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

Hackathon Project Phases Template

Project Title:

AI Personalized Study Planner

Team Name:

Team Smart Plan Architects

Team Members:

- Hrithik. T
- Anush. K
- Joshita. B
- Lohith. S
- Shashidhar

Phase-1: Brainstorming & Ideation

Objective:

Develop an AI-powered personalized study planner that adapts to students' strengths, weaknesses, and learning styles using Google Gemini AI and Streamlit Web Framework.

Key Points:

1. **Problem Statement:**

AI Personalized Study Planner" is an intelligent application designed to create customized student study plans based on their specific goals, and preferences. Utilizing the Google Gemini AI, this tool helps students optimize their study schedules to achieve their academic targets efficiently.

2. **Proposed Solution:**

   - An AI-powered web application that creates customized study plans based on student input, performance tracking, and adaptive learning techniques.
   - The tool dynamically adjusts schedules, suggests study resources, and evaluates progress in real time.
   - **Key Features:**
     - AI-generated **dynamic study plans**.
     - **Real-time progress tracking** with performance reports.
     - **AI-based tutoring** assistance (Gemini API).
     - **Study material PDFs** and **AI-generated quiz** for assessments.

3. **Target Users:**

   - School and college students preparing for exams.
   - Competitive exam aspirants.
   - Anyone looking for structured, goal-oriented learning plans.

4. **Expected Outcome:**

   - A fully functional AI-powered study planner that provides personalized schedules, study materials, and progress tracking through a streamlit web application.

Phase-2: Requirement Analysis

Objective:

Define the technical and functional requirements of the AI Study Planner.

Key Points:

1. **Technical Requirements:**

   - Programming Language: **Python**

- Frontend: **Streamlit Web Framework**
- AI model: **Google Demini API**
- Data Handling**: Pandas, Matplotlib for visualisation**
- PDF Notes**: ReportLab Library**
- Database: **Not required initially (API-based queries)**

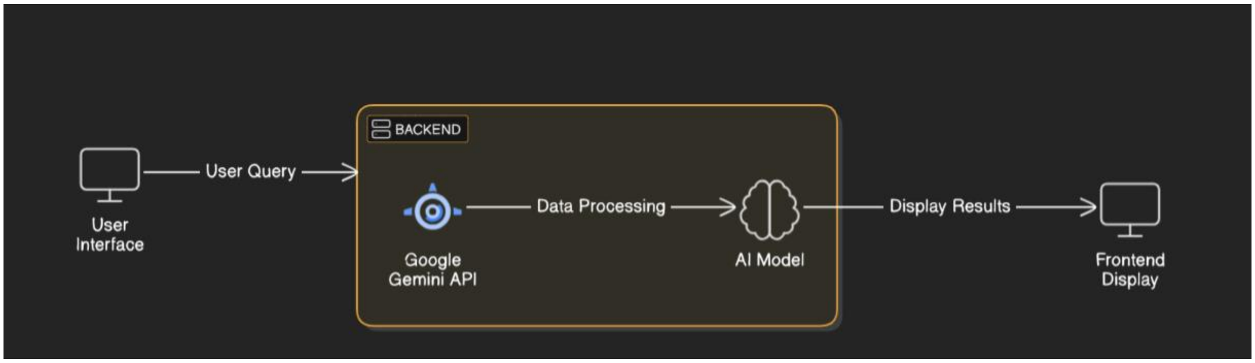
2. **Functional Requirements:**
   - Accept student data, including subjects, and time availability.
   - Generate a customised study schedule dynamically.
   - Track progress and completed tasks.
   - Provide AI-powered notes and quizzes.
   - Allow study materials in PDF Format.
   - Generate performance reports with visual graphs.

3. **Constraints & Challenges:**
   - Ensuring accurate AI-generated schedules.
   - Handling API rate limits efficiently.
   - Optimising real time performance tracking.

Phase-3: Project Design

Objective:

Develop the architecture and user flow of the application.



Key Points:

1.System Architecture:

User Input: Subjects, study hours, learning preferences.**AI Processing:** Google Gemini API generates study plans & quizzes.

Study Plan Generation: Dynamic schedule allocation based on user input.

Resource Recommendation: PDFs, AI-generated notes, quizzes.

Progress Tracking: Visual reports using **Matplotlib**.

AI Chatbot Assistance: Help with doubts and learning strategies.

1. **User Flow:**

- User enters study preferences.
- AI generates a personalized study plan.
- User accesses study materials and AI-generated notes.
- Progress tracking through checkbox-based task completion.
- AI-generated quizzes to assess knowledge.
- Performance report and final recommendations.

2. **UI/UX Considerations:**

    - Minimalist and intuitive interface for easy navigation.
    - Gamification elements to keep students motivated.
    - Multi-device compatibility (web and mobile).

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	Environment Setup & API Integration	🔴 High	6 hours (Day 1)	End of Day 1	Lohith	API keys, Firebase setup	API connection established
Sprint 1	UI Design & Prototyping	🟡 Medium	2 hours (Day 1)	End of Day 1	Joshita	Design tools ready	Basic UI prototype

Sprint 2	AI Study Plan Algorithm	🔴 High	3 hours (Day 2)	Mid-Day 2	Anush	User input data formatted	Personalized plans generated
Sprint 2	AI Chatbot Integration	🔴 High	1.5 hours (Day 2)	Mid-Day 2	Hrithik	OpenAI API working	AI tutoring assistant available
Sprint 3	Testing & Enhancements	🟡 Medium	1.5 hours (Day 2)	Mid-Day 2	Sashidhar	Functional prototype ready	Bug-free, optimized app
Sprint 3	Final Presentation & Deployment	🟢 Low	1 hour (Day 2)	End of Day 2	Entire Team	Completed project	Demo-ready submission

Sprint Planning with Priorities

Sprint 1 – Setup & Integration (Day 1)

(🔴 **High Priority)** Set up the **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 **search & comparison functionalities**.
(🔴 **High Priority)** Debug API issues & handle **errors in queries**.

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

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

Phase-5: Project Development

Objective:

Implement core features of the AI Study Planner.

Key Points:

1. **Technology Stack Used:**
   - **Frontend:** Streamlit
   - **Backend:** Python (Google Gemini API)
   - **Data Handling:** Pandas, Matplotlib
   - **PDF Generation:** ReportLab
   - **Programming Language:** Python
2. **Development Process:**
   - **Checkbox-based progress tracking** with **session state management.**
   - **AI Study Plan Generation** using **Google Gemini API**.
   - **AI-generated notes & quizzes** dynamically based on subjects.
   - **Matplotlib-based Performance Report** for study insights.
   - **PDF Notes Generation** using ReportLab.
3. **Challenges & Fixes:**
   - **Challenge:** Inaccurate AI-generated study plans.
     Fix: Refined Prompt Engineering and response filtering.
   - **Challenge:** Handling diverse student inputs.
     Fix: Implemented session state management for better tracking.

Phase-6: Functional & Performance Testing

Objective:

Ensure that the AI Study Planner works as expected.

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	User inputs study preferences	AI generates study plan	☑ Passed	Joshita
TC-002	AI adaptability	User skips a study session	AI adjusts plan dynamically	☑ Passed	Sashidhar

TC-003	Performance Testing	response time under 500ms	Fast API responses	⚠️ Needs Optimization	Lohith
TC-004	Bug Fixes & Improvements	Incorrect study plan suggestions	Accurate adjustments	☑️ Fixed	Hrithik
TC-005	UI Testing	Mobile responsiveness	Works on all devices	✖ Failed - UI broken on mobile	Anush
TC-006	Deployment Testing	Host the app	Web and mobile accessibility	🚀 Deployed	Entire Team

---

Final Submission

1. **Project Report: Documenting the entire development process.**
2. **Demo Video (3-5 Minutes)**
3. **GitHub/Code Repository Link**
4. **Presentation**