Quality Assessment Checklists for Requirements

PREVIOUS PAGE TABLE OF CONTENT NEXT PAGE

With the above framework in mind, we can move on to assessing the quality of each requirements set at its appropriate level of completeness. To assist your team in this endeavor, in this section we provide guidelines in the format of a quality assessment checklist for each team skill requirements artifacts set (Tables 29-2 through 29-7).

Many of the checklist items apply at any point of completion; others might apply only at some later and more final stage. In any case, those doing the assessment must keep in mind that the level of specificity and completeness must be appropriate for the particular iteration. There should be no "polishing of the artifact apple" in a contemporary and dynamic iterative software development process.

Table 29-2. Quality Assessment Checklist for Team Skill 1: Analyzing the Problem

Problem statement	Has a problem statement been drafted?	☐
	Is it written in an easy-to-understand way?	☐
	Does the team understand it?	☐
	Has it been circulated for agreement to the key stakeholders, including management?	☐
	Do the team members have agreement that this is the problem they are trying to solve (or the opportunity you are trying to address)?	☐
Root cause analysis	Was a root cause analysis performed?	☐
	Can the team members be sure they are addressing a real problem and not a symptom of a more basic problem?	☐

	Was sufficient effort invested in experimentation or other techniques to identify the root cause?	☐
Systems model	Is the solutions system boundary identified?	☐
	Have you identified all the things that interact with the system?	☐
	Has the system been partitioned into subsystems? If so, was the system decomposition driven by the right optimization criteria?	☐
	If so, have all the subsystems been identified?	☐
	Are the boundaries of each subsystem understood ?	☐
	Is there a plan for identifying and addressing derived requirements?	☐
List of stakeholders and users	Have you identified all the users of the system?	☐
	Have you identified all stakeholders who will be affected by the system?	☐
	Have you looked outside the sets of readily perceivable users and stakeholders and found the people dealing with administration, installation, and support or training?	☐
	How do the team members know they have identified them all?	☐
List of design and development constraints	Has the team identified all the constraints to be imposed on the system itself?	☐
	Has the team identified all the constraints to be imposed on the development process or project contracts?	☐
	Have all constraints sources (such as budget, product cost, political or contractual requirements, system requirements, environmental factors, regulations, staffing, software processes and tooling) been considered ?	☐
List of actors	Have you found all the actors? That is, have you accounted for and modeled all the things (users, devices, other systems and applications) that interact with the system?	☐
		☐

	Do any actors play similar roles in relation to the system? (If so, you should merge them into a single actor.)	☐
	Will a particular actor use the system in several completely different ways, or does the actor have several completely different purposes for using the use case? (If the actor uses the system in different ways, you should probably have more than one actor.)	☐
	Do the actors have intuitive and descriptive names? Can both users and customers understand the names ?	☐
Business use-case model	Is a business use-case model required to understand the intended functions of the proposed system?	☐
	Is a business object model required to understand the entities involved in the business processes?	☐
	Does the team understand what specific functions will be allocated to the proposed system?	☐

Table 29-3. Quality Assessment Checklist for Team Skill 2: Understanding User and Stakeholder Needs

Structured interview, process, and results	Was a structured interview employed?	☐
	Did it cover all the major facets of product requirements, purpose, usage, reliability, performance, deployment, support, and so on?	☐
	Was it sufficiently free of interviewer biases so as to assure a quality result?	☐
	Were a sufficient number of users or stakeholders identified and interviewed?	☐
	Are there other key influencers whose needs must be understood?	☐
Understanding of users and user needs	Do you understand who the users are and what capabilities they possess to apply your application?	☐
	Did you discover any primary user or demographic n the product?	☐

	Did the highest-priority needs converge after a reasonable number of interviews?	☐
	Are the user data, needs data, and any suggested features summarized somewhere for future reference?	☐
Requirements workshop process and results	Was a workshop conducted that included the requisite stakeholders?	☐
	Was it conducted in such a way as to encourage input by all stakeholders?	☐
	Did the results converge on a common understanding of the system to be built?	☐
	Was the development team engaged in such a way as to provide reasonable assurances of technical and project timeline feasibility?	☐
Preliminary list of prioritized features	Does a prioritized list of features exist?	☐
	Did the development team define rough estimates of effort for each?	☐
	Was the risk of each feature established?	☐
	Is this information captured somewhere for continuous reference?	☐
Storyboards, example use cases, and other expository artifacts	If the application is innovative, did you develop some means to demonstrate the application to the user?	☐
	Was their reaction taken into consideration and is it now reflected in your current understanding of the system?	☐
	Can you describe a few exemplary use cases that describe how the system is intended to be used?	☐

Table 29-4. Quality Assessment Checklist for Team Skill 3: Defining the System

Requirements	Have you established a plan for organizing requirements?	☐

organization	Do you understand what tooling you will apply to manage this process?	☐
	Does your organization system allow for capture of all types of requirements?	☐
	Are you on the lookout for design constraints?	☐
Vision document	Do you have a vision for the project?	☐
	Does it include input from relevant sources (authors/inventors, stakeholders, subject matter experts, development team) about key aspects of the project (system requirements, constraints, other systems and applications, competitive products)?	☐
	Is the vision captured in an established template (the Vision document) for this purpose?	☐
	Does it contain the requisite elements: user's profile, types, environments, product overview/perspective, product position statement, product features, applicable system requirements, and so on?	☐
	Have you established a Delta Vision document mechanism for future releases?	☐
Identification of initial use cases	Have you identified (named and described) the basic use cases that will be used to drive system development?	☐
Empowerment of product manager/project champion	Is there a product manager or project champion empowered by the team?	☐
	Is he or she the official source of feature-level changes?	☐
	Have you identified a product road map that defines external releases and the features currently planned for each release?	☐
	Do you know how you will describe the product (messaging) to the outside world?	☐
Definitions of commercial facto	Have you defined and captured (whole product plan) lation; pricing; ing; and	☐

	product naming, branding, and labeling?	

Table 29-5. Quality Assessment Checklist for Team Skill 4: Managing Scope

Prioritization and estimation of features	Have you estimated, prioritized, and assessed the risk for the various features that constitute the product vision?	
Requirements baseline	Have you established a requirements baseline for the release you are working on?	
	Do you understand what features are critical to this releases, as well as those that are important and useful?	
Recognition and communication of achievable scope	Does your project fit "in the scope box?" (Can it be executed with the available resources and within the available time line?)	
	Have you made the hard decisions for what can and can't be done during the known time line?	
	Have key managers and customer stakeholders agreed to this scope?	
Agreed-on expectations	Are expectations for the current release understood by the team?	
	Have the expectations been communicated and has agreement been reached with the key stakeholders outside the team, including the end user/customer?	

Table 29-6. Quality Assessment Checklist for Team Skill 5: Refining the System Definition

Use-case model(s)	If the system is composed of subsystems, does the use-case model appropriately reflect that?	
	Have you found all the use cases?	
	Do the use cases have unique, intuitive, and explanatory names so they cannot be mixed up at a later stage?	
	Are all required system behaviors identified in one or more use cases?	

	Do customers and users understand the names and descriptions of the use cases?	☐
	By looking at the use-case model, can you form a clear idea of the system's functions and how they are related ?	☐
Use-case model(s)	Do the elaborated use cases meet all the functional requirements?	☐
	Does the use-case model contain any superfluous behavior?	☐
	That is, does it present more functions than were called for in the requirements?	☐
	Does the model need the identified include and extend relationships?	☐
	Can the model be simplified with additional relationships?	☐
Use-case specifications	Is each use case involved with at least one actor?	☐
	Does the brief description give a true picture of the use case?	☐
	Is it clear who wishes to perform a use case? Is the purpose of the use case also clear?	☐
	Do the elaborated use cases contain the necessary sections and the appropriate content for name , actors, brief description, primary and alternate flow of events, pre- and post-conditions, and special requirements?	☐
	Is it clear how and when the use case's flow of events starts and ends?	☐
	Is each use case independent of the others?	☐
	Do any use cases have very similar behaviors or flows of events?	☐
	Has part of a use case's flow of events already been modeled as another use case?	☐
	Should the flow of events of one use case be inserted into the	☐

	Does the use case meet all the requirements that obviously govern its performance? Are use-case-specific nonfunctional requirements referenced where necessary?	☐
	Does the communication sequence between actor and use case conform to the user's expectations?	☐
	Is there a description of what will happen if a given condition is not met?	☐
	Are any use cases overly complex?	☐
	Are the actor interactions and exchanged information clear?	☐
Supplementary specification(s)	Have you established an appropriate template for your specific purposes?	☐
	Are almost all functional requirements included in the use-case model, and the balance, if any, reflected in the supplementary specification?	☐
	Have nonfunctional requirements such as usability, reliability, performance, and supportability all been identified and captured?	☐
	Have the appropriate design constraints been identified and captured?	☐
	Have supplementary requirements been linked to the use cases where appropriate?	☐
Ambiguity and specificity considerations	In general, has your team reached the appropriate level of specificity (the sweet spot) for your project context?	☐
	How do you know that this has been achieved?	☐
Technical methods (if any)	Have appropriate technical methods been employed to remove ambiguity in those cases where you cannot afford to be misunderstood?	☐
	If so, can these methods themselves be understood by the key stakeholders?	

Table 29-7. Quality A

Transitioning method (from design to code)	Do you understand the mechanism by which you'll be transitioning from requirements into design and implementation?	☐
	Is there a use-case realization (collaboration) for all use cases in the use-case model?	☐
	Are there realizations for other functional requirements as well?	☐
Test cases (derived and traceable from use cases)	Have the use cases been used to seed test case development?	☐
	Have you followed the four-step process (identify scenarios, identify test cases, identify test conditions, add data values)?	☐
	Are there one or more test cases for every use case?	☐
Requirements traceability	Have you established a plan for requirements traceability?	☐
	Have you identified and implemented adequate tooling?	☐
	Have you identified and followed a specific traceability model for this project?	☐
	Have you exploited implicit traceability to the maximum extent possible?	☐
	Have you applied explicit traceability in all critical areas?	☐
Requirements change management process	Do you understand the change sources and change dynamics for this project?	☐
	Do you know a change when you see it?	☐
	Does the project champion/product manager have control of this process?	☐
	Is an appropriate change control board established and is it functional for your project?	☐

	Can you capture and manage change effectively with the tooling you've deployed?	☐
	Do you have a way to capture and track defects on the project?	☐
Requirements method [*]	Did you pick an appropriate requirements method?	☐
	Does it reflect the key priorities of criticality and safety on the project?	☐
	Does the method eliminate unnecessary documentation and overhead?	☐
	Does the tooling adequately support the method you've chosen ?	☐

[*] We'll talk more about these issues in Chapter 30.

PREVIOUS PAGE TABLE OF CONTENT NEXT PAGE

Validating Form Data
Publishing Form Templates
Connecting Forms to Databases
Designing InfoPath Web Service Clients
Appendix D Problem Change Report
Appendix P Project Statement of Work
Appendix V Functional Configuration Audit (FCA) Checklist
Appendix Y Supplier CM Market Analysis Questionnaire
Process Switching
Neighbor Authentication
Sample Legacy DDR Configurations
Encrypted Tunnels

CISCO CALLMANAGER FUNDAMENTALS (2ND EDITION)

Call Hunting Constructs
Case Studies
Definition of Station Devices
Overview of Station Devices Supported by CallManager
H.323 Signaling

ORACLE SQL*PLUS: THE DEFINITIVE GUIDE (DEFINITIVE GUIDES)

Creating and Loading the Sample Tables
Saving and Retrieving the Buffer
The Concept of Null
Saving Your Work
Looping in SQL*Plus

PYTHON STANDARD LIBRARY (NUTSHELL HANDBOOKS) WITH

The operator Module
The array Module
The grp Module
The symbol Module
The readline Module