

Title Page

**APPLYING AUTOMATION TESTING FRAMEWORK AS AN  
EFFICIENT MEDIUM FOR TESTING WEB APPLICATIONS**

Dissertation submitted in part fulfilment of the requirements for the  
degree of M.Sc. in Information Systems with Computing at  
Dublin Business School

Shravan Kumar  
Student Id: 10319743

M.Sc. Information Systems with Computing

22<sup>nd</sup> August, 2016

## Declaration

I, Shravan Kumar declare that this research is my original work and that it has never been presented to any institution or university for the award of Degree or Diploma. In addition, I have referenced correctly all literature and sources used in this work and this this work is fully compliant with the Dublin Business School's academic honesty policy.

Signed: \_\_\_\_\_

Shravan Kumar

Date: 22-August-2016

## Acknowledgements

This piece of work could not have been possible without the guidance and support from my supervisor, Harnaik Dhoot. I wish to thank him for his comments, feedback and patience.

I also want to express my gratitude to my course coordinator Greg South who was kind enough to help me whenever required. I also want to thanks to Shazia Afzal, my data analytics teacher. I am thankful to all my friends who participated and helped me during data collection.

Finally, I would like to thank my family and my wonderful wife, Priyanka for everything.

-Shravan Kumar

## Abstract

Testing is one of the technique to ensure the quality of the product by finding bugs. It becomes essential to improve the quality of the product by increasing the test coverage. This paper gives the idea of automation testing by using a reusable framework. The main goal of this research is to design a Hybrid Test Automation Framework to test web bases applications. A framework which can take input data from Excel file and execute the test cases. It also creates test reports, notifies the user via email and capture screenshots in the event of failure. The idea is to give a concept of modular and reusable automation framework.

## Contents

Title Page.....	i
Declaration.....	ii
Acknowledgements.....	iii
Abstract.....	iv
List of Figures.....	viii
List of Tables.....	ix
Chapter 1: Introduction.....	1
1.1 Rationale behind automation testing.....	2
1.2 Problems of automation testing.....	2
1.3 Objectives of Research.....	3
1.4 Scope and limitations of this research.....	3
1.4.1 Scope of the Research.....	3
1.4.2. Limitation of the Research.....	3
1.5 Dissertation roadmap.....	4
1.6 Major contribution of this research.....	4
Chapter 2: Literature Review.....	5
2.0 Introduction.....	5
2.1 The need for test automation.....	5
2.1.1 Should one automate or not?.....	6
2.1.2 Advantages of automation testing.....	7
2.1.3 When test cases should be automated.....	7
2.1.4 When test cases should not be automated.....	8
2.1.5 Limitations of automation testing.....	8
2.2 Tool for the web automation testing.....	8
2.2.1 Criteria for automation tool selection.....	9
2.2.2 Selenium.....	10
2.2.3 Advantages of Selenium.....	11
2.2.4 Limitations of Selenium.....	11
2.2.5 Quick Test Professional (QTP).....	11
2.2.6 Advantages of QTP.....	12
2.2.7 Limitation of QTP.....	12
2.2.8 Comparison between Selenium & QTP.....	12
2.2.9 Conclusions.....	13
2.3 Automation Framework.....	13
2.3.1 Introduction to Framework.....	13

2.3.2 Data-driven automation framework .....	14
2.3.3 Advantages of data-driven framework .....	15
2.3.4 Disadvantages of data-driven framework.....	15
2.3.5 Keyword driven framework .....	15
2.3.6 Advantages of keyword driven framework.....	16
2.3.7 Disadvantages of keyword driven framework .....	16
2.3.8 Hybrid automation framework.....	16
2.3.9 Structure of hybrid framework.....	17
2.3.10 Conclusion .....	18
<b>Chapter 3: Research Methodology and Methods.....</b>	<b>19</b>
3.1 Introduction.....	19
3.2 Research Methodology .....	19
3.2.1 Research Philosophy .....	20
3.2.2 Research Approach .....	21
3.2.3 Research Strategy.....	21
3.2.4 Time horizon.....	22
3.2.5 Sampling .....	22
3.2.6 Data Collection .....	22
3.2.7 Data Collection Instruments .....	22
3.2.8 Questionnaire.....	23
3.2.9 Data Analysis Procedure .....	24
3.2.10 Research Ethics.....	25
3.2.11 Limitations of Methodology .....	25
<b>Chapter 4: Artefacts Design and Development.....</b>	<b>26</b>
4.1 Introduction.....	26
4.1 Selection of software development model.....	26
4.1.1 Advantages of the Prototype Process Model.....	27
4.1.2 Disadvantages of Prototype Model .....	27
4.2 Requirement Analysis .....	27
4.2.1 High-level requirements for a framework.....	28
4.2.2 Breakdown of functional requirements.....	28
4.3 Design .....	30
4.4 Implementation.....	33
4.4.1 Hybrid automation framework development.....	33
4.5 Test Execution .....	41
<b>Chapter 5: Research Results .....</b>	<b>45</b>

5.1 Introduction.....	45
5.2 Qualitative Research Methods .....	45
5.3 Research Findings and Analysis .....	45
5.3.1 Professional Background of Respondents .....	45
5.3.2 Professional Experience of Respondents .....	46
5.4 Summary.....	56
<b>Chapter 6: Research Discussion .....</b>	<b>56</b>
6.1 Introduction.....	56
6.2 Research conclusions .....	56
6.3 Research question achieved .....	57
6.4 Limitations of research.....	57
6.5 Research Recommendations .....	58
<b>Chapter7: Reflection and Learning.....</b>	<b>59</b>
7.1 Introduction.....	59
7.2 Learning Styles.....	59
7.3 Reflection on learning .....	60
7.4 Skills developed.....	60
7.5 Future Application.....	61
<b>Bibliography .....</b>	<b>62</b>
<b>Appendix 1: Research Plan Grid .....</b>	<b>68</b>
<b>Appendix 2: Questionnaire.....</b>	<b>69</b>
<b>Appendix 3: Focus Group Report .....</b>	<b>70</b>
<b>Appendix 4: Survey Questionnaire Answers.....</b>	<b>71</b>

## List of Figures

- Figure 2.1: data-driven framework
- Figure 2.2: Sample Input test data prepared by the researcher
- Figure 2.3: Keyword driven framework
- Figure 2.4: Keyword files
- Figure 2.5: Hybrid framework
- Figure 2.6: Structure of hybrid framework
- Figure 3.1: Research Onion
- Figure 3.2: Research approach
- Figure 4.1: Prototype Model
- Figure 4.2: Driver scripts interacting with Application under test using libraries
- Figure 4.3: High-level architecture of Hybrid automation framework
- Figure 4.4: Structure of config package
- Figure 4.5: Test data file
- Figure 4.6: Test case suite
- Figure 4.7: Test scripts for Login
- Figure 4.8: Config property
- Figure 4.9: Keyword file
- Figure 4.10: Object repository
- Figure 4.11: Test app folder
- Figure 4.12: Excel Reader file
- Figure 4.13: Utility files
- Figure 4.14: Test report package
- Figure 4.15: TestNG
- Figure 4.16: Jar files
- Figure 4.17: Log4j file
- Figure 4.18: Test report folder
- Figure 4.19: Test report
- Figure 4.20: Screenshot path
- Figure 4.21: Screenshot for error
- Figure 4.22: Email notification
- Figure 4.23: Application Log
- Figure 4.24: Selenium log
- Figure 5.1: Profession of respondents
- Figure 5.2: Professional experience of respondents
- Figure 5.3: Respondents involvement in testing
- Figure 5.4: Respondents preference
- Figure 5.5: Reasons to select automation over manual
- Figure 5.6: Tools used by respondents
- Figure 5.7: Features of automation tool
- Figure 5.8: Automation saves time and effort
- Figure 5.9: Framework choice of respondents
- Figure 5.10: Features of automation framework
- Figure 5.11: Framework type used by respondents
- Figure 5.12: Framework used by respondents
- Figure 5.13: Hybrid framework features
- Figure 5.14: Events handling by framework

Figure 5.15: Programming skills required for automation

Figure 5.16: Is framework maintainable

Figure 5.17: Is framework compatible with other tools

Figure 7.1: Kolb's Learning Style

## List of Tables

Table 2.1: Criteria for automation tool selection

Table 2.2: Comparison between Selenium and QTP

Table 4.1: High-level requirement for a framework



## Chapter 1: Introduction

Software testing is one of the crucial parts of software development process and its importance growing every day. Studies say that it is very essentials for developing a high-quality software product. According to ANSI/IEEE 1059 standard, Testing can be defined as – “A process of analyzing a software system to detect the differences between existing and required conditions (that is defects/errors/bugs) and to assess the features of the software item” **(Kaur and Singh, 2014)**.

**Kaner (2008)** defines testing as “an investigation conducted to provide stakeholders with information about the quality of the product or service under test”. Therefore, the primary goal of software testing is to find bugs and improve the expected behaviour of the product **(Constantinescu and Iacob, 2008)**.

Since the complexity growing for the web applications, hence the testing process is becoming complex nowadays. A similar study from the **University of California** claims that “testing is the inevitable part of any responsible effort to develop a software product.” **(Chandrababha et al., 2015)**.

A Software can be tested in two ways- either manually or using some tools. **C. Ebert (2012)** claimed that testing work consumes 30 to 60 percent of all life-cycle cost, and it varies according to the complexity and criticality of the product. It is important to control and reduce the testing costs to make a profit. Since this research explored that testing improves the quality of the product, hence it cannot be cut short. Here test automation plays a key role. Test automation reduces costs and increases the quality of the testing tasks **(Polo et al., 2006)**.

Today most software applications are web-based and runs run in an internet browser. Agile methodology is one of the popular software development methods nowadays. In this approach, requirement changes very frequently and hence the product needs to be tested more frequently for better quality. The requirement for better quality means more workload on software testers. **Laukkanen (2006)** mentioned the above situation in his research and claimed the reason why test automation is popularly becoming a necessity for software projects. Test automation is always the answer for repeated tests. Test automation means using an automation tool to run repeatable tests for the application under test.

Regression testing is best suited for automation testing **(Introduction — Selenium**

**documentation, 2008).**

**Laukkanen (2006)** also stated in his research that a core component of testing is the test case. Test cases have pre-conditions, test data to set up a test environment and steps to reproduce against the application under test (AUT). Test automation controls the execution of tests, provides the comparison of actual output and expected output, defines the test preconditions and generates test reports.

However, test automation cannot exist in isolation. **Johnson and Foote (1988)** describes in their article that a system must require for implementing some core functionality like test execution, monitoring & reporting to support test automation. This system must be scalable to create a new type of test cases and achieve the reusability. This kind of system is known as test automation frameworks.

This research focuses on the concept of automation framework for the purpose of the testing web application.

### **1.1 Rationale behind automation testing**

This research is having some rationale behind automation testing. According to **Fewster and Graham (1999)** – Automation tests makes regression testing faster. It helps to run more test more frequently. It is a way to utilize the resources in a better way. It is consistent and reusable. It is easy to maintain. It provides better test coverage. The user runs automation tests with the intention to find more bugs. It saves time and money.

### **1.2 Problems of automation testing**

This research investigates about automation testing in terms of efficiency and reusability. The expectation from automation testing is that it will run many test cases repeatedly on different versions of the application under test. It is expected that it will lower the workload and help testers from their repeating tasks. Ultimate goal of automation testing is to improve the quality of software product and reduce the testing time.

Above benefits make automation testing attractive and promising. However, study says that achieving these features are not simple. It demands a considerable amount of hard work along with planning. According to All these promises make test automation look attractive but achieving them in real life requires plenty of hard work. **Fewster and Graham (1999)** claimed that unrealistic expectations and poor test practices are main reasons behind not achieving the expected result.

Research by **Kaner et al. (2002)** describes that automation projects might fail if it does not

follow the required process. It can also fail if it is unmanaged. There must be a proper approach required for test automation.

### **1.3 Objectives of Research**

The objective and purpose of this research are to provide a comparative view of available automation frameworks and design a reusable hybrid framework for the testing of web applications.

#### **Explanation of the research question**

The main goal of this research is- to create proof of concept for hybrid automation framework. The study follows the below steps-

1. Define the requirement for test automation
2. Define the suitability of automation tool
3. Discuss different automation frameworks
4. Design a framework which meets the requirement
5. Implement the designed framework (prototype)
6. Test the designed framework against the defined requirements
7. Collect test results and validate the feasibility of the framework
8. Adapt changes based on the information if necessary

### **1.4 Scope and limitations of this research**

This section discusses the scope & limitations of the research. Automation testing is a vast subject, and it has many dimensions. Therefore, it was difficult to cover all aspects in a single research.

#### **1.4.1 Scope of the Research**

This research is focused on designing and implementing a hybrid automation framework.

The framework must be able to read & write inputs from Excel file, execute tests, generate test report and handle exceptions. It should also generate logs for every test execution. The framework must be reusable and maintainable.

#### **1.4.2. Limitation of the Research**

This research does not discuss the support for test planning and design. It also does not discuss the effect of automation testing in testing process. This research does not focus on the timing of automation testing. For e.g. when is the right time to execute automation test? What to automate and How to automate are beyond the scope of this research.

Estimation of effort and cost saved by adopting automation testing is also not part of this

research.

### 1.5 Dissertation roadmap

This research has divided into broadly seven chapters. Each chapter discusses and focuses on a particular area.

**Chapter 2** discuss literature review. It defines the importance of automation testing and why it is required. It also describes the selection of automation tool based on several factors. After that, it discusses the different automation frameworks and their pros & cons. It talks about how the hybrid framework meets the requirement and provide a reusable framework.

**Chapter 3** discuss the different methodology accepted during this research. It also discusses the data collection and questionnaire part.

**Chapter 4** discuss design and implementation of a hybrid framework which meets the research objectives. This section executes the tests after implementing prototype framework and collect the results for analysis purpose.

**Chapter 5** discuss the qualitative data collection, test results, and findings. Based on the results, the suggested framework evaluated, and some changes made to the requirement set which is defined earlier.

**Chapter 6** talks about the conclusion. It evaluates how the objective of the research has accomplished. Also, it lists out the things which could not achieve in this research.

**Chapter 7** discuss reflection and learning.

### 1.6 Major contribution of this research

This research highlights the importance of automation testing. It discusses how time & effort can save during testing process which ultimately improves the quality of the product. It discusses automation tool and hybrid framework for test automation. It explains the benefits & features of the hybrid framework. Its major contribution is to develop a prototype framework as a proof of concept which is reusable. This research ultimately shows the way to streamline the testing process.

## Chapter 2: Literature Review

### 2.0 Introduction

This chapter consists of three major sections. The first section provides a brief discussion over the need for automation testing. The second section reviews the suitability of automation tool for automating complex web applications. It also discusses the features of automation tool and its compatibility with different environments. The third section gives an idea about automation framework which can be evolved using the automation tools along with other supporting tools.

### 2.1 The need for test automation

Every company tries to complete testing of its product before delivery to the client. Testers try hard to find the bugs as much as they can. **Hayes (1995)** says that if the application is large and complex, then manual testing requires more time. Also, there is a chance of human error which leads to compromise the quality of the product. According to **Crispin (2009)**, the agile development environment has created pressure for frequent changes and implementation of new functionalities. It has increased the pressure on software testers to improve the test coverage in a short period. **Mailewa et al. (2015)** claim in his research that certain tests are difficult to perform manually in different environments. Since manual testing can be repetitive and boring, hence accomplishing good test coverage with the high-quality product is a difficult task.

Companies want to test software not only effectively, but also as promptly and methodically as possible. Organizations are moving towards automated testing to accomplish this objective (**Ghanakota, 2012**). Since manual testing requires exhaustive testing and consequently utilizing man hours, hence using automation for software application testing is one of the best option (**Chandraprabha, Kumar, and Saxena, 2015**). Software test automation is used to run the different test cases in a different environment in a lesser time. It also monitors the errors and logs. It compares the actual with expected output and shows the test result accordingly. It enforces preconditions required for testing. It provides the test results after completion (**Laukkanen, 2006**).

Research done in the automation area by **Mailewa et al. (2015)** claims that the automation testing enhances test scope. It improves accuracy in a short time at low cost. As he says- the main advantage of automation testing is that it optimizes the testing process and helps testers to increase their efficiency of test coverage without investing huge money. It is

faster compare to manual testing.

According to the interpretation of **Ying Wu's** work by **Mehta et al. (2003)**- manual tasks are involved during the user acceptance testing of the web-based application. There is a possibility of human error while performing tasks manually. It is time-consuming as well.

**Mehta et al. (2003)** also explained in his research that – web based application contains complex business logic which needs to test thoroughly. These applications have short release cycles. It demands the web application test cases be more comprehensive and efficient. The difficulty and pressure are also high for the web application testing. **Hayes (1995)** demonstrates for automation testing in place of regression testing. The web application needs repetitive testing, and it makes the testing tedious to accomplish.

Nowadays most of the software applications are web-based which runs on different platforms and different browsers. Hence, testing these web based applications are a big challenge. It requires much effort. If development environment is Agile, then testing becomes very frequent and necessary. Test automation is always the answer to address these problems. When there are repeatable tests to run, then automation tools help. For repeated tests like regression testing, automation is very helpful. There are many benefits to testing automation. It is related to frequent test execution within a specified period **(Introduction — Selenium documentation, 2008)**.

**Lamba, Rishiwal, and Rana (2015)** claims in their research that test automation optimizes the testing process. It improves the long-term effectiveness of the testing process.

It supports:

- Repeated testing like Regression testing
- Quick feedback to developers
- Repetition of test case execution
- Extreme programming support
- Systematic documentation for test cases
- User can customize the test report template
- Finding hidden bugs missed by manual testing **(Introduction — Selenium documentation, 2008)**

### 2.1.1 Should one automate or not?

The question arises here for discussion is that whether automation is beneficial or not? Is

automation always helpful? At what point of time testers should decide to automate their test cases? **Rahman (2014)** demonstrates in his research that automation is not always fruitful. There are times when manual testing might be more useful. For example, if the developers have made many changes in their application's user interface then manual testing is preferable. Also, if the build is not stable, then the user cannot automate it. The unstable build can break and will fail the automation effort. In this case, again manual testing is the only option. When the timeline is very strict as it usually happens in a case of extreme programming, then automation testing is not useful. Automating new test cases take a lot of efforts and time to run properly. However, it is easy to do manual testing when the user has the short duration of time. Hence manual testing is appropriate in these cases.

### 2.1.2 Advantages of automation testing

**Lamba et al. (2015)** explain in his research that while regression testing testers perform repeated testing of new as well as the existing features which takes a lot of man hour. Also, there is a possibility that testers miss something due to human error. If automated tests are available, then it can be executed any number of times in a short duration. According to **Mailewa et al. (2015)**- the big advantage of automation is that testers can perform regression testing more efficiently. If test automation script has written correctly and it covers all test cases, then automation also increases the test coverage. Automation testing increases the reusability of the test cases. Maintenance of test scripts is not complex. It is a one-time effort. Also, maintaining test data is completely independent of updating the test scripts. Test data can be updated easily without changing the main test scripts. Automation reduces the manual effort and hence saves the time. Test automation is beneficial for long term. Even if any employee leaves the organization then also it will not affect the quality of the product. Research says that automation improves the test coverage and saves time and cost.

### 2.1.3 When test cases should be automated

**Rahman (2014)** explains in his research that if any test needs to run frequently for an application, then that test need to be automated. Automation testing also has some shortcomings. It is not possible to automate all type of testings. E.g.– usability testing, user interface testing. However, in the case of Regression testing, sanity testing or data based

testing, automation is very useful since it involves repeated testing. The automation is suitable for

- Vast and complex projects
- Build or application is stable
- Repeated testing of same test cases
- the application runs on multiple platforms
- when manual testing is time-consuming
- applications having the high severity of bugs in manual testing

#### 2.1.4 When test cases should not be automated

Test automation should not perform in the below cases:

- If it is end user testing
- If the test is not repetitive
- If the build or application is not stable
- If the tester is inexperienced
- If time is not sufficient for automating tests (**Introduction-Selenium Documentation, 2014**)

#### 2.1.5 Limitations of automation testing

Even though automation has been advantages but studies says that automation testing has its limitations. It is not 100% full proof. These disadvantages include:

- Tester must be skilled and experienced
- Automation tool itself can have defects.
- Debugging is still a challenge
- Maintenance of automated script is costly, and it needs update time to time (**Rehman, 2014**)

### 2.2 Tool for the web automation testing

This section discusses the feasibility of right tool for automation testing. **Kaur and Gupta (2013)** proposes in their research that before we start evaluating the right automation testing tool, it is important to prepare a list of criteria when choosing a tool for evaluation. Evaluation criteria are crucial. Otherwise, we may waste time downloading, installing and evaluating tools without any objective.

**Sharma and Angmo (2014)** demonstrates in their research that before choosing the best tool for automation, there are many factors which need to consider. First thing, its compatibility with the existing environment. Secondly, how it performs in the environment with network traffic and load. Also, the tool must be compatible with the design and implementation of the application. These evaluations must complete for cost and performance.

### 2.2.1 Criteria for automation tool selection

Automation tools can compare to different parameters. Some of them have similar characteristics, and they serve the same purpose. There are few basic criteria based on that testers can evaluate the tool for their testing activity. Research into the tool evaluation criteria suggests below criteria (**Chmur̃ciak, 2013, p. 19**):

Supported platforms	Does it support different operating systems and their versions? Does it also provide support for mobile operating systems?
Supported programming languages	Automation tool should support all object-oriented programming languages.
Supported web browsers	Does it support all popular web browsers? Most web based applications support multiple browsers. Does it support mobile applications?
GUI for the test development	Does it provide any plugin/add-ons to the browser for record & play? Does it include other applications which can be used to record test scenarios?
Parallel test execution	Does it possible to execute tests in parallel and in distributed environment?
Compatibility with other automation tools	Options to integrate the tool with other supporting testing tools.
Technical support	Does user support and documentation available?
Licensing and price	Is tool open source or paid one? Does any licensing requirements for commercial use?

**Table 2.1:** Criteria for automation tool selection

**GmbH (2013)** writes that web-based applications are growing. It has observed that these web based applications are having some degree of complexity. It becomes difficult to select an automation tool which fulfills all the needs of automation testing. Since automation testing has more benefits over manual testing, hence many companies started developing various automation tools for different applications.

There are two types of automation tools available in market-

1. Open source test tools
2. Commercial test tools

**Open Source Test Tools**- These automation tools are free for the users. It can download from the internet without any charges e.g. Selenium, waiter.

**Commercial test tools**- These test tools are not free and need a license to use e.g. QTP, Load Runner (**Selecting the right automated testing tool, 2016**).

This research evaluates briefly about two popular automation tools that are Selenium & QTP by their characteristics, test execution capability, reusability, maintainability, and reporting. Discussion over other automation tools is out of the scope of this research.

### 2.2.2 Selenium

Selenium was released in 2004 and initially developed by Jason Huggins for testing web applications. Selenium official documentation says that- “it is an open source software testing tool used as interface while testing web applications” (**Selenium WebDriver — Selenium documentation, 2008**).

According to **Bhagyashree Bhondokar et al. (2015)** demonstrates in his research that Selenium uses web driver, and it supports different browsers and multiple platforms. Web driver provides object-oriented API that gives enhanced backing for testing issues of cutting edge web applications be the fundamental high point of Web Driver. It gives native support to browsers by making direct interaction with programs. It supports multiple platforms and also supports browsers like Internet Explorer, Mozilla Firefox, Chrome, Safari.

**Chandrabha, Kumar, and Saxena (2015)** explain about selenium web driver in their research that it supports wait for conditions, handling flash objects, multiple language support. Selenium WebDriver fundamentally works in two ways first find the web element i.e. object and after that perform event & activity on them. It identifies web objects by using

HTML id, name, XPath, CSS. Selenium also provides huge libraries for a testing web application.

**Elizabeth Hendrickson** from Sauce Labs did a market survey and found that the Selenium being the leader with the counts of jobs specifying in automation technology ("The Future of Testing," 2010).

**Metha et al. (2003)** summarize the advantages & limitations of Selenium automation tool in his research below-

### 2.2.3 Advantages of Selenium

1. It is available free of cost and download and runs the test.
2. Test cases can write in any high-level programming languages, such as Perl, Net, PHP, Ruby, Java and Python.
3. User Acceptance Test is carried out with Selenium.
4. Test execution can be done in parallel and does not require any dedicated machine.

### 2.2.4 Limitations of Selenium

1. It supports only web application. It is not compatible with the Windows application.
2. There is no official technical support available.

### 2.2.5 Quick Test Professional (QTP)

Quick Test Professional is known as QTP which has a graphical interface to record and playback user actions. It is a commercial tool and requires a license from HP. Its trial version can download from the official website of HP (**HP QuickTest Professional for business process testing user guide, 2010**).

**Kaur and Gupta (2013)** explains about QTP in their research that- it supports web-based applications as well as windows based application for automating the functional testing and regression testing. Applications which are developed using Java or .net languages can also test by using QTP. It identifies objects from user interface or web page and saves into object repository. It uses object repository for performing actions.

According to **Metha et al. (2003)** in his research claims that QTP uses VBScript language for writing test scripts. QTP is helpful for UI-based test case automation. However, it can use for writing functional tests as well.

### 2.2.6 Advantages of QTP

1. A non-programmer can also easily use QTP to create test cases.
2. Strong object identification feature
3. QTP supports VBScript which is easy to learn.

### 2.2.7 Limitation of QTP

1. Mostly supports Windows environment
2. It supports limited browsers
3. Licensing cost is high
4. Execution time is more compared to other tools

### 2.2.8 Comparison between Selenium & QTP

Several researchers (Gupta, Kumar, and Saxena, 2015; Selukar, 2013; Metha, 2003) explains the comparison between Selenium and QTP as-

Features	Selenium	QTP
Cost	Open source	Paid
Language support	Java, C#, Python, Ruby, Perl, PHP	Only VBScripts
Application support	Only web based	Desktop & Web-based both
Operating system	Windows, Linux, Android	Windows
Browser support	IE, Chrome, Firefox, Opera, HtmlUnit	IE, Chrome, Firefox
Reports	Quality centre	Integrate with TestNG, Junit
Programming skill	Good programming skill required	Less programming skill required
Execution speed	Faster than QTP	Slower than Selenium
Product support	Open source community	Dedicated support available
Test development environment	Test scripts can develop in various IDEs like Eclipse, Visual Studio, NetBeans	Test scripts can be developed only in QTP.

**Table 2.2:** Comparison between Selenium and QTP

## 2.2.9 Conclusions

Selenium and QTP are a most popular tool for automation frameworks. Both tools have its advantages and disadvantages. One can choose a testing tool depends on the requirement for their testing. QTP is a commercial tool and because of its high-cost individual cannot afford it. Whereas Selenium, an open source tool and available free of cost. This research selecting Selenium automation tool for developing proof of concept because it is available free of cost.

## 2.3 Automation Framework

This section discusses the different automation frameworks and their structure. Research provides a comparative study of data driven, keyword drive and hybrid-driven framework.

### 2.3.1 Introduction to Framework

**Ghanakota (2012)** says that “a testing framework is an execution environment for the automated test cases.” It is overall a system where test scripts Develop and execute. A framework is supposed to be a set of protocols, rules and guidelines that can consolidate into a single package. Every automation tool needs a framework to execute and maintain test cases.

According to **Grater (2005)** in this research about the requirement of the test automation framework. Test execution should be fully automatic. It should be easy to use and maintainable. The advantages of the framework are- scripting, scalability, modularity, reusability, cost & maintenance. Automation testers need a framework to get all these benefits

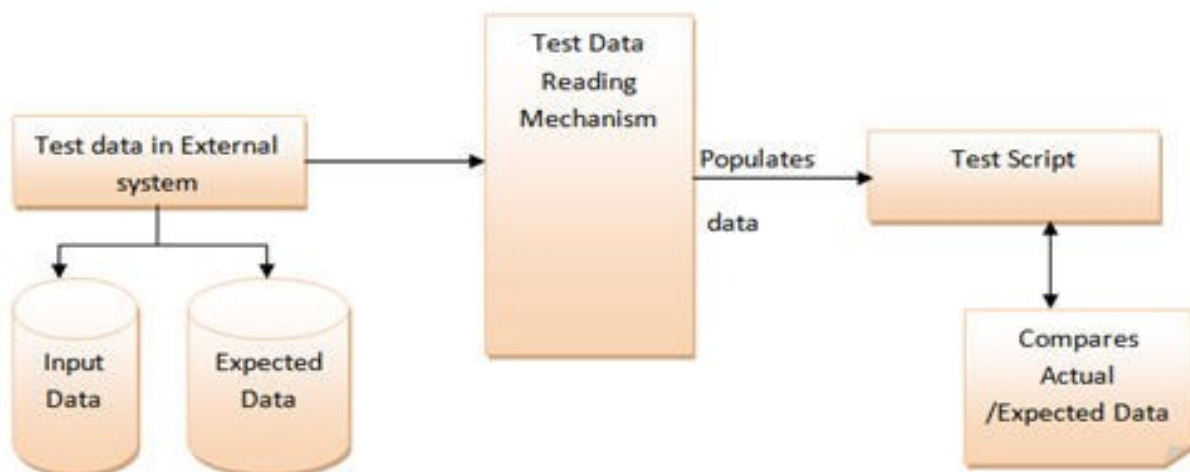
As per the interpretation of **Laukkanen (2006)** of **Fewster and Graham’s** work- he lists out the capabilities of an automation framework:

1. It must be capable of executing test automatically.
2. It must handle errors.
3. It must be able to verify expected output with actual output and verify results.
4. It must provide status to the executed test. E.g.- pass, fail, error, warnings, info
5. It must create a detailed logging.
6. It must generate a test report.

This research discusses the different type of automation frameworks and their feasibility.

### 2.3.2 Data-driven automation framework

**Laukkanen (2006)** demonstrates his point about the data-driven framework in his research. He says that- some test automation scripts have test data rooted into them. Test data creates a problem. The user needs to change test script if he wants to modify the test data. During automation testing, it is required to test same functionality with a different set of input data. In that situation, if test data attached with test script then it is hard to change every time. It is better to keep test data separately from the test script. It will make it easy to update test data without disturbing the test script. A better solution is reading the test data from an external file and executing test script based on it. This method is known as data-driven framework, and it has shown in figure **(Data Driven Testing Framework, 2016b)**



**Figure 2.1:** data-driven framework

Test data files can save in the spreadsheet as comma separated value (CSV) or tab separated value (TSV). Apache POI jar files can use for reading and writing from Excel files **(Apache POI - Java Excel APIs, 2016)**.

Below is the sample input test data for flight booking-

TO	FROM	TIME	DATE
Delhi	Mumbai	10:00	10-Aug-16
Delhi	Kolkatta	13:00	20-Sep-16
Coachin	Chennai	14:10	5-Aug-16
Delhi	Patna	22:25	11-Oct-16
Bangalore	Lucknow	10:05	22-Sep-16
Delhi	Coachin	11:10	25-Sep-16
Chandigarh	Bangalore	21:00	11-Aug-16
Lucknow	Mangalore	15:20	15-Aug-16

**Figure 2.2:** Sample Input test data prepared by the researcher

### 2.3.3 Advantages of data-driven framework

**Ghanakota (2012)** claims that data-driven approach helps to reduce the number of test cases. It also reduces the amount of code for creating test cases. **Laukkanen’s (2006)** interpretation of Kaner’s work explains that updating and developing tests is easy and requires less programming skills.

### 2.3.4 Disadvantages of data-driven framework

The test cases created for data-driven framework are similar in nature and creating different tests requires programming effort for new driver script (**Ghanakota, 2012**)

### 2.3.5 Keyword driven framework

Since data driven tests are similar in nature and creating different test needs more effort.

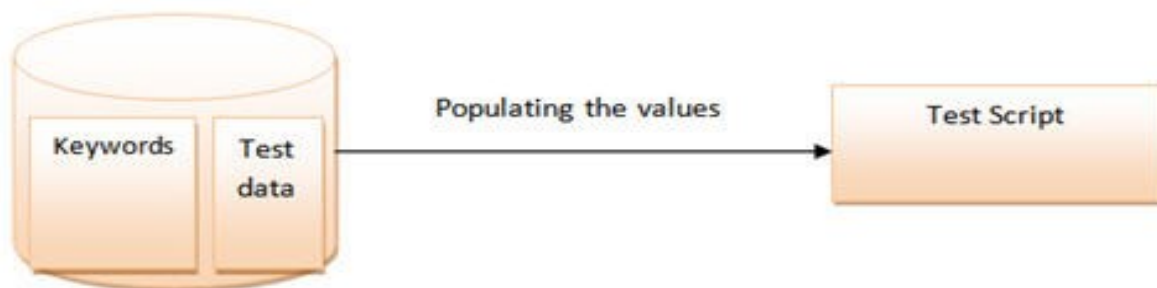
Hence **Rashmi and Bajpai (2012)** suggest a keyword driven approach in their research.

According to them some directives are present for test data and put into external files.

These directives tell what to do with the test data. These directives are known as keywords.

As per the interpretation of **Fewster and Graham’s** work, **Laukkanen (2006)** claims that it is an extension of the data-driven approach.

Once the test data parsed then, driver script must be able to interpret keywords and execute the specified action using assigned arguments (**Fewster and Graham, 1999**).



**Figure 2.3:** Keyword driven framework (**Data Driven Testing Framework, 2016a**)

Test data files and keyword files can save in the spreadsheet as comma separated value

(CSV) or tab separated value (TSV). Below is the sample input keywords file for the login event-

TSID	Decription	Keyword	Object
TS01	Wait For 10 sec	waitfor	email
TS02	Input data in email	input	email
TS03	Click on Next	clickLink	next
TS04	Validate EmailId	getText	erroremail
TS05	Input data in Password	input	passwrđ
TS06	Click on Signin	clickLink	signin
TS07	Click on Compose	clickLink	compose
TS07	Wait For 10 sec	waitfor	to
TS08	Input in To Field	input	to
TS09	Input in Subject Field	input	subject
TS10	Input in Message Field	input	message
TS011	Click on Send	clickLink	send
TS012	Click on logout	clickLink	logout
TS013	Click on signout	clickLink	signout

**Figure 2.4:** Keyword files (**Data Driven Testing Framework, 2016a**)

### 2.3.6 Advantages of keyword driven framework

**Randhawa and Pandey (2006)** states that keyword driven framework has all the benefits of data driven framework. No special expertise require for maintaining test cases. Same keywords can use across different test cases.

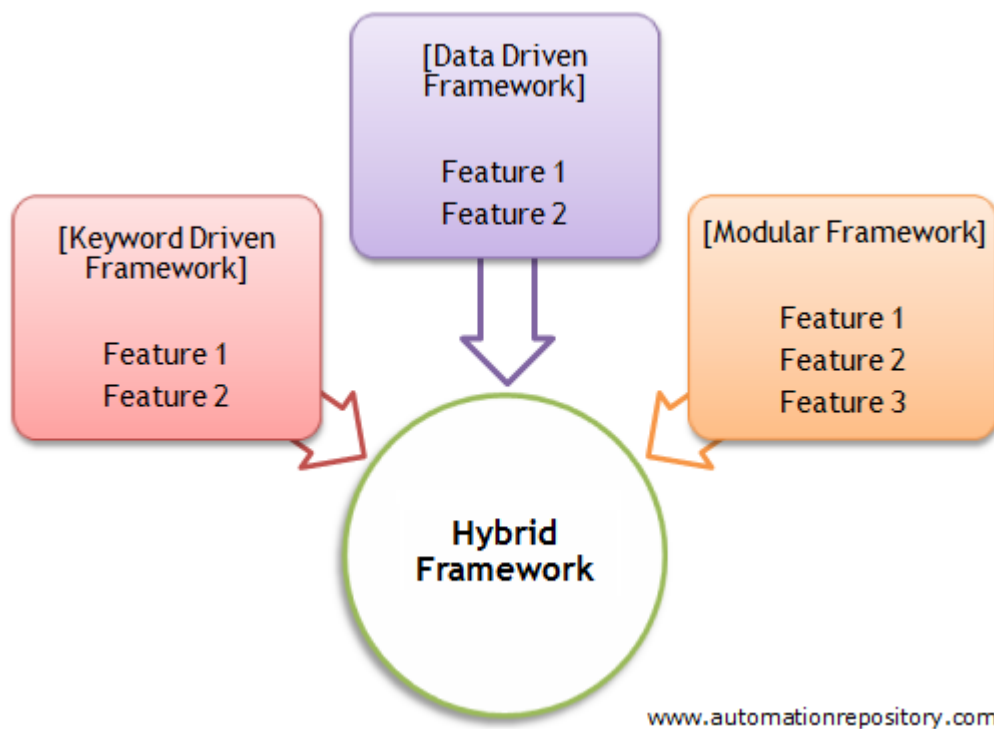
### 2.3.7 Disadvantages of keyword driven framework

**Rashmi and Bajpai (2012)** explains the disadvantages in their research by saying that keyword driven requires complex framework than data-driven. Test cases get complex due to implementing keyword features. **Ghanakota (2012)** suggests to combine the strengths of all the frameworks and remove their weaknesses we need to use the hybrid testing framework.

### 2.3.8 Hybrid automation framework

**Pillai (2011)** explains in his article that Hybrid framework is a framework which has created by using features of other frameworks. Every framework has its advantages & disadvantages. The hybrid framework is customized framework which uses the positive features of all the existing frameworks.

The hybrid framework is configurable. According to **Patwa (2009)**, the framework must configure in such a way that it should be reusable and scalable. The code should be maintainable. The hybrid framework can be configured in the folder structure to achieve these features.



**Figure 2.5: Hybrid framework (Pillai, 2011)**

A similar study by **Pillai (2011)** says about the folder structure of the hybrid framework. It explains that all components in the framework can organize separately. For e.g.- test cases, test data, test reports, test logs, libraries, object repository should organize separately.

**Mathew (2011)** supports this view in his research and claims that folder structure would help to maintain them separately. He also argues that once the framework needs to be updated, then it would be easy to update the concerned folder only.

### 2.3.9 Structure of hybrid framework

**Bhondkar et al. (2015)** propose a structure of hybrid framework in his research by separately defining its components. The main components are:

1. Reusable Library
2. Library Files
3. Test Data
4. Test Report
5. Test Screens AUT
6. Test Scripts

1. **Reusable Library:** It consists of reusable functions which can use again. These are

independent of application codes which are used to create a generic framework.

2. **Utility Files:** Utility files must store here. This folder contains all supporting various jar files. For ex: selenium-server.jar etc.

3. **Test Data:** Test data along with test objects must be kept here in the excel sheet as an input for the framework.

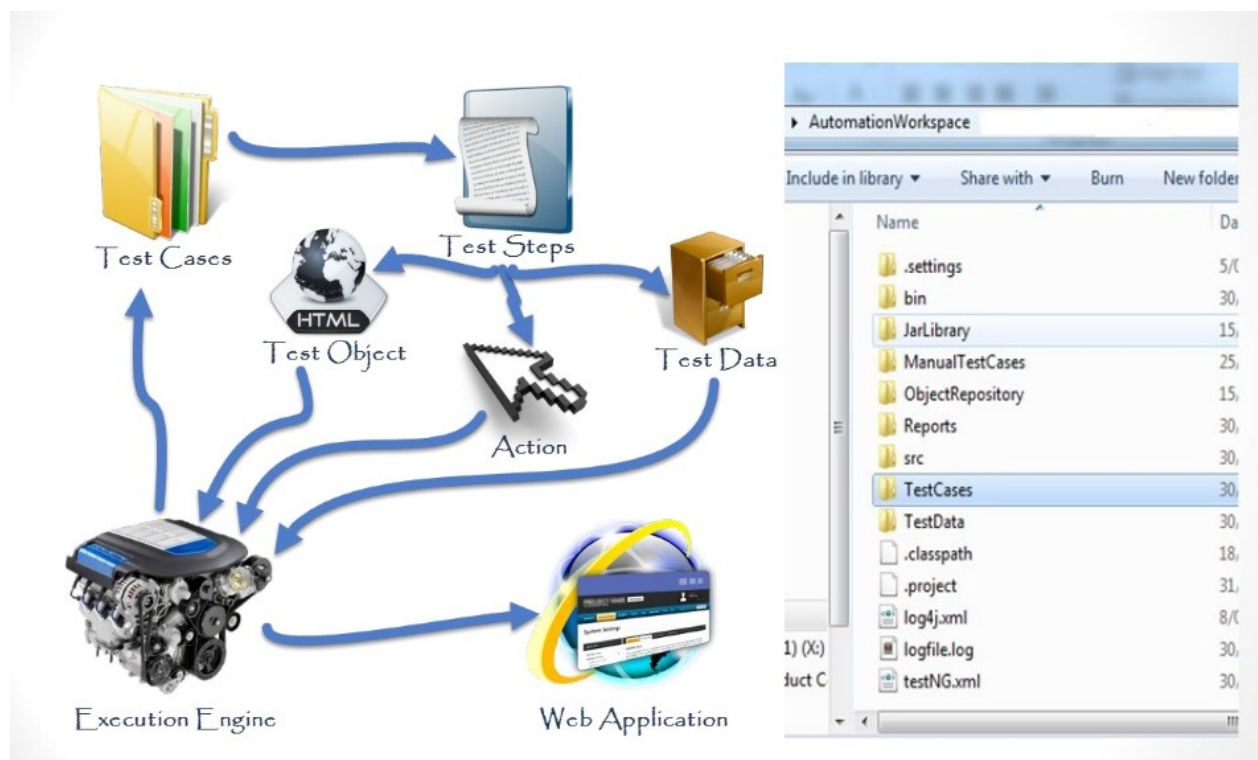
4. **Test Report:** The framework will generate a report for executed test cases. Log files will be generated with a timestamp of execution of test cases and also their Result Pass/Fail. It will

also create email ready HTML report.

5. **AUT:** This is an application on which our framework is working.

6. **Test Scripts:** This folder keeps all the test cases for test execution (Pillai, 2011).

Below figure displays the various component of the framework-



**Figure 2.6:** Structure of hybrid framework (Falcao, 2015)

### 2.3.10 Conclusion

In this section, the researcher has discussed the different frameworks and their pros & cons.

Since the hybrid framework is a combination of keyword driven and data-driven frameworks, hence it is having features of both the frameworks. This research tries to develop reusable hybrid framework as a proof of concept.

## Chapter 3: Research Methodology and Methods

### 3.1 Introduction

The literature review has discussed the basic issues which are essential for automation testing and developing a reusable automation framework. For e.g.- need for automation testing, selection of automation tool and identification of reusable framework. From the previous chapter, it has observed that there are limited academic resources available in the field of automation framework. Therefore this research can be perceived as an effort to fill this information gap. It discusses the assessments of research methodology which have used to collect and analyse data for the development of reusable hybrid automation framework.

This section justifies the qualitative approach of the research. Firstly, it discusses the research design, research philosophy, research approach and research strategy. Secondly, the identification and selection of sample data have discussed. Finally, the ethics and the limitations of the research has considered.

### 3.2 Research Methodology

According to **Purohit's (2012)** interpretation of **Blaikie's** work- Research methodology is defined as the technique to gain the information. It defines how theories are generated and tested. It creates the relationship between theoretical perspectives and research problem. As **Remenyi** suggest that- Methodology belongs to a procedure which helps to conduct the research. The purpose of the research methodology is to assist the researcher in answering the research question by collecting relevant information about the researcher topic (**Purohit, 2012**).

**Hurley (2009)** states that there are many ways researcher can conduct their research. However, the quality of the research depends on completely the method chosen during the study. **Hurley's (2009)** interpretation of **Saunders's** research onion is that the research process is iterative and interactive in nature. Hence the research onion offers a structure for the research process which is not a step by step process. As the research develops and the information changes, the research also develops and changes. Therefore, learning adopted from an inner layer of the onion may influence the researcher's attitude to an outer layer.

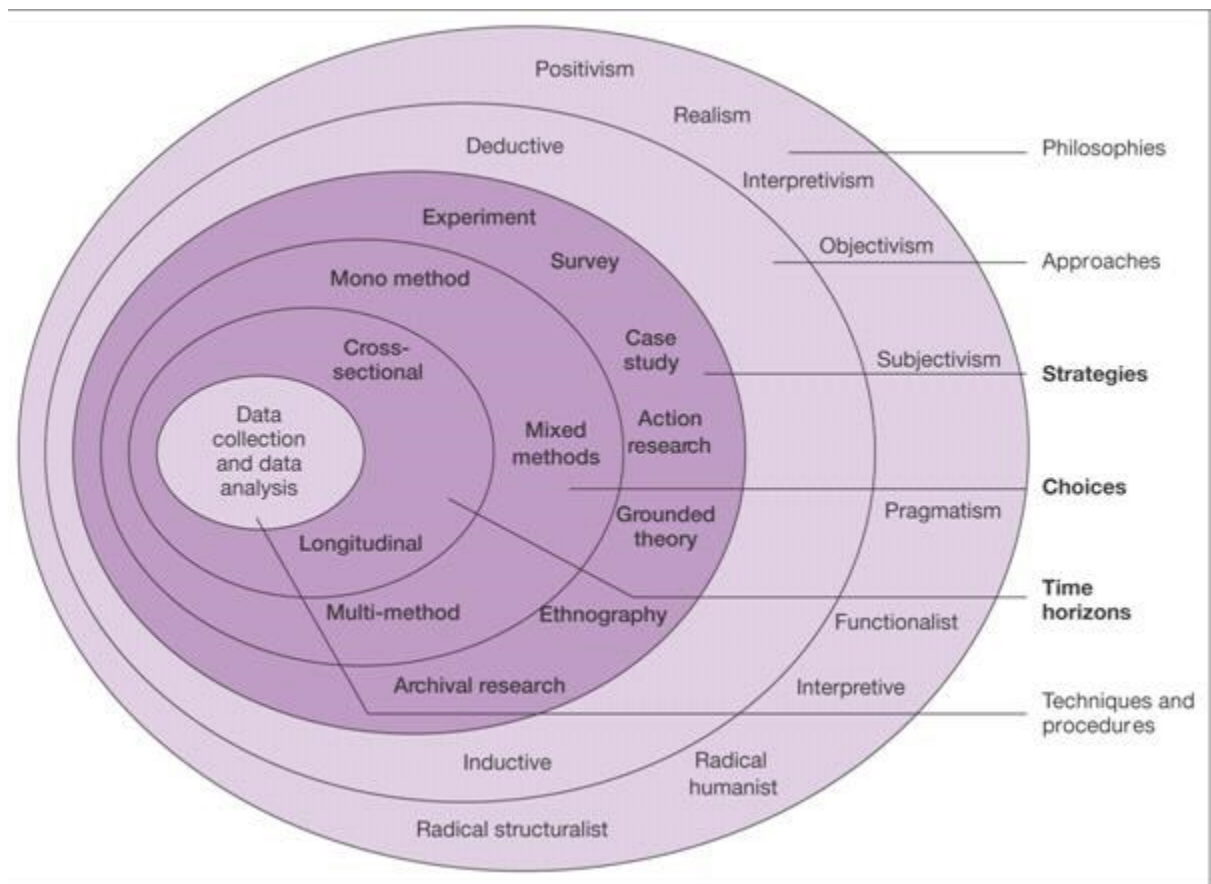


Figure 3.1: Research Onion (Based on Saunder’s Research Onion, 2013)

The research methodology represents the path to find the answer to the research question and guide in the right direction (Meller, 2013).

### 3.2.1 Research Philosophy

The research philosophy of this study is based on interpretivism. It relies on participant and their observation. **Saunders et al. (2009)** define- “Interpretivism is an entomology that advocates its importance for the researcher to understand the differences between humans in the role of social actors”. Interpretivism is defined as the ‘the recognition that subjective meaning plays a crucial role in the social actions’.

Interpretivism plays a significant role in qualitative research. The interpretive approach depends on the real approach of information gathering. E.g.- interviews and observations. This research questions emphasis on understanding the automation framework and the way stakeholders understood this. The data gathered during this research philosophy cannot be generalised, since data is influenced from personal viewpoints (**WHAT IS INTERPRETIVE RESEARCH, 2009**).

### 3.2.2 Research Approach

This research approach is inductive. An inductive approach related to qualitative research where researcher first collects the data and then cultivate the theory based on the result of the data analysis (**Saunders et al., 2009**). Since this approach is a concern with the context of subject, hence small sample of the subject is sufficient for the inductive approach. This research is following bottom up approach. In this approach, the researcher first collected the information by observation or interviews then recognized the pattern. After that researcher developed some hypothesis. Finally, this research illustrated conclusion and crafted theory. The research questions are open ended with subjectivity. Below diagram displays the bottom up approach of this research.

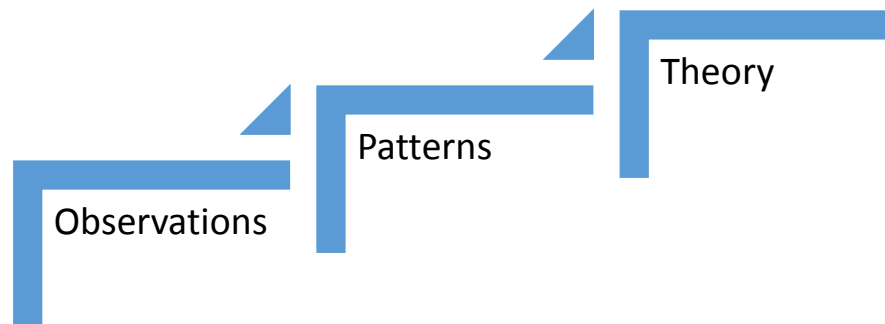


Figure 3.2: Research approach (**Induction approach, 2006**)

### 3.2.3 Research Strategy

**Saunders et al. (2009)** explain that the research strategy is conducted by the research questions and objectives. It is known as a crucial step in the research in which researcher chooses for collecting the data. There are many research strategies available. Like- survey, case study, focus group, interviews. No research strategy is better than others. It is credible only when if it helps the researcher to find the answer to the research questions.

For the research purpose, the researcher adopted the strategy of using the questionnaire as the data collection tool. The questionnaire was designed to focus on fact findings from the focused group. 15 questions had framed in such a way to get more and more information from the respondents. For data collection purpose, a questionnaire sent to the focused group in the form of a survey. Information was also collected through direct interview & online chatting (Google hangout).

### 3.2.4 Time horizon

The period in which the research is carried out is known as the Time Horizon. **Sanders et al. (2009)** describe that if the research is done as a 'snapshot' taken at a particular time, then it is called cross-sectional research. This research is cross-sectional research. It is based on the survey strategy for data collection.

### 3.2.5 Sampling

As **Meller (2013)** interprets **Williamson's** work in his research that sampling is the process of selection of some respondents from a specific group for the purpose of data collection to obtain the general view of large population. It is not possible to get the opinion from the entire population due to time constraint and limited resources.

Therefore, the research has to focus on a specific group to draw conclusions.

Professionals from IT companies had been chosen as a sample for this research.

Professionals had experience in software development, quality assurance or some other IT related work. These focus group had been approached based on personal contact.

Email, chat, and phone are used to approach the respondents. Research targeted 21 respondents with professional experience in their relevant field.

### 3.2.6 Data Collection

The primary research conducted in this qualitative research needs that a data collection method in the restrictions of a small sample and cross-sectional time horizon. There are several qualitative data collection methods are available. For e.g. observation, interviews or open-ended questionnaires **Maxwell (2005, p.80)**.

**Maxwell (2005)** explains that the inductive approach to the research demanded an in-depth understanding of the automation framework. The researcher believes that interviews and survey questionnaire solves this purpose.

### 3.2.7 Data Collection Instruments

This research has chosen survey questionnaire and interview as a data collection instrument. As defined by **Hair et al. (2003)**, "a questionnaire is a predetermined set of question designed to capture data from respondents". Survey questionnaire had sent to the focused group through email. 17 respondents submitted their responses by answering the questionnaire. The interview was also carried out to gather the

information about automation framework. The researcher conducted two interviews face to face and three interviews by using electronic medium Google hangout.

### 3.2.8 Questionnaire

The survey questionnaire was prepared for this research with 15 open-ended questions. All the questions were framed in a manner to collect information as much as possible from the focused group. The researcher organized the questions in such a way that all the questions were linked with each other. Some questions were used in the interview to collect data. The questionnaire had two parts. In the first part, the researcher tried to understand the background of the respondents. In the second part, questions were asked related to the research question.

#### **First part of Questionnaire**

- A. You are a software professional with experience in
  - a. Quality Assurance
  - b. Software Development
  - c. Others
- B. You are having experience of
  - a. **0-3 years**
  - b. **4-6 years**
  - c. **7-10 years**
  - d. **10+ years**
- C. You are currently working in ..... (optional)

#### **Second part of Questionnaire**

1. Have you ever involved in software testing activity?
2. What do you prefer- Manual testing or automation testing?
3. If you prefer automation testing then what are the reasons to select automation over manual?
4. Have you come across any automation tool? Please specify the name.
5. What features do you expect from an automation tool?
6. Do you believe that automation testing saves testing efforts and time? If yes then how?
7. Which automation framework you use for testing?

8. What type of features you expect from an automation framework?
9. Automation framework which you use is application-specific or generic? How?
10. Which model of Automation framework you use- keyword driven, data driven or hybrid? Why?
11. What are the benefits of Hybrid model framework over other models?
12. What are the other events your framework capable of handling apart from test execution?
13. Have your framework designed in such a way that user can easily run the tests and monitor the status without any programming skills? Briefly, explain.
14. How your framework has designed to add and maintain new features easily?
15. Is your framework design compatible with other tools like TestNG, Maven, SVN, GitHub?

Responses collected from the focused group has been discussed in next chapter.

### 3.2.9 Data Analysis Procedure

Open-ended questions provide detailed information which cannot be captured through closed end questions. Techniques have been developed to analyse qualitative data which are in the form of open-ended questions. During analysis, the researcher observes patterns and trends in the responses to reach definite conclusions **(Brief guide to open-ended survey questions, n.d.)**.

The researcher has attempted several steps for the analysis of open-ended responses.

1. The researcher studied the responses multiple times.
  2. Developed coding after understanding the themes.
  3. Once coding category defined, researcher used excel sheet and provided number of responses against each category
  4. After observing the excels sheet, researcher tried to find some patterns
  5. Once major patterns have been identified, researcher analysed the response
  6. Summarized what researcher has found in form of charts or graph in excel sheet
- (Qualitative analysis strategies for Analysing open-ended survey questions in MAXQDA, n.d.)**

Data analysis has been discussed in details in chapter 6.

### 3.2.10 Research Ethics

**Saunders et al. (2009)** demonstrate that research ethics means ensuring the design of your research is fair and morally justifiable to all those involved. The ethics of this research is based on- objectivity, no harm, confidentiality, and integrity.

Objectivity- The researcher has avoided any self-deception during the research.

No harm- This study ensures that no harm involved to any participants who have participated in this research for discussion purpose.

Confidentiality: The Participants who took part in this research can expect privacy from the researcher about their personal details.

Integrity: This research has carried out without any data distortion. The integrity of data has maintained.

### 3.2.11 Limitations of Methodology

According to **Meller's (2013)** interpretation of Bryman and Bell's work - the research findings from the selected focused group can be generalized only to the group from which the sample was taken. Hence, this is a natural limitation to any research and is not particular to the methodology used. Therefore, a large-scale research would be more representative and would also improve the quality of the research.

**Meller (2013)** says in his research that the level of bias is another limitation during the process. The researcher has taken precaution during data collection process.

Despite these limitations, this research provides valuable contributions for reusable automation framework.

## Chapter 4: Artefacts Design and Development

### 4.1 Introduction

In this chapter, the Researcher describes our overall approach and the design details of our test automation framework using Selenium tool. This tool automatically runs the test cases by taking data from input files and generate test report by using a framework on the top of Selenium Web Driver. The programming language used for developing this framework is Java.

First, the researcher gave an overview of the approach as per the software development life cycle. It has described the software development model first and after that, it described the different phases of the SDLC model. There are four sections in this chapter. The first section starts with requirement gathering; the second section explains designing, the third section shows implementation and fourth section explains test report after execution of tests.

### 4.1 Selection of software development model

“A software process is a set of related activities that leads to the production of a software product” (**Sommerville, p.28, 2000**). The software development model is a structured process for the construction of a software product. The main aim of this model is to improve the quality of the product (**Introduction to Software Engineering, p. 19-21**).

In any software model, the primary goal is to solve the problem. This research followed number of steps like-

1. Understanding the problem
2. Identifying a solution
3. Coding
4. Testing (**Sabale and Dani, p. 21-22, 2012**)

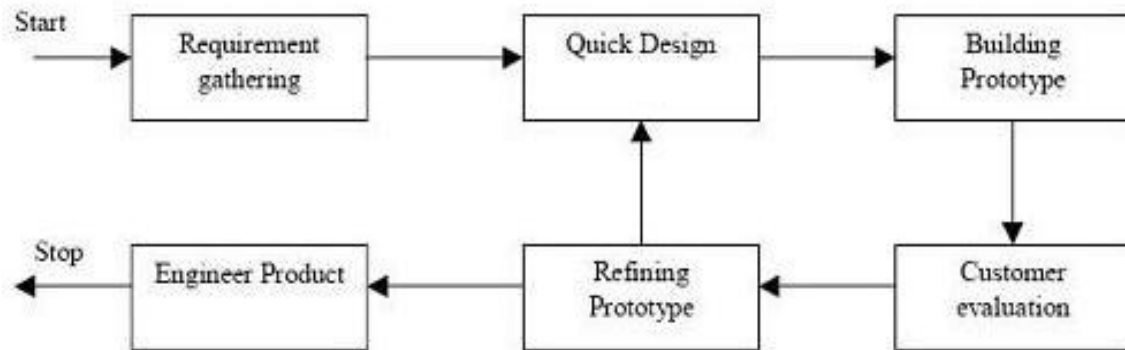
For the development of a hybrid framework, the prototype process model was selected. As **Walker (2007)** states that prototyping is a development methodology in which a model or prototype is first developed by using the preliminary ideas of the requirement and deliver a throwaway to the end user for their evaluation and feedback.

The basic idea was to build a prototype according to the requirements. This model was first developed based on the known requirements. During this model, the end user gets an idea of the real system. It helps end users to understand the requirements of the desired system better. The prototype is updated in a continuous fashion until the system user wants to be

produced. This development methodology is also sometimes known as incremental development (**Sabale and Dani, p. 21-22, 2012**).

Experiments have shown that the prototype software model reduces the issues with the requirements specification and it also lowers the overall development costs (**Boehm et al. 1984**).

**Diagram of prototype model:**



*Prototyping Model*

Figure 4.1 (what is prototype-model advantages disadvantages and when to use it, n.d.)

#### 4.1.1 Advantages of the Prototype Process Model

End users participate actively in the development process. The user gets a better understanding of the system by the throwaway prototype. It gives a fair knowledge of the product which is being developed. It also helps to detect the errors at an earlier stage. Frequent user feedbacks contribute to developing a better and quality product. In case any functionality is missing then it can be identified easily at the right time. It also identifies the unwanted features which are not required. Requirement and its validation go together (**Walker, 2007**).

#### 4.1.2 Disadvantages of Prototype Model

In general, prototype method increases the difficulty level of the product which is sometimes out of scope to fix. Research also says that if there is confusion between user and developer, then it is identified as the prototypes has demonstrated which leads to complexity at the end of development (**Walker, 2007**).

#### 4.2 Requirement Analysis

The aim of this section is to define functional and non-function requirements for the development of automation framework. It also discusses the different approaches to

identify the requirements. A framework is a set of rules that help to automate the web applications. Identifying those rules are part of the requirement (Laukkanen, 2006).

Sommerville (2000) states that “the requirements for a system are the descriptions of what the system should do.” These requirements show the need of customers to find the solution of some problems. The whole process of requirement gathering, analysis, documentation, finding constraints and feasibility is known as requirements engineering (RE).

Requirements are often mentioned as high-level statements. It is also called as user stories in Agile development methodology. The requirement term is not common across different software companies (Sommerville, p. 83, 2000).

Research says that automation frameworks are having two type of requirements- Functional and Non-Functional. Functional requirement handles the automatic test execution part which is considered to be a high-level requirement. Non-functional requirements are related to maintainability and ease of use (Laukkanen, 2006).

#### 4.2.1 High-level requirements for a framework

Automatic test execution	Test cases must run automatic in the framework. It should be able to handle exceptions and must provide test reports.
Easy to use	The user must be able to use the framework easily. It must design in such a way that user should not require any programming skills to execute and monitor the test cases (Kit, 1999).
Maintainable	The framework must be easily maintainable whenever new feature. Test data and test script should be easy to update (Fewster and Graham, 1999).

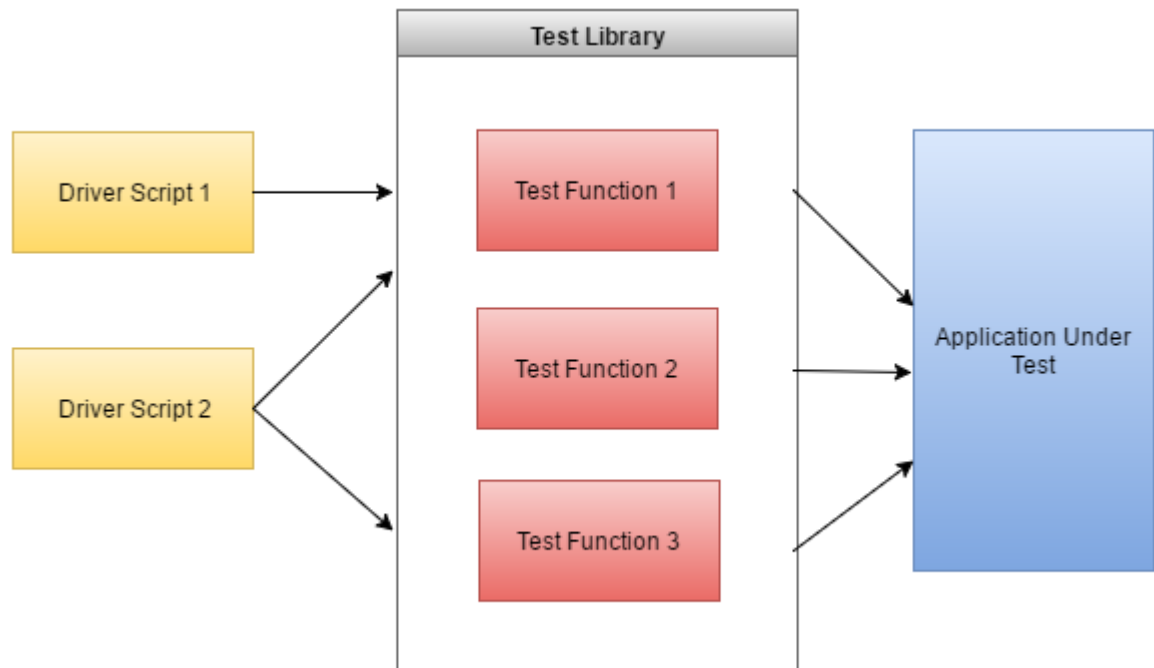
**Table 4.1:** High-level requirement for a framework

#### 4.2.2 Breakdown of functional requirements

1. Smooth test execution: Framework must be able to check all pre-conditions and once start execution then it should run the tests smoothly by its own (Fewster and Graham, 1999).
2. Error handling: Pettichord (1999) says that framework must handle errors and continue testing without manual intervention.

3. Verify test results: Verifying test output is crucial. **Fewster and Graham (1999)** define verification process as a comparison between actual and expected output. If it matches, then the test would be called as pass.
4. Assigning test status: once a test executed and its result validated then it must be given a status. According to **Laukkanen's (2006)** interpretation of Fewster and Graham (1999) work- if executed test's outcome matches with expected result then the test is a pass. If it does not match, then the test is a fail. For failed status, it can give details about the cause of the problem.  
  
Fewster and Graham (1999), propose having more statuses than pass and fail. For e.g.- Error, warnings, failure, info, with a meaningful message.
5. Detailed logging: Test automation framework should log each and every event during test execution. It should provide enough information that developers can debug it in the case of test failure.  
  
The main problem with logging, as stated by **Pettichord (2002)**, is how much to log. The log must have relevant information to identify the cause. It should neither short nor huge.
6. Test reporting: As per **Laukkanen's (2006)** interpretation adapted from **Buwalda et al. (2002)** - After successful test completion, test report must generate with below information:
  - version details of the system
  - total number of tests executed
  - number of tests passed
  - number of the test failed
  - number of error found
7. Email notification: Framework must be designed in such a way that it must notify the user through email in case of test completion or any failure. So that, the user can take further action (**Bhondokar et al., 2015**).
8. Modularity: Modularity is also one of the core requirement of any automation framework. In Modular framework, each module has kept separately and independently. It makes it easy to maintain (**Gulia, 2010**).  
  
In his research **Laukkanen (2006)** states that in small applications, it is easy to keep all functions in the same file as the main test script. However, in large applications,

functions and libraries are placed separately from the test script. Since libraries do most of the work, hence test scripts can be short. Since scripts only driving the test execution, hence it is known as driver scripts.



**Figure 4.2:** Driver scripts interacting with Application under test using libraries

### 4.3 Design

This section discusses the architecture of the framework. In this section, the researcher has described the design details of our proof-of-concept hybrid automation framework.

Architectural design is basically a structure of the framework which shows how it is going to work. It is the initial phase of the software design process. It is the basic connection between design and requirement as it identifies the different component of the system and the connections between them. The output determines how the system is organized and how it is communicating with other components (**Sommerville, 2000**).

The architecture of any software system is essential as it determines the performance, robustness and maintainability of a system (**Bosch, 2000**). As **Bosch** discusses, individual components of the system implement the functional requirements of the system. Also, the non-functional requirements are depending on the way the system architecture is designed.

As per **Sommerville** interpretation adapted from (**Booch, 2009**) that system architectures are mostly demonstrated by block diagrams. Each box represents a component. Boxes within boxes are known as sub-components. Arrow indicated the data flow from one component to another component. Block diagrams are easy to understand, and it represents a high-level design of the system architecture. People can easily understand these block diagram representations during design process even though they come from different discipline. Block diagrams are the perfect way of demonstrating the system as it is a better choice of engaging communication between the people.

**Laukkanen (2006)** says in his research that the structure of automation frameworks like hybrid framework can be broadly divided into three components- test design system, monitor test system & test execution system.

**Test design system**: It is used for developing new test scripts as well as updating the existing ones. It keeps test cases for application under test. Test cases, object repository, and test data are its sub-components. Test data has provided in the form of Excel sheet as an input to the Hybrid framework.

**Test monitor system**: The framework generates test report for test case execution. It generates log files for every pass or fails execution. It also sends an email report to the user. The test report, Log files, screenshot and email notifications are its sub-components.

**Test execution system**: It is the main part of the framework. It contains-

**Reusable library**: These are common functions files that can be used n number of times. This folder contains files and code which are independent of the application. To achieve a generic framework, most of the reusable files and codes should be kept there. This separate folder makes the framework more reliable and can be used for different other applications.

**Library files**: This folder contains all the files which are required for test automation. It consists of various jar files which are supported by Selenium web driver. For ex: selenium-server.jar, selenium-java-clientdriver.jar.

**Utility files**: Utility folders consists of small programs which help to increase the capabilities provided by the automation framework. These utility files can be added separately. For e.g.- mail monitoring utility, connection database utility (**Bhondokar et al., 2015**).

Hybrid framework architecture has shown in figure 4.3.

**Technology overview**:

Java: The programming language used for developing this automation framework is Java. Java is a high-level object-oriented programming language. It supports multiple platforms. Java is fast, reliable and secure language (What is Java and why do I need it, 2014).

Selenium: The automation tool which has been used for developing hybrid framework is selenium. It is an open source software testing tool. It uses Selenium web driver API which supports all web based applications with multiple browsers and multiple platforms (Selenium WebDriver – Selenium documentation, 2008).

Eclipse IDE: Eclipse is an open source IDE. It helps programmers /testers write code much faster and easier way than using a text editor (Guindon, 2016b).

TestNG: We have used TestNG for running the automation tests. TestNG is a testing framework having several new functionalities. Like- Annotations. It provides an adjustable test configuration method which makes easy for test execution (Welcome TestNG, 2004).

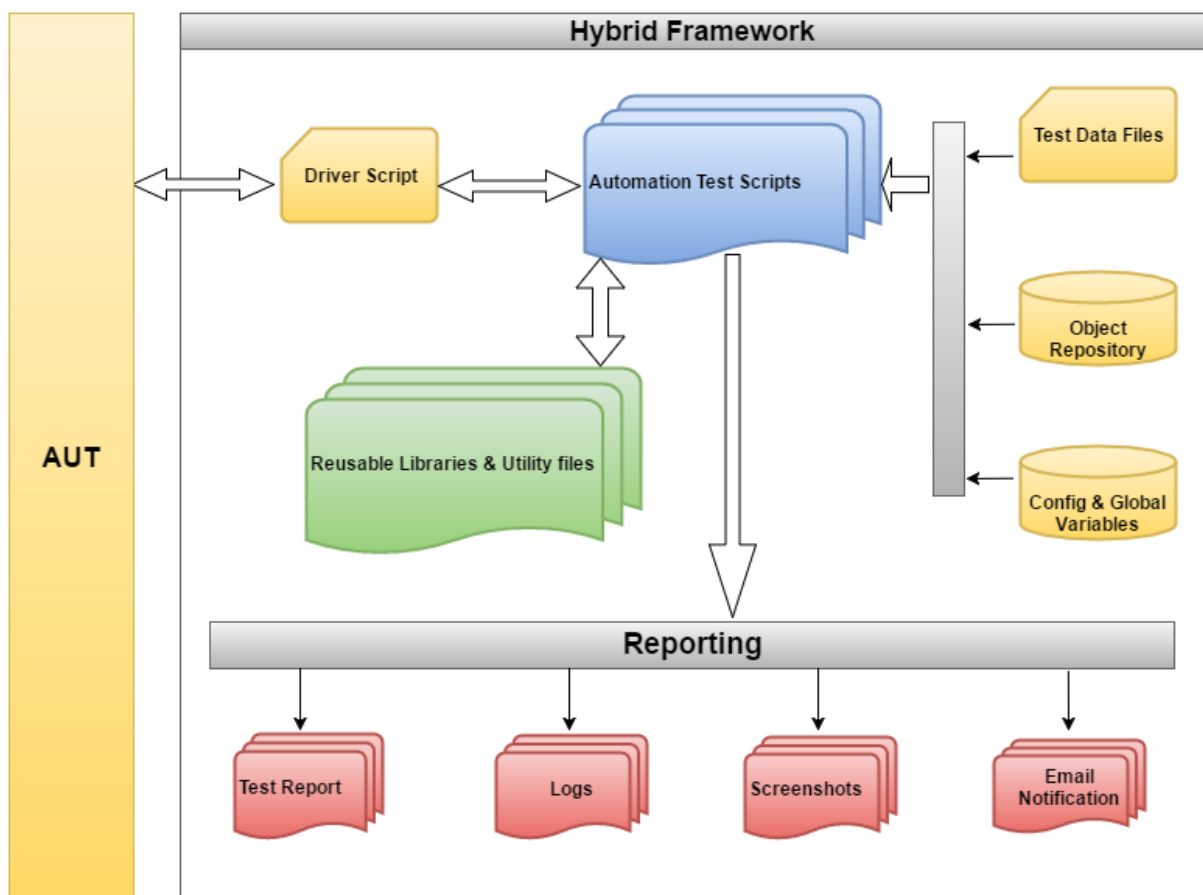


Figure 4.3: High-level architecture of Hybrid automation framework (Most popular test automation Frameworks with pros and cons of each – Selenium Tutorial, 2016)

Log4j: Log4j has been used in automation framework for generating logs. It is a configurable logging framework (APIs) written in Java and available under the Apache Software License. log4j is a popular logging package available in Java (**Apache log4j 1.2 - , 2012**).

## 4.4 Implementation

In this section, the researcher discussed the steps taken to implement Hybrid test automation framework. This study investigated how the different components have been used to test a web based application.

### Technical decisions

For test data: Researcher has used Microsoft Excel file format for storing and editing the test data.

For programming language: Java has been used for developing driver scripts and reusable libraries. Since it is high-level object-oriented programming, hence it is easy to implement.

For an application under test: Researcher has decided to use Gmail for automation testing. Gmail is a web-based application, and it is easy to use.

For reading writing from excel work book: In this study, the researcher has used Apache POI jar files to read & write different format of excel in selenium web driver.

For logging: Researcher has used log4j jar files. It helps to write the output log in files.

For reporting: For test reporting, TestNG has been used. TestNG is a testing framework which helps to organize the test and generate a test report.

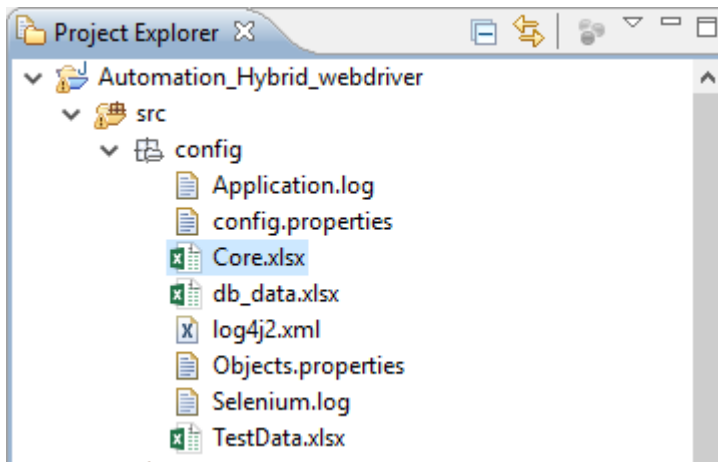
Framework approach: This study has implemented the automation framework of Hybrid type which is the combination of data-driven & keyword driven approaches.

### 4.4.1 Hybrid automation framework development

#### **Step1 Test data creation**

Test data used in the framework has created inside config package. Config package contains test case, test data, object repository and properties file. Apart from above files, it also has Application.log & Selenium.log files.

1. Structure of config package has shown in figure 4.4



**Figure 4.4:** Structure of config package

2. Test data file has shown in figure 4.5

	A	B	C	D	E	F
1	signin	passwr	mailid	sub	msg	password
2	<a href="mailto:shravandissertation@gmail.com">shravandissertation@gmail.com</a>	Soamin12	<a href="mailto:priyankasingh125@gmail.com">priyankasingh125@gmail.com</a>	Hello	This is a test mail	happytesting
3						
4						

**Figure 4.5:** Test data file

3. Test case suite has shown in figure 4.6

1	TCID	Description	Runmode
2	Navigate	Check URL	Y
3	LoginTest	LoginTest	Y
4			

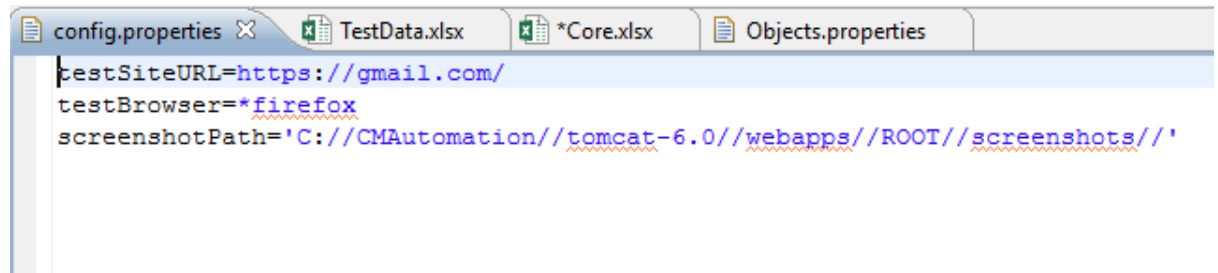
**Figure 4.6:** Test case suite

4. Test scripts for Login Test have demonstrated in figure 4.7

	A	B	C	D	E	F	G
1	TSID	Decription	Keyword	Object	ProceeedOnFail	Data_Column_Name	
2	TS01	Wait For 10 sec	waitfor	email	N		
3	TS02	Enter email Id	input	email	N	signin	
4	TS03	Click on Next	clickLink	next	N		
5	TS04	Enter Password	input	passwr	N	passwr	
6	TS05	Click on Signin	clickLink	signin	N		
7	TS04	Validate Password	getText	erroremail	N		
8	TS06	Navigate to the site	navigate	testSiteURL	N		
9	TS07	Enter email Id	input	email	N	signin	
10	TS08	Validate EmailId	getText	erroremail	N		
11	TS08	Click on Next	clickLink	next	N		
12	TS09	Enter Password	input	passwr	N	password	
13	TS010	Click on Signin	clickLink	signin	N		
14	TS01	Wait For 10 sec	waitfor	compose	N		
15	TS021	Wait For 10 sec	waitfor	logout	N		
16	TS022	Click on logout	clickLink	logout	N		
17	TS023	Click on signout	clickLink	signout	N		
18							
19							
20							
21							
22							
23							

**Figure 4.7:** Test scripts for Login

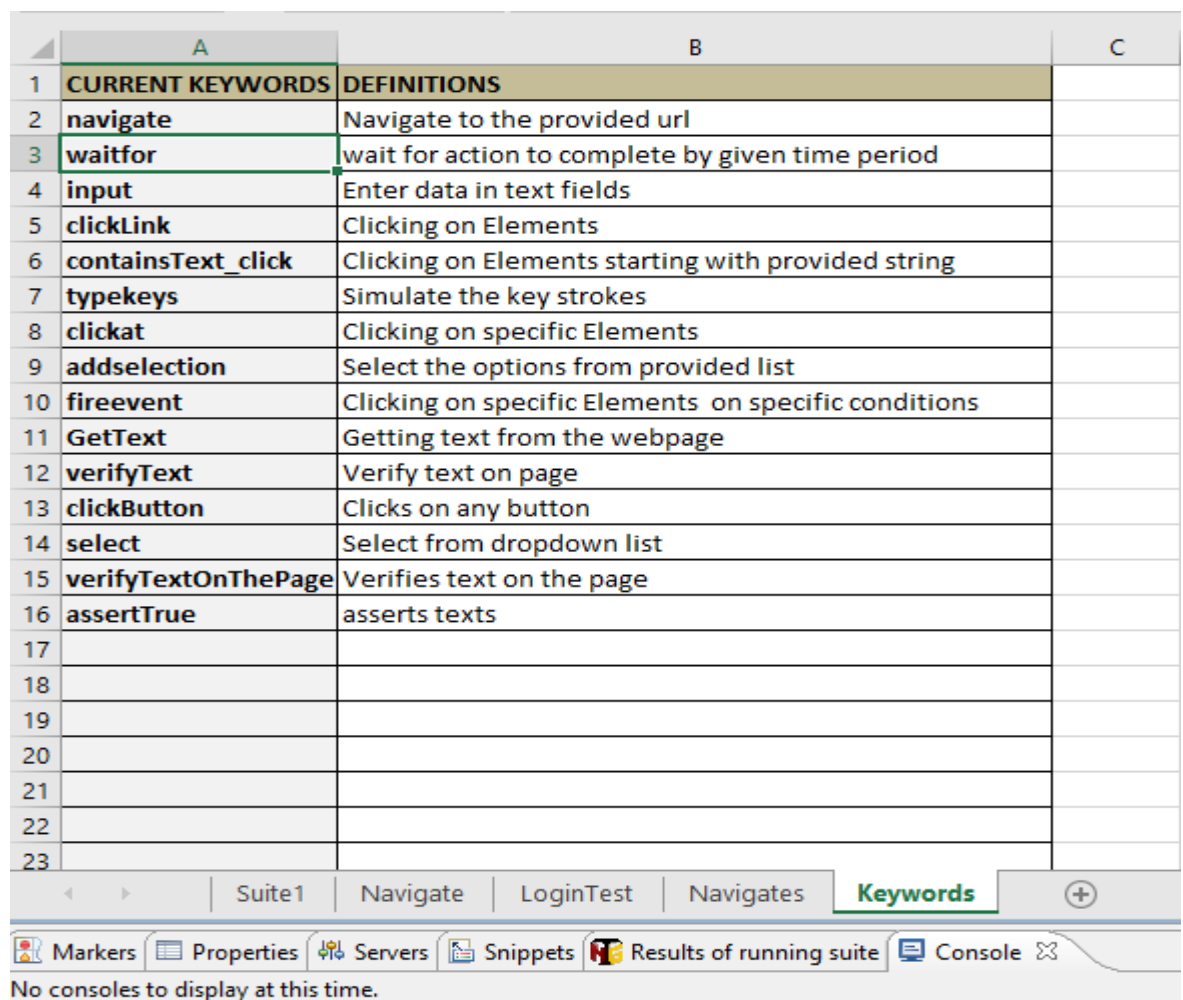
5. Config property file has shown in figure 4.8



```
config.properties X TestData.xlsx *Core.xlsx Objects.properties
testSiteURL=https://gmail.com/
testBrowser=*firefox
screenshotPath='C://CMAutomation//tomcat-6.0//webapps//ROOT//screenshots//'
```

**Figure 4.8:** Config property

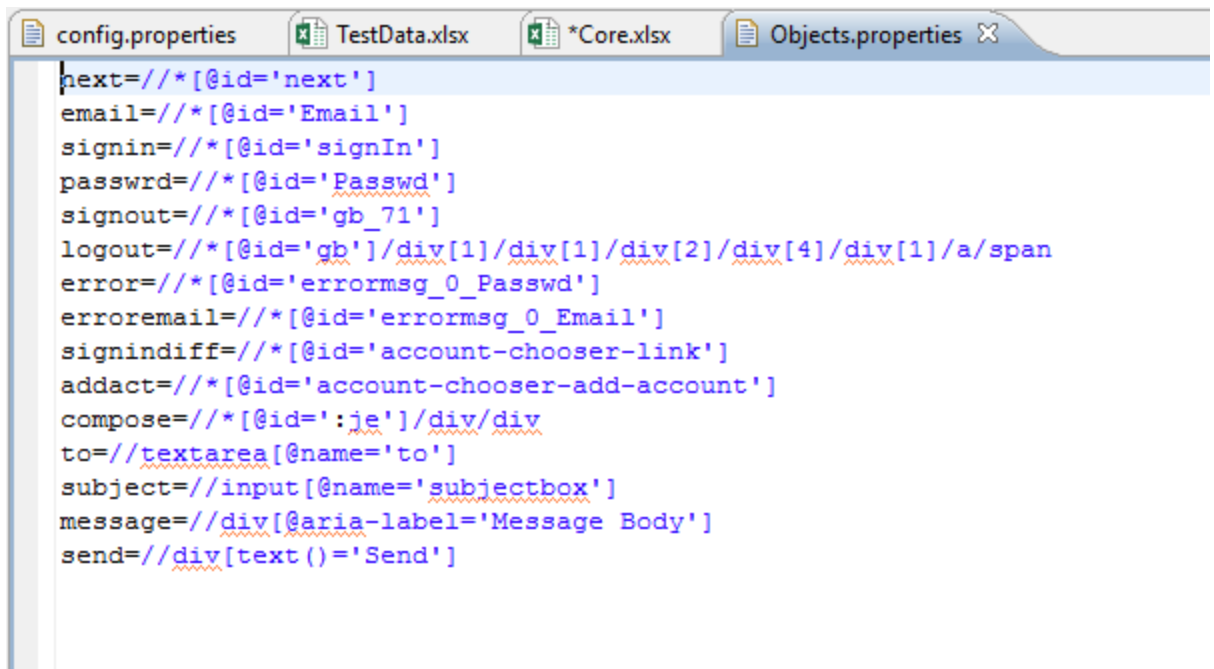
6. Keywords file for test data has shown in figure 4.9



	A	B	C
1	<b>CURRENT KEYWORDS</b>	<b>DEFINITIONS</b>	
2	navigate	Navigate to the provided url	
3	waitfor	wait for action to complete by given time period	
4	input	Enter data in text fields	
5	clickLink	Clicking on Elements	
6	containsText_click	Clicking on Elements starting with provided string	
7	typekeys	Simulate the key strokes	
8	clickat	Clicking on specific Elements	
9	addselection	Select the options from provided list	
10	fireevent	Clicking on specific Elements on specific conditions	
11	GetText	Getting text from the webpage	
12	verifyText	Verify text on page	
13	clickButton	Clicks on any button	
14	select	Select from dropdown list	
15	verifyTextOnThePage	Verifies text on the page	
16	assertTrue	asserts texts	
17			
18			
19			
20			
21			
22			
23			

**Figure 4.9:** Keyword file

7. Object repository file has shown in figure 4.10



```
next=//*[@id='next']
email=//*[@id='Email']
signin=//*[@id='signIn']
passwd=//*[@id='Passwd']
signout=//*[@id='gb_71']
logout=//*[@id='gb']/div[1]/div[1]/div[2]/div[4]/div[1]/a/span
error=//*[@id='errmsg_0_Passwd']
erroremail=//*[@id='errmsg_0_Email']
signindiff=//*[@id='account-chooser-link']
addact=//*[@id='account-chooser-add-account']
compose=//*[@id=':je']/div/div
to=//textarea[@name='to']
subject=//input[@name='subjectbox']
message=//div[@aria-label='Message Body']
send=//div[text()='Send']
```

**Figure 4.10:** Object repository

A core file is an excel file which contains test cases, test steps and keywords for test execution. Test data file holds the data related to the corresponding object which is mentioned in the core file. Config.property file holds the web address of the application under test and the browser details. Object repository file holds the XPath of all the corresponding objects of web application which is defined in the core file.

### **Step 2 Driver Scripts**

Driver scripts are the core component of the framework. Test app package contains DriverApp.java & KeywordApp.java scripts which read the test cases from input files by using Xlfile\_Reader.java function. The driver script which has implemented in this framework has shown in **Listing 4.1**.

Test app folder having driver scripts has shown in figure 4.11

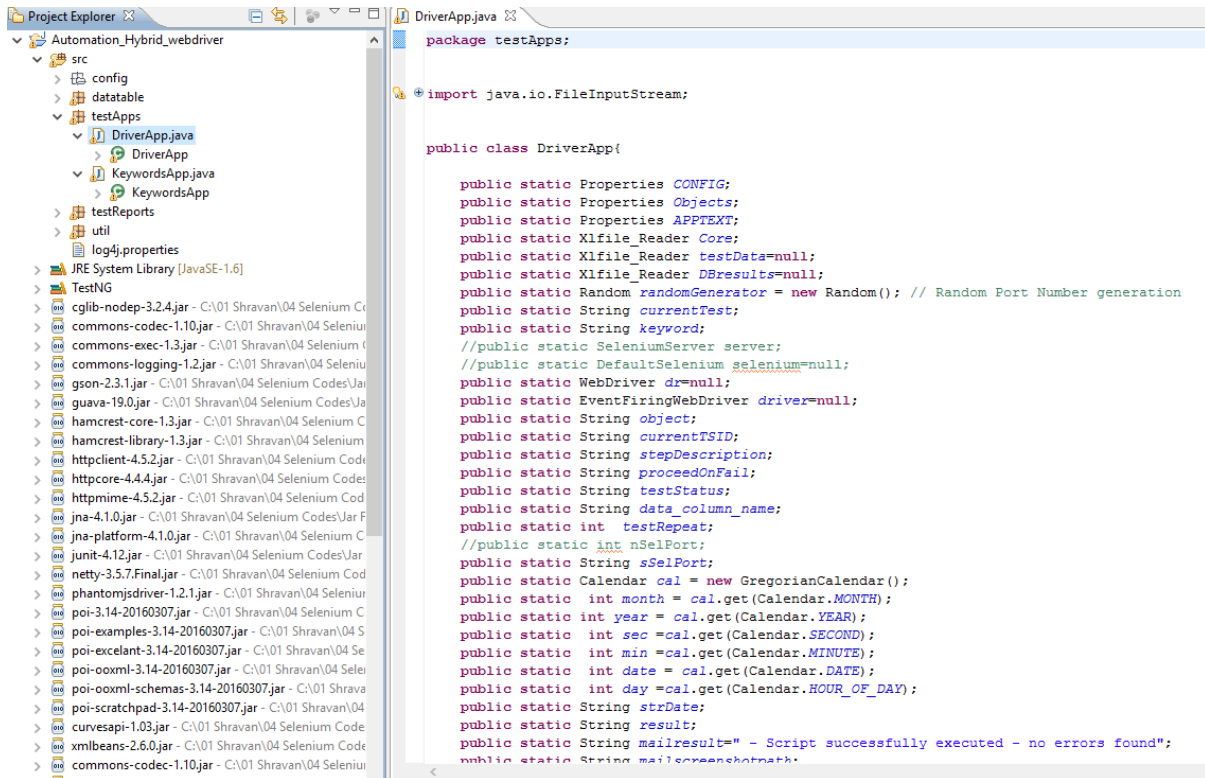
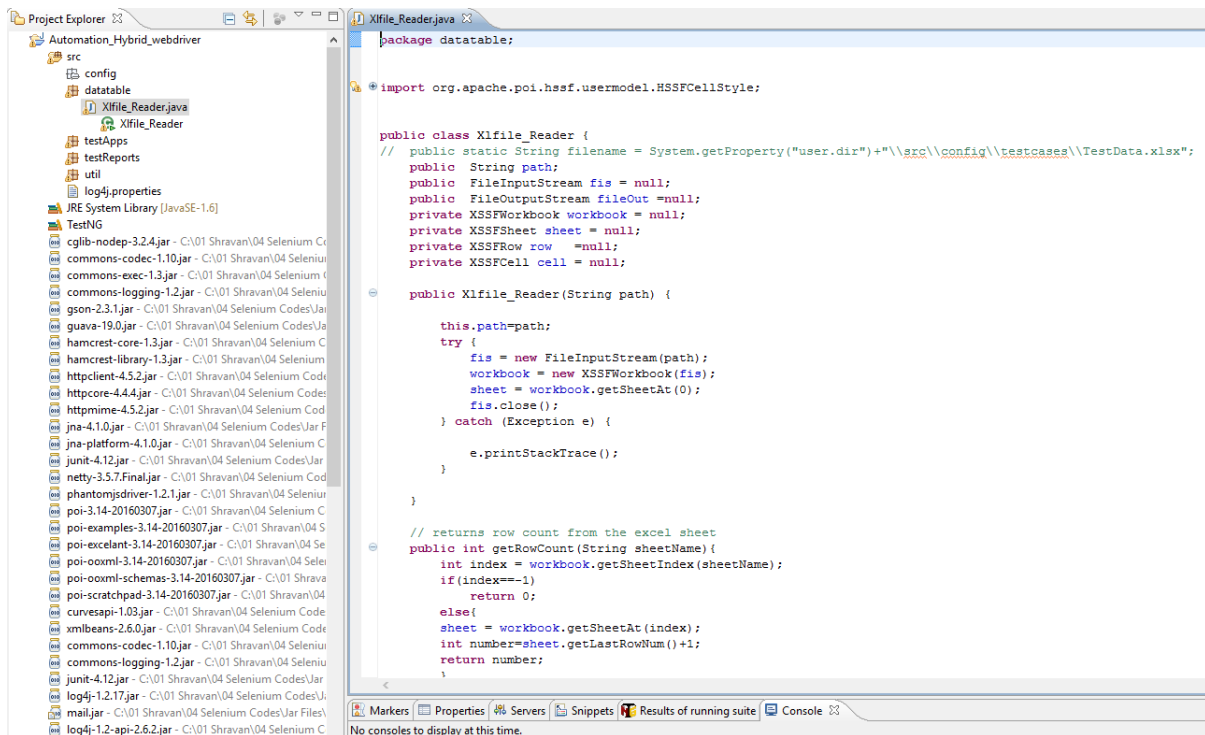


Figure 4.11: Test app folder

### Step 3 Excel Reader file

To read test data and test cases from excel sheet, the researcher have implemented XlFile\_Reder.java script. These scripts help to read input files.

Excel Reader file has shown in figure 4.12. Script has shown in Listing 4.2

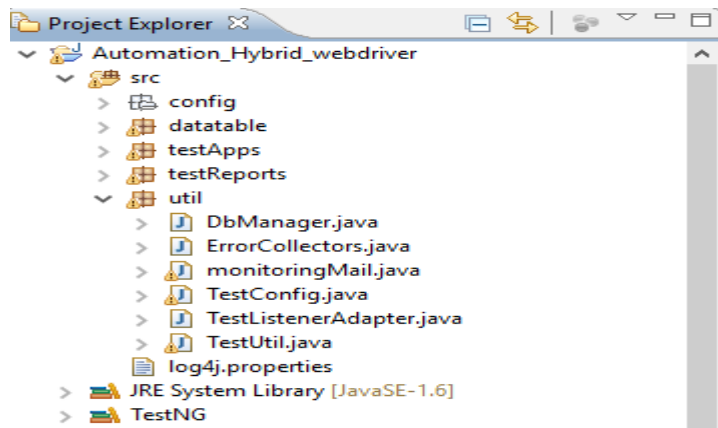


**Figure 4.12:** Excel Reader file

#### **Step 4 Utility files**

The researcher has implemented some reusable files to achieve reusability and maintainability. Utility package has been created with TestReports.java, monitoringMail.java, DbManager.java, TestConfig.java, TestListnerAdapter.java, TestUtil.java and ErrorCollectors.java libraries.

Utility files package has shown in figure 4.13



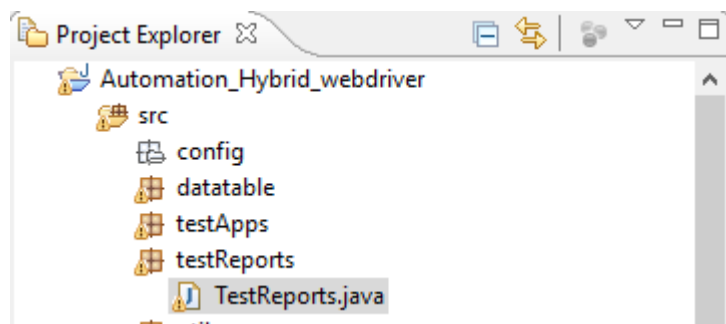
**Figure 4.13:** Utility files

MonitoringMail.java utility files help to send an email notification after completion of test execution. TestReports.java file helps to generate a report. DbManager.java file can be used to connect database. Data base connection is not part of this research. TestUtil.java file helps to run test cases smoothly. ErrorCollectors.java file helps to generate log during test execution.

#### **Step 5 Test reports**

TestReport.java scripts generate test report after successful completion of the test execution.

Test report package has shown in figure 4.14

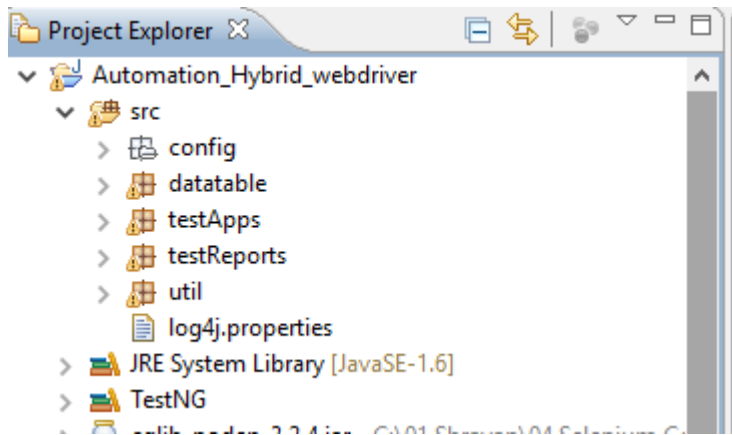


**Figure 4.14:** Test report package

### **Step 6 Integrated TestNG tool**

The researcher has also implemented the TestNG tool for creating the test report. TestNG helps to execute the code by using test annotations.

Figure 4.15 has shown the TestNG integrated into the hybrid framework.

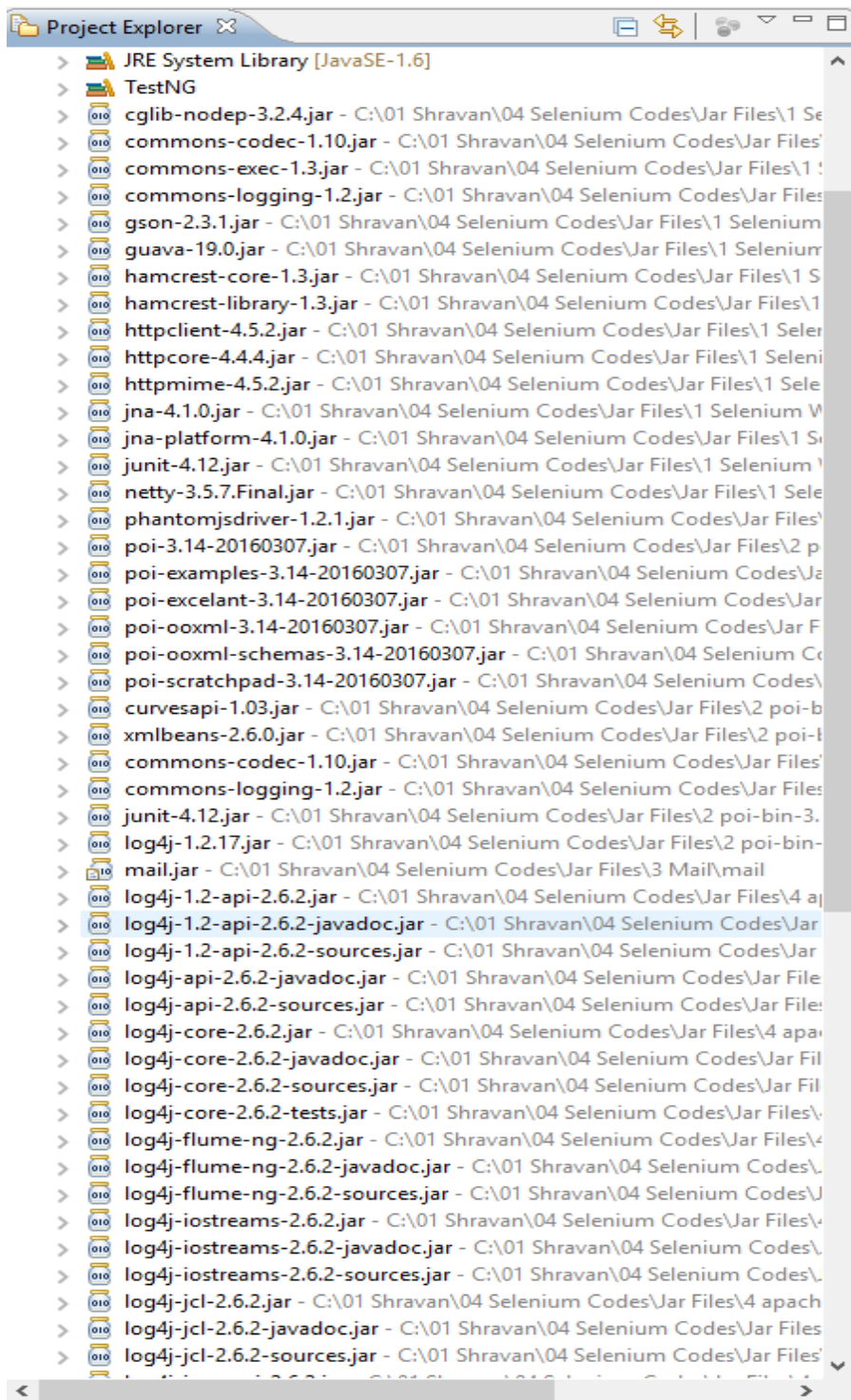


**Figure 4.15:** TestNG

### **Step 7 Many jar files have been used to support the framework**

The researcher has implemented many supporting jar files in the hybrid framework. These jar files help into smooth test execution.

Jar files have shown in figure 4.16.



**Figure 4.16:** Jar files

### **Step 8 Log4j files**

The researcher has implemented Log4j files in the automation framework. These files help to generate logs during test execution. It generated log in application.log file and Selenium.log files. Log4j file has shown in figure 4.17.

```

#Root logger option
log4j.rootLogger=debug,file
log4j.appender.file=org.apache.log4j.RollingFileAppender
log4j.appender.file.File=C:\\Users\\shrav\\workspace\\Test Project3\\Automation_Hybrid_webdriver\\src\\config\\Selenium.log
log4j.appender.file.maxFileSize=5000KB
log4j.appender.file.maxBackupIndex=1
log4j.appender.file.layout=org.apache.log4j.PatternLayout
log4j.appender.file.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n
#do not append the old file. Create a new log file everytime
log4j.appender.file.Append=false

#Application Logs
log4j.logger.devpinoyLogger=DEBUG, dest1
log4j.appender.dest1=org.apache.log4j.RollingFileAppender
log4j.appender.dest1.maxFileSize=5000KB
log4j.appender.dest1.maxBackupIndex=3
log4j.appender.dest1.layout=org.apache.log4j.PatternLayout
log4j.appender.dest1.layout.ConversionPattern=%d{dd/MM/yyyy HH:mm:ss} %c %m%n
log4j.appender.dest1.File=C:\\Users\\shrav\\workspace\\Test Project3\\Automation_Hybrid_webdriver\\src\\config\\Application.log
#do not append the old file. Create a new log file everytime
log4j.appender.dest1.Append=false
log4j.configurationFile=off

```

**Figure 4.17:** Log4j file

## 4.5 Test Execution

In this section, the researcher discussed the output after successful test execution.

Researcher executed the test scripts, and the framework generated the test report after completion of the test. Figure 4.18 has shown the test report generated at the given location.

e	index08-13-2016_13-02-01	8/13/2016 2:02 PM	HTML File	5 KB
e	index08-13-2016_12-59-36	8/13/2016 1:59 PM	HTML File	5 KB
e	index08-13-2016_12-57-05	8/13/2016 1:57 PM	HTML File	5 KB
e	index08-13-2016_12-54-30	8/13/2016 1:54 PM	HTML File	4 KB
e	index08-13-2016_12-53-21	8/13/2016 1:53 PM	HTML File	5 KB
e	index08-13-2016_12-51-53	8/13/2016 1:52 PM	HTML File	5 KB
e	index08-13-2016_12-50-56	8/13/2016 1:51 PM	HTML File	5 KB
e	index08-13-2016_12-49-42	8/13/2016 1:50 PM	HTML File	5 KB
e	index08-13-2016_12-47-43	8/13/2016 1:48 PM	HTML File	5 KB
e	index08-13-2016_12-46-47	8/13/2016 1:47 PM	HTML File	5 KB
e	index08-13-2016_12-45-34	8/13/2016 1:45 PM	HTML File	5 KB
e	index08-13-2016_12-44-26	8/13/2016 1:44 PM	HTML File	5 KB
e	index08-13-2016_12-43-17	8/13/2016 1:43 PM	HTML File	5 KB
e	index08-13-2016_12-42-01	8/13/2016 1:42 PM	HTML File	4 KB
e	index08-13-2016_11-27-14	8/13/2016 12:27 PM	HTML File	4 KB
e	index08-13-2016_00-10-16	8/13/2016 1:10 AM	HTML File	4 KB

**Figure 4.18:** Test report folder

Test report produced by automation framework has shown in figure 4.19.

## Automation Test Results

**Test Details :**

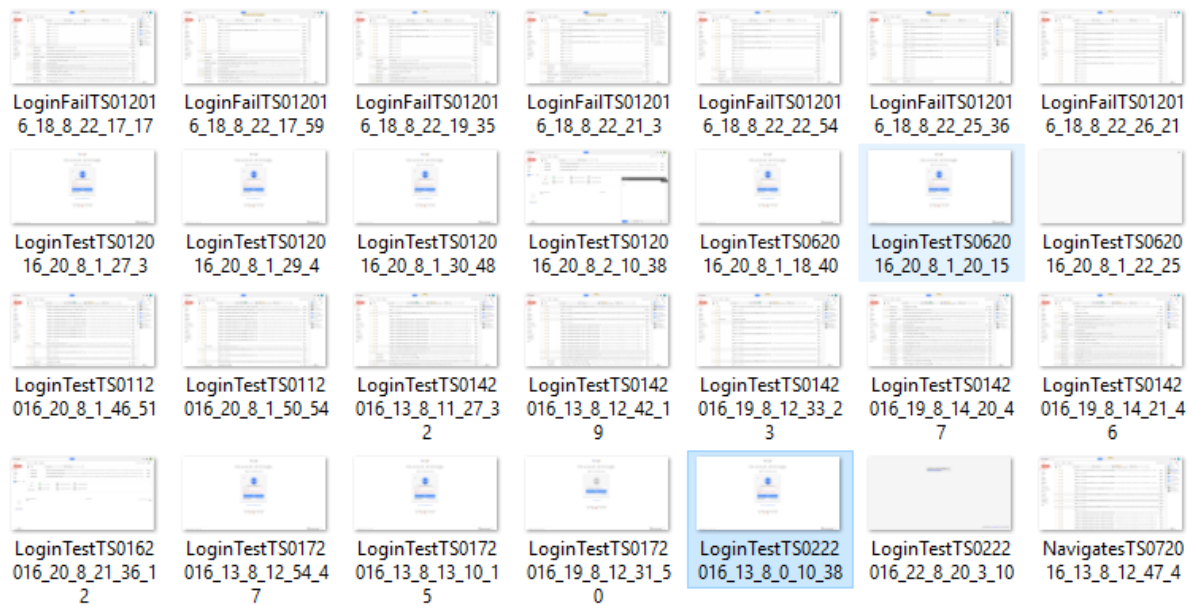
Run Date	13.August.2016
Run StartTime	13.August.2016 12.10.16 AM
Run EndTime	13.August.2016 12.10.38 AM
Environment	Monitor Prod
Release	3.1

**Suite 1 Report :**

Test Script#	Test Case Name	Status	Run Start Time	Run End Time
1	<a href="#">Navigate</a>	Pass	13.August.2016 12.10.22 AM	13.August.2016 12.10.23 AM
2	<a href="#">LoginTest</a>	Fail	13.August.2016 12.10.23 AM	13.August.2016 12.10.38 AM

**Figure 4.19: Test report**

In case of test failure, screen shot captured by framework has shown in figure 4.20 and 4.21

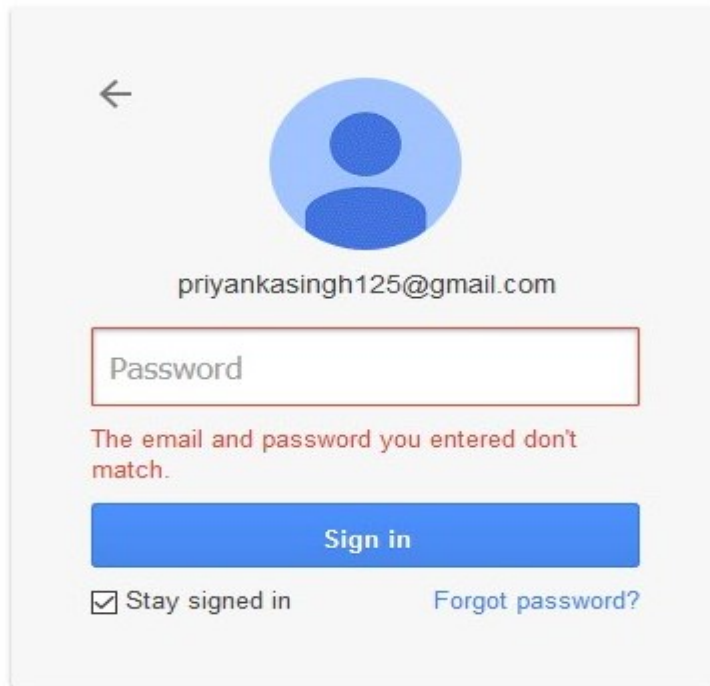


**Figure 4.20: Screenshot path**



# One account. All of Google.

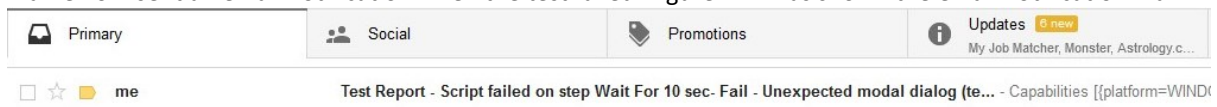
Sign in to continue to Gmail



[Sign in with a different account](#)

**Figure 4.21:** Screenshot for error

Framework sent an email notification when the test failed. Figure 4.22 has shown the email notification mail.



**Figure 4.22:** Email notification

Log4j generated logs for application as well as Selenium. Both logs have saved into the application.log file and Selenium.log files. Figure 4.23 and 4.24 have shown the logs file.

```
Application.log X
11/08/2016 01:30:07 devpinoyLogger Executing the test Navigate
11/08/2016 01:30:07 devpinoyLogger Executing Navigate
11/08/2016 01:30:08 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:08 devpinoyLogger Error came
11/08/2016 01:30:08 devpinoyLogger #####Navigate --- Pass
11/08/2016 01:30:08 devpinoyLogger Executing the test LoginTest
11/08/2016 01:30:08 devpinoyLogger Executing wait Keyword
11/08/2016 01:30:09 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:09 devpinoyLogger Executing input Keyword
11/08/2016 01:30:09 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:09 devpinoyLogger Executing clickLink
11/08/2016 01:30:10 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:10 devpinoyLogger Validating input Keyword
11/08/2016 01:30:10 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:10 devpinoyLogger Executing input Keyword
11/08/2016 01:30:11 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:11 devpinoyLogger Executing clickLink
11/08/2016 01:30:11 devpinoyLogger ***Result of execution -- Pass
11/08/2016 01:30:11 devpinoyLogger Validating input Keyword
11/08/2016 01:30:45 devpinoyLogger Error while validating input -errorUnable to l
Command duration or timeout: 33.23 seconds
```

Figure 4.23: Application Log

```
Selenium.log X
01:30:07,856 DEBUG devpinoyLogger:183 - Executing the test Navigate
01:30:07,859 DEBUG devpinoyLogger:67 - Executing Navigate
01:30:08,931 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:08,932 DEBUG devpinoyLogger:247 - Error came
01:30:08,932 DEBUG devpinoyLogger:256 - #####Navigate --- Pass
01:30:08,937 DEBUG devpinoyLogger:183 - Executing the test LoginTest
01:30:08,938 DEBUG devpinoyLogger:310 - Executing wait Keyword
01:30:09,346 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:09,348 DEBUG devpinoyLogger:83 - Executing input Keyword
01:30:09,439 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:09,440 DEBUG devpinoyLogger:168 - Executing clickLink
01:30:10,247 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:10,250 DEBUG devpinoyLogger:116 - Validating input Keyword
01:30:10,273 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:10,274 DEBUG devpinoyLogger:83 - Executing input Keyword
01:30:11,350 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:11,351 DEBUG devpinoyLogger:168 - Executing clickLink
01:30:11,998 DEBUG devpinoyLogger:201 - ***Result of execution -- Pass
01:30:11,999 DEBUG devpinoyLogger:116 - Validating input Keyword
01:30:45,232 DEBUG devpinoyLogger:149 - Error while validating input -errorUnable
Command duration or timeout: 33.23 seconds
```

Figure 4.24: Selenium log

## Chapter 5: Research Results

### 5.1 Introduction

This chapter outlines the research results and reports on significant outcomes. Research results study the data collected from both primary and secondary data sources. Results from the secondary data source, i.e. literature review, is already demonstrated in chapter 2. This chapter presents the findings from survey questionnaire and interviews of the focused group. Qualitative research method was applied to collect primary data in this research.

### 5.2 Qualitative Research Methods

The qualitative research method used for this studies in the form of the survey questionnaire and interviews. Survey questionnaire was distributed to 30 IT professionals. Out of 30, only 17 IT professionals participated in this survey. The researcher carried out five separate interviews with senior IT professionals from different companies. Out of five interviews, 2 of them were conducted face-to-face and three interviews was carried out through Google hangout. However, the response rate was 73% overall from different respondents. The researcher took notes during each interview and analyzed their response without getting bias. The purpose of this interview was to understand their view about automation and the automation framework.

### 5.3 Research Findings and Analysis

Summary of the findings from the questionnaire and interviews explained below. The results

have shown in the form of charts with their respective percentage against each question.

The

researcher asked open-ended questions in the survey questionnaire and interviews. During analysis of those responses, researcher prepared themes based on their inputs and prepared

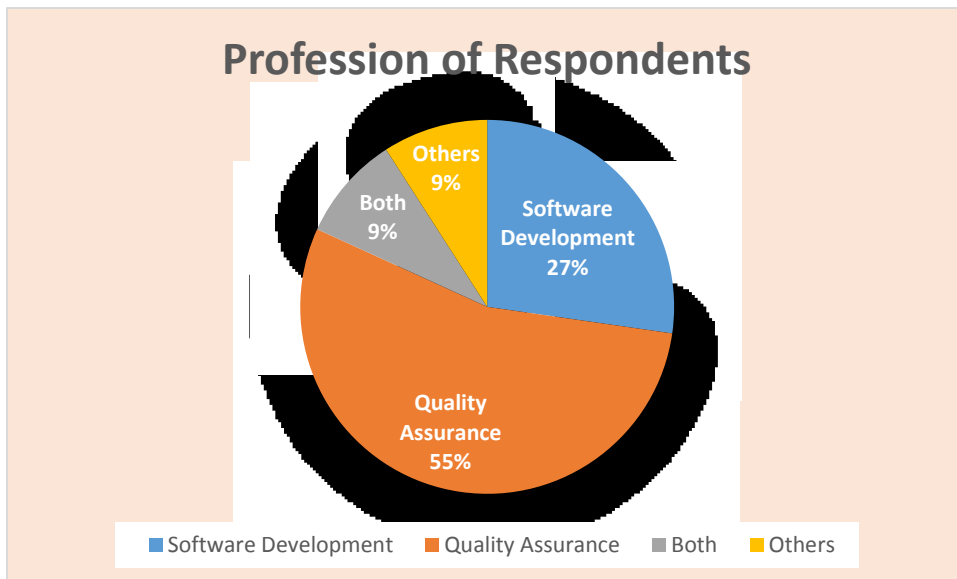
charts. The specific questionnaire which was given to participants has shown in Appendices 1.

#### 5.3.1 Professional Background of Respondents

The target audience of this research was IT professionals. The research carried out with 22 professionals. Out of 22, six people have been working as software developer. 12 are from

quality assurance background. Two of them have been worked in both developments as well as in testing field. Two of them works on different technologies in support.

Below chart shows the percentage of participants and their professional backgrounds-



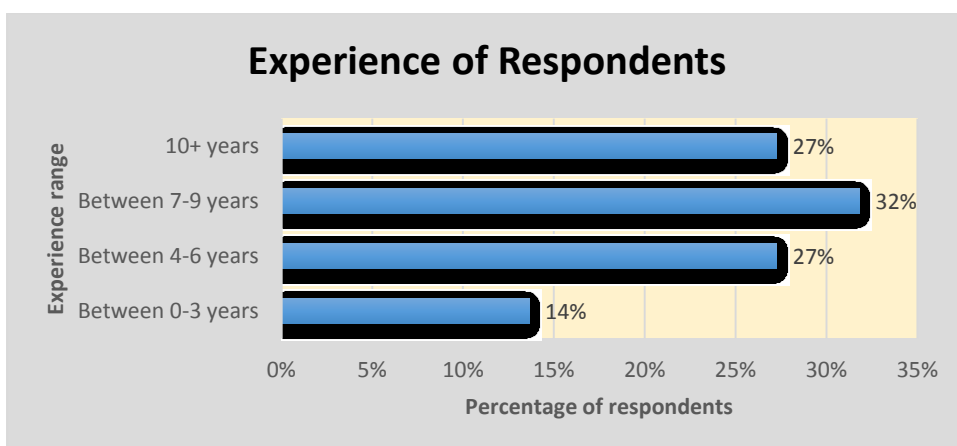
**Figure 5.1:** Profession of respondents

Above chart shows 27% of respondents are a developer, 55% are testers, 9% have worked in both area and rest 9% belongs to other technologies.

### 5.3.2 Professional Experience of Respondents

14% of the respondents were having experience between 0-3 years. 27% respondents had between 4 to 6 years of experience. 32% respondents had experience between 7-9 years.

Respondents who were having more than 10+ years of experience was 27%.

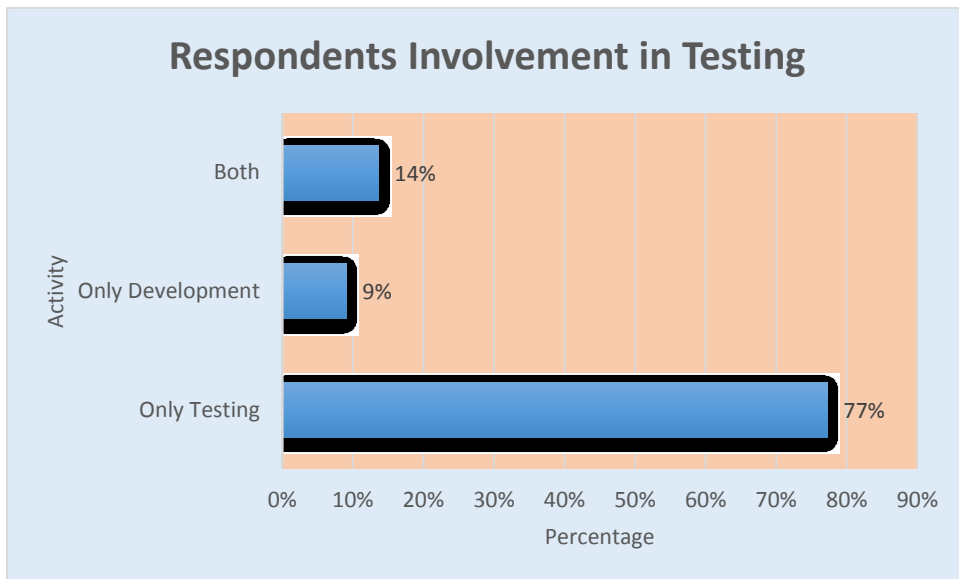


**Figure 5.2:** Professional experience of respondents

Below are the details of questions and their responses. These questions were asked in survey questionnaire and interviews.

#### **Question1: Have you ever involved in software testing activity?**

In response to the above question, 77% respondents said they were involved in testing. 9% of respondents felt that they were not engaged in testing. Rest 9% indicated that they had done both developments as well as testing.



**Figure 5.3:** Respondents involvement in testing

**Question2: What do you prefer- Manual testing or automation testing?**

To answer the above question, 5% respondents said they prefer manual testing. 45% of respondents stated that they prefer automation testing. 45% respondents were in both manual and automation testing. Rest 5% did not show any preference.

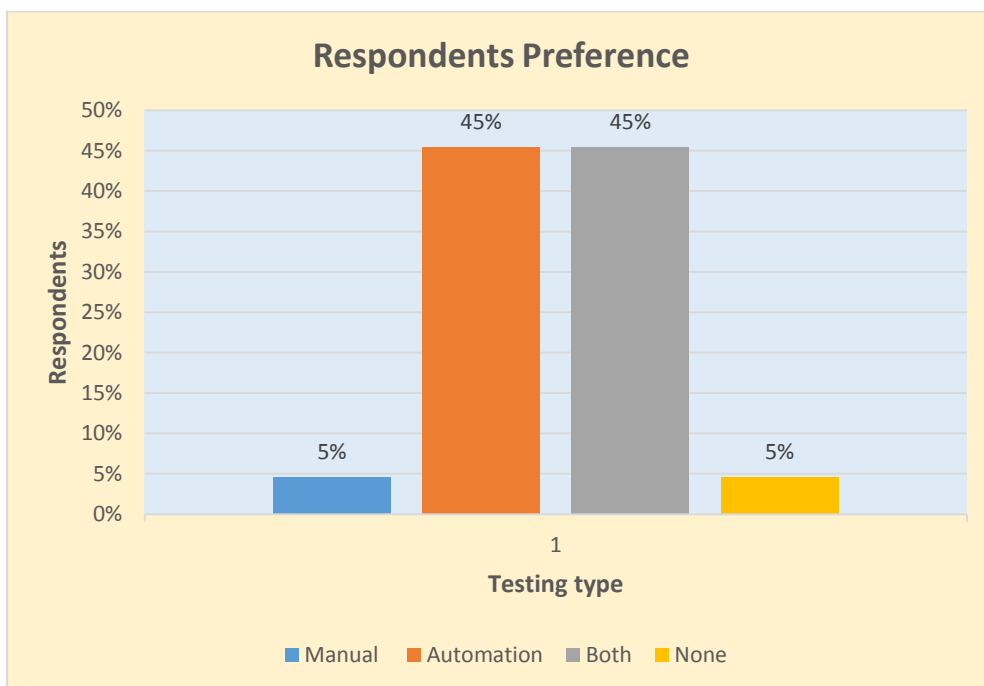


Figure 5.4: Respondents preference

**Question3: If you prefer automation testing then what are the reasons to select automation over manual?**

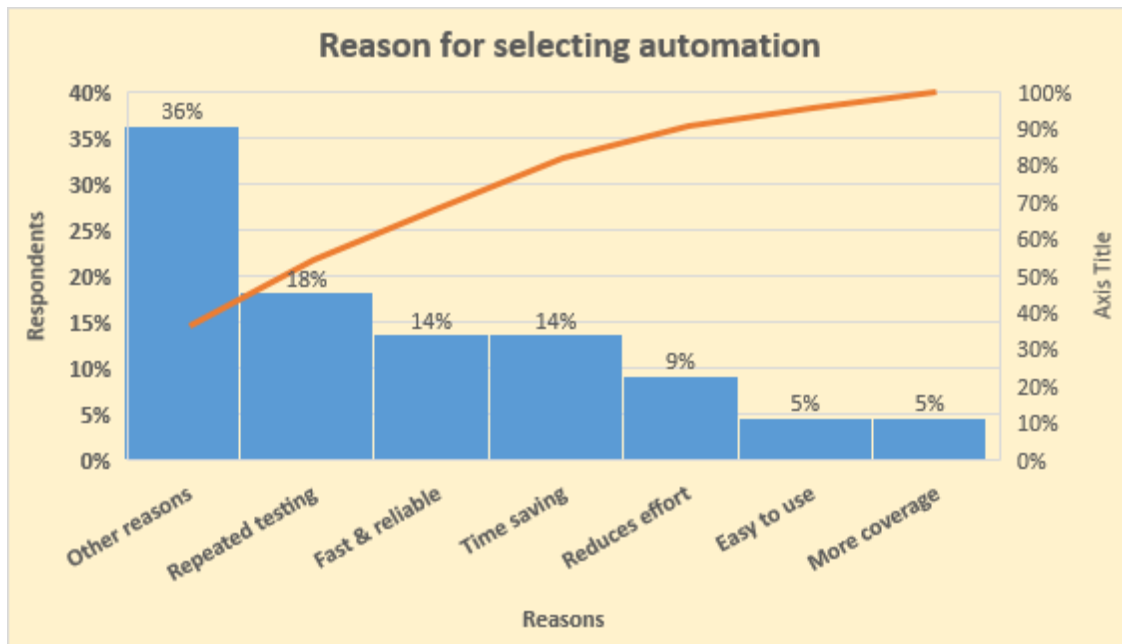


Figure 5.5: Reasons to select automation over manual

In the response to above question, respondents gave a mixed opinion. 5% respondents liked automation for more test coverage. Other 5% preferred automation because it is easy to use.

9% respondents preferred automation due to a reduction in effort. 14% chosen automation because it is time saving. Rest 14% said that it is fast and reliable. 18% preferred it due to help

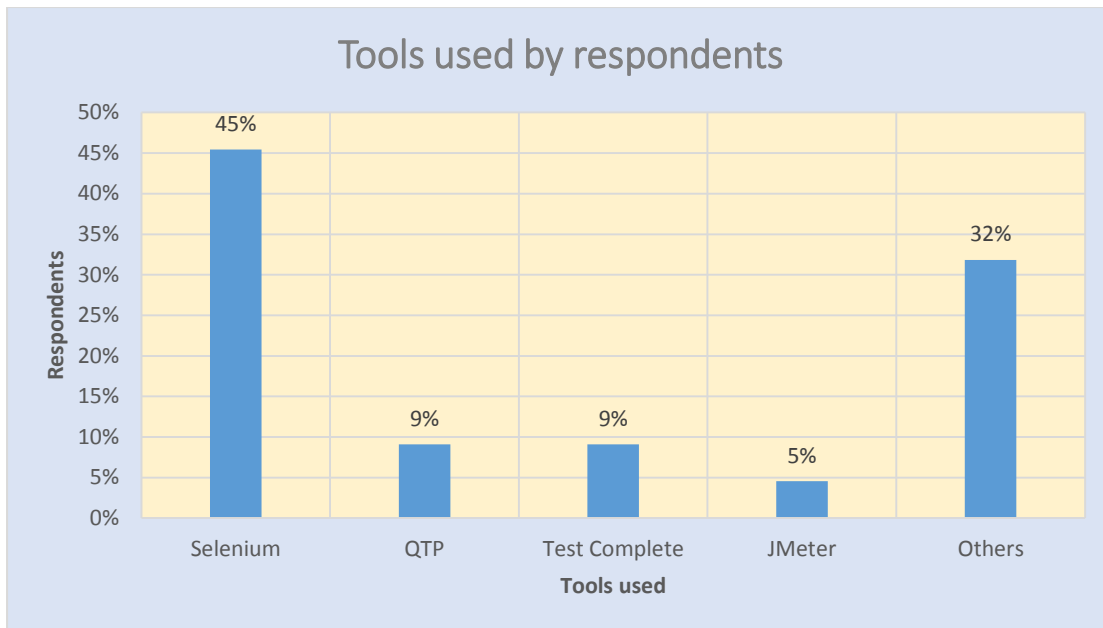
in repeated testing. 36% respondents believed in other reasons to select automation over the manual.

**Question4: Have you come across any automation tool? Please specify the name.**

In response to above question, 45% respondents had come across the selenium tool. QTP tool

was used by 9% of the interviewees. Other 9% respondents had used Test Complete tool.

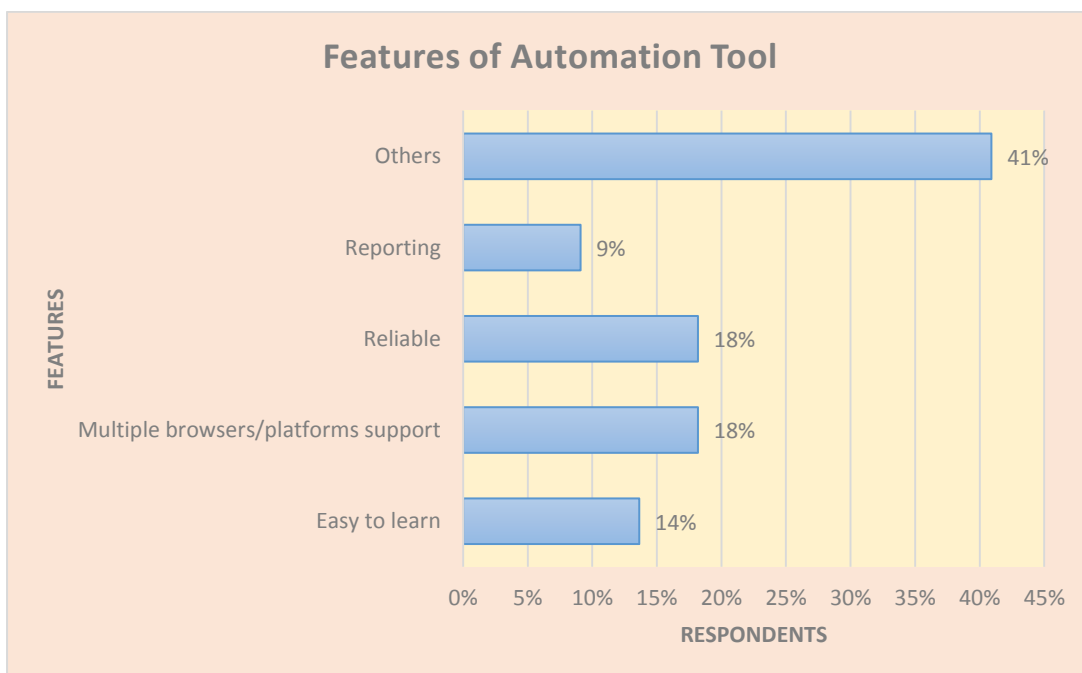
5% of respondents had used JMeter. 32% of respondents had not used any specific tool.



**Figure 5.6:** Tools used by respondents

**Question5: What features you expect from an automation tool?**

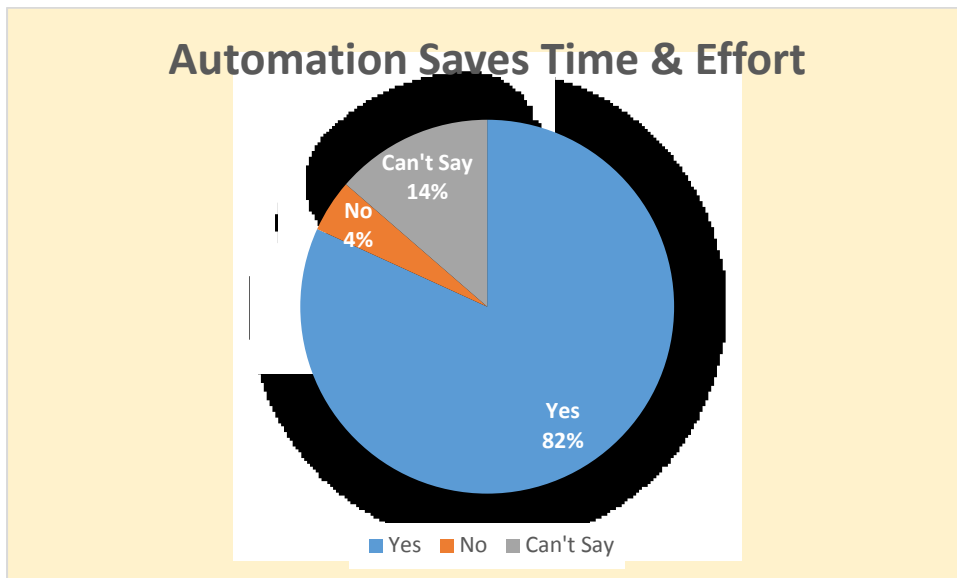
In response to above question, 14% respondents preferred a tool which is easy to use. 18% respondents wanted an automation tool which supports multiple platform and browsers. 18% preferred reliability and 9% chosen reporting features. 41% gave a mixed opinion which was not very specific.



**Figure 5.7:** Features of automation tool

**Question6: Do you believe that automation testing saves testing efforts and time? If yes, then how?**

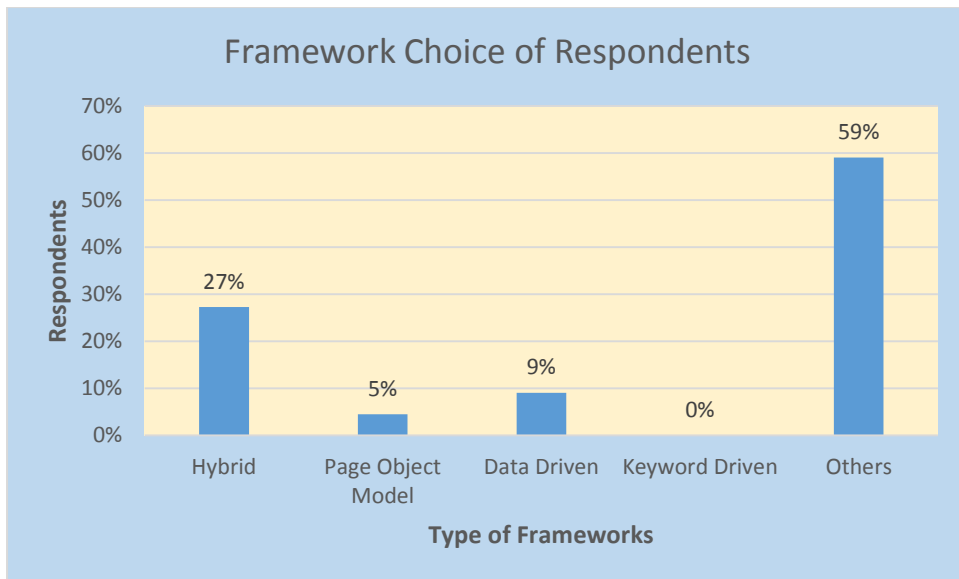
In response to above question, 82% of respondents said that yes, automation saves time and effort. On the other hand, 4% did not agree with the question. 14% respondents had no idea about this question.



**Figure 5.8:** Automation saves time and effort

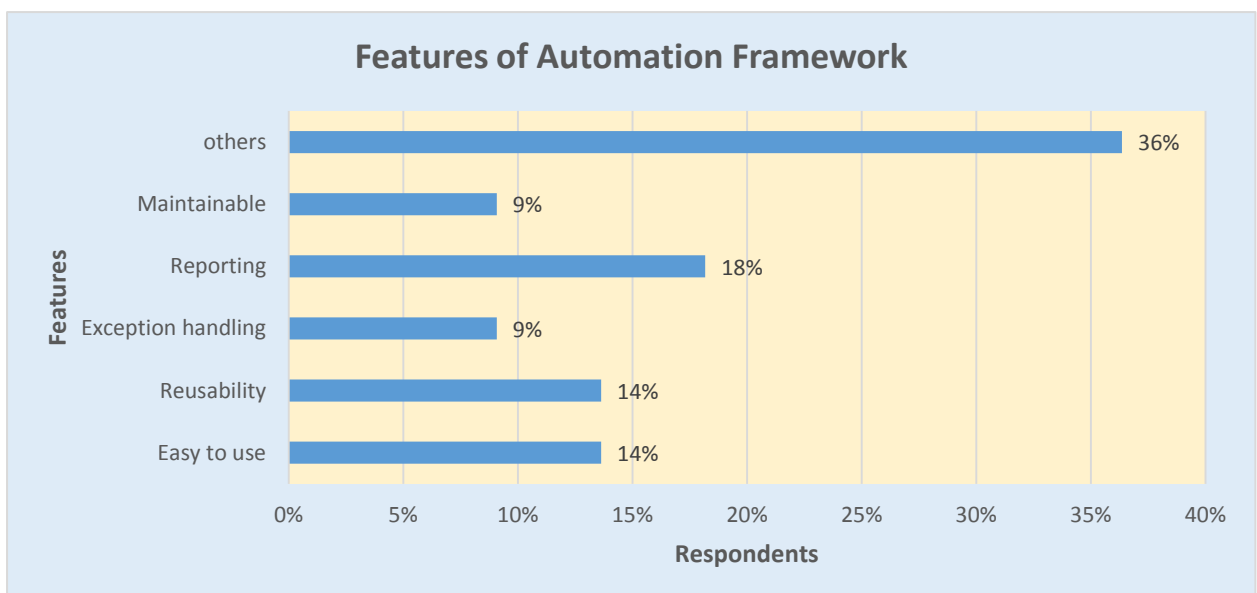
**Question7: Which automation framework you use for testing?**

In the response to above question, 27% of respondents used the hybrid framework. 5% had used page object model framework. 9% had used data driven framework. No respondent gave any view about keyword driven framework. 59% respondents had either not used or had no idea about the framework.



**Figure 5.9:** Framework choice of respondents

**Question8: What type of features you expect from an automation framework?**

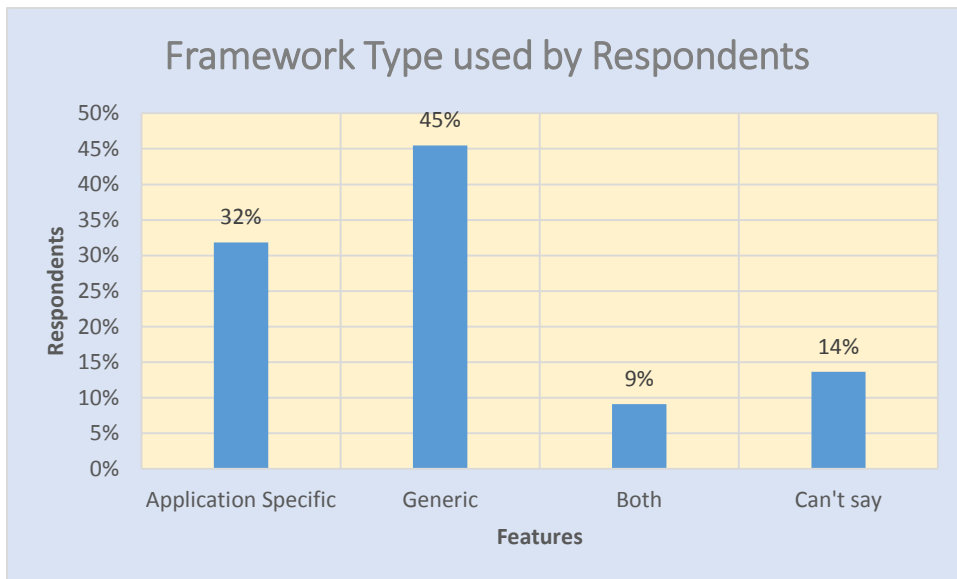


**Figure 5.10:** Features of automation framework

In answer to the above question, 14% of respondent anticipated that automation framework must be easy to use. 14% wanted a reusable framework. 9% preferred exception handling feature. 18% of the interviewees expected reporting features in the framework. 9% wanted the framework must be maintainable. 36% of respondents had either no idea or did not respond.

**Question9: Automation framework which you use is application-specific or generic? How?**

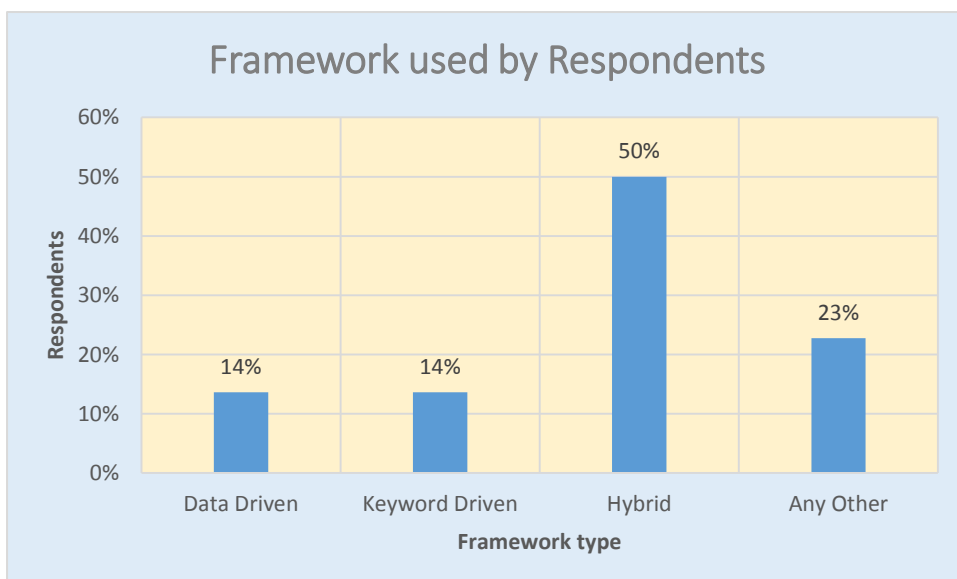
In response to above question, 32% respondents liked application-specific framework. On the other hand, 45% respondents wanted a generic framework. 9% were in the favor of both types of framework. 14% of respondents did not answer.



**Figure 5.11:** Framework type used by respondents

**Question10: Which model of Automation framework you use- keyword driven, data driven or hybrid? Why?**

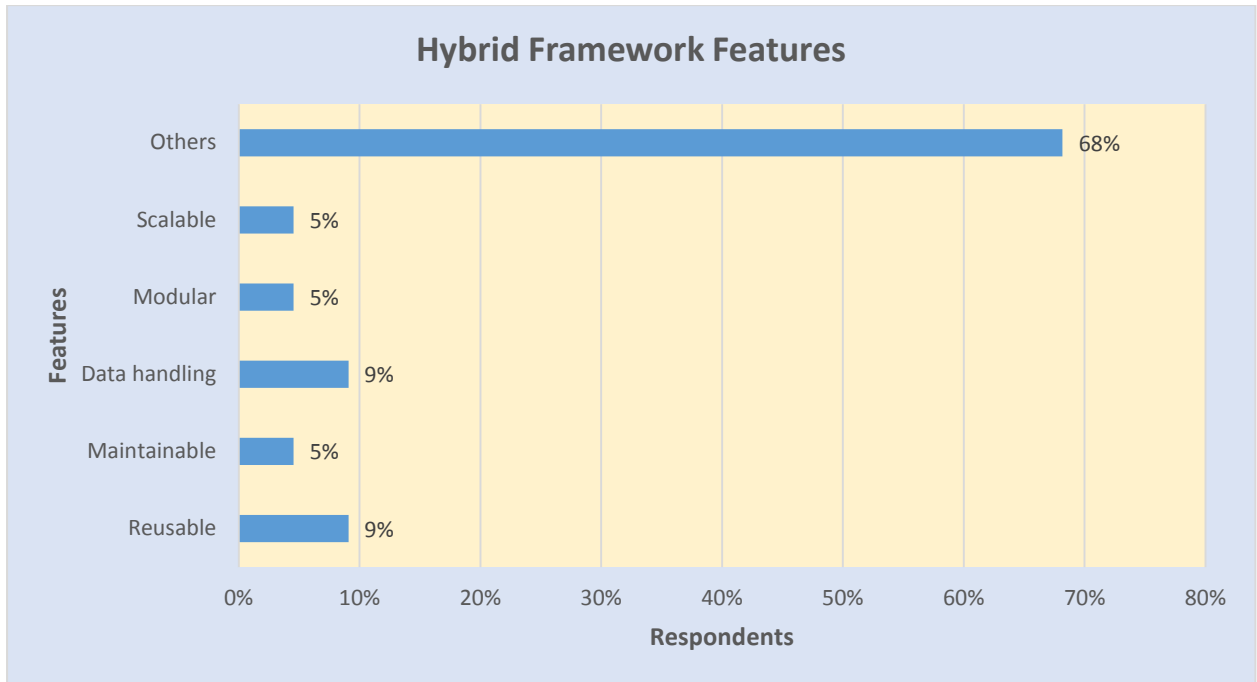
In the response to above question, 50% respondents selected a hybrid framework, 14% liked data-driven & keyword driven framework. 23% of respondents had a different view or no view.



**Figure 5.12:** Framework used by respondents

**Question11: What are the benefits of Hybrid model framework over other models?**

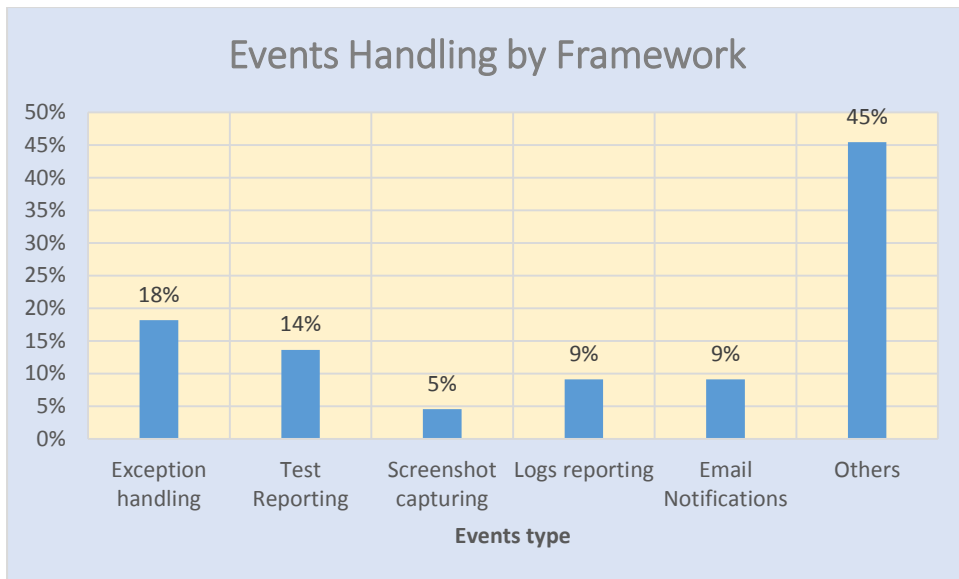
9% respondents answered the above question for reusability. 5% respondents wanted their hybrid framework as maintainable. 9% respondents wanted data handling feature. 5% of respondents liked modularity and 5% like scalability feature in their hybrid framework. 68% respondents had no specific view.



**Figure 5.13:** Hybrid framework features

**Question12: What are the other events your framework capable of handling apart from test execution?**

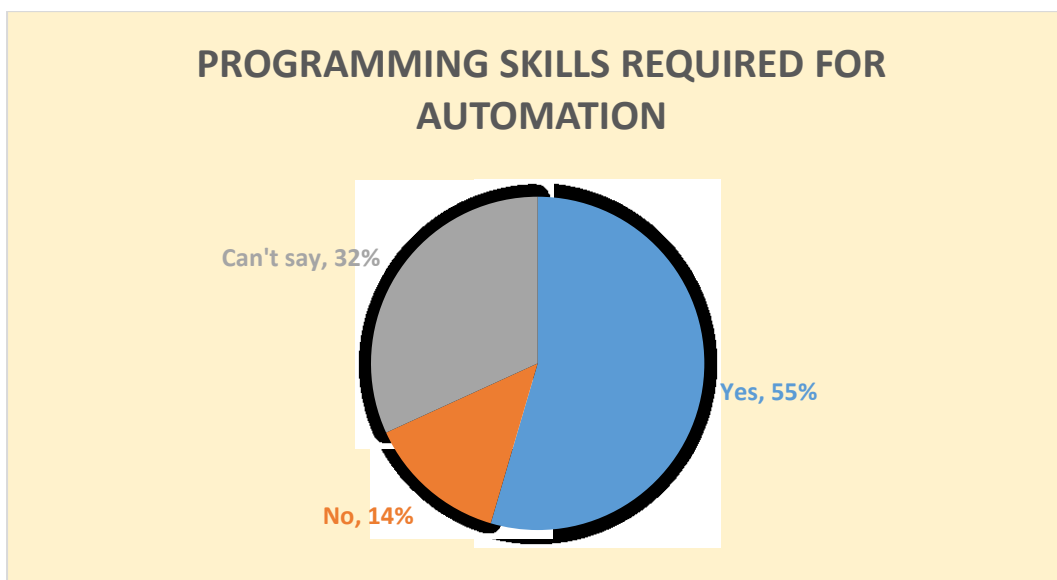
In response to above question, exception handling feature is preferred by 18% of respondents. 14% liked test reporting feature. 5% preferred screenshot capturing feature. 9% of respondents chosen log reporting feature. 9% respondents liked email notifications features. 45% of respondents had no specific view.



**Figure 5.14:** Events handling by framework

**Question13: Have your framework designed in such a way that user can easily run the tests and monitor the status without any programming skills? Briefly, explain.**

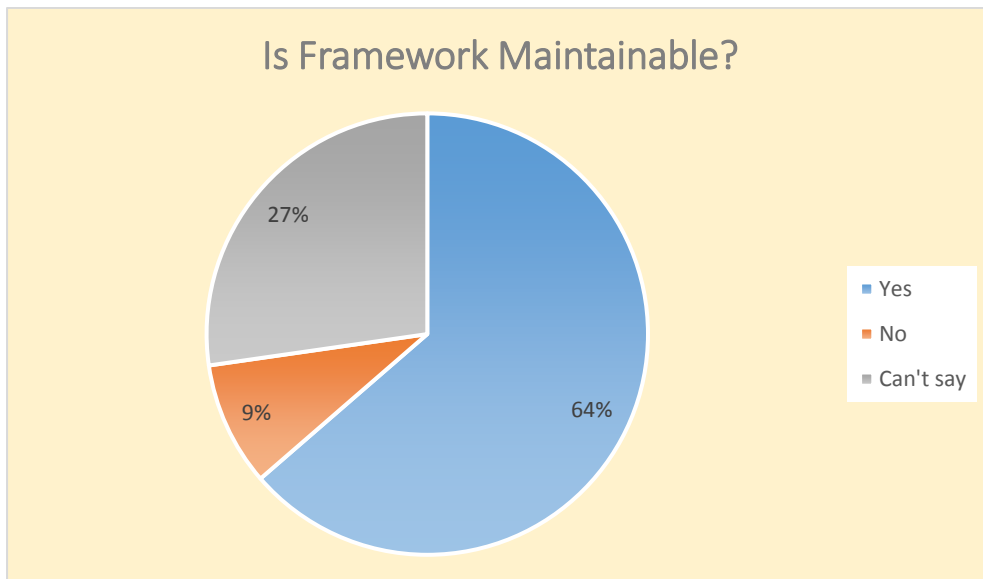
In the response to above question, 55% of respondents suggested that programming skill is required for test automation execution. 14% respondents did not agree with above view. 32% of respondents had no specific view.



**Figure 5.15:** Programming skills required for automation

**Question14: How your framework has designed to add and maintain new features easily?**

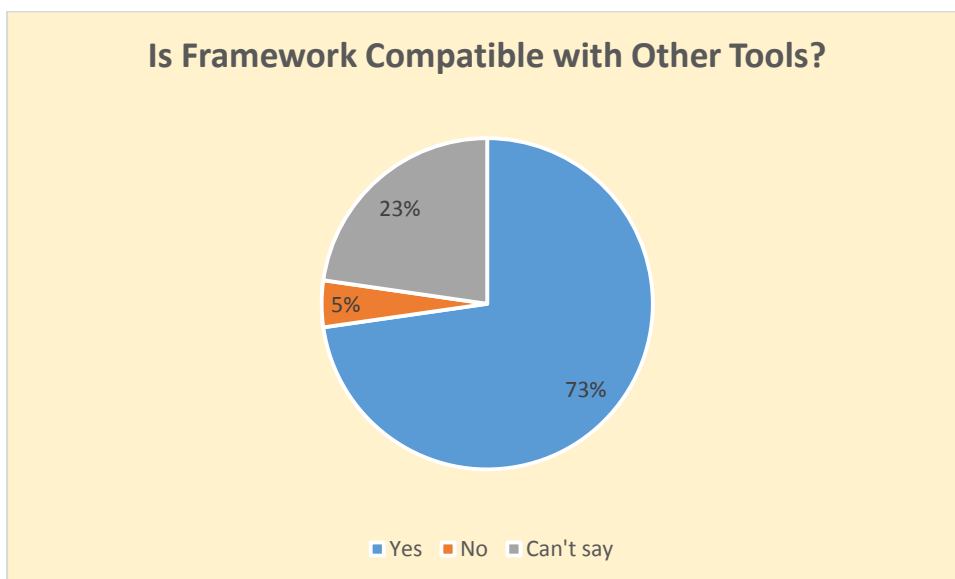
To answer this question, respondents were not specific about answering how. However, 64% of The respondents said that their framework had been designed to maintain easily. 9 % were not agree with this. 27% did not answer about this.



**Figure 5.16:** Is framework maintainable

**Question15: Is your framework design compatible with other tools like TestNG, Maven, SVN, GitHub?**

Answering above question, 73% of respondents said that their framework is compatible with other tools. On the other hand, 5% did not agree. 23% respondents had no specific answer.



**Figure 5.17:** Is framework compatible with other tools

## 5.4 Summary

The above results are combined responses from survey questionnaire & interviews. The researcher analysed all findings together to reach the conclusion. Next chapter discusses the research results.

## Chapter 6: Research Discussion

### 6.1 Introduction

The main objective behind this chapter is to introspect the requirements that were specified in the first chapter and evaluate how many of them have been achieved. This chapter outlines the conclusion and the recommendations regarding the findings of this research. The reason of this research is to summarize the theoretical suggestions which have taken from the literature review and the findings carried out in qualitative research. The objective of this study was to gain a better understanding of reusable automation framework. This research purpose has been achieved by answering the research question and satisfying all the specified objectives. The strength and limitations of the study have discussed, and recommendations for further study has been suggested.

### 6.2 Research conclusions

The purpose of this research was to collect more and more information about automation framework. The questions were arranged in a systematic way keeping in mind the exposure of the respondents towards automation framework. The participants enjoyed this survey. The questions had composed in such a way that it first started with basics and then went into details. The majority of the participants (77%) had exposure with quality assurance.

**In first three questions, the** researcher tried to investigation their interest towards automation testing. 45% of the respondents felt that automation is better than manual testing. In response to the third question, participants explained various benefits of automation testing. Like- test coverage, reduction in effort, time saving, fast and reliable, repeated testing. Literature review chapter also discussed the need for automation and its advantages.

**In next three questions,** the researcher tried to investigate about the automation tool which is mostly used for automation testing. Survey revealed that participant mostly uses

selenium. 45% were in the favor of selenium tool. Their selection criteria for an automation tool was based on features like- reliability, multiple platforms, and browser support, easy to use and test report.

**Questions from 7-10** tried to investigate about automation framework. 27% of respondents gave their opinion in the favor of Hybrid framework. The hybrid framework is far ahead in the survey from other frameworks. Another question that was related to the features of automation framework. Most of the respondents answered that they want a framework which must be easy to use, maintainable, generate reports, handle exceptions and reusable. When was it asked what kind of automation framework they prefer? 45% were for the generic framework and only 27% shown interest in the application-specific framework. When respondents had given a chance to choose among three frameworks (question no 10), then 50% of them chosen a hybrid framework. Respondents reiterated the same features for the hybrid framework.

As per the responses, people expected from framework to handle many events. Some respondents prefer exception handling, test reporting, screenshot capturing, logs reporting, email notification.

It had been asked whether programming skill required for test automation? 55% of respondents agreed with it. In response to asking about maintainability, 64% respondents were in favour. 73% agreed with the compatibility of the framework with another tool.

### **6.3 Research question achieved**

The requirements for this research was specified in chapter 1. This research meets with its requirements. A reusable hybrid framework prototype has been developed with features like test execution, reporting, log generation, email notification and screenshot capturing. These features and requirements had been gathered during primary and secondary research. Based on the above facts researcher concluded that this research meets its requirements.

### **6.4 Limitations of research**

Test automation is a very vast subject. There are numerous possible approaches available which suit in different cases. This research only concentrates on developing a reusable prototype hybrid automation framework. This framework is for test execution, reporting, log generation, email notification and screenshot capturing. A lot of other approaches to

bulk test execution, deployment, parallel test script development has not considered. It requires more time and resources to achieve those features.

## 6.5 Research Recommendations

The researcher has acknowledged the following recommendations that should be considered to improve the efficiency of the hybrid framework. By integrating new tools, its usability can be enhanced.

1. By integrating with ANT tool, running new build and deployment would become faster with adding this tool. It will save more time.
2. Integrating code with GitHub, parallel test script development would be possible.
3. By integrating with CI tools like Jenkins, bulk execution of test cases can be achieved.

All the above recommendation will further improve the automation framework structure.

## Chapter7: Reflection and Learning

### 7.1 Introduction

This chapter discussed the personal learning and the key skill gained by the researcher throughout the MSc. Program. It also evaluates the learning process throughout the course duration. It discusses the area where the researcher was weak and has improved during the study. This chapter also focuses on how the MSc. Information Systems with Computing has added value to the researcher's expertise and how it has provided learning with personal development.

### 7.2 Learning Styles

It is better to discuss first the different learning styles which best describes the researcher.

**Kolb's (1984)** learning theory provides a way to identify different type of learning style.

Kolb's explained four type of learning styles which are based on a four-stage learning cycle.

People with a Convergent learning style can resolve issues. They use their learning to find a solution to the problem. They are extrovert kind of person. They are more interested in technical tasks than social issues. Such people have their decision skills.

In the Divergent learning style, people are the introvert type, and they prefer to observe rather than perform. Such people collect information and use imagination to resolve the issue. They are quite emotional. They perform only in the situation which requires thought generation. These people are social by nature having cultural interests.

The Assimilating learning style is a short and a reasonable approach. In this learning, ideas are more important than people. These people have a clear, logical format for every problem.

When it comes to research situation, they stress on theory and logical analysis.

The Accommodating learning style beliefs on perception, not on logic. People use others analysis and try to experiment. They like new challenges. They depend on others information

but work to their analysis (Kolb and Kolb, 2005).

Acquiring different learning style depends on various situations. Below diagram shows Kolb learning style.

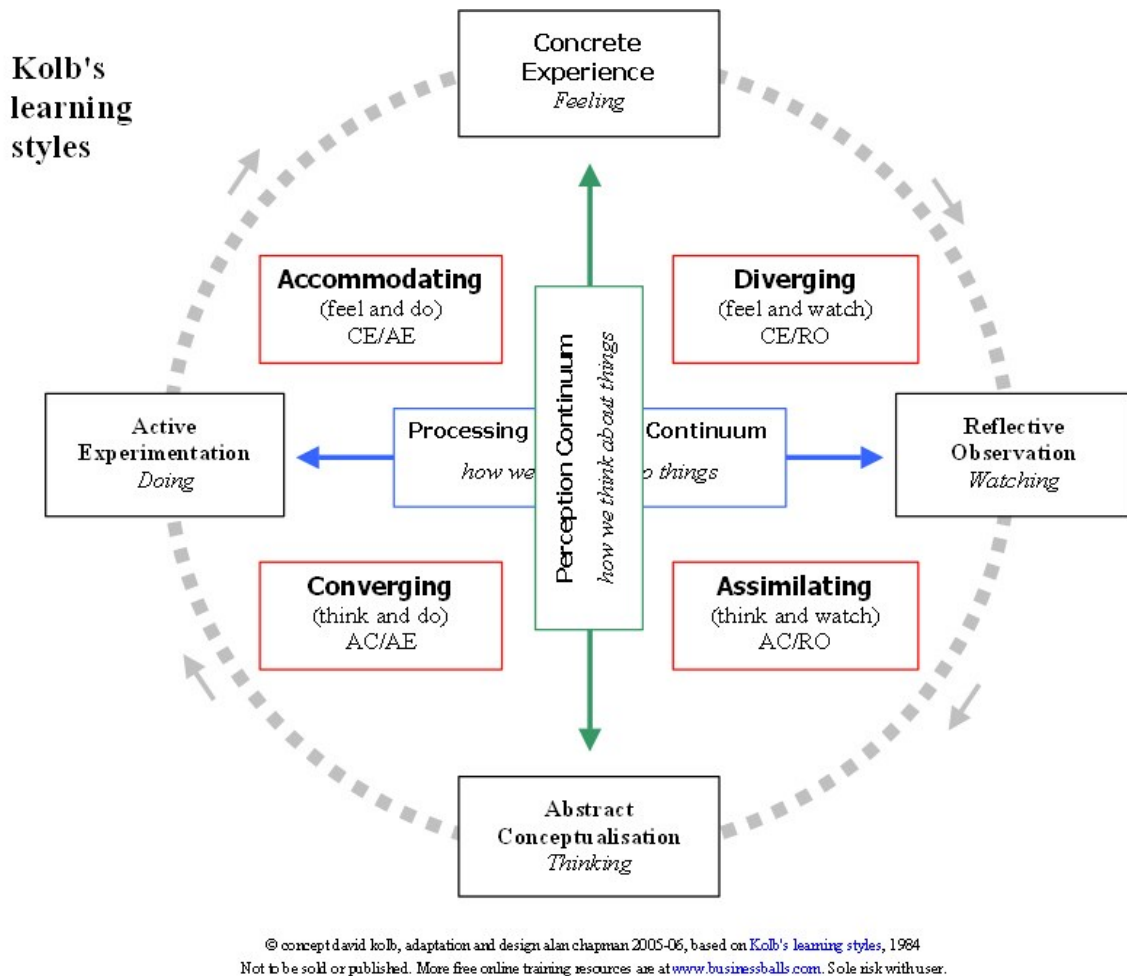


Figure 7.1: Kolb's Learning Style (Kolb's learning styles, 2016)

### 7.3 Reflection on learning

Researcher's learning style is more inclined towards convergent. Since this research belongs to the technical background where people have to identify the problem and find a solution, hence convergent learning style matches with it. The researcher is open to adapt changes, and technology field enforces to be ready for changes. Technical field also demands to take a decision and taking right decision makes a person be an extrovert. Therefore, researchers learning matches with convergent learning style.

### 7.4 Skills developed

The researcher has improved his skills during the MSc course. These skills are technical and

vital for choosing a professional career path. Through the course, the researcher was introduced to modules like software engineering, object oriented programming, database, mobile application and enterprise applications. It helped to improve technical skills as well as problem-solving skills. Assignments and presentations contributed improving the communication skill. During the research, researcher learned how to design questionnaire, how to research and investigate. Data collection and data analysis also helped to improve the analytical skill.

### **7.5 Future Application**

The aims and objectives of the MSc program were designed to improve the technical skills which are required for the job market. It has added value to the researcher to choose a career path into the technical field. The researcher is confident that this experience will be helpful to him personally and professionally and looking forward to succeeding in a career.

## Bibliography

\* (2015) 'Hybrid test automation framework', *Test Automation Guru*, 28 August. Available at: <http://www.testautomationguru.com/hybrid-test-automation-framework/> (Accessed: 16 August 2016).

*Apache log4j 1.2* - (2012) Available at: <https://logging.apache.org/log4j/1.2/index.html> (Accessed: 10 August 2016).

*Apache POI - java excel APIs* (2016) Available at: [http://www.tutorialspoint.com/apache\\_poi/apache\\_poi\\_java\\_excel.htm](http://www.tutorialspoint.com/apache_poi/apache_poi_java_excel.htm) (Accessed: 16 August 2016).

*Based on Saunder's Research Onion* (2013) Available at: <http://writepass.com/journal/2013/04/sample-dissertation-methodology/> (Accessed: 13 August 2016).

Bhondokar, B., Ranawade, P., Jadhav, S. and Vibhute, M. (2015) 'Hybrid Test Automation Framework for Web Application', *International Journal of Engineering Research & Technology (IJERT)*, 4(4), pp. 1–3.

*Brief guide to open ended survey questions* (no date) Available at: [https://cms.cerritos.edu/uploads/ResearchandPlanning/Brief\\_Guide\\_to\\_Open-Ended\\_Survey\\_Questions.pdf](https://cms.cerritos.edu/uploads/ResearchandPlanning/Brief_Guide_to_Open-Ended_Survey_Questions.pdf) (Accessed: 22 August 2016).

Chandraprabha, C., Kumar, A. and Saxena, S. (2015) 'Data driven testing framework using Selenium WebDriver', *International Journal of Computer Applications*, 118(18), pp. 18–23. doi: 10.5120/20845-3497.

Constantinescu, R. and Iacob, I.M. (2008) *Quality in statistics TESTING: FIRST STEP TOWARDS SOFTWARE QUALITY Ioan Mihnea IACOB*. Available at: [http://jaqm.ro/issues/volume-3,issue-3/pdfs/iacob\\_constantinescu.pdf](http://jaqm.ro/issues/volume-3,issue-3/pdfs/iacob_constantinescu.pdf) (Accessed: 11 August 2016).

Crispin, L. (2009) *An overview of agile testing*. Available at: <http://lisacrispin.com/downloads/AgileTestingOverview.pdf> (Accessed: 14 August 2016).

*Data Driven Testing Framework* (2016a) Available at: <http://www.softwaretestinghelp.com/test-automation-frameworks-selenium-tutorial-20/> (Accessed: 16 August 2016).

*Data Driven Testing Framework* (2016b) Available at: <http://www.softwaretestinghelp.com/test-automation-frameworks-selenium-tutorial-20/> (Accessed: 16 August 2016).

Falcao, G. (2015) *Hybrid automation framework Developement*. Available at: [http://www.slideshare.net/glasdonfalcao/hybrid-automation-framework-developement?qid=993d60ae-49b0-4810-85f9-8090c445ccd0&v=&b=&from\\_search=1](http://www.slideshare.net/glasdonfalcao/hybrid-automation-framework-developement?qid=993d60ae-49b0-4810-85f9-8090c445ccd0&v=&b=&from_search=1) (Accessed: 16 August 2016).

Fewster, M. and Graham., D. (1999) *Software Test Automation*. Addison-Wesley.

Ghanakota, G. (2012) *TESTING FRAMEWORKS*. Available at: <https://www.cs.colorado.edu/~kena/classes/5828/s12/presentation-materials/ghanakotagayatri.pdf> (Accessed: 14 August 2016).

GmbH, G.S. (2013) *Software testing tools*. Available at: <http://www.testingtools.com/test-automation/> (Accessed: 15 August 2016).

Graham, D. and Fewster, M. (1999) *Software test automation: Effective use of test execution tools*. United States: Addison-Wesley Educational Publishers.

Grater, M.T. (2005) *Benefits of Using Automated Software Testing Tools to Achieve Software Quality Assurance*. Available at: <https://scholarsbank.uoregon.edu/xmlui/bitstream/handle/1794/7817/2005-grater.pdf?sequence=1> (Accessed: 16 August 2016).

Guindon, C. (2016a) *Eclipse desktop & web IDEs*. Available at: <https://eclipse.org/ide/> (Accessed: 10 August 2016).

Guindon, C. (2016b) *Eclipse desktop & web IDEs*. Available at: <https://eclipse.org/ide/> (Accessed: 10 August 2016).

Gulia, A. (2010) 'Automation Framework: Modular Automation Framework', *Software Testing Glossary*, 29 November. Available at: <http://bugfinding.blogspot.ie/2010/11/automation-framework-modular-automation.html> (Accessed: 8 August 2016).

Gupta, S., Kumar, S. and Saxena, C. (2015) 'Review Paper on Comparison of Automation Testing Tools Selenium and QTP', *MIT International Journal of Computer Science and Information Technology*, 5(2), pp. 55–57.

Hair, J.F., Celsi, M.W., Money, A.H., Samouel, P. and Page, M.J. (2003) *ESSENTIALS OF Business Research Methods*. Available at: [http://samples.sainsburysebooks.co.uk/9781317471233\\_sample\\_1013989.pdf](http://samples.sainsburysebooks.co.uk/9781317471233_sample_1013989.pdf) (Accessed: 22 August 2016).

Hayes, L.G. (1995) *Automated testing handbook*. Software Testing Inst.

*HP QuickTest professional for business process testing user guide* (2010) Available at: <http://www.asi-test.com/Docs/QTPv11/QTP4BPT.pdf> (Accessed: 15 August 2016).

Hurley, J. (2009) *A Study of the Effectiveness of an online teacher professional development programme and its impact on ICT integration*. EMBA thesis.

*Induction* (2006) Available at: <http://www.socialresearchmethods.net/kb/dedind.php> (Accessed: 14 August 2016).

*Interpretivism (interpretivist) research philosophy - research methodology* (no date) Available at: <http://research-methodology.net/research-philosophy/interpretivism/> (Accessed: 14 August 2016).

*Introduction — Selenium documentation* (2008) Available at: [http://docs.seleniumhq.org/docs/01\\_introducing\\_selenium.jsp#test-automation-for-web-applications](http://docs.seleniumhq.org/docs/01_introducing_selenium.jsp#test-automation-for-web-applications) (Accessed: 15 August 2016).

Johnson, R.E. and Foote, B. (1988) *Designing reusable classes*. Available at: <http://www.cse.msu.edu/~cse870/Input/SS2002/MiniProject/Sources/DRC.pdf> (Accessed: 17 August 2016).

Kaner, C. (2008) 'A Tutorial in exploratory testing', Available at: <http://www.kaner.com/pdfs/QAIExploring.pdf> (Accessed: 11 August 2016).

Kaner, C., Pettichord, B. and Bach, J. (2002) *Lessons learned in software testing: A context driven approach*. New York: Wiley, John & Sons.

Kaur, H. and Gupta, D.G. (2013) 'Comparative Study of Automated Testing Tools: Selenium, Quick Test Professional and Testcomplete', *International Journal of Engineering Research and Applications*, 3(5).

Kaur, M. and Singh, R. (2014) 'A Review of Software Testing Techniques', *International Journal of Electronic and Electrical Engineering*, 7(5).

Kolb, A.Y. and Kolb, D.A. (2005) *The Kolb learning style Inventory—Version 3.1 2005 technical Specifications*. Available at: [http://learningfromexperience.com/media/2010/08/Tech\\_spec\\_LSI.pdf](http://learningfromexperience.com/media/2010/08/Tech_spec_LSI.pdf) (Accessed: 21 August 2016).

*Kolb's learning styles* (2016) Available at: <http://www.businessballs.com/kolblearningstyles.htm> (Accessed: 21 August 2016).

Lamba, S., Rishiwal, V. and Rana, A. (2015) 'AN AUTOMATED DATA DRIVEN CONTINUOUS TESTING FRAMEWORK', *International Journal Of Advanced Technology In Engineering And Science Wwww.Ijates*, 1(0), pp. 2348–7550.

Laukkanen, P. (2006) *Data-Driven and Keyword-Driven Test Automation Frameworks*. Available at: <http://eliga.fi/Thesis-Pekka-Laukkanen.pdf>.

Mailewa, A., Herath, J. and Herath, S. (2015) *A survey of effective and efficient software testing*. Available at: [http://www.micsymposium.org/mics2015/ProceedingsMICS\\_2015/Mailewa\\_2D1\\_41.pdf](http://www.micsymposium.org/mics2015/ProceedingsMICS_2015/Mailewa_2D1_41.pdf) (Accessed: 14 August 2016).

Malhotra, P. (no date) *Test automation tool comparison – HP UFT/QTP vs. Selenium - Prashant Malhotra*. Available at: <http://www.aspiresys.com/WhitePapers/QTPvsSelenium.pdf> (Accessed: 15 August 2016).

Mathew, J.R. (2011) *Hybrid framework best test automation framework for ERP projects*. Available at: <http://www.isqtinternational.com/Step-Auto/ERPPresentations2011/Paper%2007.pdf> (Accessed: 16 August 2016).

Maxwell, J.A. (2005) *Designing a Qualitative Study*. Available at: [https://us.corwin.com/sites/default/files/upm-binaries/23772\\_Ch7.pdf](https://us.corwin.com/sites/default/files/upm-binaries/23772_Ch7.pdf) (Accessed: 22 August 2016).

Meller, I. (2013a) *Use of technology and its influence on project success A study of e-commerce companies in Ireland*. MBA thesis. Liverpool John Moores University and the Dublin Business School.

Meller, I. (2013b) *Use of technology and its influence on project success A study of e-commerce companies in Ireland*. MBA thesis. Liverpool John Moores University and the Dublin Business School.

Memon, A.M., Pollack, M.E. and Soffa, M.L. (1999) 'Using a goal-driven approach to generate test cases for GUIs', *Proceedings of the 21st international conference on Software engineering - ICSE'99*,. doi: 10.1145/302405.302632.

Metha, K., Chavan, S. and Patil, M. (2003) 'Towards developing a Selenium based GUI automation test pro: A comparative study', Vol. 3 - Issue 2 (February - 2014)(Vol. 3 - Issue 2 (February - 2014)).

MIT Publications @BULLET Automation Testing overcome all the problems of manual testing with using automation framework like Selenium , QTP , Win runner etc (2015) *Review paper on comparison of automation testing tools Selenium and QTP*. Available at: [http://www.mitpublications.org/yellow\\_images/1441275591\\_logo\\_1.pdf](http://www.mitpublications.org/yellow_images/1441275591_logo_1.pdf) (Accessed: 15 August 2016).

*Most popular test automation Frameworks with pros and cons of each – Selenium Tutorial #20* (2016) Available at: <http://www.softwaretestinghelp.com/test-automation-frameworks-selenium-tutorial-20/> (Accessed: 10 August 2016).

Nemeth, L. (2015) *Keyword driven testing in theory and practice*. Available at: [http://www.hustef.hu/hustef/web\\_2015.nsf/0/237DE64F27A4E799C1257EE5002EA414/\\$FILE/HUSTEF-2015\\_Levente-Nemeth\\_Keyword-Driven-Testing-In-Theory-And-Practice.pdf](http://www.hustef.hu/hustef/web_2015.nsf/0/237DE64F27A4E799C1257EE5002EA414/$FILE/HUSTEF-2015_Levente-Nemeth_Keyword-Driven-Testing-In-Theory-And-Practice.pdf) (Accessed: 16 August 2016).

Patwa, P. (2009) *Hybrid test automation Frameworks implementation using QTP*. Available at:  
<https://www.stickyminds.com/sites/default/files/article/file/2014/Hybrid%20Test%20Automation%20Frameworks%20Implementation%20using%20QTP.pdf> (Accessed: 16 August 2016).

Pillai, A. (2011) *Hybrid Framework in QTP – The Basics*. Available at:  
<http://www.automationrepository.com/2013/07/qtp-hybrid-framework-the-basics/> (Accessed: 16 August 2016).

Polo, M., Tendero, S. and Piattini, M. (2006) 'Integrating techniques and tools for testing automation (PDF)', 17(1), pp. 3–39.

Purohit, K.G. (2012) *Influence of Project Management Office*. MBA thesis. .

*Qualitative analysis strategies for Analysing open-ended survey questions in MAXQDA* (2016) Available at:  
[http://www.surrey.ac.uk/sociology/research/researchcentres/caqdas/support/analysingsurvey/qualitative\\_analysis\\_strategies\\_for\\_analysing\\_openended\\_survey\\_questions\\_in\\_maxqda.htm](http://www.surrey.ac.uk/sociology/research/researchcentres/caqdas/support/analysingsurvey/qualitative_analysis_strategies_for_analysing_openended_survey_questions_in_maxqda.htm) (Accessed: 22 August 2016).

Rahman, M.F. (2014) *BROWSER-BASED AUTOMATION TESTING USING SELENIUM WEBDRIVER*. Bachelor thesis. TURKU UNIVERSITY OF APPLIED SCIENCES. Available at:  
[https://www.theseus.fi/bitstream/handle/10024/79740/thesis\\_final\\_rahman.pdf?sequence=1](https://www.theseus.fi/bitstream/handle/10024/79740/thesis_final_rahman.pdf?sequence=1) (Accessed: 2016).

Randhawa, R. and Pandey, L. (2006) *Automated regression testing framework using Keyword driven approach*. Available at:  
[http://www.sapient.com/content/dam/sapient/sapientcorporation/images/ImageDownloader/149/Sapient\\_AutomatedRegression.pdf](http://www.sapient.com/content/dam/sapient/sapientcorporation/images/ImageDownloader/149/Sapient_AutomatedRegression.pdf) (Accessed: 16 August 2016).

Rashmi and Bajpai, N. (2012) 'A Keyword driven framework for testing web applications', *International Journal of Advanced Computer Science and Applications*, 3(3). doi: 10.14569/ijacsa.2012.030302.

Saunders, M.N.K., Lewis, P., Thornhill, A. and Thorn., A. (2009) *Research methods for business students (5th edition)*. Available at:  
[https://is.vsfs.cz/el/6410/leto2014/BA\\_BSeBM/um/Research\\_Methods\\_for\\_Business\\_Students\\_\\_5th\\_Edition.pdf](https://is.vsfs.cz/el/6410/leto2014/BA_BSeBM/um/Research_Methods_for_Business_Students__5th_Edition.pdf) .

*Selecting the right automated testing tool* (2016) Available at:  
<https://smartbear.com/learn/automated-testing/selecting-automated-testing-tools/> (Accessed: 15 August 2016).

*Selenium open source test automation framework introduction* (2009) Available at:  
[http://www.open2test.com/QAT\\_Tech\\_Docs/Selenium\\_framework\\_introduction.pdf](http://www.open2test.com/QAT_Tech_Docs/Selenium_framework_introduction.pdf) (Accessed: 15 August 2016).

*Selenium WebDriver — Selenium documentation* (2008) Available at:  
[http://www.seleniumhq.org/docs/03\\_webdriver.jsp](http://www.seleniumhq.org/docs/03_webdriver.jsp) (Accessed: 10 August 2016).

Selukar, M. (2013) *QTP Vs SELENIUM*. Available at:  
<http://automationinqtp.blogspot.ie/2013/01/qtp-vs-selenium.html> (Accessed: 15 August 2016).

Sharma, M. and Angmo, R. (2014) *Web based automation testing and tools*. Available at:  
<http://ijcsit.com/docs/Volume%205/vol5issue01/ijcsit20140501196.pdf> (Accessed: 15 August 2016).

*Software testing - overview* (2016) Available at:  
[http://www.tutorialspoint.com/software\\_testing/software\\_testing\\_overview.htm](http://www.tutorialspoint.com/software_testing/software_testing_overview.htm)  
(Accessed: 11 August 2016).

Sommerville, I. (2010) *Software engineering (9th edition)*. 9th edn. Boston: Addison-Wesley Educational Publishers.

Walker, D. (2007) *Notes on Prototyping why use Prototyping?* Available at:  
[http://fhs.mcmaster.ca/OHPToolkit/Content/TK\\_Prototyping.pdf](http://fhs.mcmaster.ca/OHPToolkit/Content/TK_Prototyping.pdf) (Accessed: 22 August 2016).

*Welcome TestNG* (2004) Available at: <http://testng.org/doc/index.html> (Accessed: 10 August 2016).

*WHAT IS INTERPRETIVE RESEARCH?* (2009) Available at:  
<http://www.ipia.utah.edu/imps/html/research.html> (Accessed: 14 August 2016).

*What is java and why do I need it?* (2014a) Available at:  
[https://java.com/en/download/faq/whatis\\_java.xml](https://java.com/en/download/faq/whatis_java.xml) (Accessed: 10 August 2016).

*What is java and why do I need it?* (2014b) Available at:  
[https://java.com/en/download/faq/whatis\\_java.xml](https://java.com/en/download/faq/whatis_java.xml) (Accessed: 10 August 2016).

*Why do we need framework for test automation?* (2016) Available at:  
<http://www.softwaretestinghelp.com/why-do-we-need-test-automation-framework/>  
(Accessed: 17 August 2016).



## Appendix 2: Questionnaire

### Questions

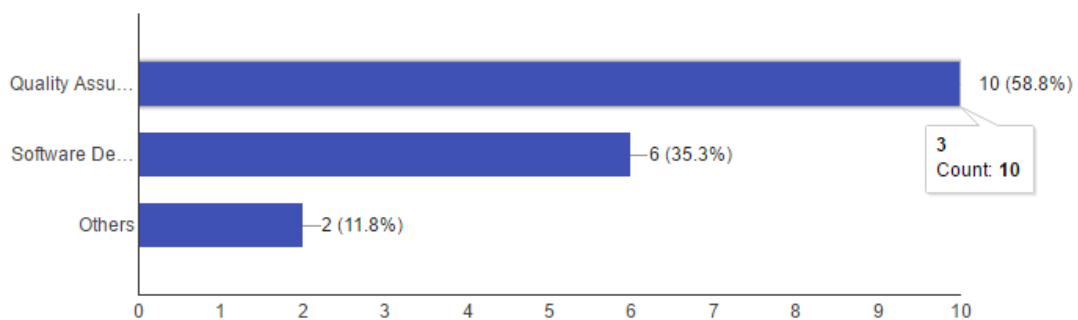
- D. You are a software professional with experience in
  - d. Quality Assurance
  - e. Software Development
  - f. Others
- E. You are having experience of
  - e. 0-3 years
  - f. 4-6 years
  - g. 7-10 years
  - h. 10+ years
- F. You are currently working in .....

1. Have you ever involved in software testing activity?
2. What do you prefer- Manual testing or automation testing?
3. If you prefer automation testing then what are the reasons to select automation over manual?
4. Have you come across any automation tool? Please specify the name.
5. What features you expect from an automation tool?
6. Do you believe that automation testing saves testing efforts and time? if yes then how?
7. Which automation framework you use for testing?
8. What type of features you expect from an automation framework?
9. Automation framework which you use is application specific or generic? How?

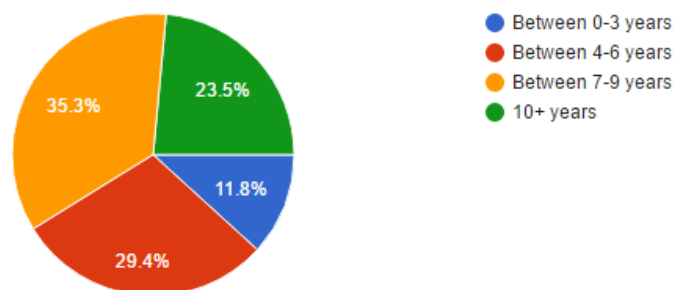
10. Which model of Automation framework you use- keyword driven, data driven or hybrid? Why?
11. What are the benefits of Hybrid model framework over other models?
12. What are the other events your framework capable to handle apart from test execution?
13. Have your framework designed in such a way that user can easily run the tests and monitor the status without any programming skills? Briefly explain.
14. How your framework has designed to add and maintain new features easily?
15. Is your framework design compatible with other tools like TestNG, Maven, SVN, GitHub etc.?

### Appendix 3: Focus Group Report

You are a Software professional with experience in (17 responses)



You are having experience (17 responses)



## Appendix 4: Survey Questionnaire Answers

S.No.	Timestamp	You are a Software professional with experience in	You are having experience	You work in organization (optional)	1. Have you ever involved in any type of software testing activities?	2. What do you prefer- Manual testing or automation testing?	3. If you prefer automation testing then what are the reasons to select automation over manual?	4. Have you come across any automation tool? Please specify the name.	5. What fe
1	2016/08/07 12:38:16 PM GMT+1	Software Development	10+ years		Yes	Both	It's very quick and can run multiple test cases in one go. Cost	Selenium	Easy to use
2	2016/08/07 12:39:05 PM GMT+1	Software Development	Between 7-9 years		No	Automation	QTP	QTP	Na
3	2016/08/07 1:34:59 PM GMT+1	Quality Assurance	Between 7-9 years		Yes	Automation testing	Gives job satisfaction.challenges your ability.learning curve	Selenium,uft	Robust,shy technolog trends
4	2016/08/07 5:32:09 PM GMT+1	Quality Assurance	Between 4-6 years		Yes	Automation Testing	Rather than doing tedious manual testing , automstion is cool	Selenium, Qualitia	Dataprovic Flexible tc
5	2016/08/07 6:09:10 PM GMT+1	Quality Assurance	Between 7-9 years		Yes	Manual		Test Complete	Reliability
6	2016/08/08 5:20:57 AM GMT+1	Quality Assurance	Between 4-6 years		Yes	Automation	easy to use.it can execute many test cases in short time in	QTP,Selenium	#NAME?
7	2016/08/08 9:18:05 AM GMT+1	Software Development	Between 0-3 years		yes	both	as i have java experience	no	selenium
8	2016/08/08 12:37:30 PM GMT+1	Others	Between 7-9 years	Wipro	No	Na	Na	Na	Na
9	2016/08/10 1:02:46 PM GMT+1	Quality Assurance	10+ years	Synerzip	Yes [ Functional testing , Performance testing]	Both	Less time , More coverage	Selenium Web Driver	Parameter
10	2016/08/11 7:30:07 PM GMT+1	Quality Assurance	Between 7-9 years		Yes mainly functional testing via manual and automation testing.	Both	To improve on skill set	Selenium	Easy to lea
11	2016/08/11 7:30:58 PM GMT+1	Quality Assurance	Between 7-9 years		Yes mainly functional testing via manual and automation testing.	Both	To improve on skill set	Selenium	Easy to lea
12	2016/08/15 3:48:07 PM GMT+1	Software Development	Between 0-3 years		yes	Automation Testing	Automated testing is more reliable,faster and accurate as compare to manual testing.	Yes, Selenium	#NAME?
13	2016/08/15 3:53:04 PM GMT+1	Quality Assurance; Software Development	10+ years		Yes tested enterprise product.	automation testing.	It reduces effort.	Webdriver	it should b
14	2016/08/18 9:35:17 PM GMT+1	Quality Assurance	Between 4-6 years	ZS Associates	yes	depends but automation is preferable	fast, less effort, more test coverage	selenium	multi brow
15	2016/08/18 9:42:27 PM GMT+1	Others	Between 4-6 years	IBM	yes, build deployment & testing	both	time saving	selenium, test complete	reliable, e
16	2016/08/18 9:58:03 PM GMT+1	Software Development	Between 4-6 years	Capita,Dublin	Yes I hav done Unit Testing ,Manual Testing and Automated Testing.	I prefer automation testing over Manual Testing.	We can reuse the tests and everyone can see the results.	Selenium/Jmeter.	Reporting,