"SESSION BASED MEDICAL AND DOCTOR RECOMMENDATION SYSTEM"

CAPSTONE PROJECT

Course Code: CAP5001

"Project Proposal Report"

By

Ankam Srinivas – 21MIS7009

Vishal R Nadagoudar- 21MIS7060

Murari Jayasurya – 21MIS7085

Under the guidance of

Prof. Dr. Naresh Sammeta

School of Computer Science and Engineering

VIT-AP University

Amaravathi –522237

First Review

TITLE OF THE PROJECT: SESSION BASED MEDICAL AND DOCTOR RECOMMENDATION SYSTEM.

PURPOSE OF THE SYSTEM:

The purpose of the system is to provide users with an intelligent, accessible platform that predicts diseases based on their reported symptoms and recommends suitable doctors, thereby bridging the gap between patients and healthcare providers through accurate machine learning predictions and an easy-to-use web interface.

PROBLEMS IN THE EXISTING SYSTEM:

- **Lack of Immediate Diagnosis**: Traditional systems often require physical visits or lengthy consultations for diagnosis, leading to delays in identifying and addressing health issues.

- **Limited Accessibility**: Many existing platforms do not provide region-specific doctor recommendations, making it difficult for users to find nearby and relevant healthcare providers.

- **Over-Reliance on Manual Processes**: Manual data entry and symptom analysis can be time-consuming and error-prone, reducing the efficiency of healthcare delivery.

- **Poor Integration of Services**: Existing systems may lack a unified interface for both patients and doctors, resulting in fragmented communication and service delivery.

SOLUTION OF THESE PROBLEMS:

1. **Automated Disease Prediction**: The system uses machine learning to provide instant and accurate disease predictions based on user-reported symptoms, reducing the need for initial physical consultations.
2. **Location-Based Doctor Recommendations**: It offers region-specific doctor suggestions (focused on Vijayawada), improving accessibility and helping users connect with nearby qualified professionals.
3. **Streamlined Data Handling**: By automating symptom analysis and storing user and doctor data in a structured database, the system minimizes manual errors and speeds up the diagnosis process.
4. **Enhancing Scalability**: The system is designed to handle large datasets and diverse plant species, making it applicable in various agricultural settings.

SCOPE OF THE PROJECT:

The scope of this project lies in enhancing healthcare accessibility and efficiency through a smart, AI-driven web application that predicts diseases and recommends suitable doctors based on user-input symptoms. Designed primarily for urban and semi-urban populations, the system offers accurate and timely medical suggestions using a fine-tuned K-Nearest Neighbors (KNN) machine learning model. By integrating both patient and doctor interfaces, it creates a comprehensive ecosystem that facilitates smooth interaction between users and healthcare providers. The system's web-based design ensures wide accessibility, while the use of a lightweight ML model makes it feasible for real-time deployment even on low-resource servers. Currently focused on doctors in Vijayawada, the system can be expanded to include more regions and specialties, offering scalable potential. Additionally, its modular architecture allows future integration of features such as prescription suggestions, appointment booking, and medical history tracking, making it a robust foundation for advanced health tech applications.

FUNCTIONAL COMPONENTS OF THE PROJECT:

- **User Authentication and Session Management**: Allows patients and doctors to securely register, log in, and manage sessions for personalized access.

- **Symptom-Based Disease Prediction**: Enables users to input symptoms and receive accurate disease predictions using the K-Nearest Neighbor (KNN) machine learning model.

- **Doctor Recommendation System**: Suggests suitable doctors based on the predicted disease and user location, with a focus on professionals in Vijayawada.

- **Doctor Registration and Profile Management**: Allows doctors to create profiles, update their details, and make their services available to patients through the platform.

- **Web Interface and Database Integration**: Provides a user-friendly frontend (HTML, CSS, JavaScript) connected to a Flask backend and SQLite database for smooth data handling and functionality.

STUDY OF THE SYSTEM:

The system is designed to bridge the gap between patients and healthcare providers by leveraging machine learning for disease prediction and doctor recommendation. It begins with user authentication, allowing patients and doctors to register and access personalized features. Patients can enter their symptoms through a simple interface, and the system uses a fine-tuned K-Nearest Neighbors (KNN) algorithm to predict possible diseases with high accuracy. Based on the prediction, it recommends relevant doctors, particularly from Vijayawada, whose information is manually curated. Doctors can also register, manage their profiles, and offer services. The entire system is built on a web-based platform using HTML, CSS, and JavaScript on the frontend, Flask on the backend, and SQLite for data storage, ensuring seamless performance and accessibility.

Modules Involved

User/Patient Module:

- Handles patient and doctor registration, login, authentication, and session management. It ensures secure and personalized access for different types of users.

Symptom Input and Disease Prediction Module:

- Allows users to input symptoms and utilizes the K-Nearest Neighbors (KNN) algorithm to predict the most probable disease based on trained machine learning models.

Doctor Recommendation Module:

- Recommends suitable doctors based on the predicted disease and user's location, using a manually curated database of verified professionals in Vijayawada.

Doctor Management Module:

- Enables doctors to register, update their profiles, and manage their availability and specialization for patient recommendations.

Web Application & Database Module:

- Includes the frontend (HTML, CSS, JavaScript), backend (Flask), and database (SQLite) integration for data handling, UI interaction, and smooth functioning of all features.

INPUT / OUTPUT Specifications:

Input Specifications:

1. **User Registration Details**:
   - **Patient**: Name, Email, Password.
   - **Doctor**: Name, Email, Password, Specialization, Location, Contact Info

2. **Login Credentials**:
   - Email and Password for both patients and doctors.

3. **Symptom Input:**
   - A list of symptoms selected or typed by the user

4. **Doctor Information (Admin Input):**
   - Manually entered doctor data including name, specialization, location, and availability

Output Specifications:

1. **Login/Registration Status:**
   - Success or error messages (e.g., “Login Successful”, “Invalid Credentials”)

2. **Disease Prediction Result:**
   - Predicted disease name based on input symptoms

3. **Recommended Doctors List:**
   - Display of suitable doctors with names, specializations, contact info, and location

PERFORMANCE REQUIREMENTS:

- **Accuracy**: The disease prediction module should maintain an accuracy rate of at least 90%, with the KNN model consistently delivering reliable results across varied symptom inputs.

- **Response Time**: The system should predict diseases and recommend doctors within 2–3 seconds after symptom input, ensuring a smooth user experience.

- **Scalability**: The system must efficiently handle multiple user sessions simultaneously without performance degradation, especially during peak usage times.

FEASIBILITY REPORT:

Technical Feasibility:

- **Technologies Used**: The project uses widely supported and well-documented technologies such as Python, Flask, HTML, CSS, JavaScript, and SQLite, ensuring easy development and maintenance.
- **Machine Learning Integration**: The use of lightweight ML models like KNN ensures compatibility with low-resource servers, making deployment on platforms like Render technically feasible.
- **System Compatibility**: The application is web-based, accessible via any standard browser, and does not require high-end hardware, making it technically feasible across devices and platforms.

Operational Feasibility:

- **User Interface**: The system is designed with a simple and intuitive interface, ensuring ease of use for both patients and doctors without needing technical expertise.
- **Role-Based Functionality**: The platform supports different user roles (patients and doctors), enabling smooth operation and task management as per user needs.
- **Scalability**: Designed to accommodate a growing user base and expanding dataset.

Economic Feasibility:

- **Low Development Cost**: The project uses open-source tools and platforms, minimizing software licensing and development costs.
- **Affordable Hosting**: Deployment on platforms like Render offers free or low-cost hosting options, making it economically viable for small-scale or educational purposes.
- **Cost-Efficient Maintenance**: Due to its lightweight design and minimal resource usage, the system incurs low operational and maintenance expenses.