HOSPITAL MANAGEMENT SYSTEM

(An End-to-End Solution for Healthcare Management)

Abstract

The **Hospital Management System (HMS)** is a web-based solution designed to **digitize and automate hospital operations**, improving efficiency in managing **patient records, appointments, lab tests, and billing**. The system ensures **role-based access** for administrators, doctors, receptionists, and lab technicians, reducing manual errors and enhancing workflow. Key features include **real-time patient tracking, secure authentication, automated report generation, and inventory management** for medicines and medical supplies. By integrating **role-specific dashboards, appointment scheduling, and lab report automation**, the system **enhances hospital administration** and improves **patient care delivery**.

Keywords:

Hospital Management, Patient Records, Role-Based Access, Lab Test Automation, MERN Stack, Healthcare System

Existing System:

Traditional hospital management relies on **manual paperwork or disconnected systems**, causing **data loss, appointment conflicts, inefficiencies in medical record handling, and delays in lab test processing**.

Proposed System:

The proposed **MERN-stack-based HMS** will provide:

1. **Centralized patient records** for quick access
2. **Role-based access control** (Admin, Doctor, Receptionist, Lab Technician)
3. **Automated appointment scheduling & reminders**
4. **Efficient lab test processing & report management**
5. **Doctor-specific dashboards** for case tracking
6. **Secure login authentication & patient confidentiality**

Modules & Contributors:

1. **M. Vigneshwar Reddy (160123737315)**

   Backend Development: Database Design, API Development (Node.js, MongoDB)

   Lab Test & Report Module: Automation of test requests and report generation

2. **N. Shashidhar (160123737317)**

   Frontend Development: React.js UI, Dashboard Designs, User Authentication

   Appointment & Patient Management: Role-based access & scheduling features

Project Timeline:

Month 1: Backend & Database Setup

- **Week 1:** Project Initialization & GitHub Setup
- **Week 2:** Database Design & Role-Based Authentication

Month 2: Core Feature Development

- **Week 3-4:** Patient Management (Registration, Appointments, Medical Records)
- **Week 5-6:** Doctor & Staff Management (Scheduling, Prescriptions, Lab Requests)

Final Week: Deployment & Testing

- **Week 7:** Bug Fixes, Testing, and Deployment

Technologies:

Frontend: React.js,HTML,CSS

Backend: Node.js with Express.js

Database: MongoDB

Software Requirements:

Version Control: GitHub

Visual Studio Code

Postman

MongoDB Compass

Hardware Requirements:

Processor: Intel/AMD Quad-Core or higher

RAM: 8 GB or more

Storage: SSD recommended for faster performance