

The Value of Requirements Management

The Four Fundamentals and What It Means
for Your Bottom Line



Executive Summary

Because products are increasingly becoming more and more complex, they often fail due to poorly managed requirements. A company, typically using some basic document to track all the requirements needed to create a new product, doesn't "see" the complete product development process, and the team doesn't always fully understand what it is building and why. With the right requirements management in place, however, an entire team can work seamlessly together because everyone is aware of all the goals, feedback and decisions, which increases the chance the company will complete the product on time and within budget.

Requirements Explained

A requirement defines what you are planning to create; it identifies what a product needs to do and what it should look like; and it describes the product's functionality and value. Naturally, requirements vary in complexity. They can be rough ideas sketched on a whiteboard or structured "shall" statements. They can be text, detailed mockups or models and can be part of a hierarchy with high-level requirements broken down into subrequirements. They may also be detailed specifications that include a set of functional requirements describing the behavior or components of a product. In short, they are the information that best communicates to an engineer what to build, and to a quality-assurance manager what to test.

Does everyone know what we're building and why? That's the value of requirements management.

Depending on the development process, requirements might be referred to using different terminology. High-level requirements are sometimes referred to simply as "needs" or "goals." Software development practices might refer to requirements as "use cases," "features" or "functional requirements." Agile development methodologies often capture requirements as "epics" and

"stories." Regardless of the terminology, requirements are essential to the development of all products. Without clearly defining requirements, companies risk creating incomplete or defective products.

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.

Four Fundamentals

Today's sophisticated products and software applications often take hundreds or thousands of requirements to sufficiently define the scope of a project or a release. Teams must be able to access, collaborate, update and test each requirement through to completion because requirements naturally change and evolve over time during the development process. They must also understand the four fundamentals of requirements management.

1. Good Requirements

A good requirement should be valuable and actionable. It should provide a pathway to a solution, and everyone on the team should understand what it means. 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?" With accurate requirements, stakeholders can understand their part of the plan. If they lack this knowledge, if requirements are unclear or vague, the final product could be defective or fail.

2. Collaboration and Buy-in

It's difficult to get a company to agree on requirements, particularly for large projects with many stakeholders. In practice, it's not necessary to achieve consensus through compromise. It's more important to have team buy-in (before or after management approves the project) so the development process can move forward. With buy-in, the team backs the best solution, makes a smart decision and does what is necessary to move forward (batimes.com/kupe-kupersmith/dont-bother-building-consensus.html).

Team collaboration is key to establishing good requirements. Collaborative teams work hard to make sure everyone has a stake in the project and provides feedback. When there is a commitment and understanding of project goals, team members tend to support other's decisions. It's when developers, testers or other stakeholders feel "out of the loop" that communication issues arise, people get frustrated and projects get delayed.



3. Traceability and Change Management

Requirements traceability is a way to keep everyone in the loop. It organizes, documents and keeps track of all requirements, from initial idea to testing. A simple metaphor for traceability is connecting the dots to identify the relationships between items within a project. The following figure shows an example of a common downstream flow.

Companies should be able to trace each requirement back to its original business objective. By tracing requirements, companies can identify the ripple effect changes have, see if they have completed a requirement, and see if they have tested it properly. With traceability, and by managing changes effectively, managers gain the visibility to anticipate issues and ensure continuous quality.

Traceability also makes sure the product meets all the vital requirements that come from different stakeholders. By tracing requirements, all team members stay connected to each other and to all the interdependencies. And by managing changes well, a company can avoid scope creep, unplanned changes that occur when requirements are not clearly captured, understood and communicated. The benefit of good requirements is a clear understanding of the 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 right the first time means better quality, faster development cycles and higher customer satisfaction with the product. Concise, specific requirements can help companies detect and solve problems earlier rather than later, when they are much more expensive to fix.

Research has shown that project teams can eliminate 50 percent to 80 percent of project defects by effectively managing requirements. In addition, according to Borland Software, it can cost up to 100 times more to correct a defect later in the development process, after it's been coded, than when it's still in written form.

By integrating requirements management into their quality assurance process, companies can help teams increase efficiency and eliminate rework. According to Carnegie Mellon Software Engineering Institute, 60 percent to 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 they don't perform correctly the first time.

Conclusion

Requirements management can appear to be a complex topic; but at its core, it's a simple concept: It helps teams answer the question, "Does everyone—business analysts, product managers, project leaders, developers, and QA managers and testers—understand what they are building and why?"

When everyone is collaborating and has full context and visibility into the discussions, decisions and changes involved in product development, they maintain high quality and almost always ensure success.

About Jama Software

Jama Software is the product development platform for companies building complex, smart and connected products. The Jama solution enables enterprises to accelerate development time, mitigate risk, slash complexity and verify regulatory compliance. More than 600 product-centric organizations, including NASA, Thales and Caterpillar, use Jama Software to modernize their process for bringing complex products to market. For more information, visit www.jamasoftware.com.

