# Hackathon Project Phases Template

## Project Title:

AI-Powered Study Planner

## Team Name:

The Code Crushers

## Team Members:

K. Saishruthi  
N. Archana  
N. Vaishnavi  
P. Swapnika Srivally  
D. Krishna Hasitha

## Institution:

Vasavi College of Engineering

## Phase-1: Brainstorming & Ideation

### Objective:

Develop an AI-powered study planner that generates a structured study timetable based on subjects, available study hours, and topic complexity.

### Problem Statement:

Students often struggle with effective time management, structured study planning, and optimizing their learning schedules. A lack of tools that generate personalized study schedules based on topics and available time makes it difficult for learners to efficiently manage their studies.

### Proposed Solution:

Our solution is an AI-powered study planner that dynamically generates a structured study timetable based on subjects, available study hours, and topic complexity. The AI model (Mistral-7B via Hugging Face) ensures efficient distribution of topics across the days specified by the user.

### Target Users:

1. Students preparing for exams.  
2. Self-learners planning structured courses.  
3. Professionals looking to upskill.  
4. Teachers assisting students with study plans.

### Expected Outcome:

A functional AI-powered study planner application that generates personalized study schedules based on user inputs.

## Phase-2: Requirement Analysis

### Technical Requirements:

• Programming Language: TypeScript  
• Frontend Framework: Next.js  
• Backend: Hugging Face API (Mistral-7B)  
• Database: Not required initially (API-based queries)

### Functional Requirements:

• Users can input subjects and available study hours.  
• AI generates a structured study plan, allocating time per subject.  
• Users can view, edit, and modify study plans based on preference.

### Constraints & Challenges:

• Ensuring real-time responses from the Hugging Face API.  
• Handling API rate limits and optimizing API calls.  
• Providing a seamless UI experience with Next.js.

## Phase-3: Project Design

### System Architecture:

1. User enters subjects and hours via the UI.  
2. Query is processed using Hugging Face API.  
3. AI model generates a structured study plan.  
4. The frontend displays the study plan in a tabular format.

### User Flow:

Step 1: User enters topics and selects the number of days.  
Step 2: The backend calls the Hugging Face API to generate the study plan.  
Step 3: The app processes the data and displays the study plan in a user-friendly format.

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint | Task | Priority | Duration | Deadline | Assigned To | Expected Outcome |
| Sprint 1 | Environment Setup & API Integration | High | 6 hours | End of Day 1 | K. Saishruthi | API connection established & working |
| Sprint 2 | Frontend UI Development | Medium | 2 hours | End of Day 1 | N. Archana | Basic UI with input fields |
| Sprint 3 | Study Plan Generation | High | 3 hours | End of day 1 | N. Vaishnavi | Study plan generation functionality |
| Sprint 4 | Error Handling & Debugging | High | 1.5 hours | Mid-Day 2 | P. Swapnika Srivally & D. Krishna Hasitha | Improved API stability |
| Sprint 5 | Testing & UI Enhancements | Medium | 1.5 hours | Mid-Day 2 | N. Archana | Responsive UI, better user experience |
| Sprint 6 | Final Presentation & Deployment | Low | 1 hour | End of Day 2 | Entire Team | Demo-ready project |

## Phase-5: Project Development

### Technology Stack Used:

• Frontend: Next.js  
• Backend: Hugging Face API (Mistral-7B)  
• Programming Language: TypeScript

## Phase-6: Functional & Performance Testing

Basic testing was performed to verify study plan generation and UI responsiveness. Further optimizations and deployment testing will be conducted later.