**Project Title:**  
Study Bud: AI Study Planner

**Team Name:**  
MAPV

**Team Members:**

* Chereddy Mahigeethika
* Akkisetti Amulya
* Banothu Priyanka
* Battu Vagdevi

**Phase-1: Brainstorming & Ideation**

**Objective:**  
Develop an AI-powered study planner that helps students create personalized schedules, optimize study time, and enhance productivity using AI.

**Key Points:**

**Problem Statement:**

* Students struggle with managing study schedules efficiently.
* Lack of personalized recommendations for study patterns based on learning habits.
* Difficulty in tracking progress and staying motivated.

**Proposed Solution:**

* An AI-based study planner that generates optimized schedules based on subjects, priorities, and available time.
* Adaptive recommendations based on student performance and feedback.
* Features like smart reminders, productivity analysis, and progress tracking.

**Target Users:**

* School and college students.
* Competitive exam aspirants.
* Self-learners looking for structured study plans.

**Expected Outcome:**

* A functional AI-powered study planner that helps students optimize their study time and track progress effectively.

**Phase-2: Requirement Analysis**

**Objective:**  
Define the technical and functional requirements for StudyBud.

**Key Points:**

**Technical Requirements:**

* **Programming Language:** Python
* **Backend:** Flask/FastAPI
* **AI Model:** NLP-based study recommendation system
* **Frontend:** React/Streamlit
* **Database:** Firebase/PostgreSQL

**Functional Requirements:**

* User authentication and profile setup.
* Input-based study plan generation (time availability, subjects, goals).
* AI-powered adaptive scheduling and reminders.
* Performance tracking and analytics.

**Constraints & Challenges:**

* Efficient AI model for personalized scheduling.
* Handling large amounts of user data securely.
* Ensuring user engagement with smart features.

**Phase-3: Project Design**

**Objective:**  
Develop the architecture and user flow of the application.

**Key Points:**

**System Architecture:**

1. User inputs subjects, priorities, and study hours.
2. AI model processes data and generates an optimized study plan.
3. Backend stores progress and updates schedules dynamically.
4. Frontend displays study plan, reminders, and progress analytics.

**User Flow:**

* **Step 1:** User sets up profile and study goals.
* **Step 2:** AI generates a personalized study schedule.
* **Step 3:** User receives reminders and adjusts schedules dynamically.
* **Step 4:** Performance tracking dashboard provides insights.

**UI/UX Considerations:**

* Clean, intuitive dashboard.
* Dark & light mode for comfort.
* Drag-and-drop study schedule modifications.

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

| **Sprint** | **Task** | **Priority** | **Duration** | **Deadline** | **Assigned To** | **Dependencies** | **Expected Outcome** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Sprint 1 | Environment Setup & API Integration | 🔴 High | 6 hours | End of Day 1 | Member 1 | Python, React, Database Setup | API connection established & working |
| Sprint 1 | UI Development | 🟡 Medium | 4 hours | End of Day 1 | Member 2 | API response format finalized | Basic UI with input fields |
| Sprint 2 | AI-based Study Plan Generation | 🔴 High | 5 hours | Mid-Day 2 | Member 3 | ML model, API response | Study plan generated using AI |
| Sprint 2 | Error Handling & Debugging | 🔴 High | 2 hours | Mid-Day 2 | Member 4 | API logs, UI inputs | Improved API stability |
| Sprint 3 | Testing & UI Enhancements | 🟡 Medium | 2 hours | Mid-Day 2 | Member 2 & 3 | API response, UI layout completed | Responsive UI, better user experience |
| Sprint 3 | Final Presentation & Deployment | 🟢 Low | 1 hour | End of Day 2 | Entire Team | Working prototype | Demo-ready project |

**Phase-5: Project Development**

**Objective:**  
Implement core features of StudyBud.

**Key Points:**

**Technology Stack Used:**

* **Frontend:** React/Streamlit
* **Backend:** Flask/FastAPI
* **Database:** Firebase/PostgreSQL
* **AI:** NLP for schedule optimization

**Development Process:**

* Implement AI-based schedule optimization.
* Develop UI components for input and output.
* Set up database storage for user progress tracking.

**Challenges & Fixes:**

* **Challenge:** Slow AI model predictions.  
  **Fix:** Optimize model inference using caching.
* **Challenge:** Low user engagement.  
  **Fix:** Implement gamification (badges, streaks, leaderboards).

**Phase-6: Functional & Performance Testing**

| **Test Case ID** | **Category** | **Test Scenario** | **Expected Outcome** | **Status** | **Tester** |
| --- | --- | --- | --- | --- | --- |
| TC-001 | Functional | Generate study plan for given inputs | Optimized schedule displayed | ✅ Passed | Tester 1 |
| TC-002 | Functional | Modify study plan dynamically | Updated plan should reflect | ✅ Passed | Tester 2 |
| TC-003 | Performance | API response time under 500ms | Fast schedule generation | ⚠ Needs Optimization | Tester 3 |
| TC-004 | Bug Fixes | Fix incorrect study plan calculations | Accurate schedules provided | ✅ Fixed | Developer |
| TC-005 | UI Testing | Ensure responsive UI | Works across devices | ❌ Failed - UI broken on mobile | Tester 2 |
| TC-006 | Deployment | Host the app online | Accessible via web | 🚀 Deployed | DevOps |