



Requirements definition as the foundation for effective software delivery.

Contents

2 Introduction

4 Understanding requirements definition

4 The challenges of requirements definition

6 The benefits of requirements definition

7 Developing the areas that can lead to success

15 Conclusion

Introduction

There are many challenges in requirements definition and management—all centered around the difficulty of getting requirements right, which means getting them complete and correct before the software product is released. Many studies have correlated requirements issues to the high rate of IT project failures. Some even indicate that approximately 60 to 70 percent of IT project failures occur as a direct result of poor requirements gathering, analysis and management. A lack of clearly defined and up-to-date requirements plays a major role in these failures.

When defining requirements, it's often difficult to develop unambiguous and thorough requirements before the product is shipped. Typically, this process is very informal. Even after the requirements information has been captured, it is often not updated to reflect changes that occur later in development. But for many organizations, improving the requirements definition process can help generate a better return on investment (ROI) than improving the requirements management process.

In this paper, we'll look at the challenges and benefits of requirements definition, and we'll describe how solutions like IBM Rational® Requirements Composer software can help you address the challenges to achieve the benefits.

Highlights

Released software can fail for many reasons, but more often than not, it's because code was written and never executed or requirements were poorly gathered, analyzed and managed.

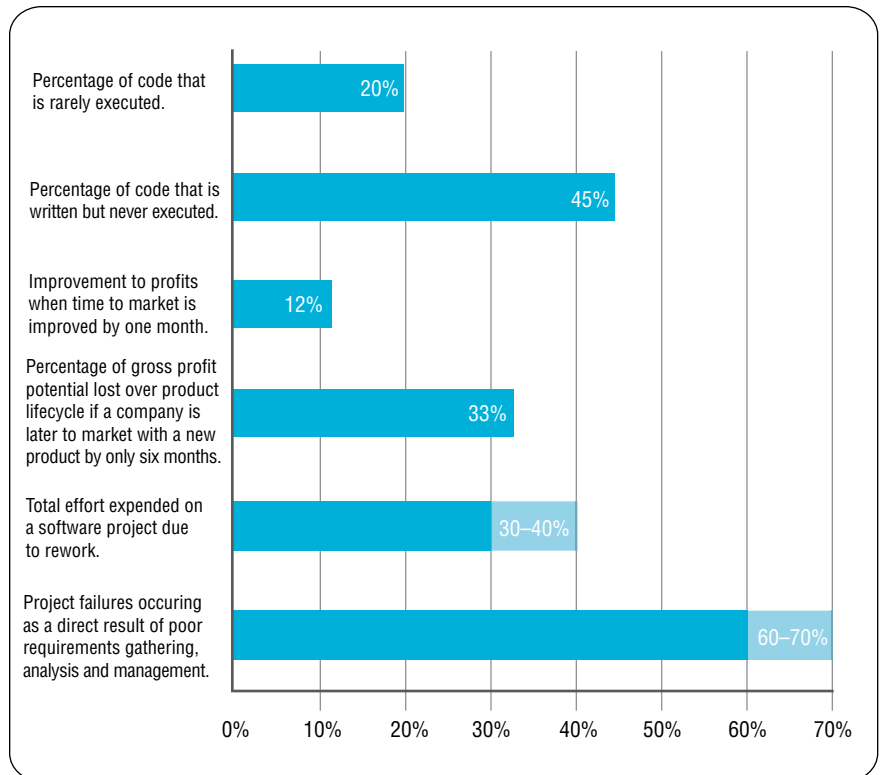


Figure 1: Numerous factors contribute to the high failure rate for released software. But the most significant factor is related to poor requirements gathering, analysis and management.

Highlights

Requirements definition involves eliciting, analyzing, specifying and validating stakeholder needs. Requirements management involves prioritizing, implementing and tracking those needs as they change throughout the project.

The biggest challenge to effective requirements definition is reaching consensus.

Understanding requirements definition

Requirements definition, also known as requirements development, is a set of activities that identify the needs that a project must satisfy as expressed by its many stakeholders. An iterative process, requirements definition enables the business to elicit and elaborate on problem statements. And it allows you to conceptualize possible solutions and their impacts, resulting in a set of requirements that will drive a project. In contrast, requirements management is the process of communicating and controlling the project scope, while incorporating the inevitable changes that occur during a project's duration. Requirements activities include gathering, storing, tracking, implementing and updating requirement priorities.

Critical to laying a firm foundation for any project is defining the right requirements. Given that there is little information about best practices in this area—and few supporting tools—businesses often end up developing incorrect and incomplete requirements. This dereliction frequently results in delays, wasted resources and stakeholders' dissatisfaction with project results.

The challenges of requirements definition

To realize the benefits of requirements definition—while at the same time mitigating the risks of trying to satisfy new and existing stakeholders—it's necessary to overcome challenges in how you collect, analyze and evaluate ideas for requirements. These challenges impact all artifacts and activities, affecting your information, communications, processes and tools. But the basic challenge is in collaboration: getting people to work together, ensuring that they have a common understanding of the project, and providing them with information on what they need to do and when they need to do it. This required coordination is difficult, as many of today's organizations have a global software supply chain that spans departments, organizations, geographies and time zones.

Highlights

The techniques you use to define requirements for a project of 10 people often don't work well when you have a project of 100 people.

When collecting requirements information, teams use all sorts of tools – such as documents, presentations and spreadsheets – which don't integrate well and provide no traceability.

Scalability of practices is another challenging area for requirements definition. People often take informal elicitation techniques and try to use them unsuccessfully on larger projects. What works on projects with two or three people doesn't work on projects with 10 people, and what works with 10 people doesn't work with 100. Additionally, a project where most people work remotely functions very differently from a project where everyone is in the same location. The degree of documentation and the need for tooling vary based on these and other factors.

Many organizations use Microsoft® Word documents, Microsoft PowerPoint slides or Microsoft Visio diagrams to define requirements and help explain concepts to stakeholders. Spreadsheets, like those developed in the Microsoft Excel application, are used to track requirements and their sources. This rudimentary approach may work for small projects, but it becomes increasingly challenging as you move to larger or multiple projects.

Integration of information is another limitation with focused tools like word processing, presentation and spreadsheet programs. They may allow for embedding and updating some information, but they do not capture relational data well, so visibility and context around linkages are quickly lost or forgotten. This results in a lack of traceability across deliverables, an inability to ensure solution coverage, and reuse of past work all but impossible. An additional consideration is that these specific use tools do not facilitate dialogue. Organizations using these manual approaches may create repositories for the artifacts or encourage the use of discussion boards or e-mail to address issues, but they're also creating disconnected information that needs to be collected and cataloged to provide context.

Highlights

By defining requirements thoroughly and accurately up front, you can reduce rework, speed time to market, increase productivity and improve software delivery.

The accuracy of your requirements greatly improves if you engage your stakeholders early and often.

The benefits of requirements definition

By thoroughly and accurately defining requirements, organizations can realize major benefits, including reduced rework and faster time to market, improved quality, increased productivity, and more effective software delivery.

Correctly defined requirements can first help organizations lower project costs through reducing rework, and therefore drive faster time to market. Research has found that rework can consume 30 to 40 percent of the total effort expended on a software project. On top of this cost savings from labor avoidance, there are revenue benefits to finishing a project a few months early, as it means there are that many more months for additional revenue, productivity gains and competitive advantage. Research also shows that delaying product introduction by only six months can reduce the gross profit potential by as much as one-third over the product lifecycle. In contrast, accelerating product introduction by as little as one month has the potential to improve profits by more than 10 percent.

Trusted and accurate requirements can also help improve quality. Requirements are often cited as one of the leading causes of software problems. When requirements are added or changed late in the development process, resulting issues can become increasingly expensive to fix. But when you define your requirements by engaging your stakeholders early and often, the accuracy and completeness of those requirements can significantly improve product quality.

Successful requirements definition also improves productivity, provides the foundation for effective software delivery and enables differentiation. Differentiation refers to providing capability that propels you ahead of your competition. Organizations are under increasing pressure to do more with less, so they must reconcile

Highlights

Rational Requirements Composer allows business clients to communicate their requirement needs to an analyst, who then defines requirements to meet those needs, verifies that they support the business objective, and then submits the defined requirements to the clients for feedback.

escalating system maintenance costs with the resources that continue to drive innovation. Successful organizations have found that helping people work smarter and be more productive within their budget constraints helps them optimize their tools and techniques, enabling them to make trade-offs to meet current and future business challenges.

Developing the areas that can lead to success

There are four areas that organizations need to develop to realize the rewards of well-defined requirements. And IBM Rational Requirements Composer can help you address them.

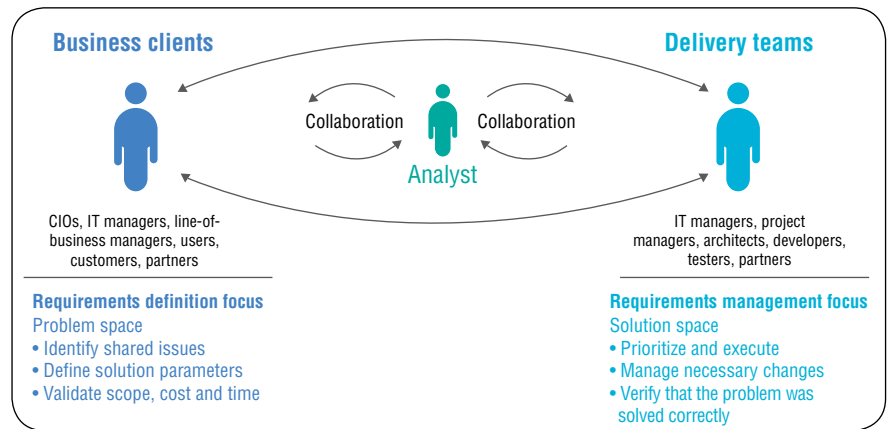


Figure 2: Aligning software requirements with stakeholders' business goals and objectives helps ensure project success.

Highlights

To define requirements as thoroughly as possible, organizations need to focus on improving collection techniques, collaborating for consensus, organizing information, and guiding the process.

Teams should be able to use the best techniques for expressing requirements—techniques that allow business and IT audiences to understand each other and contribute to requirements definition. When stakeholders can visually express requirements and teams can iterate on them quickly, organizations are poised for agile software delivery. To meet the challenge of distributed teams and various roles in an organization, collaboration and communication need to be formalized to help ensure that requirements are commonly understood, appropriately analyzed and validated. Additionally, a single repository should exist that allows information to be connected, tagged and grouped, helping to facilitate elaboration, traceability and reuse. Lastly, to help ensure repeatability, process guidance should be embedded into the fabric of the requirements definition solution.

Visualization techniques to break “language” barriers

Rational Requirements Composer provides several techniques for eliciting, defining and elaborating requirements, such as business process and use-case diagrams, user interface sketching and storyboarding, and rich-text editing to capture structured and unstructured information.

An immensely popular and very effective visualization technique, business process modeling is used heavily in business process management (BPM) and continuous process improvement programs. These diagrams capture task-level details to help business professionals see where issues are occurring within their current business operations, as well as evaluate the impact of changes on their processes. Process diagrams allow users to visually characterize issues, identify the root causes of those issues, and propose ways to solve them, thus allowing business processes to create business value. To help business analysts develop current and future process diagrams in a matter of minutes, Rational Requirements Composer has a business process editor that uses a subset of Business Process Modeling Notation (BPMN).

Highlights

Stakeholders use various tools to communicate their needs for requirements; Rational Requirements Composer can help integrate those tools.

With business process modeling, Rational Requirements Composer allows business analysts to model as-is (current) and to-be (future) business processes. In a simple and easy-to-learn technique, business process modeling supports a business-driven development strategy. Rational Requirements Composer enables business analysts and IT to trace from high-level business goals, business processes and business tasks to software requirements.

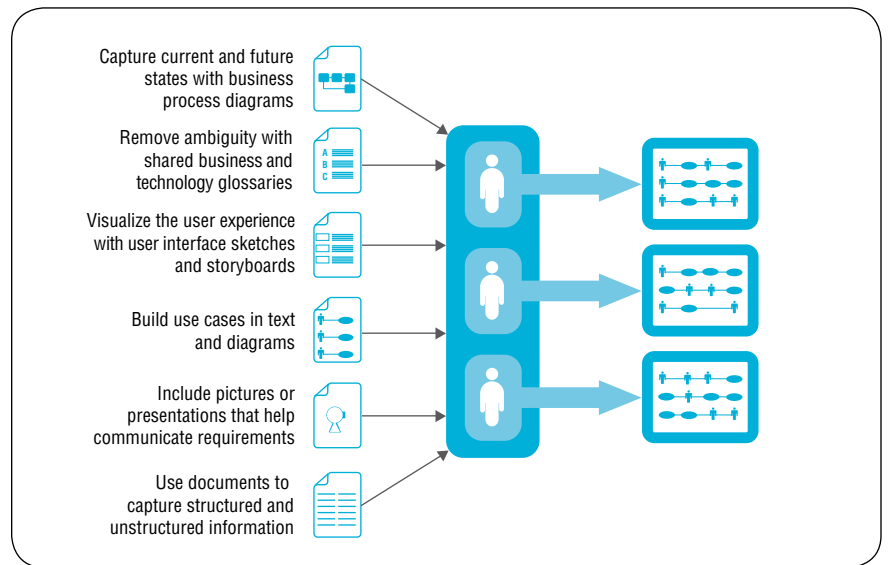


Figure 3: Requirements information can be presented in various ways. Rational Requirements Composer helps you capture and connect the web of information to define requirements.

Highlights

Designed to include sketching and storyboarding capabilities, Rational Requirements Composer allows users to communicate their requirements needs in the context of their preferred system interface.

Rational Requirements Composer also has a use-case diagram editor that allows you to create use-case models. Providing a good visual for describing and validating what a system will do, use-case diagrams—and use-case specifications—are valuable in discussions with stakeholders to describe the behavior of the system at a high level, helping stakeholders better understand what the system will do rather than how it will do it. Moreover, use-case diagrams allow designers to derive the relationships between participants, actors and activities, and they help drive out supplemental or nonfunctional requirements, such as performance or regulatory requirements.

In today's digital world, great focus has been placed on creating an enhanced customer experience based on optimizing the interactions between users and their preferred system interface. Rational Requirements Composer provides sketching and storyboarding capabilities to help define user needs. Sketches, or wireframes, are visual representations of the interfaces, or screens, that a user would interact with to perform a task. Storyboards are the coupling of sketches or wireframes in a specific sequence to create scenarios based on a series of tasks. Because they represent the way users work in applications and on the Web, sketches and storyboards have proved to be valuable in the definition, as well as the validation, of user requirements.

Highlights

Because Rational Requirements Composer allows you to integrate the tools that stakeholders use, you can move easily from one technique to another.

Rational Requirements Composer also has the flexibility of rich text, allowing users to create composite documents with text and embedded diagrams, images, hyperlinks, etc., that can capture the richness of project requirements data to create easy-to-understand specification and sign-off documents. Rich-text documents in Rational Requirements Composer can also link to terms you define in the software's glossaries. When defining terms, you can also specify related descriptions, acronyms and synonyms.

Since all of these techniques are designed to work together, you can quickly and easily move from one technique to another. You can form relationships between the artifacts created with these techniques, and you can use Rational Requirements Composer features with large and small teams, whether they are colocated or geographically distributed.

Facilitating collaboration and communication

Requirements definition and management is a team sport. The ability to create a team environment—where roles, goals, activities and measures of success are known—increases the possibility that the team can compete effectively. By promoting dialogue, wherever teams may be, you can help ensure that various perspectives are evaluated and understood when defining requirements.

Highlights

Built on the IBM Jazz platform, Rational Requirements Composer enables users to collaborate on requirements definition no matter where they are.

It's good to see what others are working on and to comment on their work. It's even better to maintain a history of comments and work artifacts. Wikis and other self-regulated content management tools have become very popular in getting communities to create shared content. Discussion threads and chats can bind participants to specific topics and keep them on the same page. This approach also creates a record of the "conversation" for later use.

The amount of requirements information can be overwhelming for stakeholders. Governing the level of participation through access controls allows the roles of authors, readers, contributors and reviewers to be defined and managed in the areas where those stakeholders interact with the project, helping to ensure that information is consumable and not overwhelming, alarming or inadvertently changed.

Rational Requirements Composer capabilities leverage the open and extensible architecture provided by the IBM Jazz™ solution, a next-generation platform for collaborative and productive software delivery. Helping to transform how people work together to develop and deliver software, the Jazz platform provides the underlying technology that allows Rational Requirements Composer to help you define project workspaces for teams, build shared glossaries for terminology, create comments about work items and artifacts at various levels of granularity, develop conversation threads with a person or a group, and see the history of responses. Transparency and open dialogue allow teams to focus on business needs, identify conflicts earlier and validate requirements more quickly.

Highlights

You can categorize, organize, search and find requirements easily using Rational Requirements Composer tagging, filtering and searching capabilities.

But collaboration does not end with the definition of requirements. As requirements evolve and change, you can use the robust IBM Rational RequisitePro[®] tool to help your global development team create requirement baselines, as well as trace and integrate requirements to support better design, code and test cases.

Organizing complex relationships between “things”

The sheer volume of requirements data makes managing, searching and using it difficult. When you also have to accommodate both structured and unstructured data such as rich text, images, use-case diagrams and discussion threads, the task seems almost impossible. The ability to see the larger picture through artifact links, associated metadata, discussion threads and artifact-level comments provides a cohesive story that can be understood and used to join the conversations.

Rational Requirements Composer has the facility to organize and retrieve information quickly. Using the software’s tagging, filtering and searching capabilities, you can categorize, organize, search and find requirement artifacts easily, helping to improve your requirements management capabilities.

To facilitate tracing among requirements data, you can create links in Rational Requirements Composer from one element or artifact to another. This navigational ability provides rich context around requirements and can greatly simplify the transfer of knowledge from one stakeholder to another.

Additionally, Rational Requirements Composer has project and user dashboards where you can see recent comments, requirements, artifacts and links from other team members. This helps keep your distributed team on the same page.

Highlights

Embedded practices in Rational Requirements Composer can help you overcome common requirements issues.

Using Rational Requirements Composer, you can author your own practices to enhance your requirements definition process.

Providing process guidance

When organizations choose a requirements definition solution, they also need a tool that supports the practices they use. Using Rational Requirements Composer to define requirements, you can leverage the software's embedded practices for overcoming common requirements issues. And you can make these practices configurable with IBM Rational Method Composer, whose process library contains out-of-the-box practices as well as tools to create your own practices. You can also take advantage of IBM's path for extending capabilities through simpler adoption of new notation styles and techniques, and IBM can show you how to bring these notations together. For example, for those who come from a use-case heritage, the emphasis is on text. But this approach can be supplemented with visual techniques like use-case diagrams, storyboarding and business process diagrams.

Rational Requirements Composer provides contextual process guidance as you move from one editor to another using the various features within the tool. The practices are based on the IBM Rational Unified Process® (RUP®) methodology and the Open Unified Process (OpenUP) framework. Through Rational Method Composer, you get authoring capabilities and access to a library of processes that can enhance your usage of Rational Requirements Composer, including process guidance from IBM and OpenUP, or you can author your own set of best practices.

Highlights

Rational Requirements Composer offers various requirements definition techniques, organizational capabilities and process guidance on a collaborative platform to help you effectively deliver better software.

Conclusion

The various requirements definition techniques, collaboration features, organizational capabilities and process guidance capabilities found in Rational Requirements Composer are designed to help you address the areas of development that can most dramatically affect accurate requirements definition. By directly addressing these areas, you can align business goals and objectives with IT, reduce rework and speed time to market, increase reuse and productivity, and help ensure that project requirements are of high enough quality to lead directly to project success and effective software delivery.

For more information

To learn more about IBM Rational Requirements Composer software, contact your IBM representative or IBM Business Partner, or visit:

ibm.com/software/awdtools/rrc

To learn more about the IBM approach to defining accurate and complete requirements, visit:

ibm.com/software/rational/offerings/irm



© Copyright IBM Corporation 2008

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
December 2008
All Rights Reserved

IBM, the IBM logo, ibm.com, and Rational are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml

Microsoft is a trademark of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates. The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software.