**1. Introduction**

1.1 Purpose

The purpose of this document is to outline the requirements for the development of a To-Do List web application.

1.2 Scope

The To-Do List application will allow users to add, edit, delete, and manage tasks. It will support task prioritization, due dates, searching, marking tasks as completed and exporting tasks as a downloadable file.

**2. Functional Requirements**

2.1 Task Management

* Users can add a new task with a title, due date, and priority (High, Medium, Low).
* Tasks can be edited to update the title, due date, or priority.
* Users can delete tasks from the list.

2.2 Task Display

* All tasks are displayed in a list format showing the title, due date, and priority.
* Overdue tasks are visually highlighted.

2.3 Task Interactions

* Users can mark tasks as completed by clicking on them.
* Tasks can be searched by their titles using a search functionality.

2.4 Task Export

* Users can download the list of tasks in a text file.
* The file includes the creation date of the task list and all task details.

**3. Non-functional Requirements**

3.1 Performance

* The application should be responsive and work efficiently across various devices and screen sizes.
* The task management operations should have minimal latency.

3.2 User Interface

* The user interface should be intuitive, with easy-to-use controls for adding, editing, and deleting tasks.
* Visual cues should help differentiate between completed, overdue, and pending tasks.

3.3 Security

* Users' task data should be stored securely.
* No unauthorized access to the task lists should be allowed.

**4. User Stories**

* As a user, I want to easily add tasks to the list with a title, due date, and priority.
* As a user, I want the ability to edit task’s details.
* As a user, I want to delete tasks that are no longer needed.
* As a user, I want to mark tasks as completed when they are done.
* As a user, I want to download my task list for reference.

**5. Constraints**

* The application should be developed using HTML, CSS, and JavaScript.
* Compatibility with modern web browsers (Chrome, Firefox, Safari, Edge) should be ensured.

**6. Future Enhancements**

* Implement user authentication to allow multiple users to have their individual task lists.
* Add drag-and-drop functionality for task reordering.
* Include the ability to set reminders for tasks.

**7. Glossary**

* SRS: Software Requirements Specification
* HTML: HyperText Markup Language
* CSS: Cascading Style Sheets
* JavaScript: Programming language for web development

**Conclusion**

This SRS serves as a guideline for the development of the To-Do List application. It outlines the functional, non-functional requirements, user stories, constraints, and potential future enhancements for the application.