Project Plan Document (The Reminder)

# Project Scope

This project will include:

* Task addition, deletion, updating(editing), saving
* Accessibility Profile Management
* Task Prioritization
* Reminder type selection per task
* Android mobile application
* Security

Will not include:

* Calendar Synchronization
* Desktop or IOS Versions
* Cross user task sharing
* Task Protection

# Project Organization

| **Team Member** | **Role** | **Responsibilities** |
| --- | --- | --- |
| Beyzanur Zeybek | **Project Coordinator** | Manages project timeline, ensures task tracking, handles documentation deadlines |
| Taha Mert Ağım and Feyza Coşkun | **Frontend Developer** | Builds UI screens in Flutter, handles navigation, accessibility styling |
| Beyzanur Zeybek and Talha Akbulut | **Backend Developer** | Implements task logic, reminder system, local storage (SQLite), pattern logic |
| Feyza Coşkun and Özge Doğan | **UI/UX Designer** | Designs wireframes, accessibility settings, visual consistency |
| All team members | **Tester & QA Lead** | Writes and executes acceptance tests, reports bugs, verifies accessibility flows |

# Objectives

Defined Role Distribution:

Each team member will be assigned a specific role (e.g., frontend, backend, design, testing) to ensure a balanced workload. Even though everyone has a specific role, there is a possibility to work with rotation. Responsibilities will be tracked using GitHub commits and a shared task board.

Timely Delivery of Project Phases:

The team will follow a project timeline that includes all major phases such as requirements gathering, design, development, testing, and deployment. Regular check-ins will help ensure deadlines are met.

Clear Documentation and Traceability:

All project-related decisions and progress will be documented and version-controlled. Each deliverable will include clear task ownership.

Collaborative Communication:

Weekly team meetings and instant messaging platforms will be used to coordinate tasks and resolve any issues. Important decisions will be recorded for future reference.

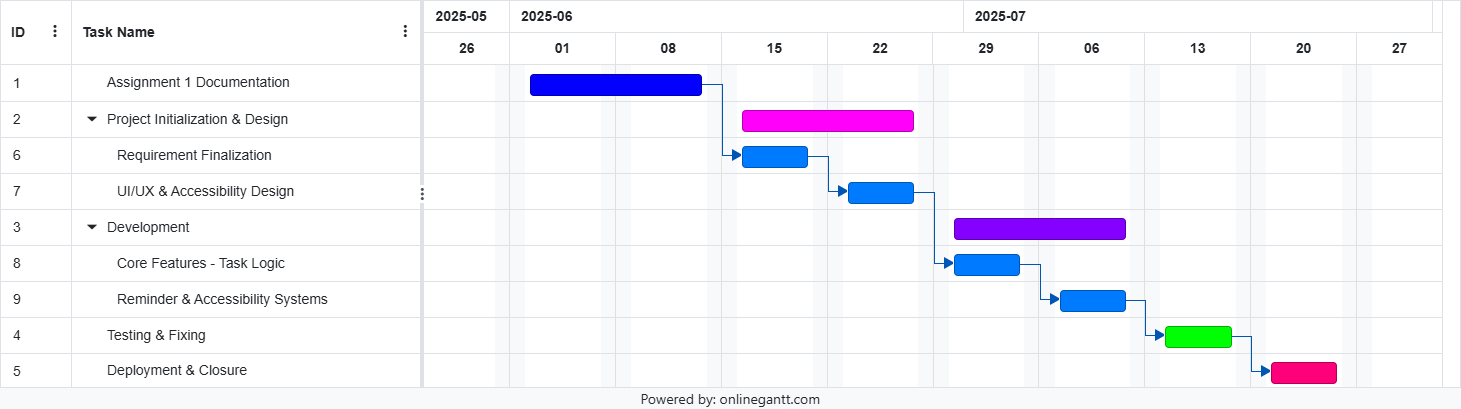
Awareness of Project Risks:

Potential risks will be considered throughout the project. Detailed strategies for risk mitigation will be provided in the Risk Management section.

Quality and Testing Focus:

The team will apply continuous testing and peer reviews to ensure the application remains stable, user-friendly, and adheres to proper design pattern implementation.

# Key Phases and Timeline (Simple Gantt Chart or List)



**Assignment 1 Documentation:** The team will prepare key documents including the Project Definition, Project Plan, Software Requirements, Architecture Overview, and Pattern–Use Case Mapping. The initial GitHub repository setup will also be completed.

**Project Initialization & Design:** The team will plan its internal workflow and define initial UI/UX goals with a focus on accessibility. Design choices will consider user needs from the start.

**Requirement Finalization:** Use cases and acceptance criteria will be finalized. Accessibility and reminder-related needs will be clearly mapped to functional requirements.

**UI/UX & Accessibility Design:** Wireframes and mockups will be created, covering various accessibility profiles. Strategy Pattern will guide the dynamic selection of UI behaviors.

**Development:** Main features will be implemented using Flutter. Reminder systems and accessibility logic will be integrated, leveraging selected design patterns.

**Core Features – Task Logic:** CRUD operations for tasks and local data storage (SQLite) will be developed. Shared logic like task handling may be managed via Singleton.

**Reminder & Accessibility Systems:** Reminder behaviors (audio, visual, vibration) will be implemented via Strategy Pattern. Reminder triggering will follow the Observer Pattern. Accessibility behavior switching will also use Strategy.

**Testing & Fixing:** Manual and accessibility-focused testing will be done across devices. Bugs will be fixed and UI refined. Strategy and Observer implementations will be verified.

**Deployment & Closure:** The APK will be generated, final documentation completed, and the project will be submitted. The team will present the app and evaluate final feedback.

# Resource Planning

Resources, suitable for application development (Android), are to be used

* **Development Environment:** **Android Studio** will be used for developing the Android application. All team members have access to this IDE on their local machines.
* **Programming and UI Framework:** The UI and application logic will be developed using **Flutter**, an open-source SDK supported by Google. It is freely available and easy to set up.
* **Database:** **SQLite** will be used for local data storage within the application. It is lightweight and well-integrated with Flutter for handling offline data.
* **Version Control and Collaboration:** **GitHub** will be used for code sharing and task tracking. All team members will contribute to a shared repository.
* **Testing Devices:** Both **Android emulators** and real Android devices will be used during the development and testing phases to ensure proper functionality across environments.

These resources are sufficient for reaching the project goals and are freely accessible to all team members

# Risk Management

Throughout the development of *TheReminder*, several potential risks have been identified. The team has prepared mitigation strategies to minimize their impact and ensure steady progress:

| **Risk** | **Likelihood** | **Impact** | **Mitigation Strategy** |
| --- | --- | --- | --- |
| **Team member becomes unavailable** | Medium | High | Distribute tasks with some overlap; maintain shared documentation so others can take over if needed. |
| **Reminder features not working reliably** | Medium | High | Test on multiple Android devices early; isolate platform-specific logic for easier debugging. |
| **Miscommunication between team members** | Medium | Medium | Weekly online meetings and task tracking through a shared Kanban board (e.g., Trello or GitHub Projects). |
| **Difficulty integrating accessibility features** | Low | Medium | Research accessibility plugins early; test small components as soon as they are built. |
| **Missed deadline or scope creep** | Medium | High | Stick to original scope; track progress weekly and hold short status meetings. |

# Communication Plan

Effective communication is essential for the successful coordination of tasks and timely progress in the project. The team will follow a structured but flexible communication strategy:

* **Weekly Meetings**: The team will hold scheduled online meetings once a week to discuss progress, assign tasks, and resolve blockers. The day and time will be agreed upon based on everyone's availability.
* **Instant Messaging**: Day-to-day communication will take place through a shared messaging platform (WhatsApp) to ensure quick feedback and real-time collaboration.
* **Decision Logging**: Key decisions and changes will be documented in a shared document(on Github) to ensure traceability and alignment among all members.

# Change Management Plan

To keep the project organized and avoid confusion, the team will follow a simple change management process. If there’s a proposed change to the requirements or scope (like adding a new feature or changing a reminder behavior), it will go through these steps:

1. **Request:** A team member creates a short written proposal explaining the change and why it’s needed.
2. **Review:** The team discusses the impact of the change during a weekly meeting, focusing on feasibility, time, and design pattern implications.
3. **Decision:** The team votes or agrees by consensus. If the change is approved, it's added to the task board with updated timelines.
4. **Documentation:** The change is logged in a shared “Change Log” document in the GitHub repo, with the date, decision, and responsible person.

This lightweight process will help us stay flexible while keeping track of all important changes throughout the project.

# Budget Plan

Since this is a student project and we don’t have real financial costs, our “budget” mainly refers to how we plan and distribute time and team effort.

* **Effort Allocation:** Effort will be distributed according to project phases. For example:  
  + 15% on planning and documentation
  + 30% on core feature development
  + 20% on accessibility and reminders
  + 20% on testing and bug fixing
  + 15% on finalization and deployment
* **Time Allocation:** Tasks are assigned weekly and team members are expected to put in regular hours based on sprint goals.
* **Tools and Resources:** We’ll use open-source tools (Flutter, GitHub, etc.), so there’s no direct cost. Team time and coordination are our main resources.

This plan helps us keep things fair and focused, making sure no part of the project is overlooked.

# 

# 9. Acceptance Tests and Acceptance Criteria

To make sure the application meets both functional and accessibility requirements, the team will define and run a set of acceptance tests during the final weeks of development. These tests will help verify that the app behaves as expected and is usable by all types of users.

**General Acceptance Criteria:**

* The user can create, update, delete, and complete tasks without any issues.
* Reminder types (audio, vibration, visual) work correctly and trigger at the correct time.
* Accessibility preferences can be selected and applied successfully.
* The app is stable and responsive across different devices.

**Accessibility Acceptance Criteria:**

* High contrast mode adjusts the UI colors properly.
* Font size preferences are applied across all views.
* Reminder types are distinct and easily recognizable.
* The app remains fully usable for users with visual or hearing limitations.
* The interface supports screen reader navigation where applicable.

**Reminder Feature Tests:**

* Saving a task with a reminder successfully schedules it.
* Switching between reminder modes updates the behavior without needing to recreate the task.
* The app can handle multiple reminders without conflict.

**Test Execution Plan:**

* Tests will be performed manually on both emulators and physical Android devices.
* A shared test checklist will be used to track pass/fail results.
* Acceptance tests will be carried out mainly in Weeks 7 and 8 of the project timeline.

# Task Matrix

| **Task** | **Responsible** |
| --- | --- |
| -Project Scope  -Resource Planning  -Risk Management  -Communication Plan | Taha Mert Ağım |
| -Project Organization  -Objectives  -Key Phases and Timeline  -Change Management Plan & Budget Plan  -Acceptance Tests and Acceptance Criteria | Feyza Coşkun |