

# REQUIREMENTS MANAGEMENT 101

## FOUR FUNDAMENTALS THAT EVERYONE SHOULD KNOW

**Wish someone would explain** requirements management in plain English? Have stakeholders that could benefit from understanding the value at a high-level? Your executives might not care about CMMI, BABOK or the nitty gritty details of functional specifications, but they do care about delivering what was promised to customers on time. And that's the value of requirements management.

Too often projects fail due to poorly managed requirements. At the core of the issue is that projects are increasingly complex, changes occur and communication is challenging. In this paper we will discuss the significance of requirements management without using industry jargon, as well as, the four fundamentals every team member and stakeholder needs to know. No Ph.D. required, no 2-week certification course, just the basics to help your team deliver what you promised.

# WHY SUCCESSFUL TEAMS DO REQUIREMENTS MANAGEMENT

Requirements management is about keeping your team in-sync and providing visibility to what is going on within a project. It is critical to the success of your projects for your whole team to understand what you are building and why – that’s how we define requirements management. The “why” is important because it provides context to the goals, feedback and decisions being made about the requirements. This increases predictability of future success and potential problems, allowing your team to quickly course correct any issues and successfully complete your project on time and within budget. As a starting point, it’s valuable for everyone involved to have a basic understanding of what requirements are, and how to manage them.

## Let’s start with the basics

A requirement is a document (a.k.a. artifact) that defines what you are looking to achieve or create – it identifies what a product needs to do, what it should look like, and explains its functionality and value. A requirement can be expressed with text, sketches, detailed mockups or models, whatever information best communicates to an engineer what to build, and to a QA manager what to test.

Depending on your development process, you might use different terminology to capture requirements. High-level requirements are sometimes referred to simply as “needs” or “goals”. Within software development practices, requirements might be referred to as “use cases”, “features” or “functional requirements”. Even more specifically, within agile development methodologies, requirements are often captured as “epics” and “stories.”

Does everyone know what we’re building and why?

That’s the value of requirements management.

Regardless of what your team calls them or what process you use, requirements are essential to the development of all products. Without clearly defining requirements you could produce an incomplete or defective product. Throughout the process there can be many people involved in defining requirements. A stakeholder might request a feature that describes how the product will provide value in solving a problem. A designer might define a requirement based on how the final product should look or perform from a usability or user interface standpoint. A business analyst might create a system requirement that adheres to specific technical or organizational constraints.

For today’s sophisticated products and software applications being built, it often takes hundreds or thousands of requirements to sufficiently define the scope of a project or a release. Thus, it’s imperative that the team be able to access, collaborate, update, and test each requirement through to completion, as requirements naturally change and evolve over time during the development process.

Now that we’ve defined the value of requirements management at a high-level, let’s go deeper into the four fundamentals that every team member and stakeholder can benefit from understanding:

1. **Planning good requirements:** “What the heck are we building?”
2. **Collaboration and buy-in:** “Just approve the spec, already!”
3. **Traceability & change management:** “Wait, do the developers know that changed?”
4. **Quality assurance:** “Hello, did anyone test this thing?”

# 1. PLANNING GOOD REQUIREMENTS

So what makes a good requirement? A good requirement should be valuable and actionable; it should define a need as well as provide a pathway to a solution. Everyone on the team should understand what it means. Requirements vary in complexity. They can be rough ideas sketched on a whiteboard to structured “shall” statements. They can be part of a group with high-level requirements broken down into sub-requirements. They may also include very detailed specifications that include a set of functional requirements describing the behavior or components of the end product.

Good requirements need to be concise and specific, and should answer the question, “what do we need?” Rather than, “how do we fulfill a need?” Good requirements ensure that all stakeholders understand their part of the plan; if parts are unclear or misinterpreted the final product could be defective or fail.<sup>1</sup> Preventing failure or misinterpretation of requirements can be aided by receiving feedback from the team continuously throughout the process as requirements evolve. Continuous collaboration and buy-in with everyone is a key to success.

# 2. COLLABORATION & BUY-IN

Is everyone in the loop? Do we have approval on the requirements to move forward? These questions come up during development cycles. It would be great if everyone could agree on requirements, but for large projects with many stakeholders, this does not usually happen. Trying to get everyone in agreement can cause decisions to be delayed, or worse, not made at all. Gaining consensus on every decision is not always easy. And in practice, you don’t necessarily want “consensus,” you want “buy-in” from the group and approval from those in control so you can move the project forward. With consensus, you are trying to get everyone to compromise and agree on the decision. With buy-in, you are trying to get people to back the best solution, make a smart decision and do what is necessary to move forward. You don’t need everyone to agree that the decision is the best. You need everyone to support the decision.<sup>2</sup>

Team collaboration can help in receiving support on decisions and in planning good requirements. Collaborative teams work hard to make sure everyone has a stake in projects and provides feedback. Collaborative teams continuously share ideas, typically have better communication and tend to support decisions made because there is a shared sense of commitment and understanding of the goals of the project. It’s when developers, testers or other stakeholders feel “out of the loop” that communication issues arise, people get frustrated and projects get delayed.

Once everyone has bought-in to the scope of work, it is imperative for requirements to be clear and well documented. Keeping track of all the requirements is where things get tricky. Imagine having a to-do list a mile long that involves collaborating with multiple people to complete. How would you keep all those items straight? How would you track how one change to an item would affect the rest of the project? This is where traceability and change management add value.

# 3. TRACEABILITY & CHANGE MANAGEMENT

Requirements traceability is a way to organize, document and keep track of the life of all your requirements from initial idea through to testing. A simple metaphor for traceability is connecting the dots to identify the relationships between items within your project. Here is an example of a common downstream flow:



You should be able to trace each of your requirements back to its original business objective. By tracing requirements, you are able to identify the ripple effect changes have, see if a requirement has been completed and whether it's being tested properly. Traceability and change management provide managers peace of mind and the visibility needed to anticipate issues and ensure continuous quality.

Traceability also allows for coverage that provides the ability to make sure the product meets all the vital requirements. Because requirements come from different people – from the customers to the engineers – each person contributes different requirements to the project. By tracing requirements, you ensure your entire team stays connected to the interdependencies of other items and the people working on those items.

Managing change is important and prevents “scope creep” within projects and releases. Scope refers to “the work that needs to be accomplished to deliver a product with the specified features and functions.”<sup>3</sup> Scope creep refers to unplanned changes in development that occur when requirements are not clearly captured, understood and communicated. The benefit of good requirements is a clear understanding of the end product and the scope involved. This leads to a better development schedule and budget, which prevents delays and cost overruns.

## 4. QUALITY ASSURANCE

Getting requirements delivered right the first time can mean better quality, faster development cycles and higher customer satisfaction with the end product. Requirements management not only helps you get it right, but also helps your team save money and many headaches throughout the development process. Concise, specific requirements can help you detect and fix problems early, rather than later when it's much more expensive to fix.

Research has shown that project teams can eliminate 50-80 percent of project defects by effectively managing requirements. In addition, it can cost up to 100 times more to correct a defect later in the development process after it's been coded, than it is to correct early on while a requirement.<sup>4</sup> By integrating requirements management into your quality assurance process, you can help your team increase efficiency and eliminate rework. And, rework is where most of the cost issues occur. According to the Carnegie Mellon Software Engineering Institute, “60-80 percent of the cost of software development is in rework.” In other words, development teams are wasting the majority of their budgets on efforts that aren't performed correctly the first time. For example, a developer codes a feature based on an old specification document, only to learn later that the requirements for that feature changed. These types of issues can be avoided with requirements management best practices.

**In summary**, Requirements management can sound like a complex discipline, but when you boil it down to a simple concept – it's really about helping teams answer the question, “Does everyone understand what we're building and why?” From the business analysts, product managers and project leaders to the developers, QA managers and testers, along with the stakeholders and customers involved – so often the root cause of project failure is a misunderstanding of the scope of the project.

When everyone is collaborating together and has full context and visibility to the discussions, decisions and changes involved with the requirements throughout the lifecycle of the project, that's when success happens consistently and you maintain continuous quality. Also, the process is smoother with less friction and frustration along the way for everyone involved. And, isn't that something we'd all benefit from?

To learn more about requirements management, trends in software development and other related topics, be sure to check out Jama Software's Resource Center at:  
<http://www.jamasoftware.com/news-resources/whitepapers.php>

## ROLL THE CREDITS

Endnotes on articles referenced within Requirements Management 101.

1. To learn more about good requirement and ambiguous terms to avoid be sure to check out our “Ambiguous Terms To Avoid” whitepaper.
2. Kupersmith, Jonathan. “Don’t Bother Building Consensus.” Business Analysis Times. 2011. Web. 2011. <<http://www.batimes.com/kupe-kupersmith/dont-bother-building-consensus.html>>.
3. A Guide to the Project Management Body of Knowledge. 4th ed. Project Management Inst, 2008. Print.
4. “Effective Requirements Definition and Management.” Borland Software Corporation. 2009. Web. <[http://www.borland.com/resources/en/pdf/solutions/rdm\\_whitepaper.pdf](http://www.borland.com/resources/en/pdf/solutions/rdm_whitepaper.pdf)>.

## OTHER ARTICLES WE REFERENCED DURING OUR RESEARCH, AND RECOMMEND IF YOU’RE INTERESTED IN LEARNING MORE.

5. “Business Requirements Analysis.” MindTools. Web. <[http://www.mindtools.com/pages/article/newPPM\\_77.htm](http://www.mindtools.com/pages/article/newPPM_77.htm)>.
6. Frederick, Richard. “Introduction to Requirements - The Critical Details That Make or Break a Project.” Global Knowledge. 2007. Web. <[http://www.globalknowledge.fr/PDF/WP\\_IntroRequirements.pdf](http://www.globalknowledge.fr/PDF/WP_IntroRequirements.pdf)>
7. Larson, Elizabeth & Richard. Project Times. Web. <<http://www.projecttimes.com/articles/requirements-management-101.html>>.
8. Ludwig Consulting Services, LLC. 2009. Web. <<http://www.jiludwig.com/>>.
9. McMahon, Chris. “The Full Team Approach to Managing Requirements.” Search Software Quality. Serena, Web.
10. “Real Reuse for Requirements.” MKS. Web. <<http://www.mks.com/resources/data/documents/whitepapers/instances/real-reuse-for-requirements-14>>.
11. Robin, Goldsmith F. “Four Tips for Gathering Requirements for the New Business Analyst.” Search Software Quality. IBM, Web.
12. Wikipedia. Requirements Management, Traceability, Scope, 2011. Web. <<http://www.wikipedia.com>>.

## ABOUT JAMA SOFTWARE

From concept to launch, the Jama product delivery platform helps companies bring complex products to market. By involving every person invested in the organization's success, the Jama platform provides a structured collaboration environment, empowering everyone with instant and comprehensive insight into what they are building and why. Visionary organizations worldwide, including SpaceX, The Department of Defense, VW, Time Warner, GE, United Healthcare and Amazon.com use Jama to accelerate their R&D returns, out-innovate their competition and deliver business value. Jama is one of the fastest-growing enterprise software companies in the United States, having exceeded 100% growth in each of the past four years, during which time both Inc. and Forbes have repeatedly recognized the company as a model of responsible growth and innovation. For more information please visit <http://www.jamasoftware.com>.