### How to Measure Availability and Uptime?

Keep track of system availability with a standard monitoring tool such as [Zabbix](https://www.zabbix.com/) or [Nagios](https://www.nagios.org/). If your systems are visible to the public internet, you can also use hosted platforms like [Pingdom](https://www.pingdom.com/) to measure system availability.

## **An uptime of 95% is usually sufficient for a QA or Staging environment.** If your development is limited to a few time zones, you can also further qualify this by only measuring availability during development hours. While your Production availability commitment is often higher that 99% or 99.5%, you don’t have to treat every QA outage as an emergency. But, your developers may have other opinions—95% uptime still allows for eight hours of downtime a week. You may want to aim higher.

So, how do you calculate the uptime percentage?

The way we calculate uptime is easy to understand: we take the number of seconds that your monitor was down (in a certain time frame), and divide this by the total number of seconds your monitor was being monitored during that time frame. As a result, we get the downtime percentage, which is then being subtracted from 100% to get the uptime percentage. Pretty simple, ain't it? Let's jump into an example.

Example

Let's say we monitored a website during 24 hours (which translates to 86,400 seconds), and in that timeframe the website went down for 10 minutes (600 seconds). To define the uptime and downtime percentages, we perform the following calculation:

**Total number of seconds your website was down:** 600 seconds.  
**Total number of seconds your website was monitored:** 86,400.  
We divide 600 by 86,400, which is 0.0069.  
In percentages, this is 0.69%. **This is the downtime percentage.**  
**The uptime percentage for this website would be:** 100% minus 0.69% is 99.31%.

**What Is Agile Methodology?**

Agile methodology is a practice that helps continuous iteration of development and testing in the SDLC process. Agile breaks the product into smaller builds.

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end.

In this methodology, development and testing activities are concurrent, unlike other software development methodologies. It also encourages teamwork and face-to-face communication. Business, stakeholders, and developers and clients must work together to develop a product.

**What Is Scrum?**

Scrum is an agile process that allows us to focus on delivering the business value in the shortest time. It rapidly and repeatedly inspects actual working software. It emphasizes accountability, teamwork, and iterative progress toward a well-defined goal.

The Scrum Framework usually deals with the fact that the requirements are likely to change or most of the time not known at the start of the project.

## Scrum is just one of the many iterative and incremental agile software development methods like XP,Kanban

## Agile Vs. Scrum

|  |  |
| --- | --- |
| **Agile** | **Scrum** |
| Agile is a development methodology based on **iterative and incremental approach**. | Scrum is one of the implementations of agile methodology. In which **incremental builds**are delivered to the customer in every two to three weeks' time. |
| Agile software development has been widely seen as highly suitedto environments which have **small but expert project** development team | Scrum is ideally used in the project where the requirement is **rapidly** changing. |
| In the Agile process, the **leadership**plays a vital role. | Scrum fosters a **self-organizing**, cross-functional team. |
| The biggest advantage of agile is its **flexibility**as it quickly reacts to changes. | Compared to Agile it is a more **rigid**method. So there is not much room for frequent changes. |
| Agile involves **collaborations** and face-to-face interactions between the members of various cross-functional teams. | In Scrum, collaboration is achieved in **daily stand up meeting** with a fixed role assigned to scrum master, product owner, and team members. |
| Agile can require l**ots of up-front development process** and organizational change. | Not **too many changes** needed while implementing scrum process. |
| The agile method **needs frequent delivery**to the end user for their feedback. | In the scrum, after each sprint, a **build is delivered** to the client for their feedback. |
| In this method, each step of development like requirements, analysis, design, are **continually monitored**during the lifecycle. | A demonstration of the functionality is provided at the end of every sprint. So that **regular feedback** can be taken before next sprint. |
| **Project head** takes cares of all the tasks in the agile method. | There is no team leader, so the **entire** **team addresses the issues** or problems. |
| The Agile method **encourages feedback**during the process from the end user. In this way, the end product will be more useful. | **Daily sprint meeting** is conducted to review and feedback to decide future progress of the project. |
| **Deliver and update**the software on a regular basis. | When the team is done with the **current sprint activities**, the next sprint can be planned. |
| Design and execution should be kept **simple**. | Design and execution can be **innovative and experimental**. |
| In the Agile method, the priority is always to satisfy the customer by providing **continuous delivery** of valuable software. | **Empirical Process Control** is a core philosophy of Scrum based process. |
| Working software is the most **elementary measure** of progress. | Working software is **not an elementary measure**. |
| It is best to have **face-to-face communication**, and techniques like these should be used to get as close to this goal as possible. | Scrum team focus to deliver **maximum business value**, from beginning early in the project and continuing throughout. |
| Following are Agile principles:  -Welcome changing requirements, even late in development. Agile processes allow change according to **customer's competitive advantage**.  -Business people and developers will **work daily** throughout the project.  -Attention to **technical excellence** and right design enhances agility  -Agile team, work on to become more effective, for that they **adjust its behavior** according to the project. | Following are scrum principles:  **-Self-organization:**This results in healthier shared ownership among the team members. It is also an innovative and creative environment which is conducive to growth.  **-Collaboration:**Collaboration is another essential principle which focuses collaborative work. 1. awareness 2. articulation, and 3. appropriation. It also considers project management as a shared value-creation process with teams working together to offer the highest value.  **-Time-boxing:**This principle defines how time is a limiting constraint in Scrum method. An important element of time-boxed elements are Daily Sprint planning and Review Meetings.  **-Iterative Development:** This principle emphasizes how to manage changes better and build products which satisfy customer needs. It also defines the organization's responsibilities regarding iterative development. |

### Conclusion:

* Agile methodology is a practice that helps continuous iteration of the development process.
* In the Agile method, the priority is always to satisfy the customer by providing continuous delivery of valuable software.
* Scrum is an agile process that allows us to focus on delivering the highest business value.
* The main benefit of Agile is its flexibility as it quickly reacts to changes.
* In the Agile process, the leadership plays a vital role.
* In Scrum, design and execution should be kept simple.
* In Scrum, daily sprint meeting is conducted to review and feedback to decide future progress of the project.
* In Scrum, when the team is done with the current sprint activity, the next sprint can be planned.

### 

## What Is Risk In Software Engineering?

**A risk is a potential problem.** It’s an activity or event that may compromise the success of a software development project. Risk is the possibility of suffering loss, and total risk exposure to a specific project will account for both the probability and the size of the potential loss.

Risk is made up of two parts: the probability of something going wrong, and the negative consequences if it does.

Risk can be hard to spot, however, let alone prepare for and manage. And, if you're hit by a consequence that you hadn't planned for, costs, time, and reputations could be on the line.

**When to Use Risk Analysis**

Risk analysis is useful in many situations:

* When you're planning projects, to help you anticipate and neutralize possible problems.
* When you're deciding whether or not to move forward with a project.
* When you're improving safety and managing potential risks in the workplace.
* When you're preparing for events such as equipment or technology failure, theft, staff sickness, or natural disasters.

**There are primarily 3 categories of Project Risks**

**1. Organizational Risks =** It is a risk related to your **human resource** or your Testing team. For example, in your project, lack of technically skilled members is a risk. Not having enough manpower to complete the project on time is another risk.

**2. Technical Risks -** Technical Risk is the probability of loss incurred during the execution of a technical process

Your task in this project is testing a banking website. You have to set up proper test environments which mirror real business environments. If the[Test Environment](https://www.guru99.com/test-environment-software-testing.html)is not setup properly, the product will be **not**be tested correctly and many **defects**will not be detected.

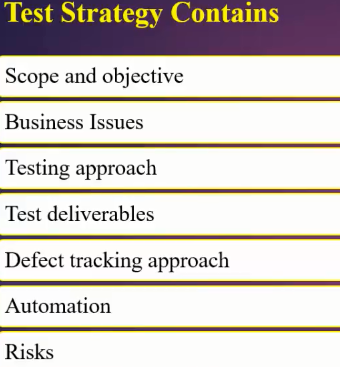
**3. Business Risks:**

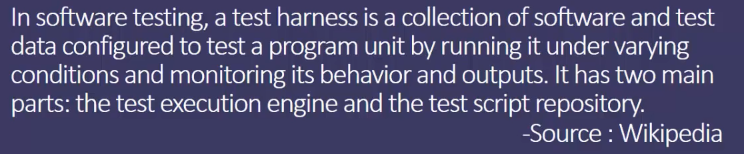
the risk involves an **external** entity. It is the risk which may come from your company, your customer but **not** from your project.

The following picture shows you an example of business risk.

## Product Risk

**Product risk**is the possibility that the system or software might fail to satisfy or fulfill the expectation of the customer, user, or stakeholder. This risk is related to the **functionality** of the product such as Performance Issues, Security Issues, Crash Scenarios, etc.





test execution engine- junit

test scrip repository- test cases,test class file

### Memory leaks:

**What is Memory Leak?**

In simple language a memory leak is loss of available memory when a program fails to return memory that it has obtained for temporary use.

A memory leak in Java (who would’ve thought heh?) can occur if you forget to close a resource, or a reference to an object is not released. e.g.

* File/Text buffers not closed. (as was in [my case](https://git.eclipse.org/r/#/c/31313/))
* Hash maps keeping references alive if equals() and hashcode() are not implemented, e.g

Memory Leakage Testing

If the program fails to release allocated memory that is called as Memory Leak!  
**Example:**  
If an object is allocated with 20 MB of memory after it's done with the task fails to release the allocated space, if a new instance is created which requires memory memory leak occurs.

If there is no memory space present for creating new objects in Heap. Java virtual machine throws OutOfMemoryError or Java.lang.OutOfMemoryErrorkm. Basically, it is nothing but a bug in program that prevents it from freeing up memory that is no longer needed. As a result the program utilizes more and more memory until it finally crashes because there is no more memory left.

As a smart QA, you will not want these issues to happen in your application. So before a customer reports it, QA should have the insight to catch the error beforehand. So, here comes in the picture of Memory Profiling. In computer programming memory profiling is a form of dynamic program analysis that measures the memory usage of the application.

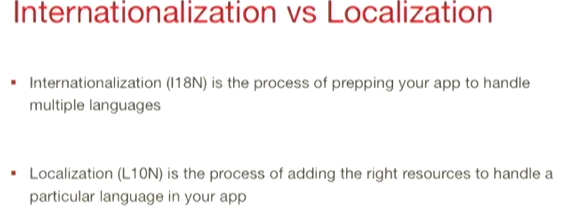
There are various tools available to do memory profiling. Ex: To profile java applications some of them are:

* Your Kit Java Profiler – <http://www.yourkit.com/> (Licensed)
* Eclipse Memory Analyzer (Eclipse MAT) – <http://www.eclipse.org/mat/> (Open Source)
* J Visual VM – <http://visualvm.java.net/> (GNU General Public License)
* An open-source **tool** that allows you to easily visualize information produced by **JVM**. .
* **JProfiler** – A thread, **memory**, and CPU profiling **tool** that can also be used to analyze **memory leaks** and other performance bottlenecks.

Stack is used for static memory allocation and Heap for dynamic memory allocation, both stored in the computer's RAM . Variables allocated on the stack are stored directly to the memory and access to this memory is very fast, and it's allocation is dealt with when the program is compiled.

The RAM is the physical memory of your computer. Heap memory is the (logical) memory reserved for the heap.  
So, only part of the RAM is used as heap memory and heap memory doesn't have to be fully loaded into RAM (e.g. part of it may be swapped to disc by the OS).

### Globalization testing



1,internationlization

2,localization

issues-right language displayed

alignment issue

right content right place

Localization- currency

date format

pincode format

image format

What needs to be Tested ?

* Sensitivity to the language vocabulary
* Date and time formatting
* Currency handling
* Paper sizes for printing
* Address and telephone number formatting
* Zip Code Format

Advantages of Globalization ;Testing

* It reduces overall testing and support costs
* It helps us to reduce time for testing which result faster time-to-market
* It is more flexible and product is easily scalable

### Globalization Testing Checklist

**Given below are the checklist of this Testing:**

* Ensure if the required installations are being done to set up test environment.
* Ensure if the database is Unicode compatible.
* Verify if there are no hard-coded strings in the code.
* Ensure if the required locale is installed on the client machine or not.
* Check the resource bundles for all the required language property files available in it.
* Verify if the UI of the application is displayed in the native language of the client’s locale.
* Verify if the default display language is English when no specific locale is selected or when the language properties file is not available in the resource bundle.
* Validate if the application can handle wide-ranging data including native language character set, ASCII characters, special characters, etc.
* Verify if the ordering of data on the UI is fine as per the client’s locale.
* Verify if the filtering and searching functionality is working fine as per the client’s locale.
* Verify if the correct date and time format are displayed across the application.
* Verify if the currencies are displayed in the correct format.
* Verify if the telephone numbers and pin codes are displayed in the correct format.
* Verify if the cursor is aligned to the correct side of the input fields based on the language script direction.
* Check if all the specified customer requirements are being tested?
* Check if all the input/output corresponding to each function is tested?
* Check if all the functionalities based on the native language inputs are being tested?
* Check if all the entire database requirements are being tested?
* Check if all the specified UIs are being tested?
* Verify that there are no characters overlapping on the screens.
* Verify that no junk characters are displayed on the screens.
* Verify that the graphics are appearing correctly in the UI.
* Verify if the user manuals/help files are displayed in the native language as per the client’s locale

1. Clipboard operations with multilingual text

2. Font independence

3. Handling buffer sizes f

or multibyte character set (MBCS) text





### Types of Mobile App Testing

### Web testing checklists

**1)**Functionality Testing  
**2)** Usability testing  
**3)**Interface testing- is the **testing** done on AUT which actually verifies whether the communication between two different software systems are done correctly.   
**4)** Compatibility testing  
**5)** Performance testing  
**6)**Security testing

7) Database Testing

# Let’s start with the Web Application Testing Checklist:

## [1] Functionality Testing:

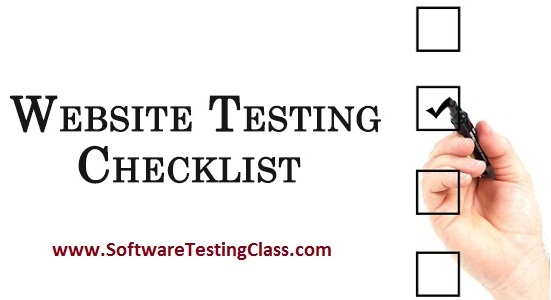
The following testing must be carried out while doing Website [Functionality testing](https://www.softwaretestingclass.com/functional-testing/):

### A) Validation testing:

» You should make sure that if the valid HTML is used for your website. To check this you can use [W3C validator](http://validator.w3.org/).

» In functionality testing the different fields used in the website should be validated like Textboxes, dropdowns, radio options, check boxes, Combo box inputs, links etc.

» Now a day’s most of the website preferred to use CSS means Cascading Style Sheet. In the market many CSS validator tools are available, one of the good tool is W3C CSS validator which will help you to validate the CSS used in your site user test.

[](https://www.softwaretestingclass.com/wp-content/uploads/2014/10/Website-Testing-Checklist.jpg)

### ****B) Links/URL Testing:****

This testing is very much interesting and can be helpful for SEO of you r page. Following types of testing should be carried out for Link or URL testing:

* **Internal links**: The link which are pointing to the pages of same websites. This testing make sure that the internal links are properly linked to expected pages of your websites link like Home page, Contact Us, About Us etc.
* **External links**: The link which are pointing to the pages of external websites. This testing makes sure that the internal links are properly linked to expected pages of external websites.
* **Email links**: Such link need to make sure that the if user clicks on the email link then default email client should open with To address should be pre-filled.
* **Broken links**: Broken links are also called as Dead links. Such links are not linked to any of the pages either internal or external pages of the website. Such link generated with the spelling mistakes in the link URL or linked page is removed or no more exists. To check broken link you can use online tools to validate the broken links in the website.

### ****C) Web Forms Testing:****

In Website Testing Checklist the web forms are the most commonly used in the websites, so it is one of the most important part of the website testing. Consider a scenario where user fills an enquiry form and click on Submit button, now what next or they just fill in the form and do nothing, the details do not get captured correctly and so are lost. While doing forms functional testing make sure that they should be consistent and should contain the required input and output controls. The data should be captured properly.

### ****D) Database Testing:****

Now days with the new technologies like android and smart phones computer applications are more complex.

If the front end is more complicated then the back ends are also convoluted. As a result, introduce more complex database schema to support such intricate computer applications. So it’s more important to validate the databases to make sure the quality and website able to handle the data processing effectively.

### ****E) Cookies Testing:****

A Cookie is information stored in text file on user’s hard drive (client side) by web server. This information is used later by web browser while accessing the website. Ideally the cookie is used to store the personalized user information or data in encrypted or secured manner. This is small size files which act as unique identifiers and allow websites to remember a particular user for a given time. These files are not harmful for users. Sometimes if the user’s personal information is stored in the cookie and if hackers stolen the cookie then hacker can get the confidential information which leads to security issues. That’s why the testing of Cookie is most important. There are two types of cookies Persistent Cookie and Non-Persistent Cookie.

### ****F) Testing of Error Messages****:

In the well developed website the error messages are very much helpful to guide users for success and erroneous conditions. While navigating through application if poorly designed error messages will easily misguide the end users. Many of the websites are used different interesting pages when [404 error](http://blog.hubspot.com/blog/tabid/6307/bid/33766/10-Clever-Website-Error-Messages-From-Creative-Companies.aspx) is displayed.

### ****G) Required field and optional field validation:****

The proper handling of required and optional fields should be efficiently handled. Ideally the application should not be submitted unless and until all required fields are filled properly. The required error message should be displayed when user proceed with not filling the mandatory fields. It should not restrict you for proceeding further if the optional fields are not filled.

**H) Client-side Testing:** This type of testing is subset of Security testing. In this testing need to check if the sensitive [modafinil online sale](http://onhealthy.net/product/provigil/) data is not stored in the temporary internet files or stored in encrypted format like passwords, credit card information, bank number etc.

## [2] Performance Testing:

Software [Performance testing](https://www.softwaretestingclass.com/what-is-performance-testing/) is type of testing perform to determine the performance of system to major the measure, validate or verify quality attributes of the system like responsiveness, Speed, Scalability, Stability under variety of load conditions. The main aim of Performance testing is not to find the defects in the application but to get rid of bottlenecks and find out the amount of load/stress the site is able to sustain. Performance testing is used to understand the scalability of application under test, response time means time required to send request from client machine to time required to reply by the server under different varity of load conditions which mimic the real world use.

Parameters

### Response time

Total time to send a request and get a response.

### Throughput-Throughput is a measure of how many units of information a system can process in a given amount of time

### CPU utilization

How much time the CPU needs to process requests.

### Memory utilization

How much memory is needed to process the request?

### Requests per second

How many requests are handled?

Performance testing can be applied to understand the website’s scalability, any loopholes in the load balancing and to test the response time between a request (from the client) and the reply (from the server) and the amount of load/stress the site is able to sustain.

# [4] [Compatibility Testing](https://www.softwaretestingclass.com/top-10-cross-browser-testing-tools-reduce-browser-compatibility-testing-effort/):

In Website Testing Checklist, the Compatibility Testing make sure that the web pages are properly rendering different browsers like IE8, IE9, IE10, IE11, Chrome, Firefox, Safari, Opera etc. Also verify if it is working properly on different Operating systems like Windows XP, Windows 7, Vista, Linux, Mac etc on different hardware configurations.

Here is the most typical compatibility tests should be considered while testing your application:

» Check on different browsers and its versions.

» Check on different Operating systems and its versions.

» Check on different hardware configurations

» Check on different network environments.

» Check on different screen resolutions

# [5] [Web Security Testing](https://www.softwaretestingclass.com/security-testing-approach-for-web-application-testing/):

Some of the major aspects of web security testing are:

» Penetration Testing- Penetration testing is the procedure of detecting vulnerabilities in the system, software, network or web applications. Penetration testing services find vulnerabilities that an attacker could exploit

» Password cracking

» Vulnerability

» URL manipulation

» SQL injection

» Cross Site Scripting (XSS)

 But what exactly is the role of QA in agile?

What can a tester do to help initiate a cooperative working relationship with the development team? Here are 6 things software testers should do when working with an agile scrum team:

agile scrum stories

1. Attend sprint-planning sessions.

2. Attend daily stand ups.

3. Don’t save all the testing for the end; test throughout the sprint.

4. Meet with developers for short hand-off demonstrations.

5. Attend sprint retrospectives.

6. Document test cases.

**107. What is verification and validation?**

Verification is a process of evaluating software  at development phase and to decide whether the product of a given  application satisfies the specified requirements. Validation is the process of evaluating software at the end of the development process and to check whether it meets the customer requirements.

**What are the tables in testplans?**

A test plan can be defined as a document describing the scope, approach, resources and schedule of testing activities and a test plan should cover the following details.

Test design, scope, test strategies , approach are various details that Test plan document consists of.

1. Test case identifier
2. Scope
3. Features to be tested
4. Features not to be tested
5. Test strategy & Test approach
6. Test deliverables
7. Responsibilities
8. Staffing and training
9. Risk and Contingencies

**131. What are the common mistakes which creates issues ?**

* Matching resources to wrong projects
* Test manager lack of skills
* Not listening to others
* Poor Scheduling
* Underestimating
* Ignoring the small problems
* Not following the process

**123. What is fault masking explain with example?**

1. When presence of one defect hides the presence of another defect in the system is known as fault masking.

**122. What are the categories of debugging?**

1. Categories for debugging
2. a)      Brute force debugging
3. b)      Backtracking
4. c)      Cause elimination
5. d)      Program slicing
6. e)      Fault tree analysis

Atest plan

It's a document mainly created by one representative of QA team, it's a continuous activity

**What a test plan should contain**

**What to consider for the Test Plan:**1.Objectives

2.Scope

3.Out of Scope

4.Scedules

5. Roles and Responsiobility(Communication plan and Team roaster, Roles expectations)

6.Test Startegy

7.Test Assumptions

7.1Availability of test environment  
7.2Use of test tools

7.3Availability of test team

8.Test Principles

9. Test Deliverables(Test plan, Functional Test Cases, Daily/weekly Status report, Logging defect Report, Closure Report)

10.Tools

11.Enviroments

12.Risk and Risk Management

13.Exit Criteria

**Test Strategy**

**The Test Strategy document is a static document meaning that it is not updated too often. It sets the standards for testing processes and activities and other documents such as the Test Plan draws its contents from those standards set in the Test Strategy Document.**

A Test Strategy document is a high level document and normally developed by project manager. This document defines “[Software Testing](http://www.testingexcellence.com/) Approach” to achieve testing objectives. The Test Strategy is normally derived from the **Business Requirement Specification document.**

Some companies include the “Test Approach” or “Strategy” inside the Test Plan, which is fine and it is usually the case for small projects. However, for larger projects, there is one Test Strategy document and different number of Test Plans for each phase or level of testing.

**Components of the Test Strategy document**

* Scope and Objectives
* Business issues
* Roles and responsibilities
* Communication and status reporting
* Test deliverability
* Industry standards to follow
* Test automation and tools
* Testing measurements and metrices
* Risks and mitigation
* Defect reporting and tracking
* Change and configuration management
* Training plan
* **Test Plan: the set of ideas that guide a test project**
* **Test Strategy: the set of ideas that guide test design**

**Functional Testing:** Testing the application against business requirements. Functional testing is done using the functional specifications provided by the client or by using the design specifications like use cases provided by the design team.

**Non-Functional Testing:** Testing the application against client's and performance requirement. Non-Functioning testing is done based on the requirements and test scenarios defined by the client.

**Explain your web application architecture?  
Web application is tested in 3 phases  
1. Web tier testing –> browser compatibility  
2. Middle tier testing –> functionality, security  
3. Data base tier testing –> database integrity, contents**

**Entry and Exit criteria**

Entry and Exit criteria are required to start and end the testing. It is must for the success of any project. If you do not know where

to start and where to finish then your goals are not clear. By defining exit and entry criteria you define your boundaries.

**Entry criteria should be**

* All developed code must be unit tested. Unit and Link Testing must be completed and signed off by development team.
* Functional and Business requirement should be cleared, confirmed and approved.
* Test plan, test cases reviewed and approved.
* Test environment/test ware gets prepared
* Test data should be available
* Availability of application
* QA/Tester gets significant knowledge of application.
* Resources should be ready

**Exit criteria will include what all required to end the test cycle as follows:**

* Deadlines (release deadlines, testing deadlines, etc.)
* Test cases completed with certain percentage passed
* Test budget depleted
* Coverage of code/functionality/requirements reaches a specified point
* All defects are fixed or closed
* Closure reports are singed off
* All the test cases have been executed and passed
* Beta or alpha testing period ends
* Budget allocated for testing is exhausted
* The risk in the project is under acceptable limit.

**What is webserver’?**

Web servers are [computers](http://www.webopedia.com/TERM/C/computer.html) that deliver (serves up) [Web pages](http://www.webopedia.com/TERM/W/web_page.html). Every Web server has an [IP address](http://www.webopedia.com/TERM/I/IP_address.html) and possibly a [domain name](http://www.webopedia.com/TERM/D/domain_name.html). For example, if you enter the [URL](http://www.webopedia.com/TERM/U/URL.html) http://www.pcwebopedia.com/index.html in your [browser](http://www.webopedia.com/TERM/B/browser.html), this sends a request to the Web server whose domain name ispcwebopedia.com. The server then fetches the page named index.html and sends it to your browser.

Any computer can be turned into a Web server by installing server [software](http://www.webopedia.com/TERM/S/software.html)and connecting the machine to the [Internet](http://www.webopedia.com/TERM/I/Internet.html). There are many Web server software applications, including public domain software from NCSA and Apache, and commercial packages from [Microsoft](http://www.webopedia.com/TERM/M/Microsoft.html), [Netscape](http://www.webopedia.com/TERM/N/Netscape.html) and others.

# Difference between web based application and client server application

1) Web Base Application is a 3 tier application. It consists of a Browser, Back end and a Server.

Client server Application is a 2 tier Application with a Front End and a Back end .

2) In the Web Base Application, tester test for the Script error like java script error VB script error etc, that shown at the page.

 In the Client server Application tester does not test for any script error.

3) In the Web Base Application once changes perform reflect at every machine so tester has less work for test.

Whereas in the Client server Application every time application need to be install hence ,it maybe possible that some machine has some problem for that Hardware testing as well as software testing is needed.

4) Changes can't reflect in this. used to install again and test whether application works or not.  
 Single user system. Difficult to test scripting problem in this.

**5}**)Security issues are very less in client server as there are Minimum users,but in Web server Security issues are Maximum as end users are more

6)Performance issues will not be taken in to consideration in client server, but consideration will be done web server applications.

7) In client server application we will not use Web servers, but in web server applications same will be used.

**Structural Testing**

It's a White Box Testing Technique. Since the testing is based on the internal structure of the program/code & hence it is called as Structural Testing.

**116. What is the difference between a test strategy and a test plan?**

**TEST PLAN:** IT IS PLAN FOR TESTING.IT DEFINES SCOPE, APPROACH, AND ENVIRONEMENT.   
  
**TEST STRATEGY:** A TEST STRATEGY IS NOT A DOCUMENT.IT IS A FRAMEWORK FOR MAKING DECISIONS ABOUT VALUE.

**Error:** a human action that produces an incorrect result.  
  
**Fault:** a manifestation of an error in software.  
  
**Failure:** a deviation of the software from its expected delivery or service.  
  
**Reliability:** the probability that the software will not cause the failure of the system for a specified time under specified conditions.

**63. What is exact difference between Debugging & Testing?**  
Testing is nothing but finding an error/problem and its done by testers where as debugging is nothing but finding the root cause for the error/problem and that is taken care by developers.

**6. What is Entry and Exit Criteria in Software Testing?**  
  
Entry Criteria is the process that must be present when a system begins, like,

* SRS (Software Requirement Specification)- whole system flow
* FRS (Functional Requirement Specification) - how system willl flow after implementing the requirement
* BRS- whole rquirement
* Usecase
* Test Case
* Test plan

Exit Criteria ensures whether testing is completed and the application is ready for release, like,

* Test Summary Report
* Metrics
* Defect Analysis report

**31. What is Defect Leakage?**  
Defect leakage occurs at the Customer or the End user side after the application delivery. After the release of the application to the client, if the end user gets any type of defects by using that application then it is called as Defect leakage. This Defect Leakage is also called as Bug Leakage.

**Traceability Matrix-**

this matrix defines the mapping between customer requirements and

prepared testcases by testengineers.this matrix is requirements

traceability matrix or requirements validation matrix.this is

used by testing team to verify how far the testcases prepared

have covered the requirements of the functionalities to be tested.

negative Test Cases-

Test cases for ATM Machine :

1. Machine is accepting ATM card

2. Machine is rejecting expired card

3. successful entry of PIN number

4. unsuccessful operation due to enter wrong PIN number 3 times

5. successful selection of language

6. Successful selection of account type

7. Unsuccessful operation due to invalid account type

8. successful selection of amount to be withdraw

9. successful withdrawal.

10. Expected message due to amount is greater than day limit

11. unsuccessful withdraw operation due to lack of money in ATM

12. Expected message due to amount to withdraw is greater than possible balance.

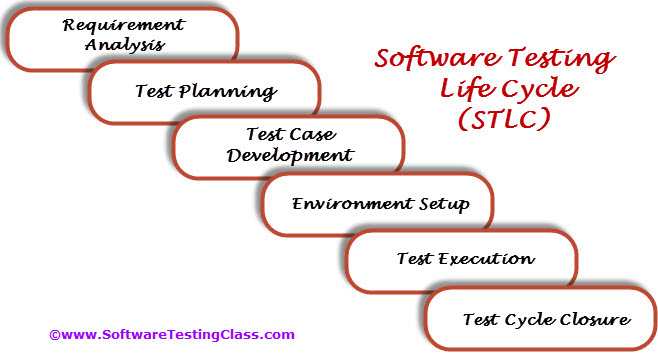
13. unsuccessful withdraw operation due to click cancel after insert card``

# Software Development Life Cycle phases?

1. Requirement gathering and analysis
2. Design
3. Implementation or coding
4. Testing
5. Deployment
6. Maintenance

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**STLC**



**Volume Testing** = Large amounts of data

**Load Testing** = Large amount of users

**Stress Testing** = Reducing the system resources and keeping the load as constant checking how does the application is behaving is called stress testing.

- Volume testing is a way to test functionality.

- Stress testing is a way to test reliability.

- testing is a way to test performance.

Load **Security Testing**  
**Ans.** Validating whether all security conditions are properly implemented in the software (or) not is called security testing.

**Risk Analysis in Software Testing**

Risk Analysis attempts to identify all the risks and then quantify the severity of the risks. A threat as we have seen is a possible damaging event. If it occurs, it exploits vulnerability in the security of a computer based system

**How to perform Risk Analysis during software testing**

When a test plan has been created, risks involved in testing the product are to be taken into consideration

along with the possibility of their occurrence and the damage they may cause along with solutions; if any. Detailed study of this is called Risk Analysis.

Some of the risks could be:

New Hardware.,New Technology.,New Automation Tool.Sequence of code delivery.Availability of application test resources.

**Risk Identification**

There can be different type of risks include as follows-

Software Risks: Knowledge of the most common risks associated with Software development, and the platform you are working on.

Business Risks: Most common risks associated with the business using the Software.

Testing Risks: Knowledge of the most common risks associated with Software Testing for the platform you are working on, tools being used, and test methods being applied.

Premature Release Risk: Ability to determine the risk associated with releasing unsatisfactory or untested Software Products.

Risk Methods: Strategies and approaches for identifying risks or problems associated with implementing and operating information technology, products and process; assessing their likelihood, and initiating strategies to test those risks

To overcome these risks, the following activities can be done.

Conducting Risk Assessment review meeting with the development team.

Profile for Risk coverage is created by mentioning the importance of each area.

Using maximum resources to work on High Risk areas like allocating more testers for High risk areas and minimum resources for Medium and Low risk areas.

Creation of Risk assessment database for future maintenance and management review.

Identify and describe the risk magnitude indicators: High, Medium and Low

**Agile Meeting**

Sprint Planning Meeting

Daily Scrum Meeting

sprint Review Meeting

Retrospective Meeting

Backlog Refinement Meeting

**Q.** What is Process?  
**Ans.** A process is set of a practices performed to achieve a given purpose; it may include tools, methods, materials and or people.

Verification process are done in Static Testing- Verification is intended to check that a product, service, or system (or portion thereof, or set thereof) meets a set of design specifications.

Validation process are done in Dynamic Testing

**Methods of Verification**

1. Walkthrough

2. Inspection

3. Review

**Methods of Validation**

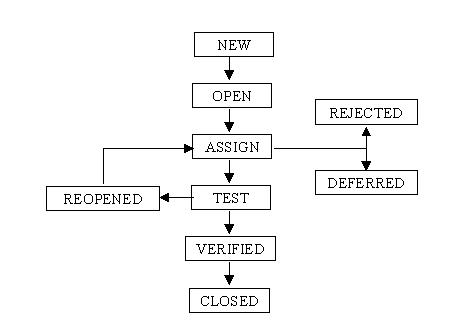
1. Testing

2. End Users

**Walkthrough-walkthrough** or **walk-through** is a form of [software peer review](http://en.wikipedia.org/wiki/Software_peer_review) "in which a designer or programmer leads members of the development team and other interested parties through a software product, and the participants ask questions and make comments about possible errors, violation of development standards, and other problems"

Static analysis is performed in a non-runtime environment. Typically a static analysis tool will inspect program code for all possible run-time behaviors and seek out coding flaws, back doors, and potentially malicious code. Dynamic analysis adopts the opposite approach and is executed while a program is in operation. A dynamic test will monitor system memory, functional behavior, response time, and overall performance of the system. This method is not wholly dissimilar to the manner in which a malicious third party may interact with an application. Having originated and evolved separately, static and dynamic analysis have, at times, been mistakenly viewed in opposition. There are, however, a number of strengths and weaknesses associated with both approaches to consider.

**Bug Life Cycle-** In software development process, the bug has alife cycle. The bug should go through the life cycle to be closed. A specific life cycle ensures that the process is standardized. The bug attains different states in the life cycle. The life cycle of the bug can be shown diagrammatically as follows:



**Desktop application** runs on personal computers and work stations, so when you test the desktop application you are focusing on a specific environment. You will test complete application broadly in categories like GUI, functionality, Load, and backend i.e DB.

In **client server application**you have two different components to test. Application is loaded on server machine while the application exe on every client machine. You will test broadly in categories like, GUI on both sides, functionality, Load, client-server interaction, backend. This environment is mostly used in Intranet networks. You are aware of number of clients and servers and their locations in the test scenario.

**Web application** is a bit different and complex to test as tester don’t have that much control over the application. Application is loaded on the server whose location may or may not be known and no exe is installed on the client machine, you have to test it on different web browsers. Web applications are supposed to be tested on different browsers and OS platforms so broadly Web application is tested mainly for browser compatibility and operating system compatibility, error handling, static pages, backend testing and load testing.

**What is Verification?**  
The standard definition of Verification goes like this: "Are we building the product RIGHT?" i.e. Verification is a process that makes it sure that the software product is developed the right way. The software should confirm to its predefined specifications, as the product development goes through different stages, an analysis is done to ensure that all required specifications are met.  
  
During the Verification, the work product (the ready part of the Software being developed and various documentations) is reviewed/examined personally by one ore more persons in order to find and point out the defects in it. This process helps in prevention of potential bugs, which may cause in failure of the project.

**Few terms involved in Verification:**  
Inspection:  
Inspection involves a team of about 3-6 people, led by a leader, which formally reviews the documents and work product during various phases of the product development life cycle. The work product and related documents are presented in front of the inspection team, the member of which carry different interpretations of the presentation. The bugs that are detected during the inspection are communicated to the next level in order to take care of them.  
  
**Walkthroughs:**  
Walkthrough can be considered same as inspection without formal preparation (of any presentation or documentations). During the walkthrough meeting, the presenter/author introduces the material to all the participants in order to make them familiar with it. Even when the walkthroughs can help in finding potential bugs, they are used for knowledge sharing or communication purpose.

**Verification**: The process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. [IEEE-STD-610].

**Validation**: The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. [IEEE-STD-610]

|  |  |
| --- | --- |
| **Validation** | **Verification** |
| Am I building the right product | Am I building the product right |
| Determining if the system complies with the requirements and performs functions for which it is intended and meets the organization’s goals and user needs. It is traditional and is performed at the end of the project. | The review of interim work steps and interim deliverables during a project to ensure they are acceptable. To determine if the system is consistent, adheres to standards, uses reliable techniques and prudent practices, and performs the selected functions in the correct manner. |
| Am I accessing the right data (in terms of the data required to satisfy the requirement) | Am I accessing the data right (in the right place; in the right way). |
| High level activity | Low level activity |
| Performed after a work product is produced against established criteria ensuring that the product integrates correctly into the environment | Performed during development on key artifacts, like walkthroughs, reviews and inspections, mentor feedback, training, checklists and standards |
| Determination of correctness of the final software product by a development project with respect to the user needs and requirements | Demonstration of consistency, completeness, and correctness of the software at each stage and between each stage of the development life cycle. |

**Defect and Issue**

Risk is concerned about future happenings which could have a negative or positive impact with an uncertain probability   
  
Issue is a known fact with a certainty   
  
Risk is a future event that may have an impact on triple constraint (Budget, scope and schedule).

**What is a cookie?**

Cookies are usually small text files, given ID tags that are stored on your computer's browser directory or program data subfolders. Cookies are created when you use your browser to visit a website that uses cookies to keep track of your movements within the site, help you resume where you left off, remember your registered login, theme selection, preferences, and other customization functions. The website stores a corresponding file(with same ID tag)to

<http://www.allaboutcookies.org/cookies/what-information-in-cookie.html>

**Are all cookies the same?**

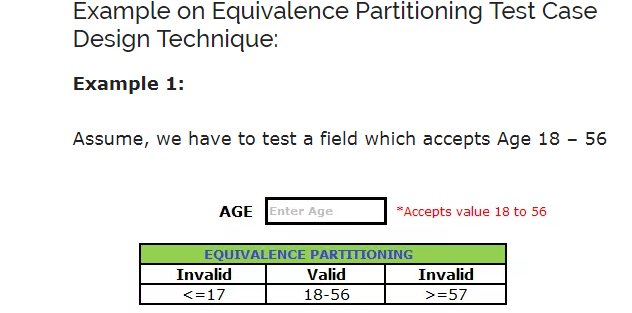
**There are two different types of cookies:**

**Session cookies** - these are temporary and are erased when you close your [browser](http://www.allaboutcookies.org/faqs/browser.html) at the end of your surfing session. The next time you visit that particular site it will not recognise you and will treat you as a completely new visitor as there is nothing in your browser to let the site know that you have visited before ([more on session cookies](http://www.allaboutcookies.org/cookies/session-cookies-used-for.html)).

**Persistent cookies** - these remain on your hard drive until you erase them or they expire. How long a cookie remains on your browser depends on how long the visited website has programmed the cookie to last ([more on persistent cookies](http://www.allaboutcookies.org/cookies/persistent-cookies-used-for.html)).

Agile model

1: **Test Case writing techniques.**   
Boundary value analysis.   
Equivalence class partitioning.



### Statement Testing & Coverage

This technique involves execution of all the executable statements in the source code at least once.  The percentage of the executable statements is calculated as per the given requirement. This is the least preferred metric for checking test coverage

* Error Guessing
* In this technique, the testers anticipate errors based on their experience, availability of data and their knowledge of product failure.  Error guessing is dependent on the skills, intuition, and experience of the testers.

Decision Table **Testing**.

* State Transition Diagrams.
* Use **Case** **Testing**.

### [Exploratory Testing](https://reqtest.com/testing-blog/3-simple-tricks-to-make-exploratory-testing-more-efficient/)

This technique is used to test the application without any formal documentation.  There is minimum time available for testing and maximum for test execution.  In this, the test design and test execution are performed concurrently.

<http://www.softwaretestinghelp.com/white-box-testing/>

**White Box Testing is coverage of the specification in the code.**

**Code coverage:**

**Segment coverage:**  
Ensure that each code statement is executed once.

**Branch Coverage or Node Testing:**  
Coverage of each code branch in from all possible was.

**Compound Condition Coverage:**  
For multiple condition test each condition with multiple paths and combination of different path to reach that condition.

**Basis Path Testing:**  
Each independent path in the code is taken for testing.

**Loop Testing:**  
These strategies relate to testing single loops, concatenated loops, and nested loops. Independent and dependent code loops and values are tested by this approach.

<http://www.codeproject.com/KB/testing/white-box-testing.aspx>

Web Services offer many benefits over other types of distributed computing architectures.

* Interoperability - This is the most important benefit of Web Services. Web Services typically work outside of private networks, offering developers a non-proprietary route to their solutions. Services developed are likely, therefore, to have a longer life-span, offering better return on investment of the developed service. Web Services also let developers use their preferred programming languages. In addition, thanks to the use of standards-based communications methods, Web Services are virtually platform-independent.
* Usability - Web Services allow the business logic of many different systems to be exposed over the Web. This gives your applications the freedom to chose the Web Services that they need. Instead of re-inventing the wheel for each client, you need only include additional application-specific business logic on the client-side. This allows you to develop services and/or client-side code using the languages and tools that you want.
* Reusability - Web Services provide not a component-based model of application development, but the closest thing possible to zero-coding deployment of such services. This makes it easy to reuse Web Service components as appropriate in other services. It also makes it easy to deploy legacy code as a Web Service.
* Deployability - Web Services are deployed over standard Internet technologies. This makes it possible to deploy Web Services even over the fire wall to servers running on the Internet on the other side of the globe. Also thanks to the use of proven community standards, underlying security (such as SSL) is already built-in.

Question

[[Sign in to vote](https://login.live.com/login.srf?wa=wsignin1.0&wtrealm=social.msdn.microsoft.com&wreply=https://social.msdn.microsoft.com/Forums/en-US/435f43a9-ee17-4700-8c9d-d9c3ba57b5ef/advantages-disadvantages-of-webservices?forum=asmxandxml&prof=required&stoAI=10&wp=MBI_FED_SSL&wlcxt=microsoft$microsoft$microsoft)](https://login.live.com/login.srf?wa=wsignin1.0&wtrealm=social.msdn.microsoft.com&wreply=https%3a%2f%2fsocial.msdn.microsoft.com%2fForums%2fen-US%2f435f43a9-ee17-4700-8c9d-d9c3ba57b5ef%2fadvantages-disadvantages-of-webservices%3fforum%3dasmxandxml%26prof%3drequired%26stoAI%3d10&wp=MBI_FED_SSL&wlcxt=microsoft%24microsoft%24microsoft" \o "Vote as helpful)

[2](https://login.live.com/login.srf?wa=wsignin1.0&wtrealm=social.msdn.microsoft.com&wreply=https%3a%2f%2fsocial.msdn.microsoft.com%2fForums%2fen-US%2f435f43a9-ee17-4700-8c9d-d9c3ba57b5ef%2fadvantages-disadvantages-of-webservices%3fforum%3dasmxandxml%26prof%3drequired%26stoAI%3d10&wp=MBI_FED_SSL&wlcxt=microsoft%24microsoft%24microsoft" \o "Vote as helpful)

Here are some general pros and cons..

**Advantages**

Web Services offer many benefits over other types of distributed computing architectures.

* Interoperability - This is the most important benefit of Web Services. Web Services typically work outside of private networks, offering developers a non-proprietary route to their solutions. Services developed are likely, therefore, to have a longer life-span, offering better return on investment of the developed service. Web Services also let developers use their preferred programming languages. In addition, thanks to the use of standards-based communications methods, Web Services are virtually platform-independent.
* Usability - Web Services allow the business logic of many different systems to be exposed over the Web. This gives your applications the freedom to chose the Web Services that they need. Instead of re-inventing the wheel for each client, you need only include additional application-specific business logic on the client-side. This allows you to develop services and/or client-side code using the languages and tools that you want.
* Reusability - Web Services provide not a component-based model of application development, but the closest thing possible to zero-coding deployment of such services. This makes it easy to reuse Web Service components as appropriate in other services. It also makes it easy to deploy legacy code as a Web Service.
* Deplorability - Web Services are deployed over standard Internet technologies. This makes it possible to deploy Web Services even over the fire wall to servers running on the Internet on the other side of the globe. Also thanks to the use of proven community standards, underlying security (such as SSL) is already built-in.

**Disadvantages**

* Although the simplicity of Web services is an advantage in some respects, it can also be a hindrance. Web services use plain text protocols that use a fairly verbose method to identify data. This means that Web service requests are larger than requests encoded with a binary protocol. The extra size is really only an issue over low-speed connections, or over extremely busy connections.
* Although HTTP and HTTPS (the core Web protocols) are simple, they weren't really meant for long-term sessions. Typically, a browser makes an HTTP connection, requests a Web page and maybe some images, and then disconnects. In a typical CORBA or RMI environment, a client connects to the server and might stay connected for an extended period of time. The server may periodically send data back to the client. This kind of interaction is difficult with Web services, and you need to do a little extra work to make up for what HTTP doesn't do for you.
* The problem with HTTP and HTTPS when it comes to Web services is that these protocols are "stateless"—the interaction between the server and client is typically brief and when there is no data being exchanged, the server and client have no knowledge of each other. More specifically, if a client makes a request to the server, receives some information, and then immediately crashes due to a power outage, the server never knows that the client is no longer active. The server needs a way to keep track of what a client is doing and also todetermine when a client is no longer active.
* Typically, a server sends some kind of session identification to the client when the client first accesses the server. The client then uses this identification when it makes further requests to the server. This enables the server to recall any information it has about the client. A server must usually rely on a timeout mechanism to determine that a client is no longer active. If a server doesn't receive a request from a client after a predetermined amount of time, it assumes that the client is inactive and removes any client information it was keeping. This extra overhead means more work for Web service developers.

# Integration Testing and Types-

Here are three different types of **integration testing** approach in software testing.

1. Big Bang

2. Top down

3. Bottom up

**1. Big Bang**

Big Bang **Integration testing** approach used to find the bugs when all the developed modules are interacted with each other and create a complete software system then its produced result satisfying with original requirement.

**2. Top down**

In Top down **integrated testing** approach, all Top level integrated modules are tested first and its sub modules tested from top to down step by step.

**3. Bottom up**

In Bottom up **integrated testing** approach, all bottom (Sub Modules) level integrated sub modules are tested first and its main modules tested from bottom to up step by step.

**Drivers and Stubs**

*While doing an Integration , If we dont have all the modules get ready and Need to test a particualr module which is ready then We Use Stubs and Drivers.  
  
Stubs and drivers used in Integration testing for a Top Down Integration testing and Botton Up Integration Testing.  
  
For EX : If we have Modules x,y,z . X module is ready and Need to Test it , But it calls functions from y and z.(Which is not ready)To test at a particular module we write a Small Dummy piece a code which Simulates Y and Z Whch will return values for X, These peace of Dummy code is Called Stubs in a Top Down Integration  
  
So Stubs are called Functions in Top Down Integration.  
  
Similar to the above ex: If we have Y and Z modules get ready and x module is not ready, and we need to test y and z modules Which return values from X,So to get the values from X We write a Small Pice of Dummy code for x which returns values for Y and Z,So these piece of code is called Drivers in Botton Up Integration  
  
So Drivers are calling Functions in Bottom Up Integration. '*

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Software Estimation Techniques - Common Test Estimation Techniques used in SDLC

<http://www.softwaretestingclass.com/software-estimation-techniques/>

# Software Estimation Techniques

There are different Software **Testing Estimation** Techniques which can be used for estimating a task.

**1) Delphi Technique**  
**2) Work Breakdown Structure (WBS)**  
**3) Three Point Estimation**  
**4) Functional Point Method**

## 1) Delphi Technique:

Delphi technique – This is one of the widely used software testing estimation technique.

In the Delphi Method is based on surveys and basically collects the information from participants who are experts. In this estimation technique each task is assigned to each team member & over multiple rounds surveys are conduct unless & until a final estimation of task is not finalized. In each round the thought about task are gathered & feedback is provided. By using this method, you can get quantitative and qualitative results.

In overall techniques this technique gives good confidence in the estimation. This technique can be used with the combination of the other techniques.

## 2) Work Breakdown Structure (WBS):

A big project is made manageable by first breaking it down into individual components in a hierarchical structure, known as the **Work breakdown structure**, or the WBS.

The WBS helps to project manager and the team to create the task scheduling, detailed cost estimation of the project. By using the WBS motions, the project manager and team will have a pretty good idea whether or not they’ve captured all the necessary tasks, based on the project requirements, which are going to need to happen to get the job done.

In this technique the complex project is divided into smaller pieces. The modules are divided into smaller sub-modules. Each sub-modules are further divided into functionality. And each functionality can be divided into sub-functionalities. After breakdown  the work all functionality should review to check whether each & every functionality is covered in the WBS.

Using this you can easily figure out the what all task needs to completed & they are breakdown into details task so estimation to details task would be more easier than estimating overall Complex project at one shot.

Work Breakdown Structure has four key benefits:

* **Work Breakdown Structure forces the team to create detailed steps:**  
  In The WBS all steps required to build or deliver the service are divided into detailed task by Project manager, Team and customer. It helps to raise the critical issues early on, narrow down the scope of the project and create a dialogue which will help make clear bring out assumptions, ambiguities, narrow the scope of the project, and raise critical issues early on.
* **Work Breakdown Structure help to improve the schedule and budget.**WBS enables you to make an effective schedule and good budget plans. As all tasks are already available so it helps in generating a meaningful schedule and makes scheming a reliable budget easier.
* **Work Breakdown Structure creates accountability**The level of details task breakdown helps to assign particular module task to individual, which makes easier to hold person accountable to complete the task. Also the detailed task in WBS, people cannot allow hiding under the “cover of broadness.”
* **Work Breakdown Structure creation breeds commitment**The process of developing and completing a WBS breed excitement and commitment. Although the project manager will often develop the high-level WBS, he will seek the participation of his core team to flesh out the extreme detail of the WBS. This participation will spark involvement in the project.

## 3) Three Point Estimation:

Three point estimation is the estimation method is based on statistical data. It is very much similar to WBS technique, task is broken down into subtasks & three types of estimation are done on this sub pieces.

Optimistic Estimate (Best case scenario in which nothing goes wrong and all conditions are optimal.) = A

Most Likely Estimate (most likely duration and there may be some problem but most of the things will go right.) = M

Pessimistic Estimate (worst case scenario which everything goes wrong.) = B

Formula to find Value for Estimate (E) = A + (4\*M) + B / 6

**Test Estimate = P + (4\*N) + E / 6**

Whereas

* P = Positive Scenarios
* N = Negative Scenarios
* E = Exceptional Scenarios

**test estimation**

Web definitions

1. The calculated approximation of a result (e.g. effort spent, completion date, costs involved, number of test cases, etc.) which is usable even if input data may be incomplete, uncertain, or noisy.

Standard Deviation (SD) = = (B – A)/6

Now a days, planning poker and Delphi estimates are most popular testingtest estimation techniques.

## 4) Functional Point Method:

Functional Point is measured from a functional, or user, point of view.

It is independent of computer language, capability, technology or development methodology of the team. It is based on available documents like SRS, Design etc.

In this FP technique we have to give weightage to each functional point. Prior to start actual estimating tasks functional points are divided into three groups like Complex, Medium & Simple. Based on similar projects & Organization standards we have to define estimate per function points.

Total Effort Estimate = Total Function Points \* Estimate defined per Functional Point

Let’s take a simple example to get clearer:

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

### Difference between tomcat and apache?

1. Both are web server used for hosting java Applications

2. If you are going to serve only static(such as HTML) pages then Apache webserver would suffice but if the pages being (dynamically) served are in JSP/Servlet then you need a container that can interpret them, for which we use Tomcat.

3. My question is If we can use Tomcat as a standalone server than why we will ever need to combine apache + tomcat as a server

It used to be that Tomcat/Java was so much slower than Apache HTTPD when it came to serving static files that you had to combine them to get a production ready application. This is no longer the case. Most often, the gains you get by combining the two (if any) are not worth the hassle of administering both + the connector.

4. Apache Tomcat is used to deploy your Java Servlets and JSPs. So in your Java project you can build your WAR (short for Web ARchive) file, and just drop it in the deploy directory in Tomcat.

So basically Apache is an HTTP Server, serving HTTP. Tomcat is a Servlet and JSP Server serving Java technologies.

First of all, Tomcat is a Web server and a servlet container and Apache is a Web server. Tomcat is often integrated with Apache to enhance Apache with servlet capabilities and to capitalize on Apache's optimized and robust static-page delivery mechanisms. IIS (Internet Information Services) is Microsoft's Web server.

**an apache server is an http server which can serve any simple http requests, where tomcat server is actually a servlet container which can serve java servlet requests.**

**Web server [apache] process web client (web browsers) requests and forwards it to servlet container [tomcat] and container process the requests and sends response which gets forwarded by web server to the web client [browser].**

**HTTP vs HTTPS**

HTTPS (Hypertext Transfer Protocol over Secure Socket Layer, or HTTP over SSL) is a web protocol developed by Netscape.

One can say: **HTTPS = HTTP + SSL**

HTTPS uses Secure Socket Layer (SSL) as a sublayer under its regular HTTP application layering

<http://theprofessionalspoint.blogspot.in/2012/04/http-vs-https-similarities-and.html>

<http://www.slideshare.net/simplyharshad/http-vs-https>

|  |  |
| --- | --- |
| **HTTP** | **HTTPS** |
|  |  |
| It uses port 80 for communication | It uses port 443 for communication |
|  |  |
| Operates at Application Layer | Operates at Transport Layer |
|  |  |
| No certificates required | Certificates required |

**Payment Card Testing**

**4 Types of card**

1. Valid card

2. Expired card

3. Blocked card

4. Invalid Card

Test Cases-1) Test with valid card number with valid expiry but invalid CVV.

2) Test with valid card number with invalid expiry but valid CVV.

3) Test with valid card number with invalid expiry and CVV.

4) Test with valid details of expired/blocked card.

5) Test with valid card number, CVV and expiry date.

6) Test for negative transaction.

* 1. Check what happens if session goes time out during the payment process
  2. Test after successful payment all the necessary data retrieved to our application or not
  3. Check what happens if payment gateway goes down during the payment process

10)Check what happens if payment process went successful but do not return to our application

11)Check what happens if payment process is fail

12)Verify DB entries for the transaction whether they store credit card details and all or not

**For Security testing:-**

 1) Try out "cookie poisoning" technique in payment gateway.

 2) Test session timeout scenario.

 3) Test try to access the page with using back button of the browser to check that session is still active on successful confirmation of payments.

 4) Try to test payment gateway using SQL injections tricks

Security testing

**2. URL manipulation through HTTP GET methods:**

**3.** **SQL Injection:**

**4. Cross Site Scripting (XSS):**

Attacker can use this method to execute malicious script or URL on victim’s browser. Using cross-site scripting, attacker can use scripts like JavaScript to steal user cookies and information stored in the cookies.

5.Cookise posioning/altering

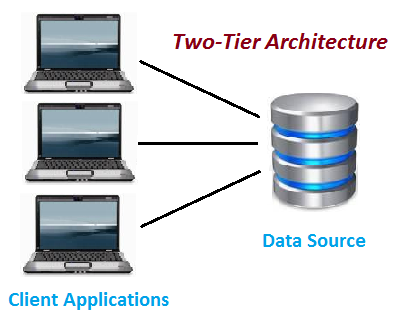
# cookies poisoning

On the Web, cookie poisoning is the modification of a [cookie](http://searchsoftwarequality.techtarget.com/definition/cookie) (personal information in a Web user's computer) by an attacker to gain unauthorized information about the user for purposes such as [identity theft](http://searchsecurity.techtarget.com/definition/identity-theft). The attacker may use the information to open new accounts or to gain access to the user's existing accounts.

Service oriented architecture (SOA) is an architecture where independent systems and applications communicate with each other by exposing and using services. Services are defined using open standards, making inter-communication much easier to implement, and less dependent on proprietary communication protocols.

# Two-Tier Architecture:

The two-tier is based on Client Server architecture. The two-tier architecture is like client server application. The direct communication takes place between client and server. There is no intermediate between client and server. Because of tight coupling a 2 tiered application will run faster.

[](http://www.softwaretestingclass.com/what-is-difference-between-two-tier-and-three-tier-architecture/two-tier-architecture/)

Two-Tier Architecture

The Two-tier architecture is divided into two parts:

**1) Client Application (Client Tier)  
2) Database (Data Tier)**

On client application side the code is written for saving the data in the SQL server database. Client sends the request to server and it process the

request & send back with data. **The main problem of two tier architecture is the server cannot respond multiple request same time,** as a result it cause a data integrity issue.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

When I think of 2-tier architecture, I vision 2 computers talking to each other, with one computer being the client and the other being the Server. Now, with this in mind, is it possible to see this web application as a 2-tier architecture.

Client Machine-> Web Browser

Server Machine -> IIS web server hosting the website, the machine possibly hosts the database also.

Is this a 2-tier web application architecture?.

Now, during web application development using Visual Studio, if I am debugging the website using local IIS and database, will this constitute as a 1-tier architecture?.

Yes, you're right about the 2-tier architecture.  If you have a basic web site and each web page makes requests to a database server directly that's a 2-tier.  3-tier would be if you expanded this and had another server components such as a class library that included business logic (BLL) or a dedicated DAL (data access layer) such as a set of API's to talk to the database that run on the server side.

Even if you moved it to one machine though, it would still be 2-tier because you would still have a web server (the web server included in Visual Studio) and the database server.

The only way it would be 1-tier would be something like a windows application that talks to a local access database or something like that.

**N-Tier Architecture** is a physical structuring, while **a N-Layer Architecture** is a logical structuring.

Question

Sam, here it is:N-tier and 3-tier are architectural deployment styles that describe the separation of functionality into segments in much the same way as the layered style, but with each segment being a tier that can be located on a physically separate computer. They evolved through the component-oriented approach, generally using platform specific methods for communication instead of a message-based approach.

N-tier application architecture is characterized by the functional decomposition of applications, service components, and their distributed deployment, providing improved scalability, availability, manageability, and resource utilization. Each tier is completely independent from all other tiers, except for those immediately above and below it. The nth tier only has to know how to handle a request from the n+1th tier, how to forward that request on to the n-1th tier (if there is one), and how to handle the results of the request. Communication between tiers is typically asynchronous in order to support better scalability.

N-tier architectures usually have at least three separate logical parts, each located on a separate physical server. Each part is responsible for specific functionality. When using a layered design approach, a layer is deployed on a tier if more than one service or application is dependent on the functionality exposed by the layer.

An example of the N-tier/3-tier architectural style is a typical financial Web application where security is important. The business layer must be deployed behind a firewall, which forces the deployment of the presentation layer on a separate tier in the perimeter network. Another example is a typical rich client connected application, where the presentation layer is deployed on client machines and the business layer and data access layer are deployed on one or more server tiers.

**The main benefits of the N-tier/3-tier architectural style are:**

* **Maintainability**. Because each tier is independent of the other tiers, updates or changes can be carried out without affecting the application as a whole.
* **Scalability**. Because tiers are based on the deployment of layers, scaling out an application is reasonably straightforward.
* **Flexibility**. Because each tier can be managed or scaled independently, flexibility is increased.
* **Availability**. Applications can exploit the modular architecture of enabling systems using easily scalable components, which increases availability.
* High performance, lightweight persistent objects
* Scalability – Each tier can scale horizontally
* Performance – Because the Presentation tier can cache requests, network utilization is minimized, and the load is reduced on the Application and Data tiers.
* High degree of flexibility in deployment platform and configuration
* Better Re-use
* Improve Data Integrity
* Improved Security – Client is not direct access to database.
* Easy to maintain and modification is bit easy, won’t affect other modules
* In three tier architecture application performance is good.

**Mob Testing Question**

* What is the extension of application developed on different platforms OS : Answer is \*.ipa for iOS,\*.apk for Android, Blackberry \*.jad, for windows  phone \*.exe and for palm \*.prc file.
* How can a testable build installed on device: For iOS platform iTunes is required, for Android you need android SDK &; adb install command is used to install build on device.
* What is latest OS version: 7.1.1 for iOS , Blackberry has OS 10.2.1, Android has 4.4.2 (Kit Kat){at the time of writing blog}
* 14) Types of mobile applications?

**Mobile applications can be broadly categorized into three categories i.e, Native app, Web app and Hybrid App.**  
 **Native App:**Native app are developed specifically for one platform, which is coded with a specific programming language (like Objective C for iOS, Java for Android) and installed directly onto the device and can take full advantage of all the device features — they can use the camera, the GPS, the accelerometer, the compass, the phone book etc. Native apps can use the device’s notification system and can work offline. Native apps are installed through an application store (such as Google Play or Apple’s App Store).Native mobile apps provide fast performance and a high degree of reliability. Example of native app: Temple Run, Candy Crush etc.   
   
**Web App:**Web applications are mobile web portals that are designed, customized and hosted specifically for mobiles. They are accessed through the mobile device’s web browser using a URL. Web apps became really popular when HTML5 came around and people realized that they can obtain native-like functionality in the browser. Mobile web applications cannot use device functionality. Example of web app: google.com, m.snapdeal.com, m.yahoo.com etc.  
 **Hybrid App:** Hybrid Apps are a way to expose content from existing websites in App format. They can be well described as a mixture of Web App and Native App.

* 12) How to check CPU usage and memory utilization of an app ?
* **You can use various tools like Usemon,CPU Usage Monitor, CPU Usage& Process List Viewer etc available in Google Play store and if you want to use sdk, then you can use systrace feature of Android Monitor**
* 10) what tools you use for performance testing and automation ?
* **@ Performance testing of the Web service which your application uses you can use jMeter, it is an open source tool which can be used to test the api's performances.**  
   **@Automation: It is very subjective term & totally depends on the project need and type of application , there are several paid tools available in the market like SeeTest, Ranorex, Silk Mobile etc while good free automation tools are  Calabash, Appium , Robotium for Android, KIF for iOS and using free tools you require some coding skills like ruby or Java.**  
  **If you can send me you specific requirement, i can help you more on this.**

**Why Cross-Browser Testing is an Important Part of QC**

One of the main aspects of cross-browser testing to keep in mind is that CSS styles render differently across browsers/browser versions, especially in terms of what is supported and what is not. Older versions of Internet Explorer are generally the most affected by unsupported CSS elements and newer HTML tags.In order to resolve these issues, front-end web developers can target specific versions of IE and write new code that will override the unsupported code without affecting any other browser. As a tester, it is important to determine which exact browsers/browser versions are affected by an issue so only those variations are updated through the CSS.

Since browsers are constantly updating and new versions are being released on a regular basis, keeping track of any changes that might impact how sites work is important. It is essential to continue to test across a wide variety of browsers/devices in order to accommodate a broader audience.

 Cross Browser testing involves testing both the client side and server side behavior of your Web application when it is accessed using different Web Browsers.

## What’s the difference?

### Continuous Integration

Continuous Integration is a software development practice in which you build and unit-test software every time a developer checks in new code. This provides Agile software teams the rapid feedback they need to respond to market demands and eliminate problems quickly.

### Continuous Delivery

Continuous Delivery (CD) is a software development practice in which continuous integration, automated testing, and automated deployment capabilities allow high-quality software to be developed and deployed rapidly, reliably and repeatedly with minimal manual overhead.

Continuouls delivery is a capaibility to deploy a software to any particualr enviroment to any given time, it includes any file like binary file,

configuration file or any change in environment

Continuous delivery is about putting the release schedule in the hands of the business, not in the hands of IT. Implementing continuous delivery

means making sure your software is always production ready throughout its entire lifecycle - that any build could potentially be released to

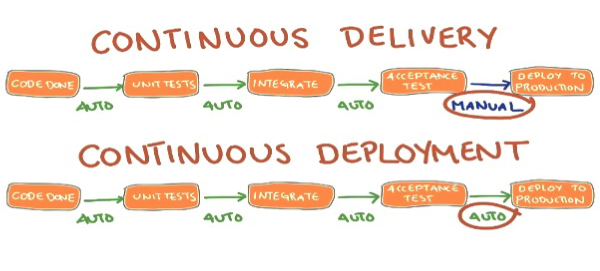
users at the touch of a button using a fully automated process in a matter of seconds or minutes.

## Continuous Deployment

Continuous Deployment is a software development practice in which every code change goes through the entire pipeline

and is put into production, automatically, resulting in many production deployments every day.

With Continuous Delivery your software is **always release-ready**, yet the timing of when to push it into production is a business decision, and so the final deployment is a **manual step**. With Continuous Deployment, any updated working version of the application is **automatically pushed** to production. Continuous Deployment mandates Continuous Delivery, but the opposite is not required.

**

**Continuous Delivery is sometimes confused with Continuous Deployment.** Continuous Deployment means that every change goes through the pipeline and automatically gets put into production, resulting in many production deployments every day. Continuous Delivery just means that you are able to do frequent deployments but may choose not to do it, usually due to businesses preferring a slower rate of deployment. In order to do Continuous Deployment you must be doing Continuous Delivery.

# Why are my emails going into recipients' spam folders?

Every recipient server is different and may apply different spam filtering criteria for the messages being received. In most cases, recipient servers don't provide any information about spam filtering to the sender of an email that was filtered. It would give too much information to spammers who would use that data to get around spam filters.

Every recipient server is different and may apply different spam filtering criteria for the messages being received. In most cases, recipient servers don't provide any information about spam filtering to the sender of an email that was filtered. It would give too much information to spammers who would use that data to get around spam filters.

Spam filters are constantly changing to adapt to new techniques and types of spam messages, so what lands in the inbox today could be spam filtered tomorrow. There are some things you can do to help ensure your emails are being delivered to the inbox.

***Engagement***

Most of the major email providers, including Yahoo!, AOL, Hotmail, and Gmail, use engagement-based filtering to help detect spam. This means that the more your recipients interact with your emails by opening, clicking, and reading, the more highly engaged your audience is, and the more likely you'll end up in their inboxes. If a lot of users are marking a message as spam, it's more likely to end up in other people's spam folders. If you've moved emails out of the spam folder, that's a positive indicator and can help ensure future emails that are similar are delivered to the inbox instead.

In addition to engagement, most spam filters use some kind of scoring mechanism to figure out how likely it is that an email is spam. If the email exceeds a certain threshold, it's filtered to the spam folder. The threshold changes for every receiving server, and some are configurable by the recipients. Learn more about [types of spam filters](http://templates.mailchimp.com/concepts/spam-filters/), and check out the MailChimp Knowledge Base for information on

**Rules and Guidelines for Responsive Design Testing**

Here are some important rules and guidelines for effective responsive design testing.

Text Alignment – Ensure that image and text are aligned perfectly

Clickable Zones – Pay special attention to these zones

Padding – Check for precise padding across all boundaries

Fonts – Ensure that font size, style and color are consistently maintained across the website/application

Scrolling – Ensure that flexible scrolling is offered to end user

Navigation – Simplify navigation between internal web pages

Boundaries – Make sure that text, images and frames are within the boundaries

Analytics – Utilize Google analytics to summarize the devices and browsers used by consumers

Indexing – Ensure responsive apps are indexed by search engines

Menus – Navigation menus must be designed for a broad range of devices

**Test cases for testing responsive website**

Some of the important factors to keep in mind while testing the RWD are as follows.

To verify all the images on the web page are displayed properly on all the different devices and resolution.

To verify text and headings on the web page are properly aligned.

To verify all the clickable links on the web page are readable and work as expected.

To verify scrolling of the web page works as expected.

To verify if there are input boxes and text areas to enter data then we need to make sure that the text entered is displayed properly on the web page and they are aligned as expected.

To verify image size, Font size and font type are consistent across all the web pages.

To verify if contents of the page are displayed consistent on all resolutions.

To verify the color changes after hover over the elements.

To verify the consistency of color combination on different resolutions.

To verify images, text, different controls are not going beyond the screen border.

To verify if there should not be any horizontal scrolling bar since everything should be fit according to the size of the screen.

To verify on rotating your mobile device, all the contents should be rotated and displayed as expected without any technical glitch.

To verify if the user able to click on clickable area.

To verify padding of elements on the edges.

To verify if enter text in input box are displayed as expected without any UI glitches.