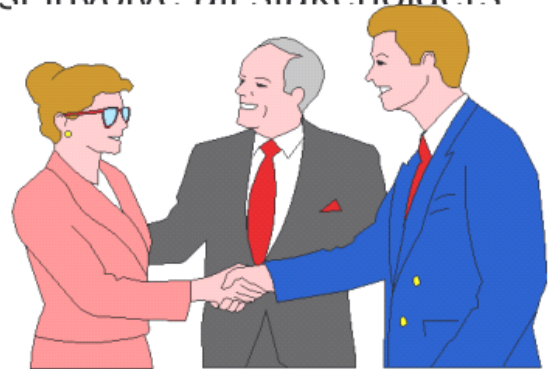


Requirements

31 de outubro de 2023 22:08

- High-level description of what a system should do
- Must be detailed enough to distinguish between the “right” and the “wrong” system
- Capture the **what** not the **how**
- The specification process must involve all stakeholders
 - Customers
 - Engineers
 - Regulatory agencies
 - Users



- The Engineering Argument
 - Engineering is about developing solutions to problems
 - A good solution can only be developed if the engineers have a solid understanding of the problem
- The Economic Argument
 - Errors cost more to correct the longer they go undetected
 - Cost of correcting requirements errors is (at least) 100 times more in the maintenance phase than in the requirements phase
- The Empirical Argument
 - Failure to understand and manage requirements is the biggest single cause of cost and schedule over-runs

- Requirements capture what a proposed system shall do
 - But avoids design detail as much as possible
 - Written in the user's language

■ Poor requirements are the source of all evil

- Requirements problems are the
 - Most costly
 - Most difficult to correct (they are conceptual)

o service
o constraint

- Market Requirements Definition (MRD)
 - Statements in natural language (plus diagrams) of the services the system provides and its operational constraints
 - Written for customers in their language

Modelo de
laminas, + use
cases

■ Software Requirements Specification (SRS)

- A structured document setting out detailed descriptions of the system services
- Written as a contract between client and contractor

Requisitos funcionais
e não funcionais

■ Software Design Description (high-level design)

- A detailed software description which can serve as a basis for a design or implementation
- Written for developers

mockups

Definitions and Specifications

Market Requirements Definition

1. One person must be able to load the boat on the car rack



(Software) Requirements Specification

- 1.1 The boat must be lighter than 100 lb.
- 1.2 The boat must have handles to help one person lift it
- 1.3 The car rack must be padded so the boat can easily slide into the rack
- 1.4 Etc.

SRS Requirements

- Functionality
 - ↳ For what or why?
- External interfaces
 - ↳ How does it interact with people?
- Performance
 - ↳ Speed, Recovery time, etc.
- Design constraints

■ Requirements may be functional or non-functional

- Functional requirements describe system services or functions
- Non-functional requirements is a constraint on the system or on the development process

→ Something the system must do.

→ Testable

→ "The system shall ..."

- These can be high level or low level (generally we're at high level in this class)
- High level: The system shall charge users credit cards for purchases
- Low level: The system shall validate all passwords contain upper and lowercase characters and one number

■ Requirements must do **ONE THING.**

– Bad:

- The system shall accept credit cards and accept pay pal

– Good:

- The system shall accept credit cards
- The system shall accept pay pal.

■ Requirements must be testable. Use precise language.

– Bad:

- The system shall work with any browser

– Good:

- The system shall work with Firefox
- The system shall work with IE

Specific

– Bad:

- The system shall work with IE
- **Bad:**
 - The system shall respond quickly to user clicks
- **Good:**
 - The system shall respond within 10ms to any user click

Non-Functional Requirements

- Define system properties and constraints
 - Reliability, response time and storage requirements
 - Constraints are I/O device capability, system representations, etc.
- Process requirements may also be specified mandating a particular CASE system, programming language, or development method
- Non-functional requirements may be more critical than functional requirements
 - If these are not met, the system is useless

↳ CLASSIFICATION:

- Product requirements
 - Requirements which specify that the delivered product must behave in a particular way
 - Execution speed, reliability, etc.
- Organizational requirements
 - Requirements which are a consequence of organizational policies and procedures
 - Process standards used, implementation requirements, etc.
- External requirements
 - Requirements which arise from factors which are external to the system and its development process
 - Interoperability requirements, legislative requirements, etc.

Requirements Template (suggestion)

Number:	A unique requirements number
Use Case:	Reference to use case using Req.
Introduction:	What is the requirement about?
Rationale:	Why is the requirement here?
Source:	Who came up with the requirement?
Author:	Who wrote it down?
Inputs:	What comes in?
Required Function:	What is the requirement?
Outputs:	What comes out?
Related Reqs:	What else is related?
Conflicts:	Requirements in conflict with this one
Support Material:	Docs., Figures, Tables, Etc.
Test Cases:	How do we test the requirement?
Date:	When the requirement was modified
Priority:	How important is this requirement

Each Requirement Must Be

- Correct
 - The requirement is free from faults.
- Precise, unambiguous, and clear
 - Each item is exact and not vague; there is a single interpretation; the meaning of each item is understood; the specification is easy to read.
- Complete
 - The requirement covers all aspects of the user function.
- Consistent
 - No item conflicts with another item in the specification.

- Relevant
 - Each item is pertinent to the problem and its solution.
- Testable
 - During program development and acceptance testing, it will be possible to determine whether the item has been satisfied.
- Traceable
 - Each item can be traced to its origin in the problem environment.
- Feasible
 - Each item can be implemented with the available techniques, tools, resources, and personnel, and within the specified cost and schedule constraints

1. Validate charge card number

1.1 The system shall validate the charge card number entered against the on-line master corporate charge card number list.

1.1.1 If the charge card number is not found on the list, an error message shall be displayed and the order shall not be accepted

1.2 If the on-line master corporate charge card number list is not available, the order shall not be accepted.

Requirements Traceability

- Requirements traceability means that related requirements are linked in some way and that requirements are (perhaps) linked to their source
- Traceability is a property of a requirements specification which reflects the ease of finding related requirements
- Some CASE tools provide traceability support facilities
 - For example, they may be able to find all requirements which use the same terms
- Assign a unique number to all requirements
- Cross-reference related requirements using this unique number
- Produce a cross-reference matrix for each requirements document showing related requirements.
 - Several matrices may be necessary for different types of relationship