

REQUIREMENT ANALYSIS AND SPECIFICATION

Prof. Sudip Misra

Department of Computer Science &
Engineering

Indian Institute of Technology, Kharagpur

<http://cse.iitkgp.ac.in/~smisra/>



GOALS

- To clearly understand the customer requirements.
- Remove inconsistencies, anomalies, etc. from requirements.
- Systematically organize the requirements into a specification document.



FINAL OUTCOME

- The Software Requirements Specification (SRS) document.

MAJOR ACTIVITIES

- Requirements Gathering and Analysis.
- Requirements Specification.

REQUIREMENTS GATHERING AND ANALYSIS

Requirements Gathering : Ways of gathering requirements.

- Studying the existing documentation.
- Interview.
- Task analysis.
- Scenario Analysis.
- Form Analysis.

REQUIREMENTS GATHERING AND ANALYSIS

Analysis:

- To analyze the collected information to obtain a clear understanding of the product to be developed.
- Removing all ambiguities, incompleteness, and inconsistencies from the initial customer perception of the problem.



REQUIREMENTS GATHERING AND ANALYSIS (CONT.)

Main problems in the requirements that are needed to be identified :

- Ambiguity.
- Inconsistency.
- Incompleteness.



REQUIREMENTS GATHERING AND ANALYSIS (CONT.)

Ambiguity:

- Any ambiguity in the requirements can lead to development of incorrect systems, since an ambiguous requirement can be interpreted in the several ways.

- Example:

In a process control application, a requirement expressed by one user is that when the temperature becomes high, the heater should be switched off. Here **high** is ambiguous statement.



REQUIREMENTS GATHERING AND ANALYSIS (CONT.)

Inconsistency :

Some part of the requirement contradicts with some other part.

Example:

- One customer says turn off heater and open water shower when temperature > 100 degree Celsius.
- Another customer says turn off heater and turn ON cooler when temperature > 100 degree Celsius.



REQUIREMENTS GATHERING AND ANALYSIS (CONT.)

Incompleteness :

Some requirements have been omitted due to oversight.

Example :

In the chemical plant automation software, one of the requirements is that if the internal temperature exceeds 200 degree Celsius then an alarm bell must be sounded. However, there is no provision for resetting the alarm bell in any of the requirements.



SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

- What is SRS ?

A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

- **Main aim :**
- Systematically organize the requirements arrived during requirements analysis
- Document requirements properly.



ROLE OF SRS : IEEE 830

- “The SRS must correctly define all of the software requirements, but no more.”
- “The SRS should not describe design, verification, or project management details, except for required design constraints.”



CLEAR AND COMPLETENESS OF SRS

Unclear	Clear
The system shall be able to read updates from Medical Image.	The system shall be able to import new tumor patient data supplied by Medical Image to the radiology management system, for evaluating the tumor to be malignant or benign.
The system shall be able to provide historical reports.	The system shall be able to provide patient tumor data for the past five calendar years.

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

The SRS document is useful in various contexts:

- Statement of user needs.
- Contract document.
- Reference document.
- Definition for implementation.



SOFTWARE REQUIREMENTS SPECIFICATION (SRS) : USERS

- Users, customers, and marketing personnel.
- Software developers.
- Test engineers.
- User documentation writers.
- Project managers.
- Maintenance engineers





CHARACTERISTICS OF GOOD SRS DOCUMENT

- Concise.
- Structured.
- Black box view.
- Traceable.
- Response to undesired events.
- Verifiable.
- Easy to change.
- Complete.
- Consistent.





EXAMPLES OF BAD SRS DOCUMENTS

- Over-specification.
- Forward references.
- Wishful thinking.

CATEGORIES OF CUSTOMER REQUIREMENTS

- Functional requirements.
- Non-functional requirements.
- Goals of implementation.



CATEGORIES OF CUSTOMER REQUIREMENTS:

FUNCTIONAL REQUIREMENTS

- Specify which outputs should be produced from the given inputs.
- Some are :
 - Specification of the product's functionality.
 - Actions the products must take – check, compute, record, and retrieve.
 - Derived from basic purpose of the product.
 - Normally business-oriented, rather than of the product.
 - Derived mostly from the use case scenarios.

CATEGORIES OF CUSTOMER REQUIREMENTS:

FUNCTIONAL REQUIREMENTS

- **Example :**

Input: An author's name.

Output : Details of the author's books and the locations of these books in the library.



CATEGORIES OF CUSTOMER REQUIREMENTS : NON-FUNCTIONAL REQUIREMENTS

- Non functional requirements are properties, characteristics, or qualities that a software product must have for it to do a task.
- For example, the user may want that the product be :
 - Fast.
 - Accurate.
 - User friendly.
 - Attractive.



CATEGORIES OF CUSTOMER REQUIREMENTS : NON-FUNCTIONAL REQUIREMENTS

- **Characteristics of the system which can not be expressed as functions:**

- Maintainability.
- Portability.
- Usability, etc.



CATEGORIES OF CUSTOMER REQUIREMENTS : NON-FUNCTIONAL REQUIREMENTS

- **Nonfunctional requirements include:**

- Reliability issues,
- Performance issues,
- Human-computer interface issues,
- Interface with other external systems,
- Security, maintainability, etc.



CATEGORIES OF CUSTOMER REQUIREMENTS : GOALS OF IMPLEMENTATION

- Offers some general suggestions regarding development.
- Documents, issues such as revisions to the system functionalities that may be required in the future.
- A goal is something that is desirable but would not be tested by the customer for acceptance.



A SAMPLE ORGANIZATION OF SRS DOCUMENT

1. Introduction

- Purpose
- Overview
- Environmental Characteristics:
 - Hardware
 - Peripherals
 - People

2. Goals of Implementation



A SAMPLE ORGANIZATION OF SRS DOCUMENT (CONT.)

3. Functional Requirements

- User class1
 - Functional requirement 1.1
 - Functional requirement 1.2
- User class2
 - Functional requirement 2.1
 - Functional requirement 2.2



A SAMPLE ORGANIZATION OF SRS DOCUMENT (CONT.)

4. Non-functional Requirements

- External interfaces
- User interfaces
- Software interfaces
- Communication interfaces

5. Behavioral Description

- System states
- Events and actions



SOME SAMPLE SRS DOCUMENTS

- IEEE template:



Microsoft Word
7 - 2003 Document

- Example 1 :



Adobe Acrobat
Document

- Example 2 :



Adobe Acrobat
Document

- Example 3 :



Adobe Acrobat
Document

THANK YOU

