**Hackathon Project Phases Template** for the **AI Study Planner** project.

Hackathon Project Phases Template

# Project Title:

**Ai Study Planner**

# Team Name:

Uppari Bharath Team

# Team Members:

* Member 1: Surishetty Padmavathi

# Phase-1: Brainstorming & Ideation

## Objective:

The main objective of an AI study planner project is to create a system that uses artificial intelligence to help students plan and organize their study schedules more effectively. The AI study planner aims to optimize the learning process by providing personalized recommendations and insights based on various factors.

## Key Points:

1. **Problem Statement:**
   * Students often struggle with managing their study schedules effectively due to various factors, including time constraints, procrastination, lack of personalized guidance, and difficulties in balancing multiple subjects and assignments. Traditional methods of studying, such as fixed timetables or generic study plans, fail to account for individual learning preferences, productivity patterns, and academic needs, leading to suboptimal study outcomes and increased stress.
2. **Proposed Solution:**
   * An AI-powered study planner that creates personalized study schedules based on individual learning preferences, deadlines, and progress. It uses machine learning to optimize time management, prioritize tasks, track progress, and offer motivational support.
3. **Target Users:**
   * **Students** (High school, college, university)
   * **Self-learners** (Independent or online learners)
   * Part-time students/working professionals
   * **Tutors/educators** (to guide students)
4. **Expected Outcome:**

 **Improved Time Management**: Students will be able to better manage their study time, reducing procrastination and ensuring all tasks and deadlines are met on time.

 **Increased Productivity**: By following a personalized study plan, students will focus on the most important tasks, leading to more efficient learning and better use of time.

 **Reduced Stress and Burnout**: The AI will help students maintain a balanced schedule with proper breaks, minimizing the risk of stress and burnout during intense study periods.

 **Better Academic Performance**: With customized schedules and task prioritization, students will be able to focus on weaker subjects, resulting in improved grades and overall academic success.

 **Enhanced Motivation**: Regular progress tracking, feedback, and motivational reminders will keep students engaged, encouraging them to stay on track and remain committed to their study goals.

 **Data-Driven Insights**: Students will receive valuable insights into their study habits and performance, helping them identify areas for improvement and refine their approach to learning.

# Phase-2: Requirement Analysis

## Objective:

Define the technical and functional requirements for the AutoSage App.

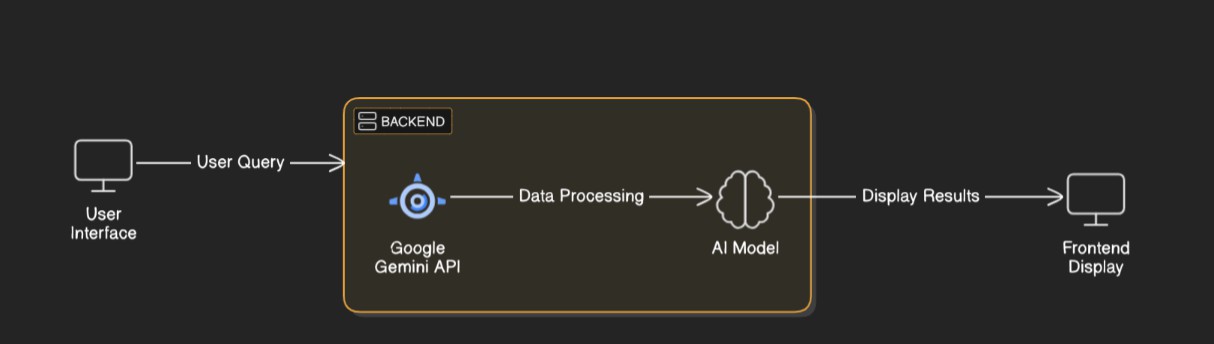
## Key Points:

1. **Technical Requirements:**
   * Programming Language: **Python**
   * Backend: **Google Gemini Flash API**
   * Frontend: **Streamlit Web Framework**
   * Database: **Not required initially (API-based queries)**
2. **Functional Requirements:**
   * **User Profiles**: Personalized account creation and management.
   * **Study Plan Creation**: AI-generated, customizable study schedules.
   * **Task Management**: Input tasks, deadlines, and automated prioritization.
   * **Time Management**: Optimized study sessions with break scheduling.
3. **Constraints & Challenges:**
   * **Data Privacy & Security**: Ensuring user data is protected and complies with privacy laws.
   * **Personalization Accuracy**: Collecting enough accurate data for effective study plan recommendations.
   * **User Engagement**: Keeping users motivated and engaged over time.
   * **Adaptability**: Catering to diverse learning styles and preferences effectively.
   * **System Scalability**: Managing increased traffic during peak times.
   * **Task Prioritization**: Accurately predicting task durations and balancing schedules.
   * **User Adoption**: Convincing users to trust and use the AI planner.
   * **Resource Constraints**: Balancing time, budget, and technical resources for development.

# Phase-3: Project Design

## Objective:

Develop the architecture and user flow of the application.



## Key Points:

1. **System Architecture:**
   * **Frontend**: Responsive UI (React/React Native) for web/mobile.
   * **Backend**: Scalable server (Node.js/Django), and APIs.
   * **AI/ML**: Personalization engine for study plans, task prioritization, and time management.
2. **User Flow:**

Step 1: **Registration/Login**:

* Sign up/login via email/Google/Facebook.

Step 2 **Profile Setup**:

* Enter study preferences, goals, and sync calendar.

Step 3: **AI Study Plan Creation**:

* AI generates a personalized study plan based on user inputs.
* View/modify study plan, add tasks, and adjust priorities.
* Start study sessions, track time, and take breaks.
* View progress reports and receive feedback.

1. **UI/UX Considerations:**
   * Simple, intuitive design with easy navigation.
   * Visual progress charts and task management.

# 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 | Research & Design | 🔴 High | 6 hours  (Day 1) | End of Day 1 | Member 1 | Define technical architecture (backend, database, APIs) | Conduct research, and define system architecture and UI/UX design. |
| Sprint 2 | Core Development - Backend | 🟡  Medium | 2 hours  (Day 1) | End of Day 1 | Member 2 | Set up backend architecture | Start developing the core backend functionalitie |
| Sprint 3 | Frontend Development | 🔴 High | 3 hours  (Day 2) | Mid-Day 2 | Member 1& 2 | Develop UI for user profiles, study plan view, task management | Develop the frontend interface with key features. |
| Sprint 4 | AI Model Refinement & Integration | 🔴 High | 1.5 hours  (Day 2) | Mid-Day 2 | Member 1 | API logs, UI inputs | Refine AI models and integrate them into the planner. |
| Sprint 5 | Integration & Testing | 🟡  Medium | 1.5 hours  (Day 2) | Mid-Day 2 | Member 2 | Run usability tests and fix any bugs | Test full system functionality and integration. |
| Sprint 6 | User Feedback & Refinement | 🟢 Low | 1 hour  (Day 2) | End of Day 2 | Entire Team | Finalize user documentation and support resources | Gather feedback from initial users and refine the system. |

# Phase-5: Project Development

## Objective:

Implement core features of the AutoSage App.

## Key Points:

1. **Technology Stack Used:**
   * **Frontend:** Streamlit
   * **Backend:** Google Gemini Flash API
   * **Programming Language:** Python
2. **Development Process:**

 **equirement Gathering**:

* Define goals, features, and target users.

 **Design & Architecture**:

* Create system architecture, database design, and UI/UX wireframes.

 **Frontend Development**:

* Build user interface (React/React Native) for web and mobile.

 **Backend Development**:

* Develop server-side logic (Node.js/Django), APIs, and database integration.

 **AI/ML Integration**:

* Implement AI algorithms for personalized study plans and task prioritization.

 **External Integrations**:

* Integrate with calendars and task management tools.

 **Testing**:

* Perform unit, integration, and user testing to ensure functionality.

**Monitoring & Maintenance**:

* Monitor app performance and provide ongoing updates based on user feedback.

1. **Challenges and Fixes:**
   * **Challenge:** Achieving accurate AI-based recommendations for study plans based on diverse user preferences.

**Fix:** Continuously trained AI models using real-time user data, incorporated user feedback for better adjustments, and improved task prioritization algorithms.

* + **Challenge:** Keeping users engaged and motivated over long study periods.

**Fix:** Implemented gamification features, such as progress badges, task completion rewards, and regular motivational messages to keep users interested.

# Phase-6: Functional & Performance Testing

## Objective:

Ensure that the AutoSage App works as expected.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Category** | **Test Scenario** | **Expected Outcome** | **Status** | **Tester** |
| TC-001 | Functional Testing | User registration via Google account | User should be successfully registered and logged in. | ✅ Passed | Tester 1 |
| TC-002 | Functional Testing | AI generates a personalized study plan | A valid, personalized study plan should be displayed. | ✅ Passed | Tester 2 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TC-003 | Performance Testing | API response time for study plan generation under 500ms | API should return study plan within 500ms | ⚠ Needs Optimization | Tester 3 |
| TC-004 | Bug Fixes & Improvements | Fixing incorrect task prioritization in study plans. | Tasks should be prioritized correctly based on deadlines and importance. | ✅ Fixed | Develop er |
| TC-005 | Final Validation | Ensure UI responsiveness across mobile and desktop | UI should be fully responsive and work 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 and function without errors. | 🚀 Deployed | DevOps |

# Final Submission

1. **Project Report Based on the templates**
2. **GitHub/Code Repository Link**
3. **Presentation**