Software Testing & Audit

Unit-5

Jayash Kumar Sharma
Department of Computer Science & Engineering
Hindustan Institute of Technology & Management
Mob: +91-9639325975
Email: jayash.sharma@gmail.com

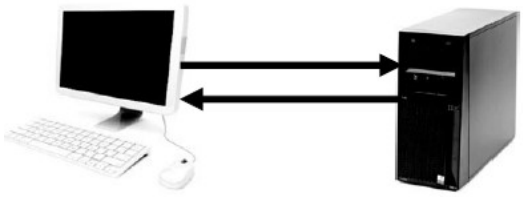
Testing Web Applications

What is Web Testing?

- To find common software errors.
- To test associated quality related risks that are specific to a web application.
- The quality of a web application must be assured in terms of response time, ease of use, number of users, ability to handle varied spikes in traffic, provide accurate information, etc.
- To achieve these objectives, we should know the architecture and key areas of web application to effectively plan and execute the testing.

Web Application versus Client Server Application

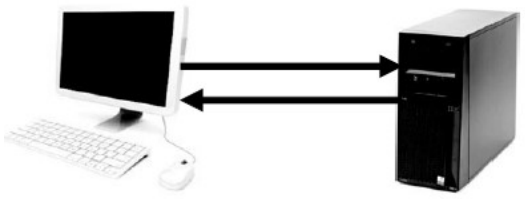
- In client-Server Application (2-tier Client Server Architecture) the client program is installed on each client machine that provides user interface.
- The clients are connected to a server machine which serves the client by providing requested information.



Client-Server Architecture

Web Application versus Client Server Application

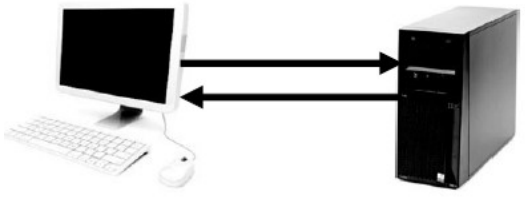
- In client-server architecture the functionality of an application is distributed between the client and a server.
- For example, business logic may reside on a server machine, user interface may reside on the client machine and database may reside on either client machine or the server.



Client-Server Architecture

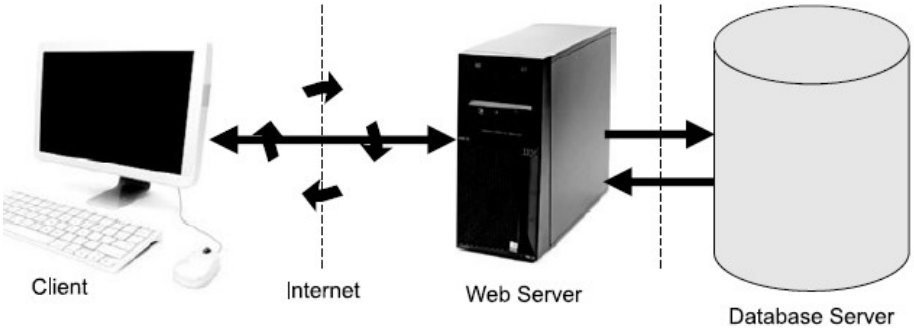
Web Application versus Client Server Application

- Any upgrade at the server side would require upgrading of client software that has been installed on all client machines.



Client-Server Architecture

Web Application versus Client Server Application



Three tier Architecture of web based applications

Web Application versus Client Server Application

- A web application may consist of multiple web servers and database servers.
- These applications need only web browser to be installed on the client machine.
- A web application depicts a three-tier architecture which comprises of client machines, web server incorporating business logic and database server for handing data storage.

Web Application versus Client Server Application

- In three-tier architecture, the applications are partitioned into three separate layers; hence changes in one tier do not have any effect on the other tier.

Advantages of three-tier architecture

- Less disk space required at the client machine
- Automatic up gradation of software and robustness.

- The client application may comprise of many active contents written in Java script, VBscript, DHTML and other technologies. Web servers use dynamic technology (ASP, JSP, Perl, CGI, Python) to render the active content. This may invoke incompatibility issues due to existence of varied browsers.

Web Application versus Client Server Application

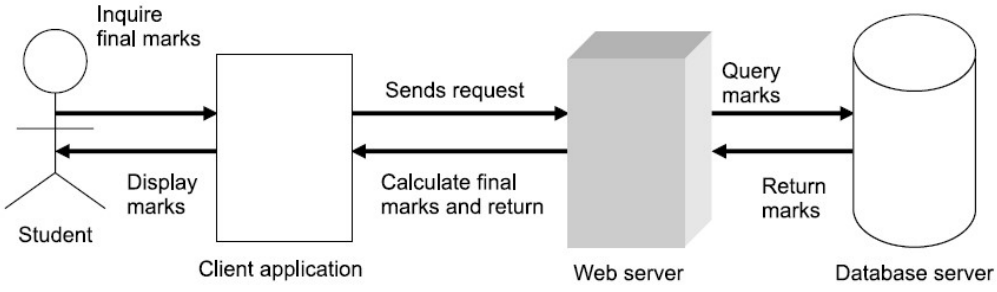
Client Server Application	**Web Application**
It is a 2-tier application.	It is a 3-tier application.
Additional software needs to be installed.	No additional software needs to be installed; only a web browser is required.
It is an expansive activity.	It is cheaper as compared to client/server application.
It supports a limited number of users.	It supports an unlimited number of users.
Users are well-known.	Any user can login and access the content.

Web Application versus Client Server Application

Client Server Application	Web Application
Manageable security issues.	Security issues are critical and complex in nature.
It is less robust as if one server fails, client request can not be fulfilled.	It is more robust as compared to client/server application.
More resources are required on the client side.	Fewer resources are needed on the client side.

Example:

If a student wants to know his/her final result, the following steps may be followed:



Key Areas in Testing Web Applications

- Functionality
- User Interface
- Usability
- Browser compatibility
- Security
- Load and stress
- Storage and Database

Functional Testing

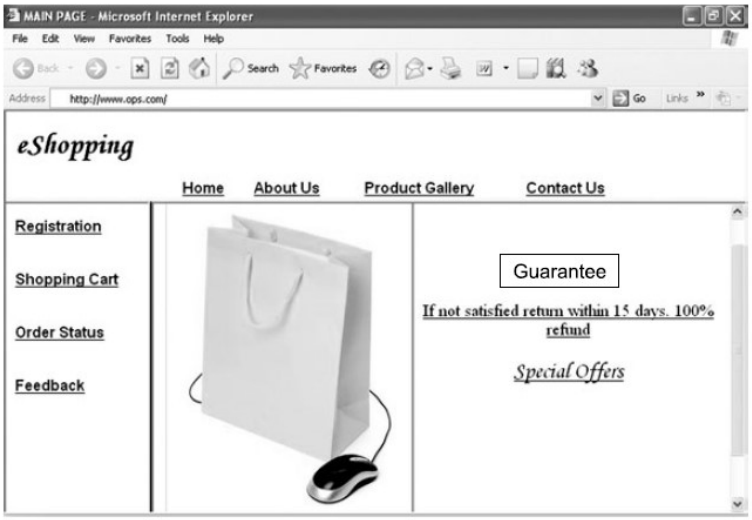
- Functional testing involves checking of the specified functionality of a web application.

- Functional test cases for web applications may be generated using:
  - Boundary value analysis
  - Equivalence class testing
  - Decision table testing and many other techniques.

Functional Testing - Example

An e-commerce application sells products such as computers, mobile phones, cameras, electronics, etc. For each item it lists the name, quality, price and brief description. It also displays an image of the item. The user may browse through the product and search any product by its name, price or any other descriptive keyword. The user is required to register on the website to obtain access to the online shopping cart (a common tool used by users to place an order online). The user selects various items and adds them to the shopping cart. After selecting items and adding those to the shopping cart, the user may choose to checkout in order to generate a bill. The application then requests the user to enter his credit card information and shipping preferences and after that the order is completed. Finally, the application displays the maximum number of days in which the item will be delivered to the user. The user may enquire about the order status any time. The option for providing feedback is also available on the website.

Functional Testing - Example



Functional Testing - Example

- Sample test cases for Order Processing

Test case id	Description (steps followed)	Inputs	Expected output
TC1	1. Search the product gallery to decide which items to purchase.	Search string	List of items searched are displayed.
TC2	1. Register on the website to place an order	Login id, password, confirm password, shipping details (address, city, state, zip code) and billing details (address, city, state, zip code).	If the information entered is valid the user is registered successfully, otherwise an appropriate error message is displayed.

Functional Testing - Example

- Sample test cases for Order Processing

TC3	1. Log into the website.	Login id, password	Item is successfully added in the shopping cart.
	2. Select item to be purchased and its quantity.	Item number, item name, quantity	
	3. Add selected item to the shopping cart.	-	
TC4	1. Log into the website.	Login id, password	Item is not added in the shopping cart.
	2. Select item to be purchased and its quantity.	Item number, item name, quantity	
	3. Do not add selected item to the shopping cart.	-	

Functional Testing - Example

TC5	1. Log into the website.	Login id, password	Items are successfully added in the shopping cart.
	2. Select items to be purchased and their quantity.	Item number, item name, quantity	
	3. Add selected items to the shopping cart.	Item number, item name, quantity	
	4. Select some more items and add them to the shopping cart.	Item number, item name, quantity	
TC6	1. Log into the website.	Login id, password	If deletion is confirmed, item is successfully deleted from the shopping cart.
	2. Select two or more items to be purchased and their quantity.	Item number, item name, quantity	
	3. Add selected items to the shopping cart.	Item number, item name, quantity	
	4. Delete one item from the shopping cart.	Item number	

Functional Testing - Example

TC7	1. Log into the website.	Login id, password	If updation is confirmed, quantity is successfully updated.
	2. Select item to be purchased and its quantity.	Item number, item name, quantity	
	3. Add selected items to the shopping cart.	Item number, item name, quantity	
	4. Update quantity of the item added to the shopping cart.	Quantity	

Functional Testing - Example

TC8	1. Log into the website.	Login id, password	User check outs from the shopping cart and total bill of the purchased items is displayed.
	2. Select items to be purchased and their quantities.	Item number, item name, quantity	
	3. Add selected items to the shopping cart.	Item number, item name, quantity	
	4. Checkout	-	
TC9	1. Log into the website.	Login id, password	After authentication, amount is transferred and items are delivered successfully.
	2. Select items to be purchased and their quantities.	Item number, item name, quantity	
	3. Add selected items to the shopping cart.	Item number, item name, quantity	
	4. Checkout	-	
	5. Enter valid credit card information	Bank name, credit card type, credit card number	

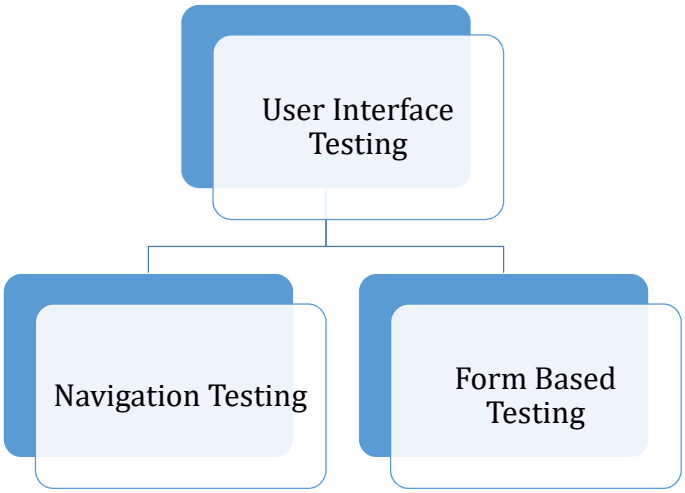
Functional Testing - Example

TC10	1. Log into the website.	Login id, password,	Appropriate error message is displayed.
	2. Select items to be purchased and their quantities.	Item number, item name, quantity	
	3. Add selected items to the shopping cart.	Item number, item name, quantity	
	4. Checkout	-	
	5. Enter invalid credit card information	Bank name, credit card type, credit card number	

User Interface Testing

- User interface testing tests that the user interaction features work correctly.
- These features include hyperlinks, tables, forms, frames and user interface items such as text fields, radio buttons, check boxes, list boxes, combo boxes, buttons and dialog boxes.
- User interface testing ensures that the application handles mouse and keyboard events correctly and displays hyperlinks, tables, frames, buttons, menus, dialog boxes, error message boxes, and toolbars properly

User Interface Testing



Navigation Testing

- Navigation testing investigates the proper functioning of all the internal and external links.
- Navigation testing must ensure that websites provide consistent, well-organized links and should also provide alternative navigation schemes such as search options and site maps.
- The placement of navigation links on each page must be checked.

Navigation Testing

- Search based navigation facility must also be thoroughly tested and search items should be consistent across one page to another.
- All the combinations of keywords and search criteria must be verified in navigation testing.

Navigation Testing

- Search based navigation facility must also be thoroughly tested and search items should be consistent across one page to another.
- All the combinations of keywords and search criteria must be verified in navigation testing.
- Example: Navigation testing test cases for online shopping website

Navigation Testing

Test case id	Description	Inputs	Expected output
TC1	Check all links on each web page.	Link1=Home Link2=About us Link3=Product Gallery Link4=Contact us Link5=Registration Link6=Shopping Cart Link7=Order Status Link8=Feedback	Appropriate web page is opened with respect to each link.

Navigation Testing

TC2	Click on all links on each web page to test the appearance of content on each web page.	Link1=Home Link2=About us Link3=Product Gallery Link4=Contact us Link5=Registration Link6=Shopping Cart Link7=Order Status Link8=Feedback	Appropriate horizontal and vertical scroll bars are present and the user can view the page contents properly
TC3	Search for items in the available product gallery	Search string	The user is able to navigate across multiple search pages successfully.
TC4	Click on 'back' link present on each page.	-	The appropriate page is displayed.

Navigation Testing

- Manual checking of hyperlinks can be very time consuming.
- Software tools are available for checking broken links, accuracy and availability of links and obtaining advice on search engines.
- Some tools for navigation testing include TestLink, Link checker, LinkSleuth, Dead Links, LinkTiger, LinkRunner, LinkScan, Link Validator, MQMspider and WebLight.

Form Based Testing

- Websites that include forms need to ensure that all the fields in the form are working properly.

- Form-based testing involves the following issues:

1. Proper navigation from one field of the form to another using the tab key.
2. Ensures that the data entered in the form is in a valid format.
3. Checks that all the mandatory fields are entered in the form.

Form Based Testing - Example

Sample Online Registration Form

Form Based Testing - Example

Test case id	Description	Inputs	Expected output
TC1	Navigate using tab from one field to another in the form	-	The user follows the correct sequence while navigating from one field to another.
TC2	Check maximum and minimum length of all fields in the form (boundary value analysis).	Login id, password, confirm password, shipping details (address, city, state, zip code) and billing details (address, city, state, zip code)	If the characters are entered within the range, the information is added successfully; otherwise an appropriate error message is displayed.

Form Based Testing - Example

TC3	Check data validations of all fields in the form.	Login id, password, confirm password, shipping details (address, city, state, zip code) and billing details (address, city, state, zip code).	If the characters are valid the information is added successfully, otherwise an appropriate error message is displayed.
TC4	Check whether all the mandatory fields are entered in the form.	Login id, password, confirm password, shipping details (address, city, state, zip code) and billing details (address, city, state, zip code).	If all the required fields are entered the information is added successfully, otherwise an appropriate error message is displayed.

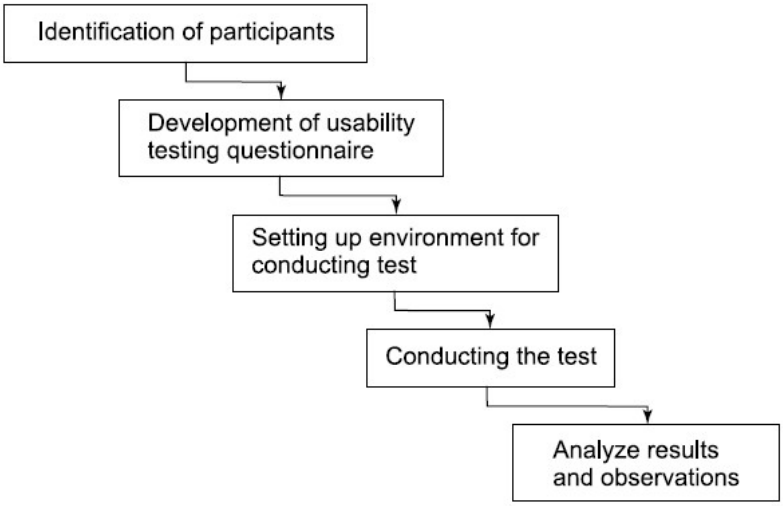
Usability Testing

- Usability is one of the quality attributes that a web application must possess.
- Usability is concerned with the degree to which the software fulfils the user's specifications and expectations.
- It is the measurement of the amount of satisfaction of the user.
- It also assesses the extent to which the user finds the software easy to use, learn and understand.

Usability Testing - Attributes

Attribute	**Definition**
Accuracy	Specifies the degree to which the website meets the user's needs with precision and correctness.
Efficiency	It refers to the correctness, easiness and quickness of use of the website by a user. It is generally measured in terms of time.
Completeness	The extent to which a website implements specified functionalities.
Learnability	Defines the user's ability to effectively perform operations on the website.
Satisfaction	It refers to the user's feeling and opinion about the website. It is usually captured through a questionnaire or survey. Users are more likely to use websites that satisfy their needs as compared to other ones.
Clarity and accuracy of online help and written documentation	The extent to which the online help and documentation is clearly and accurately written.

Usability Testing - Approach



Usability Testing - Approach

Identification of Participants

- Who are the target users of a web application?
- It is important for an organization to identify the characteristics of the participating users.
- Such characteristics may include age, gender, profession and application specific experience.

Usability Testing - Approach

Identification of Participants

- We must select an appropriate number of participants from different groups of target users as each group of users will use the website with a different purpose.

Participant name _____	
Date_____	
Age	16-20
	21-30
	31-40
	41-50
	>50
Gender	Male
	Female
Profession	_____
Education level	Higher secondary Graduate Post graduate
Customer type	Whole sale purchaser Retail purchaser
Shopping experience	
Shopping frequency	

Usability Testing - Approach

Development of Usability Questionnaire

- Preparation of the questionnaire is an important activity and should take into consideration the usability features of the web application.
- The participant's feedback and reactions are recorded in the questionnaire.
- The research questions must include the participant's likes and dislikes about the website.

Usability Testing - Approach

Development of Usability Questionnaire – Sample for online shopping

1. How easily and successfully are you able to register on this website?
2. What paths did you take to complete an order?
3. How closely did the order process meet with your specifications?
4. What problems were encountered while placing an order?
5. How do you feel about the time taken to complete the order process (in terms of time and number of steps)?

Usability Testing - Approach

Setting up Environment for Conducting Test

- The decision of location and setups is based on various factors such as:
  - Whether the tester's interaction with the user is required or not?
  - Whether enough space is available at the developer's/ tester's site in order to conduct usability tests?
  - Is the location of the site easily accessible to target participants?
  - What equipments will be required for conducting the tests?
  - How many participants and observers will be required?
  - Will the identity of the organization result in biased results?
  - What is the availability of the participants?
  - Is testing required at multiple geographic locations?

Usability Testing - Approach

Conducting the Test

- This step consists of the execution of usability testing.
- The usability tests will be based on the questions prepared.
- These questions may be tailored as per the type of a web application.

Usability Testing - Approach

Analyze the Results and Observations

- The process of generation of usability testing report involves organizing, summarizing and analyzing the collected data.
- This report contains a summary of the user's preferences, list of errors and difficulties encountered, identification of customer trends, and analysis of recordings.
- Finally the observations are statistically analyzed and based on these analysis recommendations are proposed. The tasks that do not meet the desired specifications are identified and prioritized.

Configuration & Compatibility Testing

- One of the significant challenges of web testing is that it must ensure the proper functioning of a web application on all the supported platforms and suitable environments.
- The goal of configuration and compatibility testing is to detect faults in the application while it is tested on varied platforms and environments.
- The performance and system's requirement specifications formed during the start of the project provides a baseline for creating configuration and compatibility test cases.

Configuration & Compatibility Testing

- Configuration testing determines the behaviour of the software with respect to various configurations.
- Compatibility testing determines whether the web application behaves as expected with respect to various supported configurations.
- The configuration and compatibility testing concerns checking web application with different:

Configuration & Compatibility Testing

- Browsers such as Internet Explorer (IE), Chrome, Opera, Mozilla, etc.
- User interface components such as Java Applets, Active X, etc.
- Operating systems such as Windows, Macintosh, Linux, etc.
- Internet connection types such as broadband, Dial-up, leased line, etc.
- Hardware devices such as CPU, RAM, CD-ROM, input devices (mouse, keyboard, joystick), output devices (such as printers, plotters) network card, monitor, graphic display card, etc.
- Multimedia services such as text, audio, video, etc.
- Database such as MS Access, SQL Server, Oracle, Sybase, etc.
- Mobile devices such as Nokia, Motorola, Samsung, etc.

Browser Testing

- There are a large number of browsers available and the behaviour of each of them may vary.
- Although it is impractical to test the web application with all the browsers, it is necessary to verify the web application with specified and prioritized platforms to ensure its correct functioning.
- Browser testing verifies the functioning of web application in terms of text, audio, video and operating system corresponding to different browsers

Security Testing

- Security is the procedure used to protect information from various threats.
- It is important to protect sensitive and critical information and data while communicating over the network.
- The user wants implementation of a safeguard to protect personal, sensitive and financial information.
- We want data to be accurate, reliable and protected against unauthorized access.

Security Testing

The primary requirement of security includes:

- Authentication: Is the information sent from an authenticated user?
- Access Control: Is data protected from unauthorized users?
- Integrity: Does the user receive exactly what is sent?
- Delivery: Is the information delivered to the intended user?
- Reliability: What is the frequency of a failure? How much time does the network take to recover from a failure? What measures are taken to counter catastrophic failure?
- Non-repudiation: Is the receiver able to prove that the data received came from a specific sender?

Performance Testing

- To ensure that the web application can bear the load during the peak hours along with serving the user in a timely and reliable manner, performance tests including load and stress tests need to be conducted.

- Several factors that may influence performance include:
  - Response time
  - Memory available
  - Network bandwidth
  - Number of users
  - User type
  - Time to download
  - Varied client machine configurations

Performance Testing

- The goal of performance testing is to evaluate the application's performance with respect to real world scenarios.

- Following issues must be addressed during performance testing:
  - Performance of the system during peak hours (response time, reliability and availability).
  - Points at which the system performance degrades or system fails.
  - Impact of the degraded performance on the customer loyalty, sales and profits.

Performance Testing – Load Testing

- Load testing involves testing the web application under real world scenarios by simulating numerous users accessing the web application simultaneously.
- It tests the web application by providing it maximum load.
- The development of plans for load testing should begin as early as possible during the software life cycle.
- Early testing will help in detection of problems prior to deployment of the web application.

Performance Testing – Stress Testing

- Stress testing involves execution of a web application with more than maximum and varying loads for long periods.
- Stress testing evaluates the response of the system when the system is given a load beyond its specified limits.
- It is also used to monitor and check the reliability of a web application when available resources are on beyond maximum usage.

Performance Testing – Stress Testing

- The behaviour of the system is monitored to determine when the system under stress test fails and how does it recover from the failure.
- Stress tests may test the web application for the following:
  - CPU and memory usage
  - Response time
  - Backend database
  - Different types of users
  - Concurrent users

Performance Testing – Stress Testing

- The system performance is expected to degrade when a large number of users hit the web site simultaneously.
- After the completion of stress tests, the testing team must analyze the noted system's performance degradation points and compare them with the acceptable performance of the system.

Database Testing

- Testing data-centric web applications is important to ensure their error-free operation and increased customer satisfaction.

- Important issues in database testing may include:
  - Data validation
  - Data consistency
  - Data integrity
  - Concurrency control and recovery
  - Data manipulation operations such as addition, deletion, updation and retrieval of data.
  - Database security

Database Testing

- A database must be tested for administrative level operations such as adding, deleting and updating an item in the database, and user operations such as searching an item from the database or providing personal details.

Administrative operations

- Inserting a new item into the database
- Deleting an existing item from the database
- Updating an existing item from the database
- Viewing an existing item from the database

Database Testing

User Operations

- Searching items from the database
- Registering into the website involves storing the user's personal details
- Placing an order involves storing user preferences and purchase details into the database
- Providing feedback involves storing information in the database
- Tracking status of the order placed

Post Deployment Testing

- Post-deployment testing may reveal those problems which went undetected before deployment of the web application.
- Despite all the planning and testing carried out before deployment, obtaining user opinion is important for improvement of a website and it ensures that the website adapts to the needs of the user.
- User feedback may come in various forms, ranging from reporting of faults to suggestions for improvement of the website.

Post Deployment Testing

- The effective way to obtain a user's opinion is to get a questionnaire or survey filled by the user.
- The questionnaire/survey can be used to detect trends and may provide valuable information for improvement of the website.
- The response obtained from this survey may help the developer/owner of the website to improve the website.

Post Deployment Testing

Consistency				
1.	Are the colour schemes consistent across displays?	**Bad**	1 ○ 2 ○ 3 ○ 4 ○ 5 ○	**Good**
2.	Is the format of display of topics consistent?	**Bad**	1 ○ 2 ○ 3 ○ 4 ○ 5 ○	**Good**
3.	Is the location of labels consistent?	**Bad**	1 ○ 2 ○ 3 ○ 4 ○ 5 ○	**Good**
4.	Is the format of labels consistent?	**Bad**	1 ○ 2 ○ 3 ○ 4 ○ 5 ○	**Good**
5.	Is the orientation (scrolling) of items consistent?	**Bad**	1 ○ 2 ○ 3 ○ 4 ○ 5 ○	**Good**

Post Deployment Testing

Flexibility								
6.	Are the users able to customize windows?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
7.	Can the user interface items be resized flexibly?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
8.	Can you enter data flexibly?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
9.	Can you select data?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
10.	Can display be expanded by using zooming option?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**

Post Deployment Testing

Learnability								
11.	Is the text clearly written and easy to understand?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
12.	Is data grouping easy to learn?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
13.	Are the names of user interface items logical?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
14.	Are the links working correctly?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
15.	How easy is it to learn various user interfaces in the website?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**

Post Deployment Testing

User Guidance								
16.	How helpful were the error messages?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
17.	Does online help provide useful information?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
18.	Are error messages informative?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
19.	Is the 'undo' option available to revert back the user's actions?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
20.	Can the user navigate easily through the website?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**

Post Deployment Testing

21.	Is sufficient information available for the intended audience?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
22.	Are you able to find the required information you were locking for?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
23.	Would you recommend this website to other users?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
24.	How will you overall rate the website?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**
25.	Is website compatible with your browser?	**Bad**	1 ○	2 ○	3 ○	4 ○	5 ○	**Good**

Post Deployment Testing

- Once the user's opinion is obtained, it is important to identify useful fault reporting, suggestions and recommendations.

1. Frequency of suggestion: How many users have given the same suggestion or recommendation? If a small number of users are making the same request, then we must think twice before implementing the suggestion.

2. Source of feedback: Who is providing the suggestion? It is vital to make sure that suggestions come from regular users and not accidental users.

Post Deployment Testing

3. Cost of implementing the suggestion: Is the suggested idea worth implementing? The correctness of the proposed change and its impact on the cost and schedule must be analyzed carefully. The benefits of implementing the suggested idea to the business must be determined.

4. Impact of implementing the suggestion: Will implementing the suggestion increase complexity of the website? Will the change be compatible with the other functionalities of the website?

Post Deployment Testing