

# Hackathon Project Phases

**Project Title:**

*Carewise - AI-Powered Healthcare Assistance*

**Team Name:**

Codeze

**Team Members:**

- Shiva Sai

- Nihal

- Jitender

- Adnan

- Sheraz

## Phase-1: Brainstorming & Ideation

Develop an AI-powered healthcare assistant using Google Gemini Flash to provide users with health insights, symptom analysis, and medical guidance.

**Problem Statement:**

- Many people struggle to find reliable, real-time medical advice.
- Users need guidance on symptom analysis, medication suggestions, and doctor recommendations.

**Proposed Solution:**

- An AI-powered chatbot that provides symptom analysis, health tips, and doctor recommendations.
- Integration with trusted medical databases for accurate insights.
- It analyses users previous data to predict any health problems

**Target Users:**

- Individuals seeking quick medical advice.
- Patients looking for symptom evaluation.
- Users in need of general healthcare information.

**Expected Outcome:**

- A functional AI-powered healthcare assistant that provides real-time medical insights.
- Analyze prior and current symptoms to predict potential health issues.

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

**Technical Requirements:**

- Programming Language: Python
- Backend: Google Gemini Flash API
- Frontend: Streamlit Web Framework
- Database: SQLite3 for user interactions and history

**Functional Requirements:**

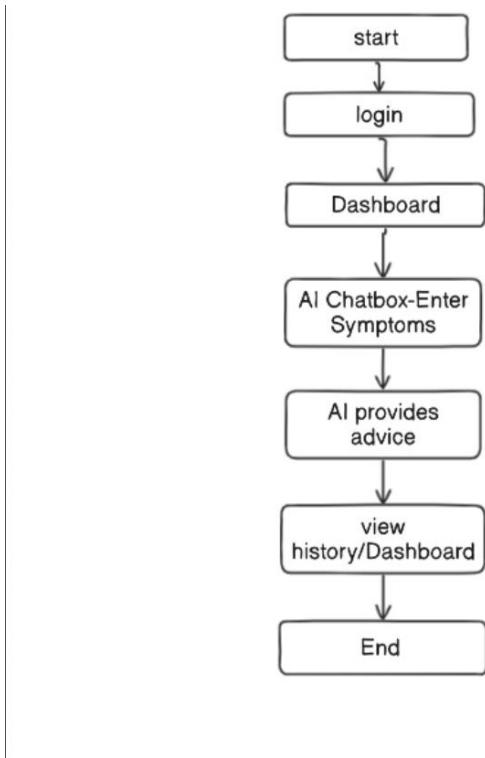
- AI-based symptom analysis and recommendations.
- User-friendly chatbot for healthcare queries.
- Secure and encrypted storage for user health data.

**Constraints & Challenges:**

- Ensuring accuracy in AI-driven medical responses.
- Managing API rate limits efficiently.
- Providing an intuitive and accessible UI .
- Analyzing user previous 10 days data

## Phase-3: Project Design

### Architectural Flow;



### System Architecture:

- User inputs symptoms into the chatbot UI.
- Query is processed using Google Gemini API.
- AI retrieves and processes medical data, including prior user inputs, to provide accurate predictions of potential health issues.
- The frontend displays personalized health insights.

### User Flow:

- Step 1: User enters a health-related query.
- Step 2: Backend calls the Gemini Flash API for data retrieval.
- Step 3: App processes the data and presents results.

## UI/UX Considerations:

- Clean and simple interface for ease of use.
- Accessibility features for visually impaired users.
- Dark and light mode options.

## Phase-4: Project Planning

Sprint	Task	Priority	Duration	Deadline	Assigned To	Dependencies	Expected Outcome
Sprint 1	Streamlit Environment	High	2 hrs	Mid-Day 1	Team	Python Installation	Working dev environment
Sprint 1	Setup & Dark Theme	Medium	1.5 hrs	Mid-Day 1	Adnan	Base UI Setup	Dark theme applied
Sprint 1	Gemini API Integration	High	3 hrs	EOD -1	Nihal	Google API Key	Functional API connection
Sprint 1	Database Schema Design	High	1 hr	Mid-Day 1	Shiva	SQLite Installation	ER Diagram & Tables
Sprint 1	Basic Login System	Medium	2 hrs	EOD 1	Adnan,Shiva	Streamlit Setup	OTP Auth Framework
Sprint 2	Medical Chat Interface	High	2.5 hrs	DAY-1	Nihal,Jithender	API Integration	Working chat with history
Sprint 2	Patient History Module	High	2 hrs	DAY-1	sheraz,shiva	DB Schema	Historical data retrieval
Sprint 2	Symptom Analysis Engine	High	3 hrs	DAY-1	Nihal,Jithender	Gemini API	Accurate medical responses
Sprint 2	Session Management	Medium	1.5 hrs	DAY-1	shiva	Login System	User state persistence
Sprint 3	Hospital Theme CSS	Medium	2 hrs	DAY-1	Adnan,Shiva	Base UI Setup	Professional medical UI
Sprint 3	Patient Profile Card	Medium	1.5 hrs	DAY-1	shiva	DB Integration	Interactive history display
Sprint 3	Error Handling System	High	2 hrs	EOD 1	Nihal,Jithender	All Components	Comprehensive error logs
Sprint 3	Security Audit	High	1.5 hrs	EOD 1	Adnan	Full System	HIPAA Compliance Check
Finalization	Performance Testing	Medium	1.5 hrs	EOD 1	Team	Complete Features	Optimized response times
Finalization	Deployment Packaging	Low	1 hr	DAY -2	sheraz,shiva	Testing Completion	Docker/EXE setup
Finalization	Documentation	Low	1 hr	DAY-2	sheraz,shiva	Final Code	User & Tech Docs

## Sprint Planning with Priorities

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

(● High Priority) Streamlit Environment

(● High Priority) Integrate Google Gemini API.

(○ Medium Priority) Setup & Dark Theme

(● High Priority) Database Schema Design

(○ Medium Priority) Basic Login System

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

(● High Priority) Medical Chat Interface

(● High Priority) Patient History Module

(● High Priority) Symptom Analysis Engine

(● Medium Priority) Session Management

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

(● Medium Priority) Hospital Theme CSS

(● Medium Priority) Patient Profile Card

(● High Priority) Error Handling System

(● High Priority) Security Audit

## Phase-5: Project Development

Implement core features of the Carewise app.

Technology Stack Used:

- Frontend: Streamlit
- Backend: Google Gemini Flash API
- Database: SQLite (optional)

Development Process:

- Implement API authentication and integration.
- Develop chatbot logic for healthcare queries.
- Optimize response generation for accuracy.

## Phase-6: Functional & Performance Testing

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional	User describes "migraine with visual aura"	Shows advice with OTC medications and warnings	<input checked="" type="checkbox"/> Passed	Shiva Sai
TC-002	Security	Enter invalid OTP during login	Access denied with error message	<input checked="" type="checkbox"/> Passed	Nihal
TC-003	Data Integrity	Add consultation record with valid symptoms	Record stored with proper blockchain hash	 Deployed	Jithender
TC-004	UI Testing	Verify patient history display after login	Recent symptoms tags show last 2 consultations	<input checked="" type="checkbox"/> Passed	Shiva
TC-005	Error Handling	Enter invalid phone number (9 digits) for OTP	Shows validation error message	<input checked="" type="checkbox"/> Passed	Sheraz
TC-006	Session management	Logout and try to access consultation page	Redirects to login screen	<input checked="" type="checkbox"/> Passed	Adnan

## **Final Submission**

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