

A Survey on Cross Browser Inconsistencies in Web Application

Nepal Barskar
PG Scholar

Department of Computer Science and Engg.
IET-DAVV –Indore, India

C.P. Patidar

Assistant Professor
Department of Information Technology
IET-DAVV –Indore, India

ABSTRACT

Because of the advancement of Web technologies and the inconsistency of Web standards to be consistent with every single technology evolution, Web developers are been faced with certain challenges. It become difficult for them to keep track of their applications that are been correctly rendered across wide range of browsers and platforms. To find out cross-browser inconsistency, developers keep checking that every document produced by the application is appropriately passed across all useful browser-platform combinations. But this requires more time and results are error-prone. The existing cross-browser compatibility testing tools speed up this process by automating the rendering of a Web document in multiple browsers and platforms, and using either image analysis or Document Object Model (DOM) analysis to feature out cross-browser inconsistency. This paper comprises the problem of cross-browser inconsistent testing of the modern web applications as a functional accuracy check of web application's behaviour with various web browsers and present a solution for it. Also, reasons of cross-browser inconsistency issues and solutions to them are been presented in the paper.

General Terms

Web application, web server, cross browsers inconsistency, Document Object Model .

Keywords

Cross-browser inconsistency, web application, web server, software testing, web testing, and IE tester.

1. INTRODUCTION

Web is been considered as a platform for various web applications & websites to communicate with the users by sharing resources such as data, audio, video, applications. These web applications are used for interacting over wide-network to cover many purposes.

Web servers: Web server is a program which can communicate through World Wide Web over the internet. This communication with web takes place through HTML (Hyper Text Mark Up language). Web servers receives request from browser such as Google Chrome, Firefox & Mozilla, sends the request to the web applications. Web server serve the files that form web pages to users, in response to their requests, which are forwarded by their computers.[3].Web servers are used for hosting scripts, multimedia files, pages, programs etc for the purpose of sending & receiving files through IEEE based protocols.

Working of web application:

The three layered web application model is discussed in figure 1.The first layer is comprised of a user interface or web browser. A web browser is a software application for retrieving, presenting and traversing information resources on

the World Wide Web. An information resource is been identified by a Uniform Resource Identifier (URI/URL) and it may be a web page, image, video or other piece of content. Users can easily navigate from their browsers to obtain information through hyperlinks. Second layer is a tool which comprise of JSD (java servlets) or ASP (active server page) which are used for dynamic content generation and third layer is database which consist of data containing card details, login id & password etc. Web application are of two types, former is Presentation-oriented (XML, HTML page) called web server. A web server is a computer program that processes requests through HTTP, the basic network protocol used to distribute information on the World Wide Web. The term can refer either to the entire system, or specifically to the software that accepts and supervises the HTTP requests [3, 4].

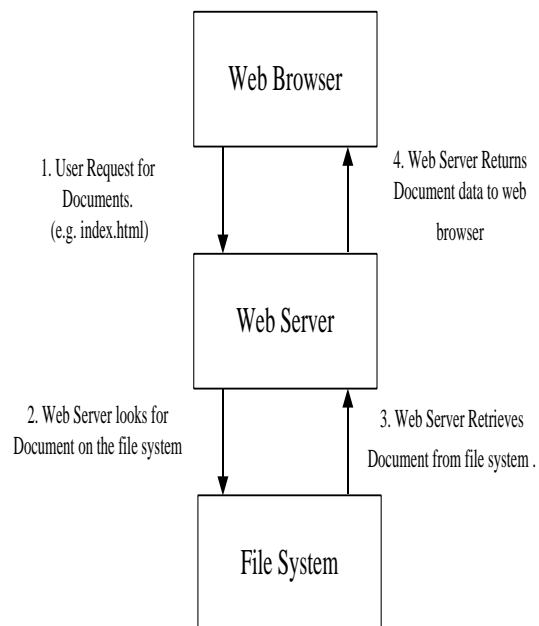


Fig.: 1 Three layered Web Application Model

Web Application: Web applications are platform independent cross-browser which support multiple web browsers. Web applications are client-server software applications which run within a web browser. It is easy to maintain & update without affecting the web application to be installed on different computers. Web software applications are used for online actions messaging business web mail and other services as shown in figure 2. Web application are website based navigation in which client input affects the nature of business instead of hit counters access logs of the web applications [5].

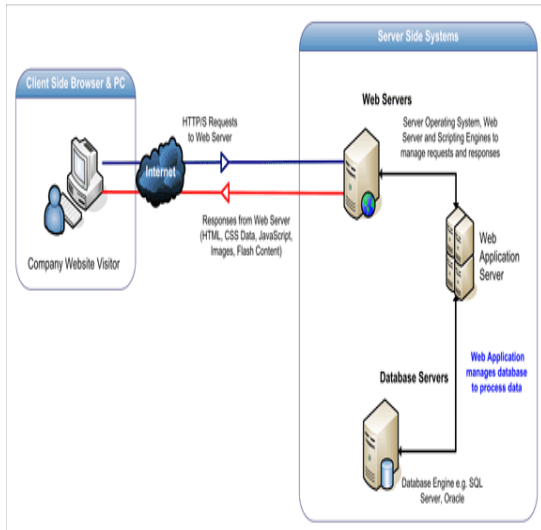


Fig.-2 Working of Web Application

2. CROSS-BROWSER INCONSISTENCIES (XBIS)

XBIs is an acronym used for cross-browser inconsistencies. XBIs create discrepancies in nature & also the behaviour of web browsers running on different environment together to host with operating system is been affected. XBIs lead to serious concern for organizations which are developing web based software due to increasing popularity of different browsers & web applications. For example, according to stack over flow.com, posts target on “cross browser” have increase over 1600 in recent years[6]. The classification of XBIs, are described in this section, three main types of XBIs are as follows[7].

(i) **Structural XBIs:** These XBIs affects the structure, or layout of individual web pages as shown in figure 3. The web page structure is basically a particular arrangement of elements, which in case of structural XBIs is erroneous in a particular browser. For example, the incorrect positioning of one or more web page elements on a given web page, in a particular browser, can be constituted within a structural XBI. We found that this was the most common category of XBIs, occurring in 57% of the subjects with XBIs [8].

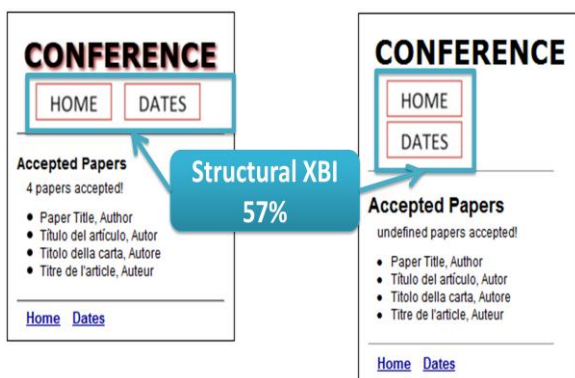


Fig.-3 Example of Structural XBIs

(ii) **Content XBIs:** This type of XBIs is been observed in the content of individual components on a web page. Content XBIs are divided into two types text content and visual content as shown in figure 4 and figure 5. Such differences can occur, where the visual look of a web page element, or the

textual value of an element, vary across different browsers. In this literature survey, we found that 30% and 22% of the sites are contained with these XBIs.

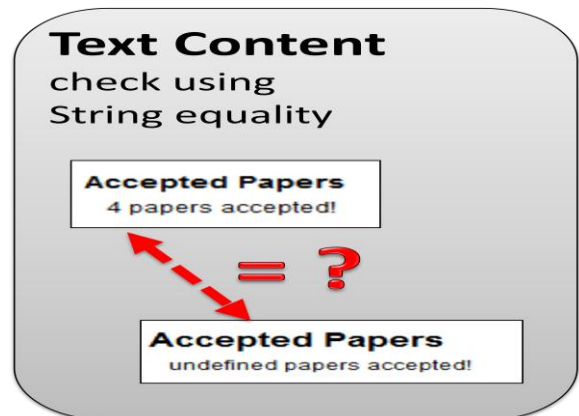


Fig.-4 Example of Text Content XBIs

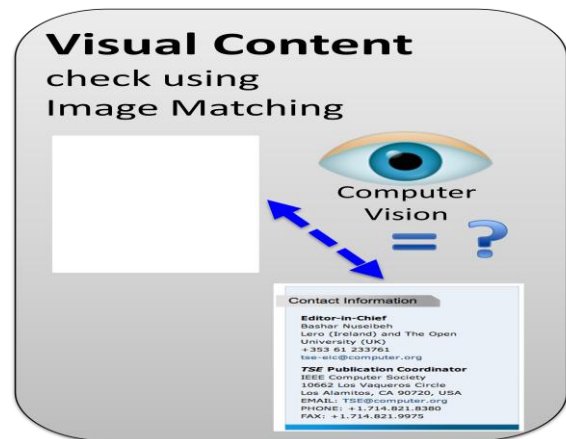


Fig.-5 Example of Visual Content XBIs

(iii) **Behavioural XBIs:** These type of XBIs involve variations in the behaviour of individual widgets on a web page. An example of such an XBI would be of a button that performs a particular action within one browser and a totally different action, or no action at all, in another browser. Another example of behavioural XBI is the presence of an HTML link, which works in one browser but is broken in another one. In our study, such XBIs occurred in 9% of the web applications with XBIs as shown in figure 6.

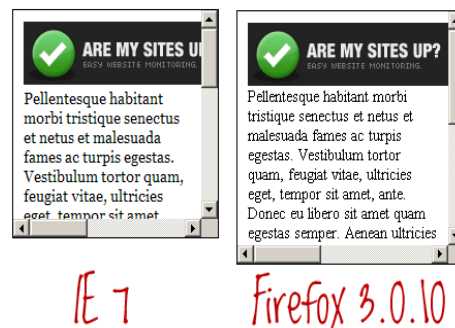


Fig:6 Example of Behaviour XBIs

Behavioural XBIs affects the functionality of the components, resulting in broken navigation between different screens. Structural and content XBIs, conversely, involve differences in the arrangement or rendering of elements on a

particular web page. In the next section, we describe how our technique detects each of these XBIs [9].

3. LITERATURE REVIEW OF XBIS

WEBDIFF [10] and CROSST [11] techniques provide the foundations of the approach that is been discussed in this paper. WEBDIFF detects XBIs on an web page through a two step process. Firstly, DOM-matching step is been performed to find pairs of corresponding screen elements in displaying of the pages in two different browsers. And then it performs visual comparison of the pairs of screen elements to find XBIs. This visual comparison uses a hand-crafted approach that considers many visual properties of the screen elements. In contrast, CROSST focuses on finding trace level XBIs in the dynamic state-space of a web application. CROSST uses automatic crawling to make a graph of the state space of a web application under corresponding browsers. To detect the trace-level XBIs, CROSST then checks the isomorphism of two graphs .CROSST also performs a operation on matched states to find DOM-level differences (although this differencing can generate a large number of false positives).

The two complementary approaches, WEBDIFF and CROSST are been used for automatically searching and reporting XBIs. The author proposed CROSSCHECK, a more powerful and comprehensive approach for XBI detection that adapts these two approaches in a way that expands their strengths. CROSSCHECK is testing on a set of real-world web applications. The result of CROSSCHECK was both effective and efficient in detecting XBIs, and it can outperform with existing techniques. The CROSSCHECK technique consists of three phases. The first phase crawls the target web application automatically within each of the two browsers environments. The observed behaviour is been captured and recorded in this process in the form of two navigation models, M1 and M2, one for each browser. The crawling is performed in an similar fashion within each browser, so as to exercise accurately the same set of sequences of user interaction within the web application under each case. This phase is implemented by function `genCrawlModel()` in the Algorithm. The second phase compares models M1 and M2 to check whether they are equivalent and exhibit a set of model differences, which may display one or more XBIs. And the third phase analyses this set of model differences and combines them into a set of XBIs, which are then presented to the end-user. This operation is implemented by function which XBIS does[12].

X-PERT is a tool for identifying XBIs in web applications automatically, without the effort from the developer. X-PERT implements a comprehensive technique for identifying XBIs and has been results into an effective technique for finding real-world XBIs in our evaluation. The architecture of X-PERT consists seven phase which are discussed below [14]:-

The Model Capture module uses the Crawljax tool [4], which in tern uses the Selenium testing framework (<http://seleniumhq.org>) to explore the web application in the different web browsers. Crawljax is been expanded to prevent the model from its exploration along with the screenshot and DOM structure of every page. The DOM structure is obtained by querying the browser through its JavaScript interface. These properties include for each element, its textual content, style information, its XPath location, its size, and its coordinates on screen.

The Differential Model Store / Loader components are used to load the models to/from the `_le` system as XML `_les`. These components are implemented using the Object

serialization support in Java and are required essentially for the model capture and comparison of components to operate independently. For instance, model capture might be used to run to collect models from different machines, whereas model comparison may compare these models on a every single machine.

Graph Isomorphism Checker: This module performs an equivalency check between the graph based models from the two browsers to identify mismatched states and transitions across the two models.

The DOM Tree Matcher module matches corresponding elements from web pages across the different browsers by computing the match index metric. This metric considers the Levenshtein distance between the XPath of the elements, and is computed using the corresponding implementation in the Apache String Utils library.

Content Comparator: This module performs analysis of textual content of corresponding elements to find text-content XBIs. For finding image-content XBIs, it compares screen images of the corresponding elements on the web page by leveraging the OpenCV toolkit [1].

Structure Comparator: The layout of the page extracted by the model captures the module analysed by the Layout Analysis module to create alignment graphs, which represent the corresponding alignment of web page elements.

Report Generator: This module generates an HTML report tabulating the set of detected XBIs. It is implemented using the Apache Velocity library (<http://velocity.apache.org>). The generated reports enlarge jQuery along with the HTML5 Canvas for displaying and highlighting the XBIs on the screen images. Behavioural XBIs are presented in the HTML report by highlighting them on the models converted to SVG format using [14].

4. REASONS INCONSISTENCIES IN CROSS BROWSERS

Following are the reasons for inconsistencies in Web browser.

(i) **Cascading Style Sheets:** CSS is an acronym for cascading style sheets. Style sheet language deal with the formatting & changing the look (presentation) of document written in the Hypertext Mark up language. Cascading style sheet version 1,2,3 have some of the add-on features that help web developers to design a good website using the latest CSS-3 version whose features have compatibility with the cross browser. The CSS to work well in both the cross browser i.e. Mozilla Firefox and internet explorer should have height of 245 px to show perfectly on both the cross browser [15].

(ii) **HTML Tags:** HTML standard produces their own tags & attributes to build web pages for browser and to support languages for numerous cross browsers. The HTML tags supports “HTML extension” languages for different versions of browser. Browser supports newer versions which display newer presentation of web pages. Browser’s newer add-on features gives better display of web pages.

(iii) **Font Rendering:** Font rendering deals with the display of font on the screen how the font looks within the browser and shape of each letter, Word looks perfect due to font rending [8].

(iv) **Page Rendering:** Page rending deals with the behaviour & appearance of page in the browsers. It appears different in different browsers [19].

(v) **DOM:** Dom is an acronym for document object model. The structure overview of HTML document as request

received from web server to web browser is termed as document object model. In the Early 2000, the Microsoft Internet explorer and netscape navigator as well as Jscript and Java are engaged between the “browser war” to make their browser popular. Nowadays, modern browser version of Mozilla firefox supports document object modal level 2.

(vi) Scripting Languages: Scripting languages takes more time in running scripts than the compiled language program as scripting languages executes each instruction first rather than the compiled language in which codes are been written in C and C++. Scripting languages combines the codes of other compiled languages to turn and execute programs. The current version of scripting language supports JAVA in Mozilla and Netscape & visual basic script and JAVA script in internet explorer [20].

(vii) Plugin/Add-on: Add-on or plug-in are defined as additional features which are used for making the browser to become compatible with the current version available today. The add-on or add-in allow browsers to add some more features to the browser according to the client or user needs [12].

For example:-When user want to watch a video but the browser is not having a video player feature then, the add-on will provide functionality of the video player in the browser such that the user can watch the video in the video player by using the browser. Actually installing plug-in is not a big deal; many of these are freely available on the web [18].

(viii) Third Party Entities: Third party entities are third party applications that deals with the compatibility of web applications. For example:

- Web analytics package
- Social networking websites
- Embedded advertisements
- Message boards , forums & blogs
- Check-out processes
- Live chat
- RSS feeds
- Search plug-in
- Audio- player and embedded video player

4.1 Problem Statement

As we know that Cross-browser inconsistencies (XBIs) are a serious problem for web developers. Based on literature survey three problems are been identified that are relevant to the web browser platform testing and maintenance: 1) automated identification of inconsistencies in the same application's behaviour across multiple platforms, 2) Detecting features that are present in the application on one platform, but missing on another platform version of the same application. 3) automated migration of test suites and possibly other software artefacts across platform .Cross-browser automation tool is easy method for testing the cross-browser on different platforms. Version provides Cross-browser tools as free and paid cross-browser testing tools are available for testing the cross-browser. Depending upon user need, the compatibility of cross-browser testing tools is secured [14].

5. CROSS BROWSER TESTING TOOLS

Free Cross Browser Testing Tools: - There are some tools that allow you to view your website in multiple browsers with the click of a button. These tools can be helpful for viewing

your website at different resolutions. It's important to test at different resolutions so that we can ensure that users viewing your site on different devices will receive the same positive user experience. Following are some of the best, free cross-browser testing tools:

i) The Spoon Browser Sandbox: It allows browser to run on your system with out installation. Using this you can access any browser on the web. These support various browser & their versions are- IE browser opera 9 and opera 10 version. IE browser: IE6 IE7 IE 8 version. Google chrome: Google chrome, google chrome 5 google chrome 6[15].

ii) Browser Shots : Browser shot free cross browser testing tool is widely used due to its customization tools & features. It allows the user to test website on any platform independent operating system & web browser. It supports many browsers such as mine field, safari, firefox, & various versions of web browser [16].

iii) IE NetRenderer: This IE net render browser testing tool tests website on the Microsoft Internet explorer browser. It also allows add-on features of web pages on mozilla firefox[17].

iv) IE Tab: Internet explorer tap allows the software developers & testers to test how the web page is been displayed on the browser in a single click on chrome & firefox browsers. This feature allows to test quickly on the browser[18].

v) IE Tester: IE tester allows free cross browser testing tool to test website on browser like in window 7, window Xp operating system & various IE version like IE 5.5,IE6 IE7 IE8 IE9. It allows the browser to test website to check browser incompatibility [18].

vi) Microsoft Super Preview: Microsoft web super preview is a free cross-browser tool which checks the website in different web browsers. It allows the software to test & debug the layout issues that arises on different platforms .Web super view allows the web pages to test on IE6 IE7 IE8 version depending on which browser your system is used[19].

Paid Cross Browser Testing Tools:-

i) Browsera:- Browsera is used to test the website scripting error & web page layout. It is automated browser compatibility testing tool [20].

ii) Adobe Browser Lab:-Adode Browser lab allows web pages to be tested by zooming the content of web pages, browser alignment and layout screen shots. Web pages are tested on various versions of chrome (windows), safari (Mac OS X), firefox (Windows and Mac OS X), internet explorer (Windows) [21].

iii) BrowserCam:- With the help of Browsercam web pages can be viewed automatically or manually on different browsers. 200 screen capture screen shots are available as free trial version for a day[22].

iv)Browserseal:- Browserseal is a fast and easy to use cross browser testing and screenshot tool. Browser seal free trial version is available for one screen shot per session and two browser (Internet Explorer & firefox). Browser seal supports google chrome , safari, opera, firefox & internet explorer[24].

v) Cross Browser Testing:-Cross Browser Testing tool test the websites written in Ajax, JavaScript & flash functionality to check the website on different platforms & operating

system. One week free trial version of this software is available [25].

vi) X-PERT:- X-PERT can work with any web applications that runs on desktop browsers. Due to X-PERT's capability of analysing the client-side of such applications, it is agnostic to any server-side technology. X-PERT is written in Python and Java and can run on a variety of desktop operating systems, including Windows, Mac OS X, and Linux. X-PERT implements a comprehensive technique for identifying XBIs and has been shown to be effective in detecting real-world XBIs in our empirical evaluation[12-13].

vii) Cloud Testing:-Cloud testing tool tests websites on real operating browsers in cloud environment. This allows all browsers to test websites safari, internet explorer, firefox [25].

6. CONCLUSION

This paper presents a literature review on existing cross browser inconsistencies. As the technology is changing rapidly, the application designers believe to give the best features and functionalities to the applications. Users have performed cross browser on different platforms and get results on web browser. Designers cannot assume that their application will run well and work for all browsers without cross browser testing. Hence to avoid loss of business and reputation it is very important to pay attention to cross browser issues. In this paper we have pointed out a few important limitations regarding cross browsers inconsistencies. As the problem it is easy to plan out the areas that should be cross verified.

7. REFERENCES

- [1] R. T. Fielding and R. N. Taylor, "Principled Design of the Modern Web Architecture," ACM Trans. Internet Technol., vol. 2, pp. 115–150, May 2002.
- [2] A. Grosskurth and M. W. Godfrey, "A Reference Architecture for Web Browsers," 21st IEEE International Conference on Software Maintenance, pp. 661–664, September 2005.
- [3] C. Eaton and A. M. Memon, "An Empirical Approach to Evaluating Web Application Compliance Across Diverse Client platform Configurations," Int. J. Web Eng. Technol., vol. 3, no. 3, pp. 227–253, 2007.
- [4] Microsoft, "Expression web," <http://www.microsoft.com/expression/products/WebOverview.aspx>, May 2010.
- [5] Stuttard, D., & Pinto, M. (2011), "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws", John Wiley & Sons.
- [6] Stackoverflow, <http://data.stackexchange.com/stackoverflow/query/77488/posts-for-Cross-Browser-Issues>, August 2012.
- [7] J. Rode1 and M. A. P.-Q. Mary Beth Rosson, "The Challenges of Web Engineering and Requirements for Better Tool Support", Technical Report TR-05-01, Computer Science Department, Virginia Tech., 2002.
- [8] Wen-Kui Chang, Shing-Kai Hon, "Assessing the Quality of Web-Based Applications via Navigational Structures", IEEE Multimedia, Volume 9, Issue 3, Jul-Sep 2002.
- [9] Patidar, C., "Cross Browser Testing: A Challenge for Web Testing", International Journal of Scientific Research in Computer Science and Engineering, Vol-1, Issue-3, pp.-62-64, 2013.
- [10] S. Roy Choudhary and A. Orso, "Webdiff: Automated identification of cross-browser issues in web applications," in ICSM '10: Proceedings of the International Conference on Software Maintenance. IEEE, September 2010.
- [11] Mesbah, A., & Prasad, M. R., "Automated cross-browser compatibility testing". 33rd International Conference on Software Engineering, ACM Proceedings, pp. 561-570, 2011.
- [12] Choudhary, S. R., Prasad, M. R., & Orso, A., "Crosscheck: Combining crawling and differencing to better detect cross-browser incompatibilities in web applications", In Software Testing, Verification and Validation (ICST), 2012 IEEE Fifth International Conference on, pp. 171-180, 2012.
- [13] Choudhary, S. R., Prasad, M. R., & Orso, A., "X-PERT: A Web Application Testing Tool for Cross-Browser Inconsistency Detection", 2014.
- [14] S. Roy Choudhary, M. R. Prasad, and A. Orso. X-PERT: Accurate Identification of Cross-browser Issues in Web Applications. In Proceedings of the 2013 International Conference on Software Engineering, ICSE '13, pages 702-711. IEEE Press, 2013.
- [15] Tom Dahm, Avoid the Cutting Edge[Online] Available: <http://www.netmechanic.com/products/Browser-Tutorial.shtml>.
- [16] Webcredible [Online]. Available <http://www.webcredible.co.uk/user-friendly-resources/css/>
- [17] Google Code, Optimize Browser rendering [Online]Available: <http://code.google.com/speed/pagespeed/docs/rendering.html>
- [18] Adobe, "Browser lab," [Online] Available: <https://browserlab.adobe.com/>, Dec 2010.
- [19] By hallvord r. m. steen, same dom errors, Different Browser Interpretations ,available at: <http://thinkvitamin.com/>
- [20] Louis Lazaris,7 JavaScript Differences Between Firefox & IE. [Online] Available:<http://www.impressivewebs.com/search/7+JavaScript+Differences+Between+Firefox+&+IE>
- [21] Important Plugin Information [Online] Available: <http://devedge-temp.mozilla.org/viewsource/gecko-compatibility/#intro>
- [22] Netrenderer [Online] Available: <http://ipinfo.info/netrenderer/>
- [23] How does your website look at safari [Online]. Available: <http://www.webdevlab.com/app/screenshots>
- [24] Automation Browser Compatibility Testing. [Online] Available: <http://www.browsera.com/>
- [25] Codefusion, 18+ handy tools to check cross-browser compatibility issues [online]. Available: codefusionlab.blogspot.com/.../18-handy-tools-to-check-crossbrowser.html.