Software quality assurance: requirements, standards and resources

Alexey A. Gurkin, Broward College

ABSTRACT

Testing is an important part of the software development life cycle. It is the process that helps to identify defects in software and fix them. The main objective of testing is to ensure that the software meets the requirement specified by the customer and it works as per design. Software testing also helps in improving quality of product and detecting errors at an early stage. In this blog post, we will discuss different types of software testing and their role in different stages of software development life cycle.

INTRODUCTION

The different software development cycles require different types of testing. In some cases, multiple tests may need to be conducted during different stages of the cycle. It is important to have a plan in place so that all the necessary tests are conducted and the product meets all quality standards. Once the product is released, it is also important to establish a good customer relationship so that the customers are satisfied and continue to use the product.

DIFFERENT SOFTWARE DEVELOPMENT LIFE CYCLES ARE:

- Waterfall model: In this model, testing is done after the completion of each phase of software development. All the requirements are gathered at the beginning of the cycle and then coding is done followed by testing. This model is suitable for small projects where requirements are well known in advance.

- Agile model: In agile model, testing is done throughout the software development process. Requirements are divided into small modules and each module is developed and tested separately. This model is suitable for large projects where requirements are not known in advance.

- Iterative model: In this model, the software development process is divided into small cycles or iterations. Each iteration consists of development and testing phases. This model is suitable for projects where requirements are not known in advance and change frequently.

Software testing plays an important role in all the above mentioned software development life cycles. It helps to ensure that the software meets the requirements specified by the customer and it works as per design. Software testing also helps in improving quality of product and detecting errors at an early stage.

DIFFERENT TYPES OF SOFTWARE TESTING ARE:

- Functional testing: This type of testing is done to ensure that the software functions as per requirement specification.

- Non-functional testing: This type of testing is done to ensure that the software meets all the non-functional requirements such as performance, security, usability, etc.

- Unit testing: This type of testing is done to test individual units or components of the software.

- Integration testing: This type of testing is done to test how different modules of the software interact with each other.

- System testing: This type of testing is done to test the complete system as a whole.

Software testing is a very important part of the software development process. It helps to ensure that the software meets the requirements specified by the customer and it works as per design. Software testing also helps in improving quality of product and detecting errors at an early stage. Different types of software testing are functional testing, non-functional testing, unit testing, integration testing and system testing. Each type of testing has its own importance and role in different stages of the software development life cycle.

Functional testing is a type of black box testing which is done to ensure that the software functions as per requirement specification. In functional testing, test cases are designed based on requirement specification. Functional testing is done to verify that all the features of the software work as expected.

Non-functional testing is a type of black box testing which is done to ensure that the software meets all the non-functional requirements such as performance, security, usability, etc. Non-functional testing is done to verify that the software meets all the required non-functional properties.

Unit testing is a type of white box testing which is done to test individual units or components of the software. In unit testing, test cases are designed based on code structure. Unit testing is done to verify that each unit of the software works as expected.

Integration testing is a type of white box testing which is done to test how different modules of the software interact with each other. In integration testing, test cases are designed based on code structure. Integration testing is done to verify that the interface between different modules works as expected.

System testing is a type of black box testing which is done to test the complete system as a whole. In system testing, test cases are designed based on requirement specification.

System testing is done to verify that the complete system meets all the requirements.

REQUIREMENT OF SYSTEMS.

The first step in any software development cycle is to understand the requirements of the system. This understanding is essential for all subsequent steps, including design, development, testing, and deployment. Without a clear understanding of the requirements, it is impossible to create a quality software product.

There are many different ways to gather requirements for a software system. The most common method is to use interviews and focus groups. However, other methods, such as surveys and questionnaires, can also be used. Once the requirements have been gathered, they must be analyzed and organized so that they can be used in the next stage of the software development cycle.

PRODUCT & STANDARDS.

After the requirements have been gathered and analyzed, the next step is to standardize the product. This standardization ensures that all products meet the same basic requirements. It also allows for easy comparison between products.

One of the most important aspects of product standardization is creating a set of specifications. Specifications detail what the product must do and how it must do it. They are an essential part of the development process, as they provide a roadmap for the entire team.

Creating a set of specifications can be a challenging task. They must be comprehensive enough to cover all the requirements of the product, but not so detailed that they are impossible to implement. It is also important to ensure that the specifications are achievable and realistic.

TESTING AND CALIBRATION.

Once the product has been standardized, the next step is to test it. Testing is essential to ensure that the product meets all the requirements set forth in the specifications. It also allows for any bugs or errors to be found and corrected before the product is released.

Calibration is another important part of the testing process. This ensures that the product is able to function correctly under a variety of conditions. Calibration is often done using specialized equipment, such as temperature chambers and humidity controllers.

QA TROUBLESHOOTING.

Even with thorough testing, there may be some problems that only become apparent after the product is released. That is why it is important to have a troubleshooting plan in place. This plan should detail how to identify and correct any problems that may arise.

DESIGN AND DEVELOPMENT.

After the product has been tested and calibrated, the next step is to design and develop it. The design phase is when the product is created, and the development phase is when the product is perfected.

During the design phase, the team will create a prototype of the product. This prototype will be used to test the feasibility of the design and to identify any potential problems. Once the prototype has been approved, the development phase can begin.

During the development phase, the team will work to improve the quality of the product. They will also add any features that were not included in the original design. Once the product is complete, it will be ready for release.

PROCESS CONTROL & INSTRUMENTATION AUTOMATION FOR QUALITY.

After the product has been designed and developed, the next step is to implement process control and instrumentation automation for quality. This step ensures that the product is manufactured correctly and meets all the quality standards.

To do this, the team will create a set of process control charts. These charts detail all the steps in the manufacturing process and identify any areas where there is room for improvement. The team will also create a set of instrumentation automation scripts. These scripts will be used to test the product during production.

STATISTICAL QUALITY CONTROL.

After the product has been manufactured, the next step is to implement statistical quality control. This step ensures that the product meets all the quality standards set forth in the specifications.

To do this, the team will take a sample of the products and test them for defects. They will also compare the results of these tests to the quality standards. If the product meets all the standards, it will be released. If not, the team will work to improve the quality of the product.

PRINTING AND CONVERSION.

After the product has been manufactured, the next step is to print and convert it. This step ensures that the product is printed correctly and meets all the quality standards.

To do this, the team will create a set of printing and conversion templates. These templates will be used to test the product during production. The team will also create a set of inspection scripts. These scripts will be used to inspect the product during production.

CUSTOMER RELATIONSHIP.

After the product has been released, the next step is to establish a customer relationship. This step ensures that the customer is satisfied

with the product and that they will continue to use it.

QA SOFTWARE.

QA software is vital for any business that wants to ensure the quality of their products. Here are the top ten QA software programs that can help you test your products and ensure they meet the highest standards.

Jira is a popular tool for quality assurance, used by many large companies. It has a wide range of features and plugins to make it customizable for your needs. You can create tickets for each issue you find, track progress, and assign team members to work on specific tasks.

TestRail is another great tool for managing your QA testing process. It helps you organize tests into suites, track results, and generate reports. This makes it easy to see where improvements need to be made and identify areas of regression.

Zephyr is a tool that allows you to create, execute, and manage your test cases. It integrates with Jira so you can easily track your testing progress. Zephyr also offers a course on quality assurance to help you get started with using the tool.

qTest is a tool that helps you automate your testing process. It has features for managing test cases, executing tests, and generating reports. qTest also integrates with popular tools like Jira and Jenkins.

Xray is a plugin for Jira that adds test management capabilities. With Xray, you can create and execute test plans, track defects, and generate reports. Xray also offers a free trial so you can try it out before deciding if it's the right tool for you.

SoapUI is a tool that allows you to test web services. It supports all major protocols and has a wide range of features for creating, executing, and reporting on tests. SoapUI also has a free version so you can try it out before deciding if it's the right tool for you.

Postman is a tool that allows you to test API calls. It has a rich feature set for creating, executing, and documenting tests. Postman also has a free version so you can try it out before deciding if it's the right tool for you.

Fiddler is a proxy server that allows you to intercept and inspect traffic. This makes it useful for debugging web applications. Fiddler also has a free version so you can try it out before deciding if it's the right tool for you.

Wireshark is a network protocol analyzer that allows you to capture and inspect traffic. This makes it useful for troubleshooting networking issues. Wireshark also has a free version so you can try it out before deciding if it's the right tool for you.

To do this, the team will create a set of customer service scripts. These scripts will be used to help the customers use the product and to troubleshoot any problems they may have. The team will also create a set of marketing materials. These materials will be used to promote the product to new customers.

CITATIONS

1. Quality Assurance in Industries: A Comprehensive Guide to Quality Assurance in Industries. (2021). (n.p.): Notion Press.

2. Laporte, C. Y., April, A. (2018). Software Quality Assurance. United Kingdom: Wiley.

3. Naik, K., Tripathy, P. (2011). Software Testing and Quality Assurance: Theory and Practice. Germany: Wiley.

4. Quality Assurance in Industries: A Comprehensive Guide to Quality Assurance in Industries. (2021). (n.p.): Notion Press.

5. Donabedian, A. (2002). An Introduction to Quality Assurance in Health Care. United Kingdom: Oxford University Press.

6. Chopra, R. (2018). Software Quality Assurance: A Self-Teaching Introduction. United States: Mercury Learning & Information.

7. Sao, R. B. (2016). Perfect: Quality Assurance & Quality Control. (n.p.): CreateSpace Independent Publishing Platform.

8. Wellens, P. (2015). Practical Web Development. United Kingdom: Packt Publishing.

9. Krause, J. (2016). Introducing Web Development. United States: Apress.

10. Engström, J. (2021). Web Development with Blazor: A Hands-on Guide for .NET Developers to Build Interactive UIs with C#. United Kingdom: Packt Publishing.

11. Grinberg, M. (2018). Flask Web Development: Developing Web Applications with Python. United States: O'Reilly Media.

12. Fain, Y., Gamov, V., Rasputnis, V., Tartakovsky, A. (2014). Enterprise Web Development: Building HTML5 Applications: From Desktop to Mobile. United States: O'Reilly Media.