

How to Reuse Requirements: 7 Tips for Reusability Best Practices

Many safety-critical industries reuse requirements as a way of saving time, however, if this is done incorrectly or without proper steps taken to ensure the reused requirements are still accurate and relevant to the current project, critical errors can occur which cost more than itwould to simply author the requirements from scratch. This guide covers best practices on how to reuse requirements properly to avoid errors and miscommunications while saving valuable time and effort.

Requirements reusability is the process of reusing is in most cases, exactly the same. Since this phase requirements that have already been authored and requires a significant amount of time and effort, reusimplemented before in previous projects. This tech- ability makes perfect sense as the idea is to reduce nique is used by requirements engineers to ensure time and effort and achieve consistency. that maximum productivity and consistency are achieved throughout the product development lifecycle. When it comes to industry-specific compliance and standards, the information gathered and used,

WHAT CAN GO WRONG?

Requirements reusability is an effective method for improving efficiency in project development teams, but it must be noted that for these projects to be successful, proper planning must first occur.

Without the use of a framework and a culture of reusability within a team, the reuse of requirements can cause significant errors. For example:

- If not done correctly, there is the risk of blindly copying or linking requirements that may be outdated or have unmatched dependencies.
- If the requirements are tightly coupled with the product itself, reusability can cause confusion and inconsistency.
- Implementing the process without having the right tools and frameworks can consume more time than the amount intended to save with reuse.
- Requirements usually present alongside additional data, including rationales, design elements, data definitions, etc. so maintaining the context is crucial.

It's important to keep in mind that the reuse of requirements is not something that can be implemented instantaneously. To yield the full benefit of the increased productivity and decreased cost that is associated with this practice, a solid framework must first be created and enforced.

WHY SHOULD REQUIREMENTS BE REUSED?

Reusing requirements is a process to save time gives companies a competitive advantage as the final and effort, but its major benefit comes in the form requirements documents are standardized, and there of requirement standardization as reusing require- is a reduced chance of forgetting the core features ments helps create a repository of previously imple- and standards required for the end product. mented and tested requirements. Reuse also assists in creating an improved user experience and better consistency in the project development process. It





Requirements Reusability has its own set of best 2. CREATE A STANDARD FOR AUTHORING practices, which can streamline this process and make it easier to introduce and establish within project development teams. The main objectives are to avoid writing requirements from scratch, better utilization of available resources and knowledge, and to save time.

1. USE A REQUIREMENTS MANAGEMENT TOOL

Attempting to reuse requirements without the use of a smart tool can lead to unseen errors and lost productivity. It's recommended to use a tool that can help manage, maintain and reuse previously written requirements. From creating a categorized set of reusable requirements to tracking the changes made via change control capabilities, a requirements management tool can help increase the efficient use of time. It can also help with creating a standardized process for requirements engineering in addition to establishing reusability.

A good requirements management tool will have features that can support the reusability process and make it easier to manage and maintain. When deciding on a tool for your team, look for these features:

- Requirements traceability throughout the project development cycles
- Customizable requirement attributes
- Hierarchical organization of requirements and related data
- Ease of working collaboratively within the tool
- Ability to identify similarities and redundancies in the project repository

Other factors such as cost, ease of implementation and company compatibility are also important aspects to consider but do not directly impact the reusability of your requirements.

REQUIREMENTS

Creating a standardized language set for documenting requirements is imperative to not only the success of your project but also to ensure future requirements projects follow a similar format. This will help maintain a level of similarity between requirements, preventing discrepancy when reviewing. The following rules of thumb can be followed to ensure a standardized language set for requirements writing:

- Create an authoring standard, which includes using a proper syntax for both functional and non-functional requirements
- Create a standard for documenting business models and business rules
- Create a custom template of relevant attributes, including data definitions, standards, rationale, design elements, etc
- Use standard units, terms, and definitions to maintain consistency
- Use standard templates for documenting user personas, roles and privileges, etc
- Avoid ambiguity and vagueness to ensure clarity and quality

Much of this can be automated using QRA Corp's requirements analysis tool, QVscribe. Harnessing Natural Language Processing, QVscribe automatically checks for best practice compliance such as INCOSE and ensures there are no redundancies or ambiguity in the requirements.

3. WRITE GOOD REQUIREMENTS

Nothing beats this. Period. Authoring clear a concise requirements ensures that engineering tear reap the maximum benefits from requirements reu ability. Good requirements have no ambiguities an are supplemented by rationale, data definitions, an acceptance criteria. Every requirement should ha a unique identifier and a flag for marking its priori It is important to use a predetermined level of det in a requirement, adding product specific details details about implementation can add confusion if the requirement gets reused for another proje without reconsidering contextual information.

For tips on how to write a requirements docume that conforms to these specifications, check o our ultimate guide to Writing an Exceptionally Cle Requirements Document.

4. BE MINDFUL OF INTELLECTUAL **PROPERTY RIGHTS**

Reusing requirements can cause Intellectual Property rights violations if you are reusing something that is not yours by law or available free for public use. When creating a process, ensure that requirements that came from your users aren't part of your repository of reusable requirements.

5. CAREFULLY SELECT WHAT TO REUSE

Not all requirements are fit to be reused. When choosing which requirements to reuse, consider if they may change over time, or if they are standardized to be relevant to a number of projects.

	Here are a few examples of requirements of	
С		tegories that may be eligible for reuse:
nd		
ns	•	Non-functional requirements
ls-		(performance criteria, page load time, etc.)
nd	•	Functional Requirements
nd	•	User classes and personas
ve	•	Business models and rules
ty.	•	Security requirements
ail	•	Compliance and standards
or		(safety requirements, building codes, etc.)
on	•	Glossary and data definitions
ect	•	Accessibility requirements
		(web content accessibility guidelines, etc.)
	•	Symmetrical user functions
nt		(create, read, update, delete and search, etc.)
ut	•	Core functions (signup, sign in and recover pass-
ear		word, device reset, etc.)
	•	Document templates (requirement
		specifications, design documents, etc.)
	•	Information sources on domain knowledge

Requirements engineers can create reuse patterns to ensure that only relevant and useful requirements are being copied or linked to newly conceived project ideas.

6. MAINTAIN THE REPOSITORY

Once you have a categorized set of reusable requirements, the next step is maintenance. Regardless of the requirements style or context, the repository should be kept up-to-date at all times. This is done to reduce the risk of reusing obsolete information and to take note of all changes being made. Make sure to update requirements as new information or data is available, this ensures reusable requirements longevity.

7. DO A REUSABILITY RETROSPECTIVE

Another best practice is to host retrospective meetings periodically to ensure that the requirements reusability process is up to date. Collecting feedback and lessons learned should be part of the process, with a way to track which pointers are being implemented to improve the process. Do this often to keep track of problems and work on solutions collaboratively. Keep a backlog of ideas/changes to be tested/implemented in the process and review in every retrospective meeting; this will reset the sail towards an exponentially improved requirements reusability process.



CONCLUSION

Requirements Engineering is time-intensive, and much of a project's success depends on getting it right. Reusing requirements is a way to save time and to ensure the quality of requirements as they have been written, implemented and tested before. If done hastily or without care, reusing requirements can cause waste, so project engineers must streamline the process by following its best practices and creating a culture that values smart work and reuse of available resources and knowledge.

in 🍠

To learn more about QVscribe, visit <u>qracorp.com/qvscribe</u>

qracorp.com



