**Hackathon Project Phases Template** for the **Auto Sage App** project.

# Hackathon Project Phases Template

**AI-Powered Code Generation using Code Llama**

**Team Name:**

Tourist with AI

**Team Members:**

1.K. BAVANI

2.K. SANDEEP

3.B. PRIYANKA

4. CH. GNANESHWAR

## Phase-1: Brainstorming & Ideation

**Objective:**

Creating an AI code generator for brainstorming and ideation involves building a tool that can assist users by generating creative ideas, concepts, or even solutions for various problems

**Key Points:**

**1.Define the problem:**

**Clarify what you’re trying to solve or achieve. A well-defined problem leads to more focused ideas.**

* **2.Encourage Wild Ideas:**

**No idea is too out there Sometimes the most unconventional or**

**Creative the idea, the better Don’t dismiss any thoughts at first; even wild**

**Ideas and find ways to build on them or combine them into something**

**New.**

**3.Focus on the “How” and “Why”:**

**Ask” How can this idea work? ”or” why might this idea be**

**Valuable?” This helps refine ideas and focus on practical application.**

## Phase-2: Requirement Analysis

**Objective:**

Define the system or application model including entities relationships and processes.

**Key Points:**

1. **Frameworks & Libraries: Hugging**
2. **Languages: Python**
3. **State Management: Redux, Zu -stand, Recoil**

1. **Functional Requirements:**

* 1. Ability to **fetch vehicle details** using Gemini Flash API.

○ Display **specifications, reviews, and comparisons** in an intuitive UI.

○ Provide **real-time vehicle maintenance tips** based on seasons.

○ Allow users to **search eco-friendly vehicles** based on emissions and incentives.

1. **Constraints & Challenges:**

* 1. Ensuring real-time updates from **Gemini API**.

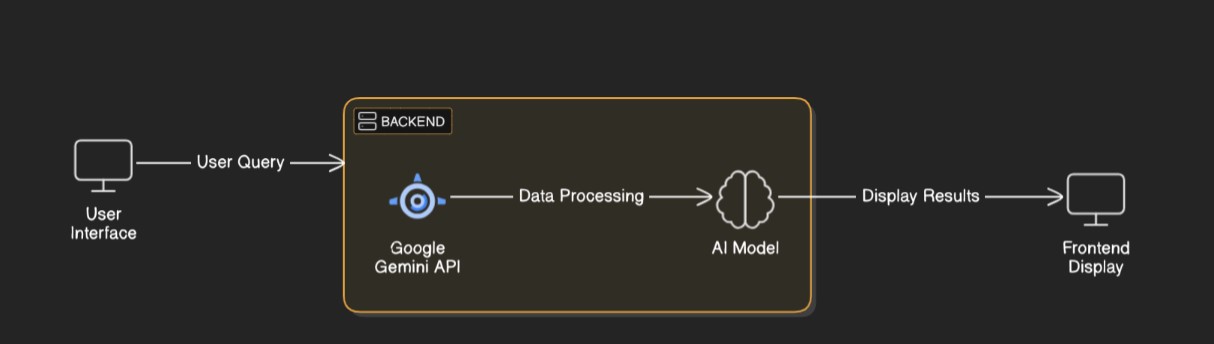
○ Handling **API rate limits** and optimizing API calls.

○ Providing a **smooth UI experience** with stream lite

## Phase-3: Project Design

**Objective:**

Develop an AI system that can automatically generate code based on human impact such as natural language using python.



**Key Points:**

1. **System Architecture:**

* 1. User enters vehicle-related query via UI.

○ Query is processed using **Google Gemini API**.

○ AI model fetches and processes the data.

○ The frontend displays **vehicle details, reviews, and comparisons**.

1. **User Flow:**

* 1. Step 1: User enters a query (e.g., "Best motorcycles under ₹1 lakh").

○ Step 2: The backend **calls the Gemini Flash API** to retrieve vehicle data.

○ Step 3: The app processes the data and **displays results** in an easy-to-read format.

1. **UI/UX Considerations:**

* 1. **Minimalist, user-friendly interface** for seamless navigation.

○ **Filters for price, mileage, and features**.

○ **Dark & light mode** for better user experience.

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

**Objective:**

**Enable developers to write software by simply describing the**

**Problem or functionality in natural language The AI will translate these descriptions**

**Into functioning code, drastically reducing the time spent writing code manually.**

## Phase-5: Project Development

**Objective:**

The main goal is to automate and enhance the software development process by generating

high-quality, efficient, and error-free code.

**Key Points:**

1. **Technology Stack Used:**

* 1. **Frontend:** Stream lite

○ **Backend:** Google Gemini Flash API

○ **Programming Language:** Python

1. **Development Process:**

* 1. Implement **API key authentication** and **Gemini API integration**.

○ Develop **vehicle comparison and maintenance tips logic**. ○ Optimize **search queries for performance and relevance**.

1. **Challenges & Fixes:**

* 1. **Challenge:** Delayed API response times.

**Fix:** Implement **caching** to store frequently queried results.

○ **Challenge:** Limited API calls per minute.

**Fix:** Optimize queries to fetch **only necessary data**.

## Phase-6: Functional & Performance Testing

**Objective:**

To ensure that the system operates correctly and efficiently Each type of testing

Serves a distinct purpose contributing to the AI-based code generation tool.