

AI Hospital Management System - Documentation

1. Project Overview

The **AI Hospital Management System** is an advanced web-based healthcare solution integrating AI-driven disease prediction, patient management, and doctor-patient interactions. The system leverages machine learning to provide accurate medical insights and streamline hospital operations.

Objectives:

- Provide an intelligent symptom-based disease prediction system.
- Facilitate seamless patient-doctor management.
- Store and manage patient medical records securely.
- Implement role-based authentication for different user types.

2. Features

User Authentication

- Role-based access for **Patients, Doctors, and Admins**.
- Secure login and registration system.
- Password hashing for security.

AI-Based Disease Prediction

- Uses **Random Forest Model** (`random_forest_model.pkl`) to predict diseases based on user symptoms.
- Provides detailed information on diseases, including **description, medications, diets, and precautions**.
- Fetches disease-related details from uploaded CSV datasets.

Patient Management

- Patients can store and update their medical history.
- Doctors can view assigned patient details and suggest treatments.
- Admins can manage patient records.

Admin Dashboard

- Manage **doctors, patients, and medical history**.
- Assign doctors to patients.

- Monitor system analytics.

Dataset Integration

- Preloaded datasets for diseases, symptoms, medications, diets, and precautions.
- CSV-based retrieval and analysis.

Interactive Web-Based UI

- Developed using **HTML, CSS, JavaScript (Frontend)** and **Flask (Backend)**.
 - Responsive and user-friendly design.
-

3. Project Directory Structure

```
/ai_hospital_management_project
|— authapp.py      # Handles authentication (login/register)
|— config.py       # System configurations and database settings
|— models.py        # Database models for patients, doctors, and records
|— projectwinger.ipynb # Jupyter notebook for data analysis and AI model
|— random_forest_model.pkl # Pretrained machine learning model
|— routes.py        # Flask routes for handling web requests
|— templates/
|   |— index.html    # Homepage
|   |— login.html     # Login page
|   |— register.html  # User registration page
|   |— dashboard.html # Admin/Doctor dashboard
|   |— predictor.html # Disease prediction input form
|— static/          # CSS, JavaScript, and image files
|— datasets/
|   |— description.csv # Disease descriptions
|   |— diets.csv       # Recommended diets
|   |— medications.csv # Suggested medications
|   |— precautions.csv # Precautionary measures
```

```
|— Project Requirement.pdf # Detailed project requirements and objectives  
|— requirements.txt    # Required Python dependencies  
|— app.py          # Main Flask application entry point
```

4. Installation Guide

◆ **Prerequisites:**

- **Python 3.8+** installed on your system.
- **MySQL Database** for storing user and medical records.
- **Virtual Environment** (recommended for dependency management).

◆ **Setup Instructions:**

Step 1: Clone the Repository

```
git clone <repository-link>  
cd ai_hospital_management_project
```

Step 2: Create & Activate Virtual Environment

```
python -m venv venv  
source venv/bin/activate # On Mac/Linux  
venv\Scripts\activate # On Windows
```

Step 3: Install Required Dependencies

```
pip install -r requirements.txt
```

Step 4: Configure MySQL Database

1. Update config.py with MySQL credentials.
2. Run migrations to set up the database schema:

```
python manage.py migrate
```

Step 5: Run the Flask Application

```
python app.py
```

Step 6: Access the Web Interface

- Open a browser and go to: <http://127.0.0.1:5000>
- Log in as a **Patient** or **Doctor** to access functionalities.

5. Usage Instructions

◆ Patient Portal

- **Register/Login** to the system.
- Enter symptoms to receive **AI-driven disease predictions**.
- View suggested **medications, diets, and precautions**.
- Save medical history for future reference.

◆ Doctor/Admin Portal

- Manage **patients and their records**.
- View **assigned patients and their medical history**.
- **Assign treatments and provide medical advice**.

◆ AI-Based Prediction

- Uses the **Random Forest model** to predict diseases based on symptoms.
 - Pulls disease details from CSV datasets.
 - Displays recommendations, including medications and diets.
-

6. Dataset Details

The system uses CSV datasets for disease-related information:

Dataset File	Purpose
--------------	---------

description.csv Contains disease descriptions.

diets.csv Recommended dietary plans for diseases.

medications.csv Suggested medications for treatment.

precautions.csv Precautionary measures for each disease.

7. Machine Learning Model Integration

- The AI system is powered by a **Random Forest Classifier**.
- The model (random_forest_model.pkl) is pre-trained on symptom-disease mapping.

- Predictions are generated in real time when users enter symptoms.
 - Model performance evaluated using **accuracy, precision, recall, and F1-score**.
-

8. Future Enhancements

-  **Integration with Deep Learning models** for better accuracy.
 -  **Real-time chat feature** for patient-doctor interactions.
 -  **Automated prescription generation** based on AI predictions.
 -  **Mobile application version** for remote accessibility.
-

9. Conclusion

The **AI Hospital Management System** is a cutting-edge project that integrates AI into healthcare, making patient management and disease prediction more efficient. By leveraging machine learning and structured patient data, the system provides intelligent insights, improving medical decision-making and hospital workflow. 

For further details, refer to **Project Requirement.pdf**.