HOSPITAL MANAGEMENT SYSTEM

PROBLEM STATEMENT

IDEAL SITUATION:

The project Hospital Management system includes registration of patients, storing their details into the system, and also computerized billing in the pharmacy, and labs. The software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. It includes a search facility to know the current status of each room. User can search availability of a doctor and Doctors can view the details of a patient using the id.

REALITY:

At present or traditional Hospital Management system in any hospital is time consuming and lengthy process. Patients goes to hospital and they have to go through various process for treatment. After this patient goes to billing counter and patients have to wait there in queue where the billing is generated manually by receptionist, this process is more time consuming and lengthy. To overcome such drawbacks, we design a Hospital Management System.

CONCLUSION:

By implementing this web-based application, the website and customized application on the tablet. The management of the patients will be very much easier, efficient and less time consuming. It will be easy for the doctors and patient to access the records and reports as the history and reports are already present in the system ,so the patient will not have to carry all the reports and big x-rays and MRI films etc. The patient details are already present in the database while registration so there is no need to fill a form during emergency cases. The doctors can check details of the patients on their system, provide prescription on a click which will be sent to the pharmacist this will reduce a huge amount of time as the pharmacist knows which medicines to be kept ready before hand, the communications among the doctor and patient are enhanced as the patient can get as much help online. It will help to reduce many manual efforts, time taken and cost.

OBJECTIVES:

- Recording information about the Patients that come.
- Recording information related to diagnosis given to patients.
- Keeping track of the appointment dates
- Keeping record of the immunization provided to children/patients.
- Keeping information about various diseases and medicines available to cure the patients.
- Tracking the bill payments.

DEVELOPER'S OBJECTIVES:

- Create Database to store patient's personal & appointment details, doctor's personal & schedule details, availability of rooms & facilities.
- Provide permissions to administrative employees for Creating patient's profile, Update availability of rooms & facilities, Update schedules of doctors.
- Provide access to doctors for viewing and modifying reports and prescriptions.
- Provide access to patient & pharmacist for viewing reports, prescriptions, diet details, appointment dates etc.
- To create login interfaces for the above-mentioned entities and authenticate them accordingly.

HOSPITAL MANAGEMENT SYSTEM

SOFTWARE REQUIREMENT SPECIFICATION

1. INTRODUCTION

The SRS is produced at the culmination of the analysis task. The function and performance allocated to software as part of the system engineering and refined by establishing a complete information description, a detailed functional description, a representation of system behaviour, indication of performance requirements and design constrains, appropriate validation criteria and the other information related to requirements.

The SRS is technical specification of requirement of Hospital Management system. This specification describes what the proposed system should do without describing how it will do it. It also describes complete external behaviour of proposed system.

1.1. Purpose

The main purpose of our system is to make hospital task easy and is to develop software that replaces the manual hospital system into automated hospital management system. This document serves as the unambiguous guide for the developers of this software system.

1.2. Scope

The document only covers the requirement specification for the hospital management system. This document does not provide any references to the other component of the hospital management system. All the external interfaces and the dependencies are also identified in this document.

1.3. Definition, Acronyms, Abbreviations

IDE: - Integrated Development Environment
Java:-Platform Independent, Object-oriented programming language
SQL: - Structured Query Language
SRS: - Software Requirement Specification.

1.4. References

1. https://www.google.com
2. https://www.medesk.com
3. https://www.docpulse.com
4. https://www.apollohospitals.com/apollo_pdf/ahel-risk-management-policy.pdf

1.5. Overview

- Hospital Management System is a process of implementing all the activities of the hospital in a computerized automated way to fasten the performance.
- This project is to maintain the patient details, lab reports and to calculate the bill of the patient. You can also manually edit any patient details and issue bill receipt to patient within few seconds.

2. OVERALL DESCRIPTION

2.1. Product perspectives

This project gives the procedural approach how a patient gets treatment, details about date of treatment and finally depending on different criteria like room allocated, lab reports, treatment and medicine taken…. etc., how billing is calculated. During billing health care facility is also considered.

2.2. Product Function

The data represented in hospital management application will perform the following major function

- Patient Details: - It includes inpatient and outpatient details.
- Lab reports
- Billing Details

This software will help to calculate the bill much quicker and simpler way. This enables the organization to keep the information in efficient and systematic way.

2.3. User Characteristics

This software is developed such that total appearance of the product to make it more user friendly. The operator will be provided with login Id and password. General users with basic computer skills can use this software.

2.4. Constraints

Any update regarding the patient's information from the hospital are to be recorded to have updated and correct values.

2.5. Assumption and Dependencies

All the data entered will be correct and up to date. This software package is developed using java as front end which is supported by sun micro system, MS SQL server 2022 as the back end which is supported by Microsoft windows 10 or 11.

3. EXTERNAL INTERFACE REQUIREMENTS

3.1. User Interface

User interface is designed in a user-friendly manner and the user, in another end he has to give the order, for that he will interface with keyboard and mouse.

3.2. Hardware Interface

Hard disk – 2 TB
RAM – 16 GB
Keyboard – Standard QWERTY keyboard for interface
Mouse – Standard mouse with 2 buttons
Database Server along with backups

3.3. Software Interface

OS – Windows 10 or 11
Front end – Java language
IDE – Apache Net Beans 13
Back end – MS SQL Server 2022

3.4. Communication Interface

Windows

4. SYSTEM FEATURES.

- **Appointment Management**
  Patients visiting the hospital's website can book online appointments with ease.

- **Billing Management**
  Integrated Billing with treatments, Lab and Radiology. Alerts will be sent on Discount Authorisation. Automatic due capture, Option to bill before and after consultation.

- **Prescription Management**
  Manage commonly and recently used medicines. Option to show medicines available in the pharmacy.

- **Lab Management**
  Automatic notification can be sent to customers on test results. Lab notifications are displayed on the patient interface's home page and the results are displayed under the Lab Test section.

- **Pharmacy Management**
  Comprehensive pharmacy management handles prescription integration and Stock Management.

- **Manage Multiple Locations**
  Any number of branches can be added and managed using a single account.

- **Easy Patient data retrieval**
  HMS makes it possible to access all the data related to a patient via a system by the means of a few simple clicks. Information like patient history, current illness, doctors involved, tests reports taken, billing information and many more can be made visible to the user. These data will help to connect the dots about the patient, like specific diagnosis, related treatment, and medication.

- **Increased Data security**
  The patient data can be kept a hundred percent safe by using HMS in your hospital. It can be made accessible by only a limited amount of authorized personnel. With HMS, all the data is stored on a server or cloud and kept safe by just securing the login information safe.

- **Ease to Access System Facilities**
  Hospital Management System makes it easy to get access to the management system facilities for the authorized users and keep it safe from unauthorized users.

- **Cost Effective**
  HMS not only saves time in the hospital but also is cost-effective in decreasing the number of people working on the system of manual entry of data and paperwork. The implementation of this will decrease the human intervention into the system thereby avoiding human-caused errors.

5. OTHER NON-FUNCTIONAL REQUIREMENTS

5.1 Performance Requirements

The performance features which are required in the proposed system are

User Friendliness

The proposed system should be user friendly, understandable and easy to use. It should provide on line help and error messages for user ease. User should be able to take the output of reports on the screen.

Requirements

- This software should not breakdown suddenly in any disaster like power failure. The timeline of this software must be in our mind.
- The performance of the functions and every module must be well.
- At every step the output of the one phase is the input of the other phase and it will be reliable and accurate.
- The risk factor must be taken at initial step for better performance of the software.
- For individual function the performance will be well.
- For login to the software password and user name will be matched to the password and name saved in the database and thus only authenticated users are allowed to the login.
- There will be various ways of retrieving data and it takes less time.

5.1.1 Capacity

The System must support 1000 people at a time.

5.1.2 Dynamic requirements

The system shall give responses in 1 second after checking the patient's information.

5.1.3 Quality

The Quality of the system is maintained in such a way that it can be very user-friendly. The software quality attributes are assumed as under:

- Accurate and hence reliable.
- Secured.
- Fast Speed.
- Compatibility.

5.2 Software System Attributes

5.2.1 Reliability

The HMS ensures reliable storage of data on a server or in the cloud with backups and firewalls to avoid data breaches. The data can be stored completely safely in the hospital and can only be made available to a small number of authorized individuals.

5.2.2 Availability

The System works even at the times of power outage with the help of UPS or Generators as power backups.

5.2.3 Security

Each member is required to enter an individual Username & password when accessing the software. Administrators have the option of increasing the level of password security their members must use. The data in the database is secured through multiple layers of Protection. One of those security layers involves member passwords. For maximum Security of your software, each member must protect their password.

5.2.4 Maintainability

Back Up - The system shall provide the capability to back-up the Data
Errors - The system shall keep a log of all the errors.

5.3 Business Rules

The business rules of the software are as follows:

- The Administrator has the authority to Create Database to store patient's personal & appointment details, doctor's personal & schedule details, availability of rooms & facilities.
- The staff at the hospital perform the following:
  - Making the entries in the system regarding all the details of the patient.
  - Keeping the details of the patient updated as soon as changes are encountered so that the data is in consistent state.
  - Blocking or seizing of the account of user on discovery of any illegal access.
  - Update schedules of doctors as soon as changes are encountered.

6. OTHER REQUIREMENTS

None.

HOSPITAL MANAGEMENT SYSTEM

SOFTWARE CONFIGURATION MANAGEMENT PLAN

In Software Engineering, Software Configuration Management is a process to systematically manage, organize, and control the changes in the documents, codes, and other entities during the Software Development Life Cycle. The primary goal is to increase productivity with minimal mistakes.

Change in Requirements:

Software change happens for different reasons, for example, in order to fix faults, to add new features, or to restructure the software to accommodate future changes. During product development and evolution, the set of requirements may be changed; where modifications to existing requirements or additions of new requirements may affect existing requirements. Some requirements changes may impact the success of the product within established schedules.

- Changes related to fixing faults and adding new features can be handled by the developer without any major changes to user experience.
- Changes involving major developments such as restructuring should be discussed well with all the business stakeholders and the office associates need to be educated about the changes in the interface.

Change in Team/Organization:

Changes in the team members gives rise to multiple inconveniences and conflicts.

- When new members are added into the team, they need to be educated about the whole project and its progress and they should catch up with the existing team members.
- When the team members are changed, the coordination between the team gets disturbed and may directly affect the work process.

Change in Government Policies & Rules:

Change in government policies such as terms and conditions related to data security, subsidies and tax related conditions may impact a software in various levels .
Once if the software is created it is difficult to implement changes into the system.
If there are any elements which violate the new government policies, it is essential to evaluate and modify those elements of the software.

Change in Project Schedule:

If there are any changes in the project schedule causing the schedule to be shortened, it will severely affect the development process and the work fashion of the team.
It will require the team to work faster and the reviews should also be organised in such a way that the product will be delivered on time.

Tasks in SCM Process:

1. **Configuration Identification**
   The purpose of configuration identification is to maintain control of an evolving system by Uniquely identifying the system, revisions of the system and the component parts of each revision and understanding the status of configuration items as they progress through the development process.

   In HMS, the identified configuration items are Appointment Management, Test Report Management, Pharmacy Management and Resource Management. These configuration items and their components are assigned with unique identifiers and are tracked for their status as they progress through the development process.

2. **Baseline**
   A baseline is a reference point in the software development life cycle marked by the completion and formal approval of a set of predefined work products. The objective of a baseline is to reduce a project's vulnerability to uncontrolled change by fixing and formally change controlling various configuration items at critical points in the development life cycle.

   The baselines are created based upon the progress of development of the configuration items as functional, preliminary design, detailed design, product and operational baselines.

3. **Change Control**
   Change control is a procedural method which ensures quality and consistency when changes are made in the configuration object. Its purpose is to ensure that all changes to a complex system are performed with the knowledge and consent of management.

   Configuration control tasks include initiating, preparing, analysing, evaluating and authorising proposals for change to a system. Configuration control has four main processes:

   1. Identification and documentation of the need for a change in a change request
   2. Analysis and evaluation of a change request and production of a change proposal
   3. Approval or disapproval of a change proposal
   4. Verification, implementation and release of a change.

4. **Configuration Status Accounting**
   Configuration Status Accounting records and reports the status of configuration items, proposed changes and the implementation status of the approved changes. The activities under status accounting are

   1. Identify list of configuration items along with the complete log of changes.
   2. Complete listing of dates when each version of each configuration item was baselined.
   3. Tracking progress based on baselines for previous releases/versions to be extracted for testing.

5. **Configuration Audits and Reviews**
   Configuration Audits & Reviews are held for a software product to check the integrity of the product prior to delivery. Audits are conducted by Auditors to check whether the defined processes are being followed and ensuring that the SCM goals are satisfied.

HOSPITAL MANAGEMENT SYSTEM

SOFTWARE RISK MANAGEMENT PLAN

Software Development: Risk Management

Most software engineering projects are risky because of the range of serious potential problems that can arise. The primary benefit of risk management is to contain and mitigate threats to project success. You have to identify and plan, and then be ready to act when a risk arises—drawing upon the experience and knowledge of the entire team to minimize the impact to the project.
Software Risk management includes the identification and classification of technical, programmatic and process risks, which become part of a plan that links each to a mitigation strategy. The project manager monitors risk during the project. If any materialize, a specific owner implements a mitigating action. In this article, we explain the elements of an effective software risk management plan and provide examples of plan elements.

Software Risk Management Plan

After cataloging risks according to type (technical, project, process, organizational), the software development project manager crafts a plan to record and monitor these risks. As part of a larger, comprehensive project plan, the risk management plan outlines the response that will be taken for each risk—if it materializes. The core of the risk management plan is the risk register, which describes and highlights the most likely threats to a software project.

Software Development Risk Register

To ensure that risks remain in the forefront of project management activities, it's best to keep the risk management plan as simple as possible. For both conventional and agile software project management methodologies, a risk register is a proven tool for organizing and referring to known projects risks. A comprehensive risk register would contain the following attributes:

- **Description of risk** — Summary description of the risk—easy to understand.
- **Recognition Date** — Date on which stakeholders identify and acknowledge the risk.
- **Probability of occurrence** — Estimate of probability that this risk will materialize (%).
- **Severity** —The intensity of undesirable impact to the project—if the risk materializes.
- **Owner** —This person monitors the risk and takes action if necessary.
- **Action** —The contingent response if the risk materializes.
- **Status** — current team view of the risk: potential, monitoring, occurring, or eliminated.
- **Loss Size** —Given in hours or days, this is measure of the negative impact to the project.
- **Risk Exposure** —Given in hours or days, this is a product of probability and loss size.

Software Development Risk Register:

PROJECT NAME	Hospital Management System
CLIENT	X Hospitals
POINT OF CONTACT	Administrator
VERSION NUMBER	1.0
DATE PREPARED	
AUTHOR	Manager
BEGIN DATE	
END DATE	
DURATION	

Software Development Risk Management Table:

Impact Levels	HIGH	MEDIUM	LOW	
Response Status	RISK OPEN		RISK CLOSED	
Plan Status	NO PLAN	PLAN IDLE	PLAN IMPLEMENTED	PLAN FAILED

RISK OVERVIEW				
RISK ID	**DESCRIPTION**	**OWNER**	**REPORT DATE**	**LAST UPDATED**
R001	Data Breach	Admin, Developer		
R002	Loss of Data Storage Devices	Admin		
R003	Data Communication	Admin		
R004	Billing & Payments	Accountant		
R005	Data Corruption	Admin		
R006	Pharmaceutical Errors	Developer		

RISK ASSESSMENT				
RISK ID	**IMPACT LEVEL**	**IMPACT DESCRIPTION**	**LIKELIHOOD OF OCCURRENCE**	**PLAN STATUS**
R001	HIGH	Devices running on older versions of OS and not having security updates may open a door for data breaches.	HIGH	PLAN IN PROGRESS
R002	HIGH	It might be possible that the data stored might get lost due to damage of hard disk.	MEDIUM	PLAN IDLE
R003	HIGH	Communication gap between the developing members of the project.	MEDIUM	PLAN IDLE
R004	MEDIUM	Negligence of billers may lead to errors in payments.	LOW	PLAN IN PROGRESS
R005	MEDIUM	Power Outages or other power related problems may lead to Data Corruption.	MEDIUM	PLAN IMPLEMENTED
R006	HIGH	Errors in delivery of drugs to patients by pharmacist.	LOW	PLAN IMPLEMENTED

RISK RESPONSE				
RISK ID	**PLANNED ACTION**	**COMPLETED ACTION**	**RESPONSE STATUS**	**DATE CLOSED**
R001	Make sure all devices are running the latest versions of OS and also use secure websites. (and) Prevent Hacking & Breach of Data by using End-to-End Encryption and Firewalls.	YES	RISK OPEN	
R002	Work done should have at least two backups with frequent checking.	NO	RISK OPEN	
R003	Frequent meetings should be organized for better understanding.	NO	RISK OPEN	
R004	Validation of Billing and Payments should be done on regular basis to prevent Huge Losses.	NO	RISK OPEN	
R005	Use Uninterruptible Power Supply (UPS) units for devices and servers.	YES	RISK CLOSED	
R006	Add verification abilities to software at the pharmacy to verify the drugs delivered with doctor prescriptions.	YES	RISK CLOSED	