Summary of web application testing methodologies and tools

Gowri Shankar Palani
Published on March 29, 2011

Web application testing methodologies

In this article, *web application* refers to all applications that are accessed through browser. This section outlines some of the testing methodologies you can use to test your web applications.

Usability testing

For an application to be effective, the user interfaces should comply to standards Follow globally accepted conventions wherever applicable. For example, color coding conventions indicate that red is used to stop a process and green to start one.

Usability testing plays a pivotal role with applications that are designed to make manual tasks easier. The applications should comply with accessibility standards For example, the widely used Captcha code has an option for spelling for people who are visually challenged.

Keep the following guidelines in mind when undertaking usability testing:

- Ensure proper navigation between web pages.
- Be sure to have a site map.
- Use appropriate color combinations and best practices.
- Avoid over-crowded content.
- Practice user friendliness to all types of users, from novice to expert.
- Provision support for physically challenged people.

User acceptance testing

The objective of user acceptance testing is to make sure your application meets the expectations of the user. It ensures that the application is fit enough to be deployed and used effectively. The following are tips for user acceptance testing

- Ensure browser compatibility.
- Make sure that mandatory fields are given data in forms.
- Check for time outs and field widths.
- Be sure that proper control is used to feed data. For example, when requestin gender information, use an option button.

Alpha and beta testing are the two types of user acceptance testing.

Alpha testing
Testing in a development environment by developers.

Beta testing
Testing in a deployment or client environment by end users.

Performance testing

Performance testing on web applications measures the performance under various scenarios. Performance tests include:

Stress testing
To determine the maximum performance limits of an application.

Scalability testing
To find out how adaptable the application is to changes in software and hardwa

Load testing
To get an idea of how the application behaves under a heavy load. This test yield information and details about memory usage, CPU usage, and so forth.

Security testing

Security testing for your application is very important if data leaks or modificatio are unacceptable and intolerable. For example, if a university application include the academic grades of a student, then security testing should ensure that the system cannot be hacked. For e-commerce applications, which sometimes invo banking transactions, security testing is critical. It should also ensure that sufficient authentication and authorization mechanisms are in place.

Security testing can be static or dynamic.

Static
Static testing involves doing a static code analysis to check for any vulnerabilities The goal is to understand the code flow and check for security threats by walkir through the code.

Dynamic
Dynamic testing entails running the application to see if the response is as expected for the associated request. It is very similar to black box testing.

Functional testing

Functional testing ensures that individual functions are working well. Test cases should ensure that boundary conditions are tested. Invalid inputs should prompt appropriate error messages.

In web applications, functional testing can range from testing whether links are working to checking whether changes made by users in a web page are reflecte in the database. Some of the functional tests for web applications include:

- Database testing
- Configuration testing
- Compatibility testing
- Flow testing

Interface testing

Conduct interface testing to ensure that individual components are connected properly. The output of one module should be fed to the intended module withou any issues. Interface testing plays a vital role for your applications that are developed to work on multiple platforms. The following are considerations to kee in mind during interface testing:

- Ensure that data flow occurs smoothly and as expected between modules in single application and between applications.
- Ensure that the interfaces exposed by components are generic and extensibl They should be able to accommodate changes to the components while remaining backward compatible.

Web application testing tools

This section summarizes the various testing techniques and tools available for testing web application technologies. See Related topics for links to the tools mentioned in this article.

Basic web applications that have primarily HTML content can be tested with the following tools:

Hypertext markup language (HTML)

Rational Functional Tester (RFT)	An IBM tool that can test several web application technologies, including HTML, using a record and playback method. You can also edit and customize the recorded script to suit your needs.
Selenium	An open source project based on a record and playback test framework.

Testing your client-side script is essential for proper validation. JavaScript can be tested using the following tools:

JavaScript

JSpec	A testing framework with its own grammar and preprocessor. It supports multiple modes of testing.
RhinoUnit	An Ant-based JavaScript testing framework for performing unit testing.

Dojo testing

For Dojo, Dojo Objective Harness (DOH) testing provides both command line and browser support. DOH is flexible and easy to use.

For testing RIA-based applications such as Flex, you can use the following:

Adobe Flex

FlexMonkey	A record and playback-based testing framework for Flex applications. It is an open source tool.
FlexUISelenium	An extension to the Selenium test framework for testing Flex applications.

You need to test the HTML links in your website to be sure that none are broken. Many online and offline tools are available for this type of testing, including:

HTML links

WebLight	A tool used for testing broken links and non-standard HTML.
LinkTiger.com	An online analysis of broken links for your websites.

Corresponding test frameworks are available for server-side scripts such as PHI ASP, and JSP.

Server-side scripting

PHPUnit	A unit testing framework for PHP based on JUnit for Java.
ASPUnit	A unit testing framework for ASP.
Cactus	A JSP test framework from Apache.

A website should be resistant to hacking attempts and denial of service attacks. Security threats can include SQL injection, command injection, cross-site scripting, and server configuration errors. An array of tools is available for securit testing.

Web application security testing

Nikto	An open source tool used for checking web servers for security flaws.

Rational AppScan	An IBM tool that supports automatic test creation and modification. It can also be easily integrated into existing testing environments.
Acunetix Web Vulnerability Scanner	A tool that helps check cross-site scripting and other similar vulnerabilities.

Make sure that the web content for your site is uniform in all browsers. Be aware that some functions are browser-dependent; users will expect and use functions specific to their chosen browser. Several tools are available for browser testing, including:

Browser compatibility testing

Microsoft Expression Web SuperPreview	A tool that can perform various tasks, such as checking element alignment, showing attributes of DOM elements, and rendering in different sizes. The overlay layout feature helps you compare two browser views of a page.
Adobe Browser Labs	An online-based solution from Adobe that lets you test a website in various browsers and versions. It has features to change the page size and take snapshots of the screen.

Downloadable resources

PDF of this content

Related topics

- Read about the features and benefits of Rational Functional Tester.

- Learn more about the Selenium suite of tools to automate web application testing.
- Get an overview of Jspec (JavaScript Testing Framework).
- Read all about RhinoUnit.
- Get an overview of the Dojo testing framework and other dojo documentatior
- FlexMonkey is an Adobe AIR application used for testing Flex and AIR-based applications.
- View examples of FlexUISelenium testing.
- Learn how WebLight can help maintain XML site maps and find markup, CSS and link problems so you can maintain error-free, fully indexed sites.
- Use LinkTiger for testing to identify broken links.
- Read about unit testing with PHPUnit.
- Learn how Cactus can be used to test JSP.
- Read about ASPUnit, a unit testing framework based on the architecture of JUnit.
- Explore how to work with Nikto, an open source (GPL) web server scanner th performs comprehensive tests against web servers for multiple items.
- Learn how Rational AppScan offers static and dynamic security testing in all stages of application development.
- Adobe BrowserLab can help with cross-browser testing.
- Get more information about Microsoft Expression Web.
- Download Selenium.
- Download RhinoUnit.
- Download IBM product evaluation versions or explore the online trials in the IE SOA Sandbox and get your hands on application development tools and middleware products from DB2®, Lotus®, Rational®, Tivoli®, and WebSphere®.

Comments

Subscribe me to comment notifications

Sign in or register to add and subscribe to comments.