Software Requirement Specification for Task Management Application

# Introduction

* 1. **Purpose**

The purpose of this document is to outline the requirements for the development of a Task Management Application. This application will enable users to create, manage, and track their tasks in an organized manner.

# Scope

The Task Management Application will include features such as task creation, categorization, priority setting, due date tracking, and task completion management. The application will target both web and mobile platforms.

# Definitions, Acronyms, and Abbreviations

* **SRS:** Software Requirement Specification
* **UI**: User Interface
* **API**: Application Programming Interface

# System Overview

* 1. **System Description**

The Task Management Application will provide a user-friendly interface for users to create and manage tasks. Users will be able to access the application via web browsers and mobile devices.

* 1. **User Roles**

1. **User:** Regular users who can create, edit, and manage tasks.
2. **Admin:** Administrators who can manage user accounts and system settings.

# Functional Requirements

* 1. **User Registration and Authentication**
* Users must be able to create accounts using a valid email address.
* Users must log in to the application using their registered credentials.

# Task Creation and Management

* Users can create new tasks by providing a title, description, priority, and due date.
* Users can categorize tasks into predefined categories (e.g., work, personal, shopping).
* Users can mark tasks as completed or reopen completed tasks.

# Task Sorting and Filtering

* Users can sort tasks based on priority, due date, and completion status.
* Users can filter tasks by category and completion status.

# Notifications

* Users should receive email notifications for approaching task due dates.
* Users should receive notifications when tasks are assigned to them by other users (if collaboration features are added in the future).

# Non-Functional Requirements

* 1. **Performance**
* The application must respond to user actions within 2 seconds under normal load conditions.
* The application should support up to 10,000 concurrent users.

# Security

* User passwords must be securely stored using encryption.
* User authentication must be implemented to prevent unauthorized access.

# Usability

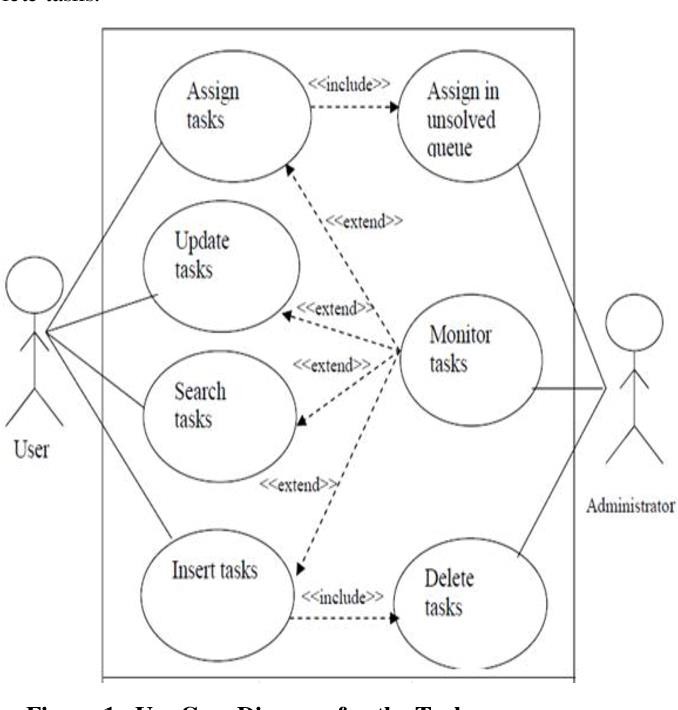
* The user interface should be intuitive and user-friendly.
* The application should be responsive and provide a consistent experience across different devices.

**5. System Architecture**

The application will follow a client-server architecture, with the following components:

* **Front-end:** Web and mobile UI for users to interact with the application.
* **Back-end:** Server-side logic for user authentication, task management, and data storage.

**Flowchart:**

****

# 6. User Interface Mock-ups(Designing part)

User interface mock-ups or wire frames provide visual representations of how the application's user

interface will look and function. They help stakeholders, including developers, designers, and clients, visualize the application's layout and flow before actual development begins. Mock-ups can be

created using tools like Adobe XD, Sketch, Figma, or even hand-drawn sketches.

# For a task management application, mock-ups might include screens like:

1. **Login and Registration**: Show how users will input their credentials and create accounts.
2. **Task List:** Display how tasks will be presented with details like title, description, priority, and due date.
3. **Task Creation**: Illustrate the process of creating a new task, including form fields and buttons.
4. **Filter and Sort**: Demonstrate how users can filter tasks based on criteria like category or due date.
5. **Notifications**: Visualize how notifications will be displayed to users.

# Glossary

The glossary section is a reference area where you define and explain technical terms, acronyms, abbreviations, and other specialized language used in the SRS. This helps ensure a shared

understanding among all stakeholders, especially those who might not be familiar with the project's domain or technical jargon.

# For instance:

**SRS (Software Requirement Specification):** A document that outlines the requirements and features of the software to be developed.

**UI (User Interface):** The visual elements and controls that users interact with on a software application.

**API (Application Programming Interface):** A set of rules that allows different software applications to communicate with each other.