Software Requirements Specification (SRS)

Project: Todo App

Version: 1.0

Date: August 09, 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 Task Management [0]

**Description:**

Users can create, edit, and delete tasks.

**Priority:**

High

**Subtasks:**

- Implement task creation functionality

- Develop task editing features

- Design task deletion functionality

## 3.2 Task Scheduling [1]

**Description:**

Users can assign due dates and set reminders.

**Priority:**

High

**Subtasks:**

- Implement due date functionality for tasks.

- Allow users to set reminders for tasks with customizable notification times.

- Display due dates and reminders prominently in the task view.

## 3.3 Task Categorization [2]

**Description:**

Users can organize tasks using categories or tags.

**Priority:**

High

**Subtasks:**

- Implement a system for creating and managing categories.

- Allow users to assign multiple categories to a single task.

- Provide a clear visual representation of categories and their associated tasks.

- Enable filtering and searching tasks by category.

## 3.4 Deadline Notification [3]

**Description:**

The app should notify users of upcoming deadlines.

**Priority:**

High

**Subtasks:**

- Send push notifications for tasks nearing their deadline.

- Allow users to customize notification frequency and time.

- Implement email notifications as an alternative to push notifications.

## 3.5 Data Synchronization [4]

**Description:**

Tasks should sync across devices using cloud storage.

**Priority:**

High

**Subtasks:**

- Implement cloud storage integration (e.g., AWS S3, Firebase).

- Develop synchronization logic to handle task creation, updates, and deletions.

- Ensure secure data transfer between devices and the cloud.

- Implement conflict resolution strategies for concurrent modifications.

## 3.6 System Scalability [5]

**Description:**

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

**Priority:**

High

**Subtasks:**

- Implement load balancing across multiple servers.

- Optimize database queries for scalability.

- Utilize caching mechanisms to reduce database load.

# 4. Non-Functional Requirements

**NFR-1**: The application should be responsive and mobile-friendly.

**NFR-2**: Tasks should sync across devices using cloud storage.

**NFR-3**: The system should support at least 10,000 users simultaneously.