

System Integration & Architecture

Instructor: Maham Khan

Lecture 5

Systems Requirements

- Characteristics or features that must be included to satisfy business requirements
 - Outputs
 - Inputs
 - Processes
 - Timing
 - Controls
 - Volumes, sizes, and frequencies
- Data/Information collected can be about; people, organisation, work and work environment.

Fact – Finding Methods

- Sampling (of existing documentation, forms, and databases).
- Research and site visits. (Participation)
- Observation of the work environment.
- Questionnaires.
- Interviews.
- Prototyping.
- JAD/Joint requirements planning (JRP).

Types of Requirements

- **User Requirements:** these are statements in Natural language plus diagrams of services the system provides, together with its operational constraints. These can be categorised into 2; functional requirements and non-functional requirements

- **Functional requirements**

- Describe *what* the system should do

- **Non-functional requirements**

- Consists of **Constraints** that must be adhered to during development (design and implementation)

- Remember ‘**Constraints.**’

- **System requirements**

- What we agree to provide

- Describes system services

- Contract between Client and contractor

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

Non-functional requirements

- Non-functional requirements are global constraints on a computer system
 - e.g. development costs, operational costs, performance, reliability,
- The challenge of Non-functional requirements:
 - Hard to model
 - Usually stated informally, and so are:
 - often contradictory,
 - difficult to enforce during development
 - difficult to evaluate for the customer prior to delivery

Non-functional requirements

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

Examples of NFR

- Interface requirements
 - how will the new system interface with its environment?
 - User interfaces and “user-friendliness”
 - Interfaces with other systems
- Performance requirements
 - Time - response time
 - Throughput - transactions per second

Examples of NFR

- Security
 - permissible information flows
 - Or who can do what
 - Survivability – e.g. system will need to survive fire natural catastrophes, etc
- Operating requirements
 - physical constraints (size, weight),
 - personnel availability & skill level
 - accessibility for maintenance
 - environmental conditions

Examples of NFR

- Lifecycle requirements
 - Maintainability, Enhanciability, Portability, expected market or product lifespan
- limits on development
 - E.g. development time limitations, resource availability and methodological standards.
- Economic requirements
 - e.g. restrictions on immediate and/or long-term costs.

Requirements Documentation

- There are basically two types of documents realised from the requirements elicitation phase. These include;
 - User Requirements Specification Document
 - System requirements specification Document

User Requirements Specification – URS/URD

- The URS document outlines precisely what the **User (or customer) is expecting from this system.**
- User Requirement Specification may incorporate the **functional requirements** of the system or may be in a separate document labelled the **Functional Requirements Specification - the FRS.**

The URD has the following information:

1. Functional Requirements
2. Non-Functional Requirements

System Requirements Specification Document

A detailed description of the system services.

- What do we agree to provide?
- A structured document setting out detailed descriptions of the system services.
- Written as a contract between client and contractor.

TOOLS THAT AID IN DEVELOPING & UNDERSTANDING SYSTEM REQ'TS

- Affinity diagrams
- Force-field analysis
- Ishikawa fishbone (cause-and-effect) diagrams
- Pareto diagrams
- Pugh charts
- Quality function deployment (QFD)

Comparison of the tools

Tool	Failure Analysis	Concept Development	Best Concept	Requirements Analysis	Market Forces	Service Systems
Affinity		•		•		
Force field					•	
Ishikawa	•					
Pareto	•					
Pugh		•	•			
QFD		•		•	•	•
Functional decomposition		•		•		
Wymore		•	•	•		•
RDD-100	•			•		
Slate				•		
CORE				•		