Introduction to Software Requirements Specification (SRS)

MACQUARIE University

Programming Fundamentals – Year 11

MACQUARIE University

Learning Objectives

- Define what an SRS is and explain its importance.
- Understand different types of software requirements.
- Learn about the characteristics of a good SRS.
- Discuss the role of SRS in the software engineering lifecycle.
- Hands-on activity: Create basic requirements using GitHub and the Hugo website generator

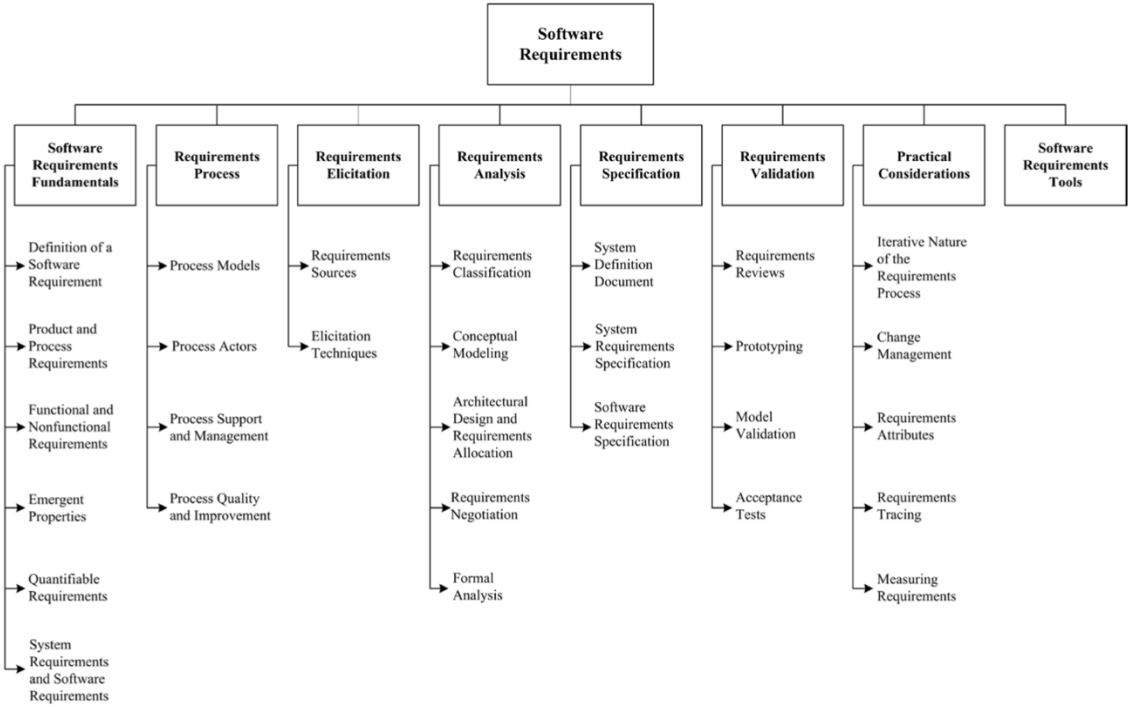
MACQUARIE University

What is a System Requirements Specification (SRS)?

- A formal document detailing what a software system must do.
- Forms a contract between the customer and the software team.

SRS in Context

- Used by software engineering teams to define project boundaries.
- Ensures developers build what the customer expects.



Breakdown of Topics for the Software Requirements

IEEE Computer Society. (2014). Guide to the Software Engineering Body of Knowledge (SWEBOK v3).

Scenario: Building Software for a School

- Example: Creating a School Event Management System.
  - What features would be required?

Understanding Software Requirements

Requirements describe what the software shall do and how it will must perform.

Functional Requirement Example:

- "The system ***shall*** allow students to register for events."

Non-Functional Requirement Example:

- "The system ***shall*** handle at least 500 users simultaneously."

In the context of an SRS:

- "**shall**" indicates a requirement that is mandatory and must be implemented
- "**must**" conveys a similarly strong imperative for essential requirements, while
- "**should**" suggests a recommended, but not compulsory, feature or behaviour.

MACQUARIE University

Types of Requirements

- ***Functional***: Specific functions the system must perform.
- ***Non-Functional***: Performance, usability, and security aspects.
- ***Security***: Measures to protect data and user privacy.
- ***System***: Hardware, software, network infrastructure, and other environmental requirements necessary for the software's operation

Where Do Requirements Come From?

- ***Stakeholders, business needs, regulations, and standards.***

- ***Security***: Government Standard

  Australian Cyber Security Centre (ACSC) Information Security Manual (ISM)

  https://www.cyber.gov.au/resources-business-and-government/essential-cyber-security/ism

- Software engineering environment prepares requirements for project use. See:

  https://brokenhill-h.polemos.ai/workbooks/standards/ism/

MACQUARIE University

Workbooks: Documenting Requirements

- Tools like GitHub to keep a version-controlled record of requirements.
- Markdown makes it easy to collaborate and track changes.
- Web frameworks like Hugo (https://gohugo.io/) simplify display of requirements across team.
- Use tools such as Pandoc (https://pandoc.org) to convert text and web material into formal Word and PDF documents

- Software Engineering projects use Workbooks to organise and track project artefacts.
  - Let us review the courses online workbook https://brokenhill-h.polemos.ai/

MACQUARIE University

Characteristics of a Good SRS

- Correct
- Unambiguous
- Complete
- Consistent
- Ranked
- Verifiable
- Traceable: (REST) Web systems greatly simplify this.

Use “shall” or “must” statements to define .

Example Requirement Statement

- ID: REQ-001
- Title: User Authentication
- Description: “The system must allow users to log in using a username and password.”
- Acceptance Criteria: Users can log in with valid credentials.Traceable: (REST) Web systems greatly simplify this.

Use “shall” or “must” statements to define .

MACQUARIE University

What is Traceability?

- The ability to track each requirement throughout its lifecycle.
- Essential for ensuring all requirements are implemented and tested.
- This includes:
  - Why the requirement
  - Planning to implement the requirement
  - Where the requirement is met
  - How the requirement is tested

Let us review the tooling.

Common Mistakes to Avoid

- Designs in specification
- Vague statements
- Keep project management constraints separate
- ”Desireaments” – Unrealistic or unnecessary personal requirements (e.g. I desire a Ferrari as my first car)

MACQUARIE University

Where Does SRS Fit in SDLC

- **Planning**: Defines scope and requirements.
- **Estimation, Costing & Negotiation**: Defines scope and requirements.
- **Construction**: Guides the development process.
- **Testing**: Ensures the final product meets requirements.
- **Acceptance**: SRS is a key item in getting paid to complete a project
- **Getting Paid**

MACQUARIE University

Hands-On Activity: Write Your Own Requirement

- Create a functional, non-functional, and security requirement for a simple system.
- Use the SRS template in the Hugo site.

MACQUARIE University

Summary & Next Steps

- Today, we covered what an SRS is, its types, characteristics, and how to document requirements
- Homework: Refine your requirements and check in to GitHub.