

Compatibility testing: A necessity for web apps



Information

Computers have become integral parts of our lives. There are multiple software applications that have been developed to assist in teaching people valuable skills that bring resources and ease to their careers, work, shop, and in many other fields of their life.

People are buying a lot of stuff online these days. These people often use applications to browse for their desired products and then order it. While selling the product or software, the seller has to keep in mind that the product that is being sold is bug-free. Otherwise, the person on the other side of the application would lose the trust they have on them. Defective software might also result in the loss of a lot of capital that went into the software for its development, testing and launch.

To stay competitive in the market, the sellers have to ensure that the end-user is getting exactly what they paid for. The seller and the developer have to ensure that the product they are selling is flawless in terms of the quality that was assured to the consumer. There are a few ways of doing this. These methods are known as *Testing*. Testing comes in different stages. There are several stages to testing such as quality, compatibility, reliability, and delivery.

When most developers hear the word “compatibility”, most of them don’t know what it means. Hence, we are here to explain it to you. Compatibility is the ability of the software program to run on different kinds of environments without misbehaving. These environments may include different brands of hardware, software, operating systems and updates.

Most Quality Assurance teams test their software on all of their available equipment, but this, unfortunately, doesn’t cover all of the types of systems that are available to target users in the market. Hence, if the software is sold to a consumer who does not have a reputed brand of hardware or operating system, the software won’t work well there. Hence, in situations like these, social media becomes a powerful weapon in the hands of that particular software user. They can use social media to taint and tarnish the reputation of the software company even if the software works perfectly well in other operating systems it was tested on.

Compatibility is the ability to live and work together without any discrepancy. Compatible software also has the ability to work on inherently same software setups too. For example, if Google.com site is compatible, then it should open in all browsers and operating system. Furthermore, google.com must not have and show any errors to the user who is using it on different kinds of operating systems or using a browser that is not very popularly used. It is the ability of that website to work and function on an arena of different scenarios and environments that constitutes compatibility.

Compatibility can also be defined as non-functional testing to give assurance to the end-user. It is made to determine accurately whether your software application or product is well-made and programmed enough to run in different environments of different computers. These environments might constitute differences in browsers, database, hardware, operating system, mobile devices, and networks; most of which depend from user to user.

The application could also be affected as a result of different editions and versions, resolutions, connectivity speed and configuration etc. Consequently, it is important to test the software in all possible methods so as to reduce the chances of failures and overcome the potential of bug’s leakage. As a rule of thumb, compatibility tests should always be performed in real environments instead of virtual environments.

Compatibility testing can be of two types - forward compatibility testing and backward compatibility testing.

- Operating system Compatibility Testing - Linux, Mac OS, Windows
- Database Compatibility Testing - Oracle SQL Server

- Browser Compatibility Testing - IE, Chrome, Firefox
- Other System Software - Web server, networking/ messaging tool, etc.

Purpose of Compatibility Testing

There are a few very strong reasons why we test software for compatibility. It helps us to make sure that the customers who are the end-users of the software can install and run their software with ease and without bugs. This type of compatibility testing makes sure that the overall operation of the software does not get hampered in any way. There are users who use the software on different devices and for different operating systems. Hence, we need to make sure that compatibility testing takes care of the following things:

1. The software can be installed and can function on multiple different, distinct and unique environments
2. The testing needs to ensure that some variations in screen size, resolution, and operating systems of the consumers do not corrupt the software or make it run in a manner that was not intended by the developers
3. The minimum specs required to run the software – The software is tested against various hardware interfaces like graphics cards and RAM types to gather data on what suits the software most. Hence, in this way, developers and the software team can gather information related to the software in-depth and relay the same to the consumer to ensure a perfect experience

This type of compatibility testing makes sure that your software is fit to be run on different environments that the end-user could be using. It is also used to figure out how many configurations the software can hold before it starts developing glitches and errors during runtime.

Hence, this is the purpose of compatibility testing.

Why Conduct Compatibility Testing

Softwares are created to solve problems. They are often employed by big corporations that have a lot of data and resources that rely on the smooth functioning of this software. Hence, if a software program does not run like it was intended to, the resulting consequences could spell a load of terror in regards to financial and social loss. Hence, if your software is not compatible with a lot of different operating systems and hardware components, your market gets limited and you cannot scale your product. Usually, loss of scale means that you won't be able to sell enough of the product to actually make net

profits. Hence, these are the reasons why you should consider getting software testing done for web applications especially.

There are a lot of businesses who do not conduct compatibility tests on their software and regret it later. When their software encounter compatibility issues after the software have been launched to the public, it is too late to change the core code. Hence, due to the lack of a simple precaution, they fail to scale their product and get it to the hands of several customers who need the software.

Conducting compatibility tests before the software is released can help your product stay away from a number of different pitfalls. If your Quality Assurance team ever needs help with the workload, you also have the option of considering to outsource your compatibility testing to labs that do this kind of work. In this way, you are taking precautions and making sure that you don't take up too much time releasing the product.

Compatibility Testing: How to conduct it for your software

If you're wondering how to conduct the compatibility testing for your software before you release it to the public, here are some things that you will find helpful:

1. The beginning and initial phases of compatibility testing are to accurately define the set of environments; configurations or platforms that the application is expected to work perfectly on
2. To conduct perfect and reliable compatibility testing, the tester should have adequate expertise in knowing about the platforms, software and hardware to perfectly understand the expected behaviour of the application under the different conditions that it will be put under
3. The different environments need to be set up for testing with different platforms, devices, operating systems and networks to find out whether or not the software runs properly under different operating systems and conditions.
4. After all the required data is collected, the tester is to report the bugs and fix the defects. It is after the fixing of all of such defects that the tester has to test the software under the same circumstances again. If there are no bugs to be found, the software is said to be perfect and adequately compatibility tested. However, the tester has to make sure that the testing is done in lots of different and varied environments to really challenge the software to perform its best under such conditions. Since there is no easy way to figure out what devices the software will be run on, it is smart to test everything and then move forward.

For instance, to test the compatibility of site ebay.com, download different versions of the internet browser Firefox and install them one by one and test the eBay site. eBay site should behave equally the same in each version.

Types of Compatibility Tests

Let's look into compatibility testing types. These are the different types of compatibility software testing that you should know for your software.

- **Hardware:** It checks whether the software is compatible with different hardware configurations that the consumers would use
- **Operating Systems:** It checks whether the software is compatible with distinctive Operating Systems like Windows, Unix, Mac OS etc. Most operating systems that are used by end-users are fairly common. Hence, this shouldn't be the hard part. However, your software has to be designed in such a way that it works specifically on open-source platforms like Linux.
- **Software:** It checks whether your fully developed software is compatible with other third-party applications and software. For instance, MS Word application should be able to work in sync with other software like MS Outlook, MS Excel, VBA etc.
- **Network:** This involves the evaluation of the software's performance on a system that is connected to a network with varying parameters such as bandwidth, connectivity speed and capacity. This testing also involves the testing of that software with all other applications with respect to different speeds and connectivity issues.
- **Browser:** Browser compatibility involves checking your software's compatibility with different internet browsers like Firefox, Google Chrome, Internet Explorer etc.
- **Devices:** This involves the checking of your software with different hardware components that come with different devices. The devices might include Bluetooth devices, printers, scanners and other connections like USB and AUX.
- **Mobile:** Checks your software's compatibility with mobile platforms like Android, iOS etc.
- **Versions:** This testing verifies the software application with different versions of the same software. For instance, checking your Microsoft Word to be compatible with Windows 7, Windows 7 SP1, Windows 7 SP2, Windows 7 SP3 etc.

Depending on the various classifications about where the testing is done on the application, there are two types of version checking in Compatibility Testing. They are:

Backward Compatibility Testing

Backward Compatibility Testing is a flawless technique that works to verify the behaviour and compatibility of developed software with older editions of the software. Backward compatibility testing is very much predictable as almost all the changes that had taken place in the previous versions are already known to the tester performing the tests right now.

Forward Compatibility Testing

Forward Compatibility Testing is a process that verifies the behaviour and compatibility of the developed with the newer versions of the software. Forward compatibility testing is a little more complicated than backward compatibility testing because the changes that will be made in the future versions of the software cannot be known at this stage in the testing process.



Common Compatibility testing defects

Here are some common compatibility testing defects that might arise when you're testing your software in such away. These defects can be totally avoided and even rectified if the proper procedures are known well. Here are some of the defects that may arise as a result of software testing:

- **Changes in the user interface** – This means that the look and feel of the software might change following the bug removals after software testing
- **Change in the font size**
- **Alignment related issues** – The text might seem distorted sometimes
- **Change in CSS style and colour**
- **Scroll bar related issues** – Some of the hardware, usually related to the scroll bar, might face some problems
- **Content or label overlapping**

- **Broken tables or Frames** – which might also ruin the feel and look of the software for the user

Here's how to pick what you need to test for compatibility testing

- Make notes of the most crucial testing parameters for the software where you think it has the potential of behaving weirdly.
- The best way to do this is to analyse the requirements and cross-check with the client or customer for the browser matrix. It is usually smart to let the customer decide which browsers, operating system and version they would like you to test the software on. It also reduces your workload a lot.
- With the help of Google Analytics or an alternative statistical analysis system that you can set up on your application, you can get clear statistics of widely used browsers with their version and operating system to make sure you're not skipping any bases when you're scanning your software for bugs during compatibility testing.

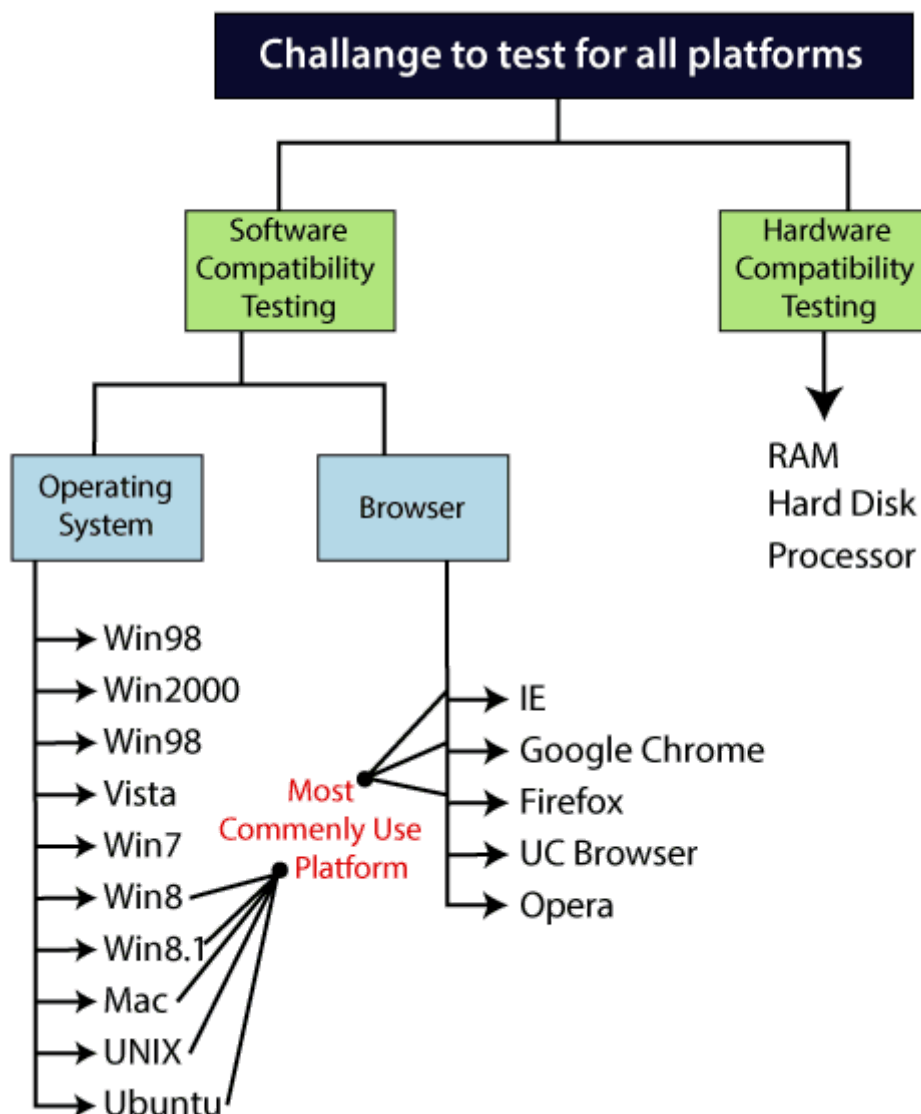
Select the pages that you want to test

- Sieve out the focused URLs and pages of your application. The selection of the pages totally depends entirely on your application. You will need to consider the most frequently used modules as an integral part of the compatibility test. If your software only consists of a certain specific and particular format, it can still work as you're only using a single part for the compatibility testing.



What is the most challenging part of compatibility testing with reference to software?

The most important and challenging part of the compatibility testing is to figure out what the necessary and crucial needs of the customer and the end-user are. Since compatibility cannot be checked on each and every system out there, it is very important to figure out what the client wants and then make sure that all those conditions are fulfilled. Since testing on each and every existing system is not possible, we must do our best to test the software on all the other systems that the consumers use commonly to find out the bugs that may occur.



Compatibility testing bugs and issues

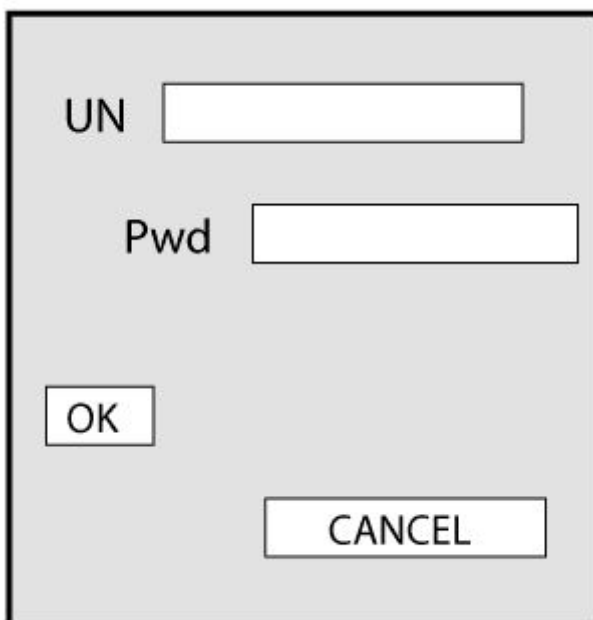
These compatibility testing bugs are the ones that may occur in one environment but may not occur in the other. Hence, they are largely environment and hardware/software specific bugs. Hence, they are known as compatibility testing bugs and issues.

Here are some of the issues that are faced by compatibility testers during runtime. These issues, as aforementioned, might be confined and/or specific to one of more environments, either software or hardware. Here are some compatibility testing issues:

- Alignment issue
- Overlap issue
- Scattered issue
- The issue of look and feel

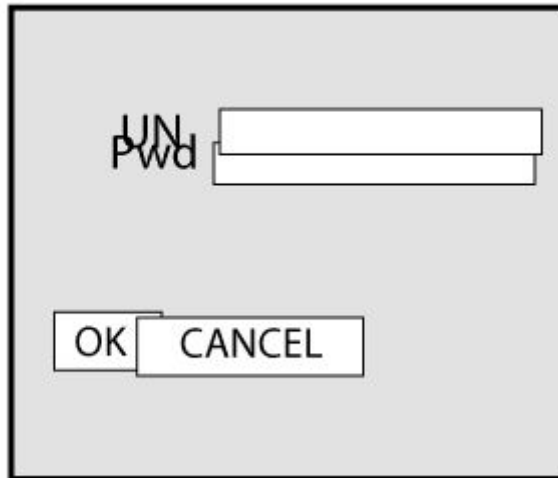
The alignment issue

The alignment issue is the one in which the various different components of a page or the software are not aligned in any manner. This makes the interface of the software seem poorly developed and haphazard. Look at this example below:



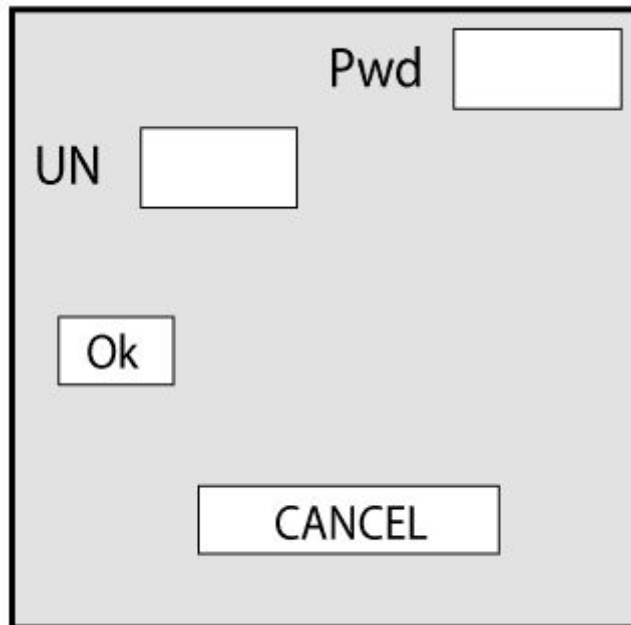
Overlap issue

When certain elements of the software and the interface overlap each other when viewed as the user, that can create a confusing picture, making working with the software difficult and decreasing efficiency. Look at the example below:



Scattered issue

The scattered issue comes up when the software is not compatible with different kinds of browsers. These kinds of browsers are used by the end-user but the software itself is not made for them. Hence, these scattered components are visible.



Compatibility issue: When the software's features are not working in one operating system but are working fine in the other operating systems, it is usually a sign of development in one single environment. Hence, the software tends to run fine on the environment it was developed in and does not run that well in the others.

Functionality issue: These issues occur when certain features do not work in all operating systems or platforms.

Compatibility testing tools

Some of the most commonly used compatibility testing tools are as follows:

Lambda Test

This is an open-source browser compatibility testing tool that usually works in the cloud environment. Thanks to this tool, you can test web applications on nearly any mobile browser or desktop browser. Lambda Test has a screenshot feature, which allows testers to take the full-page screenshots the web pages that they are testing.

Using this tool, you can check the compatibility of your web application on a mobile browser. Hence, it gives you the connectivity and the resources to check whether your software works in a mobile environment well or not.

Browsers Stack

This tool helps you to test the websites and mobile applications compatibility over numerous different browsers and platforms.

Thanks to this software, you can test a web application in various different mobile and web browsers and mobile or web applications such as Android and iOS.

BrowseEMAIL

This tool can run your application on different operating systems such as Linux (open source), Windows, and macOS. BrowseEMAIL is a cross-browser testing tool which is used to test the application on all local machines and in all local networks.

TestingBot

TestingBOT is used to perform the required compatibility tests on various web browsers like Firefox, Chrome, Edge, IE, Safari etcetera. Thanks to this software, you can compare the screenshots from the multiple browsers and hence, test the responsive layout of the application to check whether it performs well in all those simultaneously or not.

Conclusion

Compatibility testing is used to make sure that the software application is working fine with respect to any and all environments it is subjected to. Since the developers cannot know what kind of environments the software will be run on, what the hardware components will be and what operating system the user will use, we have to make sure that the software runs well every single time. Compatibility testing also makes sure that all the other aspects of the

system, like the use of browsers, databases, hardware and software is also compatible with the software. of time to confirm browser and operating system compatibility.