## **Project Title:**

**CareWise : AI Symptom Checker and Treatment Advisor**

## **Team Name:**

**WE**

## **Team Members:**

* **Rupali Rathod**
* **Sara Fathima**
* **Nehadiya Saman**
* **Snigdha Talasila**

## **Phase-1: Brainstorming & Ideation**

### **Objective:**

### Develop an AI-powered health and symptom checker tool to provide users with symptom-related guidance, treatment suggestions, and real-time health updates.

### **Key Points:**

1. **Problem Statement:**
   * Many users struggle to identify symptoms accurately and need reliable health information before visiting a doctor.
   * Users also require treatment suggestions and an understanding of recovery time based on symptoms.
   * Additionally, personalized heath recommendations for the symptoms user gives.
2. **Proposed Solution:**
   * **Carewise AI**: An AI-powered application to provide real-time symptom diagnosis, treatment suggestions, and recovery information based on user input.
   * The app leverages advanced NLP and AI algorithms to analyze symptoms, provide home remedies, suggest medicines, and provide global health data.
3. **Target Users:**
   * **People with health concerns** who want immediate advice about their symptoms.
   * **Healthcare professionals** looking for quick treatment options for common symptoms.
   * **Health-conscious individuals** seeking preventative health advice and information on illnesses like flu, colds, and other common ailments.
4. **Expected Outcome:**
   * An easy-to-use, intuitive app that helps users analyze symptoms, find treatment suggestions, and gain health insights based on real-time data.

## **Phase-2: Requirement Analysis**

### **Objective:**

Define the technical and functional requirements for the **Carewise AI** app.

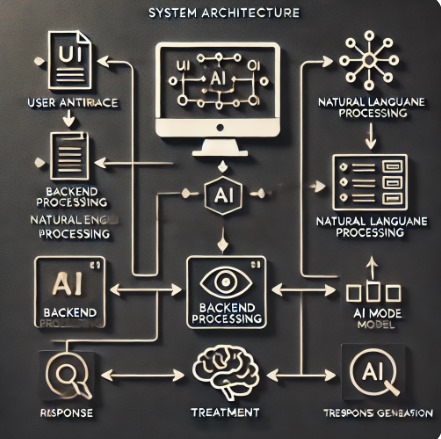
### **Key Points:**

1. **Technical Requirements:**
   * **Programming Language:** Python
   * **Backend:** Integrating health data sources and AI-powered diagnosis tools.
   * **Frontend:** Streamlit Web Framework
   * **Database:** Not required initially (API-based queries)
2. **Functional Requirements:**
   * Ability to analyze symptoms using AI models.
   * Display treatment suggestions, home remedies, and recovery time based on analyzed symptoms.
   * Natural language processing for symptom matching and diagnosis.
   * Personalized heath recommendation based on the symptoms user provides on a weekly/monthly/daily basis.
3. **Constraints & Challenges:**
   * Handling limitations with API call frequencies and data accuracy.
   * Maintaining a smooth and user-friendly interface with Streamlit.

## **Phase-3: Project Design**

### **Objective:**

Develop the architecture and user flow of the application.



### **Key Points:**

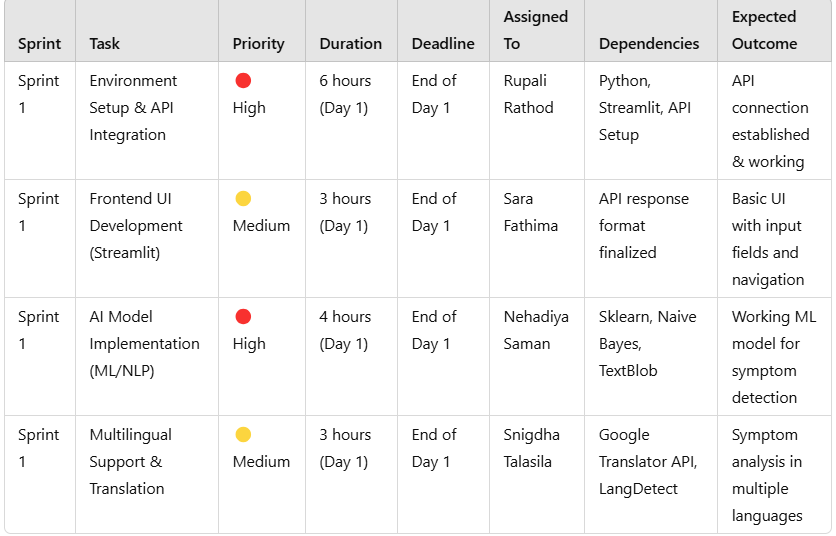
1. **System Architecture:**
   * User enters a health-related query (e.g., "I have a fever and sore throat") via the UI.
   * The query is processed using an AI model and symptom analysis algorithm.
   * The backend fetches relevant health data (e.g., treatments, medicines) and presents it to the user.
2. **User Flow:**
   * Step 1: User enters symptoms via text input (e.g., "I feel feverish and have a headache").
   * Step 2: Backend processes the input, predicts the symptom, and fetches the relevant treatment data.
   * Step 3: The frontend displays the results, including remedies, medicines, and estimated recovery time.
3. **UI/UX Considerations:**
   * Minimalist, user-friendly interface for seamless navigation.

## 

## **Phase-4: Project Planning (Agile Methodologies)**

### **Objective:**

Break down development tasks for efficient completion.



### **Sprint Planning with Priorities**

### **Sprint 1 – Setup & Integration (Day 1)**

**(🔴 High Priority)** Set up the **environment** & install dependencies.  
 **(🔴 High Priority)** Integrate **AI Implementation**.  
 **(🟡 Medium Priority)** Building **Frontend UI Development and Multilingual Support & Translation.**

## **Phase-5: Project Development**

### **Objective:**

### Implement core features of the **Carewise AI** app.

### **Key Points:**

1. **Technology Stack Used:**
   * **Frontend:** Streamlit
   * **Backend:** Google Translator API & Health APIs (e.g., COVID-19 API)
   * **Programming Language:** Python
2. **Development Process:**
   * Implement API key authentication for health data APIs.
   * Develop the logic to analyze symptoms and fetch treatment suggestions.
   * Optimize queries for faster symptom predictions and efficient data handling.
3. **Challenges & Fixes:**
   * **Challenge:** Delayed API response times.  
      **Fix:** Implement **caching** to store frequently queried results.

## **Phase-6: Functional & Performance Testing**

### **Objective:**

Ensure that the AutoSage App works as expected.

