State of California

Department of Technology

Project Requirements Development Instructions

Statewide Information Management Manual – Section 170B

August 2016

Table of Content

1. Introduction 4
1.1 Overview 4
1.2 Background 4
1.3 References 5
1.4 Requirements Evolution 6
1.5 Project Glossary 6
2. Requirements Development within the Project Approval Life Cycle 7
2.1 STEP 1: Project Business Objectives Verification 7
2.1.1 Step 1 Inputs/Outputs 7
2.2 STEP 2: Mid-Level Solution Requirements Development 8
2.2.1 Step 2 Inputs/Outputs 8
2.2.2 Mid-Level Solution Requirement Template 9
2.2.3 Mid-Level Solution Requirements Types 9
2.2.4 Mid-Level Solution Requirements Management 12
2.3 STEP 3: Detailed Solution Requirements Development 13
2.3.1 Step 3 Inputs/Outputs 13
2.3.2 Detailed Solution Requirements Template 13
2.3.3 Requirement Specifications 14
2.3.4 Categorizing Requirements 18
2.3.5 Classification Framework 19
2.3.6 Detailed Solution Requirements Management 20
3. Requirements Development Planning 21
3.1 Roles and Responsibilities 21
3.2 Identify Project Stakeholders 23
3.3 Eliciting Stakeholder Requirements 24
3.4 Define Stakeholder Requirements 26
3.4.1 Define Constraints 26
3.4.2 Define Operation and Support Activity Sequences 26
3.4.3 Identify Human-System Interactions 27
3.4.4 Specify Critical Qualities 27
3.5 Analyze Stakeholder Requirements 27
3.5.1 Analyze Elicited Requirements 29
3.5.2 Resolve Requirements Problems 29
3.5.3 Provide Feedback 30
3.5.4 Validate Requirements 30
3.5.5 Document Requirements 30
3.5.6 Maintain Requirements Data 31
4. Requirements Management 31
4.1 Relationship to other Project Management Plans 31
4.2 Roles and Responsibilities 32
4.3 The Requirements Management Process 34
4.4 Establish a Controlled Requirements Repository 34
4.4.1 Requirements Repository 34
4.5 Create Requirements Baseline 35
4.5.1 Load Requirements Baseline 35
4.5.2 Identify and Tag Requirement 36
4.5.3 Set Requirement Attributes 36
4.6 Perform Requirements Traceability 36
4.7 Managing Requirements Change 36
4.7.1 Requirements Change Impact Assessment 37
4.8 Updating Baseline Requirements 41

4.8.1 Modification Update 41
4.8.2 Addition Update 41
4.8.3 Deletion Update 41
4.9 Requirements Traceability 42
4.10 Repository Updates 46

Table of Figures

Figure 1: Project Approval Lifecycle (PAL) 6
Figure 2: Stage 2 Alternatives Analysis Mid-Level Solution Requirements Template 9
Figure 3: Stage 2 Mid-Level Solution Requirements (SIMM Section 19B.1) 10
Figure 4: Stage 3 Detailed Solution Requirements Template 14
Figure 5: Category Group Levels 19
Figure 6: View of Requirements Analysis 28
Figure 7: Modified Requirements Assessment 38
Figure 8: Added Requirement Assessment 39
Figure 9: Delete Requirement Assessment 40
Figure 10: V-Model Example 43
Figure 11: Fan-out and Fan-in Example 44
Figure 12: Horizontal Traceability Example 45

Table of Tables

Table 1: SMART Objectives Verification 7
Table 2: Mid-Level Solution Requirements 11
Table 3: Detailed Solution Requirements Examples 16
Table 4: Roles and Responsibilities of Requirements Development 21
Table 5: Example Stakeholder Groups 23
Table 6: Example Elicitation Approach 25
Table 7: Roles and Responsibilities for Requirements Management 32

1. Introduction

These instructions, based on proven industry practices, methods and experience on California information technology (IT) projects, are designed to assist Agencies/state entities during the development of project requirements. Though listed as Steps 1-6, projects may follow a different process and use specific steps, or go through one or more steps several times. In general, most projects will follow a similar process of requirement development, requirement management, and requirement refinement.

Where applicable, refer to the Statewide Information Management Manual (SIMM) 170A General Requirements Guidelines document for instructions during Project Approval Lifecycle (PAL). The assumption is made that the progression of requirements work products that a project creates will follow the PAL in some manner.

Document Format Information:

- Guidance information is provided in normal text.
- Samples are provided in *italicized* text.

1.1 Overview

Bidding communities design solutions to specified requirements recorded in IT solicitation packages. Solicitation requirements often have a substantial associated cost. Due to the associated cost, it is critical that strong, well-developed requirements provide a clear understanding of the project need and eliminate ambiguity, thereby reducing costs. As a result, this instruction was created to assist Agency/state entities in the development and management of project requirements.

This document is divided into three separate sections with step-by-step instructions for IT solicitation requirements development – from the establishment of project objectives, to their maturity into mid-level solution requirements, and ultimately, detailed solution requirements.

Initially during PAL Stage 1 Business Analysis, the project identifies business objectives based on business needs. These objectives are expanded to mid-level solution requirements during Stage 2 Alternatives Analysis and further expanded to detailed solution requirements in Stage 3 Solution Development. For the purposes of the PAL, the Stage 2 mid-level solution requirements provide high level solution requirements used to guide in the development of Stage 3 detailed solution requirements, which include concise specifications of the expected quality of service and functionality. The details are to be precise, specific, and biddable simple statements of need that, when combined, indicate the manner in which the mid-level requirement must be achieved. Both the mid-level and detailed solution requirements are classified into functional, non-functional, and project/transition type requirements; however, detailed solution requirements can be further categorized in order to effectively manage, maintain, and track requirement development.

1.2 Background

Few professionals are able to create strong and well-developed requirements. Strong requirements typically go through a long methodical process with extensive critical stakeholder reviews, edits for technical correctness, reviews for legal appropriateness, and prioritized based on business need before becoming well-developed requirements. These instructions are designed to supplement in-depth learning, formal classroom training, and study of requirements to enhance an Agency/state entity's ability to develop strong, well-developed requirements.

1.3 References

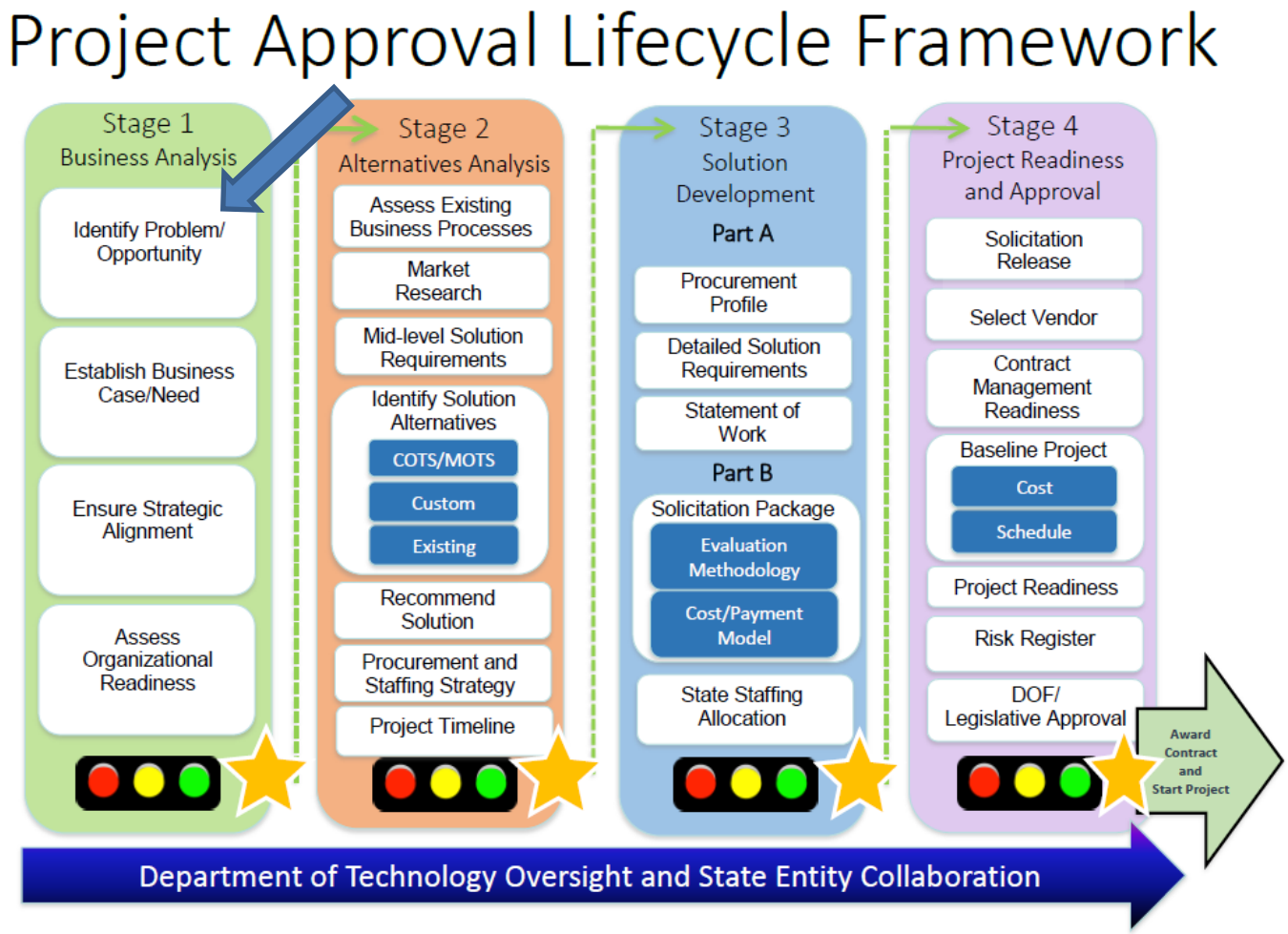
The following have been created to guide state IT project and procurement staff in one approach to use in developing strong, well-written requirements:

DOCUMENT REFERENCES
SIMM Section 170A General Requirements Guidelines
SIMM Section 170A Exhibit A Strong Requirement Samples SIMM Section 170B Exhibit B1-3 Expanded Requirement Samples SIMM Section 170B Exhibit C Glossary Sample SIMM Section 170B Exhibit D Requirements Development Workflow
SIMM Section 19 – Stage 1 Business Analysis http://www.cio.ca.gov/Government/IT_Policy/SIMM_19/SIMM19.html
SIMM Section 19A Stage 1 Section 1.10 Business Problem or Opportunity and Objectives Table http://www.cio.ca.gov/Government/IT_Policy/pdf/SIMM19/A.1-Preparation-Instructions.pdf
SIMM Section 19B Stage 2 Section 2.6 Mid-Level Solution Requirements http://www.cio.ca.gov/Government/IT_Policy/SIMM_19/SIMM19.html
SIMM Section 19B Stage 2 Mid-Level Solution Requirements Template http://www.cio.ca.gov/Government/IT_Policy/SIMM_19/SIMM19.html
SIMM Section 19C – Stage 3 Solution Development http://www.cio.ca.gov/Government/IT_Policy/SIMM_19/SIMM19.html

1.4 Requirements Evolution

In order to create strong solicitation requirements, strong project objectives precede them in PAL Stage 1 Business Analysis.

Figure 1: Project Approval Lifecycle (PAL)



These instructions are applicable when the project team members, stakeholders, and sponsors understand, agree, and approve the project's documented business objectives, problems and opportunities in a PAL Stage 1 Business Analysis. The remainder of this instruction will step through the requirement maturity development process.

1.5 Project Glossary

Beginning in Stage 1 (prior to requirements development), establish a project glossary to maintain the terms used during the PAL. Do not assume that it is not necessary to define commonly used terms (e.g., online form, real-time, user-friendly).

2. Requirements Development within the Project Approval Life Cycle

2.1 STEP 1: Project Business Objectives Verification

2.1.1 Step 1 Inputs/Outputs

- Step 1 Input(s): Stage 1 Business Objectives
- Step 1 Output(s): Verified Stage 1 Business Objectives, Established/Updated Project Glossary

Due to the possible extensive timeframe between the establishment of the business goals and objectives to the development of the business's mid-level and detailed solution requirements, it is recommended that project team members and necessary stakeholders re-familiarize themselves with the stated objectives. Evaluate that the business goals and objectives are SMART (i.e., "S" Specific, "M" Measurable, "A" Achievable, "R" Realistic, and "T" Time Bound) (http://www.cio.ca.gov/Government/IT_Policy/pdf/SIMM19/A.1-Preparation-Instructions.pdf).

Table 1: SMART Objectives Verification

SMART Objectives Verification		
SPECIFIC: Objectives must communicate what needs happen; the result. They should be action-oriented and detailed, well-defined and focused on what's most important.	What is this objective going to produce? Why is this outcome important to do? Who is going to do what? When does the project team want this result completed?	What exactly is the objective going to do, with or for whom? Does the objective clearly state what needs to happen? Is the objective outcome clear to anyone who reads it? Will this objective lead to the desired results? Is the objective well understood?
MEASUREABLE: The measurement source is identified and the project is able to track the results as progress is made towards achieving the objective. Measurement helps a project know when the objective has been achieved.	How will the project know or measure when an objective is achieved?	How will the project know that the change has occurred? Can these measurements be obtained?
ACHIEVABLE: Maintain balance between obtainable objectives while also focusing on organization growth or improved service. If the objective is too far into the future, motivation may dwindle.	Is the project's business objective(s) achievable?	Can it get done in the proposed timeframe? Does the project understand the limitations and constraints? Does the project have the right resources? Have other similar projects achieved success? Is this possible?

SMART Objectives Verification		
REALISTIC: The project has the resources to accomplish the work as expected. Resources are the necessary people, budget, skills, knowledge, equipment, and the project's priority. Achievable objectives are not always realistic.	Is the project's business objective(s) realistic?	The project has the resources available to achieve this objective? Is the project a high priority to the organization? Is it possible to achieve this objective?
TIME-BOUND: Objectives should have achievement deadlines. Deadlines help to set the necessary focus and priority, and prompts timely action.	Is the project's business objective(s) attached to a certain amount of time?	Are resources available to achieve this objective? Is the project a high priority to the organization? Is objective achievable in the timeframe specified?

2.2 STEP 2: Mid-Level Solution Requirements Development

2.2.1 Step 2 Inputs/Outputs

- Step 2 Input(s): Approved Stage 1 Business Objectives, Project Glossary, Requirement Development Plan (Section 3), Requirements Management Plan (Section 4), SIMM 170A General Requirements Guidelines
- Step 2 Output(s): Completed Mid-Level Solution Requirements Template, Requirements Traceability Matrix, Requirements Repository, Updated Project Glossary

During the Stage 2 mid-level solution requirements development, determine what business driven functionally needs to occur in order to fulfil the objective's goal (from Stage 1 Business Analysis). Focus to specific details of *WHAT* needs to occur. The Stage 3 detailed solution requirements will describe *HOW* functionality occurs.

This step initiates the requirements development process and introduces the project to requirements management. During this process:

- Identify requirement stakeholders
- Document as-is business processes
- Elicit stakeholders for information that will become requirements
- Analyze, critique, and vet stakeholder requirements
- Document requirements within the requirements repository
- Initiate requirements traceability

NOTE: Sections 3 and 4 provide an approach to perform stakeholder identification, elicitation, requirements analysis and approval processes to support these activities.

- Track requirements during the development process. The number of mid-level solution requirements can range from dozens to hundreds depending on the project size and scope.
- Establish a requirements repository.
- Enforce requirements traceability.

2.2.2 Mid-Level Solution Requirement Template

Mid-level solution requirements may be captured on the SIMM 19B.3 Mid-Level Solution Requirements Template. Agencies/state entities may use another tool to capture, track, and maintain mid-level solution requirements as long as the required fields noted in template are included in their requirements submission. Instructions to complete the template can be referenced in SIMM 19B Stage 2 Alternatives Analysis.

Figure 2: Stage 2 Alternatives Analysis Mid-Level Solution Requirements Template

	A	B	C	D	E	F	G	H	I	K
1	Project Number:			(XXXX-XXX)	Agency/state Entity:	(Agency/state entity name)				
2	Obj. Ref. (Obj. #	Requirement Category	Requirement Number	Main Business Capability/Functio	Sub Business Capability/Functio	Business Process	Requirement	Requirement Type	Priorit	Stakehold
3										
4										
5										
6										
7										
8										

Mid-level solution requirements provide a methodical, organized explanation of what a project is expected to deliver and should be heavily analyzed, however, it is not necessary to use the same rigor as for the development of detailed solicitation requirements.

A single standard development process does not exist for mid-level solution requirements. Below are some of the mid-level solution requirement development processes:

- Conduct Joint Application Design (JAD) sessions in order to elicit business needs from stakeholders
- Informally meet with users to conduct line-of-business interviews and/or create business use-case scenarios to explain current business flow(s)
- Observe line-of-business users to document business processes
- Procure off-the-shelf IT requirements, then “pick-n-pull” and tailor the requirements to best represent the business need.
- Hire contractors to conduct requirements development work.

One development approach is available for reference in Section 3. Additionally, Agencies/state entities may refer to *A Guide to the Business Analysis Body of Knowledge (BABOK® Guide)* for additional requirement and business need elicitation methodologies and processes.

2.2.3 Mid-Level Solution Requirements Types

Mid-level solution requirements describe the conditions, functionality and capabilities that a solution must have to satisfy the business needs identified in the objectives; they are sub-classified into functional, non-functional and project/transition requirements. (Refer to SIMM B.1 Section 2.6 Mid-Level Solution Requirements).

Figure 3: Stage 2 Mid-Level Solution Requirements (SIMM Section 19B.1)

Stage 1 – Business Analysis	Stage 2 – Alternatives Analysis	Stage 3 – Solution Development	Stage 4 – Project Readiness and Approval
• **Business Requirements -** Goals, objectives and outcomes identified. • **Stakeholder Needs Captured**	• **Process Flows** - The graphic representation of the business processes. • **Mid-Level Solution Requirements** - Characteristics of a solution scope and quality of service. They describe the conditions, functionality and capabilities that a solution must have to satisfy the business needs identified in the objectives; they are sub-classified into functional, non-functional and project/transition requirements. • **Functional Requirements** - Feature level information to validate the size of the system. Generally they are "what" the business has identified they want. • **Non-Functional Requirements -** Information to validate alternatives. Generally they are what/how non-business (i.e., technologists) want or need to satisfy what the business has identified they want. • **Project/Transition Requirements** - Information to validate the feasibility of cost and schedule. Generally they are temporary in nature and exist while the Project is progressing through project phases prior to project close and the solution becomes the new "as-is."	• **Detailed Solution Requirements -** They represent large groupings of concise specifications that when combined deliver an expected quality of service and functionality of a solution. They are sub-classified into functional, non-functional, and project/transition requirements. • **Detailed Functional Requirements** - Specifications to ensure the system meets stakeholder needs. • **Detailed Non-Functional Requirements** - Specifications to ensure the system operates as required; identifies qualities of the system and constraints on the system. • **Detailed Project/Transition Requirements** - Specifications to ensure the system is built on time and budget and meets quality levels. • **Detailed Mandatory/Optional Requirements** - Specifications on optional requirements (e.g., maintenance and operations after first year of operations) that will be implemented at the option of the state. • **Administrative Requirements** - Requirements that are defined by the Department of Technology, STPD and included under a separate section of a solicitation	• **Baseline Functional Requirements** - Information to test and subsequently maintain the desired functionality in the system. • **Baseline Non-Functional Requirements** - Information to test and subsequently maintain the quality and operational aspects of the system, within the defined constraints. • **Baseline Project/Transition Requirements –** Information to test and subsequently maintain the quality, budget and time constraints. • **Baseline Mandatory/Optional Requirements –** Information to subsequently maintain and validate the need to implement optional requirements (e.g., maintenance and operations after first year of operations).

Determination if a requirement is functional or non-functional can be challenging depending on the viewpoint from which the requirement was created. Requirements can exist as functional or non-functional and can be argued to reside in either column depending on who (business or non-business) wants to own it. Look at the following example:

"The solution shall support a four-digit year in all date fields and functions."

This requirement may be categorized as functional because it describes a requirement behavior of the solution. This requirement may also be categorized as non-functional because it describes a solution design constraint.

- Answer these questions, “Who wants the requirement delivered?” and “Does this requirement indicate how delivery will be achieved?”
- Apply the following logic to determine whether requirements are functional, non-functional, or project/transition requirements:

  - Functional requirements generally are “what” the business has identified they want.
  - Non-functional requirements generally are what/how non-business (i.e., technologists) want or need to satisfy what the business has identified they want.
  - Project/transition requirements generally are temporary in nature and exist while the project is progressing through project phases prior to project close and the solution becomes the new “as-is” (e.g., contractor requirements, solution transition requirements, training, and project specific requirements).

After applying the above logic, *“The solution shall support a four-digit year in all date fields and functions,”* is considered a functional requirement because the business required that the date be rendered in a specific way.

It is common to complete the PAL Stage 2 with more functional than non-functional and/or project/transition requirements since during this stage the business is providing functionality needs as opposed to how the functionality must be delivered.

Reference: Section 4 Requirements Development Guideline in the SIMM 170A General Requirements Guideline and SIMM 170A Exhibit A: Strong Requirement Samples should be referenced to provide further detail on writing strong mid-level solution requirements.

In the following example(s), the mid-level solution requirements express what is required to meet the business objective without specifying detailed requirement implementation details.

Table 2: Mid-Level Solution Requirements

Business Objective	Mid- Level Solution Requirement
1. *“Decrease manual processing of license applications by 70% within the first year of online processing availability to Californians. “*	***1.1 Functional: License applications and transaction processing will be available online.*** ***1.2 Non-functional: Solution shall implement a web application form*** ***1.3 Non-functional: Solution will process faxed or scanned license application submissions.*** ***1.4 Functional: Solution design will duplicate license application processing workflow. (Refer to Exhibit ABC: XYZ Business Process.)*** ***1.5 Functional: Transaction status notifications will be provided to users.*** ***1.6 Functional: Transactions will be stored and retained online, and searchable.*** ***1.7 Non-functional: Solution shall implement a web application form***

Business Objective	Mid- Level Solution Requirement
2. *"Process 25% more license applications annually that will cumulatively increase revenues by an estimated $260,000 within six (6) years of online availability to Californians."*	***2.1 Non-Functional: Solution will scale to accept and process concurrent license transactions*** ***2.2 Functional: Solution will generate ad-hoc financial and license reports.*** ***2.3 Functional: Solution will provide point-of-sale services to users.*** ***2.4 Functional: Solution will provide dynamic user profiles.*** ***2.5 Functional: Solution will provide self-service license purchases to users.*** ***2.6 Project/Transition: Solution will provide dual-processing until two (2) years after year end close.***

2.2.4 Mid-Level Solution Requirements Management

The project should have a set of categorized mid-level solution requirements supported by management, executives and sponsors that do not appear to be outside of scope or negatively impact cost or schedule. During the vetting process, the requirements are further scrutinized for aspects such as:

- *Does the requirement follow or conflict with state, federal or local laws?*
- *Does it define the "How" of what is being procured or is the focus on the "What" (as it should be)?*
- *Does the requirement provide the outcome as it was intended by the stakeholder?"*

Section 4 Requirements Management is designed to efficiently control analysis that results in the need to modify mid-level solution requirements. Strategies to conduct requirement reviews include:

- Hire consultants to review and report on the strength of the requirements
- Conduct market research to determine whether or not the requirements appear to ask for realistic, affordable solutions.
- Ask a partner project or Agency/state entity to review the requirement.

Regardless of how this is achieved, the goal is to present the requirements set to as many willing reviewers as possible, as efficiently as possible.

Reviewers should be kept informed of:

- How the review process functions
- How the review is progressing

Failure to engage in this type of communication may result in future unwillingness be a reviewer.

Changes to baseline mid-level solution requirements may occur. Be sure that changes to these requirements follow the project's established requirements change process, are documented/traced within the requirements repository, approved by appropriate stakeholders, and communicated to all project team personnel.

Capture and maintain traceability to project objectives on a current basis. Ensure that each requirement stakeholder (requirement owner) is identified; current, and documented and aware of the requirement change management process should modifications/requests for changes arise that impact their requirement.

Mid-level solutions requirements should be approved by project stakeholders, team members, relevant business partners and project sponsor representatives, at a minimum. Once vetting and decision making processes are completed, they will represent the baseline requirements. These baseline requirements will be used to expand into detailed solution requirements during the Stage 3; explained in Step 3 of this instruction.

2.3 STEP 3: Detailed Solution Requirements Development

2.3.1 Step 3 Inputs/Outputs

- Step 3 Input(s): Requirements Development Plan, Requirements Management Plan, Approved Mid-Level Solution Requirements
- Step 3 Output(s): Approved Detailed Solution Requirements Template, Updated Requirements Repository, Updated Glossary, Requirements Traceability Matrix

Generally, detailed solution requirements represent large groupings of concise specifications that when combined deliver an expected quality of service and functionality of a solution. This step is where the project will spend many hours developing detailed solution requirements. Once the mid-level solution requirements have been approved and properly documented, the detailed requirements can be developed.

Even though stakeholders and business partners were elicited for input, the resulting requirements may seem disjointed. There is no reason for concern, as that is normal. Until the requirements have been honed to be concise strong requirements, they will be that way.

Mid-level and detailed solution requirements exhibit a parent-child (one-to-many) relationship. It is common that mid-level (parent) requirements will expand to have many hundreds of detailed solution requirements (children) depending on the project scope.

Reference: Examples of this parent-child relationship can be illustrated in SIMM 170B Exhibit B1-3 Expanded Requirement Samples.

2.3.2 Detailed Solution Requirements Template

Detailed solution requirements may be captured on the SIMM 19C.5 Detailed Solution Requirements Template. Agencies/state entities may use another tool to capture detailed solution requirements as long the required fields are included in their requirements documentation submission. Instructions to complete the template can be referenced in SIMM 19C.1 Stage 3 Solution Development Preparation Instructions.

Figure 4: Stage 3 Detailed Solution Requirements Template

	A	B	C	D	E	F	G	H	I	J	K
1	Project Number:		0000-000	Agency/state Entity:	(Agency/state entity name)						
2	Stage 2 Mid-level Requirement Category	Stage 2 Mid-level Requirement Number	Category Name	Category ID / Requirement Number	Category ID/Category Process - Context Description/Requirement	Requirement Type	Priority	Stability	Requirement Owner	Requirement Score	Requirement Action (Use for changes only)
3											
4											
5											
6											
7											

2.3.3 Requirement Specifications

Detailed solution requirements will again be represented by the same functional, non-functional, project/transition type; as in the mid-level solution requirements. The expansion of the mid-level to the detailed requirement is matured in the “details.” The details are precise and specific; they are simple statements of need that combined with a group of other simple statements explain the manner in which a mid-level solution requirement must be achieved. These simple statements must each possess an attribute, criteria, a requirement, and a test. This is where the accuracy of the Requirements Traceability Matrix becomes critical. Refer to the SIMM 17 California Project Management Framework (CA-PMF) for the Requirements Traceability Matrix Template.

Detailed requirements represent performance specifications for a solution. A specification is more formally defined as, "A statement of required results with criteria for verifying compliance." Performance specification necessitates clear, definitive communication of required results, but should not unnecessarily limit the products, methods, or means of achieving those results. There are four essential elements of these specifications:

Attributes – The means by which performance characteristics are identified. Define the attributes that will be set at this step. They can be used to support other activities, such as testing for high priority or criticality requirements, or for reporting. *Examples: Functional, Performance, Usability/Quality-in-Use, Interface, Design Constraint, Process, Non-Functional, Quality.*

Requirements – “1. a condition or capability needed by a user to solve a problem or achieve an objective. 2. a condition or capability that must be met or possessed by a system, system component, product, or service to satisfy an agreement, standard, specification, or other formally imposed documents 3. a documented representation of a condition or capability as in (1) or (2) 4. a condition or capability that must be met or possessed by a system, product, service, result, or component to satisfy a contract, standard, specification, or other formally imposed document. (IEEE 24765:2010). These are statements of desired results, in qualitative or quantitative terms. Note that more than one requirement may be defined for a single attribute.

Criteria – Definitive statements of performance for a particular requirement, stated in quantitative or qualitative terms. Criteria must be either measurable or observable. Several criteria may be needed to completely and accurately define a requirement.

Tests – Checks for conformance with performance criteria and a measure of actual or predicted performance level. A test will be associated with each criterion and may be based on a recognized test method, calculation or engineering analysis, observation, or professional judgment. Test results may be evaluated by conducting the specified test, or simply by submitting certified results of previous testing.

In reference to the functional mid-level solution requirement from Section 2.2.2:

"The solution shall support a four-digit year in all date fields and functions."

A non-functional detailed solution requirement to support the delivery of the functional mid-level solution requirement, as the non-business (technologist) requires that the official system of record for the date be provided by the Network Time Server, could be:

"The solution shall use the time and date provided by the State Network Time Server."

Reference: Refer to Section 4 Requirements Development Guideline in the SIMM 170A General Requirements Guidelines for further detail on writing strong detailed solution requirements.

The following table provides detailed solution requirements expanded from the mid-level solution requirements developed in Step 2. They are simple concise "child" statements that when combined, indicate *how* the "parent" mid-level requirement (the "*what"*) will be satisfied. Each of the detailed solution requirements has notable requirement specifications (attributes, criteria, requirement, and test) characteristics.

Table 3: Detailed Solution Requirements Examples

OBJECTIVE	MID-LEVEL SOLUTION REQUIREMENT	DETAILED SOLUTION REQUIREMENT
1. "Decrease manual processing of license applications by 70% within the first year of online processing availability to Californians. "	*1.1 License applications and transaction processing will be available online.*	***1.1.1. Non-functional: Solution to be accessible via a Browser.*** ***1.1.2 Functional: Solution to receive online license request submissions, pay for licenses, provide receipt, and transaction notification.*** ***1.1.3 Functional: Design solutions to allow users to reset forgotten passwords, update profile information, and apply for multiple licenses.*** ***1.1.4 Functional: Solution to allow users to print license placards upon confirmation of the transaction.*** ***1.1.5 Functional: Solution to process submitted online transactions.*** ***1.1.6 Functional: Solution to provide online printable receipts, and email notifications for transactions, (See Exhibit X, Example Receipts, Transactions and Notifications for criteria and content).*** ***1.1.7 Functional: Solution to allow users to apply for multiple licenses during a single online license request submission request session.***
	1.2 Solution design will duplicate business processes to transfer work between state staff.	***1.2.1 Non-functional: Application workflow will be configurable to changes in business processes.*** ***1.2.2 Non-functional: Total transaction duration from user submission to state staff completion will be logged, retained, and searchable.*** ***1.2.3 Non-functional: User roles will be configurable.***
	1.3 Transaction status notifications will be provided to users.	***1.3.1 Non-functional: Design solution to notify users via email or text message of changes in transaction status.***

OBJECTIVE	MID-LEVEL SOLUTION REQUIREMENT	DETAILED SOLUTION REQUIREMENT
		1.3.2 Non-functional: Flagged transactions will provide status notification to users once weekly until the transaction is no longer flagged. ***1.3.3 Non-functional: Design the solution to generate automatic notification of unsuccessful processing to users.***
2. *"Process 25% more license applications annually that will cumulatively increase revenues by an estimated $260,000 within six (6) years of online availability to Californians."*	*2.1 Solution will scale to accept and process concurrent license transactions.*	***2.1.1 Non-functional: A minimum of 100 concurrent user transactions must be available during peak application season each year*** ***2.1.2 Non-functional: Design solution to implement transaction throttling.*** ***2.1.3 Non-functional: Design the solution to scale system resources to process 100 concurrent user transactions within five (5) minutes.***
	2.2 Solution will generate ad-hoc financial and license reports.	***2.2.1 Functional: Design solution to produce sales reports that span configurable durations (see Exhibit X for Sales Reporting).*** ***2.2.2 Non-functional: Reports will be stored and retrievable in the database.*** ***2.2.3 Functional: Design solution to permit state staff to modify information on printed reports.***
	2.3 Solution will provide point-of-sale services to users.	***2.3.1 Non-functional: Solution will adhere to Payment Card Industry 2.0 compliance standards.*** ***2.3.2 Functional: Design solution to process Visa, MasterCard and Discover payment card information.*** ***2.3.3 Functional: Design solution to transfer total payment amount of transactions to the Financial Balances System.*** ***2.3.4 Non-functional: Users will receive confirmation of payment during the transaction session as payment is confirmed from the payment card vendor.***

2.3.4 Categorizing Requirements

In order to more easily manage detailed solution requirements, classify them into a variety of project applicable categories. Categorization can vary depending on the project need and focus.

Functional, non-functional, and project/transition requirements are not logically categorized enough to be practically managed or applied. In order to better communicate, trace and search requirements with greater success, organizing requirement groups by category can be quite beneficial.

Classifying requirements will provide structure to the mid-level and detailed solution requirements. Structured logical groups of requirements are more easily managed and make verifying of requirements delivery easier and, once delivered, produces a valuable performance deliverable.

Figure 5 is a sample classification structure for IT system requirement elements and related work. Elements, as defined here, are major components common to most IT systems. Elements usually perform a given function, regardless of design specifications, implementation method, product or materials used. This instruction suggests a maximum three level definition. Starting from Level 1, the largest element grouping, it identifies Major Group Elements such as the Business, Information, Application, and Technology. Level 2 subdivides Level 1 elements into Group Elements. The Technology, for example, includes the Node and Communication Services. Level 3 breaks Group Elements further into Individual Elements. A Node, for example, includes physical hardware with resources with processing capability.

Figure 5: Category Group Levels

2015 DRAFT Proposed IT System Elements Classification			
	Level 1	Level 2	Level 3
Number	Major Group Elements	Group Elements	Individual Elements
01.00.00	**General**		
01.01.00		Administrative	
01.01.01			Project Management and Coordination
01.01.03			Progress Documentation
01.01.05			Submittal Procedures
01.03.00		Quality	
01.03.01			Regulatory Requirements
01.03.03			References
01.03.05			Quality Assurance
01.03.07			Quality Control
01.05.00		Product	
01.05.01			Software Licensing Requirements
01.05.03			Owner-Furnished Products
01.07.00		Execution and Closeout	
01.07.01			Preparation
01.07.03			Execution
01.07.05			Protecting Installed Systems
01.07.07			Closeout Procedures
01.07.09			Closeout Submittals
01.07.11			Demonstration and Training
03.00.00	**Business**		
03.01.00		Services	
03.03.00		Function	
03.05.00		Process	
05.00.00	**Information**		
05.01.00		Business Objects	
05.03.00		Security	
05.03.01			Identity and Policy Management
05.03.03			Authentication and Authorization Services
05.03.05			Cryptography
05.03.07			Auditing and Reporting

Performing economic analysis based on an elemental framework instead of on a product-based classification provides an opportunity to reduce time and costs for evaluating alternatives at early design stages. This encourages more economic analyses and more economically efficient choices among systems and system elements.

Ultimately, the existing stakeholder defined requirement “category” and “type” allow introduction of standardized categories to:

1. Minimize disruption to existing methods; and,
2. Maximize long-term value to the State, citizens and project quality improvements.

2.3.5 Classification Framework

The requirements classification framework, the selection of items to include, and the decisions of which parts of the classification to include are based on the following criteria:

- Selected items have a significant influence on project cost
- Selected items have high frequency of occurrence
- Selected items are distinctive
- Hierarchical structure allows aggregation and summarization at different levels
- Suitable for a range of applications, including cost control and preliminary project descriptions
- Accommodate unlisted items based on the judgment of IT professionals

- Professional judgment is used to place elements where IT professionals in current practice would normally look for such items in a classification structure.

2.3.6 Detailed Solution Requirements Management

Detailed solution requirements analysis and acceptance processes follow a similar approach as in Step 2 for mid-level solution requirements. However, some differences include:

- New stakeholder participation and increased elicitation meetings
- A much lengthier analysis and acceptance process
- Significantly more effort spent on requirements specifications and traceability
- Potential coordination of the requirements development results to other project deliverables (e.g., Work Breakdown Structure (WBS), Statement of Work (SOW), Project Plans).

The result of this work will be included in the Project's IFB/RFP, which will be included in the complete solicitation package released for vendor bid during the PAL Stage 4 Project Readiness and Approval.

<This concludes the step-by-step requirements development instructions.>

3. Requirements Development Planning

3.1 Roles and Responsibilities

Identify the individuals and groups and describe their specific roles and responsibilities for requirements development. This is not meant to be a job description of what the individual or group does on the project but rather a summary of the roles and responsibilities with respect to requirements Development.

The following is a listing of the most commonly used project individuals/groups that should be considered, though all may not be needed and others may need to be added based on the specific project. Projects must clearly identify the roles and responsibilities associated with each. Do not be generic and identify responsibilities such as "among the reviewers of …" as this does not convey any ownership responsibility. If there are groups, such as an Executive Steering Committee or reviewer groups, then identify the groups and cite where the members of the groups are defined (e.g., the Project Charter, Deliverables Management Plan).

Do not duplicate roles and responsibilities that are part of and defined within another Project Management Plan. For example, the approver of a Change Request should be defined in the Change Management Plan and not repeated here. As the role and responsibility changes in the project's lifecycle, it will be difficult to keep multiple plans consistent.

Table 4: Roles and Responsibilities of Requirements Development

TITLE	ROLE	RESPONSIBILITY
Project Executive Steering Committee	The Project Executive Steering Committee's role is to provide the final approval of the project requirements.	Their responsibility is to understand the approach used to elicit the requirements from all of the stakeholders as well as the approach used to develop and finalize the requirements. Then, when presented with the requirements, they will review a representative sample of the requirements to provide the final level of assurance that the requirements will communicate the stakeholders' needs to any potential vendor.
Executive Project Sponsor	The Executive Project Sponsor's role is to ensure that requirements' planning is executed in a timely and efficient manner and that the objectives are achieved.	The Executive Project Sponsor will provide the necessary support to the Project Manager to ensure that resources are available to support the execution of requirement planning and to provide the advocacy to all of the internal and external units and organizations that are required to participate in the requirements development effort.

TITLE	ROLE	RESPONSIBILITY
Project Business Sponsor	The Project Business Sponsor's role is to provide the overall business leadership to ensure the requirements meet the needs of users. If the requirements are deficient, they will identify issues and problems that need to be corrected. Once acceptable, the Executive Steering Committee will be responsible for approving the project requirements, which will then become baselined and fall under formal Requirements Management control.	The Project Business Sponsor is responsible for ensuring the requirements plan identifies an approach that addresses all of the organizational business users, their needs, and the organization's business support needs (e.g., Legal, IT Operations, Public Relations). The Business Sponsor also provides advocacy to the business and supports components of the organization to ensure that the documented requirements will meet the overall organization needs. Ultimately, the Sponsors (Business and Executive) are responsible for approving the requirements plan and ensuring that the activities and tasks are executed by the Project Manager.
Project Director	The Project Director's role is to provide support for the Project Manager by aiding in the removal of obstacles and resolving issues or problems during the development of the project requirements that are beyond the control of the Project Manager.	The Project Director is responsible for tracking the progress of the requirements development effort, identifying significant deviations, and assisting the Project Manager in resolving obstacles, issues, and problems in order to keep the effort on schedule without compromising the quality (completeness, correctness, readability, etc.) of the resulting requirements.
Project Manager	The Project Manager's role is to plan, schedule, resource, monitor, and track the progress of the requirements development effort.	The Project Manager is responsible for working with the Requirements Development Lead to review the requirements development approach, verify that all stakeholders are accounted for and the planned elicitation approach appears sound for each, and ensure that the tools necessary to develop, document, and maintain the requirements throughout the process are available. The Project Manager will schedule the activities, tasks, and resources necessary to execute the effort and monitor and track the execution. If obstacles, issues, or problems arise, the Project Manager will either resolve them or escalate them to the Project Director or Sponsors for resolution.

TITLE	ROLE	RESPONSIBILITY
Requirements Development Lead	The Requirements Development Lead's role is to lead or perform the activities and tasks defined within the requirements plan.	The Requirements Development Lead is responsible for executing all of the activities and tasks within the requirements plan and comply with the planned process. The Lead will work with the Project Manager to define and identify the resources needed to accomplish each activity and task, verify the tools are available, and assist in scheduling the execution of the effort.

3.2 Identify Project Stakeholders

It is important to identify, verify, and validate the individual stakeholders and stakeholder groups/classes that have a legitimate interest in the system throughout its lifecycle for requirement elicitation. When starting a project, key stakeholders will serve to verify that all requirements are identified. Therefore, the stakeholders list should be complete and ensure all necessary stakeholders are included, and exclude unnecessary ones. Ensure to identify stakeholders for the development effort and also operations and maintenance and other post-deployment users. Stakeholders will be elicited for requirements for their area of interest. Do not identify specific products (e.g., System Administration Manual for the IT Unit) but focus on the types of products and services. A product-based work breakdown structure (WBS) is an excellent starting point to identify stakeholders. From a WBS perspective, this would be a level-2 or level-3 view of the products and/or services.

The following table serves as an example of key stakeholders and stakeholder groups/classes for the project and their interest in the project execution and/or the artifacts produced.

Table 5: Example Stakeholder Groups

Stakeholder	Interest
XYZ Unit	*As a user of the system, they have an interest in ensuring that the system performs the XYZ functionality necessary for the unit to perform their business function and that reports can be generated.*
Chief Deputy Director	*As an Executive Sponsor, he/she has an interest in ensuring the project cost, schedule, quality, risks and issues are carefully monitored, managed, and provides direction to the project. The Executive Sponsor communicates the project status to other executives and intervenes and resolves escalated issues.*
IT Support Unit	*The IT Support Unit has an interest in ensuring the system can be supportable by the organization and that all artifacts delivered will be usable and accurately reflect the procedures necessary to be supported by the units' resources.*
Enterprise Architect (EA)	*The EA has an interest in ensuring the new system will meet the performance and other operational needs of the organization.*

Stakeholder	Interest
Training Office	*The Training Office has an interest in ensuring that the system documentation and training material is developed, with appropriate tools available as necessary to train all of the impacted stakeholders.*
Project Manager (PM)	*The PM has an interest in managing the execution of the project and ensuring that there are sufficient artifacts created and delivered that will allow the necessary tracking, managing, and reporting for the project.*
Department of Technology	*Department of Technology has an interest in the project to provide oversight and enforcement of the state's IT strategic plans, policies, and procedures, and assists in planning and implementing successful IT projects.*

3.3 Eliciting Stakeholder Requirements

For each stakeholder and stakeholder group/class identified in Section 2.1, an approach needs to be established to elicit their requirements. The elicitation process may be iterative and/or recursive and repeated as necessary to ensure all of the requirements have been captured. Note that not all techniques work well with all stakeholders; the project must understand the stakeholders and stakeholders classes/groups and identify an approach suitable for the specific stakeholder or class/group. As an example, a top-down business process model to detail user interactions, data, and rules might work well with specific business users but this same approach would not work well for an Executive Sponsor or the IT organization that may be maintaining the solution.

For the specific project, clearly identify the technique(s) that will be used for each stakeholder or stakeholder group/class and plan for iteration and refinement of the technique as more is known about the needs. While elaborate models are discussed in literature and standards, understand your stakeholders and determine what will work best for them.

As each identified stakeholder and stakeholder group/class is elicited, the requirements and supporting artifacts shall be documented in various forms and formats. In the early iterations, it is anticipated that the documentation created will be primarily graphical in nature, such as flow charts, pictures from white boards, etc. These types of documents will be captured in the form that they were developed -- Visio, PowerPoint, photo (jpeg), etc. However, as the requirements are refined, they should be documented in a MS Word format or other standardized format to allow for further analysis. When documenting requirements at all levels, the requirement context will always be kept with the set of requirements being collected.

The project needs to define the format and structure of the final elicited requirements documents, which are the documents that will be provided for the requirements analysis activity. This is important for a number of reasons: 1) it establishes a common and consistent expectation of the types of data that must be collected for each context area; 2) it supports the analysis activity and tasks by providing a consistent input into the categorization and filtering of requirements; 3) it reduces the amount of re-writing that the analysis team will need to do to ensure context and requirements sets are defined; and, 4) it simplifies the identification of missing or incomplete requirement sets during both elicitation and analysis. While the Use Case approach to documenting the final elicited requirements is identified here, any artifact that keeps the requirements context with the set and types of requirements elicited is sufficient.

All requirement artifacts collected shall be stored in the requirements repository. The artifacts captured will consist of all tools, techniques, diagrams, flow charts, photos, documents, files, etc., that were used in the elicitation process. While some of these artifacts will be used as reference later during the analysis activity, others will be formalized to define or support the requirements and will be incorporated into the final requirements document (e.g., the baseline requirements document).

During the elicitation effort, the collected requirements will not be reviewed or analyzed in any way other than to ensure that the captured requirements are clear, complete, and understandable. Review and analysis occurs later, as described in Section 3.8.

The following is an example of which elicitation approach was identified for each stakeholder:

Table 6: Example Elicitation Approach

Stakeholder	Elicitation Approach and Iteration/Recursion
XYZ Unit	*The initial approach that will be used is a top-down business decomposition method. Business process flows will be developed for the "as-is" process. User required changes to this "as-is" process will also be documented. This will be the first level of iteration.* *Then, "as-is" Use Cases will be developed that will identify and document the unit's interactions with the system.*
Chief Deputy Director	*Requirements will be elicited by a meeting with the Chief Deputy Director to identify/describe the specific data he or she expects to be generated and reported on by the Project to the Chief Deputy Director, the frequency of reporting, and the criteria for exception reporting. The frequency of reporting ….*
IT Support Unit	*Requirements will be initially elicited through the requirements teams' review of the IT Units' organizational procedures and manuals where initial requirements shall be captured.* *Then, additional meetings will be held that will review the existing Units' processes and procedures, which shall then be validated.* *Through the use of recursive requirements brainstorming sessions where each step of the Units' process is reviewed, additional requirements will be elicited.*

3.4 Define Stakeholder Requirements

The stakeholder requirements may include tasks and/or statements that define constraints on the requirements and aid in identifying requirements that are commonly missed during the requirements elicitation effort.

3.4.1 Define Constraints

The following identifies constraints on the system/solution that must be captured in the form of system requirements. While these constraints in themselves are typically not suitable as requirements, they shall be used to define or derive requirements for the system. These constraints may come from a variety of sources but are commonly identified within the PAL Stage 2 or other project approval documents or from Agency/state entity internal decisions. The constraints identified here will be used as the "filters" when performing the requirements analysis effort and therefore the more complete they are, the more useful they will be when reviewing the requirements elicited. Typical and common examples include:

- The system/solution must be hosted within the OTech CalCloud environment.
- Existing legacy system data exchange interfaces with external trading partners shall not be modified or changed in any way.
- The system/solution must be consistent with and use the operating systems, databases, and programming languages currently supported by the Agency/state entity.

3.4.2 Define Operation and Support Activity Sequences

It is important to identify operational and support scenarios that are out of the normal or routine business processing flows. These scenarios aid in identifying additional system requirements necessary to meet the unusual or non-routine needs of project stakeholders. The intent of adding these scenarios is to broaden the view of the requirements that need to be defined for the project.

Define a representative set of operation and support scenarios, which are not business process flows of other business-related scenarios. The objective of these scenarios is to aid in potentially identifying requirements that may not have been identified by any stakeholder or stakeholder group. Examples would include scenarios related to availability of the system, especially around key periods such as the end-of-the-month reporting, legal audits that may be performed, Public Record Act (PRA) requests, public-facing attempts to hack the system, user interactions with the system for training purposes, etc. These are primarily non-functional types of scenarios that are looking at the system as a "black box."

For example:

> Scenario X: The Department's Legal Office (Source) has received a Public Records Act request (Stimulus) for business license data. The system is in normal operations mode (Environment Conditions) and the Legal Office accesses the Reporting subsystem (System Artifact Involved) and generates a report (Response) that provides the requested data. The data included within the report only includes data that is publicly releasable and complies with all legal statutes (Response Measure).

From this sample, while the Legal Office may have been identified as a stakeholder, this scenario is used to help ensure that all of the necessary requirements to cover this situation are identified.

While it may appear difficult to develop such scenarios, the best approach is to draw the system as a box on a whiteboard and ask what could happen to the system, from the operation and support perspectives and then from different stakeholders' perspectives. Start with brief scenarios (e.g., external hackers attempt to force a denial-of services, regional power outage, major technology supplier goes out of business, end-of-month and the system goes down) and further elaborate those scenarios that appear reasonable or that could potentially become real for system development consideration. Never delete any identified scenario; simply archive the ones that are not expanded. Document the resulting scenarios within this section.

The objective is to try to find gaps or requirements missed during the requirements elicitation process. By documenting the scenarios in this section, additional attention will be focused on ensuring that the necessary requirements are defined or, during analysis, gaps identified. This effort will never be 100% complete and perfect; however, the earlier gaps are discovered, the easier they can be closed.

3.4.3 Identify Human-System Interactions

The following identifies the high-level requirements for the interfaces between human users and the system to be developed. Information systems have a significant human-to-system interface (visual and well as manual) and the requirements for these interactions must be elicited and documented.

Typically, this is not a long list of items but for some systems there may be unique human-system interactions that must be specified. An example of a typical requirement for Web-accessible systems is a requirement for Section 508 (29 U.S.C. '794 d) and/or W3C Web Content Accessibility Guidelines 1.0 compliance. Other examples may include different levels of training and/or resource skills available to use, maintain, and operate the system. Identify either the human-system interaction requirements or the approach that will be taken to identify these usability requirements. Typically, these types of requirements are missed when eliciting requirements from the stakeholders.

3.4.4 Specify Critical Qualities

In this section, identify if there are any areas that are deemed "critical." For example, a system that is externally facing that also holds personally identifiable information (PII) data may have security as a critical requirement. These requirements should already have been identified, but there may be a small subset that would be identified as critical, which are mainly security related for state IT systems. If there are, they should be identified and/or reference separately to avoid omission.

3.5 Analyze Stakeholder Requirements

Once the requirements have been defined and elicited, all of the requirements will be analyzed and maintained. Requirements analysis involves a series of steps and one approach that can be used is illustrated in Figure 6 below.

Filtering requirements into categories can also be used to group requirements with similar attributes together. Additionally, categorizing traces the progression of how a particular requirement at the business level is being solved through the mid-level and detailed solution requirements. One example of project categorization of mid-level solution requirements is provided in Figure 6:

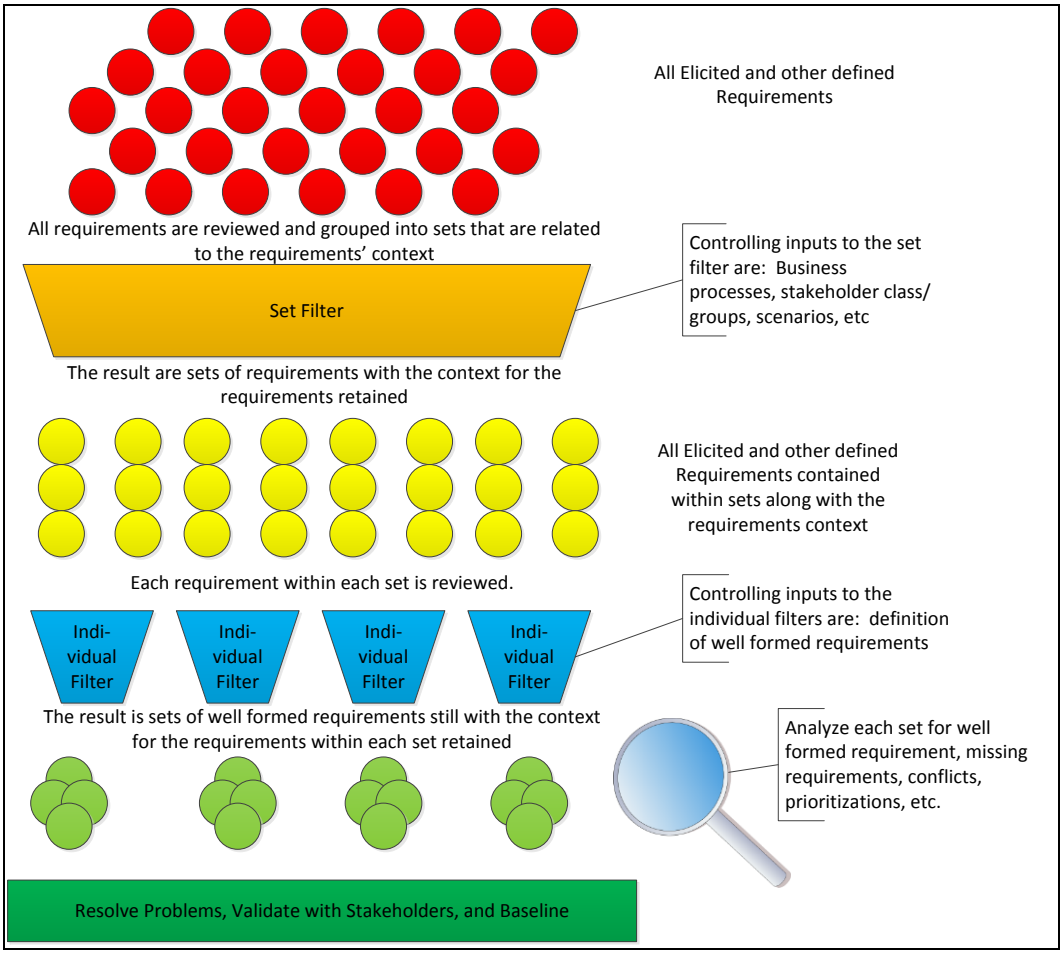
Figure 6 illustrates that all of the requirements elicited and defined (at the top of Figure 6) will be reviewed and placed into sets, which are groups of common requirements related to a specific business or other context (if the elicitation process kept the requirements context with the requirements collected). This step is typically a relatively simple effort, especially if Use Cases or a similar requirements documentation approach was used.

To perform this filtering and the creation of requirement sets, the controlling inputs are the requirements context artifacts collected during elicitations, such as the business process flows, stakeholder classes/groups, defined scenarios, operations and support approaches, etc.

The objective of this initial filtering and setting approach is to allow a more exhaustive and complete analysis to identify missing or incomplete requirements later in the analysis process. It is virtually impossible to analyze and find missing, incomplete, conflicting, or other requirement problems in a large number of individual requirements without partitioning them into subsets around some common context prior to the analysis. The result is a collection of requirements grouped and has its own unique context (e.g., a business context for cash management, patient in-processing).

Besides being necessary to perform analysis, it is also critical that the requirements context be captured with the requirements sets in order to facilitate reviews with the stakeholders, and to provide improved communications with potential vendors when defined within a solicitation.

Figure 6: View of Requirements Analysis



The initial requirement groups will then pass through another filter that reviews each individual requirement to verify that the requirement text is well formed, meaning the requirement does not use ambiguous words, is complete, readable and correct (within the requirements set context), and generally adheres to the controlling inputs, which are characteristics of individual requirements. The results of this final filtering stage are groups of well-formed individual requirements, with the requirements context retained. Each group is then analyzed to verify that they as a whole are well formed, reviewed to identify missing and conflicts between requirements, achieve level-setting[1] of requirements within the group, and identify prioritizations and other attributes.

After this analysis, problems in the form of requirements gaps in completeness, conflicts or inconsistency, affordability (not just from a cost perspective, but schedule, risk, M&O, etc.), scope, etc. will be identified. These problems will be resolved with the stakeholder and stakeholder class/group that provided the requirement for the requirement context identified with the requirements group. If conflicts exist between stakeholders or stakeholder classes/groups, they will be resolved in accordance with the method defined in Section 3.5.2. Once resolved, each requirement set, including the requirements group context, will be reviewed per Section 3.5.3 and validated per Section 3.5.4 with the stakeholder and stakeholder class/group that has a legitimate interest in the requirements.

Once validated, the requirements group with all reference material will be baselined and provided to the requirements management process to control per Sections 3.5.5 and 3.5.6.

3.5.1 Analyze Elicited Requirements

The project needs to define how they are actually going to perform each of the steps identified above based on the artifacts that they will collect from the requirements elicitation effort and other requirements either captured within this document or through other defined means. Minimally, define 1) how requirements will be grouped with the same requirements context, 2) how individual requirements will be analyzed to ensure they are well-formed, and 3) how each set will be analyzed to ensure the sets are well-formed.

3.5.2 Resolve Requirements Problems

The project needs to identify the approach they will use to resolve problems identified in the analysis step. It is extremely important to always keep the requirements context in mind when resolving problems as well as the entire set of requirements; do not attempt to identify missing requirements with the stakeholders without presenting the requirements context and the requirements that already have been identified for the group as this will result in requirements being identified that will likely be inconsistent with the other requirements in the group. The resolution of problems should not result in the need to re-work the analysis effort.

It is important to understand that requirements negotiation can be used as a tool. Negotiation might be needed among stakeholders requiring mutually incompatible features, or when there are conflicts between desired performance requirements, constraints, available budget, and delivery schedule.

This section must also identify how conflicts between stakeholders and stakeholder classes/groups will be resolved.

Regardless of how the problems are resolved, the project must identify that records will be kept on how problems and issues were resolved and by whom or what was agreed.

[1] Level-setting of requirements simply means that the requirements within a set are specified at roughly the same level of detail, except where necessary to be otherwise.

3.5.3 Provide Feedback

Sometimes agreements cannot be reached on conflicting requirements, unrealistic requirements will not be changed, or continually requested requirements are beyond the project scope. The project needs to define how these will be handled. If the project has an issue management or an escalation process in place, these might be cited. However, if neither of these exists at this point in the project, an approach must be defined.

3.5.4 Validate Requirements

Describe how the project plans to validate the requirements. As stated previously, it is important to keep a requirement group and their context together when performing any requirements validation effort; validation should not be simply a review of a long list of individual requirements as context adds significant value for understanding what is specifically being required.

The most common way to validate requirements is by performing a series of requirements reviews that are focused on the requirements context and the individual requirements within that context. For example, a requirements review session context may be for patient in-processing and all of the requirements related to this context will be reviewed and validated. As always, issues or action items are likely to be generated from these meetings so this section must define how these will be captured, tracked, and resolved.

Finally, the sign-off or formal acceptance or approval of the requirements and issues or action items must be described within this section; this sign-off or formal acceptance or approval must be made by the stakeholders that have an interest in the requirement context and the individual requirements documented.

3.5.5 Document Requirements

This section must identify the documentation that will be kept and maintained throughout the requirements analysis effort, which will also end up being passed to the requirements management process for tracking and controlling. While a project may consider keeping little documentation other than the final requirements, it is important that the document necessary to support the attribute assessments, at a minimum, be retained. For example, documentation related to the source of the requirement(s), a requirements priority, issues related to a specific requirement (such as the stakeholders' firmness or uncertainty) may relate to a volatility attribute, etc. The project must consider the attributes that will be kept of the requirements and minimally identify the documentation necessary to support these attributes and identify them within this section.

Also include here how the final document will capture context in addition to just the requirements. For example, specify if Use Cases or other types of documents will be used that capture the requirements context, requirements data, business rules, etc. Again, these are the artifacts that will form the initial baseline set of requirements that will end up being incorporated into a solicitation. Therefore, they must communicate the stakeholders' needs to potential vendors who may not have any understanding of the stakeholder processes or needs, which adds to the importance of maintaining the requirements context with the requirements.

3.5.6 Maintain Requirements Data

The project must identify how the requirements, requirements group, group context, and all additional supporting information will be delivered to the requirements management process. While the resources performing both requirements development and Requirements Management may be the same, from a process perspective, they are different processes. The main point to this description is to specify how this requirement's "snapshot in time" will be captured in order for the transferred artifacts to be complete. In practice, requirements are typically approved incrementally and transferred as each group is approved. Therefore, if this approach is going to be used, it must be described here. However, a word of caution is that once passed to the Requirements Management process, the requirements are baselined and cannot be changed without change control procedures being followed.

If an approved requirements group is passed to Requirements Management and a later review of a different group identifies a problem with the already approved, baselined group, then a change request should be created.

4. Requirements Management

The purpose of the Requirements Management Plan is to describe the roles and responsibilities for Requirements Management and the activities and tasks that will be performed as part of this Requirements Management effort. The activities and tasks include:

- Creation of a controlled and managed requirements repository
- Capturing approved project requirements within the repository
- Assessing and controlling changes to the baseline requirement
- Creation of metrics and reports

The Requirements Management Plan addresses the management, assessment, and control of changes to project requirement baselines (specifically, the requirements contained within each baseline, over the entire development lifecycle, consistent with ISO/IEC/IEEE 16326-2009, to increase the probability of a project's success).

4.1 Relationship to other Project Management Plans

The project should describe the relationship of the Requirements Management Plan to the other project plans that have been or will be created for the project. This can be done multiple ways, by identifying the other plan and how they are related or graphically display the relationship. For example:

Change Management Plan:

- *The Change Management Plan identifies the process and controls that will be enforced on the project to manage changes to the requirements. As a change is being considered, an assessment of the impacts of the change will be made that will include the assessment described within this Plan. After a change is approved, the change will be incorporated into the requirements baseline as described within this Plan.*

The objective is to provide a clear understanding of how this Requirements Management Plan integrates with the other Project Management Plans.

4.2 Roles and Responsibilities

The project should describe the specific roles and responsibilities as they have been tailored for requirements management activities. These are not meant to be general job descriptions for the role but rather a summary of the responsibilities for each role with respect to requirements management and the activities and tasks described within this Plan.

Table 7: Roles and Responsibilities for Requirements Management

TITLE	ROLE	RESPONSIBILITY
Project Executive Steering Committee	The Executive Steering Committee's role is to ensure that the baseline requirements are being managed in accordance with this Plan and that the traceability is complete and all issues with traceability are being resolved.	The Executive Steering Committee is responsible for reviewing the Requirements Management reports provided to the Committee in accordance with this Plan and for understanding the effects of all open issues with traceability and the consequences of the identified effects. They are also responsible for ensure that the Project Manager has a sound plan for resolving all open issues or mitigating the impacts and for resolving issues that have been escalated to the Committee regarding requirements.
Project Executive Sponsor	The Project Executive Sponsor's role is to ensure this Plan is executed in a timely and efficient manner and that the objectives are reached.	The Executive Sponsor will provide the necessary support to the Project Manager to ensure that state and vendor resources are available to support the execution of this Plan and to provide the necessary support to ensure the vendor is providing the necessary artifacts and Requirements Management efforts to support this Plan, in accordance with their contract.
Project Business Sponsor	The Business Sponsor's role is to provide the overall business leadership to ensure the requirements baseline is maintained, that requests for requirements changes have followed the approved Change Management process, and that approved changes to the baseline have been timely incorporated into the requirements baseline.	The Business Sponsor is responsible for reviewing the Requirements Management reports to ensure that the requirements baseline is complete, that all approved changes have been incorporated, and the impacts caused by these changes are identified within the repository.

TITLE	ROLE	RESPONSIBILITY
Project Director	The Project Director's role is to provide support for the Project Manager to aid in removing obstacles and resolving issues/problems that are beyond the Project Manager's span of control.	The Project Director is responsible for tracking the progress of the Requirements Management effort by reviewing the Requirements Management reports generated and the issues that are indicated within the reports. The Director is responsible for ensuring that the Project Manager has a sound, timely, and reasonable approach for resolving the identified issues.
Project Manager	The State Project Manager's role is to ensure the overall Requirements Management effort is being executed in accordance with this Plan.	The state Project Manager is responsible for ensuring that the entire project team, state and vendor, are following this Plan and for ensuring all of the other project processes that interact or provide input to the requirements management effort are being adhered to. The Project Manager is also responsible for ensuring that there are sufficient resources to execute this Plan and that the requirements management activities are being performed in a timely manner.
Vendor Project Manager	The Vendor Project Manager's role is to ensure the vendor team is complying with the Requirements Management process and procedures within this Plan and in accordance with the requirements in the vendor's contract.	The Vendor Project Manager is responsible for performing reviews of the Requirements Management work being performed by the vendor team and to verify that the work complies with the requirements management process described in this Plan and the requirements in the vendor's contract. The Vendor Project Manager is responsible for identifying issues to the state Project Manager timely to minimize the amount of rework necessary for the state and the vendor teams.
State Requirements Manager	The State Requirements Manager's role is to provide the leadership for the overall Requirements Management process and assume the sole ownership of the entire requirements repository.	The State Requirements Manager is responsible for the overall requirements management effort and the requirements repository containing the requirements baseline. This person is responsible for ensuring that the requirements managed by this Plan are organized, managed, and controlled and that any and all issues are identified and resolved in a timely manner in order to minimize rework.

TITLE	ROLE	RESPONSIBILITY
Vendor Requirements Manager	The Vendor Requirements Manager's role is to perform the Requirements Management activities and tasks assigned in accordance with this Plan and the vendor contract.	The Vendor Requirements Manager is responsible for ensuring that all work performed in the requirements repository by the vendor team complies with the process and procedures defined within this Plan. They are also responsible for ensuring that issues that impact the requirements management effort that are part of the vendor's responsibility are identified early and resolved in order to keep the repository current and/or minimize rework.

4.3 The Requirements Management Process

The requirements management process defined herein is consistent with the requirements defined in ISO/IEC/IEEE 16326-2009. Included are establishing the mechanisms for changes to requirements, assessing the impact of requirements changes. The controlled requirements repository should have already been created where changes can be measured and reported. Within the requirements repository, the requirements will be baselined and requirements traceability, that relates mid-level solution requirements to detailed solution requirements and artifacts, will be performed.

4.4 Establish a Controlled Requirements Repository

A controlled requirements repository is critical for requirements management. Project requirements will change! In order to ensure that these changes are managed and that all areas impacted by the changing requirements know about the change, plan their work with knowledge of the changed requirements, re-plan their work that may already be in progress due to the impacts of the change, or plan for re-work due to the change occurring after the work was already done. The only method to fully understand the impact of change is to maintain a tightly controlled requirements repository that also supports traceability, which will be discussed later. However, without a controlled requirements repository, no traceability can be performed.

4.4.1 Requirements Repository

The requirements repository is a database, spreadsheet, file system, or other data storage system that is created through the use of a tool where the project requirements will be maintained. The repository establishes and defines the projects' requirements baseline at any time throughout the project lifecycle. The repository incorporates the project approval requirements, solicitation/contractual requirements (contract baseline), and extends them based on approved changes and further refinement and elaboration as the project progresses through its development lifecycle. Every phase of the project lifecycle builds upon and/or elaborates on the requirements baselined and captured in the repository from the previous phase and the lifecycle phases correspond to the levels defined within the requirements repository.

Therefore, the requirements repository captures the results from the current level/phase to support the next level/phase and defines what must be addressed in the next level/phase. Further, the repository identifies what must be tested for all test levels and is the source for developing test cases/scripts.

The repository must be capable of incorporating all of the requirements from all levels/phases of the project. The repository must also be auditable in order to be able to identify if any requirement has changed and be capable of identifying the impacts of a change to lower level requirements and to test cases/scripts. Since the requirements repository plays such a vital role in the development of a system, it must be constantly managed, maintained, and controlled.

Identify the tool that will be used to hold the requirements repository. Identify how the tool will be organized, physically, how artifacts will be captured within the tool, what requirement attributes will be used at each Level, how traceability will be performed within the tool, both horizontally and vertically, and the specifics on how access and security will be controlled and maintained. If standard organizational policies will not be placed on the tool, specifically backup/recovery, security, etc., and describe how these will be accomplished. Also describe the approach to tagging (the unique numbering of requirements at each Level of the requirements).

4.5 Create Requirements Baseline

Upon the approval of a requirements artifact, the artifact will be loaded into the requirements repository, individual requirements identified and tagged, and the requirements' attributes set in accordance with the following procedures.

Identify the procedures for the Requirements Manager/Analyst to:

1) Load the requirement artifact into the tool
2) identify and uniquely tag the requirements, preserving the internal relationships between requirements
3) Set the requirement attributes.

Since the procedures to do this are dependent on the tool that will be used, detailed instructions must be provided by the project. However, the purpose of loading the requirement artifact is so that the artifact that forms the baseline of the requirements is captured and controlled within the repository. The reason for identifying and tagging requirements is to have a consistent method for determining what is and what is not a requirement within the requirements artifact; tagging allows a unique numbering or identification to be made for each requirement, which will be heavily used for traceability.

The purpose of determining and assigning attributes is to characterize each requirement. Common and standard attributes are items such as risk, priority, criticality, complexity, cost, status, business area, etc., and are more typically assessed in levels, such as High, Medium, and Low. Attributes support the reporting, generating of metrics, and change assessments so they are important to identify and assess when reviewing the individual requirements.

Again, these should be repeatable procedures, not guidance or a narrative. A quality assurance/quality control organization should be able to verify if these procedures are being followed or not. At a minimum, there are three (3) procedures required for this section.

4.5.1 Load Requirements Baseline

Determine the procedure that describes what is required to load the requirements artifacts into the repository. Be specific but do not describe different procedures for every requirements artifact. It is more common to describe procedures based on the document type, (e.g., MS Word, MS Excel, and MS Access). However, only describe the procedures necessary and supported by the Deliverable Expectation Document (DED) for the deliverables; if the project only allows deliverables to be submitted in an MS Word format, then only describe the procedures for artifacts that are in MS Word.

4.5.2 Identify and Tag Requirement

Identify how requirements will be identified within each Level of the requirements. This may consist of looking for key words, such as shall or must, or if the requirements are not documented consistently this may need to be more general direction but the direction must be sufficient to obtain repeatable results. Then, identify the procedures for tagging the requirements at each Level.

4.5.3 Set Requirement Attributes

Identify the procedures for setting the requirement attributes at each Level of the requirements. Clearly identify mandatory attributes that must be set (e.g., Status, Criticality, Owner), as well as optional attributes. A word of caution though on optional attributes is that these cannot be reliably queried for reporting purposes as they may not be set consistently. For example, if the Owner attribute identifies who is responsible for providing the requirement (State or vendor) and this attribute is optional, it would not be possible to run a reliable query to separate the contractual requirements from the non-contractual requirements.

4.6 Perform Requirements Traceability

A Requirement Traceability Matrix (RTM) shall be used to track and link each individual requirement from its origin through the development lifecycle. This matrix is used to track the requirements and to check that the current project requirements are met. The RTM captures all requirements proposed by the project stakeholders and their traceability in a single deliverable at the conclusion of the project life-cycle. In other words, it is a document that maps and traces user requirement with test cases. The main purpose of Requirement Traceability Matrix is to see that all test cases are covered so that no functionality is missed while testing. Refer to the SIMM Section 17 CA-PMF Requirements Traceability Matrix Template.

Vertical traceability will be performed from the PAL through to the Code Level. Horizontal traceability shall be performed from the Code Level to the Unit Test Level, Design Level to the Integration Test Level, Requirement Analysis Level to the System Test Level, and from the Project Level to the User Acceptance Test Level and Deliverables Level.

Identify the specific procedures for performing traceability. Since the steps for performing traceability are dependent on the tool being used, at least at the procedure level, this section must be written for the specific tool being used. It is critical that both downward/forward and upward/backward traceability be documented. When using a Requirements Management tool such as Requisite Pro, both downwards and upwards traceability will be performed automatically by creating a single relationship. However, if using MS Excel or some other similar tool, both of these relationships must be established manually. Also, identify the post-traceability steps to ensure the completeness of the traceability, (i.e., nothing was skipped or missed). Also, traceability to the verification Levels must be defined. When a requirement change is approved, the requirements baseline shall be modified to incorporate the change, which will create a new requirements baseline.

4.7 Managing Requirements Change

After the requirements for each Level are accepted, the requirements are baselined, loaded into the requirements repository, and any change to these requirements will be prohibited unless formally approved by the Project Change Management Plan processes. The requirements management activities defined within this Plan, in support of the Change Management process, are 1) to perform an impact analysis to identify the impact(s) of a proposed change to other baseline requirements at all Levels, 2) to formally capture and update the approved baseline change(s), and 3) to verify that all traced requirements, and their associated artifacts, are also updated to ensure that traceability is still complete, correct, and consistent after the change has been made to the repository.

4.7.1 Requirements Change Impact Assessment

The following is an example for what the project must consider when developing the procedures for this section. It can be tailored for the specific project, re-written with the following content, deleted, or moved to an appendix for use as a general overview. Regardless, the project must document the procedures on how requirement modifications, additions, and deletions will be addressed.

Per the Project Change Management Plan, when a request for a change has been submitted and reviewed, an impact assessment will be requested. The requirements management process will provide input into the impact assessment by identifying all of the requirements and artifacts that will be impacted (potentially also requiring a change to be consistent with the proposed requirement change) by the proposed change and if requirements traceability will still be complete if the change is approved. The requirements management process is not responsible for evaluating the technical, business, management or other need for the change or the value of the change to the organization. The requirements management process assesses the scope of the impacts of the change on other baseline requirements and if traceability will still be complete if the change is approved. This is done by identifying the traceability relationships of the existing requirement proposed to change to all other baselined requirements and artifacts that the requirement is traced from (i.e., parent requirements), or traced to (i.e., children requirements).

As can be seen in Figure 7, if a Project[2] Level requirement (R X.2) is proposed to be modified (R X.2^{m}) and the project has approved a Project Approval Level requirement and a Requirements Analysis Level requirement that are traced to the Project Level requirement, then its parent requirement within the Project Approval Level (e.g., FSR) shall be identified (R X), all of the children of this parent requirement shall be identified (R X.1 and R X.2), and the children requirements of the requirement being modified (R X.2) within the Requirements Analysis Level shall be identified (R X.2.1 and R X.2.2). The requirements management process shall perform the following actions:

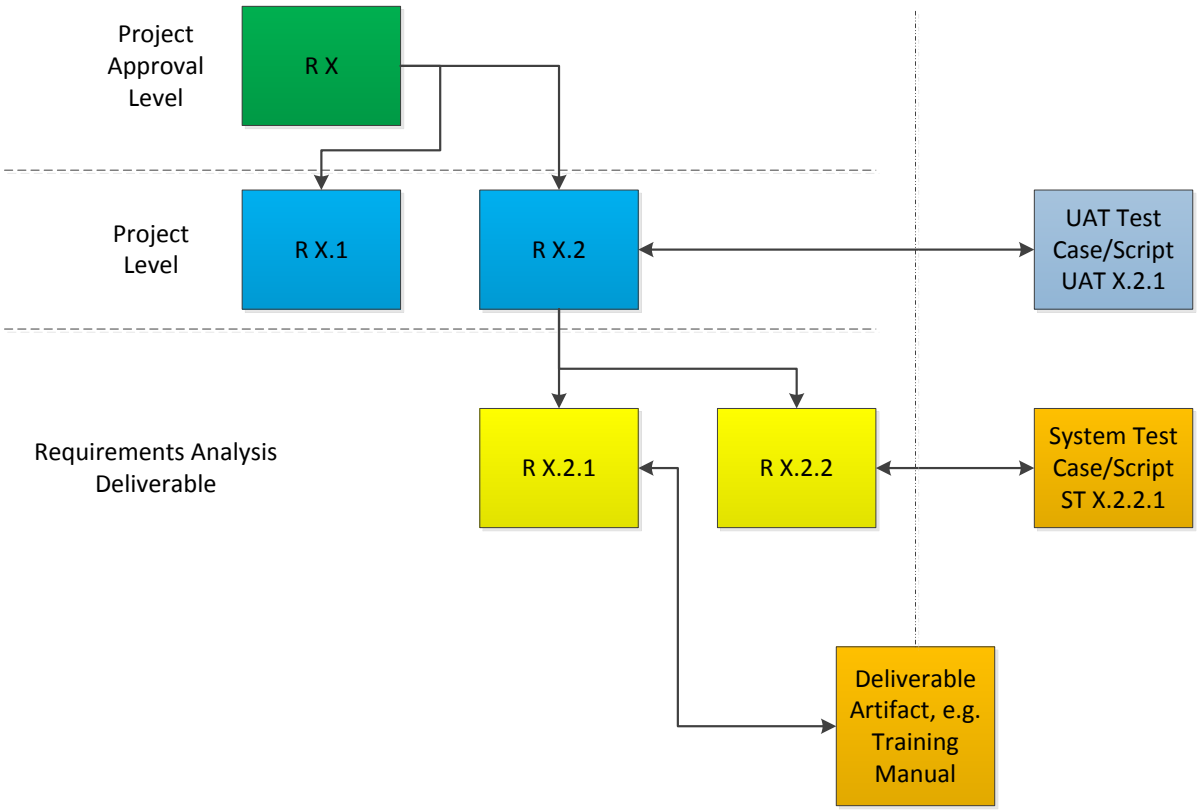
1. Identify the parent requirement (R X) of the existing requirement (R X.2) and all children of this parent requirement (R X.1 and R X.2)
2. Assess if the modified version of the requirement (R X.2^{m}) is still consistent and within the scope of its parent requirement (R X)
    - Identify if the modified requirement is still consistent with and within the scope of the parent requirement; if not, then report that the parent requirement may also need to be modified.
3. Assess if the children (R X.1 and R X.2^{m}) of the parent requirement still satisfies all of the needs of the parent requirement (R X)
    - When considering all of the children together, identify if the scope of the parent requirement is satisfied by all of the children; if not, then report that additional changes are needed to either the children requirements or to the parent.
4. Identify the children requirements (R X.2.1 and R X.2.2) of the requirement being modified (R X.2)
5. Assess if the children of the requirement (R X.2.1 and R X.2.2) are consistent, within the scope, and fully satisfy the modified requirement (R X2^{m})
    - Identify if the children of the existing requirement are still valid for the modified version and if the children are still consistent, within scope, and when considered together still satisfy the modified version of the requirement; if not, then report that additional changes may be needed.

[2] Within this section, it is assumed that the State assumed requirements, RFP requirements, and the contract requirements are maintained within the requirements repository as one (1) level, which is being termed as the "Project" level.

6. Identify the relationship between the identified requirements and test artifacts impacted at all test Levels (R X.2 ⇒ UAT X.2.1 and R X.2.2 ⇒ ST X.2.2.1) and other artifacts/Deliverable (R X.2.1 ⇒ Training Manual) that are traced to any of the requirements potentially impacted by the proposed change
    - Report the relationships to the test and deliverables that exist for the original requirement. Requirements management is not responsible for determining if there is an impact to these items but the relationships must be provided back to the Change Management Impact Analysis owner for them to review and determine if any additional impacts exist.
7. Report the identified potential impacts and the results of the assessments back to the Change Management process so that they can be considered prior to deciding on the change.

It is not within the scope of the Requirements Management Plan to assess and determine if an impact exists as this may be beyond the capabilities of the requirements management team. Requirements management only reports that a potential impact exists, which must be further examined and determined by the resources performing the impact assessment for Change Management.

Figure 7: Modified Requirements Assessment

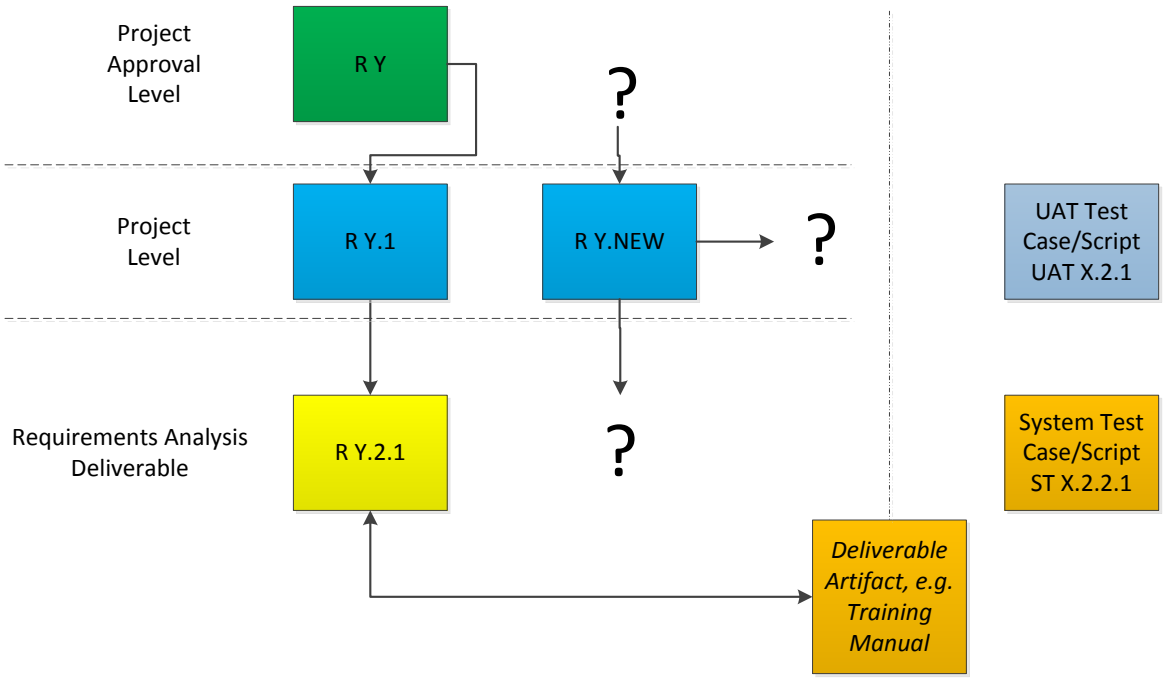


See Figure 8 for a scenario in which a requirement is being added (R Y.NEW) to the Project Level and where a parent requirement, children requirements, test cases/scripts, and/or deliverable must also be identified. To add a new requirement, the following process shall be executed:

1. Identify the parent requirement within the set of baseline Project Approval Level (e.g., FSR) requirements that identifies the scope or high-level requirement associated with the proposed new requirement (R Y.NEW)
    - If a parent requirement does not exist then report that a parent requirement must also be added as part of the change request

2. If a parent requirement exist, for this example assume the parent is RY, assess if the children (R Y.1 and R Y.NEW) of the parent requirement (R Y) still satisfy the needs of the parent requirement and that the children are within the scope of the parent and the children are not duplicative
    - If the new requirement is beyond the scope of the parent requirement, duplicative of other children requirements of the parent, or the parent requirements is still not satisfied by all of the children requirements then report that additional changes should be included in the change request.
3. Identify the requirements within the Requirements Management repository that the new requirement should trace to and if the lower-level requirements are present.
    - If traceability has been performed to the next Level, to verification, or to deliverables, then identify and report if there are missing children requirements, missing test cases/scripts, and/or missing deliverables, if any are required. (Note: Only identify missing requirements, test cases/scripts, and deliverables if traceability has been performed to these items. For example, if UAT Test Case/Scripts have not been developed and therefore not traced to, then the fact that the new requirement cannot be traced to UAT Test Case/Scripts in irrelevant and does not need to be reported. Similarly, if the Requirements Analysis Level requirements have not been baselined then there is no need to identify that there are no children requirements for the new requirement.)
4. Report the identified potential impacts and the results of the assessments back to the Change Management process so that they can be considered prior to deciding on the change.

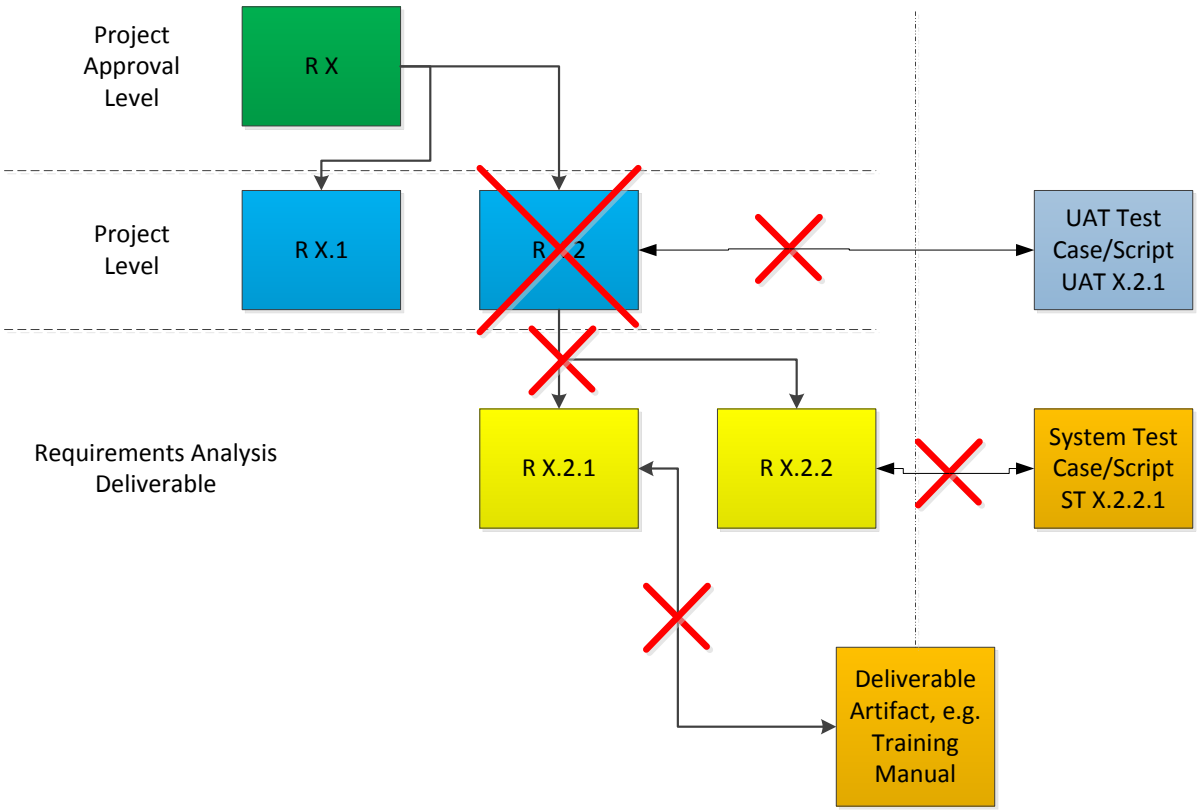
Figure 8: Added Requirement Assessment



See Figure 9, for a scenario where a requirement (R X.2) is being deleted. If a requirement is being deleted, the Requirements Management process must verify that the requirements' parent requirement (R X) is still being fulfilled by other child requirements (R X.1) and that all of the requirements' child requirements are also deleted, assuming they only have one parent requirement. To delete a requirement, the following process shall be executed:

1. Identify the parent requirement (R X) of the requirement being deleted (R X.2) and all children of the parent requirement (R X.1 and R X.2)
2. Assess if the remaining children (R X.1) of the parent requirement (R X) still satisfies all of the needs of the parent requirement
    - Identify if the remaining children requirements still provide all of the needs required by the parent requirement; in not, report that additional changes are needed to either the parent requirement, or the other children requirements must be modified, or a new requirement is needed.
3. Identify the children (R X.2.1 and R X.2.2) of the proposed deleted requirement (R X.2), the artifacts they are documented within (e.g. the contract), and the test case/scripts and deliverables that would be impacted and need to be updated should the requirement and children be deleted.
    - For the traceability that has been completed, trace the proposed deleted requirement down to its children and to the test cases/scripts and deliverables to identify if the children requirements and/or test cases/scripts must also be deleted and if the requirement traces to a deliverable that must also be updated to remove the requirement. If additional changes are required, report these additional changes.
4. Report the identified potential impacts and the results of the assessments back to the Change Management process so that they can be considered prior to deciding on the change.

Figure 9: Delete Requirement Assessment



4.8 Updating Baseline Requirements

Once a change is approved, the requirements baseline within the repository must be updated. This includes loading the approved change request into the repository, identifying its requirement(s), setting the attributes, and then updating the traceability links between the requirement, its parent requirement(s), children requirement(s), test cases/scripts, and deliverables that were identified as impacted in the Section 4.7.1 Requirements Change Impact Assessment.

Common to the modification, addition, or deletion of a requirement is the need to capture the approved change request and its requirement. The following procedures are the steps to capture the approved change request, identify and tag its requirements, and set its attributes.

NOTE: Include the procedures to 1) load the approved change request into the repository, 2) identify and tag the requirement(s) within the change request, and 3) set the requirement attributes.

Once this procedure is completed, the one of the following procedures shall be performed, depending on the type of change, to modify, add, or delete a requirement.

4.8.1 Modification Update

If a requirement is being modified, it is important to note that related/traced requirements and artifacts will likely not be submitted at the time the change request is approved or when updating the requirement. Therefore, when the requirement is being modified within the repository, all traceability links must be flagged as “suspect.” Then, each “suspect” link needs to be re-verified, either after a parent or child artifact is updated and approved and the impacted parent/child requirement is updated or, upon further analysis, the existing traceability relationship is still deemed valid.

Identify the procedure for modifying the requirement, changing all of the traceability links to suspect, re-assessing the requirement attributes, and re-verifying all of the traceability links. Critical within the procedure is the ability to maintain configuration control so that previous data is retained.

4.8.2 Addition Update

If a requirement is being added, then the requirement is added to the requirements repository into a Level/location appropriate for the structure of the repository and below the Level where the traceability relationship to a parent requirement will be established. Again, if the parent Level does not contain a requirement in which to trace the new requirement then no traceability link can be established until a new parent Level requirement is approved. Similarly, links to children requirements must be established, which may also require leaving the traceability relationship blank until the child artifact(s) and their associated requirements are updated and approved.

Identify the procedure for adding a requirement, how and where to add the requirement, assessing the requirement attributes, and establishing the traceability relationships/links.

4.8.3 Deletion Update

If a requirement is being deleted, the requirement within the requirements repository is only logically deleted by setting an attribute, such as Status, to “Deleted”; requirements themselves are never physically deleted. Therefore, the requirement’s attribute is set to logically delete the requirement. The traceability link to the parent requirement is physically deleted as are the links to the children requirements. The children requirements are not deleted unless the artifact in which they are documented is updated and approved to delete them. Then, once the child artifact is updated and approved that deletes the child requirements, the child requirements are logically deleted and the traceability relationships between each of them and other requirements, test cases/scripts, and deliverables are also deleted.

Care must be used in deleting requirements due to the potential cascade effect, meaning that by deleting a requirement at one Level may cause the deletion of all of its children, test cases/scripts and require a deliverable update, and each child may also have children, test cases/scripts, and deliverables that must be deleted or updated, and so on. This cascade effect must be carefully traced to avoid significant re-work to re-establish correct and valid traceability.

Identify the procedure to logically delete a requirement and to delete the traceability relationships/links.

4.9 Requirements Traceability

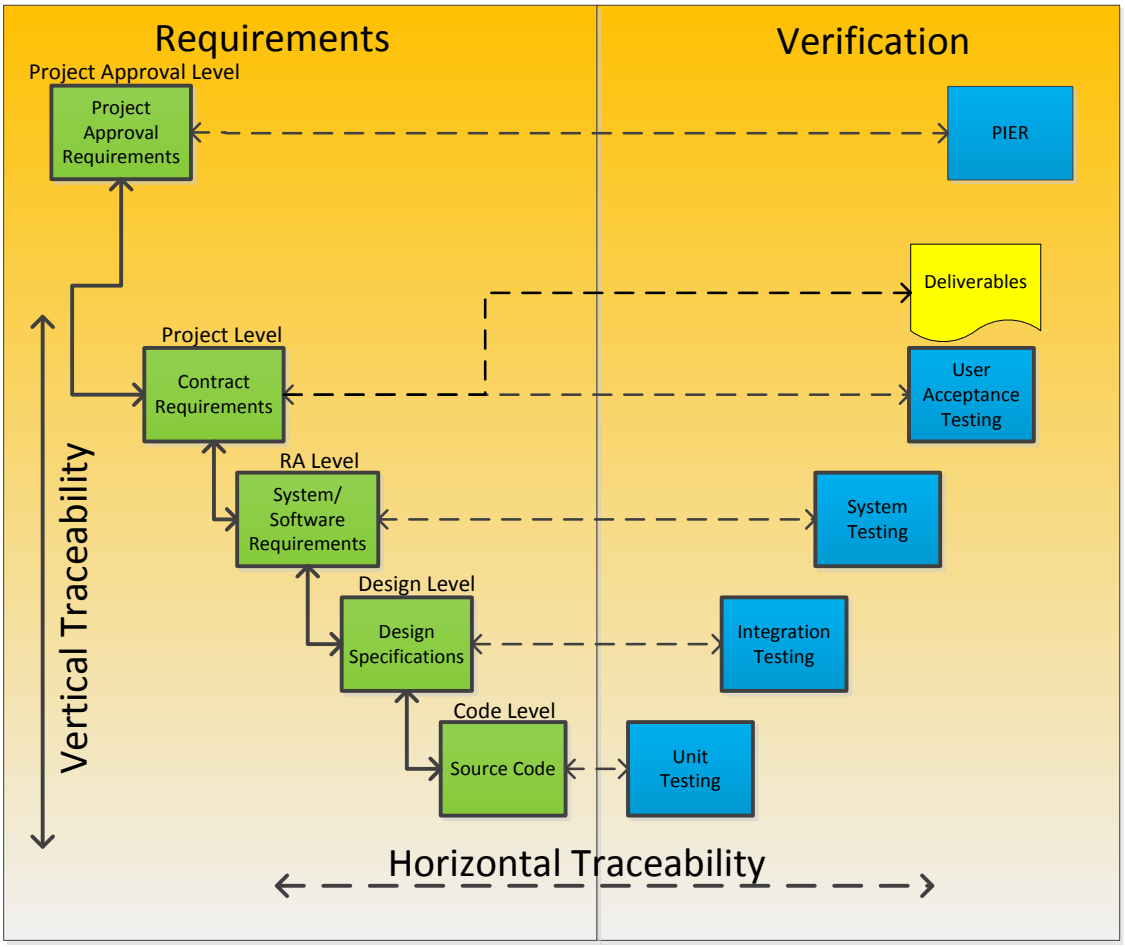
Once requirement sets from adjacent levels have been loaded, traceability shall be performed. Traceability shall establish relationships between higher level requirements (e.g., a requirement in the Project Approval Level) to lower level requirement(s), (e.g., one or more requirements in the Project Level). Within this Plan, higher level requirements are called the parent requirements of the lower level requirements and the lower level requirements are called the child or children requirements of the higher level requirement. Each parent requirement must have one or more child requirement(s) and a child requirement must have a parent requirement. By establishing a relationship between artifacts at different Levels, the project can ensure that all of the higher level requirements have been addressed and accounted for in the lower level requirements, which is termed downward traceability. Further, traceability ensures that all lower level requirements are actually required to meet the needs of a higher level requirement, which is termed upward traceability. Therefore, when a proposed change to a requirement is being considered, traceability can and will be used to identify all of the impacts associated with the change, both upward and downward.

When a requirement change is approved, the requirements baseline shall be modified to incorporate the change, which will create a new requirements baseline. The requirements baseline is kept and maintained within the requirements repository, not externally. Therefore, when a change request is approved, the change must be made in the requirements repository. Configuration control and security shall be maintained. Be sure to indicate how configuration control and security will be invoked in order to maintain the previous baseline, how the new baseline will be established, and how unauthorized access and ability to make changes will be prevented. If using a tool like MS Excel, this is a manual process whereas if using Requisite Pro, the history and version control and the security are provided by the tool itself.

Finally, when performing activities such as assessing the impact of a proposed requirements change, creating metrics, or generating status reports, the requirements repository is accessed in a read-only mode. Again, based on the tool that will be used, describe how read-only users will access the repository to collect the data necessary to perform these three functions.

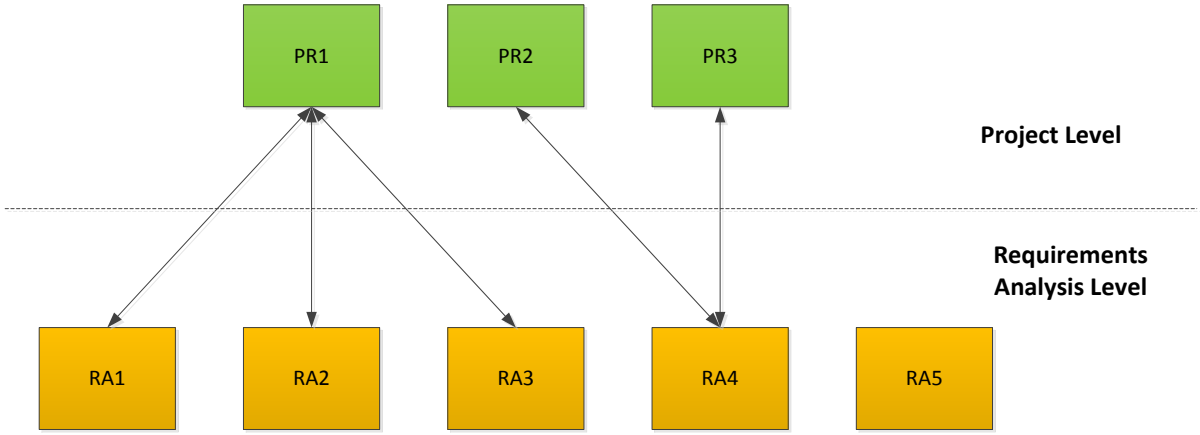
The following figure, Figure 10, graphically illustrates the relationships between and amongst various requirements Levels and general artifacts of a typical project. On the left-hand side of the figure are the requirement Levels and artifacts/requirement types. The requirement types are requirements that are documented in various artifacts, such as the project approval document (e.g., PAL S2AA) documents project approval type requirements, vendor contract documents contract type requirements, etc. On the right-hand side are verification Levels and artifacts. The artifacts are items such as test cases/scripts for each of the different Levels of testing as well as deliverables that have been approved/accepted.

Figure 10: V-Model Example



Vertical Traceability, as shown in Figure 10 on the left and by the solid arrows between the requirements Levels, is the identification of a requirement at one Level and how that requirement relates to a requirement at another Level. As an example, a specific contract requirement at the Project Level, say C1 in Figure **Error! Reference source not found.**11, may be further elaborated by equirements identified in the System/Software Requirements document, RA1, RA2, and RA3 within the Requirements Analysis (RA) Level. Vertical Traceability identifies this relationship and establishes a link, which is a reference that points from the individual contract requirement to a specific System Software Requirements document requirement. In practice, there is typically a one-to-many relationship, meaning one contract requirement is often related to one or more System/Software Requirements; this one-to-many relationship is also called the fan-out of N for the contract requirement, where N is the number of requirements related to or linked to the contract requirement. In Figure 11, Requirement PR1 has a fan-out of 3. This tracing, while vertical tracing, is also commonly called Downwards Traceability.

Figure 11: Fan-out and Fan-in Example



In addition to Downwards Traceability, Vertical Traceability also encompasses Upwards Traceability, as shown in Figure 10 and Figure 11 by the arrows also pointing upward. Upwards Traceability is performed by establishing a relationship between every lower Level requirement to a higher Level requirement. For example, each requirement in the Requirements Analysis Level must be related to a requirement in the Project Level, and a traceability link is established to capture this relationship. Notice in Figure 11 that RA1, RA2, and RA3 are all linked to requirement PR1; from RA1's perspective, it has a fan-in of 1 because it is linked to only one higher level or parent requirement. Occasionally, the fan-in may be greater than one as shown in Figure 11 for RA4, which is more prevalent when performing traceability to verification/test artifacts and deliverables discussed later. Note that RA5 does not have a traceability link identified. This means that either traceability is not complete or the RA5 requirement is not necessary and should be deleted as is does not support any approved need; this is also called "gold plating," which is adding additional requirements that are not required but may be nice to have.

By using a Requirements Traceability tool, the effort required to establish the relationships for traceability is greatly simplified. When using these tools, if a link or relationship is created between a higher Level requirement and a lower Level requirement (downwards), a link is also automatically created between the lower Level requirement and the higher Level requirement (upwards). After Downwards Traceability is complete, the only effort required for Upwards Traceability is to verify that all lower Level requirements have been traced.

Horizontal Traceability is similar but this traceability is between requirements at one Level to the artifact (a test case and/or script) that verifies the requirement. For most requirements, this will be performed between individual requirements at each Level of the requirements hierarchy on the left side of Figure 10 to a test case/script on the right side of the figure. However, some requirements within the Project Level, specifically from the contract, identify deliverables that must be accepted by the state (e.g., a Training Manual). Therefore, these Project Level requirements are traced to a deliverable and not to a test case/script[3]. As shown in Figure 10, this traceability is also bi-directional, meaning that testing should only be verifying baselined requirements, though some exceptions[4] can and do occur.

[3] A test case/script could be created for deliverables but this is generally not done; therefore, in order to complete the traceability, the requirement must be traced to the deliverable itself.

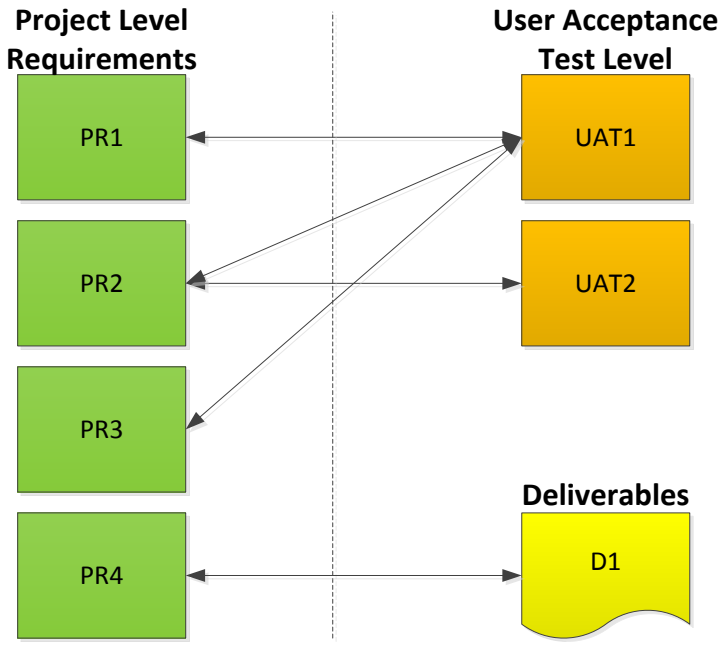
[4] Test cases and scripts may be developed for validation reasons (meaning will the system work for the current business) that do not directly tie or relate to baseline requirements. While this is done, care must be used because a defect from these types of tests does not imply that the system does not meet its requirements; the defect may be beyond the scope of the requirements and require a relatively expensive change request to implement.

Figure 12 (below) illustrates Horizontal Traceability between Project Level requirements and User Acceptance Test Cases/Scripts and Deliverables. In this example, requirement PR1 is traced only to UAT1, which means that the test case/script UAT1 verifies the entire PR1 requirement. Also note that UAT1 verifies part of the PR2 requirement and the entire PR3 requirement. This can be determined based on the fan-out of the Project Level requirements to the test cases/scripts. By looking at the fan-in of the test cases/scripts, it is evident that UAT1 supports the verification effort for three Project Level requirements whereas UAT2 only supports one Project Level requirement.

While bi-directional traceability is important for a number of reasons, one of most obvious reasons from a testing perspective is that if UAT1 is blocked or unable to execute, then this blockage is preventing the verification of three Project Level requirements; if UAT2 is blocked it is only blocking the verification of one Project Level requirement. Therefore, a Project may want to put more priority on getting UAT1 unblocked than UAT2 so more Project Level requirements can be verified. Also, if test case/script UAT1 needs to be modified then any modification must take into account all three of the linked requirements so that the requirement are still being verified by the test case/script; UAT2 only needs to be concerned with the PR2 requirement. A common situation is that a Project Level requirement is changed and will require a change to the test case/script; therefore, knowing the test case/script that verifies the changing requirement and all of the other requirements verified by the test case/script is essential in order to make the change to the test case/script and still ensure the test case/script verifies all of the requirements traced to it.

Project Level requirement PR4 traces to a deliverable, D1. While this type of traceability is not required to be bi-directional, it is generally best practice to use bi-directional traceability for baselined deliverable changes due to requirement changes. Requirements Traceability tools establish bi-directional traceability automatically, by default, so when these tools are used there is no additional work or effort required to perform bi-directional traceability.

Figure 12: Horizontal Traceability Example



A question always arises as to why a project needs to perform Requirement Traceability. While there are numerous reasons, the main reasons for performing traceability is to:

- Ensure that every requirement necessary for the approved project is developed and verified.
- Ensure that the business, technical, management, and support requirements are progressively elaborated, defined, and developed and that no requirements for functionality, needs, or services are lost (downward vertical traceability).
- Ensure that no additional or unapproved requirements have been added (upwards vertical traceability)
- Ensure that all requirements at all Levels of elaboration/abstractions (Project Approval through to Code) are verified and that the delivered system will meet the defined business, technical, management, and support needs (horizontal traceability).
- Provide a means for assessing impact if and when requirements change.

4.10 Repository Updates

Most projects engage in multiple efforts of requirements review with varying approaches and rigor to the review process. It is during this step that traceability and accuracy become critical to the requirements development process. The larger the project, the more stakeholders there are to include in a review, thus making review and modifications more time consuming. Scheduling time for both stakeholder reviews and requirement modifications can be a monumental challenge, especially during periods of decreased staff resources. Changes made to a requirement without stakeholders involvement or input can prove costly as they may lose confidence that the project is managing their concerns appropriately.
Due to these circumstances, it is critical that the project:

1. Accurately record decisions to modify requirements
    a. What is changing, who asked for the change, when, and what the impact is
2. Communicate the modification(s) to all project members and personnel who “need-to-know”
3. Verify that the modified requirement does not alter the intent or subsequently change the impact or meaning of a detailed solution requirement or mid-level solution requirement or objective.

These reasons and many more are argument for the establishment of a Requirements Change Process. Development without a process to request, analyze, document, and decide on requirement modifications could prove to severely impact cost, schedule, and/or solution expectations. At the crux of this process is the emphasis on communication of the request and its outcome.

As stated previously, the project stakeholders may still be unsatisfied with one or more sets of requirements. Decisions to resubmit them through the vetting and critique process should be made. Since a procurement that could potentially cost the state a large sum of money – whether it’s successful or not, is on the line. Therefore, every opportunity to perfect what the project is asking for should be taken. Once the solicitation is released, changes to the requirements become far more challenging and drastically more costly.