Validation Report

General Validation

The validation of requirements is necessary to ensure that developers have a correct understanding of the requirements needed to develop the system or software. Validating requirements also aids in ensuring that software requirements comply and coincide with business requirements, business rules, and business requirements. Validation is vital to ensuring that requirements are created properly and are within the scope of the project and necessary to satisfy the goals of the project.

The requirements should be validated a few times during the requirements process of a project. The main goal of validation is to ensure that any flaws or issues with the requirements should be found before the development process has gone too far towards completion. If these flaws are found close to when the software should be completed, it incurs more cost allocated to the project as developers have to spend more time fixing the issues that were already implemented. It is important throughout a project to continuously ensure that the requirements are specifying the right software that is being built.

There are many different ways of validating requirements. A few of these methods are expanded upon below.

Reviews and Inspections

Reviews and Inspections are one of the basic method of validating the requirements. A group of individuals is responsible for looking over the software requirements specification document for any errors, ambiguity, or any deviation from the goal of the software. This group of individuals should include at least one individual who represents the user of the software.

Reviews can also be done by other individuals like other members of the development team, or other stakeholders. This ensures that requirements are validated by multiple individuals who are involved in the product and its production. It also reduces the probability that mistakes will be made since multiple people will be looking at the same document. From the various feedback that is gathered, the requirements can be revised to ensure that the right software is being built.

Prototyping

Prototyping can be a helpful method of validating requirements. Creating prototypes and validating them with stakeholders is a straightforward method in ensuring that the two parties agree on what is expected in the software. Since they are visual, the two parties have an easier time explaining and conveying their viewpoint of how the software should look like. There are different types of prototyping that vary in the amount of detail that is present. Prototyping can at times be costly to create as the process can easily divert to creating a perfect prototype rather than focusing on the creation of the software. Prototyping can also cause stakeholders to get too focused on the aesthetics of the system rather than its functionality when presented with a prototype. If prototypes are used wisely and preferably early in the requirements process, they can be very helpful when validating requirements.

Acceptance Test Design

Acceptance tests can be used to verify requirements, therefore designing acceptance tests is vital in ensuring that all requirements are validated and verified.

Alphas & Betas planned

An alpha test is conducted with an early prototype where many changes are expected to be made. The prototype is released to a small number of individuals and is used to validate the requirements. By allowing other individuals to test the software prototype we can get feedback as to whether the system is doing what it should. In other words, we are ensuring that the system is satisfying its main purpose, at a very broad level.

Beta testing is quite the opposite in terms of detail. Beta testing is conducted when the product is near completion. There are few changes that are expected to be made but some bugs may exist. The purpose of beta testing is the same as alpha testing but can be a bit more detailed since the software is near completion.

The feedback from either or both of these tests can constitute an alteration or review of the requirements of the software. If the system is not doing what it is supposed to, then developers need to reassess the requirements to correct this error.

Operational Acceptance Testing Plan

Operational Acceptance Testing is conducted to assess and consider the environmental factors that interact with the system. This is conducted before release to ensure that these behaviors are considered before release the process. Operational testing helps to identify how the software will interact and be influenced by the proposed environment of the software. It is important to conduct operational acceptance testing to ensure that the new product can and will function correctly with existing products or practices.

Validation of Behavioral Requirements

The requirements engineering process supports the validation of behavioral requirements as many of the tests and the different methods of validation require users to be involved. Behavioral requirements can be defined as the how the user will interact with the system. Since the requirements are not only being validated by developers and other stakeholders, but also by users, the validation process gets a good perspective of the behavioral attributes of the system. At the core of each method of validating the requirements of the software, users are vital to ensure that the software is being built to do the right thing.

Requirements Review Memo for project on following page.

Requirements Review Memo

The requirements validation of the site for Ashlynn Glitz & Glitter Custom Boutique was conducted by a developer, customer, and the product owner. During this review and inspection, the requirements of the site were reviewed to ensure that the site is proposed to do the right thing in accordance with business rules, requirements, and specifications that are detailed in the BRS document.

The review reintroduced and solidified the goal of making the site easy to use, secure, and "fast" (See the elicitation report for more detail on the feedback). It is important to note that "fast" is an ambiguous term and should be expanded upon at a later date to ensure that the requirement can be quantified. Minor logical errors appeared in the requirements review that were quickly fixed. Some clarity and discussion were made regarding some of the implementation of the requirements. For example, requirement #18 states that "The e-commerce website shall allow customers to review previous orders that have been processed within 10 minutes." This aspect of the system was explained to product owner to note that there is some processing involved that could be affected by the number of users on the site and server connections. This was also noted to be the maximum time not the average.

To revisit the topic of some of the non-functional goals of the software, it is important to note that these need to be quantifiable in order to be tested against the system. It is also vital to note that there is a level of speed or security that is not within the scope or budget of the site. It is not entirely necessary to have the most secure website on the web if the business is relatively small and doesn't deal with a lot of users. On the other hand, some level of security is needed since the site deals with personal information such as payment details and addresses.

Another important method of validation that is to be noted is the prototype that is in the process of being created. At this stage, I am creating a clickable interactive prototype or a beta version of the site that will be reviewed and evaluated by the product owner and a few users of the site. The results from this evaluation will be accounted for in the next iteration of the cycle and the specifications will be updated to ensure the correct product is being developed. To note, a medium fidelity wireframe was created and presented to the product owner for evaluation. This was early in the process, and I do not expect the next prototype to look the same as the wireframe.

Some of the needs of the business have already been met. For example, business rule #5 that states "Due to the customized nature of the products, all sales are final, meaning no returns or exchanges" is automatically implemented into the system as there is no functionality that will be implemented to allow for returns. The specification of the requirements also allows for an ease of submitting payments as many payment options are to be available on the site. Other considerations such as health, are met through other business operations that are unrelated to the operations of the site/software.

Going forward, the site should conduct an operational acceptance test to ensure that the site will work with the existing technologies that are currently being used. In the BRS it is noted that there is already a site that is in use and functioning. It is important that the site can be replaced accordingly with minor or little difficulty. In other words, the replacement of the site should not incur more cost for the product owner, but this should be verified in the operational acceptance testing process and analysis.