



Lab Sheet: Software Requirements Specification (SRS)

Introduction

A **Software Requirements Specification (SRS)** is a document that outlines in detail the expected behavior, features, and limitations of a software product. It acts as the cornerstone for software design, development, and validation. A clear and well-structured SRS helps ensure that the development team and stakeholders share a common understanding of the system's requirements.

Objective

Your task is to prepare an **SRS document for your group project** using a **modern SRS format**. Instead of using outdated rigid templates, you are encouraged to choose one of the following more flexible and industry-relevant formats:

SRS Format Options

1. IEEE 830 Standard Format

- A well-structured, traditional format by the IEEE.
- It organizes requirements into sections such as Introduction, Overall Description, Specific Requirements, etc.
- Ideal for formal documentation in larger or regulated projects.

2. Use Case Based Format

- Focuses on **actors** and how they **interact with the system**.
- Captures functional requirements in the form of **use cases** with preconditions, flows, and postconditions.
- Useful for systems with multiple user interactions and scenarios.

3. **Agile Specification (User Stories)**

- Based on **Agile methodologies**, this format uses **user stories** (e.g., “As a user, I want to...”) to describe functional needs.
- Includes **acceptance criteria** and may be organized using tools like product backlogs.
- Suited for iterative, collaborative, and evolving project environments.

4. **Volere Template**

- A comprehensive template with a focus on **requirements quality** and traceability.
- Includes **stakeholder profiles**, **fit criteria**, **constraints**, and even requirement rationales.
- Often used in professional consulting and high-accuracy environments.

5. **Custom Format Tailored to Project Needs**

- You may design your own SRS format to suit your project domain or team workflow.
- Must still cover **essential components** like objectives, functional/non-functional requirements, and use scenarios.
- Encouraged for creative or domain-specific projects (e.g., game design, AI tools).

Tasks / Structure of the Document

Your SRS should include the following components, customized based on the format you choose:

1. **Title Page and Table of Contents**

2. **Introduction**

- Purpose, Scope, Intended Audience

3. **Overall Description**

CCS3310- Software Engineering Methods

- Product Perspective, Product Functions, User Characteristics, Constraints

4. **Specific Requirements**

- At least 5 **Functional Requirements**
- At least 5 **Non-Functional Requirements**

5. **Use Cases** (if applicable)

- Include 2–3 detailed use cases (actor, flow, pre/post conditions)

6. **Appendices**

- Diagrams (UML, ERD, DFD, architecture), glossary, or references
-

Deliverable

- Submit your final **SRS document** as a **PDF file**.