Contents

Usability testing

Usability testing has now become a vital part of any web based project. It can be carried out by testers like you or a small focus group similar to the target audience of the web application.Test the site Navigation:Menus, buttons or Links to different pages on your site should be easily visible and consistent on all WebPages

Test the Content:

- Content should be legible with no spelling or grammatical errors.
- Images if present should contain an "alt" text

Functionality Testing

Functional testing verifies that each function of the software application operates in conformance with the requirement specification. This testing mainly involves black box testing and it is not concerned about the source code of the application. Each and every functionality of the system is tested by providing appropriate input, verifying the output and comparing the actual results with the expected results.

This testing involves checking of User Interface, APIs, Database, security, client/ server applications and functionality of the Application Under Test. The testing can be done either manually or using automation

Test for – all the links in web pages, database connection, forms used in the web pages for submitting or getting information from user, Cookie testing.

Test all links in your WebPages are working correctly and make sure there are no broken links. Links to be checked will include -

- Outgoing links

- Internal links
- Mailto Links

Functional Testing	Non-Functional Testing
Functional testing is performed using the functional specification provided by the client and verifies the system against the functional requirements.	Non-Functional testing checksthe Performance, reliability, scalability and other non-functional aspects of the software system.
Functional testing is executed first	Non functional testing should be performed after functional testing
Manual testing or automation tools can be used for functional testing	Using tools will be effective for this testing
Business requirements are the inputs to functional testing	Performance parameters like speed , scalability are inputs to non-functional testing.
Functional testing describes what the product does	Nonfunctional testing describes how good the product works
Easy to do manual testing	Tough to do manual testing

Types of Functional testing are	Types of Non functional testing are
Unit Testing	Performance Testing
Smoke Testing	Load Testing
Sanity Testing	Volume Testing
Integration Testing	Stress Testing
White box testing	Security Testing
Black Box testing	Installation Testing
User Acceptance testing	Penetration Testing
Regression Testing	Compatibility Testing
	Migration Testing

End to End Testing

End-to-End Testing not only validates the software system under test but also checks it's integration with external interfaces. Hence, the name "End-to-End". End to End Testing is usually executed after functional and system testing. It uses actual production like data and test environment to simulate real-time settings. End-to-End testing is also called Chain Testing.

End to End Testing	System Testing
Validates the software system as well as interconnected sub-systems	Validates just the software system as per the requirements specifications.
It checks the complete end-to-end process flow.	It checks system functionalities and features.
All interfaces, backend systems will be considered for testing	Functional and Non-Functional Testing will be considered for testing
It's executed once system testing is completed.	It's executed after integration testing.
End to End testing involves checking external interfaces which can be complex to automate. Hence Manual Testing is preferred.	Both Manual and Automation can be performed for system testing

Positive Vs Negative Testing

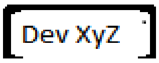
Positive testing determines that your application works as expected. If an error is encountered during positive testing, the test fails.

Negative testing ensures that your application can gracefully handle invalid input or unexpected user behaviour. For example, if a user tries to type a letter in a numeric field, the correct behaviour in this case would be to display the "Incorrect data type, please enter a number" message. The purpose of negative testing is to detect such situations and prevent applications from crashing. Also, negative testing helps you improve the quality of your application and find its weak points.

Note

The core difference between positive testing and negative testing is that throwing an exception is not an unexpected event in the latter. When you perform negative testing, exceptions are expected – they indicate that the application handles improper user behaviour correctly.

Negative Testing: ***Can be performed on the system by providing invalid data as input***. It checks whether an application behaves as expected with the negative input. This is to test the application that does not do anything that it is not supposed to do so.**For example**– Enter any Number



Dev XyZ

Negative Testing.

For the example above Negative testing can be performed by testing by entering alphabets characters from A to Z or from a to z. Either system text box should not accept the values or else it should throw an error message for these invalid data inputs.In both the testing, following needs to be considered:**Input data**

Action which needs to be performed.

Output Result

Testing Technique used for Positive and Negative Testing:

Following techniques are used for Positive and negative validation of testing is:

- Boundary Value Analysis
- Equivalence Partitioning

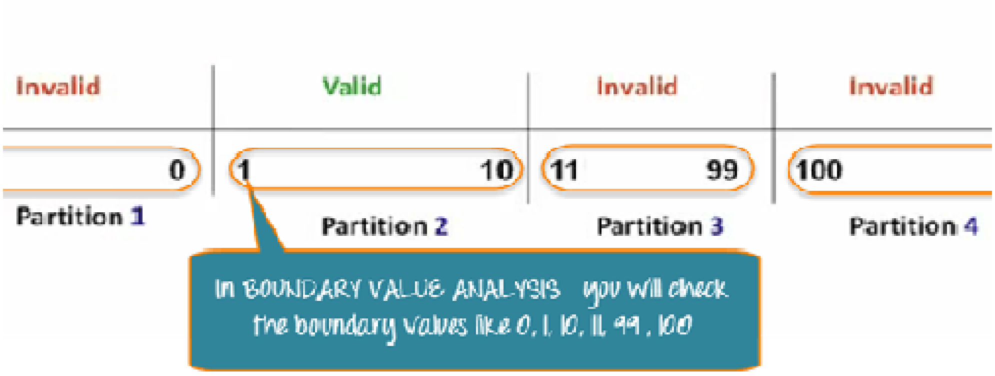
Boundary Value Analysis:

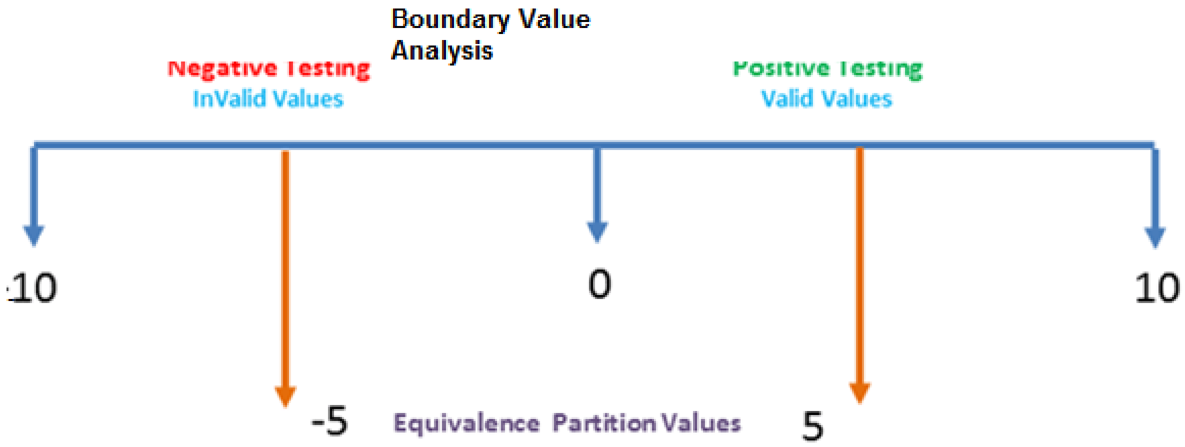
in Boundary Value Analysis, you test boundaries between equivalence partitions This is one of the software testing technique in which the test cases are designed to include values at the boundary. If the input data is used within the boundary value limits, then it is said to be Positive Testing. If the input data is picked outside the boundary value limits, then it is said to be Negative Testing**For example -**A system can accept the numbers from 0 to 10 numeric values. one value for each partition you will check the values at the partitions like 0, 1, 10, 11 and so on.

Equivalence Partitioning:

This is a software testing technique which divides the input date into many partitions .Values from each partition must be tested at least once. Partitions with valid values are used for Positive Testing. While, partitions with invalid values are used for negative testing.

For example-Numeric values Zero to ten can be divided to two(or three)partition. In our case we have two partitions -10 to -1 and 0 to 10. Sample values (5 and -5) can be taken from each part to test the scenarios


Invalid
Valid
Invalid
Invalid
0
1
10
11
99
100
Partition 1
Partition 2
Partition 3
Partition 4
In BOUNDARY VALUE ANALYSIS you will check the boundary values like 0, 1, 10, 11, 99, 100



Adhoc Testing

Structured Testing

In this approach, every activity that occurs during the testing procedure, from the creation of test cases to their sequential execution, everything is scripted. The testers follow this script to conduct tests according to it.

Unstructured Testing

In this approach, testing is commonly done through error guessing, where the testers create the test cases, during the testing process itself.

What is Adhoc Testing?

As discussed above, it is a type of unstructured testing approach, where no organized plan is created before starting the testing process.

Therefore, prior to testing, no requirement documentation or test case planning and designing are done.

Ad-Hoc testing is usually conducted by a tester who has strong knowledge of the software under test, regarding what it does and how it works.
This testing is done by randomly creating test cases through error guessing and executing them, without following any requirements for the test.

A major part of this testing is to find out the potential areas of the software, where errors could be present.

This is why it is also sometimes known as Monkey Testing or Random Testing. This is why it is important for only those testers who have good knowledge of the software to conduct this test.

An advantage of Ad-Hoc testing is that it saves quite a lot of time, which otherwise goes into the creation of documents like test requirements, test case planning, designing, etc.

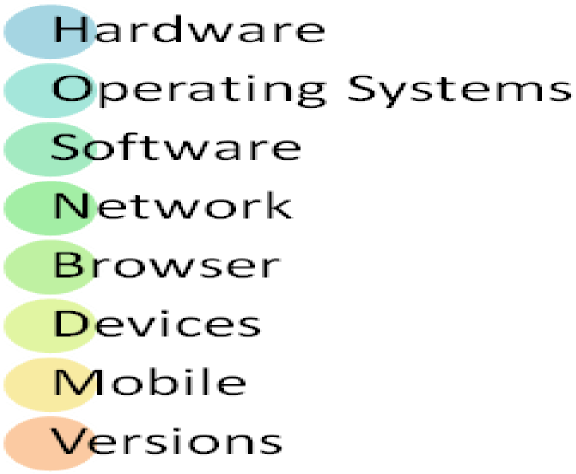
Exploratory Testing

Exploratory testing is all about discovery, investigation, and learning. It emphasizes personal freedom and responsibility of the individual tester. It is defined as a type of testing where Test cases are not created in advance but testers check system on the fly. They may note down ideas about what to test before test execution. The focus of exploratory testing is more on testing as a "thinking" activity.

Under scripted testing, you design test cases first and later proceed with test execution. On the contrary, exploratory testing is a simultaneous process of test design and test execution all done at the same time.

Compatibility testing

Compatibility testing is a type of software testing used to ensure compatibility of the system/application/website built with various other objects such as other web browsers, hardware platforms, users (in case if it’s very specific type of requirement, such as a user who speaks and can read only a particular language), operating systems etc.
This type of testing helps find out how well a system performs in a particular environment that includes hardware, network, operating system and other software etc.



GUI Testing

Graphical User Interface Testing (GUI) Testing is the process for ensuring proper functionality of the graphical user interface (GUI) for a specific application.GUI testing generally evaluates a design of elements such as layout, colours and also fonts, font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links and content.

API Testing

API Testing is a software testing type that validates API (Application Programming Interface).
It is very different from GUI Testing and mainly concentrates on the business logic layer of the software architecture.
Instead of using standard user inputs(keyboard) and outputs, in API Testing, you use software to send calls to the API, get output, and note down the system's response.
This testing won't concentrate on the look and feel of an application.

Security Testing

Security Testing is defined as a type of Software Testing that ensures software systems and applications are free from any vulnerability, threats, risks that may cause a big loss. ***Security testing of any system is about finding all possible loopholes and weaknesses of the system which might result into a loss of information.***
The goal of security testing is to identify the threats in the system and measure its potential vulnerabilities, so the system does not stop functioning or is exploited. It also helps in detecting all possible security risks in the system and help developers in fixing these problems through coding.

Model Based Testing

Model based testing is a software testing technique where run time behavior of software under test is checked against predictions made by a model. A model is a description of a system's behavior. Behavior can be described in terms of input sequences, actions, conditions, output and flow of data from input to output. It should be practically understandable and can be reusable; shareable must have a precise description of the system under test.

Mutation Testing

We mutate (change) certain statements in the source code and check if the test cases are able to find the errors. It is a type of ***white box*** testing which is mainly used for unit testing. The changes in mutant program are kept extremely small, so it does not affect the overall objective of the program.

Mutation Testing is a type of Software Testing that is performed to design new software tests and also evaluate the quality of already existing software tests. Mutation testing is related to modification a program in small ways. It focuses to help the tester develop effective tests or locate weaknesses in the test data used for the program.

Objective of Mutation Testing:

- ✓ To identify pieces of code that is not tested properly.
- ✓ To identify hidden defects that can't be detected using other testing methods.
- ✓ To discover new kinds of errors or bugs.
- ✓ To study error propagation and state infection in the program.

What is Accessibility Testing

Accessibility Testing is defined as a type of Software Testing performed to ensure that the application being tested is usable by people with disabilities like hearing, color blindness, old age and other disadvantaged groups. It is a subset of Usability Testing.

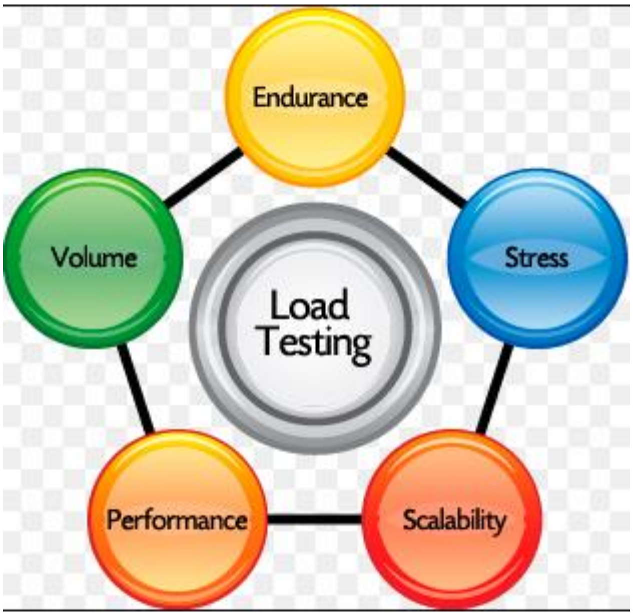
People with disabilities use assistive technology which helps them in operating a software product. Examples of such software are:

Speech Recognition Software - It will convert the spoken word to text , which serves as input to the computer.
Screen reader software - Used to read out the text that is displayed on the screen
Screen Magnification Software- Used to enlarge the monitor and make reading easy for vision-impaired users.

Performance Testing:

This will ensure your site works under all loads.

- ✓ Website application response times at different connection speeds
- ✓ Load test your web application to determine its behaviour under normal and peak loads
- ✓ Stress tests your web site to determine its break point when pushed to beyond normal loads at peak time.
- ✓ Test if a crash occurs due to peak load , how does the site recover from such an event
- ✓ Make sure optimization techniques like gzip compression , browser and server side cache enabled to reduce load times



Performance Testing is defined as a type of software testing to ensure software applications will perform well under their expected workload.

Scalability - Determines maximum user load the software application can handle.
Endurance tests - is done to make sure the software can handle the expected load over a long period of time.
Volume testing–Check the same functionality with large data.

Stress Testing

Stress testing is used to test the stability & reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions. It even tests beyond the normal operating point and evaluates how the system works under those extreme conditions. Stress Testing is done to make sure that the system would not crash under crunch situations.

The application under testing will be stressed when 5GB data is copied from the website and pasted in notepad. Notepad is under stress and gives 'Not Responded' error message.

Load Testing Vs Stress Testing: Load Testing is to test the system behaviour ***under normal workload*** conditions, and it is just testing or simulating with the actual workload.
But Stress testing is to test the system behaviour ***under extreme conditions*** and is carried out till the system failure.

Load testing does not break the system, but stress testing tries to break the system by testing with overwhelming data or resources.

Load Testing

Load testing is a kind of performance testing which determines a system's performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously

This testing usually identifies -

- ✓ The maximum operating capacity of an application.
- ✓ Determine whether current infrastructure is sufficient to run the application.
- ✓ Sustainability of application with respect to peak user load.
- ✓ Number of concurrent users that an application can support, and scalability to allow more users to access it.
- ✓ It is a type of non-functional testing.
- ✓ Load testing is commonly used for the Client/Server, Web based applications.
- ✓

Load Testing	Stress Testing
Load testing identifies the bottlenecks in the system under various workloads and checks how the system reacts when the load is gradually increased	Stress testing determines the breaking point of the system to reveal the maximum point after which it breaks.

Penetration Testing

A penetration test, colloquially known as a pen test, pentest or ethical hacking, is an authorized simulated cyberattack on a computer system, performed to evaluate the security of the system.
In the context of web application security, penetration testing is commonly used to augment a web application firewall (WAF).

Black Box

In Black Box Testing we just focus on inputs and output of the software system **without bothering about internal knowledge of the software program.**

Types of Black Box Testing

- **Functional testing** - This black box testing type is related to functional requirements of a system; it is done by software testers.
- **Non-functional testing** - This type of black box testing is not related to testing of a specific functionality , but non-functional requirements such as performance, scalability, usability.
- **Regression testing** - Regression testing is done after code fixes , upgrades or any other system maintenance to check the new code has not affected the existing code

Black box testing strategy:

- **Equivalence Class** Testing: It is used to minimize the number of possible test cases to an optimum level while maintains reasonable test coverage.
- **Boundary Value** Testing: Boundary value testing is focused on the values at boundaries.

Web Testing

Test the applications that are hosted on web in which the application interfaces and other functionalities are tested.

Web testing checklist.

1) Functionality Testing2) Usability testing3) Interface testing 4) Compatibility testing5) Performance testing6) Security testing 7) Database Testing

1) Functionality Testing:

Test for – all the links in web pages, database connection, forms used in the web pages for submitting or getting information from user, Cookie testing.

Test all links in your WebPages are working correctly and make sure there are no broken links. Links to be checked will include -

- Outgoing links
- Internal links
- Anchor Links
- Mailto Links

2. Usability testing:

Usability testing has now become a vital part of any web based project. It can becarried out by testers like you or a small focus group similar to the target audience of the web application.Test the site Navigation:Menus , buttons or Links to different pages on your site should be easily visible and consistent on all WebPages

Test the Content:

- Content should be legible with no spelling or grammatical errors.
- Images if present should contain an "alt" text

3. Interface Testing:

Three areas to be tested here are - Application, Web and Database Server

- Application: Test requests are sent correctly to the Database and output at the client side is displayed correctly. Errors if any must be caught by the application and must be only shown to the administrator and not the end user.
- Web Server: Test Web server is handling all application requests without any service denial.
- Database Server: Make sure queries sent to the database give expected results.

Test system response when connection between the three layers (Application, Web and Database) can not be established and appropriate message is shown to the end user.

4.Database Testing:

Database is one critical component of your web application and stress must be laid to test it thoroughly. Testing activities will include-

- Test if any errors are shown while executing queries
- Data Integrity is maintained while creating, updating or deleting data in database.
- Check response time of queries and fine tune them if necessary.
- Test data retrieved from your database is shown accurately in your web application

5. Compatibility testing

Compatibility tests ensure that your web application displays correctly across different devices. This would include-

Browser Compatibility Test: Same website in different browsers will display differently. You need to test if your web application is being displayed correctly across browsers , javascript , AJAX and authentication is working fine. You may also check forMobile Browser Compatibility.

The rendering of web elements like buttons, text fields etc changes with change in **Operating System**.

Make sure your website works fine for various combinations of Operating systems such as Windows, Linux , Mac and Browsers such as Firefox, Internet Explorer, Safari etc.

6. Performance Testing

This will ensure your site works under all loads. Testing activities will include but not limited to -

- Website application response times at different connection speeds
- Load test your web application to determine its behaviour under normal and peak loads
- Stress tests your web site to determine its break point when pushed to beyond normal loads at peak time.
- Test if a crash occurs due to peak load , how does the site recover from such an event
- Make sure optimization techniques like gzip compression , browser and server side cache enabled to reduce load times

7. Security testing

Security testing is vital for e-commerce website that store sensitive customer information like credit cards. Testing Activities will include-

- Test unauthorized access to secure pages should not be permitted
- Restricted files should not be downloadable without appropriate access
- Check sessions are automatically killed after prolonged user inactivity
- On use of SSL certificates, website should re-direct to encrypted SSL pages.