|  |  |
| --- | --- |
|  | QA Internship program 2019 |
|  |  |
|  | Lia MOCANAS  QA mentor  4/22/19 |

Contents

[What is Software Testing? 2](#_Toc6569773)

[Manual testing 2](#_Toc6569774)

[Automation testing 3](#_Toc6569775)

[Why Automated Testing? 3](#_Toc6569776)

[Which Test Cases to Automate? 3](#_Toc6569777)

[Types of Software testing 4](#_Toc6569778)

[Test documentation 4](#_Toc6569779)

[Defect management 4](#_Toc6569780)

[Testing an eCommerce website 6](#_Toc6569781)

# What is Software Testing?

Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is Defect free. It involves execution of a software component or system component to evaluate one or more properties of interest.

Software testing also helps to identify errors, gaps or missing requirements in contrary to the actual requirements. It can be either done manually or using automated tools. Some prefer saying Software testing as a White Box and Black Box Testing.

In simple terms, Software Testing means Verification of Application Under Test (AUT).

Links: [What is Software Testing? Introduction, Definition, Basics & Types](https://www.guru99.com/software-testing-introduction-importance.html)

[Software testing](https://en.wikipedia.org/wiki/Software_testing)

# Manual testing

Manual Testing is a type of Software Testing where Testers manually execute test cases without using any automation tools. Manual Testing is the most primitive of all testing types and helps find bugs in the software system.

Any new application must be manually tested before its testing can be automated. Manual Testing requires more effort but is necessary to check automation feasibility.

Manual Testing does not require knowledge of any testing tool.

One of the Software Testing Fundamental is "**100% Automation is not possible**".

Links: [Manual Testing Tutorial for Beginners: Concepts, Types, Tool](https://www.guru99.com/manual-testing.html)

# Automation testing

Manual Testing is performed by a human sitting in front of a computer carefully executing the test steps.

Automation Testing means using an automation tool to execute your test case suite.

The automation software can also enter test data into the System Under Test, compare expected and actual results and generate detailed test reports. Test Automation demands considerable investments of money and resources.

Successive development cycles will require execution of same test suite repeatedly. Using a test automation tool, it's possible to record this test suite and re-play it as required. Once the test suite is automated, no human intervention is required. This improved ROI of Test Automation. The goal of Automation is to reduce the number of test cases to be run manually and not to eliminate Manual Testing altogether.

## Why Automated Testing?

Automated software testing is important due to the following reasons:

* Manual Testing of all workflows, all fields, all negative scenarios is time and money consuming
* It is difficult to test for multilingual sites manually
* Automation does not require Human intervention. You can run automated test unattended (overnight)
* Automation increases the speed of test execution
* Automation helps increase Test Coverage
* Manual Testing can become boring and hence error-prone.

## Which Test Cases to Automate?

Test cases to be automated can be selected using the following criterion to increase the automation ROI

* High Risk - Business Critical test cases
* Test cases that are repeatedly executed
* Test Cases that are very tedious or difficult to perform manually
* Test Cases which are time-consuming

The following category of test cases are not suitable for automation:

* Test Cases that are newly designed and not executed manually at least once
* Test Cases for which the requirements are frequently changing
* Test cases which are executed on an ad-hoc basis.

# Types of Software testing

We, as testers are aware of the various types of Software Testing such as Functional Testing, Non-Functional Testing, Automation Testing, Agile Testing, and their sub-types etc.

Each of us would have come across several types of testing in our testing journey. We might have heard some and we might have worked on some, but not everyone has knowledge about all the testing types.

Please find the link to testing types’ description: <https://www.softwaretestinghelp.com/types-of-software-testing/>

# Test documentation

**Test strategy** - A high-level document that identifies the Test Levels (types) to be executed for the project.

**Test plan** - A test plan is a complete planning document, which contains the scope, approach, resources, schedule, etc. of testing activities.

**Test scenario** – Test scenario is an item or event of a software system that could be verified by one or more Test cases.

**Test case** – It is a group of input values, execution preconditions, expected execution post conditions and results. It is developed for a Test Scenario.

**Defect report** – Defect report is a documented report of any flaw in a Software System, which fails to perform its expected function.

**Test summary report** - Test summary report is a high-level document, which summarizes testing activities conducted as well as the test result.

Links: [Test scenario with examples](https://www.guru99.com/test-scenario.html);

[Test cases. Examples](https://www.guru99.com/test-case.html);

[Test plan](https://www.guru99.com/what-everybody-ought-to-know-about-test-planing.html);

# Defect management

1. **Bug Declaration**

**Bug should contains:**

1. The screenshot/video of the problem

2. Steps to reproduce the problem (shown in the screenshot)

3. The user that was tested

4. Environment

5. Actual Result

6. Expected Result

7. Referenced Documentation (specification)

Note: Title should contain short description of the problem (general meaning)

**Ex. bug:** Missing one element from page

**1st method of bug description:**

Element Y is missing from screen X. See also attached screenshot a.png

Note: SD XX, Rule YY

**2nd method of bug description:**

1. Actions (steps) to reproduce – write the steps to reproduce the bug

Note: In case of more difficult scenarios there is possibility to show intermediate results.

For ex:

1. Open screen X -> screen X is opened

2. Actual result – the results at that moment

3. Expected result – results that must be according to the specifications

4. If there is necessary to add some additional information- tester just add a note

For ex: Note: Screen X is not opened in Safari

5. Also tester might add information about environment, version of Web Browser, version of application etc. i.e. any information that will help developer to understand very quickly the issue.

Actions:

1. Go to screen A

2. Press X button => Screen S opened

Obtained results: Element E is missing from screen S. See also the attached screenshot a.png

Expected results: Element E is present in the screen. See also Rule YY from SD XX.

1. **Priority of Defects**

**Show-stopper/Blocker/Critical:** Prevents function from being used, no work-around, blocking

progress on multiple fronts. Prevents function from being used, no work-around. A

patch must be delivered immediately, in order to complete the tests.

**Very High:** Defect refers to a key functionality, work-around available to pursue the

test run. Need to be fixed in the very next development iteration.

**High:** Prevents function from being used, but a work-around is possible. Need to be

fixed in the following iteration.

**Medium/Normal:** A problem making a function difficult to use but no special work-around is

required. Need to be fixed before the next deployment on production.

**Low/Minor:** A problem not affecting the actual function, but the behavior is not natural. May

be fixed after the next deployment on production and delivered within next release.

Links: [Defect management](https://www.guru99.com/defect-management-process.html);

[Defect Lifecycle](https://www.guru99.com/defect-life-cycle.html)

# Testing an eCommerce website

In today's world, I bet you won’t find anyone who hasn't shopped online. E-commerce/Retail is a business that thrives on its online customers. Shopping in person vs. shopping online has many advantages. Convenience, time-saving and easy access to products worldwide, etc.

A good [E-commerce](https://en.wikipedia.org/wiki/E-commerce)/Retail site is key to its success. It must be a worthy counterpart to the storefront. Because, when you go shopping at a physical store, the customer has already made a commitment to visit and might give the brand a chance.

Online, choices are many. So, unless there is engagement from the beginning, the user might just leave.

Link: <https://www.softwaretestinghelp.com/ecommerce-testing/>

[180+ Web Application Testing Example Test Cases](https://www.softwaretestinghelp.com/sample-test-cases-testing-web-desktop-applications/)

[User Interface Elements](https://www.usability.gov/how-to-and-tools/methods/user-interface-elements.html)