

Cross Browser Testing

A complete guide for Cross Browser Testing

4/10/2017

MagikMinds

Surendra Madugula

**Contents**

[Overview: 3](#_Toc479603974)

[What is Cross-Browser Testing? 3](#_Toc479603975)

[Why is Cross Browser Testing performed? 3](#_Toc479603976)

[Reasons of Cross Browser Issues: 3](#_Toc479603977)

[How is Browser Compatibility Testing performed? 4](#_Toc479603978)

[To summarize “how” to cross browser test: 4](#_Toc479603979)

[Cross Browser testing methods: 5](#_Toc479603980)

[Tools for Cross Browser testing: 6](#_Toc479603981)

[Testing Checklist: 6](#_Toc479603982)

# **Overview:**

We believe that any concept will make a world of sense when we ask the basic question words around like- **“What, why, how, who, when, where”.**

Many times, we have encountered an issue with a website and on calling the technical support, they simply tell us to try it in another browser? When we do, it works and we end up feeling like a total idiot, even though we earn my living working in the software industry.

It’s just that the website has not been tested extensively with respect to **cross browser compatibility** testing and as an end user we have just found a bug.

# **What is Cross-Browser Testing?**

Cross browser testing is simply what its name means- that is, to test your website or application in multiple browsers and making sure that it works consistently and as in intended without any dependencies, or compromise in Quality.

(OR)

**Cross Browser Testing** is a type of functional test to check that your web application works as expected in different browsers.

This is applicable to both web and mobile applications. And all type of customer facing applications are best choice undergo to perform the cross-browser testing.

# **Why is Cross Browser Testing performed?**

Web-based applications are totally different from Windows applications. A web application can be opened in any browser by the end user. For example, some people prefer to open in **Firefox browser,**while other's can be using **Chrome browser** or**IE**.

So, we need to ensure that the web application will work as expected in all popular browsers so that more people can access it and use it.

This motive can be fulfilled with Cross Browser Testing of the product.

## **Reasons of Cross Browser Issues:**

1. Font size mismatch in different browsers.
2. JavaScript implementation can be different.
3. CSS, HTML validation difference can be there.
4. Some browser still not supporting HTML5.
5. Page alignment and div size.
6. Image orientation.
7. Browser incompatibility with OS, etc...

**Who performs this type of testing and who are these results relevant to?**

* There are a million browsers, versions and platforms out there- which ones to choose? This thankfully, is not a decision that is the tester’s responsibility. The client, business analysis team and the marketing teams have a key role in this decision. Also, companies collect usage/traffic statistics to narrow down what browsers, environment and devices are mostly in use.
* The entire project team should have an invested interest, time, money and infrastructure to support this endeavor.
* The QA team can be involved in this process or it might be the design team who are keen on knowing how the application fares in multiple browsers.
* Whether it is performed by QA or any other team, the results are interpreted by the design and development teams and the relevant changes are made.

# **How is Browser Compatibility Testing performed?**

**First things first- is it done manually or using a tool?**

It can surely be done manually- multiple machines, multiple OS’s, Multiple browsers but clearly, this leads to multiple problems, multiple investments and multiple challenges.

So, lots of tools are available in the market to make this easier.

**The tools help us with one or more of the following depending on the tool itself and the licensing types:**

1. They provide a VPN (Virtual Private machine) using which you can connect to remote machines and check the working and rendition of your JAVA, AJAX, HTML, Flash and other pages. Most of these are secure, but since you are submitting your information to a third party, a certain analysis on discretion is advised.
2. Screenshots are provided for the pages and links submitted of how they appear in multiple browsers. This is, of course, static.
3. Multiple browsers are synchronized with respect to operations performed on one and the results are presented browser wise.
4. Show the rendition of a page at multiple screen resolutions.
5. When a problem is encountered, a video or screenshots are recorded to transport the problem for further analysis.
6. Support generally is available for both web and mobile apps.
7. Private pages that require authentication to be accessed can also be tested.
8. Local, with in a private network/firewall pages can be tested too.

## **To summarize “how” to cross browser test:**

**a)** Traffic statistics help determine what browsers to test.

**b)** A detailed analysis should be done on the AUT (Application under test) itself to determine what parts of the application or if all of it should undergo this. It is advisable that all of it be tested on multiple browsers, but again costs and time must be considered. **A good strategy is to perform 100% testing on one browser per platform and for the other just test the most critical/widely used functionality.**

**c)** Once the decision of “What” to test and “Where (browsers)” is made- infrastructure decisions are to be made- do we acquire tools or perform this manually etc. Again, cost should be considered. Viability, risks, security concerns, people to be involved, time, acceptance criteria, issue/defect fixing schedules/process – are few things that have to be addressed.

**d)** Perform the testing. The regular functional testing test cases can be used when validating the efficiency of the system. For look-and-feel/rendition test cases are not necessary.

**e)** Report the results back to the design team, if they were not involved in the testing process, change follows.

**When is the best time to do this?**

Any testing reaps the best benefits when it is done early on. Therefore, the industry recommendation is to start with it as soon as the page designs are available.

But it also can be performed when the site is fully integrated and functional.

If you have missed the bus on performing cross browser testing during design, development and QA phases, it can still be done while the application is in production. However, this is the costliest of all and risky too.

**Where is browser compatibility testing performed?**

Usually, the answer to this question would be one of- Dev/QA/Production environments. But for cross browser testing this, is not a definite and irrelevant. It can be done in any one or all of them.

# **Cross Browser testing methods:**

**1.** CSS validation  
**2.** HTML or XHTML validation  
**3.** Page validations with and without JavaScript enabled  
**4.** Ajax and JQuery functionality  
**5.** Font size validation  
**6.** Page layout in different resolutions  
**7.** All images and alignment  
**8.**Header and footer sections  
**9.** Page content alignment to center, LHS or RHS  
**10.** Page styles  
**11.**Date formats  
**12.**Special characters with HTML character encoding  
**13.** Page zoom-in and zoom-out functionality

And obviously, you will have to repeat these tests on:  
**14.** Different Operating Systems like Windows, Linux and Mac  
**15.**Different browsers (with different versions) like Internet explorer, Firefox, Google Chrome, Safari and Opera.

## **Tools for Cross Browser testing:**

Free tools:

**a. Spoon Browser Sandbox.**

**b. Bowsershots.**

**c. IE NetRenderer.**

**d. IE Tab.**

**e. IE Tester.**

1. **Microsoft Super Preview.**

Commercial tools:

1. **Browsera.**
2. **Browsercam.**
3. Cross Browser testing.
4. Cloud testing. etc.…

# **Testing Checklist:**

**1.** Create System and Acceptance Tests.  
**2.**Start Acceptance Test Creation.  
**3.** Identify test team.  
**4.**Create Workplan.  
**5.** Create test Approach.  
**6.** Link Acceptance Criteria and Requirements to form the basis of acceptance test.  
**7.**Use subset of system test cases to form requirements portion of acceptance test.  
**8.** Create scripts for use by the customer to demonstrate that the system meets requirements.  
**9.** Create test schedule. Include people and all other resources.  
**10.** Conduct Acceptance Test.  
**11.**Start System Test Creation.  
**12.** Identify test team members.  
**13.** Create Workplan.  
**14.**Determine resource requirements.  
**15.** Identify productivity tools for testing.  
**16.** Determine data requirements.  
**17.** Reach agreement with data center.  
**18.** Create test Approach.  
**19.** Identify any facilities that are needed.  
**20.** Obtain and review existing test material.  
**21.** Create inventory of test items.  
**22.**Identify Design states, conditions, processes, and procedures.  
**23.** Determine the need for Code based (white box) testing. Identify conditions.  
**24.** Identify all functional requirements.  
**25.** End inventory creation.  
**26.** Start test case creation.  
**27.**Create test cases based on inventory of test items.  
**28.** Identify logical groups of business function for new system.  
**29.** Divide test cases into functional groups traced to test item inventory, design data sets corresponding to test cases.  
**30.** End test case creation.  
**31.** Review business functions, test cases, and data sets with users.  
**32.** Get signoff on test design from Project leader and QA.  
**33.** End Test Design.  
**34.** Begin test Preparation.  
**35.** Obtain test support resources.  
**36.** Outline expected results for each test case.  
**37.** Obtain test data. Validate and trace to test cases.  
**38.** Prepare detailed test scripts for each test case.  
**39.** Prepare & document environmental set up procedures. Include backup and recovery plans.  
**40.** End Test Preparation phase.  
**41.** Conduct System Test.  
**42.**Execute test scripts.  
**43.** Compare actual result to expected.  
**44.** Document discrepancies and create problem report.  
**45.** Prepare maintenance phase input.  
**46.** Re-execute test group after problem repairs.  
**47.** Create final test report, include known bugs list.  
**48.** Obtain formal signoff.