SOFTWARE REQUIREMENTS ESSENTIALS

Core Practices for Successful Business Analysis

KARL WIEGERS | CANDASE HOKANSON

Foreword by **Joy Beatty**

Praise for *Software Requirements Essentials*

"As research for a book, I once read the ten best-selling requirements engineering books of the prior ten years. This one book succinctly presents more useful information than those ten books combined. I wish I'd had it as a reference then."

—***Mike Cohn**, author of* User Stories Applied *and co-founder of the Scrum Alliance*

"Diamonds come about when a huge amount of carbon atoms are compressed. The compression crystallizes to form diamonds. Karl and Candase have done something very similar: they have compressed their vast requirements knowledge into 20 gems they call 'core practices.'

"These 20 practices give you the essence of requirements discovery, and for extra convenience they are categorized to make your requirements journey more effective. These practices are potent stuff, and I recommend that they become part of everyone's requirements arsenal."

—***James Robertson**, author of* Mastering the Requirements Process *and* Business Analysis Agility

"What a valuable resource for new and experienced business analysts alike, who want an accessible, clearly written, and well-organized introduction to key business analyst practices. Karl and Candase do a great job of breaking down a complex role into a straightforward set of practices that can be integrated into your business analysis process to make it more effective."

—***Laura Brandenburg**, author of* How to Start a Business Analyst Career

"Candase and Karl have drawn upon their deep knowledge and experience of what it takes to elicit, identify, represent, communicate, and validate requirements for software products effectively. They have produced a useful, accessible, and clear book, which is full of practical advice, great examples, and answers to the hard questions that people building software products face in the real world. If you're involved in building software in any role, this book will give you guidance on ways to make sure the product meets customer needs and delivers real value."

—***Shane Hastie**, Global Delivery Lead at SoftEd and Lead Editor, Culture and Methods at InfoQ.com*

"*Software Requirements Essentials* will be a high-value addition to your business analysis library. I give the book high marks, as it does an excellent job of selecting and comprehensively covering the most essential business analysis practices teams should be considering. I thoroughly appreciated that the content was not overdone. Lessons were succinct while remaining extremely usable. Care was taken to ensure the guidance was applicable

Each type of user has various tasks they wish to perform on the platform, certain usage privileges, and particular functional and quality expectations of the product. If you were launching such a publishing platform, you'd want to characterize the various user classes and then seek suitable representatives to understand each group's requirements and constraints. We'll revisit Speak-Out.biz in some upcoming practices.

Characterizing Stakeholders

Early in the project, perform an analysis to understand who your stakeholders are, each group's interests in the project, their influence over it, and their expectations and concerns (Gottesdiener 2005, Lucidchart 2022b). Questions like the following will provide a rich understanding of each stakeholder group (McManus 2005, Wiegers 2022).

- Who are they? How many of them are there?
- Where are they? What's the best way to communicate with them?
- What roles do they play with respect to the project?
- How much power or influence do they have over the project?
- What are their interests, concerns, and fears?
- What benefits do they wish to receive from the product? What are their needs, expectations, and success criteria?
- What information can they provide regarding operations, technology, data, or other areas?
- What do they need to know about the project?
- For users, how will they use the product?

Table 2.3 illustrates a simple template with information to record for each stakeholder profile, using one of the stakeholders from Speak-Out.biz as an example. Your organization could accumulate these profiles from multiple projects into a reusable, enterprise-level stakeholder catalog. Stakeholder profiles go into section 3.1 of the vision and scope document template, which was shown in Figure 2.2. If you already have a stakeholder catalog, you can just point to the appropriate entries in that section of the vision and scope document rather than duplicating the information.

Table 2.3 *Portion of a sample stakeholder profile for Speak-Out.biz*

Stakeholder	Roles	Interests	Influence	Needs	Concerns
Author	Writes, edits, and posts articles; tracks statistics and earnings	Interest = High; reaching a broad audience; generating revenue from articles	Power = Low: can request features and report problems or abuse	Easy-to-use text editor; submitting articles to publications; customizable statistics reports	Integrity of posted articles; long-term stability of platform

With your stakeholder catalog in hand, identify individuals who can accurately represent the interests of each stakeholder group and agree upon how they will engage with the BA and other team members. Make sure those representatives will have the bandwidth to contribute to the initiative in the necessary time frame. It can be more challenging to find and interact with stakeholders outside the developing organization than with internal people. However, their participation may be critical because of their authority, responsibilities, control over resources, or political, legal, or regulatory influence.

Consider whether each group is a collaborative partner in development, can make final decisions about certain aspects of the project, should be consulted for their expertise on specific issues, or just needs to be informed of progress and decisions that affect them (Leffingwell 2011). Some teams create a *RACI matrix* to identify stakeholder roles and their responsibilities with respect to the project. The RACI matrix shows which stakeholders are ***R***esponsible, ***A***ccountable, ***C***onsulted, or ***I***nformed (Morris 2022).

Make sure you know who is speaking for each stakeholder community. For the vital role of user representative, the *product champion* approach works well in many situations (Wiegers and Beatty 2013). A product champion is a designated and empowered key representative of a specific user class. Product champions work closely with BAs through interviews, workshops, prototype evaluations, and other collaborative activities to elicit and validate requirements. A product champion serves as the literal voice of the customer for a specific category of users. If you can't engage with actual user representatives directly, someone must still speak for the needs of each user class as a proxy.

The time spent on stakeholder analysis early on might seem like a distraction from real software work. Not so—it *is* the real work of ensuring that you engage the right participants in a collaborative effort that builds a solid base for success.

Related Practices

Practice #1. Understand the problem before converging on a solution.

Practice #2. Define business objectives.

Practice #3. Define the solution's boundaries.

Practice #5. Identify empowered decision makers.

Practice #6. Understand what users need to do with the solution.

Practice #13. Prioritize the requirements.

Next Steps

1. Examine Table 2.2 for any stakeholders for your initiative that might have been overlooked.
2. Characterize each of your significant stakeholder groups so that you can identify appropriate representatives and agree on how to engage with them.
3. Set up a template format for a stakeholder catalog and begin populating it with your stakeholder information for this project. If your organization works on projects that have recurrent stakeholders, establish a mechanism to maintain and reuse this catalog on future initiatives.
4. Identify your significant user classes. Note any user classes that are favored over others. Make sure it's clear who presents the requirements, constraints, dependencies, and risks for each user class. Confirm that those individuals have the knowledge and authority to perform their representation role well.

Practice #5 Identify empowered decision makers.

Every project faces a continual stream of decisions large and small. Individual team members can make many decisions locally and informally; other issues have a far broader impact. Making a considered decision about a requirements issue often demands obtaining input from multiple sources, having appropriate stakeholders assess the options, and communicating the outcome—and the reasons for it—to

all affected communities. Common classes of requirements-related decisions include these:

- Resolving conflicting requirements within a user class and across user classes
- Prioritizing requirements of various types
- Resolving conflicting priorities among different stakeholders
- Adjusting priorities as new needs come along and project realities change
- Making trade-off choices between conflicting quality attributes
- Defining the number of development increments or releases and the scope of each one
- Determining which new or changed requirements to incorporate into development plans (product backlog management)
- Deciding when and how to modify the scope of a planned development increment, a product release, or the entire project

Some organizations—and individuals—are better at making decisions than others. Karl used to work at a company in which decision-making was sluggish because no one wanted anybody to be uncomfortable with the outcome. That's not practical. As a colleague pointed out, "This company is not a democracy." Someone must choose among the options and set directions so that everyone can work toward the shared objectives. It was frustrating to deal with managers who vacillated, never reaching closure on issues that were appropriate for their level. Karl had more respect for managers who *would* make a decision, even if he didn't always agree with it.

It's important to determine who the decision makers will be for various requirements issues. The initiative's leaders should do this before the group confronts their first significant decision. Identifying the decision makers ensures that decisions can be made at the lowest possible level. Issues are resolved more quickly when decisions are made locally than when small matters are escalated to a higher level. Each group also should agree upon how they will reach their conclusions—that is, which decision rule or rules they'll apply—and the path forward if they're unable to resolve an issue.

Who Makes the Call?

The right people to make each category of decision depends on the situation. Major scope issues that affect schedules, resources, and existing commitments will involve senior managers or executives. Those managers could be in the developing organization, a customer organization, marketing, or combinations of those. A single

requirement change could have a big ripple effect if it forces revisions to multiple interconnected elements. Representatives from all the affected components need to know about the decision outcome, even if they don't all participate in making it.

Identifying the decision makers for requirements issues is a part of stakeholder analysis. Input from those stakeholders whose interests align most closely with the project's business objectives, such as favored user classes, should carry the most weight. Stakeholders who impose constraints—including scope, resource, regulatory, legal, business policy, or technical restrictions—may override functionality requests from other groups that conflict with the constraints. Agreeing on which stakeholders contribute most heavily to which important decisions helps the group reach conclusions more quickly and perhaps with less rancor.

Each decision-making group should identify a decision leader to coordinate their activities. The idea is not to add bureaucratic overhead, but rather to provide clear lines of responsibility, authority, and accountability. A group of people might think they have the authority to make decisions about a certain range of issues. But if someone else can override their choices, then, in effect, that group is merely an advisory body; the other "someone" is the ultimate decision maker. The decision leader makes all those roles and responsibilities clear to avoid delays, uncertainty, revisited decisions, and hard feelings.

On typical agile projects, the product owner (PO) is the decision leader for requirements-related issues. This is consistent with the PO's responsibilities for creating, prioritizing, and managing items in the product backlog to guide the team toward achieving the desired outcome (Agile Alliance 2022a). The PO's central role is sometimes described as being the "single wringable neck" in case things go awry (Bernstein 2016). We hope that's meant to be tongue in cheek.

One large project, for which Karl was the lead BA, assembled a user requirements team with representatives from four user classes: product champions, as described in Practice #4, "Identify and characterize stakeholders." The largest and most diverse user class required additional representatives from several subgroups to cover the breadth of needs. When requests from the subgroups didn't agree, the product champion for that overall user class was fully empowered to make the choice. And he did! The other participants respected the product champion's experience, wisdom, and conclusions. Having a few carefully selected and empowered user representatives who could make requirements decisions on behalf of their respective communities made life much easier for the three BAs.

How Do They Decide?

Too often, when people begin to collaborate on some initiative, they don't discuss exactly *how* they're going to work together. An important—and sometimes

adversarial—aspect of collaboration is making high-impact decisions that influence the project's direction. When Karl began his first experience coauthoring a book, he and his coauthor spent a lot of time planning how they would work together, including how they would resolve conflicts regarding how to handle a particular topic. Those involved with every multiperson activity should have this same discussion before they confront their first conflict.

Decisions related to requirements should use the business requirements as their North Star. Rely on your business objectives to make choices that keep the work focused on delivering the desired value. Some teams write their business objectives, vision statement, and scope descriptions on a large poster. They bring this poster to discussions about requirements (or present a refresher slide show to open virtual meetings) to help them choose the appropriate course of action. These actions remind everyone of the goals they're working together to achieve.

The objective of all decision-making is to expeditiously and respectfully reach closure on issues based on accurate information, thoughtful analysis, and honest negotiation. The process for making a decision is called a *decision rule*. There are numerous possible decision rules, including these (Gottesdiener 2002, Pichler 2016):

Unanimous Vote. The participants vote on the options, and all must vote the same way to resolve the issue. It can be time-consuming, and sometimes impossible, to lead a group of people with diverse interests to all agree on a given outcome. If achieved, unanimity provides the strongest group commitment to the decision. An outside facilitator can help a group achieve either unanimous agreement or consensus when buy-in across the board is essential.

Consensus. All participants agree that they can live with the group's decision, although they may vary in their commitment to it and their comfort level with it. Consensus is not as strong an agreement as unanimous voting. Reaching a consensus often requires considerable discussion and compromise. It takes longer than a simple vote, but the consensus-building process achieves more solid buy-in to significant decisions.

Plurality Vote. The decision makers vote on the options, and the one that receives the most votes is selected as the decision. Plurality (sometimes called majority) voting is most appropriate for low-impact decisions that have several clear options.

Decision Leader Decides. A single individual can make decisions more quickly than a group can. Depending on the decision leader's knowledge and expertise regarding the issue, the leader can either solicit input from others or reach a conclusion on their own. Soliciting others' input is more collaborative and promotes