

# Validation Report

Friday, April 29, 2022 12:03 AM

Requirements validation is concerned with having the right requirements for the perspective software. Validation is important and without it requirements may not encompass the entirety of the software or they may lead the development in the wrong path. Validation seeks to find errors, lack of clarity, mistaken assumptions, or requirements that don't fit the project plan.

To validate requirements there are a number of techniques. The traditional technique is to have a requirements review, where a group will look at the requirements and determine if changes need to be made based on clarity, completeness, and correctness. Prototypes are another technique that provides the customer with a visual understanding of what the development team thinks the project will be, and allows the customer to point out aspects that they may have not thought of or that the development team has misinterpreted. Acceptance tests are also used as a form of quantitative validation, and each requirement will need to satisfy its acceptance test to be deemed a valid requirement for the project. Models are also used as a visual aid in validation and can help give a better overall understanding of the system and how it is planned to interact with itself and external entities. Model validation provides the same type of benefits as prototyping in that it puts the relevant information in an easy-to-understand form that can easily be picked apart and analyzed.

Depending on the model of development, validation can occur at the end of the requirements specification phase or can occur throughout development. Because validation is used to ensure the requirements are the right requirements, if new requirements are added or current ones need to change, those much be validated too.

The software engineering process supports requirements validation by requiring it to be done before moving forward with actual coding. The first step to starting a software project is to gather all important information about the proposed software and then write up the requirements for it. This initial set of requirements needs to be validated before beginning because in a model like a waterfall once you finish a phrase you don't go back, or in agile projects requirements can change throughout the project and to avoid any mistakes further down development it's vital to validate all requirements before they are used for anything else.

My requirements review could have been done better, but due to circumstances with my sponsor, it had to be done in a less-direct way. We couldn't sit down and have a conversation so I sent the set of requirements to my sponsor and had him review them for any inconsistencies, mistakes, or lack of clarity. I also sent my sponsor my prototype to ensure I had the right idea about the project, and I feel this provided more for me in the terms of review than sending my list. My sponsor pointed out a few things that I had misunderstood or gotten wrong and I was able to use that information to make some changes to my requirements and fix my prototype. I also talked with a classmate about my requirements and had them check for the same things I asked of my sponsor, however with the same limited knowledge of requirements engineering as me the conversation didn't provide very valuable information.

For this project, I was working with a company whose patrons are members of the disabled community. Because of this, the requirements and business rules needed to reflect government protections on their information and usage of the software. The business rules I wrote up included these protections such as not storing any protected health information (PHI) or personally identifiable information (PII). The software is a source for downloading and accessing thousands of accessibility-friendly resources so public health, safety, and welfare were all covered by the business rules protecting the PHI and PII of the patrons. The software is not accessible to non-U.S citizens, so cultural or societal standards of places outside of the U.S need not be addressed, and the software does not have any effect or influence on the environment or economy due to it being an online website to access digital copies of resources.