

SOFTWARE REQUIREMENT ENGINEERING

DETAIL

This Course will cover the following Topics:

- 📍1. Basics
- 📍2. Procedure and Processes
- 📍3. Project and Risk Management
- 📍4. Responsibilities and Roles
- 📍5. Identification of Requirements

DETAILS

- ⑥6. Specification of requirements
- ⑥7. Requirements Analysis
- ⑥8. Tracking of Requirements
- ⑥9. Quality Assurance

REFERENCES

- The Requirements Engineering Handbook by Ralph R. Young
- Software requirements: Styles and techniques by Soren Lauesen

LEARNING TARGET

- 📍 What is a requirement?
- 📍 What is the meaning and purpose of requirements?
- 📍 How can requirements be classified?
- 📍 What types of requirements are there?
- 📍 What problems are there concerning requirements?

LEARNING TARGET

- ④ What concepts are important in connection with requirements?
- ④ What is the difference between RM (Requirements Management) and RE (Requirements Engineering)?
- ④ What important norms and standards exist?
- ④ Why is Requirements Engineering important?

1. WHAT IS A REQUIREMENT?

- 📌 A requirement is a condition or a skill that a user needs in order to solve a problem or arrive at a goal.
- 📌 A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.

WHAT IS THE MEANING AND PURPOSE OF REQUIREMENTS?

- 📌 Foundation for assessment, planning, execution and monitoring of the project activity
- 📌 Customer expectations
- 📌 Component of agreements, orders, project plans...
- 📌 Setting of system boundaries, scope of delivery, contractual services

CLASSIFICATION OF REQUIREMENTS

📌 Requirements consist of:

- ≡ process requirements

- ≡ product requirements

📌 Process requirements:

costs, marketing, processing time, sales and distribution, organization, documentation

📌 Describe needs and limitations of the business processes.

PRODUCT REQUIREMENTS

- ④ Product requirements consist of functional and non-functional product requirements.
- ④ Both can be regarded from the point of view of the user (external) or customer and from the point of view of the developer (internal).

FUNCTIONAL PRODUCT REQUIREMENTS

- 📌 Functional requirements describe the function of the system
- 📌 From the user's point of view: user interface, applications, services
- 📌 From the customer's point of view:
user
interface, applications, services

Note: User and customer can be different!

FUNCTIONAL PRODUCT REQUIREMENTS

- 📌 From the developer's point of view:
architecture, power supply, load
distribution

NON-FUNCTIONAL PRODUCT REQUIREMENTS

- 📌 Non-functional requirements describe the quality attributes of the system.
- 📌 from the point of view of the user:
reliability, performance, usability
- 📌 from the point of view of the customer:
reliability, performance, usability
- 📌 from the point of view of the developer:
testability, serviceability, tools

TYPES OF REQUIREMENTS

- 📍 customer requirements
- 📍 solution/system requirements,
- 📍 product/component requirements

Software Requirement Specification (SRS)

also known as a **System Requirements Specification**)

It is a **document** or set of documentation that describes the features and behavior of a system or **software** application.

Usually a combination of problems and opportunities are needed to provide motivation for a new system.

Functional

Nonfunctional

Business Requirements

Vision and Scope Document

User Requirements

Use-Case Document

System Requirements

Functional Requirements

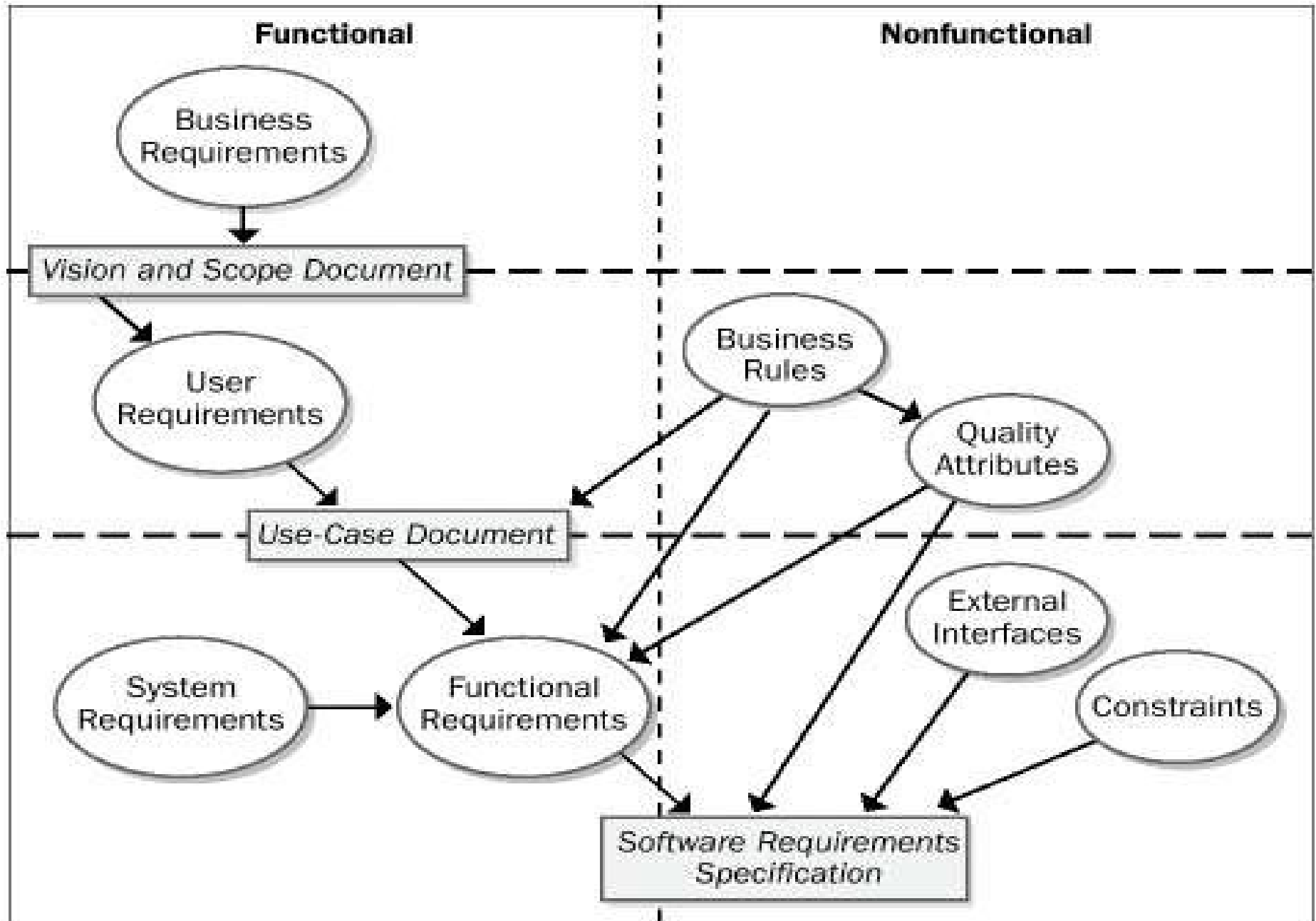
Business Rules

Quality Attributes

External Interfaces

Constraints

Software Requirements Specification



PROBLEMS WITH REQUIREMENTS

- ④ unclear objectives

- ④ communication problems

- ④ language barriers

- ④ knowledge barriers

- ④ vague formulation

- ④ too formal formulations

- ④ ambiguous, overly specified, unclear, impossible, contradictory requirements

PROBLEMS WITH REQUIREMENTS

- Ⓟ instability of the requirements
- Ⓟ bad quality of the requirements
- Ⓟ insufficient user involvement
- Ⓟ overlooked user classes
- Ⓟ inaccurate planning
- Ⓟ minimal specification

QUALITY CRITERIA FOR REQUIREMENTS

- 📌 **Necessary:** If the system can meet prioritized real needs without the requirement, it isn't necessary.
- 📌 **Feasible:** The requirement is doable and can be accomplished within budget and schedule.
- 📌 **Correct:** The facts related to the requirement are accurate, and it is technically and legally possible.
- 📌 **Concise:** The requirement is stated simply.
- 📌 **Unambiguous:** The requirement can be interpreted in only one way.
- 📌 **Complete:** All conditions under which the requirement applies are stated, and it expresses a whole idea or statement.

QUALITY CRITERIA FOR REQUIREMENTS

- 📌 **Consistent:** It is not in conflict with other requirements.
- 📌 **Verifiable:** Implementation of the requirement in the system can be proved.
- 📌 **Traceable:** The source of the requirement can be traced, and it can be tracked throughout the system (e.g., to the design, code, test, and documentation).
- 📌 **Allocated:** The requirement is assigned to a component of the designed system.
- 📌 **Design independent:** It does not pose a specific implementation solution.

QUALITY CRITERIA FOR REQUIREMENTS

- 📌 **Non-redundant:** It is not a duplicate requirement.
- 📌 **Written using the standard construct:** The requirement is stated as an imperative using “shall.”
- 📌 **Assigned a unique identifier:** Each requirement shall have a unique identifying number.
- 📌 **Devoid of escape clauses:** Language should not include such phrases as “if,” “when,” “but,” “except,” “unless,” and “although.” Language should not be speculative or general (i.e., avoid wording such as “usually,” “generally,” “often,” “normally,” and “typically”).

QUALITY CRITERIA FOR REQUIREMENTS

- ④ The requirements specification must be complete, consistent, modifiable and traceable

SOLUTION

• A solution is the implementation of the requirement Engineering & Requirement management.

PRIORITY OF REQUIREMENTS

Commitment

- ≡ Commitment is the degree of obligation
- ≡ Observing legal responsibilities, especially in case of fault

Fault

- ≡ Deviation of the current state from the target state

Evaluation of the importance/urgency

CRITICALITY OF REQUIREMENTS

- ④ **Evaluation** of the risk of a requirement by evaluating the damage in case of non-fulfillment of a requirement

VALIDATION

- ④ Process of confirmation that the specification of a phase or the entire system fulfills the customer's requirements

VERIFICATION

- ④ Comparison of an intermediate product with its specifications. It is thereby determined if the software was developed correctly and if the specifications that were determined during the previous phase were fulfilled

DELINEATION BETWEEN REQUIREMENTS MANAGEMENT AND ENGINEERING

- ④ Requirements Management (RM) includes processes for the identification and management of requirements
- ④ Requirements engineering includes the basic engineering skills

STANDARDS AND NORMS

📍 ISO 9000:

- ≡ Requirements of a quality management system:
 - ✍ defined concepts and basics of a QMS
 - ✍ domain or industry neutral

📍 ISO 9126:

- ≡ Defines a quality model with six categories:
 - ✍ Functionality, reliability, usability, efficiency, maintainability, portability

STANDARDS AND NORMS

📌 IEEE 610:

- ≡ Standard Glossary of Software Engineering Terminology

📌 IEEE 830:

- ≡ Recommended Practice for Software Requirements Specifications

📌 IEEE 1233:

- ≡ Guide for Developing of System Requirements Specifications

STANDARDS AND NORMS

📌 IEEE 1362:

📌 Guide for Information Technology – System Definition

PROCESS NORMS

📍 ISO 12207:

≡ Standard for Software Life Cycle Process

📍 ISO 15288:

≡ System Life Cycle Process

📍 ISO 15504:

≡ Software Process Improvement and Capability Determination (SPICE)

📍 Capability Maturity Model Integrated (CMMI)

REASONS WHY REQUIREMENT ENGINEERING IS OFTEN NEGLECTED

- 📌 *Neglect due to high time pressure*
- 📌 *Neglect due to an exclusive orientation toward fast results*
- 📌 *Neglect due to an exclusive fixation on costs*
- 📌 *Neglect due to misinterpretations (many things are seen as given)*

POSSIBLE CONSEQUENCES OF NEGLECTING REQUIREMENTS ENGINEERING

- 📍 Requirements become imprecise
- 📍 Requirements are ambiguous
- 📍 Requirements are contradictory
- 📍 Requirements that change
- 📍 Requirements that do not fulfill the criteria
- 📍 Requirements that can be interpreted differently
- 📍 Missing requirements