**PROJECT TITLE:**

**HealthAI:** Intelligent Healthcare Assistant Using IBM Granite

**TEAM NAME:**

• Sunkavalli Manohar Chowdary

• Indukuri Venkata Surendra Varma

• Guttula Lithika

• Gudala Priyanka Sai Ramya

**PHASE-1: BRAINSTORMING & IDEATION**

**OBJECTIVE:**

• Identify the problem statement.

• Define the purpose and impact of the project.

**KEY POINTS:**

**• PROBLEM STATEMENT:**

Access to accurate and timely medical information is a major challenge for the public, leading to confusion, delays in treatment, and unnecessary panic.

**• PROPOSED SOLUTION:**

HealthAI is an intelligent healthcare assistant powered by IBM Granite and Watson AI. It provides real-time medical insights, symptom-based disease prediction, treatment suggestions, and user-friendly interaction through a chat interface.

**• TARGET USERS:**

General public, especially people without easy access to professional healthcare guidance.

**• EXPECTED OUTCOME:**

A web platform that delivers reliable, AI-powered medical assistance to users, enhancing health awareness and early self-assessment.

**PHASE-2: REQUIREMENT ANALYSIS**

**OBJECTIVE:**

• Define technical and functional requirements.

**KEY POINTS:**

**1. TECHNICAL REQUIREMENTS:**

**- Languages:** TypeScript, JavaScript

**- Frameworks:** Vite, Tailwind CSS

**- Tools**: IBM Watson, IBM Granite API, Node.js

**2. FUNCTIONAL REQUIREMENTS:**

- Patient symptom chat interface

- Disease prediction module

- Treatment suggestions

- Responsive UI/UX for accessibility

**3. CONSTRAINTS & CHALLENGES:**

- Dependence on external APIs (IBM Watson)

- Handling user data securely and ethically

- Ensuring accuracy of predictions

**PHASE-3: PROJECT DESIGN**

**OBJECTIVE:**

• Create the architecture and user flow.

**KEY POINTS:**

**1. SYSTEM ARCHITECTURE DIAGRAM:**

- Client-side UI (React/HTML)

- Backend API integration with IBM Watson

- Output response rendering to user

**2. USER FLOW:**

- User inputs symptoms → Chat interface sends data → Watson AI processes → Prediction/Advice shown

**3. UI/UX CONSIDERATIONS:**

- Simple, clean layout using Tailwind

- Mobile-friendly design

- Intuitive chat interaction

**PHASE-4: PROJECT PLANNING (AGILE METHODOLOGIES)**

**OBJECTIVE:**

• Break down the tasks using Agile methodologies.

**KEY POINTS:**

**1. SPRINT PLANNING:**

**- Sprint 1:** UI design and chat flow

**- Sprint 2:** Watson integration and disease prediction

**- Sprint 3:** Testing and final adjustments

**2. TASK ALLOCATION:**

**- UI Design:** Sunkavalli Manohar Chowdary

**- Backend & API:** Indukuri Venkata Surendra Varma

**- Testing:** Guttula Lithika

**- Documentation:** Gudala Priyanka Sai Ramya

**3. TIMELINE & MILESTONES:**

**- Week 1:** Interface design

**- Week 2:** Backend setup and Watson API integration

**- Week 3:** Bug fixes and deployment

**PHASE-5: PROJECT DEVELOPMENT**

**OBJECTIVE:**

• Code the project and integrate components.

**KEY POINTS:**

**1. TECHNOLOGY STACK USED:**

- HTML, TypeScript, Tailwind CSS, IBM Watson API, Vite

**2. DEVELOPMENT PROCESS:**

- UI created with Vite and Tailwind

- API endpoints integrated to handle health queries

- Responses generated using IBM Granite and Watson models

**3. CHALLENGES & FIXES:**

**- Issue:** Watson output delay → Fix: Added loading states

**- Issue:** UI responsiveness → Fix: Tailwind CSS optimizations

**PHASE-6: FUNCTIONAL & PERFORMANCE TESTING**

**OBJECTIVE:**

• Ensure the project works as expected.

**KEY POINTS:**

**• Functional Testing:** Verified symptom inputs generate correct Watson responses

**• Performance Testing:** Checked load time and API response speed

**• Bug Fixes:** Resolved form validation and mobile layout issues