**Project Plan: Auto-do-list**

**1. Project Overview**

**Goal:** Develop an AI-powered **Auto-do-list** application with a graphical user interface (GUI) to assist users in managing their schedules efficiently. The application will allow users to manually input fixed schedules, add tasks with deadlines, and automate time allocation.

**2. Timeline**

**Estimated project duration:** 10 Weeks  
(Based on part-time development)

**3. Milestones & Tasks**

**Milestone 1: Project Setup & Initial Planning**

* **Task 1.1: Requirement analysis**  
  **Time:** Mar 3 - Mar 4  
  **Deliverable:** Project Requirements Document
* **Task 1.2: Technology selection and design**  
  **Time:** Mar 4- Mar 5  
  **Deliverable:** Design document with chosen frameworks (Tkinter for GUI, Python for backend logic)

**Milestone 2: Basic GUI Implementation**

* **Task 2.1: Learn GUI technique (Tkinter)**  
  **Time:** Mar 6 - Mar 21  
  **Deliverable:** Sample Tkinter interface and components
* **Task 2.2: Task input interface**  
  **Time:** Mar 6 - Mar 21  
  **Deliverable:** GUI for users to input tasks
* **Task 2.3: Fixed schedule input**  
  **Time:** Mar 24 - Apr 7  
  **Deliverable:** Interface for setting recurring schedules
* **Task 2.4: Schedule display interface**  
  **Time:** Mar 6 - Mar 21  
  **Deliverable:** GUI displaying planned schedules

**Milestone 3: Core Functionality Implementation**

* **Task 3.1: Implement schedule logic**  
  **Time:** Mar 9 - Apr 6  
  **Deliverable:** Backend logic to manage task prioritization and scheduling

**Milestone 4: Testing & Refinement**

* **Task 4.1: Testing & Refinement**  
  **Time:** Mar24 - Apr 18  
  **Deliverable:** Fully functional GUI and verified scheduling logic accuracy

**Milestone 5: Finalization & Completion**

* **Task 5.1: Completed project**  
  **Time:** Apr 28  
  **Deliverable:** Final software package

**4. Deliverables**

1. **Project Requirements Document** – Detailed app functionality and design.
2. **Source Code** – Full project codebase.
3. **GUI Components** – Implemented Tkinter-based interface.
4. **Functional Software** – Working Auto-do-list application.
5. **User & Developer Documentation** – Guides for users and developers.
6. **Final Project Report** – Summary of development, challenges, and solutions.

**5. Risks and Mitigation Plans**

* **Risk:** Difficulty in designing an intuitive GUI  
  **Mitigation:** Regular user feedback and iterative improvements
* **Risk:** Scheduling logic complexity  
  **Mitigation:** Start with simple algorithms and optimize over time
* **Risk:** Limited time for testing  
  **Mitigation:** Allocate buffer time in schedule

1. **Project Schedule Overview**

See the Gantt Chart