

REGRESSION TESTING FOR CONTOSO

Nakush Sharma and Shahid Ali

Department of Information Technology, AGI Institute, Auckland, New Zealand

ABSTRACT

This project work involves the automation regression testing of Microsoft Dynamics 365(MD365) enterprise resource planning (ERP) which is cloud based solution for Contoso. Contoso has problem to access and monitor the huge database of customers, vendors and products on traditional ERP system. Therefore, they have started to use the cloud based ERP system. The reason of this report is that there are constant updates in MD365. They also wanted to add new fields in customer, vendor and product web forms. Therefore, automation regression testing is essential for this project. Selenium web driver is selected for this type of testing. This project will help the Contoso to achieve the regression automation testing by Selenium web driver for MD365. The testing will help them to execute repeat test cases without any duplicity of the code. The report generation of the execution result will help them to understand the progress of the project. Moreover, Scrum methodology is adopted by the organization which is very flexible and helps them to provide the customer satisfaction. It is easy to adapt the requirement changes in scrum methodology.

KEYWORDS

Automation Testing, Regression testing, Microsoft Dynamics (MD 365), Selenium, Scrum

1. INTRODUCTION

Contoso is an Australia based company which have several branches in New Zealand. The company is dealing in distribution and wholesale business. They provide the merchandising, operational support for the liquor, grocery and hardware. Contoso is growing rapidly in the retail market. They provide world class services to their customers and vendors. As a result, Contoso now having huge database of the customers, vendors and products.

Contoso has decided to select the Microsoft Dynamics 365(MD 365) ERP solution so that they can manage and monitor the database on cloud. MD365 is a Microsoft official cloud base platform which includes the customer relationship and enterprise resource planning. Microsoft wants to provide great experience to their customer because of that there are continuous updates in the MD365. These updates include the new version of MD365 according to their users' feedback. These updates happen four times in a year. Moreover, Contoso also wanted to add new fields in the create customer, vendor and product form according to their business needs. Therefore, Regression testing becomes the important part of this project. This project includes the automation regression testing. This type of testing automates the repetitive test cases to check that old functionality is working perfectly after adding the new code.

The user acceptance testing is also part of this project because there are some certain requirements which need to meet according the organization expectations. Both testing plays a significant role in this project. Therefore, Selenium web driver is selected for this project. It is an open source tool for web applications. Selenium web driver supports different types of programming languages and multiple platforms as well. This tool is very effective and efficient for this project because it will

automate the tasks with the help of object oriented programming language methods. The tool is integrated with the Eclipse IDE.

The scope of the report includes automating the functional regression testing and user acceptance testing of creating new customer, vendor and product. It includes testing the new fields and generating the report of the pass test cases. The project also contains the testing of the mandatory fields of creating new customer, vendor and product which is a requirement of the organization. This report will also discuss about the recommendations for the project. The recommendations and discussions have discussed are discussed according to research. However, Performance and load testing is out of scope of this project.

The main objective of this project is to check the older functionality is working properly after new updates and adding the new functionality in MD 365. This project also discusses the scrum methodology and its benefits for the organization.

This research project is organized as follow: Section 2 focuses on the literature review of various studies focusing on regression testing. Section 3 is this research is focused on the project execution. Results of this research project are provided in section 4. Discussion to the results are provided in section 5. Section 6 is dedicated towards the future work recommendations. Finally, in section 7 conclusion to the research is provided.

2. LITERATURE REVIEW

Over the years, there are many research papers written on regression testing and automation testing. Akira R. Onam has conducted a research paper on regression testing [1]. In this research, researchers explain the importance of the regression testing in the big projects which required the repetitive test cases. This paper explains the regression testing in detail with the help of different research papers. The paper also explains the processes and methodology for regression testing. The strength of this paper is another aspect of regression testing in Information Technology industry. Researcher explains regression testing is not only for the development and maintenance. It also beneficial for the big projects that needed repetitive test cases. Hence, this paper justifies the reason to select regression testing for this project because this project also demands the regression testing due to constant updates and modification.

Another research was conducted by Monika Sharma [2]. This research explains about the importance of the automation testing. It also discusses about the (Sharma, 2014) different tools for the automation testing and importance of the automation testing for web application. The research provides several tools to automate the regression and integration testing.

Automation testing life was explained [3]. Researchers explained about the steps to achieve the automation testing with the help of automation testing life cycle. This life cycle introduces the different processes to introduce the automation testing. The paper has chosen because it helps to

understand the different stages to achieve the regression automation testing such as requirement analysis, test design, environment setup and execution. We used this research for the reference of automation testing phase to our project.

To maintain and repair the test suite for automation testing a study was conducted [4]. In this research, researchers explained the selenium web driver, Xpath and page object model. The result of their investigation is ID locator which is Xpath is the best method to save the efforts and time for the testing.

Another study was conducted to explain the TestNG frameworks, annotations and its benefits [7]. This study provides the basic and advance feature of the TestNg. This research has divided into twelve sections in which researchers have explained about the advantages of xml file, annotations, dependencies, unit, functional and regression testing in TestNg. This research was very helpful for our project because it covers most of the essential sections to our project. It also supports to understand the concept of the TestNg. This research [7] was relevant with the project because it clears the doubt on how to generate the reporting using TestNg.

Apart from that, the scrum methodology is the essential part for this project. There are many research papers has written on the importance of scrum methodology. Traditional SDLC Vs Scrum Methodology research paper is selected to understand the importance of the scrum methodology. This research was conducted by Sundararajan, D. M. [5]. The researcher explains about the frequent change requirements in the project is very common nowadays. Traditional SDLC is not capable to meet the frequent [5] requirement changes. Hence, Scrum methodology becomes very popular. This research presents the views about the benefits of adopting Scrum methodology because it accepts the changes in requirement during the process of project development. In the result, they explain that the software development life cycle methodology varies on the project requirement. Hence, Scrum methodology is suitable for this project. This project has constant updates which required a methodology to accept constant changes.

Another important research was conducted to highlights and compare Waterfall, v-model and agile methodologies [6]. This research helped us to select the right methodology for our project. According to this research, agile methodology was more flexible according to the traditional methodologies such as waterfall and v-model.

3. PROJECT EXECUTION

Project execution detail of the project is briefly elaborated below.

3.1. Automation test Phase

Automation testing phase is like software testing life cycle. In the automation testing phases is a process which has certain [3] steps to execute the test steps. Each phase has different aim and objectives. Figure 1 depicts the different stages and their goals in automation testing phase.

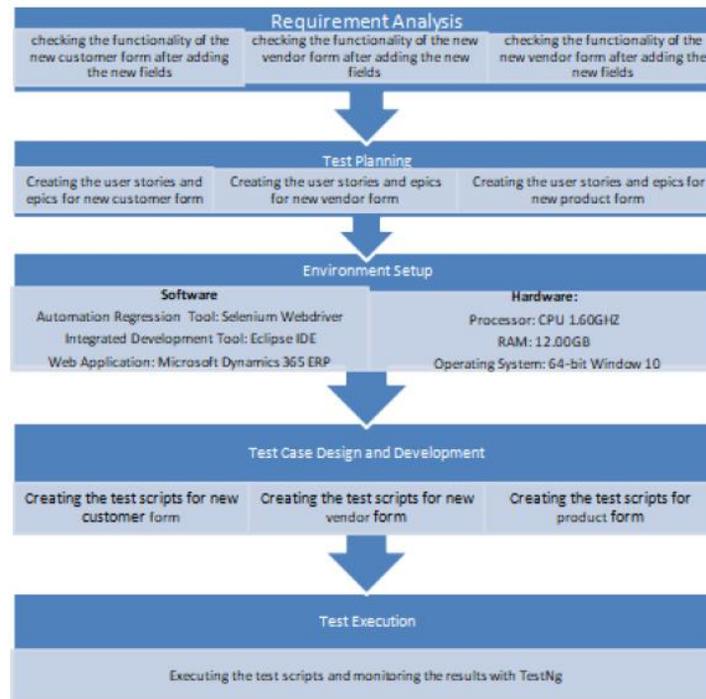


Figure 1: Stages and their goals in Automation Testing phase

3.2. Architecture of Automation test Plan

The architecture of automation test plan has started with the page object model. Page object model is a design pattern for creating the object repository for web UI elements. In this model, there is a different class for each webpage. These page classes find the web elements and contain the page method to perform the operations on the web elements. The names have given to the class according to their task they are performing. This page object method is performed in the Eclipse IDE by integrating selenium web driver libraries. Eclipse IDE is a java development environment tool which is used in this project to automate the test cases using selenium web driver libraries. It is a platform for automate the test cases. The architecture also contains the Google chrome browser because the execution of the test suite is done on this browser. Moreover, TestNg is used in this project as a framework to generate the reports of the results. TestNg is an add-on for the reporting as well. In the last, the application which needs to be test is Microsoft Dynamics 365 ERP solution. Figure 2 shows the work flow of architecture of automation test plan.

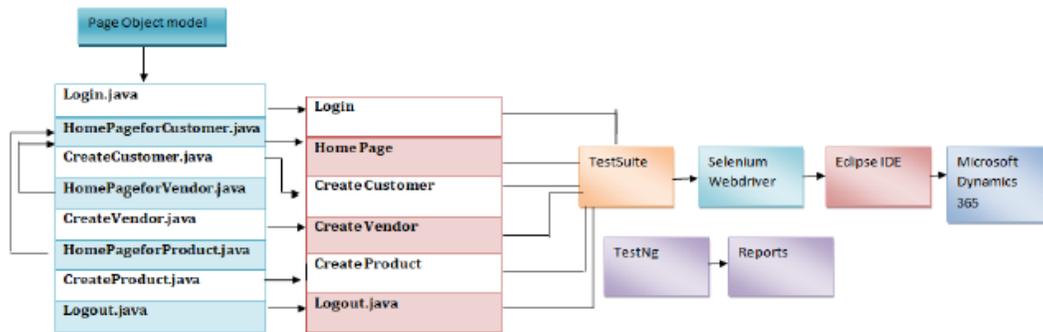


Fig 2: Workflow of Architecture of Automation Test Plan

3.3. Test Scripts

Figure 3 presents the structure of the project in Eclipse IDE. The two packages have been created under the project for page object model. Under the packages, there are different classes for calling the UI elements and page methods for performing the web elements. The selenium web driver libraries have been added in referenced libraries because they are external libraries. TestNG has added as an add-on for reporting. The classes name has created according to the task they perform.

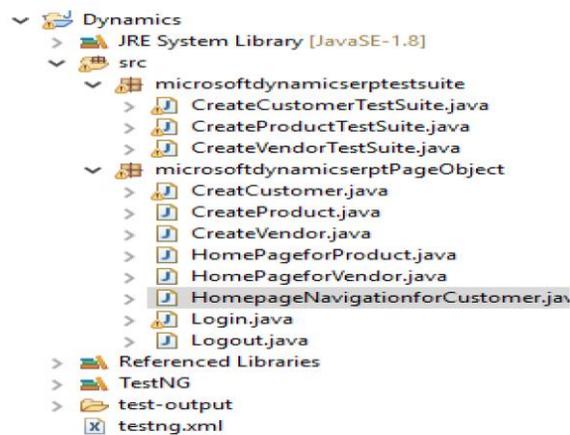


Figure 3: Structure of the project in Eclipse IDE

Figure 4 depicts about the login class. In this java class, the test scripts created for the login credentials. The browser has called in different class during the calling of all functions. Firstly, a construct has created in the class for calling the web driver. After that, a method has created to call the elements. These elements have called through the Xpath. Xpath is used to call the path expression for XML document. Thread sleep is provided for the time and the time is in milliseconds. The first Xpath is provided for the username and sendkeys are used to provide the data. The next Xpath is used to select the next button. After that, the Xpath has given for the password. The next Xpath is used for selecting the login button.

```

public class Login {
    WebDriver driver;

    public Login(WebDriver driver){
        this.driver =driver;
        //PageFactory.initElements(driver, this);
    }
    public void login() {

        // TODO Auto-generated method stub

        try
        {
            Thread.sleep(3000);
        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }

        //driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
        driver.findElement(By.xpath("//*[@name='loginfmt']").sendKeys("");
        driver.findElement(By.xpath("//*[@id='idSIButton9']").click());
        driver.findElement(By.xpath("//*[@id='idI118']").sendKeys("")); //put your password credentials
        //driver.findElement(By.xpath("//input[@value='Sign in']").click());
        WebDriverWait wait=new WebDriverWait(driver, 3);
        WebElement loginbutton= wait.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//input[@value='Sign in']")));
        loginbutton.click();
        driver.findElement(By.xpath("//*[@id='idBtn_Back']").click());
    }
}

```

Figure 4: Login Class

Figure 5 shows the navigation to create customer web page. In the page object model, there are always different classes for different web pages. In this image, the first Xpath has provided for selecting the navigation button on home page. Further, the Xpath has given for selecting the account payable link button which further open new tab. In this tab, the next Xpath has provided for the selection of all customers. Driver. Findelement is used as a command in selenium to call the Xpath.

```

package microsoftdynamicserptPageObject;

import org.openqa.selenium.By;

public class HomepageNavigationforCustomer {
    WebDriver driver;
    public HomepageNavigationforCustomer(WebDriver driver){
        this.driver =driver;
    }
    public void navigationforcustomer() {
        try
        {
            Thread.sleep(3000);
        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }
        driver.manage().window().maximize();

        //navigationpane
        driver.findElement(By.xpath("//*[@id='mainPane']/div[5]/div/div/div")).click();
        try
        {
            Thread.sleep(3000);
        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }
        //accountrecievable
        driver.findElement(By.xpath("//*[@id='mainPane']/div[5]/div/div[2]/div/a[2]")).click();
    }
}

```

```

3     driver.findElement(By.xpath("//*[@id=\ mainPane\ ]/div[5]/div/div[2]/div/a[2]")).click();
4
5     try
6     {
7         Thread.sleep(3000);
8     }
9     catch (InterruptedException e)
10    {
11        e.printStackTrace();
12    }
13    //allcustomer
14    driver.findElement(By.xpath("//*[@id=\mainPane\"]/div[5]/div/div[2]/div[2]/div[2]/div[3]")).click();
15
16    try
17    {
18        Thread.sleep(3000);
19    }
20    catch (InterruptedException e)
21    {
22        e.printStackTrace();
23    }
24    //newbutton
25    driver.findElement(By.xpath("//*[@id=\custtablelistpage_3_NewCustomer_label\")).click();
26    try
27    {
28        Thread.sleep(3000);
29    }
30    catch (InterruptedException e)
31    {
32        e.printStackTrace();
33    }
34    }
35    }
36    }

```

Figure 5: Navigation to create Customer Web Page

Figure 6 shows about creating customer details. There are mandatory fields and there are new fields added in this ERP system. Therefore, this regression automation testing has done in this project. There are many fields in this webpage and send keys are used as actions here for filling the values in textboxes.

```

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;

public class CreatCustomer {
    WebDriver driver;

    public CreatCustomer(WebDriver driver){
        this.driver =driver;
        //PageFactory.initElements(driver, this);
    }
    public void creatcustomer() {

        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_DynamicHeader_AccountNum_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_Org_Name_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_DynamicDetail_CustGroup_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_DynamicDetail_TaxGroup_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_LogisticsPostalAddress_ZipCode_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_LogisticsPostalAddress_City_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_LogisticsPostalAddress_State_input\")).sendKeys("");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_LogisticsElectronicAddressBP_Locator_input\")).sendKeys("#");
        driver.findElement(By.xpath("//*[@id=\DirPartyQuickCreateForm_4_OKButton\")).click();
        try
        {
            Thread.sleep(3000);
        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }

        driver.findElement(By.xpath("//*[@id=\custtablelistpage_3_SystemDefinedCloseButton\"]/div")).click();
    }
}

```

Figure 6: Creating Customer Details

Figure 6 below shows the methods to calling the Xpath methods from different classes. All methods have called here to execute the test suites. There are testing annotations used such as BeforeSuite, Test and AfterTest. The priority has set here to execute the multiple classes on given priority. The first method is created here to open the browser and URL.

```

.....
CreateVendor createvendorclass;
Logout logoutclass;
@BeforeSuite
public void setup() {
    System.setProperty("webdriver.chrome.driver", "D:\\chromedriver_win32\\chromedriver.exe");
    driver = new ChromeDriver();

    //System.setProperty("webdriver.ie.driver", "D:\\Nakush\\GDST_Level_07\\GDST_706\\706 resources for eclipse\\4 IE webdr
    //driver = new ChromeDriver();
    //driver = new InternetExplorerDriver();
    driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
    driver.get("https://login.microsoftonline.com/almliquor.onmicrosoft.com/wsfed?wa=wsignin1.0&wtrealm=spn%3a00000015-000

}

@Test(priority=0)
public void login() {

    loginclass =new Login(driver);
    loginclass.login();
}
@Test(priority=1)
public void createcustomernavigation() {

    navigationforcustomerclass =new HomepageNavigationforCustomer(driver);
    navigationforcustomerclass.navigationforcustomer();
}

@Test(priority=2)
public void createcustomer() {

createcustomerclass =new CreatCustomer(driver);

}

.....
}
@Test(priority=1)
public void createcustomernavigation() {

    navigationforcustomerclass =new HomepageNavigationforCustomer(driver);
    navigationforcustomerclass.navigationforcustomer();
}

@Test(priority=2)
public void createcustomer() {

createcustomerclass =new CreatCustomer(driver);
createcustomerclass.creatcustomer();
}

@AfterTest
public void logoutforvendedor() {

    logoutclass =new Logout(driver);
    logoutclass.logout();
}

```

Figure 6: Calling the Xpath Methods from different Classes

4. RESULTS

The results for this project have provided through TestNg. It is a free add-on for Selenium Web driver. TestNg stands for “next generation” which is a testing framework inspired by [7] JUnit and NUnit. TestNg mainly used annotations of java to write methods. TestNg has many advantages such as multiple annotations, data driven testing, multithreaded execution. However, the main advantage for this project is that to generate better reporting results for the execution of the test suites in XML and HTML report. Figure 7 shows the number of passed cases and number of time browser called for execution.

```

Starting ChromeDriver 75.0.3770.90 (a6dcdf7e3ec6f70a194cc25e8149475c6590e025-refs/branch-heads/3770@{#1003}) on port 16251
Only local connections are allowed.
Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
Jul 11, 2019 9:58:36 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Starting ChromeDriver 75.0.3770.90 (a6dcdf7e3ec6f70a194cc25e8149475c6590e025-refs/branch-heads/3770@{#1003}) on port 26651
Only local connections are allowed.
Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
Jul 11, 2019 9:59:20 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Starting ChromeDriver 75.0.3770.90 (a6dcdf7e3ec6f70a194cc25e8149475c6590e025-refs/branch-heads/3770@{#1003}) on port 31344
Only local connections are allowed.
Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
Jul 11, 2019 10:00:11 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
PASSED: login
PASSED: createcustomernavigation
PASSED: createcustomer
PASSED: logoutforcustomer
PASSED: setupforvendor
PASSED: loginforvendor
PASSED: homevendor
PASSED: createVendor
PASSED: logoutforvendor
PASSED: setupforproduct
PASSED: loginforproduct
PASSED: hompagenavg
PASSED: createproduct
PASSED: logoutforproduct
    
```

Figure 7: The number of Passed Cases

Figure 8 shows the emailable report which is a reporting result technique of testNg. In this figure the passed test scripts are highlighted with the green colour. However, if the test scripts are failed or skipped then they are highlighted in red colour.

Test	# Passed	# Skipped	# Failed	Time (ms)	Included Groups	Excluded Groups
Default suite						
<u>Default test</u>	14	0	0	139,071		

Class	Method	Start	Time (ms)
Default suite			
Default test — passed			
microsoftdynamicserptestsuite.CreateAll	createVendor	1562839197841	6524
	createcustomer	1562839144202	9830
	createcustomernavigation	1562839126269	17932
	createproduct	1562839242289	9321
	homevendor	1562839170326	27515
	hompagenavg	1562839221264	21023
	login	1562839117162	9105
	loginforproduct	1562839212328	8935
	loginforvendor	1562839161379	8946
	logoutforcustomer	1562839154034	4619
	logoutforproduct	1562839251611	4616
	logoutforvendor	1562839204365	4605
	setupforproduct	1562839208972	3355
setupforvendor	1562839158655	2723	

Figure 8: Emailable Report

5. DISCUSSION

The purpose of this project was doing the regression automation testing for Contoso. This testing has conducted because of the constant changes. However, this testing is done through selenium web driver by integrating with Eclipse IDE. The testing has completed by creating different packages and classes for the page object model. However, Reporting also generated through TestNg. According to the research papers, which are selected for this project have explained the importance of the regression automation testing. The scrum methodology is followed by the organization and due to constant changes in requirement the scrum team daily organize a meeting with the stakeholders to understand and inform them about the progress of project. Moreover, the research papers help to select and support the reason to select the selenium web driver for this project. Apart from that, Scrum methodology has selected for this project and there are many research papers available online for the scrum methodology. There are many research papers that emphasised the selection of scrum methodology. These research papers have explained the benefits of the scrum methodology and how this methodology is better from other methodologies. There are many issue and problems has faced during the project. The first started with the environment setup because the company is currently working on the project and every day, they are refreshing the websites because they are going to live the website application next month. Therefore, due to continuous changes and updating it was difficult to test the application. To overcome from this problem, the automation regression testing has started during non- working hours because that time it is easy to test the application. Another issue was to locate the web elements by Xpath. It was very complex to resolve this problem because in the automation testing Xpath is very essential part. However, this problem has overcome with the help of providing the class names on the place of Xpath. The purpose of selecting the happy path for the test scripts is because it was the requirement of the team and project. However, the other test scenario is also done but not with the help of automation.

6. RECOMMENDATIONS

The organization is planning to adopt the new tool which is designed by Microsoft for handle the regression testing. Regression Suite Automation Tool is the one which company wants to adapt. This tool will be helpful for them because it does not require coding and any manual efforts. This tool can be used by any of the user in the organization. However, the tool is very expensive and very hard to set the environment. So, I would recommend to first providing the training and workshops to the team. There are many related articles and videos available on internet. Also, Microsoft provides the live sessions and workshops to understand this tool. The organization should also understand that it is a very expensive tool and hard to set the environment so they should plan their budget and time according to their project requirements. Although, it is easy to use the tool and it also generate the test cases but still it is not easy to set up and it takes a lot of time.

7. CONCLUSION

In the conclusion, this project report covers the regression automation testing on Microsoft Dynamics 365 ERP solution for Contoso. The testing has done under the Selenium Web driver which is integrated with Eclipse IDE. In addition, the main scenarios of the testing were to check the functionality of creating customer, vendor and product web form. The user stories and epics have created for the regression testing on behalf of given requirements. After that, the test scripts

have created and executed in selenium web driver using page object model. This project covers the regression testing due to continuous changes on MD365. However, it was not easy to achieve this testing because company is currently doing a lot of modifications according to the requirements of the organization. Therefore, Selenium is the perfect for this project and it's an open source, so it also suits with the budget and cost. Moreover, TestNg is used for the reporting execution result. This framework is a free add-on for Selenium web driver. This framework helps a lot to achieve the structure of the execution by putting the annotations. Apart from that, the scrum methodology plays an important role in this project because it embraces the changes in requirement during the life cycle of the software development. Scrum methodology is more flexible and structured as compared to other methodology. The scrum team participating in daily stand-up meetings to understand the problems and progress of the project. The research papers help a lot to understand the concepts and it helps to support the cover topics in this project. In the end, these test scripts will help the organization to solve their issue and it also reduces the duplicity of the code so that it will be easy for them to use the code again and again without any problem.

REFERENCES

- [1] Onoma, A. K., Tsai, W. T., Poonawala, M., & Suganuma, H. (1998). Regression testing in an industrial environment. *Communications of the ACM*, 41(5), 81-86.
- [2] Sharma, M., & Angmo, R. (2014). Web based automation testing and tools. *International Journal of Computer Science and Information Technologies*, 5(1), 908-912.
- [3] Dustin, E. (2000, May). The Automated Testing Life-cycle Methodology (ATLM). In STAR EAST 2000 Conference, Orlando, Florida.
- [4] Leotta, M., Clerissi, D., Ricca, F., & Spadaro, C. (2013). Comparing the maintainability of selenium webdriver test suites employing different locators: A case study. In *Proceedings of the 2013 International Workshop on Joining AcadeMiA and Industry Contributions to Testing Automation*, ser. JAMAICA (pp. 53-58).
- [5] Mahalakshmi, M., & Sundararajan, M. (2013). Traditional SDLC Vs Scrum Methodology–A Comparative Study. *International Journal of Emerging Technology and Advanced Engineering*, 3(6), 192-196.
- [6] Balaji, S., & Murugaiyan, M. S. (2012). Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, 2(1), 26-30.
- [7] Gojare, S., Joshi, R., & Gaigaware, D. (2015). Analysis and design of selenium webdriver automation testing framework. *Procedia Computer Science*, 50, 341-346.

First Author

Nakush Sharma has experienced of developing and testing in India. She has done Graduate Diploma in Software Testing from AGI Education, Auckland. She has completed her conference paper on the basis of her internship. She is expertise in different automation tools for testing and development tools.



Second Author

Dr. Shahid Ali is a senior lecturer and IT program leader at AGI Education Limited, Auckland, New Zealand. He has published number of research papers on ensemble learning. His expertise and research interests include ensemble learning, machine learning, data mining and knowledge discovery.