## **Requirements Analysis for Todo App:**

## **Non-Functional Requirements**

1. **Usability**:
   * The application must provide an intuitive user interface, enabling users to learn basic features within 30 minutes of use.
2. **Performance**:
   * The app must load all dashboards and task lists within 3 seconds, even with 500 tasks logged.
3. **Scalability**:
   * The system should support at least 1,000 active users collaborating in real-time without any degradation in performance.
4. **Reliability**:
   * The app must have 99.9% uptime, ensuring users can always access their tasks and schedules.
5. **Security**:
   * User data, including tasks and deadlines, must be encrypted during transmission and storage.

## **Functional Requirements**

1. **Task Creation and Management**:
   * The system must allow users to create, edit, and delete tasks with descriptions, priority levels, and due dates.
2. **Notifications and Reminders**:
   * Users should be able to receive task reminders as push notifications or emails, with the ability to snooze or dismiss them.
3. **Calendar Integration**:
   * The app must integrate with external calendars (e.g., Google Calendar) to sync task deadlines and events.
4. **Collaboration**:
   * Users should be able to assign tasks to team members and track task progress collaboratively.
5. **Search and Filter**:
   * The app must provide search functionality with filters, allowing users to search for tasks by project, tag, or due date.

## **Formal Use Cases**

### **Use Case 1: Create a New Task**

**Actors**: User  
**Preconditions**: User must be logged into the app.  
**Steps**:

1. User clicks on the “Add Task” button.
2. A pop-up form appears, prompting the user to enter task details.
3. User provides a task name, description, priority, due date, and optional tags.
4. User clicks “Save,” and the task is added to the relevant project list.

**Postconditions**: The task is visible on the dashboard.

### **Use Case 2: Set Task Reminders and Notifications**

**Actors**: User  
**Preconditions**: A task with a due date exists.  
**Steps**:

1. User selects a task and clicks “Set Reminder.”
2. A menu appears with options to choose reminder intervals.
3. User sets reminders to trigger 1 day and 1 hour before the deadline.
4. User saves the settings.

**Postconditions**: The reminders are scheduled and displayed at the specified times.

### **Use Case 3: Collaborate on Tasks**

**Actors**: User, Team Member  
**Preconditions**: The user must have team members added.  
**Steps**:

1. User selects a task and clicks the “Assign to Member” button.
2. User picks a team member from the list.
3. The task is shared, and both users can view and update its progress.

**Postconditions**: The task shows both users as collaborators.

### **Use Case 4: Search for a Task**

**Actors**: User  
**Preconditions**: Tasks must exist in the system.  
**Steps**:

1. User clicks on the search bar and types a keyword or tag.
2. The system displays matching tasks in real-time.
3. User applies filters (e.g., by due date or project) to refine results.

**Postconditions**: The search results are displayed as per the user’s query.

### **Use Case 5: Integrate Calendar Events with Tasks**

**Actors**: User  
**Preconditions**: User has enabled calendar integration.  
**Steps**:

1. User selects “Sync with Calendar” from the settings.
2. User logs into their calendar account and grants permission.
3. Task deadlines are synchronized with the external calendar.

**Postconditions**: Calendar events appear within the app, and tasks are visible in the external calendar.

<https://www.figma.com/design/CGcEkgag2K6KPPEpuEDnzS/My-planner-(Community)?node-id=0-1&node-type=canvas&t=ybiPEpijdklrTLor-0>