Software Requirements Specification (SRS)

Project: todo app

Version: 1.0

Date: August 06, 2025

# 1. Introduction

## 1.1 Purpose

To help users manage their tasks efficiently.

## 1.2 Scope

- Task management

- Project planning

## 1.3 Intended Audience

- Students

- Professionals

- General users

# 2. System Overview

A simple and intuitive to-do list app designed to help users manage tasks efficiently. The app supports task creation, categorization, deadlines, and reminders.

# 3. Functional Requirements

## 3.1 User Account Management [0]

**Description:**

Users can create, edit, and delete their accounts.

**Priority:**

High

**Subtasks:**

- Implement user registration functionality.

- Design user profile management interface.

- Develop account deletion logic.

- Implement password change feature.

## 3.2 Task Creation [1]

**Description:**

Users can add new tasks with details like title, description, due date, and reminders.

**Priority:**

High

**Subtasks:**

- Add a text field for entering the task title.

- Add a text area for entering the task description.

- Implement a date picker for setting the due date.

- Add a checkbox for enabling reminders.

- Allow users to set specific reminder times.

- Display a confirmation message upon successful task creation.

## 3.3 Task Editing [2]

**Description:**

Users can modify existing tasks, including updating details and changing status.

**Priority:**

High

**Subtasks:**

- Implement an edit button for each task.

- Design an intuitive edit form for task details.

- Allow users to update task title, description, due date, and priority.

- Provide a mechanism to change task status.

- Ensure data persistence for edited tasks.

## 3.4 Task Deletion [3]

**Description:**

Users can remove tasks from their list.

**Priority:**

High

**Subtasks:**

- Implement a delete button for each task.

- Allow users to select multiple tasks for deletion.

- Remove selected tasks from the database.

- Update the user interface to reflect the deletion of tasks.

## 3.5 Task Categorization [4]

**Description:**

Users can categorize tasks using tags or labels for better organization.

**Priority:**

High

**Subtasks:**

- Allow users to add, edit, and delete tags for tasks.

- Display tasks in a list view, filtering by selected tags.

- Enable users to search for tasks based on specific tags.

- Provide a visual representation of tag usage (e.g., color-coding or tag cloud).

## 3.6 Deadline and Reminder Management [5]

**Description:**

Users can set due dates and reminders for tasks.

**Priority:**

High

**Subtasks:**

- Implement due date functionality for tasks.

- Develop reminder creation and scheduling features.

- Integrate with various notification channels (e.g., system notifications, email, SMS).

## 3.7 Task Prioritization [6]

**Description:**

Users can prioritize tasks based on importance or urgency.

**Priority:**

High

**Subtasks:**

- Implement a priority level system (e.g., High, Medium, Low)

- Allow users to assign priority levels to individual tasks

- Display tasks in prioritized order by default

- Enable sorting tasks by priority level

- Provide visual cues to distinguish priority levels (e.g., color-coding)

## 3.8 Task Viewing [7]

**Description:**

Users can view tasks in different formats, such as list or calendar view.

**Priority:**

High

**Subtasks:**

- Implement a list view to display all tasks with title, due date, and category.

- Implement a calendar view to display tasks with their due dates.

- Allow users to filter tasks in both views by category, due date, or status (completed/incomplete).

## 3.9 Task Completion Tracking [8]

**Description:**

Users can mark tasks as complete and track their progress.

**Priority:**

High

**Subtasks:**

- Display a visual indicator to show task completion status (e.g., checkbox, icon, color change).

- Allow users to toggle task completion status.

- Update the task list to reflect the completion status of individual tasks.

- Provide a summary of completed tasks, potentially with filtering options (e.g., by date, category).

- Option to archive completed tasks, separating them from active tasks.

## 3.10 Search Functionality [9]

**Description:**

Users can search for tasks based on keywords, categories, or due dates.

**Priority:**

High

**Subtasks:**

- Implement keyword-based search for task titles and descriptions.

- Allow users to filter tasks by category.

- Enable search by due date range.

- Display search results in a clear and concise manner.

- Provide suggestions for relevant search terms as the user types.

## 3.11 Cloud Synchronization [10]

**Description:**

Users' tasks and data are synchronized across multiple devices.

**Priority:**

High

**Subtasks:**

- Implement user account creation and login functionality.

- Define data structure for task synchronization.

- Develop API endpoints for cloud data access.

- Integrate a reliable cloud storage service.

- Implement conflict resolution mechanism for data conflicts.

- Ensure data privacy and security during synchronization.

## 3.12 User Interface (UI) Design [11]

**Description:**

The app has a clean, intuitive, and user-friendly interface.

**Priority:**

High

**Subtasks:**

- Design the main task list view.

- Design the task creation form.

- Design the task details view.

- Design the settings menu.

## 3.13 Security [12]

**Description:**

User data is securely stored and protected.

**Priority:**

High

**Subtasks:**

- Implement data encryption using industry-standard algorithms.

- Store encryption keys securely and protect them from unauthorized access.

- Enforce password complexity rules, including minimum length, character types, and uniqueness.

- Offer multi-factor authentication (MFA) options such as SMS codes or authentication apps.

- Regularly audit and update security protocols and measures to address emerging threats.

# 4. Non-Functional Requirements

**NFR1**: The app should be responsive and mobile-friendly.

**NFR2**: Tasks should sync across devices using cloud storage.

**NFR3**: The system should support at least 10,000 users simultaneously.