

DA-IICT

IT314: Software Engineering



1

Requirements

- Functional: Describe what the system should do
- Non-Functional: Constraints that must be adhered
- Domain: comes from the application domain of the system and that reflect characteristics of that domain

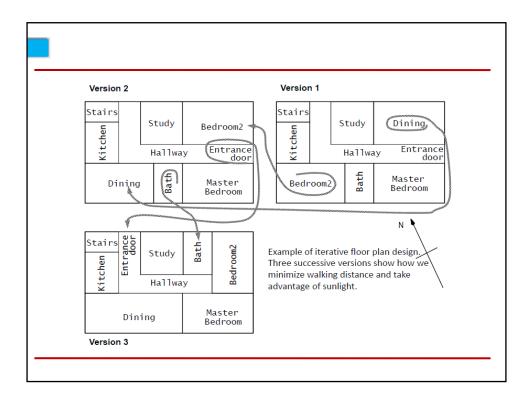
Functional Requirements

- · What inputs the system should accept
- What outputs the system should produce
- What data the system should store that other systems might use
- · What computations the system should perform
- The timing and synchronization of the above

Examples..

Non-Functional Requirements

- These define system properties and constraints e.g. reliability, response time and storage requirements. Constraints are I/O device capability, system representations, etc.
- Non-functional requirements may be more critical than functional requirements. If these are not met, the system is useless.



List of Design Goals

Modifiability Good documentation Well defined interfaces Maintainability Understandability Well-defined interfaces

User-friendliness Reuse of components

Adaptability Reusability

Reliability

Reuse of components Rapid development Minimum # of errors

Efficiency Portability Readability
Ease of learning
Ease of remembering
Traccability of requirement

Traceability of requirements

Fault tolerance

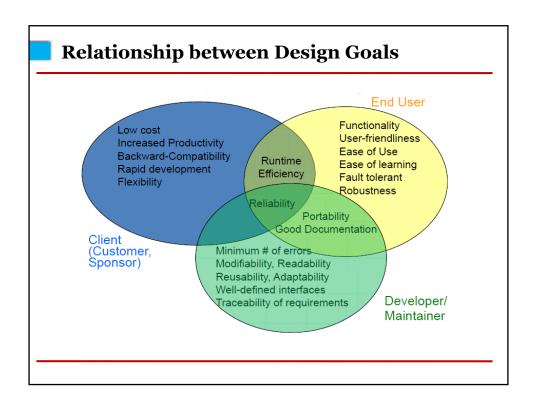
Backward-compatibility Ease of remembering

Ease of use

Increased productivity Backward compatibility Cost-effectiveness

Robustness Low-cost Flexibility

High-performance



Typical design Trade-Offs

- · Functionality vs. Usability
- · Cost vs. Robustness
- Efficiency vs. Portability (or Modifiability)
- · Rapid development vs. Functionality
- Cost vs. Reusability
- · Backward Compatibility vs. Readability

Non-Functional Testing

A part from functional testing, non-functional requirements like performance, usability, load factor are also important.

Performance testing: check and fine tune system response time. The goal here is to reduce the response time.



Loading

Non-Functional Testing

Load testing: System performance at different load i.e., number of people accessing the system



Testability of Requirements

- · Nonfunctional requirements are often not clearly defined
- How would you test:
 - "The system should be easy to operate"
 - "The system should be fast"
- Requirements should be expressed in a testable way
 - Make sure every requirement is testable
 - Make it early in the development process

Non-functional Tests

- Performance test
 - Processing speed and response time
- Load test
 - Behavior in increasing system loads
 - Number of simultaneous users
 - Number of transactions
- Stress test
 - Behavior when overloaded

Non-functional Tests (2)

- Volume test
 - Behavior dependent on the amount of data
- · Testing of security
 - Against unauthorized access
 - Service attacks
- Stability
 - Mean time between failures
 - Failure rate with a given user profile etc.



Non-functional Tests (3)

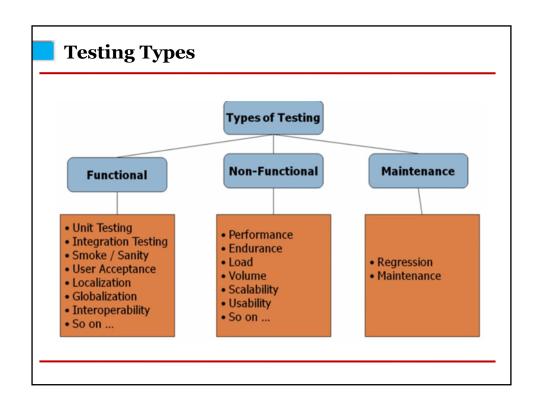
- Robustness test
 - Response
 - Examination of exception handling and recovery to errors
- · Compatibility and data conversion
 - Compatibility to given systems
 - Import/export of data



Non-functional Tests (4)

- Different configurations of the system
 - Back-to-back testing
- Usability test
 - Ease of learning the system
 - Ease and efficiency of operation
 - Understandability of the system





NFR: Classification

Product NF requirements

Requirements which specify that the delivered product must behave in a particular way e.g. execution speed, reliability, usability, performance (response time, throughput), etc.

Organizational NF requirements

Requirements which are a consequence of organizational policies and procedures e.g. process standards used, implementation requirements, etc.

External NF requirements

Requirements which arise from factors which are external to the system and its development process e.g. interoperability requirements, legislative requirements, etc.

Examples

Organizational requirement

The system development process and deliverable documents shall conform to the process and deliverables defined in XYZCo-SP-STAN-95.

External requirement

The system shall not disclose any personal information about customers apart from their name and reference number to the operators of the system

Goals and Requirements

Non-functional requirements may be very difficult to state precisely and imprecise requirements may be difficult to verify.

Goal

A general intention of the user such as ease of use.

Verifiable non-functional requirement

A statement using some measure that can be objectively tested.

Requirements Interaction

Conflicts between different non-functional requirements are common in complex systems.

Spacecraft system

To minimize weight, the number of separate chips in the system should be minimized.

To minimize power consumption, lower power chips should be used.

However, using low power chips may mean that more chips have to be used. Which is the most critical requirement?

Reliability

From the book and papers of John D. Musa

Reliability and Availability

Reliability: the probability that a system will continue to function without failure for a specified period in a specified environment. The period units.

- Natural unit: such as runs, pages of output, transactions, telephone calls, jobs, semiconductor wafers, queries, or API calls
- Failure intensity (FI): failures per natural or time unit, an alternative way of expressing reliability

Availability: the average (over time) probability that a system is currently functional in a specified environment

SRE - A Proven, Standard, Widespread Best Practice

Proven practice

Example: AT&T International Definity PBX [5, pp 167-8]

- a) Reduced customer-reported problems by factor of 10
- b) Reduced system test interval by factor of 2
- c) Reduced total development time by 30%
- d) No serious service outages in 2 years of deployment

SRE

AT&T Best Current Practice since 5/91 (based on widespread practice, documented strong benefit/cost ratio, probing review) [5, pp 219-254]

Standard practice

McGraw-Hill handbook [3, 4, 5] AIAA standard since 1993 IEEE standards

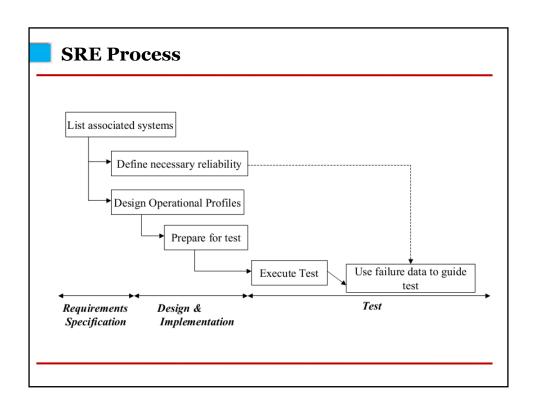
SRE

Software reliability engineering (SRE) is the prediction, measurement, and management of software intensive systems, with the purpose of achieving the highest customer satisfaction possible.

Software Reliability Engineering (SRE) is a multi-faceted discipline covering the software product lifecycle.

It involves both technical and management activities in three target areas:

- Software reliability measurement, prediction, and estimation.
- Use of reliability target to guide software development process and maintenance.
- Study of the impact of software reliability metrics and activities on operational software behavior.



Operational Profile

The operational profile of the software reflects how it will be used in practice.

The operational profile is based on the characteristics of the software environment, the functions used, their inputs and outputs, and their usage frequencies.

Defining the operational profile involves identifying major system operations, their occurrence probabilities and their initiators:

- 1. Identify initiators of operations (i.e., user types, external systems, and the system itself)
- 2. Identify and list the operations invoked by each initiator
- 3. Determine occurrence rates of the operations
- 4. Determine occurrence probabilities by dividing the occurrence rates by the total occurrence rate

Example: Fone Follower (FF)

Product Description

- 1. Subscriber calls FF, enters planned phone numbers (forwardees) to which calls are to be forwarded vs. time.
- FF forwards incoming calls (voice or fax) from network to subscriber as per program. Incomplete voice calls go to pager (if subscriber has one) and then voice mail.

Operation: major system logical task performed for initiator, which returns control to system when complete.

Illustrations - FF: Process fax call, Phone number entry, Audit section of phone number database

Operational profile (OP): complete set of operations with probabilities of occurrence

Illustration - FF:		O ccur.
	<u>O peration</u>	<u>Prob.</u>
	Process voice call, no pager, ans.	0.21
	Process voice call, pager, ans.	0.19
	Process fax call	0.17
	Process voice call, pager, ans. on page	0.13

