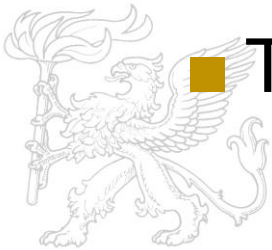


Quality criteria for documents

- ▶ The most important criteria (suggested by IEEE 830) are
 - Unambiguity and consistency
 - Clear structure
 - Modifiability and extensibility
 - Completeness
 - Traceability



Unambiguity and consistency

- ▶ Each requirement itself have to be clear and consistent
- ▶ The requirements must not to be contradictory
- ▶ Uniqueness, unique identification is important in requirements documentations



Clear structure

- ▶ The clear structure allow the reader to skip parts that are not relevant to him
- ▶ Speeds up searching
- ▶ Use standard structures



Modifiability and extendibility

- ▶ Modifying and extending in requirement is often done
 - It should be done easily
- ▶ Version control is recommended



Completeness

▶ Content factors

■ Set of all

- Include all important requirement

■ Individual

- All requirements must be documented completely

▶ Formal factors

■ Label graphs, charts and tables

■ Consistent references, indexes

■ Glossaries of used terms



Traceability

- ▶ Relationships between requirements documents and other documents is important
 - Requirement ↔ source code
 - Requirement ↔ test cases
- ▶ Supports change management



Quality criteria for requirements

- ▶ Document criteria can be used for individual requirements, but other criteria can also be defined
- ▶ Criteria for requirements
 - Agreed
 - Ranked
 - Unambiguous
 - Valid and up-to-date
 - Correct
 - Consistent
 - Verifiable
 - Realizable
 - Traceable
 - Complete



Quality criteria for requirements

- ▶ There are two fundamental rules that enhance the readability of requirements
 - Short sentences, short paragraphs,
 - Formulate only one requirement per sentence



Glossary

- ▶ A glossary is the collection of term definitions
 - Context-specific technical terms
 - Abbreviations and acronyms
 - Everyday concepts that have special meaning in the given context
 - Synonyms
 - Homonyms
- ▶ Increase understandability
- ▶ Avoid conflicts arisen from different interpretation of terms
- ▶ Glossaries are usually not project-specific



Rules of using glossaries

- ▶ The glossary must be ...
 - ... must be managed centrally
 - ... must be maintained during the whole project
 - ... must be commonly accessible
 - ... should contain the sources of the terms
- ▶ The responsibilities for maintaining the glossary must be clearly defined
- ▶ Use must be obligatory
- ▶ The stakeholders should agree upon the glossary
- ▶ The entries should have a consistent structure



Quality Aspects

- ▶ Quality criteria help to check and evaluate requirements systematically
- ▶ Quality aspects of validation
 - Content
 - All requirements are elicited and documented appropriately
 - Documentation
 - The documentation is formatted correctly (according to guidelines)
 - Agreement
 - All stakeholders understand and agree on requirements
- ▶ All aspects must be addressed by different quality criteria



Q.A. „Content”



- ▶ Validation of reqs. against errors in the content
- ▶ Error types to be addressed
 - Completeness
 - Set of all requirements
 - Individual requirements
 - Traceability
 - Correctness/adequacy
 - Consistency
 - No design decisions
 - Verifiability
 - Necessity





Q.A. „Documentation”

- ▶ Risks of ignoring documentation format
 - Impairment of development activities
 - Misunderstanding
 - Incompleteness
 - Overlooking requirements
- ▶ Error types to be addressed
 - Conformity to documentation
 - format
 - structures
 - rules
 - Understandability
 - Unambiguity



Perspective-based reading

- ▶ Requirements are reviewed from different viewpoints
- ▶ Each reviewers is assigned a perspective / viewpoint
 - Choose the right person (e.g. an expert)
 - Lay down detailed instructions for the validation (e.g. checklists, questions about document)
- ▶ Can be applied within a review process, or used alone
 - Results should be analyzed and consolidated (e.g. in the collection and consolidation phase of a review)



Perspective-based reading

▶ Possible perspectives

- User / customer
 - Describe desired functionalities
- Software architect
 - Enough information for architecture design
- Tester
 - Enough information for test case design

▶ Quality perspectives

- Content
- Documentation
- Agreement



Checklists

- ▶ A checklist
 - is a set of questions about certain circumstances that help detecting errors
 - is particularly useful when no steps should be omitted
- ▶ Can be used with previously introduced techniques
- ▶ To create checklists the following sources can be used
 - Quality aspects
 - Requirements validation principles
 - Quality criteria for documents
 - Quality criteria for individual requirements
 - Auditors' requirements engineering experience
 - Error statistics, taxonomies



Checklists

- ▶ Seek for checklist improvement
 - Add / remove questions
 - Correct ambiguous questions
- ▶ Using checklists
 - As guidelines with optional questions
 - Strictly follow with obligatory questions
 - Systematic check, can be quantitatively measured
 - Hybrid form
- ▶ Keys of success
 - Manageability and complexity of checklists
 - Short lists (maximum 1 page advisable)
 - Use specific questions instead of general and abstract ones

