**1.**

In my QA Tester career, I have been working on a variety of system platforms and operating systems like Windows 98, Windows 2000, Windows XP, Windows Vista, Windows 7, Redhat Linux and Solaris UNIX.

I started QA as intern from my college.i was very interested in writing code so one day my professor bought me a paper and old me if I am interested to do Intership for New York Times. Then I started my internship as a QA intern. I was attending more into meeting since I was intern Authorities wanted me to know more about the work and Environment .

first I had no idea what I was doing but my mentor Mr.cornors he was very nice person that he explained me about each and every single Meeting and Technical Environment. Since NYT is paper company and recently they started Launching app so user can read news paper throw apps.There I was using java language to automate application.I was using Eclips Id to write java code and eclips gives us ability to write complie and run in a same time .I used sql server for sql query run query.i used page object model to make it more business readable and avoid code duplication.

Right now I am working for Macys .Macys is eCommerce Company.we

In my current company we practice Agile to develop and deploy Software. We have 2 weeks sprint cycle, from the day first for this cycle i select which test cases are good candidates for  
automation as we can't automate everything . Daily basis i write Java code for automation scripting.

I use Eclipse as IDE, Junit and testng as a testing framework, Maven for building the automation project, in project level to organize my code i use page object model which mainly helps to avoid code duplication and it makes more bussines readable.Git as code repository, Jenkins for running our script during build process.To run cross browsing and parralel test i used selenium grid.For remotely run my project i used sauce labs.I have also used log4j as a logging framework.I have good knowledge about webservice testing for that i have used bdd and cucumber framework also.I also run sql queries to validate database,I have sound knowledge about  linux commands as our web server under Linux OS. Have sound knowledge in Web Architecture like - HTML, CSS, Javascript, Jquery, AJAX, HTTP/HTTPS protocol, Web Services like REST.

My

I have carefully tested client server applications and web-based applications (IIS, Apache, Tomcat, Weblogic) developed in Visual Basic, C++, Java, PHP, SQL, and many others computer languages.

**I have been working as a QA Engineer for the last 5 years.In my career I have used various platform like linux,mac,windows. In my current role i am mostly involved in writing automation script for web based application using Selenium WebDriver API.  
In my current company we practice Agile to develop and deploy Software. We have 2 weeks sprint cycle, from the day first for this cycle i select which test cases are good candidates for  
automation as we can't automate everything . Daily basis i write Java code for automation scripting.  
I use Eclipse as IDE, Junit and testng as a testing framework, Maven for building the automation project, in project level to organize my code i use page object model which mainly helps to avoid code duplication and it makes more bussines readable.Git as code repository, Jenkins for running our script during build process.To run cross browsing and parralel test i used selenium grid.For remotely run my project i used sauce labs.I have also used log4j as a logging framework.I have good knowledge about webservice testing for that i have used bdd and cucumber framework also.I also run sql queries to validate database,I have sound knowledge** **about  linux commands as our web server under Linux OS. Have sound knowledge in Web Architecture like - HTML, CSS, Javascript, Jquery, AJAX, HTTP/HTTPS protocol, Web Services like REST.**

**Eclips,Testng,Maven,PageBojectModel,GitJenkins,GridSauceLabs,Log4J,webservice,bdd,,commandsAchchitecture**

**Regression testing** is a type of [software testing](https://en.wikipedia.org/wiki/Software_testing) that verifies that software previously developed and tested still performs correctly even after it was changed or interfaced with other software.

**Functional testing** is a [quality assurance](https://en.wikipedia.org/wiki/Quality_assurance) (QA) process[[1]](https://en.wikipedia.org/wiki/Functional_testing#cite_note-Prasad-1) and a type of [black-box testing](https://en.wikipedia.org/wiki/Black-box_testing) that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (unlike [white-box testing](https://en.wikipedia.org/wiki/White-box_testing)).[[2]](https://en.wikipedia.org/wiki/Functional_testing#cite_note-KanerFalkNguyen1999-2) Functional testing usually describes *what* the system does.

Functional testing does not imply that you are testing a function (method) of your module or class. Functional testing tests a slice of functionality of the whole system.

**smoke testing** /**intake testing**(also **confidence testing**, **sanity testing**[[1]](https://en.wikipedia.org/wiki/Smoke_testing_(software)#cite_note-glossary-1)[[2]](https://en.wikipedia.org/wiki/Smoke_testing_(software)#cite_note-2)) is preliminary testing to reveal simple failures severe enough to (for example) reject a prospective software release. A smoke tester will select and run a subset of [test cases](https://en.wikipedia.org/wiki/Test_case) that cover the most important functionality of a component or system, to ascertain if crucial functions of the software work correctly.[[](https://en.wikipedia.org/wiki/Smoke_testing_(software)#cite_note-DustinRashkaPaul1999-3)

The process of smoke testing aims to determine whether the application is so badly broken as to make further immediate testing unnecessary

**Integration testing: test all the code and functionality together to get out put.**

**SP-Sprint Planing Metting-main purporse of this reason of this meeting so sprint become successful.Time assumption between all the team member.**

**DSU-Daily Standup meeting,If any obstacle,what I did yesterday,what I will do today.,what I have done and in order to get new task,also get to know about team members**

**Demo-**

**Retro-**

**When**

**Who lead**

**Why/purpose**

**Who are the attendance**

**What is the expected output**

**What is qa input**

**Usually 1 story point equal to 1**

**Story point is not for us its for reporting people**

**CEO,PO,BA,MANAGER,ARC-ENG,DES,B-DEV,F-DEV,QA,DEV-OP,MARKETTING**

**I DONT SEE CEO,I ONLY TIME MEET THEM WHEN THEY SETUP THE ENVIRONMENT AND THE DESIGN OF THE PREOJECT.**

**ALL BIG COMPANY NOW DON’T KEEP IN HOUSE DESIGNER SO WE GET IT DONE FROM (ROSETA) THIRD PARTY COMPANY,THEY GIVE US ONE FUNCTIONAL AND ONE NON FUNCTIONAL,NOW THEY PROVIDE PROTO TYPE(VERY MUCH SIMILLER BUT FUNCTIONALITY WONT BE WORK SAME)**

**DEV-OP THEY CREATE JENKIN OR WORK WITH SERVER TYPE WORK,**

**If you have scrum developer will finish with story then they ask you who wants to.**

**If I have canvan I decide what story point should be done by sequence**

As a detail oriented QA Tester, I successfully developed test plans, test cases, test scripts, traceability matrices and attended zillion meetings with the Business Analysts, Project Managers, Business Managers, Software Developers and QA Leads. As far as different types of testing, I have successfully performed and lead others to perform Smoke Testing, Functional Testing, Backend Testing, Black Box Testing, White Box Testing, Integration Testing, Regression Testing, Load Testing and Stress Testing, constantly documenting and sharing knowledge about test projects with team members

I have discovered bugs manually and with test automation tools like SilkTest, HP QTP and Selenium and immediately reported bugs to software developer using Bugzilla, Trace and Jira

As QA Tester, I continue to acquire and enhance technical and non-technical skills by closely working with peers and using any available opportunities for that. I continue to be on the lookout for training opportunities, and I always he find effective ways to employ those new skills in my work

This is pretty much what I have been doing as a QA Tester in the past years.

2.**Agile is set of values and principles which lead our mind to setup.**

Cover values of agile.

a. individuals and interaction process and tools

b. Working product over comprehensive documentation.

c. customer collaboration over contract negations. Crete connection using get coonection method

,create statement, get the result set to exucte qury,then u write your qury , also you close connecton using con. Close method

,driver register, using for name method

jdk pth,maven path,language mention,sequrity option,type of emailable report configure,time factor,how many buid will run if it will run multiple time,

d. Responding to change over following plan.

**Agile challenging as a QA?**

Challenging u face in Agile Environment: one of the challenging time is release time: mostly it takes 3 hours to release .Some time it takes more or less time. If we have 2 hours of release and developer take 1 hour then it takes 20-30 minute to come to the side and we will only have 30 minutes for final test manually. Releases are   very critical for us  .Specially As QA we have to do FINAL SIGN UP where we have more responsibility than everyone. Application quality depend based on our test .If Website or the whole application gets down As a QA I will be blamed for the error.

When it's middle of the sprint PO (PRODUCT OWNER) change the requirement its becomes very challenging for us . As I worked/did in past few weeks everything goes to Back Log.And I have to start working based on PO which has more priority.

4. how to enable index for maven?

Go to windows then click on preference and then enable your index from maven.

5. In order to setup testNg we have to go to help and click on install new software then past the url from the TestNg side.

DataProvider,PageFctory,POI,WindowsHandling

**How to get data from excel shit ?**

Using XXSF Work shit

Jenkins-Continuous integration tool ,Git –version control tool,Jira- using Jira to track and finding Bug and if I find any major bug that needs to fix I prioritize the Bugs that need to get fix up in order continuous working.And I also search in jira

**What is grooming?** It’s a another name of sprint meeting.

**What is story point?**

We used traditional agile so we didn’t have story point .Point mean time.

product back keeps everything of.

Every quarter or 3 month of agile were planed for

We use safe scrum of agile .most important we used velocity .

**What is velocity ?**

Velocity is a metric that predicts how much work an [Agile software development](http://searchsoftwarequality.techtarget.com/definition/agile-software-development) team can successfully complete within a two-week [sprint](http://searchsoftwarequality.techtarget.com/definition/Scrum-sprint) (or similar time-boxed period).

Velocity is a useful planning tool for estimating how fast work can be completed and how long it will take to complete a project. The metric is calculated by reviewing work the team successfully completed during previous sprints; for example, if the team completed 10 [stories](http://searchsoftwarequality.techtarget.com/definition/story) during a two-week sprint and each story was worth 3 [story points](http://whatis.techtarget.com/definition/story-point), then the team's velocity is 30 story points per sprint.

**Who sets the time for project?**

Sizing and tasking meeting where we decide about time frame about the sprint .developer will give a time how long it will take to create a module and QA will give time how long we will take time to test for the sprint Between Developer,Qa and BA.

**What is story ?**

Based on PO requirement (when BRD isn’t available)If it’s a story Ui designer will design then back end and front developer will create the story.then QA will test.

**What is RTM and TM (Required Traceability Matrix )?**

Tm Is a document where

**What is difference between list and set?**

A **Set** cannot contain duplicate elements while a **List** can. A **List** (in Java) also implies order. Conceptually we usually refer to an unordered grouping that allows duplicates as a Bag and doesn't allow duplicates is a **Set**. **List** is used to collection of elements with duplicates.

**How would you test if we gave a brand new Site or Application?**

First I will understand the application or site and I will gather important business about the application or the important functionality planning for smoke testing where I will take the important elements to run the business of the application and after I am done with smoke test I will plan for regression test

**What is White Box testing?**

**Test done by developers**

**What is Black Box testing?**

**Gui testing is part of Black Box Testing**

**What is Gui Testing / UI testing ?**

Graphical User Interface (GUI) testing is the process of testing the system's GUI of the System Under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars-tool bar, menu bar, dialog boxes and windows etc.

**It’s a ui testing which mean we look at the site without looking at the code.**

**What is load or performance testing?**

**How you will retrieve/get data from ?**

**Lets say you have to write script for 100 people registration?**

**Data provider.**

**How to retrieve data from excel sheet ?**

**How you will determine getting unique values?**

**Statement and pepare statement?**

**Who start story point meeting?**

**Do you know about canvas/**

**Difference between agile board and Jira board?**

**Why you think daily stand up meeting is important?**

**What is retrospective meeting?**

**In you agile challenge you said about frequent build fail can you tell me more about it?**

**5 agile challenging ?**

**what you don’t like about your qa manager?**

**How do you cover automation and manual ?**

**How many people in your team?**

**Did you do automation and manual both ?**

**There was any time that you weren’t able to finish you work before sprint ?**

**Tell me benefits about Agile Environment.**

**Are your fit for our job?**

**If I ask you just to be manual tester would you do ?**

**Do you know virtualization and do you know lisa ?**

**Difference between selenium webdriver and selenium rc.**

**What is protector ?**

**Are you willing to learn new tolls ?**

**Tell me about your hobby?**

**Tell me why should we hire you?**

**Tell me about your weakness?**

**What is technical debt?**

**Technical debt** (also known as design **debt** or code **debt**) is "a concept in programming that reflects the extra development work that arises when code that is easy to implement in the short run is used instead of applying the best overall solution".

**What do you like more about agile?**

**What you like between agile environment and waterfall?**

**What you do first when we give you something to test.**

**Why cant you do 100 percent automation ?**

**Tell me about 30 percent functionality cant automate?**

**What percentage of time you use to do automation and maual testing?**

**How do you prioritize what needs be automated and what should be manual?**

**Lets say you have two high priority card in a same time and you have to take decision since your manager and qa lead is not present?**

**Can you tell me about the release or deploy mechanism/process?**

**Send it o UAT and they send it to me and TEST before and test release ?**

**If you find critical bug before relase or after release.**

**Lets say you going at design document and you find wrong and you know its not right who do you lead ?or you not sure ifts a correct design or not**

**Why you became a QA?**

**How would you open fix version of Firefox browsers version 46 using selenium**

FirefoxBinary binary = new FirefoxBinary(new File("path\_to\_bin"));

FirefoxProfile profile = new FirefoxProfile();

WebDriver driver = new FirefoxDriver(binary, profile);

WebDriver driver = new FirefoxDriver();

System.setProperty("webdriver.firefox.bin", "c:\\path\\to\\firefox.exe");

File file = new File("D:\\selnium webdriver\\driver\\chromedriver.exe");

System.setProperty("webdriver.chrome.driver", file.getAbsolutePath() );

WebDriver driver = new ChromeDriver();

**Type?**

**How much coding you do everyday? How you manage your code?**

**How you keep your code and you leave that feature ?**

**Who does the code review for your company ?t**

**How you will send the code ?**

**What challenges you face as a QA in Agile ?**

**Who release your code?**

**If you have any problem what you do and who let you know during you working period.**

**How would you force you other team member using you method without any change what would you do ?**

**How would you see all the color of array**

**How would you convert one type to another data ?**

**How you debug your code ?**

**Do you know oracle and what is it ?**

**What is obsor team ?**

**What’s your responsibilities as a QA**

**If you design a framework how would you do ? or design framework from the scratch.**

**What would you do if you need fix only 4 or 5 bug every what would you chose?**

**In scrum you need backlog,Feature client requirement,**

**How frequently you release your code?waht happen your client request to you? What you do to him?**

**How would you deal with critical bug/show stoppers.**

**Lets say im writing or automating new functionality and it will effect the old functionality what would you do?**

**Who prioritize the card ?**

**Scrum master**

**What you do if you fail to do the card ? what you would do ?**

**How would you know this specific defect is relate to what feature.**

**Whitebox and Black box testing and unit testing .**

**Middle of the day you find defect what would you do /**

**Did you ever used git ?tell me a scenario what did you do ?**

**Tell me in details about the coding .**

**Have you merged any code in git?**

**How you manage your code in git ?**

**Lets say you 20-30 branch during project time what would you do after project you keep it or trash it?**

**How do you do regression testing ?integration testing?**

**Can you tell me about team size and structure.**

**Who leads the daily stand up meeting?**

**Scrum master**

**Do you have any manger in your team?**

**how many manager you have .**

**How many team you have in your IT Team?**

**What if you release something that will effect others ? how would you handle?**

**Discuss within two team and managers**

**How you manage your time in terms of automation and manual?**

**If you don’t have time for automation what would you do ?**

**What is n to n testing ?**

**How many time you actually do regression testing?**

**Difference between Parallel testing and crows browsing**

**Do you know any dev toll?**

**How would you know it’s a client side /machine error or server side error ?**

**Benefits of page object model?**

**How would you run 5 TestNg xml in a same time?**

**What is profile?**

**Three things you should have In agile work environment so you could work efficiently.**

**How you mange your self when you stress ?**

**If you find difficulty and what you would do?**

**Differences between array and linked List**

|  |  |
| --- | --- |
| **ArrayList** | **LinkedList** |
| 1) ArrayList internally uses **dynamic array** to store the elements. | LinkedList internally uses **doubly linked list** to store the elements. |
| 2) Manipulation with ArrayList is **slow** because it internally uses array. If any element is removed from the array, all the bits are shifted in memory. | Manipulation with LinkedList is **faster** than ArrayList because it uses doubly linked list so no bit shifting is required in memory. |
| 3) ArrayList class can **act as a list** only because it implements List only. | LinkedList class can **act as a list and queue** both because it implements List and Deque interfaces. |
| 4) ArrayList is **better for storing and accessing** data. | LinkedList is **better for manipulating** data. |

**10.What is maven life-cycle?**

**Ans:**

* **Validate - validate that the project is correct and all necessary information is available**
* **compile - compile the source code of the project**
* **test - test the compiled source code using a suitable unit testing framework. These tests should not require the code be packaged or deployed**
* **package - take the compiled code and package it in its distributable format, such as a JAR.**
* **integration-test - process and deploy the package if necessary into an environment where integration tests can be run**
* **verify - run any checks to verify the package is valid and meets quality criteria**
* **install - install the package into the local repository, for use as a dependency in other projects locally**
* **deploy - done in an integration or release environment, copies the final package to the remote repository for sharing with other developers and projects.**

**What is release mechanism in your company ?**

**Build, Profile, Plugin for testNg.xml to run by individual group or test from cmd.**

**how to compare application date with current date?**

**How to validate search engine?**

**Difference between equals and equals ?**

**Boundary value analysis?//equivalence partition**

**//Face amount ?**

**how to run suite from bdd cucumber?**

**What is glow?**

**What is data table? did you ever used ?**

Do you have any idea about Api testing

?

what is your ideal environment ?

I worked on different different project which had different different environment and so I could make myself comfortable with any environment .and I am proactive If I have any problem understanding about the project .then I try to discuss with my manager or BA to know about project details and sometime make my launch time with my BA or Manager so I have extra time to get more details about project knowledge.

How many server we have ? Release mechanism

For example we have 4 server 1, dev1-

2.dev2-between QA and Developer then developer push it to QA server ,QA-when QA test it for final release code goes to stage ,Stage-

//for testing environment

Unit Testing-defect log/bug log /fall out log/change log-not all log are tested for

Integration Test

In agile qa developer works in a same time and everything works

How to write test?

SME-Subject Matter Expert

Brd,comes with different different test scenario based on BRD, and write test cases for automation .

Software testing consist of verification and validation testing?

What is User Acceptance Testing (UAT) ? UAT consist of Beta and Alpha.

What is Beta Testing?

Client side testing and fully performed by real user.

What is Alfa Testing?

Testing done by developer for real user prospective

What is stage /load Testing ?

How much user can get access to specific web site or application in a same time from user prospective.

Test Plan: http://softwaretestingfundamentals.com/test-plan/

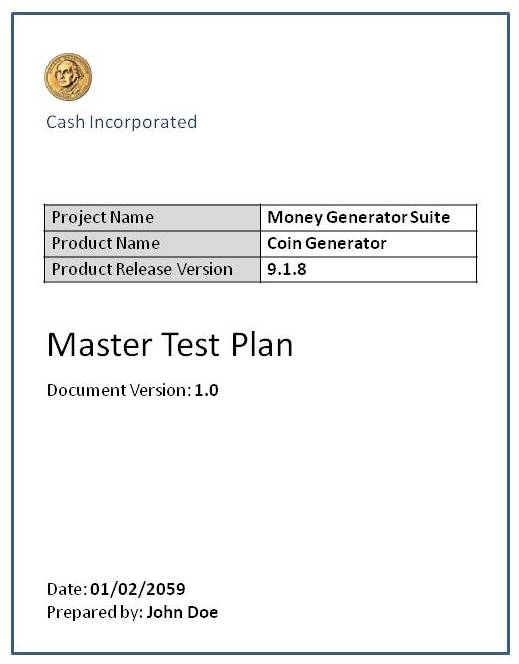
TEST PLAN Fundamentals

**TEST PLAN DEFINITION**

A Software Test Plan is a document describing the testing scope and activities. It is the basis for formally testing any software/product in a project.

ISTQB Definition

* **test plan**: A document describing the scope, approach, resources and schedule of intended test activities. It identifies amongst others test items, the features to be tested, the testing tasks, who will do each task, degree of tester independence, the test environment, the test design techniques and entry and exit criteria to be used, and the rationale for their choice, and any risks requiring contingency planning. It is a record of the test planning process.
* **master test plan**: A test plan that typically addresses multiple test levels.
* **phase test plan**: A test plan that typically addresses one test phase.



**TEST PLAN TYPES**

One can have the following types of test plans:

* **Master Test Plan:** A single high-level test plan for a project/product that unifies all other test plans.
* **Testing Level Specific Test Plans:**Plans for each level of testing.
  + Unit Test Plan
  + Integration Test Plan
  + System Test Plan
  + Acceptance Test Plan
* **Testing Type Specific Test Plans:**Plans for major types of testing like Performance Test Plan and Security Test Plan.

**TEST PLAN TEMPLATE**

*The format and content of a software test plan vary depending on the processes, standards, and test management tools being implemented. Nevertheless, the following format, which is based on IEEE standard for software test documentation, provides a summary of what a test plan can/should contain.*

**Test Plan Identifier:**

* Provide a unique identifier for the document. (Adhere to the Configuration Management System if you have one.)

**Introduction:**

* Provide an overview of the test plan.
* Specify the goals/objectives.
* Specify any constraints.

**References**:

* List the related documents, with links to them if available, including the following:
  + Project Plan
  + Configuration Management Plan

**Test Items:**

* List the test items (software/products) and their versions.

**Features to be Tested:**

* List the features of the software/product to be tested.
* Provide references to the Requirements and/or Design specifications of the features to be tested

**Features Not to Be Tested**:

* List the features of the software/product which will not be tested.
* Specify the reasons these features won’t be tested.

**Approach**:

* Mention the overall approach to testing.
* Specify the testing levels [if it’s a Master Test Plan], the testing types, and the testing methods [Manual/Automated; White Box/Black Box/Gray Box]

**Item Pass/Fail Criteria:**

* Specify the criteria that will be used to determine whether each test item (software/product) has passed or failed testing.

**Suspension Criteria and Resumption Requirements:**

* Specify criteria to be used to suspend the testing activity.
* Specify testing activities which must be redone when testing is resumed.

**Test Deliverables**:

* List test deliverables, and links to them if available, including the following:
  + Test Plan (this document itself)
  + Test Cases
  + Test Scripts
  + Defect/Enhancement Logs
  + Test Reports

**Test Environment:**

* Specify the properties of test environment: hardware, software, network etc.
* List any testing or related tools.

**Estimate:**

* Provide a summary of test estimates (cost or effort) and/or provide a link to the detailed estimation.

**Schedule:**

* Provide a summary of the schedule, specifying key test milestones, and/or provide a link to the detailed schedule.

**Staffing and Training Needs:**

* Specify staffing needs by role and required skills.
* Identify training that is necessary to provide those skills, if not already acquired.

**Responsibilities:**

* List the responsibilities of each team/role/individual.

**Risks:**

* List the risks that have been identified.
* Specify the mitigation plan and the contingency plan for each risk.

**Assumptions and Dependencies:**

* List the assumptions that have been made during the preparation of this plan.
* List the dependencies.

**Approvals:**

* Specify the names and roles of all persons who must approve the plan.
* Provide space for signatures and dates. (If the document is to be printed.)

**TEST PLAN GUIDELINES**

* Make the plan concise. Avoid redundancy and superfluousness. If you think you do not need a section that has been mentioned in the template above, go ahead and delete that section in your test plan.
* Be specific. For example, when you specify an operating system as a property of a test environment, mention the OS Edition/Version as well, not just the OS Name.
* Make use of lists and tables wherever possible. Avoid lengthy paragraphs.
* Have the test plan reviewed a number of times prior to baselining it or sending it for approval. The quality of your test plan speaks volumes about the quality of the testing you or your team is going to perform.
* Update the plan as and when necessary. An outdated and unused document stinks and is worse than not having the document in the first place.

Type of software testing: http://www.softwaretestinghelp.com/types-of-software-testing/

**Software Testing Types:**

**Black box testing** – Internal system design is not considered in this type of testing. Tests are based on requirements and functionality.

**White box testing** – This testing is based on knowledge of the internal logic of an application’s code. Also known as Glass box Testing. Internal software and code working should be known for this type of testing. Tests are based on coverage of code statements, branches, paths, conditions.

**Unit testing** – Testing of individual software components or modules. Typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. may require developing test driver modules or test harnesses.

**Incremental integration testing** – Bottom up approach for testing i.e continuous testing of an application as new functionality is added; Application functionality and modules should be independent enough to test separately. done by programmers or by testers.

**Integration testing** – Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.

**Functional testing** – This type of testing ignores the internal parts and focus on the output is as per requirement or not. Black-box type testing geared to functional requirements of an application.

**System testing** – Entire system is tested as per the requirements. Black-box type testing that is based on overall requirements specifications, covers all combined parts of a system.

**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 if appropriate.

**Sanity testing**– Testing to determine if a new software version is performing well enough to accept it for a major testing effort. If application is crashing for initial use then system is not stable enough for further testing and build or application is assigned to fix.

**Regression testing** – Testing the application as a whole for the modification in any module or functionality. Difficult to cover all the system in regression testing so typically automation tools are used for these testing types.

**Acceptance testing** -Normally this type of testing is done to verify if system meets the customer specified requirements. User or customer do this testing to determine whether to accept application.

**Load testing** – Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system’s response time degrades or fails.

------------

**Stress testing** – System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.

**Performance testing** – Term often used interchangeably with ‘stress’ and ‘load’ testing. To check whether system meets performance requirements. Used different performance and load tools to do this.

**Usability testing** – User-friendliness check. Application flow is tested, Can new user understand the application easily, Proper help documented whenever user stuck at any point. Basically system navigation is checked in this testing.

**Install/uninstall testing**– Tested for full, partial, or upgrade install/uninstall processes on different operating systems under different hardware, software environment.

**Recovery testing** – Testing how well a system recovers from crashes, hardware failures, or other catastrophic problems.

**Security testing** – Can system be penetrated by any hacking way. Testing how well the system protects against unauthorized internal or external access. Checked if system, database is safe from external attacks.

**Compatibility testing** – Testing how well software performs in a particular hardware/software/operating system/network environment and different combination s of above.

**Comparison testing** – Comparison of product strengths and weaknesses with previous versions or other similar products.

**Alpha testing** – In house virtual user environment can be created for this type of testing. Testing is done at the end of development. Still minor design changes may be made as a result of such testing.

**Beta testing** – Testing typically done by end-users or others. Final testing before releasing application for commercial purpose.