Volume 3, Issue 3
June 2014

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

A Literature Survey on Evaluation and Comparison of Regression Testing Techniques for Web Applications

[1]Muhammad Hassnain, [2] Anjum Abbas

[1]Department of Computing and Technology, Abasyn University Islamabad Campus (AUIC) Pakistan

[2] Department of Computing and Technology, Abasyn University Islamabad Campus (AUIC) Pakistan

E-mail: [1]hassnain.pk77@gmail.com, [2]abbas.anjum@gmail.com

ABSTRACT

Regression testing verifies that functioning of web applications remains after modifications in the web services. With the aim to find a base for the future study in collaboration of academia with the industry, a survey paper on evaluation and comparison of regression testing techniques has been conducted. There are total twenty papers identified for this survey. Overview of the regression testing techniques presented by researchers is evaluated for the test selection. None of the technique has been found to be clearly superior as results depend upon the several factors.

Keywords: *Regression testing, web service, slicing technique, DOM, regression testing selection*

1. INTRODUCTION

Process of retesting the software that has been already modified is known as regression testing. Regression testing is mostly used for the development of commercial software, which constitutes an essential part in the development of software. Many developers do not believe that small changes made in one part of a system can cause problems for other parts of a system. Regression testing concerns mostly such type of issues, which arise from the modification of a system. When and why regression testing is performed to ensure the correct working of software after changes have been made? Basic assumption of regression testing involves the identification of changes, which are made in the modified version of software and its impacts on other parts of the software. There has been carried an extensive research on the web systems as well as desktop systems. Research on regression testing for web system is limited. Web service providers provide the source for which both users and brokers have specifications. This survey paper is aimed to give the meaningful view of work on regression testing for web systems in recent years.

1.1. Significance of the Survey

This survey is intended to give in depth details of the available regression testing techniques for web services. In this survey, main focus is to pinpoint the existing techniques of regression testing with advantages and drawbacks. Testing of web services becomes challenge, and manual as well as automated testing tools help the testers to eliminate the bugs and reduce the web services' development costs.

2. RESEARCH METHODS

2.1. Source of Information

Main source of this survey paper is the journals and proceedings papers, and electronic books searched from Google. Following database were searched out for papers.

- ACM Digital library http://www.acm.org
- IEEE eXplore http://ieeexplore.ieee.org
- Science Direct www.sciencedirect.com
- Google scholars http://scholar.google.com.pk
- Google Books http://books.google.com.pk

The journals and conferences papers taken from above databases mainly cover the workshop on the software engineering. Grey literature (workshop reports, technical reports and work in progress) was not included in this survey paper because of two reasons: firstly the grey literature is difficult to assess in quality and secondly volume of the researches has grown unreasonably. Therefore, sources included in this paper do not overlap in their results.

2.2. Search Criteria

Search criteria used for this survey paper is broader as it covers previous research publications and books using the different terminologies. Regression testing, web services, software, title of articles and their abstracts were researched. Starting year of the papers was set as 2003 and ensured that most relevant researches in the given area of software testing. However, an earlier publication of 1994 was also included for the reference and see how the regression testing was evaluated. Most of the research papers were recently published for last 6 years. Fifteen books and 75 potentially concerned papers were located.

2.3. Study Selection

To achieve the independent assessments, a process of three-stage selection was adopted as given in the following figure 1. In the beginning stage, irrelevant and duplicated papers were recognized and excluded based on the titles. Share of these irrelevant papers was higher in the given number of papers, as papers on software regression testing were not distinguished from the search of journals databases.

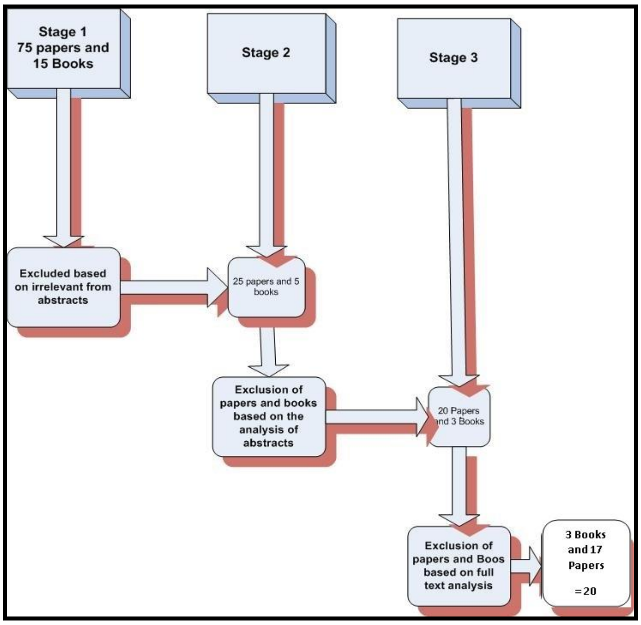
Stage 1
75 papers and
15 Books
Stage 2
Stage 3
Excluded based on irrelevant from abstracts
25 papers and 5 books
Exclusion of papers and books based on the analysis of abstracts
20 Papers nd 3 Books
Exclusion of papers and Boos based on full text analysis
3 Books and 17 Papers
=20

Fig.1 Study selection criteria

After the initial stage, 50 papers and 10 books were excluded. In the next stage, information on abstracts of papers was analyzed in perspective of two dimensions such as web services and regression testing techniques. Most of the research approaches were based on the experiments, surveys, case studies, theory, simulation and reviews. Other than case studies, experiments and survey papers were excluded at this stage. Only papers on the regression testing for web services were included. At the next stage 3, a full analysis of text was performed on the remaining 20 papers and 3 books. In this stage, questions were asked to set final quality criteria for exclusion of papers to extract the information from the selected papers and books. Is study concerned on the regression testing techniques for web services? For example, a paper, which focuses on the regression testing for web application from another aspect of testing? Are results achieved from used metrics comparing the regression testing techniques for web services? For example, a paper is only included when it reports on the prediction of bugs in the modified web services and solution of these bugs. Are rigorous means used to collect and analyze the data? If subset of paper is varying from the analysis of a component, and conclusions are made without any motivation so the paper is excluded [1]. Classification of study was made according to size and type. In this study, case studies, experiments types and surveys are included.

3. GENERAL CONSTRAINTS OF REGRESSION TESTING

Regression testing accounts the large amount of cost, time and effort. In a study of [2] it was reported that for testing of 30000 functional test cases for the software, 1000 machine hours were spent on the execution. Hundreds of hours were spent by test engineers to oversee the process of regression testing, test runs, monitoring of execution, results analysis and keeping the maintenance resources for retesting. However, regression testing technique helps in reduction of testing costs for frequent changes. For the regression testing, two major activities are involved: Firstly, the identification of most affected parts and secondly the test case selection of subsets from the test suite T, which can effectively test the affected or unmodified parts of a web application. Aim behind the selection of subset of test cases is to find the errors induced on changes.

3.1. Challenges of Regression Testing

In a work [3], it was presented that nearly 80 percent of total cost of testing is consumed for the regression testing and 50 percent budget of software maintenance is also used for the regression testing. Many challenges for the regression testing come from the rapid changes in computing environments. Much research and academia work has been done on the regressing testing in making the regression testing more efficient and effective for web systems. First of all regressing testing includes as the attempt for time reduction for regression testing by producing the test suits, which identify the test cases and do not require reusing them for the changed software. Second technique of regression testing reuses the test cases created for software one version and identifying the effective ordering of the test cases running. Third technique used for recycling of test cases is monitored through the execution in order to collect the inputs, which are used for the creation of unit test cases. In the fourth technique of regression testing, test cases can be recovered by the manipulation, identification and transformation of the obsolete test cases.

3.2. Earlier work on Regression Testing

Problem of regression testing as formally defined by [4] was as follows: Let W is the web application program and W' be the modified version of a web application. Initially, a test suite T was developed for the W testing. Regression testing technique is aimed to take the subset of test cases as following.

$$T' \subseteq T \qquad (1)$$

It is intended to execute on the W, so that fault detection with T on W also detects with T on W'. It has been observed that regression testing technique can reduce the regression testing size as compared to other retest-all approach. Retest-all approach involves the using all test suits T in order to re-validate the unmodified program W. Random selection of test cases from test suite T for regression testing cannot give optimistic results because of exposure of several regression errors. However, regression testing approach as compared to random testing or retest-all approaches selects those test cases, used for testing the affected but unmodified parts of the web application.

4. REGRESSION TESTING TECHNIQUES

4.1. Slicing

Research of [5] was based on slicing technique of regression testing for web applications. It was assumed that multiple static HTML pages with other running programs on a server made up a web application. HTML pages may undergo changes as insertion, and deletion of page elements, and deletion and insertion of pages. Complex changes may occur in the combination of these basic modifications. In a web application HTML pages depend on hyperlinks or data and linked together. Further exploring the slicing, take an example of a hyperlink inserted needs to be checked whether it is working or not as part of the same web application. Indirect dependencies arise because of the definition-usage association between variables. These indirect dependencies help to compute the slicing as an extension of the system dependence graph (SDG) model. The system dependence graph represents the each statement as a node and each edge as dependence in a program. Slicing was used as a debugging technique in the earlier researches. Objective of slicing was to extract the code portion, which contained the fault. This slicing technique was subsequently used for regression testing. Precisely, the slicing is able to extract smaller and executable slices used for handling the structures and arrays in a program [6].

4.2. Time Aware Regression Testing

In a work [7] presented that functionality of the web software increased after a programmer fixed the bugs. They called this regression testing a most expensive phase for the system development. Time aware regression testing was a problem for these researchers. They developed a novel genetic algorithm to eliminate the redundant test cases and also time saving for the remaining test cases.

4.3. Regression testing for PHP Web application

Web applications require changes more as compared to desktop systems making updates in features of system, security attacks, and preferred changes for users. Applying the regression testing overall for a system is expensive because companies cannot afford this as expected turnaround time for the patches is short. They provided a solution of this problem when companies face the challenges of security breaches. Regression testing approach in such situation applies only on the code, which has been modified. Affected areas are identified by means of code changes and new test cases generated for PHP programs. They implemented the PHP Analysis and Regression Testing Engine for an open source web system. Executable tests cases generated for the program slices are only used to make correction for the affected code areas in a web application. By resolving the input constraints, through Hampi a testing tool, some of the significant test cases were not solved. Hampi solves the string constraints produced during the program analysis by tools, which are intelligent fuzzers and automated bug finders [8].

4.4. Event Driven Selection of Test Cases

Traditional testing techniques miss the significant features (event driven, hyperlink based structures and multi-tier nature) of the multifaceted web applications. Web applications possess the multiple entry points. Although, regression testing ensures the reliability of the software in quality of web applications, but suffers many limitations in making the frequent updates to web applications. In [9] they worked on the regression testing of web applications based on the event driven selection of test cases. Event represents the system states changes. Web applications become the event driven when a user submits the pages. In a website, many components and dependencies interact with each other and result into a graphic structure known as the event dependency graph (EDG). This event dependency graph depicts the association between page and pages.

Therefore, event driven graph becomes a cyclic without any termination and creates scalability problem. This scalability problem complicates the testing process. However, cyclic redundancies can be removed by converting the event dependency graph into the event test tree. Both, event driven graphs of the modified and original web applications are compared for the identification of potentially affected and most affected nodes, which enable for selection of the test cases for web applications.

4.5. Regression Testing for AJAX Web Applications

In order to cope with dynamism [10] provided that regression testing was the most important for the web technologies such as AJAX. Test case fragility is higher in AJAX applications because of the extensive runtime manipulation of the DOM (document object model) tree. Using the oracle comparator for test case generation with the DOM templates provided the behavior of AJAX user interface as dynamic and non-deterministic. Oracle comparator normally analyses the variance between the servers as the response pages. However, use of Oracle comparator can show the tradeoffs for all web applications. This comparison approach faces the challenges from highly dynamic web applications. For unaltered AJAX applications, the multiple executions with the same input produce the different results.

4.6. Regression testing technique for .net programs

Microsoft .Net and Java virtual machine network has been proposed in the previous times. Environment of a virtual Machine provides the way a programmer can compile the program into an intermediate code as platform independent. Microsoft.net framework provides facility of programming for languages such as C Sharp, Asp.net, C++, Visual basic etc. C Sharp language is new one programming language, used for web as well as desktop applications. Purpose to include the .net framework in this survey paper is that no significant research work has been reported for the regression testing for applications developed in .net framework. In another research [11] worked on the regression testing techniques based on the graph walks. Dependencies emerged through the class hierarchies and exceptions. Their regression testing was based on the following three steps:

i. Construction of the extended Java Interchange Graph (JIG) related with the MSIL code for both modified and original programs.
ii. Instrument and execute the original source program for test coverage information from initial test suite.
iii. Then traversing the extended two JIG models for identification of edges with danger. These danger edges are selected for the execution as regression testing. This technique is more advantageous than earlier developed JIG.

4.6.1. Advantages: Model for regression testing technique provides the improved representation for exception handling and dependencies arising from class hierarchies.

4.6.2. Disadvantages: Large size applications may have millions of lines of code, which is difficult for handling in the regression testing of modified programs. Scalability of the large software is also an issue for the existing techniques. Safe technique for RTS can ensure proper relation dependencies among the target classes of an application.

4.7. Domain Analysis prior to Regression Testing

Domain analysis prior to testing is another way of regression testing for web applications. For regression testing the selection of path for generation of the optimized test sequences has been worked out. Specification, exploration and analysis of the web application is started. A specification document is prepared for all types of the functions including the buttons, links and interaction etc. Designing the representation of the web application is the next step after analysis. Specification document is used to construct the graphical web model. A test sequence is generated by traversing the graphical web model by using the All Link Coverage. Therefore, all links of the web application are tested once at least. Test sequence of a web application becomes as *S0* to *S1* and *S1* to *S2* and so on to *Sm.*

Where *S0* represents the initial page and *Sm* defines the total number of test sequences. From *S0* to *Sm* various intermediate pages are traversed. In order to generate the optimized set of the test sequence, the coverage of all links is essential. By following the same traversal paths, test sequences only differ at their end points.

S1 to *S2* and *S2* to *S3* and *S3* to *S5* and *S5* to *S6* is the first sequence WTS1 and *S1* to *S2* and *S2* to *S3* and *S3* to *S4* is the second sequence WTS2. Another criterion of All Page Coverage is used to reduce the number of test sequences. Length of a test sequence reduces because all pages are tested at least once. In modern era the rapid changes in web services (WS) require the continuous monitoring and retesting of web applications. For identification of changes in the web services, an automated engineering can be used. Deletion, modification and insertion changes may occur in a program, document, software product and web services (WS) in achieving the requirements.

4.8. Cost Reduction Model in Regression Testing

Most focus in this survey has remained attaining the quality of web systems after regression testing. Quality and maintenance of web services are directly linked with a huge amount of costs. A large portion of amount spent on system maintenance goes to regression testing. In work of [12] a model was proposed for cost reduction in regression testing for web services. Firstly identifying the modified portion of web services requires only the testing of that portion of web service. Avoidance of the new test cases and unproductive rerunning of the existing test cases will guarantee that unmodified portion of the web service code gives the same results as attained previously.

4.9. Regression testing based on the (Unified Modeling Language) UML designs

UML designs from test cases to sequence diagrams are best source of information at a higher level for the abstraction regardless of source code. However, completed, updated and consistent designs can give the better analysis and test selection for the regression testing. CASE tools provide engineering capabilities, which require a high caliber work to practice the changes in these designs. Another issue of using UML documents is that these documents cannot fully support to detect the changes in source code. Changes in methods of a class are not visible through these UML documents. Hence code-based and model-based approaches complement each other. Briand et al [13] worked on the automation of test selection of regression testing from UML information linking to test cases. Assumptions on the utilization of UML diagrams were mainly relied on the consistent links among class diagrams, use cases and sequence diagrams. Inconsistency among these diagrams cannot guarantee the validity of analysis based on UML documents. Both, use case and a sequence diagram are closely related with each other. This relationship can be explained by using the Automated Teller Machine application, the use case called as the Do-Transaction also includes another use case called as the Insert-Card. Sequence diagrams are intended in focusing on flow of events from corresponding use cases.

UML 2.0 contains a mechanism of translating the uses cases into the sequence diagrams. This mechanism is known as the interaction uses. Regression Testing Tool based on the test cases and UML design information was developed to classify the test cases for regression testing as reusable, re-testable and obsolete. UML diagrams for the two versions of a system are used as inputs, which result into generation of the XML files by the UML case tools. Diagram types such as use case, class and sequence for two versions are compared and lead towards the classification of test cases. New version of UML diagrams generates the regression test cases.

4.10. Extension in automatic regression testing techniques

Raina and Agarwal [14] extended the earlier work on the automated tool for the regression testing in the web applications. Tool developed in the PHP language consisted of three components as Crawler, Traverser and Comparator. Crawler explores the web pages of the web application in a recursive fashion. Crawler checks the all links in the web application and repeats until all links and web pages of application are not recursively enumerated. Traverser does traverse the web pages and gets the HTML DOM tree of the web pages by using the HTL DOM library. It saves the encounters with the hyperlinks in the database. Third component namely the comparator compares the HTML DOM trees with their web pages in two versions. Number of pages added or removed since the last versions are also enumerated. DOM parser has ability to explore the hypertext markup language (HTML) properties of an element. Comparator compares the HTML DOM trees of web pages for two versions. Any change in the DOM structure is tracked by a specific PHP class.

4.11. Automated Testing Techniques in Regression Testing

Automated testing has been considered as faster and often solution as compared to manual regression testing of web services. Automated regression testing raises the quality, low requirement of test selection and reduces the manual efforts. They used the automated regression testing techniques [15]. Applications of Behavioral Driven Development and Unit testing techniques for the high level systems such as the systems tests have been uncertain. Therefore, need for development of automated test techniques, which are explicit for the acceptance test and systems e.g. Record and Replay is technique supported by a tool. In such technique, users' interaction with a system is captured in the form of script, which is replayed automatically later on. However, the scripts captured are fragile component changes of GUI. Changes in GUI or API may inept suits of the entire automated tests. Hence, automated testing technique cannot meet the required standards of regression testing and need the development of more robust techniques for the regression testing of web services. Manual tests were automated as more cost efficient, which reduced the manual effort and increased the execution speed.

On some extents, the automated testing is advantageous than manual testing and provides best use for the acceptance testing. Some of the automated testing tools can support the test suits with several test scripts such as *Sikuli*. The *Sikuli* is the open source testing tool as compared to other paid testing tools, which cost thousands of Euros. Automated regression testing tool enables the tester to focus on testing the functionalities of a web service. Automated regression testing tools increases the reliability of the regression testing and checking of the unusual or complex environments. Measurements and metrics acquisition about when data is easily available can be evaluated. In spite of many advantages of using the automated regression testing tools, these tools are missing many serious features related to quality and scope of automated tests. Statically used, automated regression tests execute the same actions each time. This means test traversed the same path when executed in the automated testing software. Additional code coverage is also not yielded on using the repeatedly tests. Secondly, regression testing through automated tools is simpler because simple and short sequence of commands is used for these tests. Mostly, positive tests take limited range of the inputs or their combinations. Critical defects are sometimes unresolved because of limited negative inputs. A web application undergoes various changes in the passage of time. Therefore, corresponding tests also require the changes to be maintained. Therefore, automation requires the proper structuring, as a minor change in the Graphical User Interface (GUI) can impact the automated tests.

Hence, special attention must be given in the building of automation architecture that supports the tests maintainability and results into reduction of the vulnerability [19].

4.12. Quick Test Professional for regression testing of web applications

Like many other automated regression tools such as IBM Rational Functional Tester from IBM, Selenium as open source, Test Complete from Automated QA, Rational Robot from IBM, HPs product for web services testing the Quick Test Professional is used for testing of web applications. It identifies the object of a web application UI or a page of web application to be tested and desired operations such as keyboard events or mouse clicks are performed. A scripting language such VBScript is used for performing these actions. The VBScript is manipulated in order to perform the desired actions. Many drawbacks of manual testing are addressed by using the automated testing tool QTP as it runs faster than manual running of test cases and hidden information can be programmed and sophisticates the tests.

5. CONCURRENCY PROBLEM IN THE AUTOMATION OF REGRESSION TESTING SELECTION FOR THE WEB APPLICATIONS

Regression testing is the main practice for software maintenance. Regression test selection is aimed to reduce the test case numbers, which need retesting of these test cases. Safe regression testing selection (RTS) assures that test cases revealed with the modification are not left unselected. In spit the development of many safe RTS, but no one of them was directly applicable to web services. An automated designed approach can give a solution. However, automation of an approached for regression testing give rise to many challenging issues, which arise from the concurrent modifications in the autonomous, distributed but interconnected services. Need of solution of these challenges becomes keener as composite web services are becoming more common. Automation of regression testing selection (RTS) will require the same environment for each side. Test consistency is also critical for the success of regression testing. Fault happening will help to locate the modification and therefore this property is known as the fault locatability. When two operations are modified concurrently, then locations of the modification determine the assurance of locatability. For example, operation B calls the operation C, and operation B in turn calls for operation D. When changes in operations C and D occur at the same time, then no need to single out the cause of fault between changes at two operations, as faults occurs due to testing of B. A modification *md* for an operation in the service may impact the service operation, which directly or indirectly call *md*. A call graph represents the nodes as client pro-grams or performs operations. Race conditions also arise from the multi-tasking of test cases.

Multi-tasking generates the issue of concurrency. Therefore, concurrent execution of test cases requires additional effort to ensure the successful completion of each test case. Concurrency problem has been yet a hot and open research topic. Non determination of the concurrency issues remained difficult to solve them. Regression testing for the concurrently accessible web application is also difficult [16]. Focusing on the autonomous natures of the web applications, the feasibility of a central call graph cannot meet the requirements. No site of a graph has a control of a call. However, this can be achieved in different scenarios. A framework can be established that supports the publishing or subscription mechanism to intimate the participants about changes in a web service. If a subscriber asks for any operation in the web service, it calls for operation of web services. On reaching every call for a service, a RTS agent maintains the *Notify List* for the every operation.

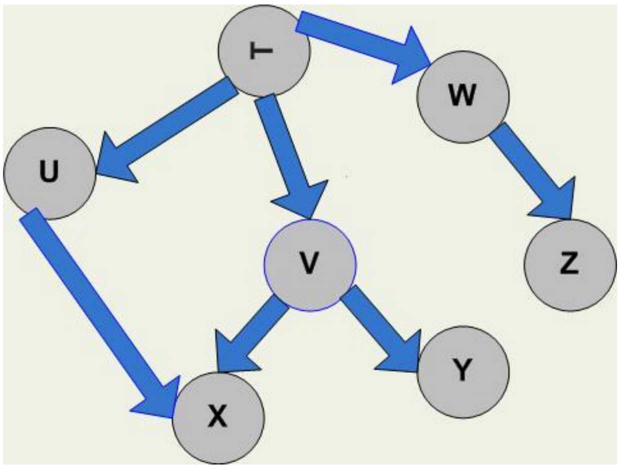

Fig 2. Control Flow Graph for Software Entity.

Association between events is recognized through a logical clock. RTS agent maintains this logical clock and its operations take place usually in three ways as get_LC (), set_LC (value of a timestamp), and increment_LC (). A timestamp value is carried on when any change occurs or when deployed any change. Each of subscribers is informed about modification in the code, and message is passed to all subscribers with three parameters: id of the changed operation, updated control flow graph (CFG) of web service, timestamp of the modification.

$$\text{Mesg} = (\text{Id of changed operation, ts, CF G}) \qquad (2)$$

Considering the software entity with the control flow graph as shown in the figure 2 four test cases exist: Case 1 covers up the T-U-X path, Case 2 covers up the T-V-X path, Case 3 covers up T-V-Y path and Case 4 covers up T-W-Z path. If the code relevant to X node was modified, the determination of the impact analysis would result into a dangerous list T-U-X and T-V-X, corresponding to test cases 1 and 2.

6. UNIQUENESS OF THE REGRESSION TESTING TECHNIQUES

Variance concerning the uniqueness of the regression testing techniques for web applications in the selected articles is a great. Some of the earlier regression testing techniques can be regarded as a novel on their first appearance in the literature, and others are only the extension or variants of the existing techniques. For example, earlier work of [4] was a great hallmark in the field of regressing testing for the structural and non-structural programs. This technique also had flexibility in automating the approach and applying to all those programs for which dependence graphs were constructed. A PDG provides the data dependence and control dependence in a single graph. Extension in this technique was made in a work of [17] in order to find out the bugs in the modified program. This type of regression testing was intended to be safe. But the safe testing generates the issue of rerunning those test cases, which cannot expose faults. Some of the techniques reviewed in papers are specific for web application developed in different languages such as PHP [8], and AJAX [10]. It is clear that each type of technique is considered for a web application developed in specific languages PHP and AJAX respectively. There is a need to develop the software regression testing technique, which is applicable for web application developed in different languages. However, this gap has been filled by developing the automated regression testing tools such as HP QTP, and Selenium etc.

Selenium is a particular framework for the regression testing of the web applications. This framework supports the scripting as well as programming languages like PHP, Ruby, Jave and CSharp etc. Classification of regression testing technique depends upon the input type for these techniques. Source code text is the main input type for these techniques and other input type can be the intermediate code in case of the virtual machines. Other input type may be code used for the analysis such as C++, Java and PHP etc. Some of the techniques are based on the object oriented paradigm while other are specified for procedural languages. Most important feature of regression testing technique is whether it is safe or unsafe. For the safe regression testing, all bugs found by the full test suite remain same for the regression testing technique. This property provides a solid base to classify the regression testing techniques in safe techniques or unsafe techniques. For safe technique, Re-test all is the simple example as it is used to select all tests cases, which guarantee the identification of all defects. On other hands, unsafe technique, which is considered to be less effective as it leaves many defects unidentified.

7. EMPIRICAL EVALUATION OF REGRESSION TESTING TECHNIQUES

Each technique and relevant discussion with the development and implementation is also given. In the history of regression testing [4] characterized the regression testing issues and proposed the solution of problems. After the identification of regression testing issues in the context of technique selection, prioritization of testing techniques, most of the researchers focused on these issues. The techniques included in this paper can be further classified on their similarities. One group of regression testing techniques for web services is based on the identification of the regression testing [4]-[2].They focused on the constraints of regression testing techniques and developed the solution of these constraints in their works. In another group of techniques [6]-[5] performed the regression testing of web services using the slicing technique. Slicing technique is mainly focused on the extraction of the affected code and removes the bugs in it after modification in the program. Third group of regression testing techniques is based on the metadata of software and does not consider the source code. Briand et al [13] used the UML designs for regression testing of web services, and [18] also performed the domain analysis before the regression testing. Another group of regression testing technique is based on the selection of techniques, which reduce the cost and time of regression testing. Testing time evaluation also reduces time of test suite selection and results in to a precision. Significant component in the cost reduction for the regression testing is the time analysis required for the selection of the suitable and priority based test cases to be executed. Another group of regression testing technique is specified for the web development languages PHP, .net and AJAX. In [20] used the Oracle comparator for the AJAX applications, and [8] focused on the regression testing for PHP web applications.

Use of Genetic Algorithm [7] and Safe regression testing [17] also constitute a group of regression testing for web services. Rests of the techniques were mainly dependent upon the automated testing of web services. Contribution in regression testing of web application was based on regression testing [19]-[15]-[17]. Group of studies concerning the reduction test cases number is compared with retest all [2] and conclusion can be drawn that achievement of reduction of a number of test cases but nothing affect on size of the program in practice. This kind of problem limits the strengths of the relevant regression testing technique. Many of the studies consider the time reduction in small programs, and difference in time is measured in million parts of seconds. However, trend of time reduction in regression testing is positive for small or medium sized programs.

8. CONCLUSION

In this survey paper, regression testing techniques for web services has been evaluated. Mostly sources of this survey paper are taken from databases of IEEE, and ACM and few papers from Google Scholars. Most recent Research publications have been included. In this survey paper, basic issues of regression testing are identified. Firstly, a great variance in the identified regression techniques for web services is found. Some of the techniques are presented as original solution of problems and others are extension in the existing techniques. Some of the techniques are given in the general manners for handling of regression testing in the AJAX, PHP and .net framework applications.

In this study, several groups of combined studies with similarities are constituted to focus on which aspects of regression testing have been considered for these groups of techniques. In future, research community needs to focus on the concepts, which are general in regression testing instead of variants of the specific regression testing techniques. Most complex environments must be scaled up by encouraging the researchers to replicate the studies in diverse contexts.

REFERENCES

[1] Engstrom Emeile Runeson Per and Skoglund Mats. A systematic review on regression test selection techniques. ScienceDirect, pages 14–30, 2010.

[2]. Ammann Paul and Offutt Jeff. Introduction to software testing. Cambridge University Press, 2008.

[3] Harrold Mary. Reduce, reuse, recycle, recover: Techniques for improved regression testing. ICMS, 2009.

[4] Rothermel Gregg and Harrold J. Mary. Selecting tests and identifying test coverage requirements for modified software. ACM, pages 169–184, 1994.

[5] Xu Lei Xu Baowen Chen Zhenqiang Jiang Jixiang and Chen Huowang. Regression testing for web applications based on slicing. pages 652–656, 2003.

[6] Mathur Aditya. Foundations of Software Testing. Pearson Education Inc, 2008.

[7] You Liang and Lu Yansheng. A genetic algorithm for the time aware regression testing reduction problem. IEEE Computer Society, pages 596–599, 2012.

[8] Marback Aaron, Do Hyunsook and Ehresmann Nathan. An effective regression testing approach for php web applications. 2012.

[9] Gole Ruchi and Kumar Akshi, Event driven test case selection for regression testing web applications. ICAESM -2012, 2012.

[10]. Dong Xinshu, Patil Kailas, Mao Jian and Liang Zhenkai, A Comprehensive Client-side Behavior

Model for Diagnosing Attacks in Ajax Applications, National University of Singapore. 2013.

[11] Bajpai Manisha and Agrawal P. Arun. A survey on regression test selection techniques for dot net and bpel programs. International Journal of Computer Science and Management Research, 1, 2012.

[12] Chaturvedi Animesh and Gupta Atul. A tool supported approach to perform efficient regression testing of web services. ScienceDirect, pages 50–55, 2013.

[13] Briand L.C. Labiche Y and He S. Automating regression test selection based on uml design. Science Direct, pages 16–30, 2013.

[14] Raina Shikha and Agarwal Arun. An automated tool for regression testing in web applications. ACM, 38(4), 2008.

[15] Borjesson Emil and Feldt Robert. Automated system testing using visual gui testing tools: A comparative study in industry. IEEE Computer Society, 2012.

[16]. Ruth Michael and Tu Shengru. Concurrency issues in automating rts for web services. ICWS -2007, 2007.

[17]. Xing Jainchun Wang Hongdaa Sony Wei and Yang Qiliang. Safe regression test selection based on program dependence graphs. 2012.

[18] Gagandeep and Sengupta Jyotsna. Automatic generation of regression test cases for web components using domain analysis and modeling. International Journal of Computer Applications, 11:0975 8887, 12. 2010.

[19] Graham Dorothy and Fewster Mark. Experience of test automation: Case studies of software test automation. Pearson Education Inc, 2012.

[20] Roest Danny, Mesbah Ali and Deursen Arie. Regression testing ajax applications: Coping with dynamism. IEEE Computer Society, 2010.

AUTHOR PROFILES

Muhammad Hassnain is a student of MS (Computer Science) at Abasyn University. He received his MCS degree from Pir Mehr Ali Shah Arid Agriculture University Rawalpindi in 2007. His research area is software requirement gathering and specifications. He has also interest in implementation of routing protocols.

Anjum Abbas is a student of MS (Computer Science) at Abasyn University. He received his MCS degree from Allama Iqbal Open University Islamabad in 2004. His research area is Manual and Automated Testing of software. He works as QA analyst at MAKABU, Software Technology Park 1, 4th floor Islamabad.