ICT-TEX course on Digital skills

Topic 7: Introduction to Software Engineering

The course is developed under Erasmus+ Program Key Action 2:
Cooperation for innovation and the exchange of good practices Knowledge Alliance

ICT IN TEXTILE AND CLOTHING HIGHER EDUCATION AND BUSINESS

7.2. Requirements Engineering

These slides are part of the topic on
"Topic 7: Introduction to Software Engineering" of the course on Digital skills in Textile and clothing industry.

Check also the other themes in this topic:

- 7.1. Software Engineering
- 7.3. Introduction to modeling and UML
- 7.4. UML Diagrams

Contents

2. Requirements Engineering

- Requirements Engineering
- Requirements
- Requirements classification
- Non-functional requirements classification
- Metrics for specifying non-functional requirements
- Requirements engineering process
- Requirements engineering process and stakeholders
- Software Requirements Specification or (SRS)
- Software requirements specification structure
- Users of software requirements specification

Requirements Engineering

Requirements engineering is *"the term for the broad spectrum of tasks and techniques that lead to an understanding of requirements. From a software process perspective, requirements engineering is a major software engineering action that begins during the communication activity and continues into the modeling activity. Requirements engineering establishes a solid base for design and construction. It must be adapted to the needs of the process, the project, the product, and the people doing the work."*

- *Pressman, R., Maxim, B. Software Engineering: A Practitioner's Approach. 9th edition, Published by McGraw-Hill Education, ISBN: 9781260548006, 2019, pp.103*

Requirements Engineering

"Requirements engineering is the subset of systems engineering concerned with discovering, developing, tracing, analyzing, qualifying, communicating and managing requirements that define the system at successive levels of abstraction."

- *Dick J., Hull E., Jackson K. Introduction. In: Requirements Engineering. Springer, Cham. https://doi.org/10.1007/978-3-319-61073-3_1, ISBN: 978-3-319-61073-3, (2017) , pp.9*

Requirements Engineering

Requirements Engineering is the process of establishing the services that the customer requires from the software system, as well as the constraints under which it operates and is created.

Requirements Engineering is a discipline of software technology that encompasses the activities of the specification of software products or systems.

Requirements

The *requirements* themselves are a description of the services and limitations that the system has, which are generated during the Requirements Engineering Process.

The *requirements* are a specification of what needs to be implemented. They describe how the system should behave or a specific property or attribute of the system. They are also a requirement for the system development process.

Requirements

- The requirements help us determine:
  - What the benefits of the system would be,
  - What it should do, and
  - Some general understanding of its capabilities.
- They can also help identify potential difficulties in implementing the system at an early stage.
- They affect the entire life cycle of a system.

Requirements

- **Well-defined** requirements contribute to a good and successful product.

- **Poorly defined** requirements lead the product to failure.

- In different cases, the requirements may have different levels of detail. From the general that the system will offer a service, to a specific description in detail.

Back to Contents

Requirements classification

The Requirements can be divided into the following categories

Depending on the group that uses or describes them

Depending on what they describe

Back to Contents

Requirements classification

Depending on the group that uses or describes them are:

User requirements

Business requirements

System requirements

Requirements classification

User requirements

- These are the requirements of the user (client) to the system. They are usually described in natural language or user terminology. They include diagrams of the services that the system will provide, as well as the corresponding operational constraints. They are written for the clients.

Requirements classification

Business requirements

- These are the requirements of the business to the system. These are the costs or budget for development, the time for development, the technologies or processes used, and others.

Requirements classification

System requirements

- These are detailed system requirements, described in non-contradictory language. They should be understandable to both technical staff and customers. Structured document with a detailed description of system functions, services, and operational constraints. These types of requirements determine what needs to be implemented and can be part of the contract for the development of the system.

Requirements classification

Depending on what the requirements describe are:

Functional requirements

Non-functional requirements

Domain requirements

Back to Contents

Requirements classification

Functional requirements

- These are requirements that describe the behavior of the system. They describe the different services that the system provides, the way the system should respond to specific inputs and behavior, and in specific situations.
  - User functional requirements can only describe the basic idea of a given service, but
  - System functional requirements must describe the system at the best possible level of detail.

Requirements classification

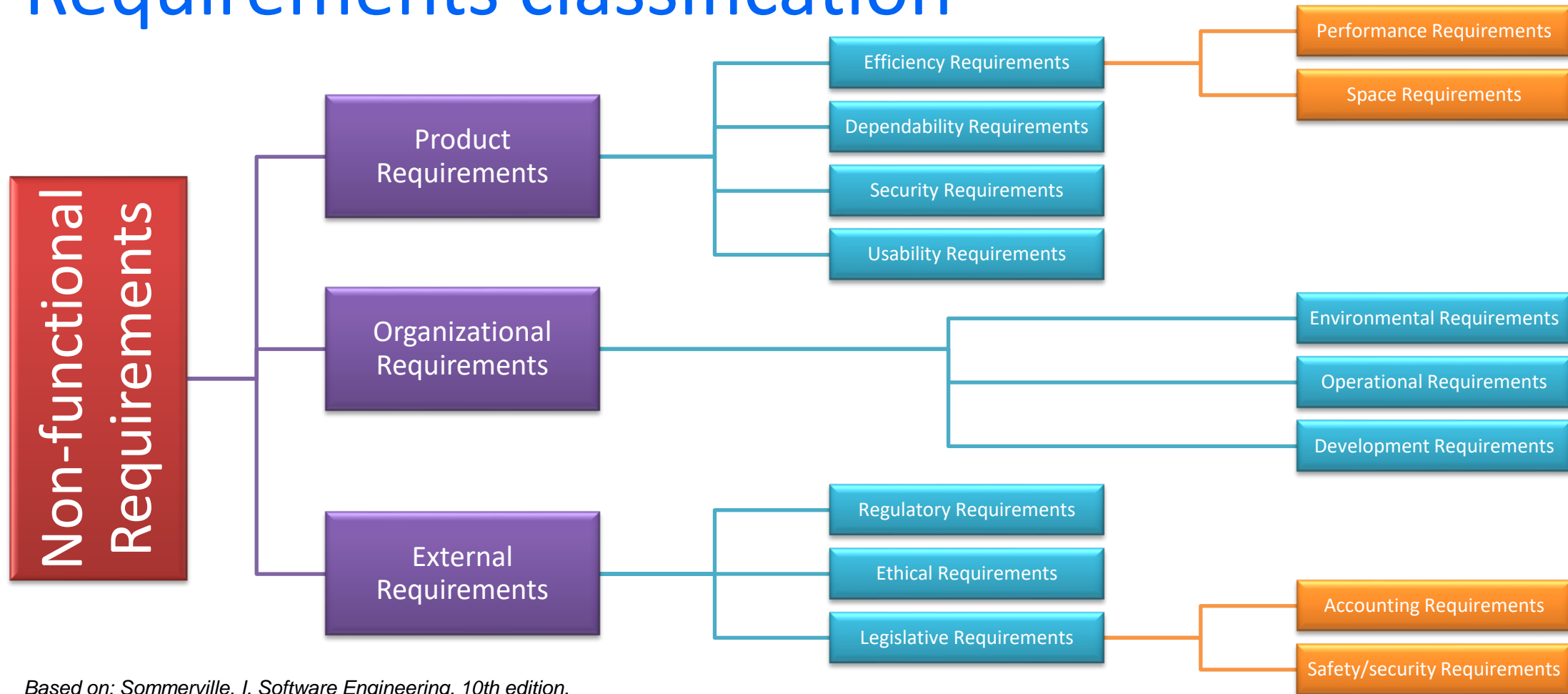## Non-functional requirements (quality attribute requirements)

- These are requirements describing limitations in the system or development process. These can be requirements for performance, reliability, security, performance, capacity, or other resources, technologies, and processes used, etc.

Domain requirements

- These are requirements produced by the application area of the system that reflects its specifics.

Requirements classification

Non-functional Requirements

Product Requirements
- Efficiency Requirements
  - Performance Requirements
  - Space Requirements
- Dependability Requirements
- Security Requirements
- Usability Requirements

Organizational Requirements
- Environmental Requirements
- Operational Requirements
- Development Requirements

External Requirements
- Regulatory Requirements
- Ethical Requirements
- Legislative Requirements
  - Accounting Requirements
  - Safety/security Requirements

*Based on: Sommerville, I. Software Engineering. 10th edition,
Published by Pearson Education, ISBN: 978-1-292-09613-1, 2016*

Back to Contents

Non-functional requirements classification

They are divided into three main categories:

- Product requirements
- Organizational requirements
- External requirements

Back to Contents

Non-functional requirements classification

Product requirements

- Specify or constrain the runtime behavior of the software.
- Examples for non-functional product requirements are:
  - ✓ Performance requirements
  - ✓ Usability requirements
  - ✓ Efficiency requirements
  - ✓ Dependability requirements
  - ✓ Security requirements, and Space requirements.

Non-functional requirements classification

Organizational requirements

- These requirements are "*all requirements that are derived from factors external to the system and its development process.*"

  -*Sommerville, I. Software Engineering. 10th edition, Published by Pearson Education, ISBN: 978-1-292-09613-1, 2016, pp. 108*

Non-functional requirements classification

External requirements

- These requirements are non-functional "*system requirements derived from policies and procedures in the customer's and developer's organizations.*"

  -*Sommerville, I. Software Engineering. 10th edition, Published by Pearson Education, ISBN: 978-1-292-09613-1, 2016, pp. 108-109*

- Examples for non-functional organizational requirements:

  - Environmental requirements;

  - Operational requirements;
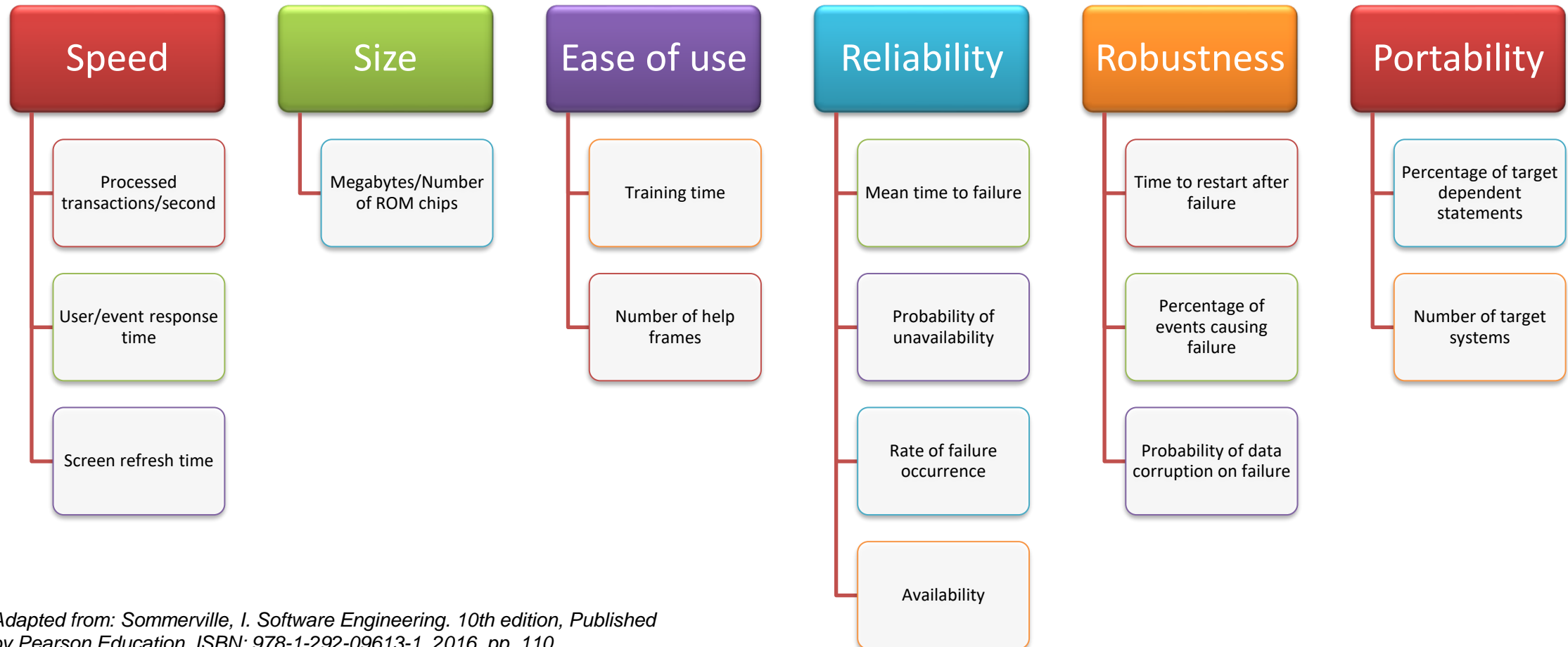
  - Development requirements.

Non-functional requirements classification

External requirements

- Examples for non-functional external requirements:
  - ✓ Regulatory requirements
  - ✓ Ethical requirements
  - ✓ Legislative requirements
  - ✓ Accounting requirements
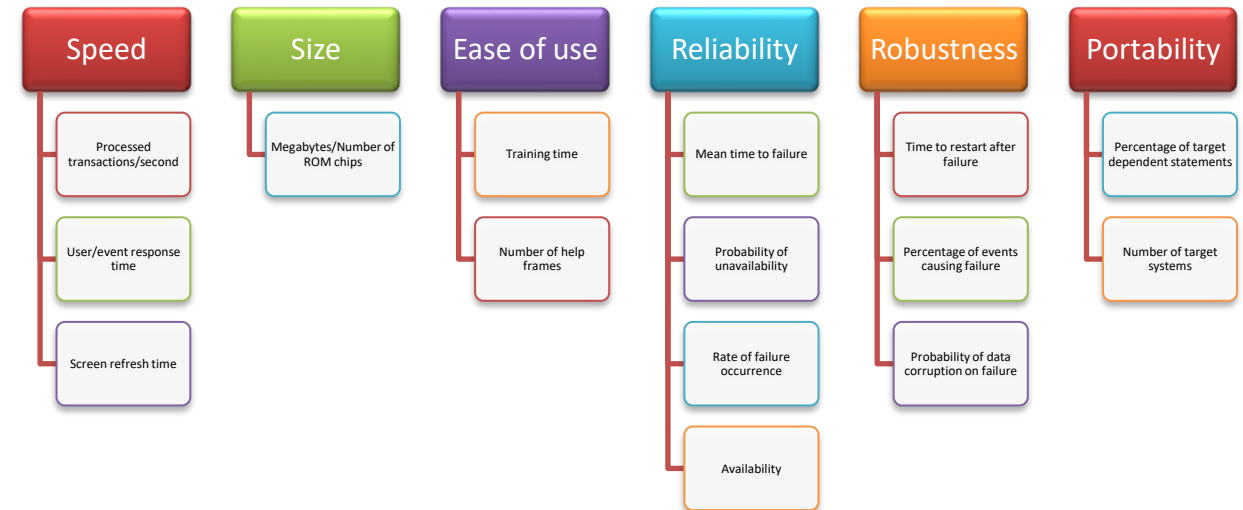 - ✓ Safety/security requirements

Non-functional requirements classification

Speed
- Processed transactions/second
- User/event response time
- Screen refresh time

Size
- Megabytes/Number of ROM chips

Ease of use
- Training time
- Number of help frames

Reliability
- Mean time to failure
- Probability of unavailability
- Rate of failure occurrence
- Availability

Robustness
- Time to restart after failure
- Percentage of events causing failure
- Probability of data corruption on failure

Portability
- Percentage of target dependent statements
- Number of target systems

Adapted from: Sommerville, I. Software Engineering. 10th edition, Published by Pearson Education, ISBN: 978-1-292-09613-1, 2016, pp. 110

Metrics for specifying non-functional requirements

- The table presents metrics for specifying non-functional system properties.

- It gives the possibility for measurement of the properties (characteristics) when the system is being tested to verify that the system has met its non-functional requirements.

Speed	Size	Ease of use	Reliability	Robustness	Portability
Processed transactions/second	Megabytes/Number of ROM chips	Training time	Mean time to failure	Time to restart after failure	Percentage of target dependent statements
User/event response time		Number of help frames	Probability of unavailability	Percentage of events causing failure	Number of target systems
Screen refresh time			Rate of failure occurrence	Probability of data corruption on failure	
			Availability		

Adapted from: Sommerville, I. Software Engineering. 10th edition, Published by Pearson Education, ISBN: 978-1-292-09613-1, 2016, pp. 110

Requirements engineering process

The engineering processes of the requirements may differ, but all of them have the following characteristic activities:

- *Identification of requirements*

- *Analysis and specification of requirements*

- *Validation of requirements*

- *Requirements management*

Back to Contents

Requirements engineering process

Identification of requirements

- The technical staff and the clients take part in this activity. The main goal is to describe in maximum detail what the software system expects to do. In addition to customers, this information may be collected based on other similar systems or information about the area in which the product will operate.

Requirements engineering process

Analysis and specification of requirements

- Usually, customer requirements are in colloquial (non-formal) language and there can be many ambiguities and inconsistencies. Clients and external sources of information can again participate in the analysis process, and this time the goal is to detail and specify the requirements, as well as to clear the contradictions if any.

Requirements engineering process

Analysis and specification of requirements

- In the analysis of the requirements, the priority of the requirements is determined and they are grouped according to different criteria. The requirements analysis results in many documents that describe the requirements in technical language using diagrams or unambiguous terminology.

Requirements engineering process

Validation of requirements

- This activity is interested in whether the system described by the discovered and analyzed requirements is what the user really wants. Requirement level error can cost much more than any implementation error.
- What is monitored in this activity is:
  - ✓ Validity
  - ✓ Consistency
  - ✓ Completeness
  - ✓ Realism
  - ✓ Verifiability

Requirements engineering process

Validation of requirements

- *Validity* - whether these are the functions that the user needs.
- *Consistency* - whether there are no conflicts between the individual requirements.
- *Completeness* - whether all possible requirements are included.

Back to Contents

Requirements engineering process

Validation of requirements

- *Realism* - is it possible for the requirements to be implemented within the given budget and technology. Usually, prototypes of single modules in the system are made in this activity to see how feasible what is required is.

- *Verifiability* - whether the requirements can be verified. Sample test scenarios are usually done here. Verifiability is very important because it allows us to verify that the resulting system does what we want

Requirements engineering process

Requirements management

- In the life cycle of a system, it is often necessary to make changes to it, either as a result of changing customer requirements or as a result of the limitations of the technologies used.

- Any sufficiently large change in the way the system operates should be reflected in the documents describing the requirements for it.

- The activity of requirements management is precisely to ensure that the documents describing the requirements are in sync with the current desired state of the system.

Back to Contents

Requirements engineering process and stakeholders

In the Requirements Engineering Process are involved many people (stakeholders) who have different roles

| Clients | Programmers | Testers | Requirements analyzers | Management and others |

Back to Contents

Requirements engineering process and stakeholders

Clients

- Customers are the people who demand the creation of the system. This includes the management of the company (if we have a product for the market), the customer who makes the order (if a product is made for a specific customer), and the users of the product.

- What they all have in common is that they understand the product on their own terms. They know what they want the product to do, but they have no technical knowledge of how it can be realized, nor the limitations imposed by the technology used.

Back to Contents

Requirements engineering process and stakeholders

Programmers

- These are the people who develop the system.
- They know the technology and its limitations and capabilities very well but rarely understand customer terminology.
- This includes hardware professionals who know the capabilities of the physical machines on which the system will run.
- These are the people who care about the realism of the requirements

Requirements engineering process and stakeholders

Testers

- These are the people who deal with the verifiability of the requirements.
- At the stage of product development, these will be the people who will monitor whether the resulting product meets the described requirements.

Requirements engineering process and stakeholders

Requirements analyzers

- These are the people who drive the process of detecting and analyzing requirements.
- Their main role is to produce documents describing customer requirements in a language accessible to technical staff (programmers and testers), as well as to ensure realism, completeness, and consistency of requirements.

Requirements engineering process and stakeholders

Management

- These are the people who manage the requirements analysis process.

- They control and specify the process itself - how many iterations of analysis and validation to perform or under what conditions the requirements are considered sufficient to move to the next stage of the product life cycle.

Software Requirements Specification or (SRS)

- **Software Requirements Specification or (SRS)** is the software requirements document

- It presents an official document between the stakeholders, which is followed and implemented by the developers of the system.

- It includes the user requirements for a system and a detailed specification of the system requirements

Software requirements specification structure

- The possible structure of a Software Requirements Specification may include the following chapters:
  - Preface
  - Introduction
  - Glossary
  - User requirements definition
  - System architecture
  - System requirements specification

Software requirements specification structure

- System models

- System evolution

- Appendices

- Index

More detailed information for the structure of the software requirement document you can find in the book *Sommerville, I. Software Engineering. 10th edition, Published by Pearson Education, ISBN: 978-1-292-09613-1, 2016, pp. 127*

Back to Contents

Users of software requirements specification

According to Ian Sommerville the possible users of the software requirements specification documents are divided into the following groups:

- System customers
- Managers
- System engineers
- System test engineers
- System maintenance engineers

References

- Sommerville, I. Software Engineering. 10th edition, Published by Pearson Education, ISBN: 978-1-292-09613-1, (2016)

- Pressman, R., Maxim, B. Software Engineering: A Practitioner's Approach. 9th edition, Published by McGraw-Hill Education, ISBN: 9781260548006, (2019)

- Dick J., Hull E., Jackson K. Introduction. In: Requirements Engineering. Springer, Cham. https://doi.org/10.1007/978-3-319-61073-3_1, ISBN: 978-3-319-61073-3, (2017)

CONTACTS

Coordinator:
Technical University of Sofia

Project coordinator:
assoc. prof. Angel Terziev, PhD
aterziev@tu-sofia.bg

Web-site: ICT-TEX.eu

Author:
Assistant professor Yavor Dankov
Sofia University "St. Kliment Ohridski"

Email: yavor.dankov@fmi.uni-sofia.bg
ResearchGate: https://www.researchgate.net/profile/Yavor-Dankov
Scopus: https://www.scopus.com/authid/detail.uri?authorId=57202891597