Software Specification Requirement

1. **PROJECT NAME**: Hospital Management System

2. **PROJECT IDEA**: Our Cardiac Hospital Management System is a full-web-based solution designed to improve cardiac hospital management and organization. Our system aims to reduce the time and effort required of hospital staff and reduce death rates while increasing capacity and patient management. The web-based approach, compared to traditional file-based systems, ensures data security and reduces the risk of physical damage, loss, or error. Patients can book appointments with their chosen doctors more easily by enabling appointments online through the hospital's website. Furthermore, doctors can use the system to efficiently manage their patient lists, allowing for more effective patient care and services. Furthermore, the system assists administrators in allocating beds to patients on a priority basis, with doctor approval. The system is used by administrators to keep track of available beds. Our system includes:
   ● Non-functional requirements can be used to improve the functioning of the computer system, but not the management of the hospital.
   ● Functional requirements, on the other hand, are requirements directly related to the hospital management. The primary areas of concern are performance, security and user interface.

3. PROJECT DRIVERS:

a. The goal of the project

The Cardiac Hospital Management System aims to streamline the operations of cardiac hospitals by automating tasks such as patient registration, appointment scheduling, bed allocation, and medical record management. The system enhances efficiency, reduces human error, and ensures better patient outcomes by enabling secure and seamless access to data for administrators, doctors, and patients.

b. Client, Customer, Stakeholders

Client: Cardiac healthcare providers seeking to optimize hospital management processes.

Customer: Cardiac hospitals, represented by their management teams and IT departments.

Stakeholders:

Hospital administrators (require efficient management tools).

Doctors (need easy access to patient records and schedules).

Patients (seek convenience for managing appointments and records).

IT and database managers (responsible for maintaining the system).

c. Users of the project

Patients: Individuals who book appointments, view medical records, and interact with doctors via the system.

Doctors: Healthcare professionals who access and update patient records, respond to queries, and manage their schedules.

Administrators: Oversee the entire system, manage resources (e.g., beds), and ensure data security and integrity.

4. PROJECT CONSTRAINTS:

a. Mandated Constraints

This specifies constraints on the way that the problem must be solved. Describe the mandated technology or solution. You should also explain the reason for using technology. The constraints are treated as a type of requirement.

Regulatory Compliance: The system must comply with HIPAA and GDPR regulations to ensure data privacy and security.

Platform Compatibility: The system must be accessible via web browsers and mobile devices.

Timeline: The system must be delivered by the specified deadline.

Security: Password-protected access and role-based permissions are mandatory to ensure data integrity.

b. Naming Conventions and Definitions

Names are very important. They invoke meanings that, if carefully defined, can save hours of explanations. Attention to names at this stage of the project helps to highlight misunderstandings. The glossary produced during requirements is used and extended throughout the project.

HIS: Hospital Information System.

QR Code: A machine-readable code that provides direct access to patient profiles.

Admin: A high-level user with full system control and oversight.

EHR: Electronic Health Records system integrated with the platform.

c. Relevant Facts and Assumptions

The hospital will provide the necessary hardware infrastructure (e.g., servers, networking).

All users (patients, doctors, administrators) must have basic computer literacy.

The system assumes real-time updates of patient records and availability status of resources (beds, doctors).

The hospital's IT department will handle system maintenance after deployment.

5. FUNCTIONAL REQUIREMENTS:

a. The Scope of the Work

The Cardiac Hospital Management System replaces manual, file-based systems with an automated web-based solution. It addresses the following areas:

- Patient registration and login.
- Appointment booking and scheduling with doctors.
- Allocation of hospital beds based on medical priority.
- Management of patient medical records, accessible via QR code.

The system aims to:

- Reduce administrative workload.
- Enhance patient safety and treatment outcomes.
- Ensure data security and regulatory compliance.

b. The Scope of the Product

A use case diagram identifies the boundaries between the users (actors) and the product you are about to build (this is sometimes called “the system”). You arrive at the product boundary by inspecting each business use case and determining, in conjunction with the appropriate stakeholders, which part of the business use case should be automated (or satisfied by some sort of product) and what part should be done by the user. This task must consider the abilities of the actors, the constraints, the goals of the project, and your knowledge of both the work and the technology that can make the best contribution to the work.

The product is a web-based application with the following functions:

- Patient Management:
- Registration, login, and profile updates.
- Access to medical records using QR codes.
- Appointment Management:
- Scheduling and rescheduling appointments.
- Doctor-patient communication via forums.
- Administrative Tools:
- Real-time bed allocation tracking.
- Approval of patient-doctor appointments.
- Oversight of all patient and doctor records.
- Doctor Tools:
- Viewing patient records and histories.
- Updating patient statuses post-treatment.
- Responding to patient queries in forums.

c. Atomic Functional Requirements Shell (Volere cards)

Requirement # 1 Requirement Type: Functional Event/BUC/PUC # Event

Description: Patients should be able to register on the system, book appointments, and view their medical records.

Rationale: This allows patients to access healthcare services conveniently and manage appointments effectively.

Originator: Patient

Fit Criterion: User should be able to successfully register, book appointments, and access medical records.

Customer Satisfaction: 5 Customer Dissatisfaction: 2

Priority: High

Conflicts: may arise with system performance or data access issues

Supporting Materials: Registration and appointment booking system documentation, medical records access guidelines.

History: Added 30 November

Requirement # 2 Requirement Type: Functional Event/BUC/PUC # Event

Description: Doctors should be able to view and update patient medical records.

Rationale: To ensure healthcare professionals have accurate, up-to-date patient information for providing the best care.

Originator: Doctor

Fit Criterion: Doctor must be able to view and edit patient medical records securely and efficiently.

Customer Satisfaction: 5 Customer Dissatisfaction: 2

Priority: High

Conflicts: Data privacy issues or conflicting medical data.

Supporting Materials: Patient record management software documentation.

History: Added 30 November

Requirement # 3 Requirement Type: Functional Event/BUC/PUC # Event

Description: Both patients and doctors should be able to send and receive messages within the system.

Rationale: Enables continuous communication between doctors and patients for queries, follow-ups, and medical advice.

Originator: Patient, Doctor

Fit Criterion: The messaging system must be secure, easy to use, and reliable for both doctors and patients.

Customer Satisfaction: 5 Customer Dissatisfaction: 2

Priority: Medium Conflicts: Potential issues with message delays or privacy violations.

Supporting Materials: Communication system documentation, message encryption standards.

History: Added 30 November

Requirement # 4 Requirement Type: Functional Event/BUC/PUC # Event

Description: State administrators should be able to view and assign hospital beds based on patient priority.

Rationale: Ensures that critical patients are assigned appropriate hospital beds according to urgency.

Originator: State Administrator

Fit Criterion: Bed assignment should be efficient, prioritized based on patient condition, and real-time.

Customer Satisfaction: 5 Customer Dissatisfaction: 2

Priority: High Conflicts: Conflicts over bed availability or miscommunication

Supporting Materials: Bed management system documentation, hospital bed usage data.

History: Added 30 November

Requirement # 5 Requirement Type: Functional Event/BUC/PUC # Event

Description: Temporal administrators should be able to manage doctor schedules and monitor appointments.

Rationale: Ensures efficient scheduling of appointments and optimized doctor availability.

Originator: Temporal Administrator

Fit Criterion: The scheduling system must allow administrators to view, edit, and optimize doctor schedules.

Customer Satisfaction: 5 Customer Dissatisfaction: 2

Priority: High

Conflicts: Scheduling conflicts between doctors or patients.

Supporting Materials: Scheduling software documentation, user interface designs.

History: Added 30 November

Volere

Requirement # 6 Requirement Type: Non-Functional Event/BUC/PUC # Event

Description: System administrators should be able to assign roles and manage user permissions.

Rationale: Ensures that the right users have the appropriate access to system features and data.

Originator: System Admin

Fit Criterion: Roles and permissions should be assignable securely, and users must only access authorized features.

Customer Satisfaction: 5 Customer Dissatisfaction: 2

Priority: High

Conflicts: Potential conflicts in permissions or system errors in access control.

Supporting Materials: Role management system documentation, security protocols.

History: Added 30 November

Requirement #	7	**Requirement Type:** Non-Functional		**Event/BUC/PUC #**	Event

Description: System administrators should be able to monitor system logs and security.

Rationale: Critical for maintaining system health, ensuring security, and identifying issues proactively.

Originator: System Admin

Fit Criterion: System monitoring tools should capture logs and alerts in real-time.

Customer Satisfaction:	5	**Customer Dissatisfaction:**	2
Priority:	High	**Conflicts:**	Conflicts related to data breaches or system failures.
Supporting Materials:	System monitoring tools, security documentation.		
History:	Added 30 November		