### **User Stories and Acceptance Criteria**

**User Stories from Two Actors:**

1. **As a Developer**, I want an intuitive task management interface with customizable views so that I can focus on daily priorities without unnecessary distractions.
   * **Acceptance Criteria:**
     + The UI should include a clean layout with minimal visual clutter.
     + Users can toggle between a simple view showing key task details and a detailed view with more information.
     + The "Add Task" button opens a form as a pop-up for easy task creation.
     + Users can expand or collapse task lists, so only relevant tasks are visible.
2. **As a Project Manager**, I want to organize tasks by priority, deadline, and project tags to quickly assess task status and ensure team alignment.
   * **Acceptance Criteria:**
     + Users can categorize tasks by project, priority level, due date, and customizable tags.
     + The system should allow for color-coded tasks to enhance visual differentiation.
     + Tasks should be sorted by due date by default, with options to change sorting criteria (e.g., priority, project).
     + A visual dashboard should be available to display task progress and priorities across projects.
3. **As a Developer**, I want to receive non-intrusive reminders with actionable buttons, so I can manage tasks effectively without interruptions.
   * **Acceptance Criteria:**
     + Notifications are brief pop-ups with snooze, dismiss, or mark-as-done options.
     + Users can customize the timing and frequency of notifications, including the option for a daily summary.
     + Users can select preferred notification channels (e.g., app, email, lock screen).
     + Low-priority task reminders can be turned off or minimized in frequency.
4. **As a Project Manager**, I want to collaborate with team members by sharing tasks and tracking their status, so the team stays aligned and progress is transparent.
   * **Acceptance Criteria:**
     + The app should support assigning tasks to team members with individual progress tracking.
     + Team members can comment on tasks and update statuses visible to all collaborators.
     + The system should send notifications to team members when task updates occur.
     + A shared dashboard view should display collaborative task statuses and deadlines.

### **Effort Estimation using Function Points**

Function points are estimated based on the complexity of subtasks required to meet the acceptance criteria. Here’s a breakdown for each story:

1. **User Story 1 (Intuitive Interface and Custom Views)**
   * **Subtasks:**
     + Design a clean layout and simple UI (~3 points, medium complexity).
     + Implement toggleable views between simple and detailed modes (~2 points, medium complexity).
     + Develop the "Add Task" pop-up form and highlight newly added tasks (~3 points, medium complexity).
     + Implement expandable and collapsible task lists (~2 points, medium complexity).

**Total Function Points:** 10 points

1. **User Story 2 (Task Organization by Priority, Deadline, and Tags)**
   * **Subtasks:**
     + Create categories for organizing tasks (project, priority, due date, tags) (~3 points, medium complexity).
     + Implement color-coded task options for quick visual identification (~2 points, low complexity).
     + Enable sorting of tasks by due date, priority, and project (~3 points, medium complexity).
     + Develop a dashboard to display task progress and priorities (~4 points, high complexity).

**Total Function Points:** 12 points

1. **User Story 3 (Non-intrusive Reminders with Actionable Buttons)**
   * **Subtasks:**
     + Design brief notification pop-ups with action buttons (~3 points, medium complexity).
     + Develop customizable reminder settings for timing and frequency (~3 points, medium complexity).
     + Allow users to choose notification channels (app, email, lock screen) (~2 points, low complexity).
     + Implement options to turn off or reduce frequency of low-priority reminders (~2 points, low complexity).

**Total Function Points:** 10 points

1. **User Story 4 (Collaboration and Task Sharing)**
   * **Subtasks:**
     + Implement task sharing and assignment features (~4 points, high complexity).
     + Allow task comments and status updates (~3 points, medium complexity).
     + Create a notification system for team updates (~3 points, medium complexity).
     + Develop a shared dashboard view for collaborative tasks (~4 points, high complexity).

**Total Function Points:** 14 points