Project Design Phase-II

Solution Architecture

Date	23 October 2023
Team ID	Team - 592529
Project Name	Project - Disease Prediction Using Machine Learning
Maximum Marks	4 Marks

The solution architecture for the healthcare model described involves a user-friendly web application designed to address the challenges of modern, fast-paced living where limited time is available for healthcare. This application utilizes a machine learning model to predict up to 42 different diseases when presented with a set of symptoms as input. Here's an overview of the solution's architecture:

1. **User Interface (UI)**
   The web application provides an intuitive and user-friendly interface that allows users to input their symptoms. It should be designed to be easily accessible and understandable for both medical professionals and the general public. The UI should be responsive and optimized for various devices.

2. **Symptom Dataset:**

   The application incorporates extensive and constantly updated datasets of symptoms associated with various diseases. This dataset serves as the foundation for the machine learning model's predictions. Collect a large dataset of medical records, symptoms, and corresponding diseases. The dataset should be diverse and well-labelled

3. **Machine Learning Model:**

   At the core of the system is a machine learning model, which has been trained on a vast dataset of medical records, symptoms, and corresponding diseases. This model uses advanced algorithms to analyze the provided symptoms and predict potential diseases. The choice of machine learning model, whether it's based on decision trees, neural networks, or other techniques, should be optimized for both accuracy and speed.

4. **CNN (Convolutional neural networks)**

- input Layer: The input layer will consist of symptom data. Each symptom can be represented as a sequence of one-hot encoded values or word embeddings.

- Convolutional Layer: Use multiple convolutional layers with different kernel sizes to capture patterns in the symptom sequences. The depth of the filters can increase with the depth of the network.

- Pooling Layers: Incorporate pooling layers (e.g., max-pooling) to reduce the spatial dimensions and retain important features.

- Fully Connected Layer: Flatten the output from the convolutional layers and pass it through one or more fully connected layers to make predictions.

5. **Evaluation and Validation:**

- Split the dataset into training, validation, and test sets to evaluate the model's performance.

- Measure key metrics like accuracy, precision, recall, and F1-score to assess the model's predictive capabilities.

The Architecture