

SOFTWARE REQUIREMENTS PROCESS PD-SE-PR-002

CHANGE HISTORY

SR NO	DATE	AUTHOR	VERSION	REMARKS
1	01-01-10	TECHSERV	1.0	FIRST RELEASE

PURPOSE:

The Purpose of this document is to describe the “SOFTWARE REQUIREMENTS” process.

It covers process:

- Objectives
- Scope
- Abbreviations used
- Inputs
- Outputs
- Tools usage
- Process Measurements and Metrics
- Activities involved
- Responsibilities
- Process aids
- Process compliance indicators

The process definition is providing both high level and detailed process flow. To supplement the process flow process aids such as Guidelines / Templates / Checklists / Standards have been provided to improve process understanding and implementation effectiveness

OBJECTIVES:

The objective of the Software Requirements process is to translate the software requirements into a clear, well formulated, and complete Software Requirements Specification Document and controlled and met throughout the product lifecycle.

A SRS has to establish the scope of the software system and identify interfaces with external systems, from the customer's point of view.

SCOPE:

This process applies to new product development projects

ABBREVIATIONS:

ST –Steering Committee

PDP – Product Development Process

SRS – Software Requirements Specification

Requirements document states *what* the software will do. It does not state *how* the software will do it.

What the software does is directly perceived by its users – either human users or other software systems. When a user performs some action, the software responds in a particular way; when an external system submits a request of a certain form, it gets a particular response. Therefore you and the users must agree on actions they can perform and response they should expect. This common understanding is captured in the requirements document. *How* the software responds to the agreed upon request is *not* addressed in the requirements document. For example, the requirements document does not include screen layouts, database schemas, descriptions of communication layers – in short, no statements of design of any sort. For example, it is a requirement for a payroll application to be able to compute and report employee earnings. It is a design issue whether to build a system to send security enabled pay slip delivery system to employee. Again it is a design issue how the privacy of the data will be met in the system

This is not to say you won't seek users' input on some of the design, most especially on user interface design, but it is very important to recognize and Why Bother with a Requirements Document?

Respect the boundary between the statement of requirements and how requirements are implemented. Design is the responsibility of the development team they should be free to choose the most appropriate way to satisfy all aspects of the requirements – features, performance, usability, etc. Usually the most appropriate way is the most simple way but sometimes other considerations may affect design decisions – such as opportunities for design or code reuse.

ENTRY CRITERIA

- APPROVED PRODUCT REQUIREMENTS

INPUTS:

- PRODUCT REQUIREMENTS DOCUMENT
- REQUIREMENTS TRACEABILITY DOCUMENT

REFERENCE DOCUMENTS:

- INTEGRATED PROJECT PLAN
- CONTRACT
- REQUIREMENTS WORKING PAPERS
- MINUTES OF MEETING
- REQUIREMENTS GUIDELINES

EXIT CRITERIA

- APPROVAL OF SOFTWARE REQUIREMENTS

OUTPUTS:

DIRECT ARTIFACTS:

- SOFTWARE REQUIREMENTS

IN-DIRECT ARTIFACTS:

- SOFTWARE REQUIREMENTS STUDY NOTES
- REVIEW RECORDS
- MEETING MINUTES
- PRODUCT REQUIREMENTS TRACEABILITY

PROCESS MEASUREMENT:

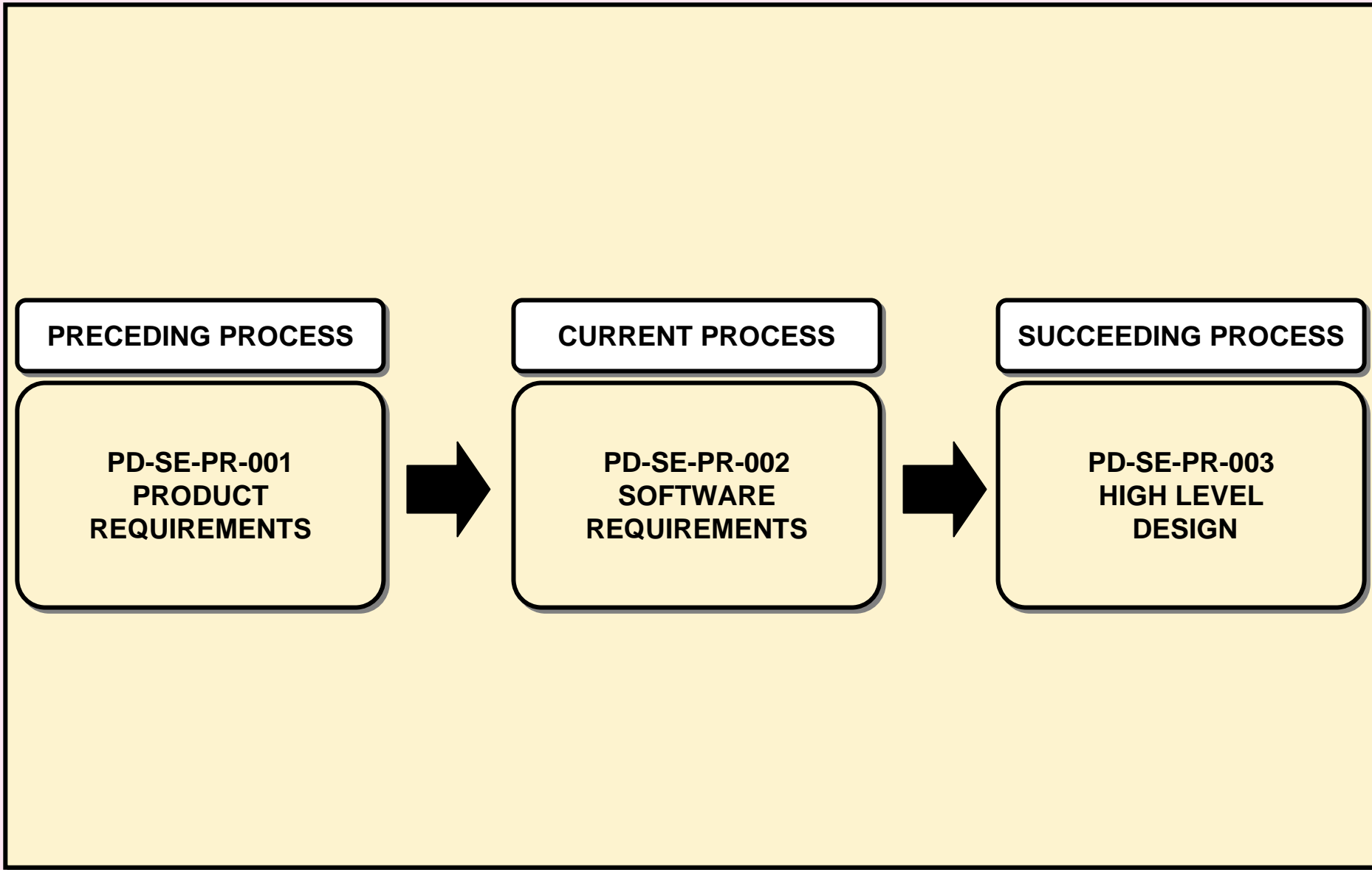
- NUMBER OF REQUIREMENTS CATEGORYWISE
 - FUNCTIONAL
 - NON-FUNCTIONAL
 - COMPLIANCE
 - INTELLUCTUAL PROPERTY
- TIME EXPENDED ON SOFTWARE REQUIREMENTS
- TIME EXPENDED FOR REVIEWS
- TIME EXPENDED FOR REWORK
- SCHEDULED DELIVERY DATE
- ACTUAL DELIVERY DATE

PROCESS METRICS:

- TOTAL EFFORT ON THIS PROCESS
- TOTAL EFFORT ON REVIEWS
- TOTAL EFFORT ON REWORK
- DELIVERY COMMITMENTS

TOOLS, TO BE USED:

- MS – WORD
- MS – EXCEL



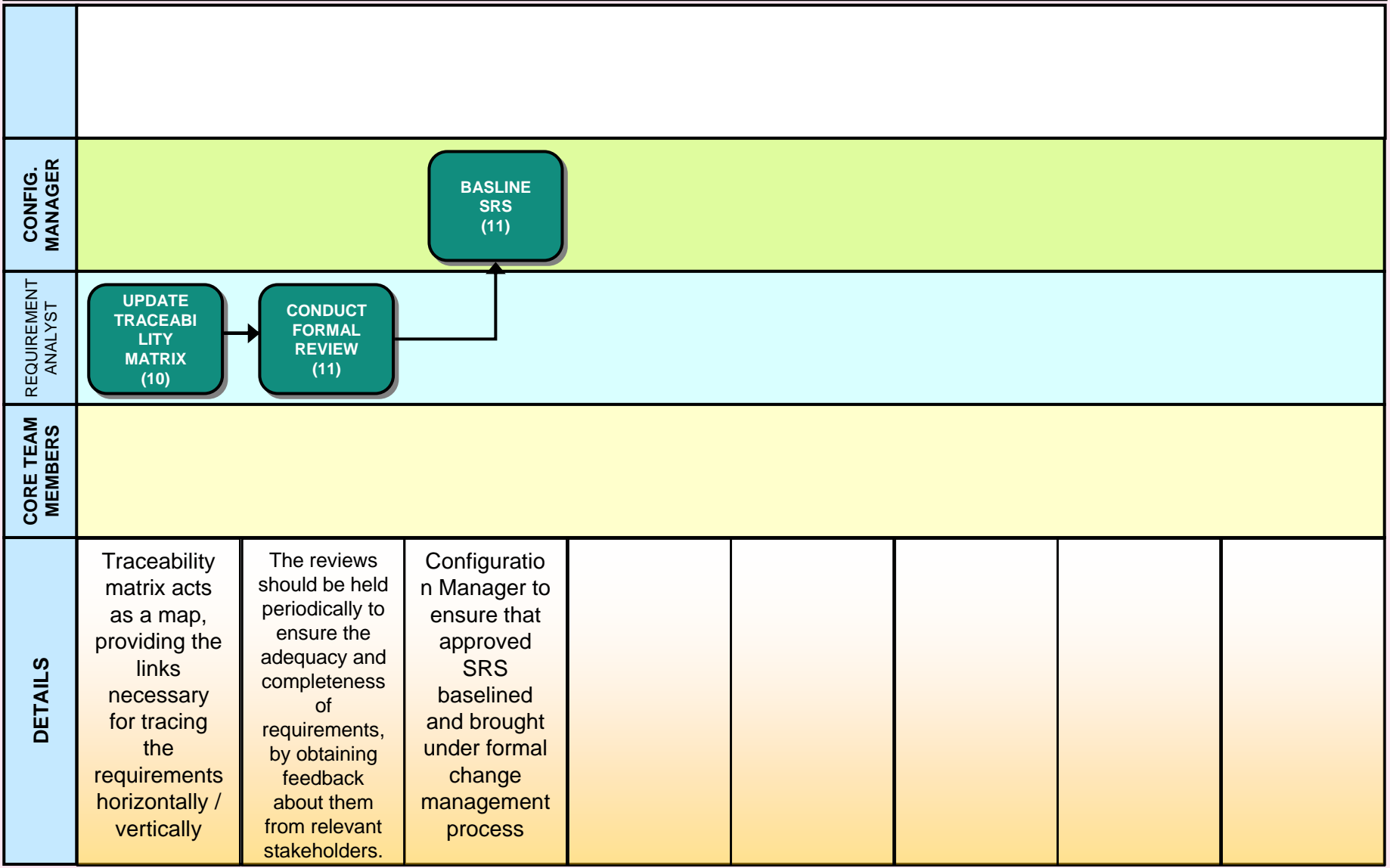
SOFTWARE REQUIREMENTS PROCESS - OVERVIEW

SOFTWARE REQUIREMENTS PROCESS - OVERVIEW								
CONFIG. MANAGER								
REQUIREMENT ANALYST	<p>PLAN SRS PROCESS (1)</p>	<p>IDENTIFY REQUIREMENTS SOURCES (2)</p>	<p>UNDERSTAND CUSTOMER NEEDS (3)</p>	<p>DEFINE AND STATE PROBLEMS (4)</p>	<p>WRITE SOFTWARE REQUIREMENTS (5)</p>	<p>ANALYZE SOFTWARE REQUIREMENTS (7)</p>	<p>VALIDATE SOFTWARE REQUIREMENTS (8)</p>	<p>DESCRIBE VERIFICATION PROCESS (9)</p>
STEERING COMMITTEE								
OVERVIEW	<p>Requirements Analyst is responsible for identifying a suitable approach for gathering the requirements from the customer.</p>	<p>As a significant activity Requirements Analyst to identify sources of requirements and techniques to be used for requirements gathering</p>	<p>The system requirements must begin with a complete understanding of the customer's needs.</p>	<p>It is imperative that the Requirement Analyst help the customer to develop a problem statement that is completely independent of solutions and specific technologies.</p>	<p>The Requirements Analyst must interact with the customer to write the S/W requirements. He must involve the customer in the process of defining, clarifying, and prioritizing the requirements.</p>	<p>This is the process of understanding and prioritizing requirements and determining a technical approach for testing requirements.</p>	<p>The purpose of Requirements Validation and Review is requirements that are consistent, complete, realistic, and unambiguous.</p>	<p>A critical element of the requirements development process is describing the tests, analysis or data that will be used to prove compliance of the final system with its requirements.</p>

PROCESS FLOW



SOFTWARE REQUIREMENTS PROCESS - OVERVIEW



**PLAN
SOFTWARE
REQUIREMENTS
PROCESS
(1)**

The Requirements Analyst in association with Team Members is responsible for identifying a suitable approach for gathering the requirements from the customer. The project manager should also identify suitable elicitation techniques that should be adopted for getting the needs. The supporting tools that would be used like checklists, sample software demos etc., should be identified before commencing the requirements study. The Requirements Analyst should identify, plan, estimate and schedule all the tasks associated with the Requirements engineering process and the work Products associated with the process. The Requirements Analyst should also plan for the resources required for this process.

PROCESS FLOW



DETAILED DESCRIPTION

IDENTIFY SOURCES OF REQUIREMENTS (2)

As a first activity for requirements specification process is to identify various stakeholder and sources of requirements:

Sources of Requirements could be:

- User Needs Document
- Interview with Users
- Interviews with Management
- Voluntary Standards
- Product classification
- Market and competitive analysis
- Statutory, regulatory requirements and compliance strategies
- Complaints/Failures and other historical data
- Risk analysis
- Product documentation
- Human factors studies

As the project matures and requirements are derived, all activities or disciplines will receive requirements. To avoid requirements creep, criteria are established to designate appropriate channels, or official sources, from which to receive requirements. The receiving activities conduct analyses of the requirements with the requirements provider to ensure that a compatible, shared understanding is reached on the meaning of the requirements. The result of this analysis and dialog is an agreed-to set of requirements.

The first step in developing requirements is to identify the customer. The term *customer* includes anyone who has a right to impose requirements on the system. This includes end users, operators, nurses, clinicians, regulatory agencies, accountants, sponsors, etc. All facets of the customer must be kept in mind during software requirements process.

PROCESS FLOW



UNDERSTAND CUSTOMER NEEDS (3)

The system design must begin with a complete understanding of the customer's needs. The information necessary to begin a design usually comes from preliminary studies and specific customer requests. Frequently the customer is not aware of the details of what is needed. Requirements Analyst must enter the customer's environment, discover the details, and explain them. Flexible designs and rapid prototyping facilitate identification of details that might have been overlooked. Talking to the customer's customer and the supplier's supplier can also be useful. This activity is frequently referred to as mission analysis.

It is the Requirements Analyst's responsibility to ensure that all information concerning the customer's needs is collected. The Requirements Analyst must also ensure that the definitions and terms used have the same meaning for everyone involved. Several direct interviews with the customer are necessary to ensure that all of the customer's needs are stated and that they are clear and understandable. The customer might not understand the needs; he may be responding to someone else's requirements.

PROCESS FLOW



DEFINE AND STATE PROBLEMS (4)

What is the problem we are trying to solve? Answering this question is one of the Requirements Analyst's most important and often overlooked tasks. An elegant solution to the wrong problem is less than worthless.

Early in the process, the customer frequently fails to recognize the scope or magnitude of the problem that is to be solved. The problem should not be described in terms of a perceived solution. It is imperative that the Requirements Engineer help the customer develop a problem statement that is completely independent of solutions and specific technologies. Solutions and technologies are, of course, important; however, there is a proper place for them later in the Product development process. It is the Requirements Analyst's responsibility to work with the customer, asking the questions necessary to develop a complete "picture" of the problem and its scope.

PROCESS FLOW



**WRITE
SOFTWARE
REQUIREMENTS
(5)**

The Requirements Analyst must interact with the customer to write the system requirements. The Requirements Analyst must involve the customer in the process of defining, clarifying, and prioritizing the requirements. It is prudent to involve users, regulators, manufacturers, maintainers, and other key stakeholders in the process.

Next, Requirements Engineering process must discover the functions that the system must perform in order to satisfy its purpose. The system functions form the basis for dividing the system into subsystems. Although this makes it sound as if requirements are transformed into functions in a serial manner, that is not the case. It is actually a parallel and iterative process. First we look at software requirements, then at software functions. Then we re-examine the requirements and then re-examine the functions. Then we re-assess the requirements and again the functions, etc.

PROCESS FLOW



DETAILED DESCRIPTION

ANALYZE SOFTWARE REQUIREMENTS (6)

The Requirements Analyst must interact with the customer to write the software requirements. The software requirements that are gathered and documented in the Software Requirements Specification should be analyzed further to identify any implied requirements. All the derived requirements should also be documented. During the analysis of requirements, the relationships between the various requirements should also be identified. The mechanism of implementing the requirements namely, through software, hardware, processes, services and documentation should be identified

During the requirement analysis, the operational concept and scenarios are identified. Operational concept refers to the Methods of how the system will be developed, deployed, operated, maintained and disposed. It has the viewpoints of different stakeholders of the system namely, customer, architects, developer, tester, maintainer, operator, end users. Operational concept should integrate the system life cycle scenarios into a timeline behavior that satisfies the viewpoint of each stakeholder. The outcome of this analysis will also provide additional requirements which should be included in the Software Requirements Specification.

The risks on cost, schedule, functionality, design and performance should also be assessed and documented for project risk mitigation and tracking.

All communications and documentation (including hard-copies) related with the requirements gathering and analysis process should be maintained.

Requirements could be classified as Stated, Implied and Derived requirements. Stated requirements are those, which are explicitly communicated by the requirement providers during the requirement-gathering phase. Implied requirements are the needs about the system/product in the minds of the requirement provider but not explicitly stated. Such implied requirements should also be recognized and documented. Derived requirements are those requirements that are identified during the analysis of requirements.

PROCESS FLOW



REVIEW REQUIREMENTS WITH CUSTOMER (7)

The Requirements Analyst must continually consult with the customer to ensure that the requirements are correct and complete. The customer should be satisfied that if these requirements are met, then the system will do what it really needs to do. All parties must agree to a way of measuring system performance to ensure that the system does what the customer wants it to do. The Requirement Analyst and the customer should identify which requirements can be used as trade-off requirements.

Sometimes the customer is not available for consultation. In such unfortunate situations, a surrogate customer will have to be used.

At these reviews it is important to ask why each requirement is needed. This can help eliminate unneeded requirements. It can also help reveal the requirements behind the stated requirements. It may be easier to satisfy the requirements behind the requirements, than the stated requirements themselves.

PROCESS FLOW



VALIDATE SOFTWARE REQUIREMENTS (8)

Validating requirements means ensuring that the requirements are consistent and that a real-world solution can be built and proven to satisfy the requirements. Each requirement should be technically feasible, and fit within budget, schedule, and other constraints.

Requirements are often validated by reference to an existing system that meets most of the requirements. The requirements that are not satisfied by the existing system are validated by argument, modeling, or simulation.

PROCESS FLOW



DESCRIBE VERIFICATION PROCESS (9)

A critical element of the requirements development process is describing the tests, analysis or data that will be used to prove compliance of the final system with its requirements. Each test must explicitly link to a specific requirement; this will help expose untestable requirements. Describing the system tests informs the producers how the system will be tested, so that they know how they will be "graded." This process frequently uncovers overlooked requirements.

At this time it may be useful to examine the following definitions.

Validating Requirements: Ensuring that the set of requirements is consistent, that a real-world solution can be built that satisfies the requirements, and that it can be proven that such a system satisfies its requirements. If Systems Engineering discovers that the customer has requested a perpetual-motion machine, the project should be stopped.

Verifying a System: Building the system right; ensuring that the system complies with its requirements. Verifying a system determines the conformance of the system to its design requirements. It also guarantees the consistency of the product at the end of each phase, with itself and with the previous prototypes. In other words, it guarantees the honest and smooth transition from model to prototype to preproduction unit to production unit.

Verifying Requirements: Examination, analysis, test, or demonstration that proves whether a requirement has been satisfied. This process is iterative. The requirements should be verified with respect to the model, the prototype, the preproduction unit, and the production unit.

PROCESS FLOW



UPDATE TRACEABILITY MATRIX (10)

For the requirements that are gathered a traceability matrix should be prepared which consists of all categories of requirements (functional, interface, service etc.) specified by the customer. During analysis this traceability matrix should be updated to include any derived requirements, which are not explicitly stated by the customer. Through out the engineering life cycle, this traceability matrix should be updated to ensure the implementation of requirements across the development phases.

PROCESS FLOW



PLAN FORMAL REVIEW (10)

Project Manager in association with the Requirements Analyst schedules the Formal Review meeting of Software Requirements Specification with the Team Members with adequate notice. The meeting to be attended by Team Members. The draft Software Requirements Specification to be circulated in advance and the comments are elicited prior to the meeting to conduct the meeting effectively.

The meeting to be led by Project Manager / Requirement's Analyst by presenting:

- Methodologies adopted in eliciting the product requirements
- Planned effort and actual effort
- Planned schedule and actual schedule
- Summary of Defects found and addressed so far
- Rework effort
- Audit report on this process, if any.

PROCESS FLOW



CONDUCT FORMAL REVIEW (11)

Requirements Analyst in association with Project Manager conducts the final review of the Product Requirements with the Team members before it is base lined.

The meeting to be led by Project Manager / Requirement's Analyst by presenting:

- Methodologies adopted in eliciting the Software requirements
- Planned effort and actual effort
- Planned schedule and actual schedule
- Summary of Defects found and addressed so far
- Rework effort
- Audit report on this process, if any.

The review comments are to be documented and addressed in the Software Requirements Specification. All review comments are to be preserved in the project file.

Team Members approval to be obtained to progress further in the product development cycle.

A Team Members is to look for errors, mistaken assumptions, lack of clarity and deviation from standard practice. The composition of the group that conducts the review is important and should include all relevant stakeholders. The reviews should be held periodically to ensure the adequacy and completeness of requirements, by obtaining feedback about them from relevant stakeholders.

After all the defects are addressed in the SRS, the same is to be sent to Customer for review and approval.

PROCESS FLOW



**BASLINE
SRS
(12)**

Based on the approvals for the Software Requirements Specification, the same would be baselined.

Defect free Software Requirements Specification is to be versioned as 1.0 and baselined. Henceforth, this document is under configuration management planning and control process, therefore, it would go through formal change management process for any required changes.

PROCESS FLOW



PROCESS AIDS:

- Software Requirements Specification Template (PD-SE-TP-001)
- Software Requirements Specification Guidelines (PD-SE-GL-001)
- Product Requirements Traceability Matrix (PD-SE-TP-002)

ADDITIONAL PROCESS NOTES:

PROCESS COMPLIANCE INDICATORS:

EXISTENCE:

- Software requirements Specification
- Product Requirements Traceability
- Review Records
- Study notes
- Minutes of Meeting

EFFECTIVENESS:

- No Major Non-Conformances
- Minimal Changes to Product Requirements
- Rework to Requirements Definition effort ratio

SR NO	TASK	ROLES									
		Requirements analyst	Project Manager	Product Architect	Verification & Validation	Developer	Quality Coordinator	Configuration In-charge	Unit Head	Steering Committee	Customer
1	PLAN SOFTWARE REQUIRMENTS PROCESS	R	A								R
2	IDENTIFY SOURCES OF REQUIRMENTS	R	A								
3	UNDERSTAND CUSTOMER NEEDS	R	A								
4	DEFINE AND STATE PROBLEMS	R	A								
5	WRITE SOFTWARE REQUIRMENTS	R	A								
6	ANALYZE SOFTWARE REQUIRMENTS	R	A								
7	REVIEW REQUIREMENTS WITH CUSTOMER	R	R						A		R
8	VALIDATE SOFTWARE REQUIRMENTS	R			R				A		
9	DESCRIBE VERIFICATION PROCESS	R			R				A		
10	PLAN FORMAL REVIEW	R							A		
11	CONDUCT FORMAL REVIEW	C							A		R
12	BASLINE SRS		A					R			

R RESPONSIBLE

A ACCOUNTABLE

C CONSULTED

I INFORMED

END