Title

AI-Powered Smart Medical Assistant for Patient Records, Report Analysis and Diagnosis Support

Author: Raj Department: Computer Science & Engineering Guide: Self-Guided Project

Abstract

This project proposes an AI-powered smart medical assistant that digitizes patient records, analyzes medical reports using artificial intelligence, and provides diagnosis support. The system is designed using a microservices architecture with Java Spring Boot and Python FastAPI, deployed using Docker.

Keywords

Introduction

Healthcare systems face increasing pressure due to growing patient data and limited time for analysis. Manual record handling leads to inefficiency and errors. This project aims to solve these issues by automating record management and analysis.

Problem Statement

Medical data fragmentation, lack of intelligent report interpretation, and delayed emergency response reduce healthcare effectiveness.

Literature Survey

Various electronic health record systems exist, but most lack AI-driven report analysis and decision support. Recent research shows AI improves diagnostic accuracy.

Proposed System

The system includes patient record management, AI report analyzer, diagnosis support, and emergency access modules.

System Architecture

The architecture follows a microservices model where services communicate via REST APIs and are containerized using Docker.

Algorithm Design

Algorithms include OCR for text extraction, machine learning classification for disease prediction, and rule-based report analysis.

Implementation

Backend services are developed in Java Spring Boot, AI services in Python FastAPI, and data stored in PostgreSQL.

Results and Discussion

The system successfully analyzes reports and provides meaningful insights, improving efficiency and response time.

Security and Ethics

Patient data security is ensured using authentication and access control. The system follows ethical AI principles.

Conclusion

The project demonstrates the feasibility of AI-assisted medical systems in improving healthcare workflows.

Future Scope

Future enhancements include wearable integration, blockchain security, and telemedicine.

SYSTEM ARCHITECTURE DIAGRAM

User (Web / Mobile)

|

Frontend (React)

|

Spring Boot Backend (Java)

|

| Patient Service |

| Authentication Service |

| Emergency Service |

|

Python AI Service (FastAPI)

|

PostgreSQL Database

All services are containerized and managed using Docker Compose.

SYSTEM FLOWCHART

Start

|

User Login

|

Upload Medical Report / Enter Symptoms

|

Data Stored in Database

|

AI Report Analysis / Disease Prediction

|

Display Results to User

|

Emergency Mode (if required)

|

End

USE CASE DIAGRAM

Actors:

- Patient

- Doctor

- Admin

Patient Use Cases:

- Register / Login

- Upload Reports

- View AI Analysis

- Emergency Access

Doctor Use Cases:

- View Patient Records

- Analyze Reports

- Provide Consultation

Admin Use Cases:

- Manage Users

- Monitor System