Requirement Engineering

Lecture 1: Introduction

Prof. Dr. Benjamin Leiding
M.Sc. Anant Sujatanagarjuna

Lecture 1: Introduction

Content

1. Motivation for Requirements Engineering
2. Requirements Engineering Overview
3. Requirements Engineering Process

MOTIVATION OF REQUIREMENTS ENGINEERING

Motivation for Requirements Engineering

Why Requirements Engineering

- Someone needs software for a professional activity or as part of a product
  - Most people cannot create this software!

- Option 1: Buy a fitting software product
- Option 2: Pay for the development of a new software
- For both options the requirements must be known!

Motivation for Requirements Engineering

Requirements Vital for Project Success

		% of Responses
1.	**Incomplete Requirements**	13.1%
2.	Lack of User Involvement	12.4%
3.	Lack of Ressources	10.6%
4.	Unrealistic Expectations	9.9%
5.	Lack of Executive Support	9.3%
6.	**Changing Requirements & Specifications**	8.7%
7.	Lack of Planning	8.1%
8.	Didn‘t Need It Any Longer	7.5%
9.	Lack of IT Management	6.2%
10.	Technology Illiteracy	4.3%
	Other	9.9%

The Standish Group (1995) – Chaos Report –https://personal.utdallas.edu/~chung/SYSM6309/chaos_report.pdf

Motivation for Requirements Engineering

Effects of Inadequate RE – Airbus

- Requirement: „Reverse thrust may only be used, *when the airplane is landed*.“
- Translation: „Reverse thrust may only be used while the wheels are rotating.“
- Implementation: „Reverse thrust may only be used while the wheels are rotating fast enough.“
- Situation: Rainstorm – aquaplaning
- Result: Crash due to overshooting the runway!
- Problem: Erroneous modeling in the requirement phase

Motivation for Requirements Engineering

Effects of Inadequate RE – General Examples

- Missing requirements
  - “Of course, we need to print reports…”
- Inadequate requirements
  - “Provide an optimal delivery route for each truck within 1 msec”
- Implicit requirements which are not explicitly available
  - “Only one train may be in a specific railway segment at the same time”
- Inconsistent requirements
  - “only approved personal may be allowed to menu level 2” **but**
  - “in order to get approval one needs to use level-2 function request approval”
- Ambiguous requirements
  - “After inserting the card and the PIN provide access to the menu within 2 sec.”

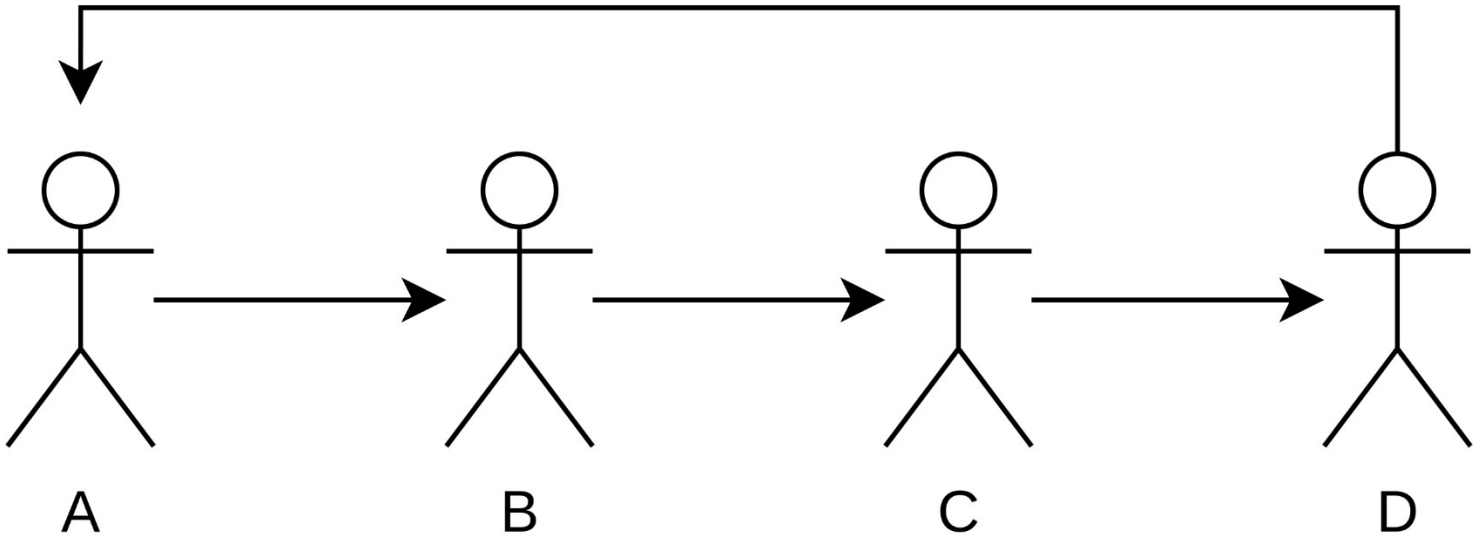
Motivation for Requirements Engineering

Tasks of Requirements

- Show what results the stakeholders want
  - A stakeholder of a system is a person or organization that has an (direct or indirect) influence on the requirements of a system.
- Represent different viewpoints
- Accept products against precise criteria
  - Requests for proposals and contract structuring
- Communication between stakeholders and developers
- Common understanding of desired product

Motivation for Requirements Engineering

Getting the Right Information is Tricky – Telephone Game



Youtube → Link

Motivation for Requirements Engineering

Challenges for the System Analysis

- Unclear objectives
  - Often multiple stakeholders
  - Bad coordination between stakeholders
  - Low imagination
- High complexity
  - No individual knows every detail of the desired product
  - Complex business processes, boundaries, rules, and wishes
- Language barriers
  - Native speakers vs. foreign language
  - Professional jargon vs. computer science jargon

Motivation for Requirements Engineering

Challenges for the System Analysis

- Changing requirements
  - Vague requirements get more detailed during the development
  - Business process changes
- Bad quality of the requirements
  - Imprecise, ambiguous, inconsistent
- Unnecessary features
  - Gold plating: functions and features that are not required are part of the system definition
- Imprecise planning
  - Results from the problems above

Motivation for Requirements Engineering

Fundamentals of Communication Theory

- Requirements must be communicated

- Mostly natural language
  - What a person (sender) says or writes (encodes) is not necessarily the same as what another person (receiver) understands (decodes)

Motivation for Requirements Engineering

Fundamentals of Communication Theory

- Communication in natural language depends on several factors
  - Cultural background
  - Educational background
  - Area of expertise
  - Everyday work life
  - Communication medium
  - ...

Motivation for Requirements Engineering

Fundamentals of Communication Theory

- Different communication media have different properties
  - Verbal communication
    - Relies heavily on redundancy, e.g., language, gestures or intonation
    - Immediate feedback possible
  - Written communication
    - Minimum of redundancy and feedback

Motivation for Requirements Engineering

Fundamentals of Communication Theory

- Sometimes required information is not transferred at all
  - Focusing
    - Certain information is left out due to a wrong/misguided focus
  - Simplification
    - Complex parts of the information are excluded
    - Oversimplified language use
    - Wrong expectation of existing knowledge
- Agreed upon common language usage improves communication
  - Usually achieved through a glossary

Motivation for Requirements Engineering

Why are Software Requirements Special?

- Software is different than hardware/materials!
  - Universal: almost no restrictions of the area of application
    - Requirements also have almost no bounds
    - Same means for many areas of application
    - Many things are taken for granted
  - Amorphous: software has no shape, cannot be visualized
  - Not monotone: Problems can always occur
    - If 3 and 5 work, 4 can still fail
  - Users and customers think anything is possible
    - Partially true
    - More possibilities means that requirements need to be detailed!

REQUIREMENTS ENGINEERING OVERVIEW

Requirements Engineering Overview

Definition – Software

Software is a collection of computer programs, procedures, directives, associated documentation and data.

IEEE 729-1983 (1983) –https://standards.ieee.org/ieee/729/967/

Requirements Engineering Overview

Definition – Requirements

The IEEE defines requirements as follows (IEEE Std. 610.12-1990):

1. A condition or capability needed by a user to solve a problem or achieve an objective.
2. 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 documents.
3. A documented representation of a condition or capability as in 1) or 2).

https://ieeexplore.ieee.org/document/159342

Requirements Engineering Overview

Definition – Requirements Engineering

"Requirements engineering is a systematic and disciplined approach to the specification and management of requirements with the following goals:

1. Knowing the relevant requirements, achieving a consensus among the stakeholders about these requirements, documenting them according to given standards, and managing them systematically.

2. Understanding and documenting the stakeholders' desires and needs, specifying and managing requirements to minimize the risk of delivering a system that does not meet the stakeholders' desires and needs."

K. Pohl, C. Rupp (2011) – Requirements Engineering Fundamentals: A study guide for Requirements Engineering Foundation Level

Requirements Engineering Overview

Four Core Activities of Requirements Engineering

1. Elicitation
   - Obtain requirements from stakeholders and other sources
   - Refinement of the requirements
2. Documentation
   - Adequate descriptions of elicited requirements.
   - Different techniques, e.g., natural language or conceptual models
3. Validation and negotiation
   - Validation of documented requirements and possibly their negotiation
   - Happens as early as possible

Requirements Engineering Overview

Four Core Activities of Requirements Engineering

4. Management
   - Orthogonal to the other activities
   - Consists of measures for
     - structuring requirements
     - preparing them for use in different roles
     - maintaining consistency after changes
     - ensuring their implementation

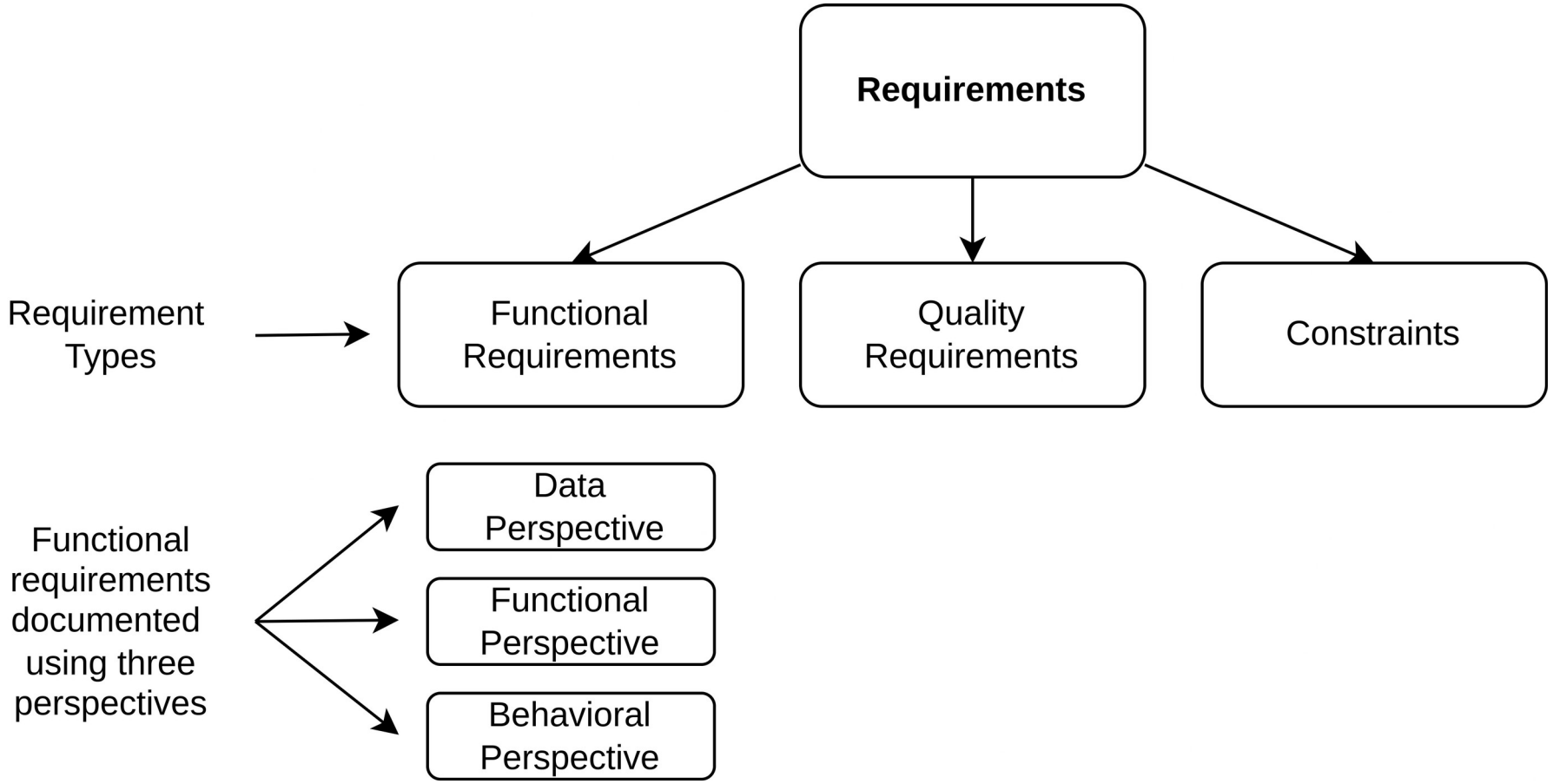
Requirements Engineering Overview

Requirement Types

- Different types of requirements
- Example:
  - Consider a calculator that should be able to perform basic arithmetic operations.
    - Which operations should be supported (e.g., add, subtract, multiply)?
    - How fast should the calculations be (e.g., 10 milliseconds, 1 second)?
    - What kinds of numbers should be supported (e.g., integer, floats)?

Requirements Engineering Overview

Requirement Types



Requirements Engineering Overview

Requirement Types – Functional Requirements

- Functional requirements
  - A functional requirement is a requirement concerning a result of behavior that shall be provided by a function of the system
  - Often has many perspectives
    - Functional perspectives
    - Behavioral perspectives
    - Data perspectives
  - Example:
    - The calculator must be able to read numbers as input.
    - The calculator must be able to add two numbers and display the result.

Requirements Engineering Overview

Requirement Types – Quality Requirements

- Quality requirements
  - A quality requirement is a requirement that pertains to a quality concern that is not covered by functional requirements
  - Typically about performance, availability, dependability, scalability, or portability of a system
  - Often called “non-functional requirements”
  - Example:
    - The result of any calculation must be provided within 10 milliseconds.
    - In average, the calculator must not crash more often than every 10,000 arithmetical operations.

Requirements Engineering Overview

Requirement Types – Quality Requirements

- Further categorization of quality requirements
  - For example, ISO Standard 9126
- Quality of system functions
  - Appropriateness, security and safety, accurateness of calculations, interoperability, conformity to standards, ...
- Dependability of functionalities
  - Robustness, fault tolerance, recoverability, ...
- Usability of a system
  - Understandability, learnability, ease of use, ...
- System efficiency
  - Time behavior, consumption behavior, ...

Requirements Engineering Overview

Requirement Types – Quality Requirements

- Changeability of a system
  - Analyzability, changeability, stability, testability, ...
- Portability of a system
  - Adaptability, installability, replaceability, ...

- Quality requirements often related to multiple functional requirements
  - Should not be mixed
  - Relationships should be well documented

Requirements Engineering Overview

Requirement Types – Constraints

- Constraints
  - A constraint is a requirement that limits the solution space beyond what is necessary for meeting the given functional requirements and quality requirements.
  - Cannot be influenced by the development team
  - Constraints are not implemented; they are adhered to
    - The constraint is not part of the solution, it simply limits how the solution will look like.
  - Example:
    - The calculator shall be implemented on hardware that allows double-precision floating point operations.
    - The calculator shall be available on the market in June 2023.

Requirements Engineering Overview

Definition – Stakeholder

"A stakeholder is either a person or an organisation that has a potential interest in the system to be developed.

A stakeholder typically has their own requirements for the system.

A person can represent the interest of different stakeholders (people and/or organisations), i.e a stakeholder can have more than one role and represent more than one stakeholder."

K. Pohl (2010) – Requirements Engineering – Fundamentals, Principles and Techniques

Requirements Engineering Overview

Stakeholder – Examples

- Customers
- System/software developers
- System users
- Architects
- Domain experts
- Testers
- Maintenance staff
- Etc.

REQUIREMENTS ENGINEERING PROCESS

Requirements Engineering Process

Overview

<table>
<tr><td colspan="6">Requirements Engineering</td></tr>
<tr><td colspan="4">Requirements Analysis</td><td colspan="2">Requirements Management</td></tr>
<tr><td>Elicitation</td><td>Negotiation</td><td>Documentation</td><td>Validation</td><td>Change Management</td><td>Tracing</td></tr>
</table>

Requirements Engineering Process

Elicitation

- Start of a project
  - Content roughly known
- Tasks:
  - Identification of stakeholders
  - Identification of additional sources of requirements
    - For example, existing systems, standards, etc.
  - Gathering of raw requirements
    - Need further refinement, but already capture the “core” of the requirements

Requirements Engineering Process

Elicitation

- Approach:
  - Visit and interview customer
    - Only few people involved
    - Documents and important names are retrieved
  - Evaluate results and determine open questions
  - Ask targeted questions in interviews
    - Customer or other stakeholders are asked
    - Possibly in form of a workshop

Requirements Engineering Process

Analysis & Negotiation

- Tasks:
  - Identification of the concrete requirements
  - Structuring of the requirements
    - Identification of relationships
    - Classification of requirements (e.g., functional requirement)
    - Merging of similar requirements
    - Grouping of requirements
      - Based on relationships
      - Based on requirement type (e.g., functional requirements, quality requirements)
  - Refinement of the requirements
    - From raw requirements to detailed requirements sufficient that can be the basis of an acceptance test
  - Identification of dependencies
  - Detection of inconsistencies
  - Resolution of inconsistencies
  - Prioritization of requirements
    - For example, in must-have requirements and optional requirements

Requirements Engineering Process

Analysis & Negotiation

- Approach:
  - Study records of meeting with customer and additional available material
  - Ask questions if needed (back to elicitation)
  - Resolve ambiguities off-line
    - For example, through a phone call
    - In case of contradictions → Negotiation
  - Parallel: Writing of the Specification (→ Documentation)
    - Called “Draft” while it is a work-in-progress
  - Distribution of specification draft
  - Workshop with all “important” people
    - Stakeholders, project management, software architects
    - Often people who can actually make decisions absent
  - Presentation of the obtained requirements
    - Slides, mock-ups
    - Interactive prioritization and concretization
      - In case of disagreement, direct mediation is possible
    - Intensive record keeping required

Requirements Engineering Process

Documentation

- Tasks:
  - Specification of requirements
    - Through the documentation, the requirements are fixed
  - Documentation of intermediate results and assumptions
  - Documentation of reasoning for requirements
  - Assignment of attributes to requirements

Requirements Engineering Process

Documentation

- Approach:
  - Writing driven by employees
    - Usually iterative and often incremental process
  - Multiple drafts, each with more information and details
  - A lot of copy-and-paste between drafts
    - Uses the results of the workshops and interviews with customers
    - In practice often a mix of natural language, tables, use cases and UML
    - Documents have long appendixes
  - Requirements Specification V1.0 a lot longer than the drafts
    - Better style, more technical
    - Reasoning behind requirements almost completely removed

Requirements Engineering Process

Validation

- Tasks:
  - Validation of the content of the specification documents
  - If possible, a formal verification of the documents
  - Verification and validation against previously existing documents (e.g., request for proposals, documentation of a legacy system) and customer wishes

Requirements Engineering Process

Validation

- Approach:
  - Documents are reviewed by experienced employees
    - Often less experienced people involved in writing the requirements
    - Checking of formal guidelines through templates
  - Checking if the customers wishes are met by the document
    - Based on the memory of the participants in workshops/interviews
    - Especially check for wishes that seem very important
  - Sometimes: Produce a prototype

Requirements Engineering Process

Change Management

- Tasks:
  - Management of change requests
  - Management of different versions of requirements
    - Each change to a requirement yields a new version
  - Well organized propagation of changes

Requirements Engineering Process

Change Management

- Approach:
  - Informed and competent decision about change requests
    - Change request → Change Control Board → Decision → Assign change task (costs money)
    - Often, this is done rather informally
  - No change control board, no clear decision making process
  - In case a RE tool is used:
    - Change and version control automatically → the change requests and changes themselves become traceable objects within the system
    - Propagation depends on the organization
  - In case no RE tool is used:
    - Changes are introduced manually
  - Changes are usually no traceable objects themselves

Requirements Engineering Process

Tracing

- Tasks:
  - Record assumptions
  - Record decisions
  - Record which assumptions lead to which decisions and how the requirements were influenced

Requirements Engineering Process

Tracing

- Approach:
  - Requires RE tool to be effective
    - Usually not available!
  - Manual tracing is a lot of work
    - Requires searching in documents and protocols
    - “In the meeting XYZ, Mr. Smith said that we should ...”

SUMMARY

Summary

- System analysis has many challenges
  - Requirements engineering is a complex task
- Requirements are the foundation of projects
  - Without good requirements, projects are in trouble
- Requirements engineering is more than just “getting the requirements and writing them down”
  - Tasks include elicitation, documentation, but also validation of requirements, change management and tracing of requirements

Questions?