

Software Requirements

Requirements

1

Background

- Problem of scale is a key issue for SE
- For small scale, understand and specifying requirements is easy
- For large problem - very hard; probably the hardest, most problematic and error prone
- **Input** : user needs in minds of people
- **Output** : precise statement of what the future system will do

Requirements

2

Background..

- Identifying and specifying requirement necessarily involves people interaction
- Cannot be automated
- Req. phase ends with a software requirements specification (SRS) document
- SRS specifies what the proposed system should do

Requirements

3

Background..

- Requirements understanding is hard
 - Visualizing a future system is difficult
 - Capability of the future system not clear, hence needs not clear
 - Requirements change with time
 - ...
- Essential to do a proper analysis and specification of requirements

Requirements

4

Need for SRS

- SRS establishes basis of agreement between the user and the supplier.
 - Users needs have to be satisfied, but user may not understand software
 - Developers will develop the system, but may not know about problem domain
 - SRS is the medium to bridge the commn. gap and specify user needs in a manner both can understand

Requirements

5

Need for SRS...

- Helps user understand his needs.
 - users do not always know their needs
 - must analyze and understand the potential
 - the goal is not just to automate a manual system, but also to add value through IT
 - The req process helps clarify needs
- SRS provides a reference for validation of the final product
 - Clear understanding about what is expected.
 - Validation - " SW satisfies the SRS "

Requirements

6

Need for SRS...

- Good SRS reduces the development cost
 - SRS errors are expensive to fix later
 - Requirement changes can cost a lot (up to 40%)
 - Good SRS can minimize changes and errors
 - Substantial savings; extra effort spent during req. saves multiple times that effort
- An Example
 - Cost of fixing errors in requirement, design, coding, acceptance testing and operation are 2, 5, 15, 50, 150 person-months

Requirements

7

Need for SRS...

- Example ...
 - After req. phase 65% requirement errors detected in design, 2% in coding, 30% in Acceptance testing, 3% during operation
 - If 50 requirement errors are not removed in the req. phase, the total cost $32.5 * 5 + 1 * 15 + 15 * 50 + 1.5 * 150 = 1152$ hrs
 - If 100 person-hours invested additionally in requirement to catch these 50 defects, then development cost could be reduced by 1152 person-hours.
 - Net reduction in cost is 1052 person-hours

Requirements

8

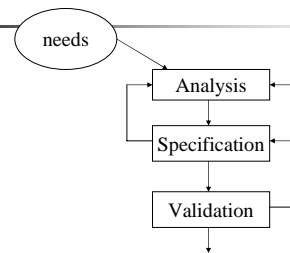
Requirements Process

- Sequence of steps that need to be performed to convert user needs into SRS
- Process has to elicit needs and requirements and clearly specifies it
- Basic activities
 - problem or requirement analysis
 - requirement specification
 - validation
- Analysis involves elicitation and is the hardest

Requirements

9

Requirements Process..



Requirements

10

Problem Analysis

- Aim: to gain an understanding of the needs, requirements, and constraints on the software
- Analysis involves
 - interviewing client and users
 - reading manuals
 - studying current systems
 - helping client/users understand new possibilities
 - Like becoming a consultant
- Must understand the working of the organization, client and users

Requirements

11

Problem Analysis...

- Some issues
 - Obtaining the necessary information
 - Brainstorming: interacting with clients to establish desired properties
 - Information organization, as large amount of info. gets collected
 - Ensuring completeness
 - Ensuring consistency
 - Avoiding internal design

Requirements

12

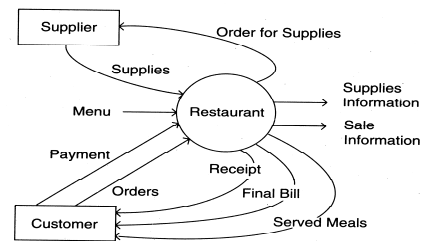
Structured Analysis Method

- Structured system analysis and design (SSAD)
- Main steps
 - Draw a context diagram
 - Draw DFD of the existing system
 - Draw DFD of the proposed system and identify the man-machine boundary

Requirements

13

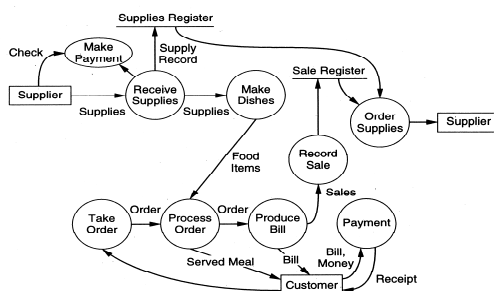
Example – context diagram



Requirements

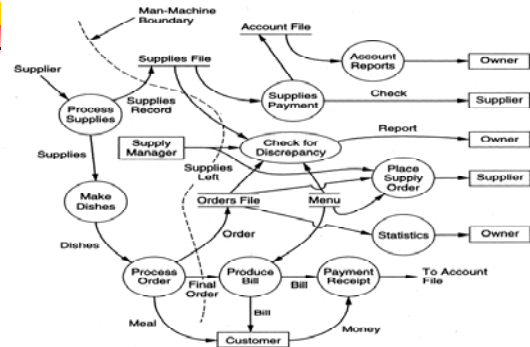
14

Example – DFD of existing system



15

Example – DFD of proposed system



6

Requirements Specification

- Final output of requirements task is the SRS
- Transition from analysis to specification is not straight forward
- knowledge about the system acquired in analysis used in specification

Requirements

17

Characteristics of an SRS

- Correct
- Complete
- Unambiguous
- Consistent
- Verifiable
- Traceable
- Modifiable
- Ranked for importance and/or stability

Requirements

18

Characteristics...

- Correctness
 - Each requirement accurately represents some desired feature in the final system
- Completeness
 - All desired features/characteristics specified
 - Hardest to satisfy
 - Completeness and correctness strongly related
- Unambiguous
 - Each requirement has exactly one meaning
 - Without this errors will creep in
 - Important as natural languages often used

Requirements

19

Characteristics...

- Verifiability
 - There must exist a cost effective way of checking if sw satisfies requirements
- Consistent
 - two requirements don't contradict each other
- Traceable
 - The origin of the req, and how the req relates to software elements can be determined
- Ranked for importance/stability
 - Needed for prioritizing in construction
 - To reduce risks due to changing requirements

Requirements

20

Components of an SRS

- An SRS must specify requirements on
 - Functionality
 - Performance
 - Design constraints
 - External interfaces

Requirements

21

Structure of an SRS

- Introduction
 - Purpose , the basic objective of the system
 - Scope of what the system is to do , not to do
 - Overview
- Overall description
 - Product perspective
 - Product functions
 - User characteristics
 - Assumptions
 - Constraints

Requirements

22

Structure of an SRS...

- Specific requirements
 - External interfaces
 - Functional requirements
 - Performance requirements
 - Design constraints
- Acceptable criteria
 - desirable to specify this up front.
- This standardization of the SRS was done by IEEE.

Requirements

23

Summary

- Having a good quality SRS is essential for Q&P
- The req. phase has 3 major sub phases
 - analysis , specification and validation
- Analysis
 - for problem understanding and modeling
 - Methods used: SSAD, OOA , Prototyping
- Key properties of an SRS: correctness, completeness, consistency, traceability, unambiguousness

Requirements

24



Summary..

- Specification
 - must contain functionality , performance , interfaces and design constraints
 - Mostly natural languages used