**Software Requirements Specification (SRS)**

**1. Introduction**

**1.1 Purpose**

Describe the purpose of the SRS and its intended audience.

Example:  
This document outlines the software requirements for the [Project Name] application. It is intended for developers, testers, project managers, and stakeholders.

**1.2 Scope**

Provide a brief description of the software being specified.

Example:  
The [Project Name] system will be a web-based platform allowing users to register, log in, and manage personal tasks. It will include authentication, task CRUD operations, and a reporting dashboard.

**1.3 Definitions, Acronyms, and Abbreviations**

List terms used in the document and their definitions.

| **Term** | **Definition** |
| --- | --- |
| CRUD | Create, Read, Update, Delete |
| UI | User Interface |

**1.4 References**

List any supporting documents or references.

Example: IEEE 830-1998, Agile Manifesto

**2. Overall Description**

**2.1 Product Perspective**

Explain how the product fits into the existing system or as a standalone.

Example:  
This system is standalone and will be deployed on a cloud server.

**2.2 Product Functions**

Provide a summary of the main features.

* User registration and login
* Task creation, editing, deletion
* Task categorization and search
* Report generation

**2.3 User Characteristics**

Describe the target user base.

Example:  
End-users with basic computer skills, ages 18–60.

**2.4 Constraints**

List limitations such as technologies, languages, or frameworks.

Example:

* Must use React.js for front-end
* PostgreSQL for database
* Must run on AWS infrastructure

**2.5 Assumptions and Dependencies**

External factors that could affect requirements.

Example:

* Internet access is required
* Users will use modern browsers (Chrome, Firefox)

**3. Specific Requirements**

**3.1 Functional Requirements**

**3.1.1 User Registration**

* The system shall allow new users to register with email and password.

**3.1.2 User Login**

* The system shall authenticate users using email and password.

**3.1.3 Task Management**

* The system shall allow users to create, update, and delete tasks.

**3.1.4 Reports**

* The system shall allow users to generate a summary report of tasks.

**3.2 Non-Functional Requirements**

| **Requirement Type** | **Description** |
| --- | --- |
| Performance | System should respond within 2 seconds |
| Availability | 99.9% uptime |
| Security | Passwords must be encrypted using bcrypt |
| Usability | UI must be accessible and mobile responsive |

**3.3 Interface Requirements**

* **User Interface:** Web-based, accessible via browser.
* **Database Interface:** Uses PostgreSQL with REST API endpoints.

**4. Appendices**

Include any additional materials or diagrams (e.g., use case diagrams, data flow diagrams, wireframes).