problem statement.

**Problem Statement:**

The purpose of the Hospital Management system is to allow for storing details of a large number of patients, different types of doctors like surgeons, specialist...etc. and allow for search of medicines and different treatments, bill details facilities separately to administrator/hospital’s owner, staff and patients. Different privileges are given to different types of users.

**Objective:**

* To understand different sections of Software Requirement Specification (SRS).
* To understand functional requirements of the system
* To understand performance requirements of the system
* To apply design constraints and appropriate validation on the system

**Theory:**

Software Requirement Specification (SRS) document usually contains a software vendor’s understanding of a customer’s software requirements. This document ensures that the software vendor and the customer are in agreement as to the features required in the software system being built. SRS is created after the initial requirement elicitation phase in which software vendor interacts with the customer to understand the software needs. Usually SRS documentation is prepared by a business analyst who has some technical background.

**1. Introduction**

**1.1 Purpose**

The main objective of this document is to illustrate the requirements of the Hospital Management System. The document gives the detailed description of the both functional and non-functional requirements proposed by the client. The purpose of this project is to provide a friendly environment to maintain the details of facilities available and hospital members. The main purpose of this project is to maintain easy facilities circulation system using computers and to provide different reports. This project describes the hardware and software interface requirements using DFD, ER diagrams and UML diagrams.

**1.2 Scope of Development Project**

Hospital Management System is basically updating the manual Hospital system into an internet-based application so that the users can know the details of their accounts, availability of books and maximum limit for borrowing.

The project is specifically designed for the use of patients and hospital staff. The product will work as a complete user interface for hospital management process and hospital usage from ordinary users. Hospital Management System can be used by any existing or new hospital to manage its availability of medicines, operation progress, reports of patients and payment of different treatment and monitoring. It is especially useful for any medical institute where modifications in the content can be done easily according to requirements.

The project can be easily implemented under various situations. We can add new features as and when we require, making reusability possible as there is flexibility in all the modules.

**1.3 Definitions, Acronyms and Abbreviations (Put it in table format)**

HMS – Hospital management system

SRS – Software requirements specification

PC – Personal Computer

HDD - Hard Disc Drive

RAM – Random Access Memory

IE – Microsoft Internet Explorer

JAVA -> platform independence

SQL -> Structured query Language

ER -> Entity Relationship

UML -> Unified Modeling Language

IDE -> Development Environment Integrated

SRS -> Software Requirement Specification

IEEE -> Institute of Electrical and Electronics Engineers

FOSS -> Free and Open Source Software

GUI -> Graphical User Interface

HIS -> Hospital Information System

RIS -> Radiology Information System

IAM -> Image Acquisition Module

PACS -> Picture Archiving and Communication System

ADT -> Abstract Data Type

**1.4 List of Stakeholders**

* Staff, patient, Doctor
* Project Manager
* Requirements Engineer
* System Administrator

**1.5 References**

* Books
* Software Requirements and Specifications: A Lexicon of Practice, Principles and Prejudices (ACM Press) by Michael Jackson.
* Software Requirements (Microsoft) Second Edition by Karl E. Wiegers.
* Software Engineering: A Practitioner’s Approach Fifth Edition By Roger S. Pressman.
* Websites
* https://en.wikipedia.org/wiki/Software\_requirements\_specification

**2. Overall Descriptions**

**2.1 Product Perspective**

The HMS is an open source system comprising of five different subsystems. The five subsystems are as follow: The FOSS RIS project is a separate program, which is a component of a larger FOSS Hospital Management System (HMS), similar to how Microsoft Word is a separate program inside Microsoft Office suite. In the FOSS HMS system, the FOSS RIS program performs all HIS operations. The RIS module uses the shared, global variables, enums, framework, and used to create the other FOSS HIS program components, just like with Microsoft Office. All data exclusive to the RIS module will be programmed in the RIS module. Hospital Information System will replace all traditional and outdated means of tracking patient information and other data useful to the hospital. A Hospital Information System shall replace forms of databases using manual or outdated hardcopy databases. Accessing data can be better monitored, organized, and time conscientious. The IAM program shall be a new management system which shall make individual systems obsolete. It shall allow one program to control all the different image acquisition devices and shall interact with the other components of the hospital management system being designed. The driving principle of this PACS is to automate and provide the infrastructure to digitally control the storage and transportation of images taken with compatible devices within a general hospital. The ADT/PRS subsystem stores patient data, which other subsystems can access as required. This is accomplished by granting the other systems access to this subsystem’s patient database.

**2.2 Product Function**

The HMS has five subsystems and these subsystems shall perform the following features:

• The RIS subsystem shall include patient list management, radiology department workflow management, request and document x-ray scanning, result entry, and reporting and printout/faxing and emailing of clinical reports.

• The PACS subsystem shall perform image importing/capturing, image encryption, local image storage, remote image retrieval, image compression, image display, and image processing.

• The HIS subsystem shall contain a secure database.

The database GUI shall be user friendly for all staff members and properly enter/obtain/modify patient information.

The DB shall utilize the token authentication for secure access and will be relative in size and flexibility of the data demand.

• The IAM subsystem shall have a simple user interface which allows the user to log in then access any imaging device connected to the imaging intranet, select what type of device and then which specific device within the hospital they will use. The images shall be controlled from one console and share these images with the hospital patient database.

• The ADT/PRS subsystem shall allow an administrator to enter patient information, such as name, age, etc. That information is then stored, and shared with other users as appropriate. It shall also alert the medical staff when a patient that requires different treatment is admitted, such as some with an infectious disease.

**2.3 Operating Environment**

The FOSS HMS program runs on Windows 7, for 32-bit/x86 and 64-bit/x64 PC architectures. The software for the RIS subsystem will be written in C#, using Microsoft Visual Studio 2010. The program will be GUI-based (like with most modern Windows software). The HIS subsystem will run off a Cloud-Based Platform. The Cloud-based server will utilize Oracle or SQL database running on the cloud. The operating system shall be a MS-Windows or UNIX. Integration to the server shall be done via a HTTPS, SFTP, or VPN to create, update, fetch, or delete data.

**2.4 Development Environment**

**Software Configuration: -**

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database.

Operating System: Windows NT, windows 98, Windows XP

Language: Java Runtime Environment, Net beans 7.0.1 (front end)

Database: MS SQL Server (back end)

**Hardware Configuration: -**

Processor: Pentium(R) Dual-core CPU

Hard Disk: 40GB

RAM: 256 MB or more

**2.5 Data Requirement**

**User details, book details,**

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their patient accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting data and putting into account. Keeping an accurate database of all patients, their data, history, records and all items is necessary as it would improve the overall of productivity of the hospital.

**3. External Interface Requirement**

The purpose of this section is to identify and document interfaces and interaction of the software with external entities in detail.

The software provides good graphical interface for the user and the administrator can operate on the system, performing the required task such as create, update, viewing the details of the book. The system should be able to interact with the user management module and external hardware devices like printer, barcode reader etc.

**3.1 Functional requirements**

Functional requirements are the following:

1. The LMS should store all information about book details, librarians and members, credentials
2. The LMS allow searching items by author, title or keywords
3. The LMS should allow librarians to add, delete and modify items in database, and check availability of the items.
4. The LMS should automatically calculates fine based on overdue.
5. The LMS should allow members to view or update their personal information
6. The LMS should provide to search, request and renew items either from the library computers
7. The LMS should allow members to view their account information
8. The LMS barcode generation module helps the librarian/user to create bar codes for books available with the library according to the book ID
9. The LMS should generate request’s reports for librarians every day, on base of which librarians could make decisions about acquiring or retirement the item(Stock Check)

**4. System Features**

The users of the system should be provided the surety that their account is secure. This is possible by providing:-

* User authentication and validation of members using their unique member ID
* Proper monitoring by the administrator which includes updating account status, showing a popup if the member attempts to issue number of books that exceed the limit provided by the library policy, assigning fine to members who skip the date of return
* Proper accountability which includes not allowing a member to see other member’s account. Only administrator will see and manage all member accounts

**5. Other Non-functional Requirements**

**5.1 Performance Requirement**

The proposed system that we are going to develop will be used as the Chief performance system within the different campuses of the university which interacts with the university staff and students. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the university.

* The performance of the system should be fast and accurate
* Library Management System shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify invalid username/password
* The response should be fast enough to avoid users’ response collisions

**5.2 Safety and Security Requirement**

* System will use secured database
* Normal users can just read information but they cannot edit or modify anything except their personal and some other information.
* System will have different types of users and every user has access constraints
* Proper user authentication should be provided

**Conclusion:**

In this assignment the major sections of a typical Software Requirement Specification (SRS) document is studied.