**Hackathon Project Phases Template**  for the **AutoSage App** project.

**Hackathon Project Phases Template**

**Project Title:** **CareWise: Al Symptom Checker and Treatment Advisor**

**AutoSage App Using Gemini Flash**

**Team Name:**

Code Cooks

**Team Members:**

* S.Sri Harshitha
* A.Srivalli
* T.Keerthi
* M.Shirisha

**Phase-1: Brainstorming & Ideation**

**Objective:**

Develop an AI-powered health assistant to analyze symptoms, provide treatment advice, and guide users on the urgency of medical care.

**Key Points:**

1. **Problem Statement:**
   * Many people rely on unverified online sources for health information, leading to misdiagnosis or unnecessary panic.
   * Limited access to immediate medical guidance, especially in remote areas.
2. **Proposed Solution:**

o **CareWise** , an AI-driven application that analyzes user symptoms and provides:

Possible causes of symptoms

Recommended medications & home remedies

Urgency assessment for medical attention

1. **Target Users:**
   * Individuals seeking quick medical insights
   * People in remote areas with limited access to healthcare
   * Users looking for pharmacy recommendations
2. **Expected Outcome:**

o A functional AI-powered symptom checker that provides actionable health

recommendations.

**Phase-2: Requirement Analysis**

**Objective:**

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

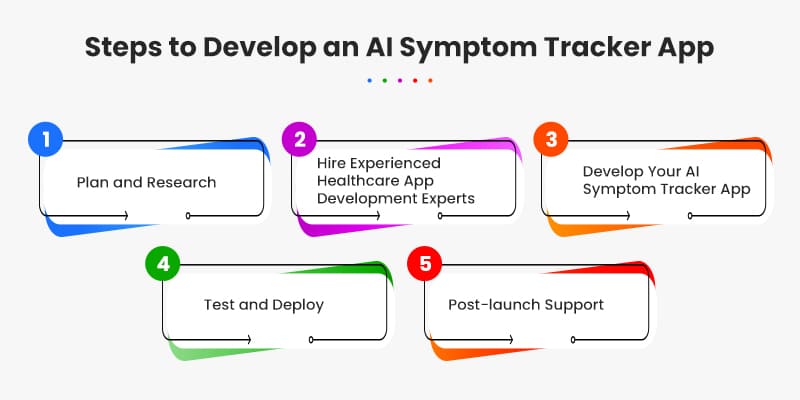
**Key Points:**

1. **Technical Requirements:**
   * Programming Language: **Python,JavaScript**
   * Backend: **Node.js,MongoDB,Express**
   * Frontend: **React.js ,Tailwind CSS,Axios**
   * Database: **Firebase Firestore**
2. **Functional Requirements:**
   * Accept user symptoms via text
   * Analyze symptoms using AI model
   * Provide medication suggestions & urgency assessment
   * Offer multilingual support for wider accessibility
3. **Constraints & Challenges:**
   * Ensuring accurate AI-generated medical suggestions
   * Complying with healthcare data regulations
   * Optimizing response time for real-time assistance

**Phase-3: Project Design**

**Objective:**

Develop the architecture and user flow of the application.



**Key Points:**

1. **System Architecture:**
   * User inputs symptoms (text)
   * AI model processes the query using google
   * Backend fetches medical data from verified sources
   * App provides diagnosis, urgency level, & treatment recommendations
2. **User Flow:**
   * Step 1: User describes symptoms (e.g., "I have a fever and headache")
   * Step 2: AI processes input and checks against medical databases
   * Step 3: The app suggests possible conditions, medications, and urgency level
3. **UI/UX Considerations:**
   * Minimalist, easy-to-use interface
   * Text-enabled input for accessibility
   * Dark & light mode for enhanced user experience

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

**Objective:**

Break down development tasks for efficient completion.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Task** | **Priority** | **Duration** | **Deadline** | **Assigned To** | **Dependencies** | **Expected Outcome** |
| Sprint 1 | Backend  Setup & API  Integration | 🔴 High | 6 hours (Day 1) | End of Day 1 | Harshitha | MongoDB  Node.js | API connection established & working |
| Sprint 1 | Frontend UI Development | 🟡 Medium | 3 hours (Day 1) | End of Day 1 | Shirisha | API response format finalized | Basic UI with Symptom input feilds |
| Sprint 2 | AI Symptom Analysis | 🔴 High | 4 hours (Day 2) | Mid-Day 2 | Keerthi | API response & processing | Symptom to condition mapping |
| Sprint 2 | Error Handling & Debugging | 🔴 High | 3 hours (Day 2) | Mid-Day 2 | Member 1&4 | API logs, UI inputs | Improved System Accurcy |
| Sprint 3 | Testing & UI Enhancements | 🟡 Medium | 3 hours (Day 2) | Mid-Day 2 | Srivalli | API response, UI layout | 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 |

**Sprint Planning with Priorities**

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

**🔴 High Priority – Set up backend & integrate Google PaLM API**

**🟡 Medium Priority – Develop basic UI with symptom input fields**

**Sprint 2 – Core Features & Debugging (Day 2)**

**🔴 High Priority – Implement AI symptom analysis**

**🔴 High Priority – Debug & handle API errors**

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

**🟡 Medium Priority – Test AI responses, refine UI**

**🟢 Low Priority – Final demo preparation & deployment**

**Phase-5: Project Development**

**Objective:**

Implement core features of the CareWise.

**Key Points:**

1. **Technology Stack Used:**
   * **Frontend:** React.js,Tailwind CSS,Axios
   * **Backend:**Node.js,Express,MongoDB
   * **Programming Language:** Python,javascript
2. **Development Process:**
   * Implement symptom input processing (text & voice)
   * AI model processes symptoms & fetches related medical insights
   * Generate treatment suggestions & urgency assessment
   * Optimize UI & API calls for faster responses
   * Challenges & Fixes:
3. **Challenges & Fixes:**
   * **Challenge**: Slow response time

**Fix**: Optimize API requests & implement caching

* + **Challenge**: Ensuring medical accuracy

**Fix**: Cross-check AI-generated insights with verified medical databases

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

**Objective:**

Ensure that the CareWise delivers accurate and work as expected.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Category** | **Test Scenario** | **Expected Outcome** | **Status** | **Tester** |
| TC-001 | Functional Testing | Query "Symptoms: Fever and Body Pain" | Correct diagnosis with medication suggestions | ✅ Passed | Harshitha |
| TC-002 | Functional Testing | Voice input for symptoms | Accurate symptom recognition | ✅ Passed | keerthi |
| TC-003 | Performance Testing | API response time under 500ms | Results within acceptable speed | ⚠ Needs Optimization | Srivalli |
| TC-004 | Bug Fixes & Improvements | Fixed incorrect AI generated responses. | Data accuracy should be improved. | ✅ Fixed | developer |
| TC-005 | Final Validation | Ensure UI is responsive across devices. | UI should work on mobile & desktop. | ❌ Failed – UI issues on mobile | Shirisha |
| TC-006 | Deployment Testing | Host app on vercel | App accessible. | 🚀 Deployed but Failed. | DevOps |

**Final Submission**

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