**HOSPITAL MANAGEMENT SYSTEM**

**Software Requirement Specification**

Information Technology Project 2021

**Internal Supervisor**

Muhammad Fahad

**Submitted to**

  Fahad Maqbool

**Submitted by** 

Ramisha Tahir (BSIF18E001)

Maryam Pervez (BSITF18E027)

Natasha Hassan (BSITF18E037)

Usama Shafique (BSITF18E032)

**Date of submission**

15-11-2021

****

**Table of Contents**

[1. Introduction 4](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990518)

[1.1 Purpose of Document 4](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990519)

[1.2 Project Overview 4](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990520)

[1.3 Scope 4](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990521)

[2. Overall System Description 5](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990522)

[2.1 User characteristics 5](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990523)

[2.2 Operating environment 5](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990524)

[2.3 System constraints 6](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990525)

[3. External Interface Requirements 6](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990526)

[3.1 Hardware Interfaces 6](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990527)

[3.2 Software Interfaces 7](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990528)

[4. Use Case Diagram 8](file:///F:\7th%20Semester\Capstone\Software%20Requirements%20Specifications%20(Template).doc#_Toc492990526)

4.1 USE CASE Description………………………………………………………………………………………………………….8

5. FUNCTIONAL REQUIREMENTS

6. NON-FUNCTIONAL REQUIREMENTS

6.1 User Manuals

6.2 Online Help

6.3 Tutorials

**Introduction**

* **Purpose**:

Over the years Medical care relies on record-keeping without accurate, comprehensive, up to date and accessible patient notes, medical personnel may not be able to offer the best treatment. He may misdiagnose the condition which can have serious consequences.SO, we are going to make a “Hospital Management system” for Associated records such as test reports, specimens, drug records, and patient records must also be well cared for if the patients are to be well protected, record management is a systematic and ongoing procedure by which the records of an organization are created, maintained and disposed of, this system also ensures records, preservations for evidential purposes accurate and efficient updating, timely availability and control of access only by authorized personnel.

* **overview**

The hospital management system is software that is used for the automation of hospital management in the context of Pakistan where most hospitals are not using computerized technology to manage hospital records, Hospital management system will be a perfect way to manage these records, CHMS should contain login form of doctor and patient registration, should allow patients to edit their information like patient name, address, contact number, etc. It also contains the feature of a chatbot for patients to get information about the doctor according to their ranking state.

* **SCOPE:**

Currently, the Civil hospital is using a manual system to handle the hospital process. When patients arrive, they make an appointment at the reception to consult a doctor. These are being recorded in a file. Then again, the patients diagnosed symptoms related to disease details, ward details, and other necessary details are being recorded and those files are being stored in special locations. Calculations of bills and inventory are done manually. As the current system is a file-based one, the management of the hospital has to put much effort into securing the files. They can be easily damaged by fire, insects, and natural disasters. Also, could be misplaced by losing data and information. Limited storage space of the files is another issue that they currently face when the management is manually done. There occurs an issue with the organization of data information and schedules and running the process methodically which leads to the manual system malfunctioning. If we want to check a previous record of a patient or other details. Management will be in a great problem. It’s a tough and time taking process to search for a record in a file.

We seek to create an online client-satisfied system by addressing the client's needs and improving the current manual system with client needs. Computer-based CHMS is a method of conducting checkups and keeping a record of all patients from entry slip to final doctor’s prescription and also keeping a record of the laboratory. Also, have the chatbot feature to get suggestions about the doctor relevant to patients’ disease. The functionalities of the Civil Hospital Management System are user-friendly dashboards, keeping a record of all specialist doctors, laboratory staff, and other hospital staff. Also, the civil hospital management system reduces the paperwork which has a time-consuming activity, and keeps a record of all patients and the medical store for the future also.

**Overall system description**

* **User characteristics**

**Admin:**

The administrator is the highest privileged user who can access the system, which means he can manage any activity associated with the system.

**Features:**

* managing employees
* equipment and patients
* allocating resources
* administering charges
* generating reports
* managing doctors.

**Employee:**

Provides service to customers by interacting with the systems most often.

**Features:**

* Keep track of patient details
* Keep track of test results
* Maintain bill details
* Maintain inventory
* Manage inventory records
* **Operating environment:**

Hardware Requirements:

* + - Core i5 processor
    - 4GB of RAM
    - 20GB of hard disk space at the end of the project

Software Requirements:

* + - Operating system, Windows 7 or above
    - Atom/ visual studio code
    - MySQL server
    - Visio
* **System constraint:**

1. It is wirelessly connected with encryption.
2. It is only accessible within the hospital premises.
3. The database is password protected.
4. It should use less RAM and processing power.
5. Each user should have a unique ID and password.
6. Only the administrator can change passwords.

**External Interface requirement**

**Hardware interface**

Laptop/desktop pc

* + - Core i5 processor
    - 4GB of RAM
    - 20GB of hard disk space at the end of the project

Patients ask about doctors, medicines, lab tests, etc. and the PC's

purpose is to provide them with information. To perform such actions requires a very efficient computer, otherwise, they have to wait for a long time to receive the information they need.

**Wi-Fi router**

 Wi-Fi routers are used to provide internet access inside of hospitals and for data transmission between computers and servers

**Printer**

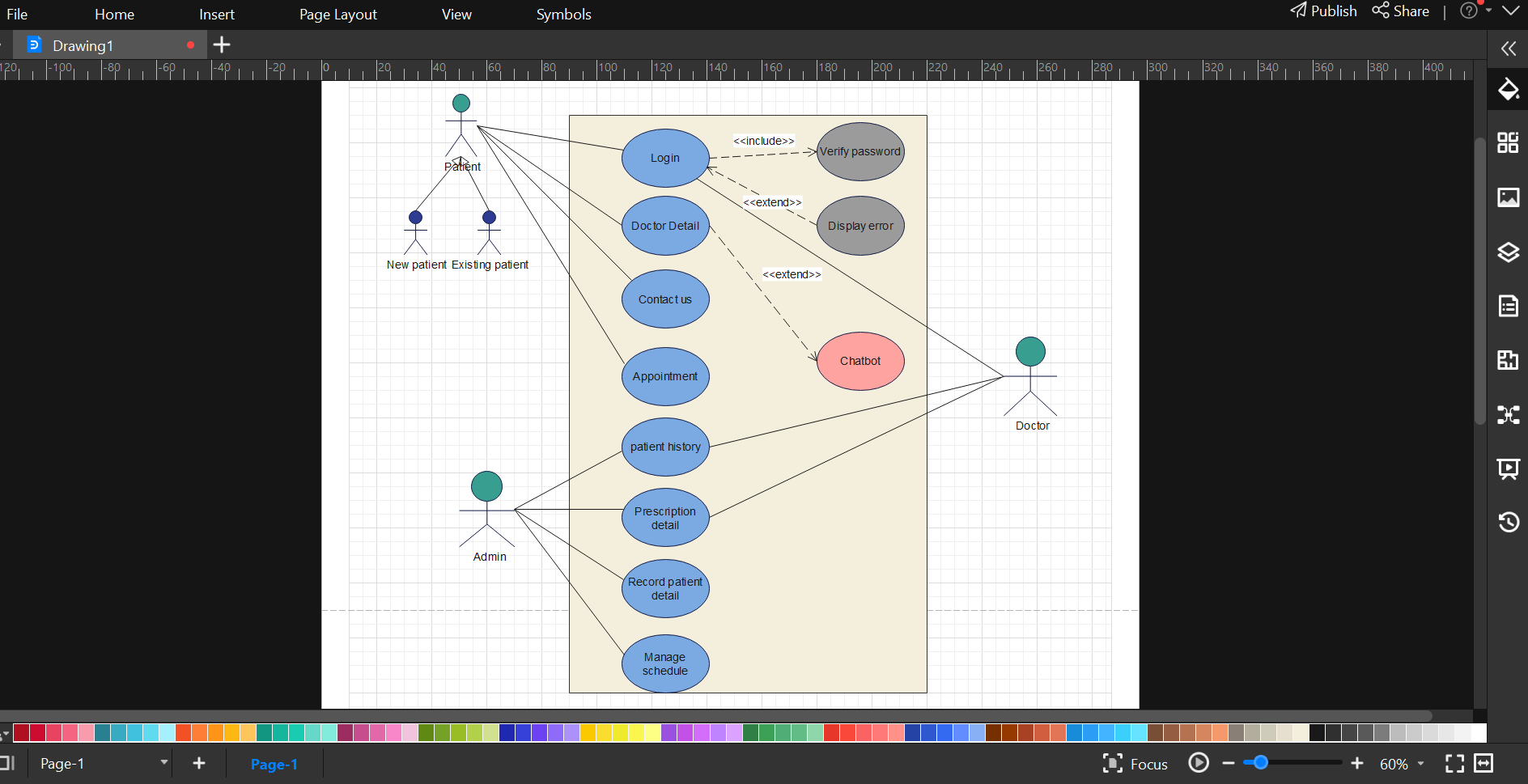
Printer use for printing bills and view reports

**Software interfaces**

* **Developing end**

* + - Atom/ visual studio code (use for developing)
    - MySQL server (database connectivity and management)
    - Visio (use for drawing diagrams)
* **Client end** 
  + - Operating system, Windows 7 or above (user-friendly and common OS)
    - MySQL server (database connectivity and management)

**USE CASE DIAGRAM**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR001: Login** | | | | |
| **Actors:**  Patients and doctors | | | | |
| **Use case Id:** | | *PR001* | | |
| **Pre-condition:** | | The system must be connected to the network | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *Enter Username* | | |  |
| **2.** | Enter Password | | |  |
| **3** | Validate Username and Password | | | Allow access to the system |
| **Alternate Scenarios:** | | | | |
| **1a:**  Invalid Username  The system shows an error message  **2a:** Invalid Password  System Shows an error message  **3a:** Invalid Password for 4 times  Application closed | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | After a successful login, the user can interact with the system | | | |
| **2.** | Users can view the doctors and medicines | | | |
| **3.** | Users can interact with doctors and can get an appointment | | | |
| **Use Case Cross referenced** | | | *Searching of doctors and medicines*  *Checkout* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR002: Doctor Detail** | | | | |
| **Actors:**  Doctor | | | | |
| **Use case Id:** | | *PR002* | | |
| **Pre-condition:** | | The actor is logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor logged in to the system. | | | Allow access to the system |
| **2.** | The actor manages the data of users | | |  |
| **3.** | The actor views the history of the user | | | Displays the patient history to the doctor |
| **Alternate Scenarios:** | | | | |
| **1a: The system** fails to authenticate the actor.  The system informs the actor and doesn’t allow the actor to proceed.  **2a:** The actor may view the wrong information. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | Data of user is displayed to doctor. | | | |
|  | Updating of system | | | |
| **Use Case Cross referenced** | | | *Log in*  *Patient history* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR004: Contact us** | | | | |
| **Actors:**  Patient | | | | |
| **Use case Id:** | | *PR004* | | |
| **Pre-condition:** | | The actor is a Patient | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor can enter system without login | | |  |
| **2.** | The system authenticates the actor | | | Allow access to the system |
| **3.** | The actor contacts to the doctor | | | Display a list of the mails sent by patients |
| **Alternate Scenarios:** | | | | |
| **1a:**  System fails to authenticate the actor  The system informs the actor and doesn’t allow the actor to proceed  **2a:** The system fails to contact  The system informs the actor | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | The System displays contact us information | | | |
| **2.** | Users can Contact us | | | |
| **Use Case Cross referenced** | | | *Patients can contact without login*  *Contact us* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR005: Appointment** | | | | |
| **Actors:**  Patient, ChatBot, and Doctor | | | | |
| **Use case Id:** | | *PR005* | | |
| **Pre-condition:** | | The actor is a registered Patient | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor logs into the system | | |  |
| **2.** | The system authenticates the actor | | | Allow access to the system for Appointment |
| **3.** | The actor searches for the Doctors | | | Display a listing page of Doctors available for Patient |
| **4.** | An actor can also interact with the Chatbot without Login | | | It will recommend the doctor according to their Disease |
| **Alternate Scenarios:** | | | | |
| **1a:**  System fails to authenticate the actor  The system informs the actor and doesn’t allow the actor to proceed  **2a:** The chatbot fails to recommend the doctor  The system informs the actor  **3a:** The system fails to search  The system informs the actor | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | The System outputs the results for the search of Doctors. | | | |
| **2.** | Users can take the Appointment | | | |
| **Use Case Cross referenced** | | | *Login*  *Appointment* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR007: Patient History** | | | | |
| **Actors:**  Doctor | | | | |
| **Use case Id:** | | *PR007* | | |
| **Pre-condition:** | | The actor is logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor logged in to the system. | | | Allow access to the system |
| **2.** | The actor manages the data of users | | |  |
| **3.** | The actor views the history of the user | | | Displays the patient history to the doctor |
| **Alternate Scenarios:** | | | | |
| **1a: The system** fails to authenticate the actor.  The system informs the actor and doesn’t allow the actor to proceed.  **2a:** The actor may view the wrong information. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | Data of user is displayed to doctor. | | | |
|  | Updating of system | | | |
| **Use Case Cross referenced** | | | *Log in*  *Patient history* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR008: Prescription Details** | | | | |
| **Actors:**  Doctor | | | | |
| **Use case Id:** | | *PR008* | | |
| **Pre-condition:** | | The actor is logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor logged in to the system. | | | Allow access to the system |
| **2.** | The actor manages the data of users | | |  |
| **3.** | The actor views prescription details. | | | Displays the prescription details to the doctor. |
| **Alternate Scenarios:** | | | | |
| **1a:** The system fails to authenticate the actor.  The system informs the actor and doesn’t allow the actor to proceed.  **2a:** The actor may view information of wrong user. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | Data of user is displayed to doctor. | | | |
|  | Updating of system | | | |
| **Use Case Cross referenced** | | | *Log in*  *Prescription details* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR009: Record patient Details** | | | | |
| **Actors:**  Admin | | | | |
| **Use case Id:** | | *PR009* | | |
| **Pre-condition:** | | The actor is logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor logged in to the system. | | | Allow access to the system |
| **2.** | The actor manages the data of users | | |  |
| **3.** | The actor records the details of the patient. | | | Stores user data in the database |
| **Alternate Scenarios:** | | | | |
| **1a:** The system fails to authenticate the actor.  The system informs the actor and doesn’t allow the actor to proceed.  **2a:** data of the user is stored in a database for future use. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | Data of users is stored in a database. | | | |
|  | Updating of system | | | |
| **Use Case Cross referenced** | | | *Log in*  *Record patient details.* | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PR011: Manage Schedule** | | | | |
| **Actors:**  Admin | | | | |
| **Use case Id:** | | *PR011* | | |
| **Pre-condition:** | | The actor is logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The actor logged in to the system. | | | Allow access to the system |
| **2.** | The actor manages the data of users | | |  |
| **3.** | The actor manages the schedule | | | Stores user data in the database |
| **Alternate Scenarios:** | | | | |
| **1a:** The system fails to authenticate the actor.  The system informs the actor and doesn’t allow the actor to proceed.  **2a:** data of the user is stored in a database for future use. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | Data of users is stored in a database. | | | |
|  | Updating of system | | | |
| **Use Case Cross referenced** | | | *Log in*  *Manage Schedule* | |

**Registration Process of Software Requirements Specification**

**Adding Patients:**

The Hospital Management allows the team in the front workdesk to consist of brand-new clients to the system.

**Assigning an ID to the patient**:

The HMS enables the personnel at the front work desk to give a one-of-a-kind ID for every individual and then include them in the record sheet of the individual. The individuals can use the ID throughout their medical facility keep.

**Record Generation of SRS**:

**Deleting Patient ID:**

The staff in the administration section of the ward can delete the patient ID from the system when the Patient checks out from the health center.

**Info of the Patient:**

The Hospital Management System generates a record on every patient concerning various details like individual name, Phone number, bed number, the medical professional’s name whom it appoints, ward name, as well as much more.

**Database of SRS**

**Compulsory Individual Information**:

Every patient has some required data like phone number, their first and last name, individual wellness number, postcode, country, address, city, ‘person’s ID number, and so on

**Updating details of the Patient:**

The health center management system enables individuals to update the info of the individual as explained in the obligatory information consisting of a search feature that allows a user to hunt among various invoices if they want to credit an issued invoice

**Appointment scheduling**

In appointment scheduling, we schedule the appointment for a new patient in which we assign the date-time department and doctors are available that time. if the patient wants a particular doctor then we can search the doctor’s scheduling and available time for that doctor here we add the urgency and the remainder to the patient we can also cancel the appointment of a particular patient.

**Employee**

In this module, we can register the new employee for which we can enter the basic information about the employee and his professional details.

**Doctors**

In this we can view today doctors on-call schedule department wise here we can create the duty plan of doctors and edit or update the plan of a particular doctor here we can add delete the doctor to a particular department.

**Admin module**

Manage the department of the hospital user doctors pharmacist

Watch the appointment of doctors

Watch the transaction reports

Watch medicine status of hospital stock

Watch the patient info

**User module**

View the appointment list and the status with the doctor

View prescription detail

View medicine from the doctor

View doctor list

Manage own profile

**Doctor module**

View patient detail

Manage the appointment with the patient

Create the prescription for a patient

Provide medication for a patient

Manage own profile

**CHAT bot**

Interact with the patient

Recommend the doctor

Understand the disease and provide doctor detail

**Non-Functional Requirements:**

The non-functional requirements of the Civil Hospital Management System comprise several software requirements specifications that cover a variety of processes, including security, performance, maintainability, and reliability.

**1: Performance Expectations**

* Availability:

After reviewing the patient details and other information, the system will respond within one

second.

* Capacity:

The system must be able to support 1000 persons at once.

* User Interface:

The user interface screen will respond in less than 5 seconds.

* Conformity:

The system must adhere to Microsoft's accessibility guidelines.

**2: Safety Requirements**

If a catastrophic failure, such as a disc crash, causes extensive damage to a large portion of the database, the recovery method restores a previous copy of the database that was backed up to archival storage and reconstructs a more current state by reapplying or redoing committed transaction operations from the backed-up log, up to the time of failure.

**3: Security Requirements**

All administrative and data input operators have unique logins, so the system can tell who is currently logged in. No intruders are permitted except system administrators, and no one can edit records or sensitive data.

**4: User Documentation**

User manual:

The user manual instructs people on how to operate a product safely and efficiently. A good user manual shows your consumers that you are not just about whether they buy your product, but also about whether they have a fantastic time using it. Customers who believe you care about them are more likely to return.

Online help:

Online help gives users a platform to directly contact our team if they face any issue regarding the project. Customers can contact us using social media e.g. email, Facebook, etc.

Tutorials:

Tutorials help users understand projects by both video and audio. They provide a much better opportunity to get to know about projects.