

Web Application Testing: A Survey

Geetha J K, Monika M

Abstract: Web applications have become more crucial in many fields. This paper presents some of the basic testing techniques in the field of testing Web applications developed in current years. Some challenges related to performance testing are also outlined.

Keywords: Functional testing, Usability testing, Navigation testing, Security testing, Interoperability testing.

I. INTRODUCTION

Web applications give us new types of applications and also provide new way to set up software solutions to end users. Web applications make use of number of new technologies, languages and programming models. Current web applications are complex. It is the integration of several back end software components and interactive programs with complex graphical user interface. Web applications have more than one front end GUI's and more complicated than simple HTML web pages. Due to this nature the testing of web application is very difficult.

Dominance of change is one of the key challenges of web testing. Change may be due to ever changing platform and configurations, technologies, business model or customer expectations. Main difficulty is to find the root cause of errors which may have occurred due to network, operating system, hardware and software.

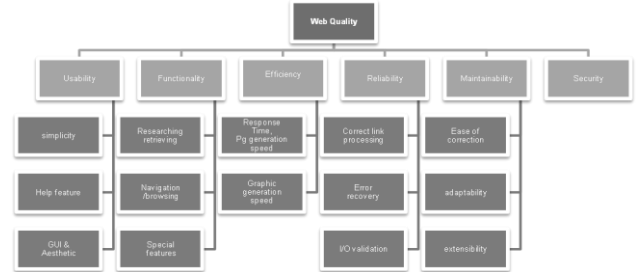
To meet these challenges here are some of testing techniques.

II. WEB APPLICATION TESTING ETHODOLOGIES

This section describes various testing methodologies to test the web applications [1][2].

- Functional testing
- Usability testing
- Performance testing
- Inter operability testing
- Security testing

Graphical representation of testing types.



2.1 Functional Testing

Functional testing guarantees that individual functions works well and appropriate error messages should be reported for invalid inputs.

In web applications, functional testing test cases ranges from checking whether links are working to ensure whether changes made by end users in a web page are echoed in the database.

Some functional testing for web application includes database testing[5], configuration testing[3] and compatibility testing[3].

2.1.1.1 Database testing:

Database is the most critical component of the web application and should be tested thoroughly. Testing actions involves in checking whether errors are reported during the execution of queries, checking the queries response time and modifying them if required, data Integrity should be maintained during creating, updating or deleting data in database and retrieval of test data from the database is shown accurately in the web application.

2.1.2 Configuration testing:

Some of the questions to be answered on the server side are,

- Is server OS completely compatible with Web application?
- When the web application is operational, are directories, system files and related system data created correctly?
- Without degrading the performance do system security measures like firewalls or encryption let the web application to execute and service users?
- Whether it has been tested with the configuration of distributed server?
- Integration of web application with database software is proper and is it sensitive to various versions of database software?
- Web application scripts execution in server-side is proper or not?

Revised Manuscript Received on 30 January 2013.

* Correspondence Author

Geetha J K*, Department of Computer Science & Engineering, Channabasaveshwara Institute of Technology, Tumkur (Karnataka), India

Monika M, Department of Computer Science & Engineering, Channabasaveshwara Institute of Technology, Tumkur (Karnataka), India

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

- The effect of system administrator errors on web application operations is examined?
- Differences in configuration of proxy servers have been addressed with on-site testing?

The components to be tested in client side are,

- Hardware—CPU, memory, storage and printing devices
- Operating systems—Linux, Macintosh OS, Microsoft Windows, a mobile-based OS
- Browser software—Internet Explorer, Mozilla/Netscape, Opera, Safari, and others
- User interface components—Active X, Java applets and others
- Plug-ins—QuickTime, RealPlayer, and many others
- Connectivity—cable, DSL, regular modem, T1

As the number of configuration variables increases complexity also increases, so we must reduce configuration variables to a manageable number.

2.1.3 Compatibility testing:

One of the important phases of web testing is the web site compatibility. Some compatibility tests to be executed are, Browser compatibility, Operating system compatibility, Mobile browsing, prints options.

Browser compatibility:

Many applications are browser dependent. The web pages should be compatible with different configurations and settings of Different browsers. The coding of web site should be compatible with cross browser platform. If you are using scripting language like java scripts or AJAX calls for implementing User interface functionality, along with performing security checks or validations more stress should be given on browser compatibility test. Web application should be tested on different browsers with different versions such as, Firefox, Internet explorer, Netscape navigator, Safari, Opera, Chrome browsers.

OS compatibility:

Some functionalities of web application may or may not be well-suited with all other operating systems. New technologies like graphics designs, different API are used in web development may not be offered in all these Operating Systems. Web application should be tested on different version of operating systems like Windows, Chorus, VX-works, UNIX, MAC, Linux, and Solaris.

Mobile browsing:

Now a day's Web applications are widely used in mobiles. So we must ensure that web pages are compatible on mobile browsers.

Prints options:

If web page-printing options are provided then make sure those fonts, alignment of page, and graphics getting printed properly. Pages should scale to size of the page or as per the user mentioned size in printing option by the user.

2.2 Usability testing

To build an effective application, the user interfaces should abide to the standards. Globally accepted conventions should be followed wherever necessary. For example, colour coding conventions indicate that a hyper link in red is already visited by the user if in blue it's not visited.

Usability testing [3][4] plays a key role for those applications that make manual tasks ease. The applications should conform to accessibility standards. For the visually or physically challenged, one of the ways of navigating a site is through the use of Access Keys [8].

The guidelines to be kept in mind when undertaking usability testing.

- Proper navigation between web pages should be ensured.
- A site map should be there.
- Use suitable combinations of colours and best practices.
- Avoid over-congested content.
- User friendly to every users, from beginner to expert.
- Proper support for physically impaired people.

2.3 Performance testing

Performance testing [1][4] is used to measure the performance of web applications under various scenarios. It includes:

- Stress testing
- Scalability testing
- Load testing

Stress testing:

The maximum performance limits of an application and usage characteristics should be determined. Under extreme conditions these scenarios should be considered. Workload profile characteristics, current peak load capacity must be determined. Disaster-risk like earthquakes, blackouts must be assessed for hardware, network architecture and data.

Scalability testing:

Scalability testing helps to find out how the application is adoptable to software and hardware changes.

Load testing:

The throughput required to support the expected peak production load, capability of a hardware environment is determined. The adequacy of a load balancer is evaluated. It detects functionality errors under load and concurrency issues.

In load testing phase data required for scalability and capacity-planning is collected. It helps to determine how many users can use the application before its performance breaks down. Maximum load that hardware can attain is determined. The main goal is to determine the overall throughput using the formula,

$$P=N*T*D.$$

Where,

N is different loading conditions,

T is the number of concurrent users,

D is the number of on-line transactions per unit time.

2.4 Interoperability testing:

Interoperability [7] is one of the key issues to be considered in testing web applications. It means the ability of the software system to share information among different operating systems, computing components, networks and applications. Interoperability testing is a way of assessing a component, application, system or service that it performs as expected while interoperating with others.

The major challenging factors are, diverse information, data retrieval and operating systems, Technical level issues (multiple protocols etc), Heterogeneity issues, Semantic issues, Organizational level issues, Issues of different Implementation standards, Seamlessness, Performance, Security, Reliability, Scalability.

There are many different approaches to solve the challenging factors of interoperability.

The major testing approaches are,

- Functional testing
- End-to-End integration testing
- Operational testing
- Path-based testing
- Fault-based testing
- Model-based testing

2.5 Security testing:

Security testing [6] is vital for e-trade website that store sensitive customer personal information like passwords. The basic concepts like confidentiality, integrity, authentication, authorization, availability need to be covered by this testing.

Testing Activities include-

- Unauthorized access should not be permitted.
- Without appropriate access restricted files should not be downloaded.
- After prolonged user inactivity, sessions should be automatically killed.
- Website should move to encrypted Secure Sockets Layer pages, if it is SSL certified.

II. CHALLENGES

Performance testing:

Some functional defects may not be detected that only appear under load, and if not carefully designed and validated, it may result in inadequate performance characteristics. Some degree of uncertainty in the results will be present unless some tests are conducted on the production hardware-the same devices used by the users.

Load testing:

It does not primarily focus on response speed; results of other related load test should be compared.

Stress testing:

Since it is impractical by design, test results may be dismissed by some stakeholder. Major challenge is to decide how much stress is worth applying.

III. CONCLUSIONS

All web applications are heterogeneous and “dynamic” in nature so testing activities should straightforwardly reliant on the implementation technologies of these applications, while other features may be reused with a lessened adaptation effort.

REFERENCES

1. Roger S Pressman, Software Engineering A Practitioner's Approach, 5th ed., McGraw Hill. International Edition.
2. William E. Perry. Effective Methods for Software Testing, 3rd ed., Wiley publishing, Inc, Indianapolis, IN46256.
3. Ron Paltson, Software testing, 2nd ed., Pearson Education, Inc and Dorling Kindersley Publishing Inc.
4. Srinivasan Desikan, Gopaldaswamy Ramesh. Software Testing Principles and Practices, 1st ed., Dorling Kindersley (INDIA) Pvt.Ltd.
5. Louise Tamers, Introducing Software Testing, Dorling Kindersley (INDIA) Pvt.Ltd.
6. Henry Bequet, Meeraj Maidoo Kunnumpurath, Sean Rhody, Andre Tost, Beginning Java Web Services, Wrox press Ltd, Arden House,1602 war wick road, Acocks Green, Birmineham.
7. Fang Youyuan and Gu Tianyang, "Research on Web Service Interoperability Testing Framework with Conformance Testing", World Academy of Science, Engineering and Technology 45 2010
8. <http://wordpress.org/support/topic/lesson-web-page-design-and-accessibility>