Software Testing #5

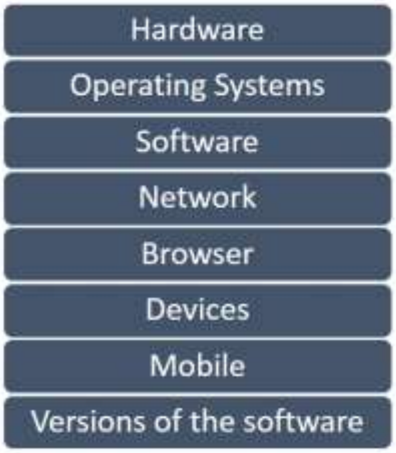
Types of Testing(cont.)

- **Comparison Testing-**
  - Testing technique which compares the product strengths and weaknesses with previous versions or other similar products.
  - Can be performed by tester, developers, product managers or product owners.

- **Compatibility Testing-**
  - Compatibility is nothing but the capability of existing or living together without any discrepancy
  - Compatibility is non-functional testing to ensure customer satisfaction.
  - It is to determine whether your software application or product is proficient enough to run in different browsers, database, hardware, operating system, mobile devices, and networks.

- To continue

Types of Compatibility Tests(1/5)



Hardware

Operating Systems

Software

Network

Browser

Devices

Mobile

Versions of the software

Types of Compatibility Tests (2/5)

- **Hardware**
  - It checks software to be compatible with different hardware configurations.
- **Operating Systems**
  - It checks your software to be compatible with different Operating Systems like Windows, Unix, Mac OS etc.
- **Software**
  - It checks your developed software to be compatible with other software. For example, MS Word application should be compatible with other software like MS Outlook, MS Excel, VBA etc.
- **Network**
  - Evaluation of performance of a system in a network with varying parameters such as Bandwidth, Operating speed, Capacity.
  - It also checks application in different networks with all parameters mentioned earlier.

Types of Compatibility Tests(3/5)

- **Browser**
  - It checks the compatibility of your website with different browsers like Firefox, Google Chrome, Internet Explorer etc.
  - Ensured in cross browser testing
- **Devices**
  - It checks compatibility of your software with different devices like USB port Devices, Printers and Scanners, Other media devices and Blue tooth.
- **Mobile**
  - Checking your software is compatible with mobile platforms like Android, iOS etc.

Types of Compatibility Tests(4/5)

- **Versions of the software**
  - It is verifying your software application to be compatible with different versions of the software.
  - For instance checking your Microsoft Word to be compatible with Windows 7, Windows 7 SP1, Windows 7 SP2, Windows 7 SP3.
  - **Backward Compatibility Testing**– Testing of the application or software in old or previous versions. It is also known as downward compatible.
  - **Forward Compatibility Testing**– Testing of the application or software in new or upcoming versions. It is also known as forward compatible

Common Compatibility testing defects(5/5)

- Changes in UI (look and feel)
- Change in font size
- Alignment related issues
- Change in CSS style and color
- Scroll bar related issues
- Content or label overlapping
- Broken tables or Frames

- **Configuration Testing-**
  - Configuration testing is defined as a software testing type, that checks an application with multiple combinations of software and hardware to find out the optimal configurations that the system can work without any flaws or bugs.
  - Configuration testing is not only restricted to Software but also applicable for Hardware which is why it is also referred as a Hardware configuration testing, where we test different hardware devices like Printers, Scanners, Web cams, etc. that support the application under test.

- **Configuration Testing-**
  - Configuration testing is defined as a software testing type, that checks an application with multiple combinations of software and hardware to find out the optimal configurations that the system can work without any flaws or bugs.
  - Configuration testing is not only restricted to Software but also applicable for Hardware which is why it is also referred as a Hardware configuration testing, where we test different hardware devices like Printers, Scanners, Web cams, etc. that support the application under test.
- **Compliance Testing-**
  - Type of testing which checks whether the system was developed in accordance with standards, procedures and guidelines.
  - It is usually performed by external companies

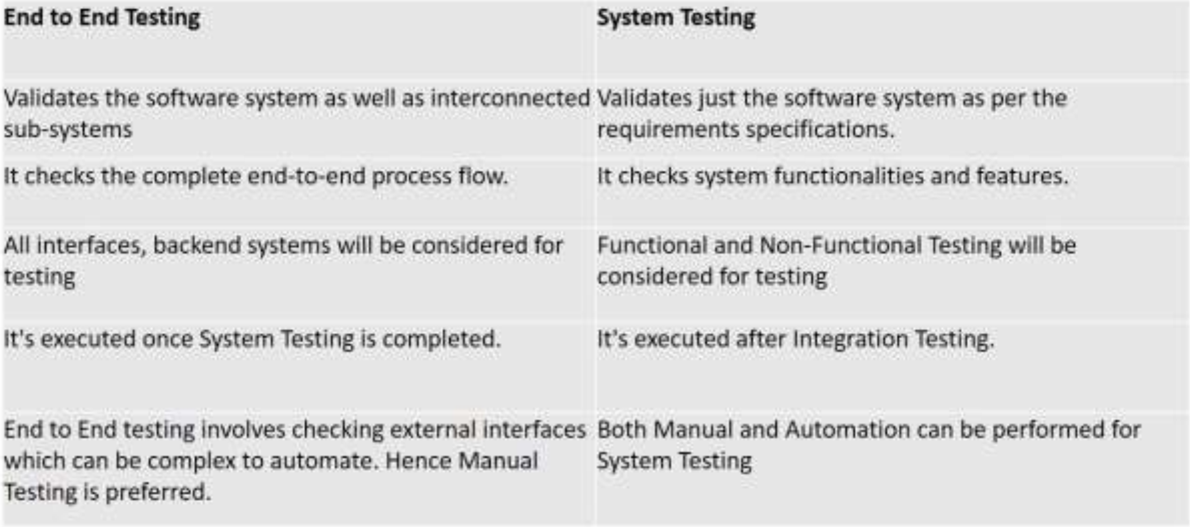
- **Destructive Testing-**
  - Destructive Testing is defined as a software testing type to find points of failure in a software program.
  - It is a testing method where an application is intentionally made to fail to check the robustness of the application and identify the point of failure.
  - For Destructive Testing, it is **not** necessary to have the knowledge of the original requirements of a software product.

- **Endurance Testing-**
  - Where a system is tested with a load extended over a significant amount of time, to analyze the behavior of the system under sustained use.
  - This type of testing is performed at the last stage of **performance run cycle**.
  - It ensures that the application is capable enough to handle the extended load without any deterioration of response time.
- In Endurance Testing following things are tested.
  - Test memory leakage
    - Checks are done to verify if there is any memory leakage in the application, which can cause crashing of the system or O.S.
  - Test connection closure between the layer of the system
    - If the connection between the layers of the system is not closed successfully, it may stall some or all modules of the system.
  - Test database connection close successfully
    - If the database connection is not closed successfully, may result in system crash
  - Test response time
    - System is tested for the response time of the system as the application becomes less efficient as a result of the prolonged use of the system.

- **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.
  - Determination of test cases during testing
  - Generation of new ideas during test execution
  - Documentation of all events during execution is difficult to record
  - Don't know when to stop the exploratory testing

- **Error-Handling Testing-**
  - Software testing type which determines the ability of the system to properly process erroneous transactions.
  - It is usually performed by the testing teams.

- **End-to-end Testing-**
  - Similar to system testing
  - Involves testing of a complete application environment in a situation that **mimics real-world** use, such as interacting with a **database**, using **network communications**, or interacting with **other hardware, applications, or systems**
  - End to end vs System testing *(P.T.O)*



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

- **Fault injection Testing-**
  - Enable the testers to concentrate on the manner in which the application under test is able to handle exceptions

- **Functional Testing-**
  - It is a Black-box type testing geared to the functional requirements of an application
  - This type of testing ignores the internal parts and focuses only on the output to check if it is as per the requirement or not.

- **GUI Testing-**
  - GUI testing is to validate the GUI as per the business requirement
  - The expected GUI of the application is mentioned in the Detailed Design Document and GUI mockup screens
  - The GUI testing includes the size of the buttons and input field present on the screen, alignment of all text, tables and content in the tables
  - It validates that the page does not fluctuate and the alignment remains same after hovering the mouse on the menu or sub-menu

- **Gorilla Testing-**
  - One module or the functionality in the module is tested thoroughly and heavily.
  - The objective of this testing is to check the robustness of the application.
  - Gorilla Testing is a testing type performed by a tester and sometimes by developer the as well.

- **Globalization Testing-**
  - Testing method that checks proper functionality of the product with any of the culture/locale settings using every type of international input possible.
  - In a globalized product, a code is separated from the messages or information. With the help of globalization, it enables software to be used with different languages without having to redesign the complete software.
  - Ex- Multi-lingual website
- **Happy Path Testing-**
  - The objective of Happy Path Testing is to test an application successfully on a positive flow.
  - It does not look for negative or error conditions.

- **Install/Uninstall Testing-**
  - Quality assurance work that focuses on what customers will need to do to install and set up the new software successfully.
  - It may involve full, partial or upgrades install/uninstall processes and is typically done by the software testing engineer
- **Load Testing-**
  - Testing technique that puts demand on a system or device and measures its response.
  - It is usually conducted by the performance engineers. Usually automated
  - It identifies-
    - The maximum operating capacity of an application
    - Determine whether the current infrastructure is sufficient to run the application
    - Number of concurrent users that an application can support, and scalability to allow more users to access it.

- **Monkey Testing-**
  - Monkey testing is a technique in software testing where the user tests the application by providing random inputs and checking the behavior (or try to crash the application)
  - There are no rules; this technique does not follows any predefined test cases or strategy and thus works on tester's mood and gut feeling.
  - We testers also have to become Monkey, think and eventually test it –
  - Smart Monkeys vs Dumb Monkeys *(P.T.O)*

- **Smart Monkeys –**
  - Have a brief idea about the application
  - They know where the pages of application will redirect to.
  - They know that the inputs they are providing are valid or invalid.
  - They work or focus to break the application.
  - In case they find an error, they are smart enough to file a bug.
  - They are aware of the menus and the buttons.
  - Good to do stress and load testing.
- **Dumb Monkey –**
  - They have no idea about the application.
  - They don't know that the inputs they are providing are valid or invalid.
  - They test the application randomly and are not aware of any starting point of the application or the end to end flow.
  - Though they are not aware of application, but they too can identify bugs like environmental failure or hardware failure.
  - They don't have much idea about the UI and functionality

- **Negative Testing-**
    - Having the mindset of “attitude to break”
    - Also known as "test to fail“
    - Using negative testing testers validate that if system or application breaks
    - It validates that if the system throws an error of invalid input and behaves as expected.

- **Performance Testing-**
    - also knows as ‘Perf Testing’
    - Load Testing
        - What is the maximum load the application is able to hold before the application starts behaving unexpectedly?
    - Stress Testing
        - What is the maximum load a system can sustain before it breaks down?
    - Capacity Testing
        - Will the application be able to support the future load?

- **Recovery Testing-**
    - It is a type of testing which validates that how well the application or system recovers from crashes or disasters.
    - Recovery testing determines if the system is able to continue the operation after a disaster.

- **Penetration Testing-**
    - Testing method which evaluates the security of a computer system or network by simulating an attack from a malicious source.
    - Usually they are conducted by specialized penetration testing companies.
    - Disadvantages of Penetration Testing-
        - Data Loss and Corruption
        - Down Time
        - Increase Costs

- **Ramp Testing-**
  - Type of testing consisting in raising an input signal continuously until the system breaks down.

- **Requirements Testing-**
  - Testing technique which validates that the requirements are correct and complete
  - Requirements should be logically consistent and allows designing a necessary and sufficient set of test cases from those requirements

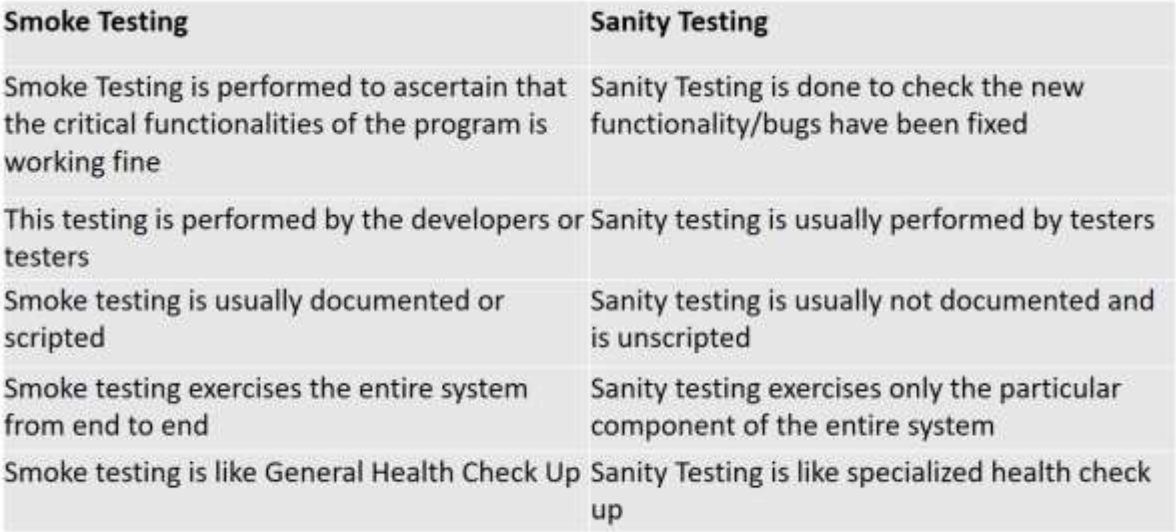
- **Sanity Testing-**
  - Testing technique which determines if a new software version is performing well enough to accept it for a major testing effort Testing technique which determines if a new software version is performing well enough to accept it for a major testing effort

- **Ramp Testing-**
  - Type of testing consisting in raising an input signal continuously until the system breaks down.

- **Requirements Testing-**
  - Testing technique which validates that the requirements are correct and complete
  - Requirements should be logically consistent and allows designing a necessary and sufficient set of test cases from those requirements

- **Sanity Testing-**
  - Testing technique which determines if a new software version is performing well enough to accept it for a major testing effort Testing technique which determines if a new software version is performing well enough to accept it for a major testing effort.

  - Sanity vs Smoke *(P.T.O)*



Smoke Testing	Sanity Testing
Smoke Testing is performed to ascertain that the critical functionalities of the program is working fine	Sanity Testing is done to check the new functionality/bugs have been fixed
This testing is performed by the developers or testers	Sanity testing is usually performed by testers
Smoke testing is usually documented or scripted	Sanity testing is usually not documented and is unscripted
Smoke testing exercises the entire system from end to end	Sanity testing exercises only the particular component of the entire system
Smoke testing is like General Health Check Up	Sanity Testing is like specialized health check up

- **Storage Testing-**
  - Testing type that verifies the program under test stores data files in the correct directories and that it reserves sufficient space to prevent unexpected termination resulting from lack of space.
- **Requirements Testing-**
  - Testing technique which validates that the requirements are correct and complete
  - Requirements should be logically consistent and allows designing a necessary and sufficient set of test cases from those requirements
- **Sanity Testing-**
  - Testing technique which determines if a new software version is performing well enough to accept it for a major testing effort Testing technique which determines if a new software version is performing well enough to accept it for a major testing effort.
  - Sanity vs Smoke *(P.T.O)*

Thank You