SOFTWARE REQUIREMENTS SPECIFICATION

MediRecords : Seamless Care Starts Here

Naman Agrahari – Team Lead
Hardik Maheshwari– Associate Technical Consultant
Abhishek Saha– Associate Technical Consultant
Athul Robert– Associate Technical Consultant
Thilagavathy ravi– Associate Technical Consultant

08/05/2024

Functional Requirements Document
Authorization Memorandum

I have carefully assessed the Functional Requirements Document for MediRecords:Seamless care starts here

The document is accepted.

We fully accept the changes as needed improvements and authorize initiation of work to proceed. Based on our authority and judgment, the continued operation of this system is authorized.

Thilagavathi
Project Manager

DATE: -08/07/2024

Ramesh Subramanium
Director

DATE:-08/07/2024

Naman Agrahari – Team Lead
Hardik Maheshwari– Technical Consultant
Abhishek Saga– Technical Consultant
Athul Robert – Technical Consultant
Thilagavathy Ravi – Technical Consultant

1.0 INTRODUCTION

1.1 Purpose

This document details the requirements for developing a MediRecords application. The system aims to streamline patient and medical data management in healthcare institutions, offering efficient database management, operational enhancements, and data visualization capabilities for improved research and analysis.

2.0 BUSINESS REQUIREMENT OVERVIEW

- Develop an automated system to significantly reduce the manual effort and time required for healthcare professionals to manage patient and medical data within healthcare institutions.
- Ensure the system accurately captures and reflects essential patient and medical information, adhering to regulatory standards and healthcare best practices.
- Support flexible input of data in various formats to accommodate diverse needs and sources of medical information, enhancing usability and compatibility across different healthcare settings.

2.1 Constraints

- Ensure seamless compatibility with various medical data formats to facilitate efficient management and integration of patient records across healthcare institutions.
- Ensure precise handling of patient and medical data to support reliable decision-making and clinical workflows, minimizing errors and optimizing healthcare operations.

FUNCTIONAL REQUIREMENTS SPECIFICATION

User requirements will be satisfied through the MediRecords system's intuitive interface, allowing users to input and manage patient and doctor information easily. The system's functionality, including patient registration, medical record management, data visualization, and data export, ensures accurate and efficient handling of healthcare data. Regular feedback mechanisms will be incorporated to address user needs and enhance user satisfaction.

1. User Management

1.1 Admin Registration

- Implement a registration form to capture essential details: id,name, contact information, credentials.
- Validate and securely store user data in the database.

1.2 Admin Authentication

- Implement secure login functionality using username/email and password.

1.3 Doctor Data Management

- Allow admins to assign the doctors to patients.
- Provide a feature to fetch doctor details by ID.

2. Patient Management

2.1 Patient Registration

- Passing the values to capture patient information: name, date of birth, gender, contact details, medical history, and insurance information.
- Store patient data in a secure and structured database.

2.2 Patient Data Management

- Provide functionality to delete patient records with appropriate warnings and confirmations.
- Allow admins to update existing patient records with new information.
- Provide a feature to fetch patient details by ID.

3. Medical Records Management

3.1 Record Creation

- Design a system to create and store medical records, including diagnoses, treatment plans, prescriptions, lab results, and visit notes.

3.2 Record Modification

- Enable doctors to update existing medical records with new information.

3.3 Record Deletion

- Provide functionality to delete medical records, ensuring compliance with data retention policies.

3.4 Record Fetching

- Provide functionality to fetch medical records of the patients based upon the id.

4. Data Export

4.1 Data view

- Ensure the export functionality is secure and accessible to authorized users only.
- We can view the generated data at the backend

5. Data Visualization

5.1 Analytic View

- This includes visualizations for patient demographics, medical conditions, blood groups, treatment outcomes, etc.

6. API Consumption

6.1 Data Integration

- Integrate with external APIs to fetch and import data in both CSV and JSON formats.
- Ensure proper handling and validation of data received from external APIs.
    -

NON-FUNCTIONAL REQUIREMENTS SPECIFICATION

· **Performance:**

- The system should handle an increasing number of users and data efficiently.
- Ensure the system responds quickly to user actions and queries
- Response times for critical operations such as patient admissions, discharge summaries, and medical record updates should be within acceptable limits to support uninterrupted healthcare delivery.

Scalability:

- The system should be scalable to accommodate growth in patient records and data volume over time without compromising performance.
- It should support horizontal scaling to handle increased concurrent user access during peak times without degradation in system responsiveness.

Portability:

- The application should be designed to be easily deployable across various healthcare environments, including hospitals, clinics, and healthcare centers.
- It should support deployment on different operating systems commonly used in healthcare IT infrastructure, such as Windows Server and Linux distributions.

Compatibility:

- Ensure compatibility with existing healthcare IT systems and standards, allowing seamless integration with Electronic Health Record (EHR) systems, medical billing software, and other healthcare management tools.

Security:

- Implement secure login functionality using username/email and password.
- Ensuring the login system provides clear and specific error messages:
- If the username already exists during registration, display "Username already exists." If there is an error in the username format, display "Invalid username format."

Usability:

- Streamlines the handling of patient records and medical histories, improving operational efficiency for healthcare institutions
- Provides data visualization tools to support research and enhance data interpretation for informed decision-making.

Maintainability:

- Using a modular architecture to facilitate easy updates and maintenance.

Interoperability:

- Providing well-documented RESTful APIs for seamless integration with other healthcare systems and third-party applications.

Data Accuracy and Consistency:

- Implementing robust data validation mechanisms to ensure data accuracy at the point of entry.