

An Efficient Automation Framework for Testing ITS Solution Using Selenium

Ramyashree. B

MTech, Information Technology, The National Institute of Engineering, Mysore

Abstract- Intelligent transport system (ITS) requires both hardware and software support to maintain its accuracy. Hardware are test for its performance whereas software is tested for its accuracy and maintainability to support end users and customers. The proposed system creates a selenium framework for automating the test cases to be executed on the web applications. The automation framework is required to provide maintainability, productivity and to make result analysis easier. Frameworks are a set of guidelines which defines how we will structure the various components in an automation environment. Proposed system creates a data driven framework to test the web application of the above mentioned ITS.

Keywords- ITS, Selenium, Selenium Framework, Data driven framework

1. INTRODUCTION

Intelligent Transport Systems (ITS), as the name specifies- it is the transport systems with some intelligence. ITS is built upon both Hardware and Software. ITS aims to provide innovative services to different modes of transportation. It is an application of computer, electronics and communication technologies and management strategies in an integrated manner to provide traveler information to increase the safety and efficiency of the surface transportation systems. Some of the examples of intelligence in transport systems include- vehicular tracking, panic alert, Wi-Fi facility etc... ITS should be managed by Hardware like Routers, Display systems, etc. and Software like Web Applications, Firewalls, Protocols, etc. Let's take an example; Bus is the most widely used public transport system. Some systems are developed to provide intelligence to this public transport. Initially the bus was setup by a GPS device in order to monitor them remotely. Next, a Wi-Fi device was setup in the bus in order to provide a free Wi-Fi for the passengers. In the Software, An App was developed to help the passengers to track the bus and then 'Alert' option was given to the passengers in case of any emergency. Figure1 shows a simple example of ITS where in there are different Transport systems with the facility like GPS, Wi-Fi, Satellite communication, etc...

Intelligent transport systems are built on as per the requirements of the client. So, each system varies in both Hardware and Software Layouts. Once the system is built and ready to setup, it has to be tested before installation. Testing phase takes an important role before Installation and

Maintenance. In Intelligent Transport Systems, Testing should be done from two aspects, Hardware and Software.



Figure1: An example of intelligent transport systems.

Routers, GPS devices, Display systems, Inductive loop detectors etc. has to go under performance testing to ensure the correctness of the devices. Similarly, every Software, Particularly web application has to go under different types of testing like functional, Integration, System, Security, Performance, End-End testing. All these types of testing are done in order to check the correctness of the application. Every Software is developed to support a business. Every software is developed to support the business. If there is bug in software, affects the business so before we launch the software to the business it should be tested such that all the bugs should be recognized and fixed. Software testing can be done both manually and automatically. Manual Software testing involve click and check method which becomes tedious task and also require lots of time to do. To overcome this disadvantage we prefer for Automation Testing. Automation testing involves writing a code, creating framework and reusing framework for other applications of similar type.

This project aims at creating one such framework for the web application which supports ITS. The framework is created with the main intension of reusability and correctness. The framework uses selenium as its tool with extension of jar file. 'User Management' Module was automated using the framework. Validation testing, Functional testing, Integration testing and some extent of penetration testing is done using selenium tool. A research work was done regarding penetration testing with selenium

tool. The end results were evaluated and reported to the organization. Selenium is an Open source tool which is used to automate the Software testing. Selenium is a set of jar files or suite of software that are intended to automate the software based on organization needs. Selenium has four components: Selenium RC, Selenium IDE, Web Driver and Selenium Grid. Selenium Remote Control (RC) allows the user to write the automation script in any programming language against HTTP request. Selenium IDE is an UI based application which have Record and Playback option which helps the user to record the test cases and execute it. But, the disadvantages of IDE are that the validation test cases cannot be executed efficiently. Selenium Web Driver allows the user to automate the software and test whether the components are working as expected. Also, WebDriver allows the user to have friendly API that is easier to maintain than in Selenium-RC. In this project we are using Selenium WebDriver to create a framework.

2. PROPOSED SYSTEM

The main purpose of the ITS web application concentrates on remotely controlling the systems, user management and to support the employee in retrieving the information about the transport system. The module which is concentrated here is "User Management". In user management module, the user will register from the companies email address to subscribe for Wi-Fi facility inside the bus. Also, the employees can track the bus through WebApp or Mobile App. The user can register only through the web application. Users (Employees of the organization) has to enter authorized employee-id, card-id, corresponding e-mail id with organization domain name and corresponding mobile number. As soon as they register, an email will be sent on to the registered mail-id with unique link for each employee. Employee has to validate themselves by clicking on the link. Once the validation is over, then employees have full access to the app. Every time when user wants Wi-fi, he/she has to login to the application. As soon as employee logs-in, they will receive an OTP using which the employee can start using the Wi-Fi. The employees also have the facility of editing their profile. If users enter invalid credentials for 3 times, the account will be blocked and can be unlocked only by admin side web application.

The proposed system creates a selenium framework for automating the test cases to be executed on the web applications. The automation framework is required to provide maintainability, productivity and to make result analysis easier. Frameworks are a set of guidelines which defines how we will structure the various components in an automation environment. Proposed system creates a data driven framework to test the web application of the above mentioned ITS.

Advantages of Proposed System:

- Efficient Data Driven Framework.

- Automatically generates validation test results.
- Penetration test cases are executed.
- Test results will be written back to excel file.
- Reusability of code.
- Better performance.

3. RELATED WORKS

As stated in [1] Urban transportation is being transformed by mobility-on-demand (MoD) systems. One of the goals of MoD systems is to provide personalized transportation services to passengers. This process is facilitated by a centralized operator that coordinates the assignment of vehicles to individual passengers, based on location data. However, current approaches assume that accurate positioning information for passengers and vehicles is readily available. Formally, we solve a batch assignment problem that routes vehicles at obfuscated origin locations to passenger locations (since origin locations correspond to previous drop-off locations), such that the mean waiting time is minimized. Our main contributions are two-fold. First, we formalize the notion of privacy for continuous vehicle-to-passenger assignment in MoD systems, and integrate a privacy mechanism that provides formal guarantees. Second, we present a polynomial-time iterative version of the Hungarian algorithm to allocate a redundant number of vehicles to a single passenger. This algorithm builds on the insight that even during peak rush hour there are unoccupied (redundant) traveling vehicles.

Paper [2] reports that, We devise an inexpensive and intuitive system for bus route navigation for locales where public transportation may serve as a prevalent mode of commute but where technologies that make arrival predictions through tracking vehicles in transit through GPS or other means do not exist. These systems typically require real-time monitoring of traffic variations. We provide a personalized approach where in a world of pervasive smart phone use, users may take advantage of sensor data to learn and personalize their bus routes, and alert them on time when a bus stop is approaching. We accomplish this through the development and implementation of two algorithms: 1) turn detection using on-board compass sensor of a smart phone, and 2) characterizing road segments in terms of turns and thereby predicting approaching bus stops. We conduct field experiments on a route with four selected bus stops in the town of Chapel Hill. Results show that the accuracy of turn detection and detection of approaching bus stops are 95.7% and 83%, respectively.

The selenium framework which is created here data driven. So we are referring below papers to create a framework. As per [3] Software testing is considered to be the most important step in Software Development Life Cycle. The main objective of the testing process is to compare the obtained results with those of expected by the end user of the software. Test Automation simplifies the work of tester by automating the execution of test scripts with the use of

special software. This paper focuses on the use of Selenium WebDriver to test a web application and to demonstrate the use of this tool in combination with other tools like the Maven, TestNG, etc., for more easier approach to testing and to improve the quality of testing process.

Paper [4] states that, The challenges met during the software projects fall into any number of categories. The development and the technical solutions bring about technical challenges, but the situations one is confronted with, may also be sociological, psychological or managerial in nature. Without any knowledge in the field of social sciences, the programmers, testers and managers might interpret the social aspects of the project improperly, and such interpretations lead to the inability to fully understand the problem and, ultimately, to inefficiency in the decision-making process. Furthermore, solid knowledge of theories in the area of the social sciences is required for a better understanding of both the context in which the application runs and of the final users who will use the developed project. The understanding of and the involvement in a software acceptance testing (SAT) project, requires the combination of multiple theories and principles from different disciplines.

According to [5] Software testing is a key process that ensures a reliable quality product, and like other activities in the development process, has a wide range of tools available, but still requires a lot of human work, where the final quality of the software can be impacted directly by several factors. In this sense, this study aims to identify the human factors (cognitive, operational and organizational) present in the test process and to define the influence of these factors during their execution. Thus, the article presents a study with quantitative methods and techniques of the survey type, in which 112 professionals from the test area participated in 17 Brazilian states. The results provide a set of human factors that correlate with the final quality of a software product or service.

4. DATA DRIVEN FRAMEWORK

Let us consider an example where in a web page has two text boxes. One for user name and the other for password. Both the text fields accepts ten characters which must include letters, numbers and special characters like @ and . We have to test for the validation of text boxes. That is, we have to functional test the text boxes for its accuracy of accepting input. To test these validations, we must create test data. Test data must include all the combinations of input. We have to pass on each of these test data to text boxes. Typing each and every test data into text boxes manually is a tedious task. So we have to automate this work which is called as data driven framework.

We need automation framework for maintainability, productivity and to make result analysis easier. Frameworks are set of guidelines which defines how we will structure the

various components in the automation environment. [6]. these components include object repository, test data, functions, reports and batch execution scripts.

Figure2 shows different components of framework. All these components work together to form an efficient framework. Driver component includes, how the scripts are executed in batch and what are the setups needed for the execution. Scripting components include, how all the key components are structured in automation framework. Reporting components include how to get the test results.

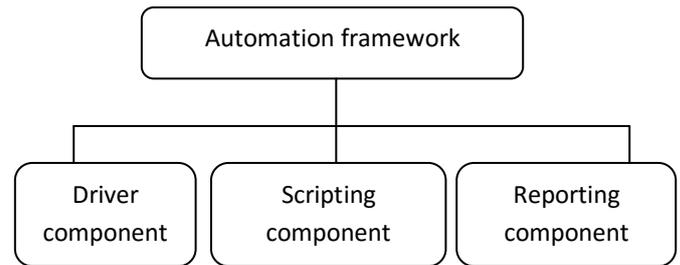


Figure2: Automation Framework Components

In Data Driven framework test case logic resides in test scripts, the test data is separated and kept outside the test scripts. Test data is read from the external file like excel files, text files, CSV files and database. These are loaded into the variable inside the test scripts. Variables are used both for storing input values and verification values. Since the data is stored outside the script, if any changes in the test data, we would need to change just one datasheet and all other test data will fit into execution. This avoids a huge maintenance effort using this framework. Test data can be changed at one central place and there is no need to modify the scripts. Changes to the test data does not affect the test scripts and test cases can be executed with multiple sets of data.

5. IMPLEMENTATION

The Implementation phase of the project is where the detailed design is actually transformed into working code. Aim of the phase is to translate the design into a best possible solution in a suitable programming language. This chapter covers the implementation aspects of the project, giving details of the programming language and development environment used. It also gives an overview of the core modules of the project with their step by step flow.

Prerequisites for implementation of this framework requires excel test data sheet to be kept ready. Output excel file will be created internally by the program. All the components should be located using id, xpath, css etc...

Algorithm1: FileHandle()

1. Create a file with output stream.
2. Print "creating file"

```
3. Create workbook copy- writeblesh
4. Create sheet with sheetname and offset value
5. for (int i = 0; i < Norows; i++)
   for (int k = 0; k < Nocol; k++)
     inputdata[i][k]=sheets.getCell(k,i).getContents()
     Label l = new Label(k, i, inputdata[i][k]);
     Label l2 = new Label(3, 0, "actual output");
     Label l3 = new Label(4, 0, "status");
     writablesh.addCell(l);
     writablesh.addCell(l2);
     writablesh.addCell(l3);
6. end for
7. end for
8. end FileHandle
```

Algorithm1 gives the steps to perform file handle operations. That is, using this algorithm the files are opened and read. Also, the test data are copied on to the output file.

Algorithm2: Provide_data()

```
1. open a file in input stream
2. open workbook copy in input stream excel file
3. get sheet in work book copy
4. read number of rows and columns
5. testdata[][]= String[no. of rows-1][columns]
6. count=0
7. for(int i=1;i<Rows;i++)
   for(int j=0;j<Columns;j++)
     Cell Excell = Exsheet.getCell(j,i);
     Testdata[count][j]=Excell.getContents();
8. end for
9. count++;
10. end for
11. end Provide_data()
```

Algorithm2 gives how the data is stored on to a variable from fetching the test data by excel. All these data are stored in the form of array variable and these array variables are responsible for sending keys on to the text boxes and verified test results I,e 'Pass' or 'fail' will be written back to the output stream excel file.

6. CONCLUSION AND FUTURE WORK

This paper creates an automation framework for ITS web applications using selenium. The main intension of automation is to test the application for its accuracy. This framework is called as "Data driven framework" which fetches the test data from an external entity like excel or database and stores it in a variable and passes it to web elements. Also, the above mention algorithms verify the results and write it back to the excel file. This framework can be reused for other similar applications hence maintaining productivity and maintainability. Further, this framework can be extended to perform Penetration testing like XSS or

SQL Injection. Also, in future work they can concentrate on collaborating this framework with other hybrid frameworks.

ACKNOWLEDGEMENT

I would like to thank Dr.Vidya Raj.C, Professor at NIE, Mysore for her guidance and support in doing case study of the above paper.

REFERENCES

- [1] "Privacy-Preserving Vehicle Assignment For Mobility-On-Demand Systems" (Published in: Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on 24-28 Sept 2017)
- [2] "Characterizing Road Segments Using Compass Sensors To Predict Approaching Bus Stops" (Danila Chenchiky, Jia Cheny, Stephen Yany, Shahriar Nirjon Undergraduate Mobile Computing Systems Research Group Department of Computer Science University of North Carolina at Chapel Hill Email: fchenchik, jiac, sjyan, nirjong@cs.unc.edu)
- [3] "Testing Using Selenium Web Driver" (Published in: Electrical, Computer and Communication Technologies (ICECCT), 2017 Second International Conference on 22-24 feb 2017)
- [4] "Applying Coding Systems In The Process Of Testing Software Applications" (2017 14th International Conference on Engineering of Modern Electric Systems (EMES))
- [5] "The Influence Of Human Factors On The Software Testing Process: The Impact Of These Factors On The Software Testing Process" Information Systems and Technologies (CISTI), 2017 12th Iberian Conference on
- [6] "Test Automation using Selenium WebDriver"- Navneesh Garg, LeoTech Press