Hackathon Project Phases Template for the **AI Symptom Checker and Treatment Advisor**.

Hackathon Project Phases Template

Project Title:

CareWise: AI Symptom Checker and Treatment Advisor

Team Name:

Ai-Alchemists

Team Members:

- P.Dinesh
- K.Harish
- S.Parveen
- U.V.N Shashank
- P.Sai krishna

Phase-1: Brainstorming & Ideation

Objective:

Develop an AI-powered symptom checker and treatment advisor that provides accurate, immediate medical advice, OTC medication recommendations, side effect warnings, and home remedies.

Key Points:

1. **Problem Statement:**

- Many individuals experience health symptoms but struggle to determine whether they need medical attention.
- Lack of accessible, accurate, and instant medical guidance leads to unnecessary panic or neglect of symptoms.
- OTC medications and home remedies are often used without proper knowledge of side effects or interactions.

2. **Proposed Solution:**
   - An **AI-powered assistant** that analyzes symptoms and provides instant insights, possible conditions, home remedies, and OTC medication recommendations.
   - **Side effect warnings and allergy precautions** ensure safe self-care.
   - **User-friendly chatbot interface** allows intuitive interactions for symptom checking.
3. **Target Users:**

   - Individuals seeking immediate medical guidance before consulting a doctor.
   - Parents looking for quick symptom assessment for their children.
   - People managing chronic conditions needing OTC medication safety checks.

   Expected Outcome:

- A working **AI-powered Symptom Checker and Treatment Advisor** providing accurate recommendations.
- An intuitive **chat-based interface** for seamless symptom assessment.
- A **reliable knowledge base** of symptoms, treatments, and precautions.

Phase-2: Requirement Analysis

Objective:

Define the technical and functional requirements for **CareWise**.

Key Points:

Technical Requirements:

- **Backend:** Node JS,Express
- **Frontend:** HTML,CSS,Javascrpit

Functional Requirements:

- Accept user **symptoms as input** and generate possible conditions.
- Recommend **OTC medications** with dosage and precautions.
- Warn users about **side effects, allergies, and contraindications**.
- Provide **home remedies** for minor symptoms.
- Include an **emergency indicator** for severe cases requiring immediate medical attention.

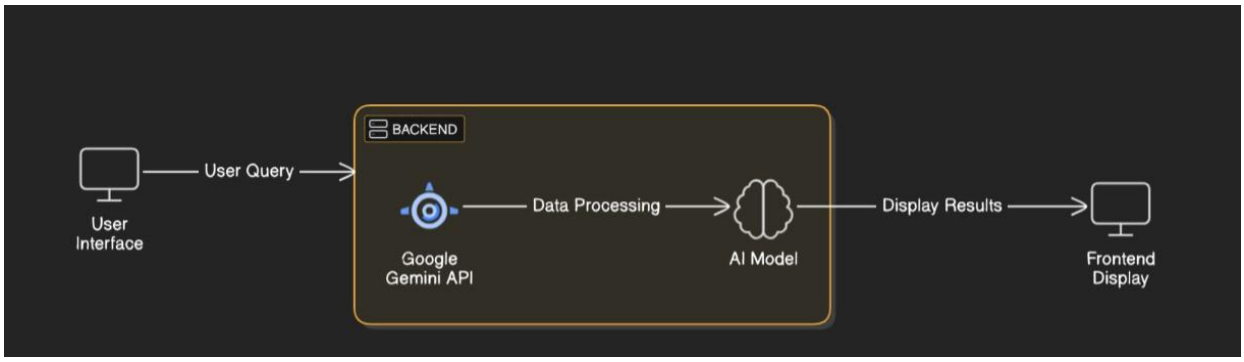
Constraints & Challenges:

- Ensuring **accurate** and **medically reliable** AI responses.
- Handling **complex symptom combinations** with AI reasoning.
- Avoiding **false positives** that may cause unnecessary concern.

Phase-3: Project Design

Objective:

Develop the architecture and user flow of the application.



Key Points:

1. **System Architecture:**
    - User Inputs Symptoms via the chatbot interface.Query is processed using Google Gemini API.
    - AI Processes the Query using Google Gemini API.
    - AI Analyzes Symptoms and determines possible conditions.
    - Emergency Indicator flags critical symptoms requiring urgent medical attention.
2. **User Flow:**
    - **Step 1:** User enters symptoms through the chatbot interface.
    - **Step 2:** The **backend processes symptoms** using the **Google Gemini API**
    - **Step 3:** AI **analyzes symptoms** and **fetches recommendations**.
    - **Step 4:** The app **displays results** with:
        - Possible **conditions**.
        - **OTC medications & precautions**.
        - **Home remedies & self-care tips**.
        - **Emergency indicator** (if applicable).
3. **UI/UX Considerations:**
    - Minimalist, user-friendly interface for smooth navigation.
    - **Chatbot-style** interaction for easy symptom checking.
    - **Clear categorization** of medications, precautions, and home remedies.

Phase-4: Project Planning (Agile Methodologies) in the format of your document:

Sprint	Task	Priority	Duration	Deadline	Assigned To	Dependencies	Expected Outcome
Sprint 1	Environment Setup & API Integration	🔴 High	6 hours (Day 1)	End of Day 1	Member 1	Google API	API connection established & working
Sprint 1	Frontend UI Development	🟡 Medium	2 hours (Day 1)	End of Day 1	Member 2	API response format finalized	Basic UI with input fields
Sprint 2	Symptom Analysis & Condition Mapping	🔴 High	3 hours (Day 2)	Mid-Day 2	Member 1 & 2	API response, UI elements ready	AI processes symptoms & generates insights
Sprint 2	Error Handling & Debugging	🔴 High	1.5 hours (Day 2)	Mid-Day 2	Member 1 & 4	API logs, UI inputs	Improved AI response accuracy & stability
Sprint 3	Testing & UI Enhancements	🟡 Medium	1.5 hours (Day 2)	Mid-Day 2	Member 2 & 3	API response, UI layout completed	Responsive UI, better user experience
Sprint 3	Final Presentation & Deployment	🟢 Low	1 hour (Day 2)	End of Day 2	Entire Team	Working prototype	Demo-ready project

Phase-4: Project Planning

Objective:

Break down development tasks for efficient completion.

Sprint Planning with Priorities:

Sprint Breakdown:

Sprint 1 – Setup & Integration (Day 1)

- (🔴 High Priority) **Set up environment & install dependencies**.
- (🔴 High Priority) **Integrate Google Gemini API**.
- (🟡 Medium Priority) **Build a basic UI with input fields**.

Sprint 2 – Core Features & Debugging (Day 2)

- (🔴 High Priority) **Implement symptom analysis & condition mapping**.
- (🔴 High Priority) **Debug API issues & handle errors in queries**.

Sprint 3 – Testing, Enhancements & Submission (Day 2)

- (🟡 Medium Priority) **Test AI responses, refine UI & fix UI bugs**.
- (🟢 Low Priority) **Final demo preparation & deployment**.

Phase-5: Project Development

Objective:

Implement core features of the **CareWise** application

Key Points:

1. **Technology Stack Used:**

- **Frontend:** html,css,javascript
- **Backend:** Google Gemini API (Gemini Pro)

Development Process:

Step 1: Implement API key authentication and integrate **Google Gemini API**.
Step 2: Develop **symptom analysis logic** to determine possible conditions.
Step 3: Implement **OTC medication recommendations** with precautions and dosages.
Step 4: Add **side effect warnings, allergy alerts, and contraindications**.
Step 5: Incorporate **home remedies** for minor symptoms.
Step 6: Develop an **emergency indicator** for critical conditions.
Step 7: Design and implement the **chatbot-style UI** for symptom assessment.
Step 8: Optimize query performance and response time.

Challenges & Fixes:

- **Challenge: AI-generated recommendations may lack medical reliability.**
- **Fix: Use a verified medical knowledge base to validate AI responses.**
- **Challenge: Complex symptom combinations may cause inaccurate predictions.**
- **Fix: Improve AI reasoning with better symptom pattern recognition.**
- **Challenge: Delayed API response times may impact user experience.**
- **Fix: Implement caching mechanisms for frequently queried symptoms.**
- **Challenge: Limited API calls per minute may affect performance.**
- **Fix: Optimize queries to fetch only essential data and reduce redundancy.**

Phase-6: Functional & Performance Testing

Objective:

Ensure that the CareWise application functions correctly, provides accurate medical advice, and delivers optimal performance.

Test Cases:

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	User inputs symptoms (e.g., "fever and headache")	AI provides possible conditions & relevant suggestions	✅ Passed	Tester 1
TC-002	Functional Testing	User asks for OTC medication for a symptom	System recommends appropriate OTC medication with dosage & precautions	✅ Passed	Tester 2
TC-003	Functional Testing	User inputs an allergy (e.g., "penicillin allergy")	System warns about contraindications & alternative medications	✅ Passed	Tester 3
TC-004	Performance Testing	API response time under 500ms	AI should return results quickly	⚠ Needs Optimization	Tester 1
TC-005	UI/UX Testing	Ensure chatbot interface is responsive	UI should work seamlessly on mobile & desktop	❌ Failed - UI broken on mobile	Tester 2
TC-006	Deployment Testing	Host the app using Streamlit Sharing	App should be accessible online	🚀 Deployed	DevOps Team

Bug Fixes & Improvements:

- ✅ **Fixed: Incorrect AI-generated responses for specific symptom combinations.**
- ✅ **Fixed: Improved accuracy of OTC medication recommendations.**
- ✅ **Fixed: Optimized API queries to reduce response time.**
- ❌ **Pending Fix: Mobile UI responsiveness issue.**

This follows the Hackathon Project Phases Template while maintaining clarity and structure. Let me know if you need any refinements! 🚀

Final Submission

1. **Project Report Based on the templates**
2. **Demo Video (3-5 Minutes)**
3. **GitHub/Code Repository Link**
4. **Presentation**