

# MedAI – AI-Powered Symptom Analyzer & Patient Management System

## Stage 1: Requirement Analysis Report

### 1. Project Overview

MedAI is a web-based healthcare system designed to assist patients in identifying diseases based on symptoms. It also provides medical recommendations and enables efficient doctor-patient management.

#### Understanding Client Requirements

- **Client Perspective:** The system should allow patients to input symptoms and receive accurate disease predictions, along with medical recommendations.
- **Input Data:**
  - Symptoms provided by patients
  - Predefined datasets containing symptoms, severity, diseases, medications, precautions, diets, and workouts
- **Expected Outputs:**
  - Predicted disease
  - Medical guidance (medications, precautions, workouts, diet)
  - Patient history tracking
- **Overall Project Goals:**
  - Create a reliable AI-powered symptom analyzer
  - Enable smooth patient-doctor interaction via an admin portal
  - Maintain accurate and secure patient records

### 2. Objectives

- **AI-powered symptom analysis and disease prediction**
- **Web-based GUI** with role-based access (patients & doctors/admins)
- **Patient history tracking** with doctor comments
- **Medical guidance:** Precautions, medications, diets, and workouts
- **Secure authentication and data privacy**

### 3. Functional Requirements

#### ◆ Patient Interface

- Input symptoms for analysis
- Receive predictive results:
  - **Disease Name**
  - **Description**
  - **Precautions**
  - **Medications**
  - **Diet Recommendations**
  - **Workout Plans**

#### ◆ Admin/Doctor Interface

- Manage doctors and patients
- Assign patients to doctors
- Track and update patient medical history

### 4. Data Insights & Observations

#### Training Dataset (ML Model Input)

- **Rows:** 4920 | **Columns:** 133
- **Purpose:** Train AI models to predict diseases based on symptoms
- **Target variable:** prognosis (Disease name)
- **No missing values**

#### Symptoms & Severity Dataset

- Symptoms are weighted (severity scores)
- Will help in **feature weighting for ML models**

#### Medical Guidance Datasets

- **Precautions, Diets, Workouts, Medications** mapped to diseases
- **Some missing values in symptoms and precautions datasets**, which require preprocessing

### 5. Machine Learning Approach

- **Algorithms to be tested:**
  - Support Vector Classifier (SVC)

- Random Forest Classifier
  - K-Nearest Neighbors (KNN)
  - Logistic Regression
  - Gradient Boosting Classifier
  - Neural Networks (if needed)
- **Model Evaluation Metrics:**
    - Accuracy, Precision, Recall, F1-score, and Confusion Matrix
    - Select the best-performing model for deployment

## 6. System Design Considerations

- **Tech Stack:**
  - **Backend:** Python (Flask/Django), Pandas, Scikit-learn
  - **Frontend:** React.js / HTML, CSS, JavaScript
  - **Database:** SQL / Firebase for patient records
- **Security:** Role-based authentication (patients & doctors/admins)

### Data Preprocessing Summary

- **Removed unnecessary columns** (Unnamed columns)
- **Checked for missing values** and filled missing data where needed
- **Encoded categorical values** (e.g., disease names into numerical labels)
- **Applied symptom severity weighting**
- **Generated visualizations:**
  - Symptom Severity Distribution
  - Most Common Symptoms
  - Disease Occurrence in Training Data
  - Missing Values Heatmap
  - Correlation Matrix
  - Boxplot of Symptom Severity