



# **iJRASET**

International Journal For Research in  
Applied Science and Engineering Technology



---

# **INTERNATIONAL JOURNAL FOR RESEARCH**

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

---

**Volume: 6      Issue: IV      Month of publication: April 2018**

**DOI: <http://doi.org/10.22214/ijraset.2018.4609>**

**[www.ijraset.com](http://www.ijraset.com)**

**Call: ☎ 08813907089**

**E-mail ID: [ijraset@gmail.com](mailto:ijraset@gmail.com)**

# Automated Web Based Test Engine Using Cloud

Manasi Manwadkar<sup>1</sup>, Shravni Ghodke<sup>2</sup>, Kirti Deshpande<sup>3</sup>, Dhanashree Kondhare<sup>4</sup>, Sonali Kadam<sup>5</sup>

<sup>1, 2, 3, 4, 5</sup>Dept. of Computer Engineering, Bharati Vidyapeeth College of Engineering For Women

**Abstract:** *Software Testing is very important in industries as it provides the information about the quality of the product to be delivered. It helps to improve the quality of the software for future use. Automated Testing is a technique that provides efficiency and improvement to the testing of software. This paper describes the automation of the two main testing processes namely functional and business logic testing. It presents the technical solutions that overcome the difficulties in application testing. In this paper authors have proposed a web based automatic software testing framework for applications which performs the functional testing of software based on the QTP(Quick Test Professionals) and also performs the business logic testing by applying various test cases.*

**Keywords:** *Cloud Platform, Business Logic Testing, Functional Testing, Test Cases, Automated Testing.*

## I. INTRODUCTION

Software testing is very crucial task to guarantee the quality of the product. For testing whole software many tests are performed like functional testing, performance testing, business logic testing, security testing and many more. Most important testing among above are functional testing and business logic testing. Automating these tests and making these tests available to user using web, will help the tester to easily test the software.

### A. Functional Testing

Functional Testing is used to check software codes and what actually the software does. It basically checks for quality of the software. Functional Testing is technique is used to test the functions of the software. It checks each and every function in the software application. It verifies the input, output and their interfaces, databases etc. It generates the report for this testing which includes all the analysis of the testing. This is mainly done by using various test cases. This mainly checks main functions, basic usability, error conditions etc.

The process of Functional Testing is as follows:

- 1) Take the input and generate output.
- 2) Apply various test cases.
- 3) Compare the generated and expected output.
- 4) Generate the report of the testing.

### B. Business Logic Testing

Business Logic Testing is an important part of the software testing. Business Logic Testing is done after Functional Testing. Here aim is to check business logic of the software i.e. to check the structure of all the page is in proper and accurate format or not. Business Logic is usually checked manually by crosschecking the mentioned elements of page with displayed elements and finding errors if any element is not present in page. So for this, tester should initially understand the whole structure and working of software product.

### C. Cloud Platform

A cloud is mainly designed for the purpose of measured IT resources and remotely accessing data. The data is virtually stored in the cloud platform. We are able to access this data anytime and anywhere. The main purpose of cloud computing is to allow users to take advantages of all technologies without any deep knowledge and expertise in that field.

AWS (Amazon Web Services) is a platform which provides services of cloud computing to the various companies, industries etc. The AWS technology is implemented all over the world and maintained in the database.

### D. Test Cases

Test cases is a set of various conditions. The tester will check whether the system satisfies all the conditions that are required or not. It helps in checking of error in the application.<sup>[1]</sup> Test cases develop an expected result for the given application. It is just like a question and answer like policy. Tester asks questions in the test cases format and answer will be in the result. By using test cases quality of the software will increase. There are two types of test cases as: Formal and Informal. In the formal test cases one positive test cases

are there. These are used when the application is in proper format. The informal test cases are used when the application having without formal requirement. The parameters of test cases are as: Test Id, Test Scenario, Test Case Description, Test Steps Prerequisite, Test data, Result, Report.

#### *E. Test Cases*

QTP (Quick Test Professionals) is an automated testing tool. QTP helps to identify any error or gap in the result of the application. It compares the actual and expected result. Test cases are run in the QTP. Technical and non-technical both testers can use QTP. It supports both playback and record. It has object identification mechanism.

## **II. TEST AUTOMATION**

In software testing, test automation is the use of special software to control the execution of tests and the comparison of actual outcomes with predicted outcomes. Test automation can automate some repetitive but necessary tasks in a formalized testing process already in place, or perform additional testing that would be difficult to do manually.<sup>[2]</sup> Automation process includes creation of detailed test cases, including predictable "expected results", which have been derived from Business Functional Specifications and other Design documentation. There are many different automated testing tools like Selenium, TestComplete, Ranorex, Sahi which automate desktop, web, and mobile application testing including UI and performance testing. These tools are generally automating the types of testing like Functional testing, Regression testing, Stress testing, Performance testing and Load testing.

## **III. PROPOSED SYSTEM**

Our proposed system is a web based test engine so it will be accessed by URL. The home page will be displayed. The home page will consist of various fields such as Home, Login, About, Procedure, Benefits and FAQs. When user wants to go any particular field then user will reach till it by clicking on that field in Menu. Firstly, main page contains the introduction field which gives introduction about the software.

Then after that, steps are given which are to be followed during testing of application. Then user can immediately start testing by clicking on "Start Now" button given below steps.

Steps are then followed by Benefits of apps. Then there are FAQs which will clear the doubts (if any) of user regarding the use. And at last there is helpline consisting of contact for help of user. Upload is main function in this application. For application which is to be tested it should be first uploaded. So, for upload user need to click on "Upload" button. Then a dialog box will appear to select the file which is to be uploaded. After that selected file will be uploaded. There is also display of various graphs, pie-diagrams, which give history about user's previous test history

There are main two modules in the system

#### *A. Functional Testing*

Functional testing is primarily used to verify that a piece of software is providing the same output as required by the end-user or business.<sup>[3]</sup> This testing mainly involves black box testing and it is not concerned about the source code of the application.

Each and every functionality of the system is tested by providing appropriate input, verifying the output and comparing the actual results with the expected results. This testing involves checking of User Interface, APIs, Database, security, client/ server applications and functionality of the Application Under Test.<sup>[4]</sup> While performing functional testing, system initially identify the functions that software is expected to perform. It analyzes all the build and identifies the function of application. Create input data based on specification of functions of application. And accordingly determines the expected output. Test cases are then executed for running the application and perform the testing. After the execution of application, the actual output is compared with the expected output. Accordingly, the result will be generated regarding the functional testing of application. For testing

#### *B. Business Logic Testing*

In computer software, business logic or domain logic is the part of the program that encodes the real-world business rules that determine how data can be created, stored, and changed. Business logic prescribes how business objects interact with one another and enforces the routes and the methods by which business objects are accessed and updated.<sup>[5]</sup>

After functional testing, the application is tested for business logic testing, as this test is done page by page of application, user need to browse and select the page. By clicking on browse button, a dialog box appears which ask user to select a particular page from the build which he wants to test. After selecting page, system will ask user to define the constraints of the selected page.

There will be GUI for user to give input. There will be field for user to specify the number for constraints such as textboxes, radio button, labels, checkboxes, combo box, update button, submit button, apply button etc. After specifying the number of constraints, user need to click on “Save” button for saving the contents of the page. Then user will click on “Start Testing” button for the system to start testing.

For testing the business logic of selected page various test case are applied on that selected page; accordingly, system will check the constraints of user developed pages with the constraints of page of uploaded application. System will check if the application’s page is having the constraints same as that of recently developed page. For example, if user specify that its application’s page contains 3 textboxes and 2 buttons then system will check from application’s page if that page actually contains 3 textboxes and 2 buttons or The system will generate report of 100% success if the test contents match with application contents, otherwise system will update the user about the fault in system by specifying the numbers and elements of constraints for failure.

specifying the number of constraints and testing first page, user need to select another page of his application and follow the above procedure for the selected page. Like that user can select any number of pages according to his application.

After successful testing and generating result of various pages of application, the System analyses these results according to some predefined parameters. The parameters can be as, if system finds three or four errors while testing, may indicate the errors of 20% then system is only 80% neatly workable, so it generates the report that the system should be repaired and specifies the area of repair.

#### IV.SYSTEM ARCHITECTURE

A. System Architecture is Divided Into 3 Sections

- 1) Front End
- 2) Middleware
- 3) Back End

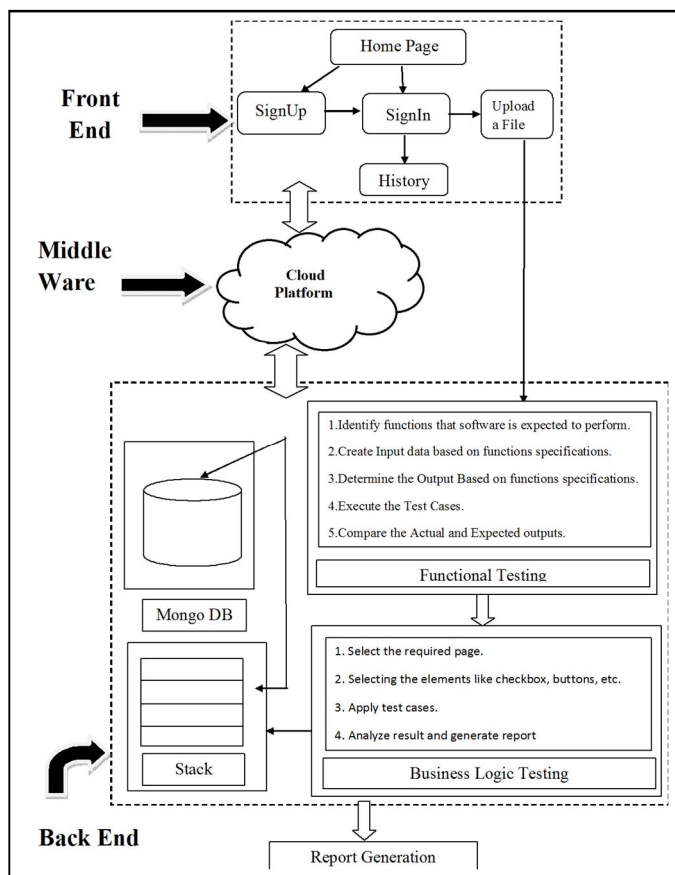


Fig. 1 System Architecture



The Front end of the system includes main Homepage of our Test Engine. Front end will be designed by using HTML5 and CSS. On Homepage we have login and registration form pages. User needs to create new username and password for his/her use. Then user needs to fill the registration page which takes personal details of user (for example: First name, last name, email, contact no). Every time user needs to use this username and password while signing up for testing. It also includes the history of the user, whether if he has tested any application before, etc. It includes upload a file option which allows user to upload file which needs to be tested. The middleware of the system is Cloud platform such as Amazon Web Service(AWS). AWS is a secure cloud service platform, offering compute power, database storage, content delivery and other functionality to help business scale and grow and also to build applications with increased scalability, flexibility and reliability. AWS supports different features like Cloud Drive, Dynamo DB, Elastic Cache, Simple storage service(S3)

The backend of the system will include Non-Relational Database. It includes Database stored in AWS load balancer and stack. Generally, it includes Mongo DB for database Management. Mongo DB Is free and open source platform document oriented database program. The back-end tasks will include functional and business logic testing. User will upload the file which it to be tested and then system that is test engine will perform functional logic testing first

Functional logic testing is nothing but testing the proper syntax of the application by following steps:

- 4) Identify functions that software is expected to perform.
- 5) Create Input data based on functions specifications.
- 6) Determine the Output Based on functions specifications.
- 7) Execute the Test Cases.
- 8) Compare the Actual and expected result.

Then after performing the functional testing it performs Business logic testing. Business logic testing is nothing but testing whether the required business is proper or not as defined in test cases .It performs in following steps:

- 9) Select required page.
- 10) Checking for tools like checkbox, buttons.
- 11) Apply test Cases.
- 12) Analysis

After performing testing, it will do analysis and generate report on the application that is tested

## V. ADVANTAGES

- A. System is Automated. Generally, system will be automated reducing the manual efforts. Mainly till now testing of business logic was done manual but now this process is automated in proposed system.
- B. It is easily Portable. As it's a web based application there is no need to install it in specific system.
- C. It is Reliable. Testing is more reliable and way quicker when running repetitive standardized tests which can not be skipped, ever, but may cause errors when manually tested.
- D. It is Secure. Every user using this engine have to sign up when using this for the first time. After that there is just need of login for security purpose.
- E. It is Available on any browser. No matter where you are in the world. You can start the tests when you leave the office and when you get back in the morning you can see the results and keep on working.
- F. It is a Web Based Test Engine. System will be web based as it is deployed on Cloud Platform.

## VI. DISADVANTAGES

The only disadvantage of proposed system is, when uploaded application consists of thousands of pages then it becomes very difficult as the proposed system doesn't make use of any artificial intelligence for identification of page, user need to select each and every page separately for testing the business logic of application.

## VII. APPLICATION

This web-based test engine will be useful for Software Developers, Testers and Debuggers. A software developer is a person concerned with facets of the software development process, including the research, design, programming and testing of computer software. Debuggers also offer more sophisticated functions such as running a program step by step, stopping (breaking) (pausing the program to examine the current state) at some event or specified instruction by means of a breakpoint, and tracking the values of variables.

### VIII. CONCLUSIONS

This paper gives the brief idea about implementing Automated Web Based Test Engine which tests the application and generates the report for the desired application. We have designed an automatic software testing framework for applications which performs the functional testing of software based on the QTP and also performs the business logic testing by applying various test cases.

Till now business logic testing was performed only manually but this paper gives the brief description how this business logic testing is automated.

It will be very beneficial for tester and debuggers to make use of this test engine to automatically test the business logic. Testing of any software will be much faster now

### REFERENCES

- [1] The Software Testing Fundamentals website. [Online]. Available: <http://softwaretestingfundamentals.com/test-case/>
- [2] The Wikipedia website. [Online]. Available: [http://en.wikipedia.org/wiki/test\\_automation](http://en.wikipedia.org/wiki/test_automation)
- [3] The Techopedia website. [Online]. Available: <https://www.techopedia.com/definition/19509/functional-testing>
- [4] The Guru99 website. [Online]. Available: <https://www.guru99.com/functional-testing.html>
- [5] The Wikipedia website. [Online]. Available: [https://en.wikipedia.org/wiki/Business\\_logic](https://en.wikipedia.org/wiki/Business_logic)
- [6] Wenhao Fu, Huiqun Yu, Guisheng Fan and Xiang Ji, "Test case prioritization approach to improving the effectiveness of fault localization," in Software Analysis, Testing and Evolution (SATE), International Conference : IEEE, November. 2016
- [7] B Roget Favilla, "Automatic Test generation and fault diagnosis," in Automation of Software Test, 2009. AST '09. ICSE Workshop : IEEE, August. 2002
- [8] Indg P.S.C.silver, Everton L.G.Alives and Wilkorson L.Andrabe, " Analysing automatic test generation tools for refactoring validation," in Automation of Software Testing (AST), 2017 IEEE/ACM 12th International Workshop : IEEE, May. 2017.
- [9] XinBian wang and Guangjun HC, "The research of data driven testing based on QTP," in Computer Science & Education (ICCSE), 2014 9th International Conference : IEEE, August.2014.
- [10] Fei Wang and Wencai Du, "A test automation framework based on WEB(on Jmeter and selenium)," in Computer and Information Science (ICIS), 2012 IEEE/ACIS 11th International Conference : IEEE, June. 2012
- [11] Matteo Meucci and Andrew Muller, "OWASP Testing Guide." [Online]. Available:
- [12] ([https://www.owasp.org/index.php/Testing\\_for\\_business\\_logic#Business\\_Logic\\_Test\\_Cases](https://www.owasp.org/index.php/Testing_for_business_logic#Business_Logic_Test_Cases))
- [13] Ron Patton – Software Testing: 2nd ed., Tata McGraw, 2009.
- [14] (2002) The IEEE website. [Online]. Available: <http://www.ieee.org/>



10.22214/IJRASET



45.98



IMPACT FACTOR:  
7.129



IMPACT FACTOR:  
7.429



# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24\*7 Support on Whatsapp)