OSTBAYERISCHE
TECHNISCHE HOCHSCHULE
REGENSBURG

Ostbayerische Technische Hochschule Regensburg
University of Applied Sciences Regensburg
Faculty of Computer Science and Mathematics

Assignment 1

DevOps Principles and Practices

Tools for Version Control and Test Automation

Presented by: Tianhua Dai
E-Mail: tianhua.dai@st.oth-regensburg.de
Student Number: 3111914
Study Subject: Business Informatics

Primary Supervising Professor: Dr. Abhijit Sen
Team: Jiawei Tang, Yue Yuan, Tianhua Dai

Submission Date: 22.05.2020

Contents

1 Introduction **1**

2 Tools for Version Control **2**
2.1 GitHub 2
2.2 Tortoise SVN 5
2.3 Conclusion 8

3 Tools for Test Automation **10**
3.1 TestNG 10
3.2 SoapUI 12
3.3 Conclusion 14

List of Figures

1 Create new repository and get the clone link . . . 3
2 Pic Git Clone . . . 4
3 Commit the local changes and push them online . . . 4
4 Pic New Branch . . . 4
5 Merge the changes and update the project . . . 5
6 Create new repository in VisualSVN . . . 6
7 Link to repository using Tortoise SVN . . . 7
8 Set authorities and checkout . . . 7
9 Commit with changes and update . . . 8
10 Pic SVN Log . . . 8
11 Different test cases . . . 11
12 Pic Test Result . . . 12
13 Create a completed test case . . . 13
14 Set the assertion and results . . . 14

List of Tables

1 Version Control Systems 2
2 GitHub description 3
4 Tortoise SVN description 6
5 TestNG description 10
6 SoapUI description 12

1 Introduction

DevOps is a concept of software culture, and there are different understandings and interpretations of the scope of this culture, especially automation practices. It aims to seamlessly extend Development to the Operation stage, break the barriers between the two in the traditional software process, simplify all aspects of the software life cycle, and promote the healthy, fast and continuous delivery of software. [1]

The concept of DevOps was introduced as early as 2009. With the increasing development of new technologies and automation tools, it has been valued by enterprises in past years rapidly and started to be widely practiced. Research shows that the continuous, efficient and high-quality delivery of DevOps depends on the support of a high degree of automation tools, and automation is also the key to obtaining rapid feedback. There are number and variety of tools for implementing DevOps functionalities, which involve multiple stages through software development. This assignment is aimed to research tools for Version Control and tools for Test Automation.

The structure of the remaining chapters of this assignment is organized as follows. The usage of selected two tools for version control would be introduced in chapter 2. And chapter 3 demonstrates the practices with selected tools for test automation. Each tool would be described in table form with specific features. After each description the implementations with examples would be explained using sample screenshots. At last each tool would be concluded in a subsection.

[1] [SL18], S. 5

2 Tools for Version Control

Version control is a system that records changes in the content of a file or set of files over time so that man can recall specific versions later. The files that contain the software source code would be usually version-controlled, but in fact, version control is just for any file type, for example a thesis or report using Word. [2] Version control systems bring also some good benefits like compare files, identify differences, and merge the changes if needed. Another great use of versioning is keeping track during the whole software development process. [3] Basically, the version control systems can differ in 3 types. (see table 1)

Typ	**Representative product**	**Website**
Local	RCS	`https://gnu.org/software/rcs`
Centralized	SVN	`https://subversion.apache.org`
Distributed	Git	`https://git-scm.com`

Table 1: Version Control Systems

The local version control system is relatively simpler from a basic point of view, which has a simple database that keeps all the changes to files under revision control. In other words, the local type is now a basic functionality for version control. Systems from the other 2 types, such as Git and SVN, have realized many more functionalities when using specific tools. So, the following contents of this chapter describe the usage of 2 representative tools, through which software release could be delivered faster and more efficiently.

2.1 GitHub

2.1.1 Introduction

It is established that Git is a version control system. But what makes GitHub-A Tool for Version Control System-so special? Because it is free and open-sourced. GitHub is a web interface for git repositories as well as a development platform for developers. It's built as a virtual community so that developers store their projects online and connect with other like-minded people. [4] Therefore, the following subsections demonstrate the description with some important attributes and then the performing by a sample.

2.1.2 Description

In this section the attributes of GitHub would be demonstrated in the following table. (see table 2)

[2] [SC14]
[3] [Aza13]
[4] [Bro19]

Attribute	Contents
Name	GitHub
Purpose	GitHub is a code sharing and publishing service based on web platform. It is also a Git repository hosting service, which provides the user-friendly GUI and many its own features.
Website	`https://github.com`
Version	-
Date	-
Features[5]	1. Support all functionalities by Git 2. Access control and collaboration features 3. Fork, pull request and merge
Requirements/ Compatibility	Anyone browser
Cost	free

Table 2: GitHub description

2.1.3 Sample

My sample is a java project using features of TestNG and Selenium, which tests the login method in a web game. This project has been published in GitHub repository (`https://github.com/Tianhua-Joseph-Dai/testAutomation_ddop`). The following content could be divided into 3 parts in order to describe the practice using GitHub.

1. Preparation: A new repository would be established in website at first. And then the link of this repository is available. (see figure 1)

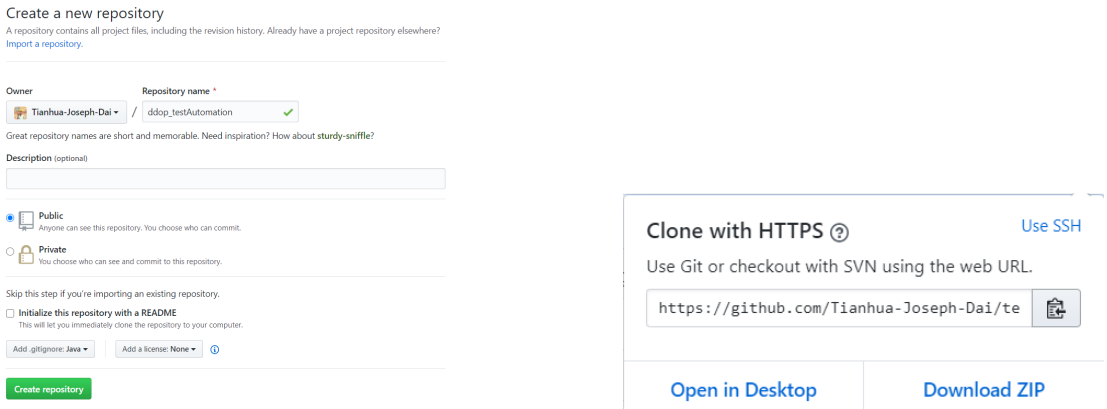

Figure 1: Create new repository and get the clone link

[5] [Fin12]

At the same time a local folder has been prepared for cloning the remote project by using Git Bash command. (see figure 2)

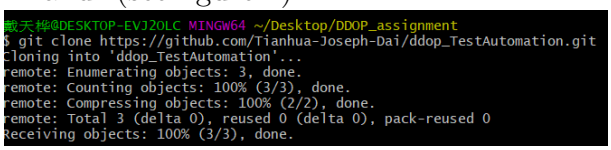

Figure 2: Clone into the local folder

2. Sample: Firstly, a new branch (test/openGoogle) was created in local, which matches the functionality of program to add. The advantage of this operation is to consider version control. After the add functionality and the commit with some infomation the project would be pushed to the web platform in GitHub. (see figure 3)

Figure 3: Commit the local changes and push them online

At last the new branch would be also visible online in the repository. (see figure 4)

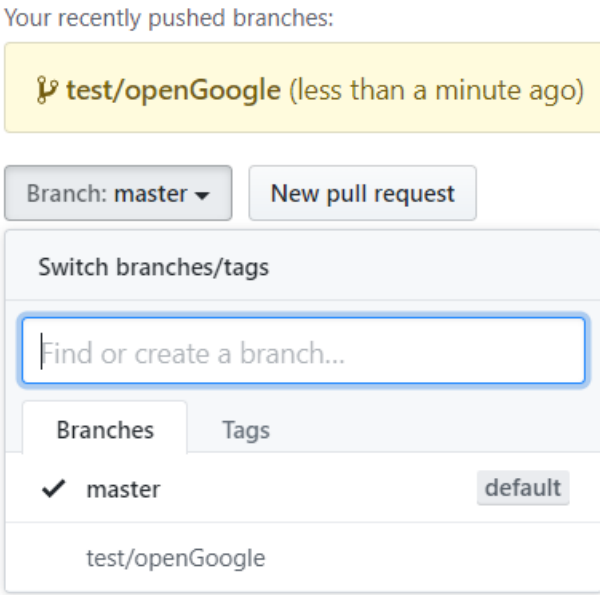

Figure 4: Update online

3. Maintenance: After any modification a new branch would be established as a new method in local repository. Therefore, the changes from other branches should be merged into master and make the whole project UpToDate. (see figure 5)

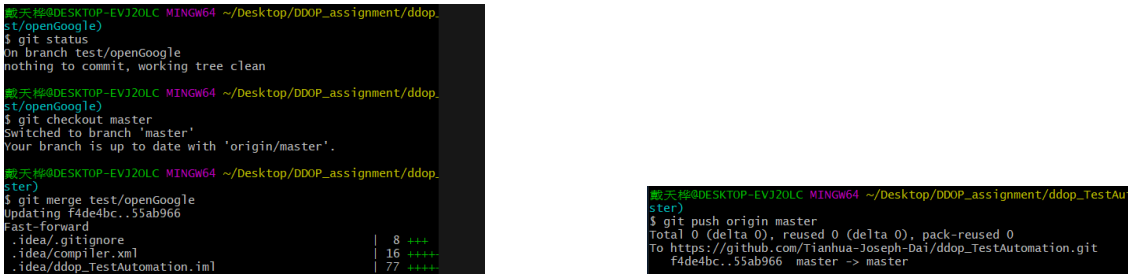

Figure 5: Merge the changes and update the project

2.2 Tortoise SVN

2.2.1 Introduction

This section is about another type of version control system, which is different from Git. Subversion is a centralized version control system, which now is developed as a project of the Apache Software Foundation. It has also been developing to a rich community for developers to widespread and collaborate.
Similar to Git SVN is also or just a command-line client. Tortoise SVN is built on the basis of SVN, which is performed as a windows shell extension. Tortoise SVN provides a easy user GUI for managing different versions of projects.

2.2.2 Description

The following table lists the attributes of Tortoise SVN. The content in the following table is referenced from the Tortoise SVN website. (see table 4)

Attribute	Contents
Name	Tortoise SVN
Purpose	Tortoise SVN is used for revision/version/source control.
Website	`https://tortoisesvn.net`
Version	1.13.1
Date	2019/11/01
Features	1. Cover functionalities of SVN 2. Support for all subversion protocols 3. Integration with issue tracking systems 4.Support from multiple language packs
Requirements/ Compatibility	1. Windows 7 or later 2. A SVN server
Cost	free
Note	1. Tortoise SVN is not available for UNIX-OS yet. 2. SVN server must be installed in advance.

Table 4: Tortoise SVN description

2.2.3 Sample

The same project would be practiced in SVN system. It would be also divided into 3 parts to describe the implementation using Tortoise SVN.

1. Preparation: Because Tortoise SVN is a SVN client, a SVN server is needed as requirement. VisualSVN Server (`https://visualsvn.vom/server`) is used to demonstrate in this sample. At first a new repository should be created. (see figure 6)

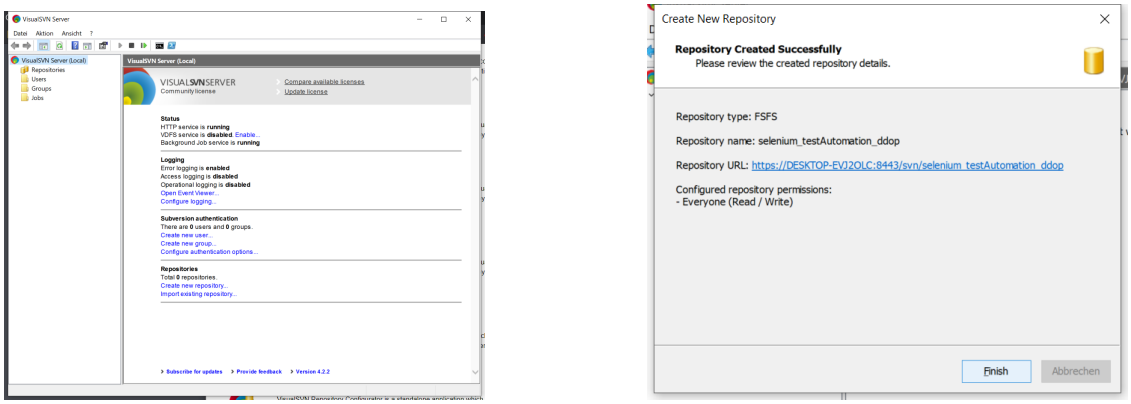

Figure 6: Create new repository in VisualSVN

Then open any windows explorer, right click, Tortoise SVN features are visible in the menu. Select the Repo-browser and enter the repository link. At last it succeed with

access the repository. (see figure 7)

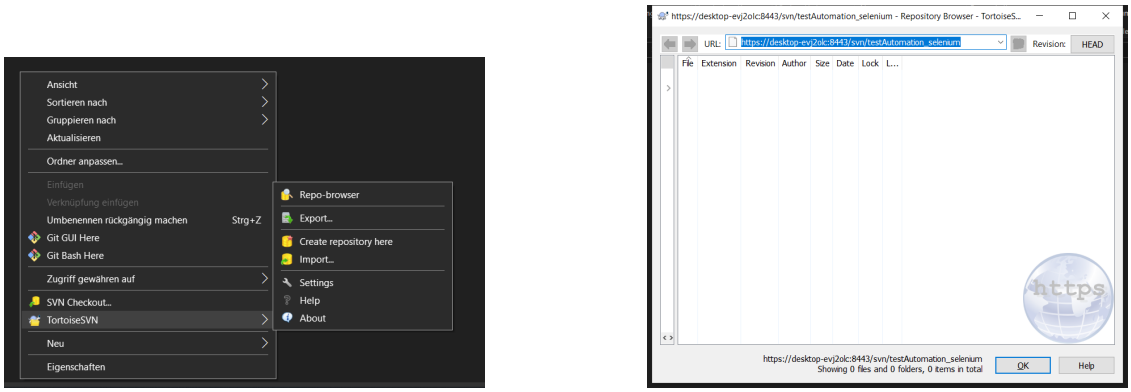

Figure 7: Link to repository using Tortoise SVN

2. Sample: SVN has a relatively strict access control so that the user must firstly determine the authorities and set in the repository. After that the connect would be established by checkout, which is like the switch branch feature from Git. (see figure 8)

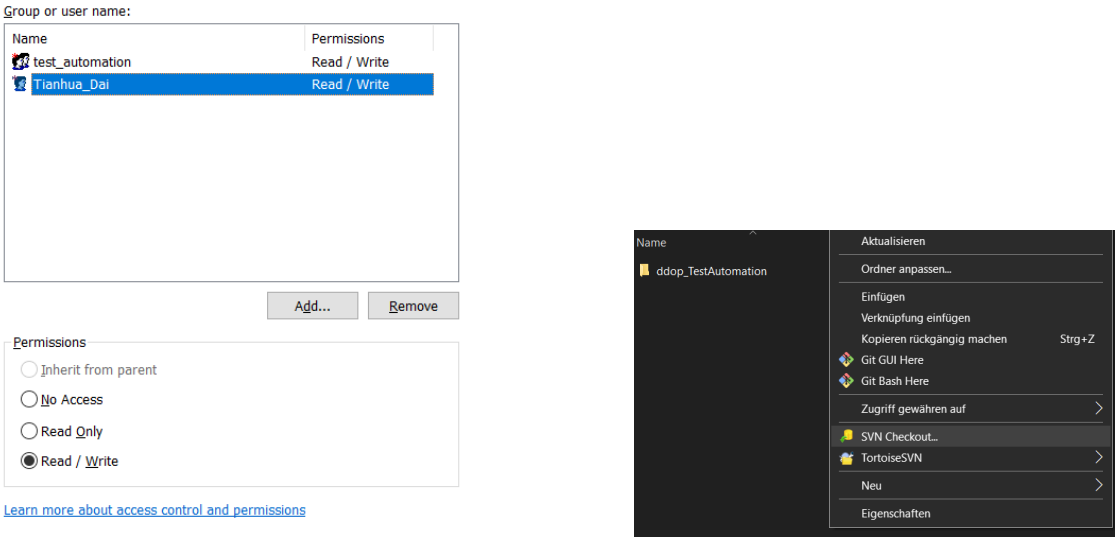

Figure 8: Set authorities and checkout

User can right click in the local repository to commit the modification. The revision number would be increased after a successful commit. At the same time the project is also available in browser repository. (see figure 9)

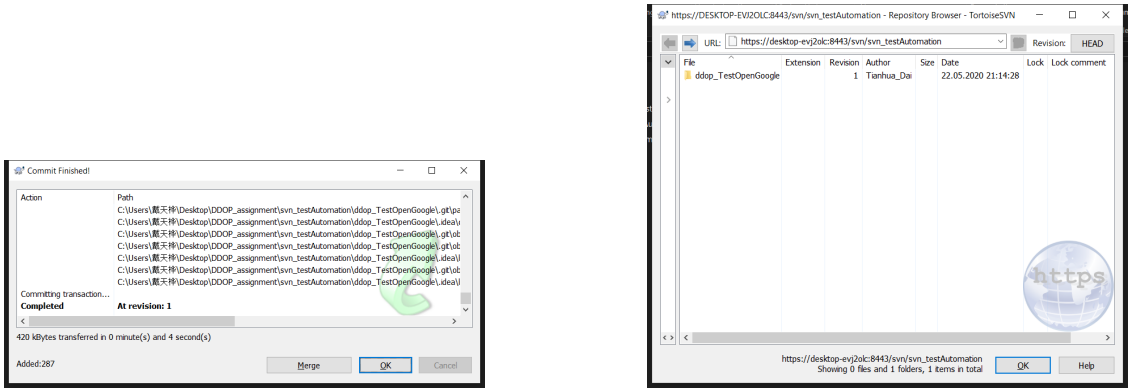

Figure 9: Commit with changes and update

3. Maintenance: Furthermore, Tortoise SVN provides a log feature which support many useful functionalities to maintain the project. (see figure 10)

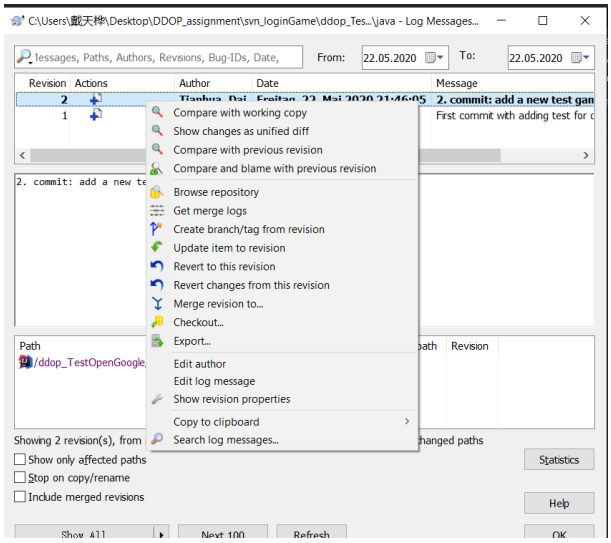

Figure 10: Log overview

2.3 Conclusion[6]

These tools support the implementation of the functions for version control systems in order to increase the work efficiency. And with the continuous development the most functionalities of version control systems have been trending to one standard. GitHub is a free, open-sourced web development platform for git repositories. It handles project versioning from some to very large projects efficiently and quickly. The biggest difference between distributed and centralized is that, on the one hand developers can commit locally, and on the other hand each developer can copy a complete repository on their own PC through git cloning. Developers can also do their works without the internet.

[6] [Che20]

By contrast SVN manages data that changes over time. This data is kept in a central repository. So this archive could be seen as a normal file server, but it remembers every changes. According to this restore to an old revision or browser through the history is allowed. Besides the core of centralized management is the server, where all developers must get code from the server at first. All version information is placed on the server. Without the server, developers basically can't work.

3 Tools for Test Automation

3.1 TestNG

3.1.1 Introduction

TestNG is one of most popular testing frameworks, which inspired from JUnit and NUnit. It is designed to be competent for all kinds of testing cases. In addition, TestNG is supported by a number of plugins and tools so that developers can make their test implementation easier and more efficient.

3.1.2 Description

In this section the attributes of TestNG would be demonstrated in the following table (see table 5). The content is referenced from the official TestNG website.

Attribute	**Contents**
Name	TestNG
Purpose	TestNG is used for tests in unit, functional, integration etc..
Website	`https://testng.org/doc`
Version	7.1.0
Date	2019/12/27
Features	1. Support almost all kinds of tests 2. Test with multithread safety 3. Supported by variety of plugins and tools
Requirements/ Compatibility	JDK 7 or higher
Cost	free
Note	-

Table 5: TestNG description

3.1.3 Sample

A test can be easily built with the TestNG annotations, some test logics and configuration in a xml file. The sample to demonstrate is about login in a web game with different information. (see listing 1)

```
1  public class gameAutoLogin {
2      @Test(dataProvider = "dataProvider")
3      public void autoLogin(String user_email, String user_pwd, String expectedResult) throws
           InterruptedException {
4          System.setProperty("webdriver.gecko.driver", "tools/geckodriver.exe");
5          WebDriver webDriver = new FirefoxDriver();
6          webDriver.get("https://naruto.oasgames.com/en/serverlist/s763");
7          webDriver.findElement(By.name("user_email")).sendKeys(user_email);
```

```
8           webDriver.findElement(By.name("user_password")).sendKeys(user_pwd);
9           webDriver.findElement(By.xpath("//a[@onclick='ajax_login();']")).click();
10
11          if(expectedResult != "") {
12              WebDriverWait wait = new WebDriverWait(webDriver, 3);
13              wait.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//p/span[@id='
                    login_username_id']")));
14          }
15          String result = webDriver.findElement(By.xpath("//p/span[@id='login_username_id']")).
                getText();
16          Assert.assertEquals(result, expectedResult);
17      }
18
19      @DataProvider
20      public Object[][] dataProvider() throws InterruptedException {
21          return new Object[][] {
22                  new Object[] {"wiidai01@163.com", "wii1350", ""},
23                  new Object[] {"wiidai02@163.com", "wii1351", "Wrong information"},
24                  new Object[] {"", "wii1351", "Wrong username"},
25                  new Object[] {"qwerdai01@163.com", "wii1351", "User does not exist"}
26          };
27      }
28  }
```

Listing 1: Login test

The selenium framework supports the necessary web functionalities in this program. And this test will run in several cases (test cases 11) through data provider. (Result of implementation)

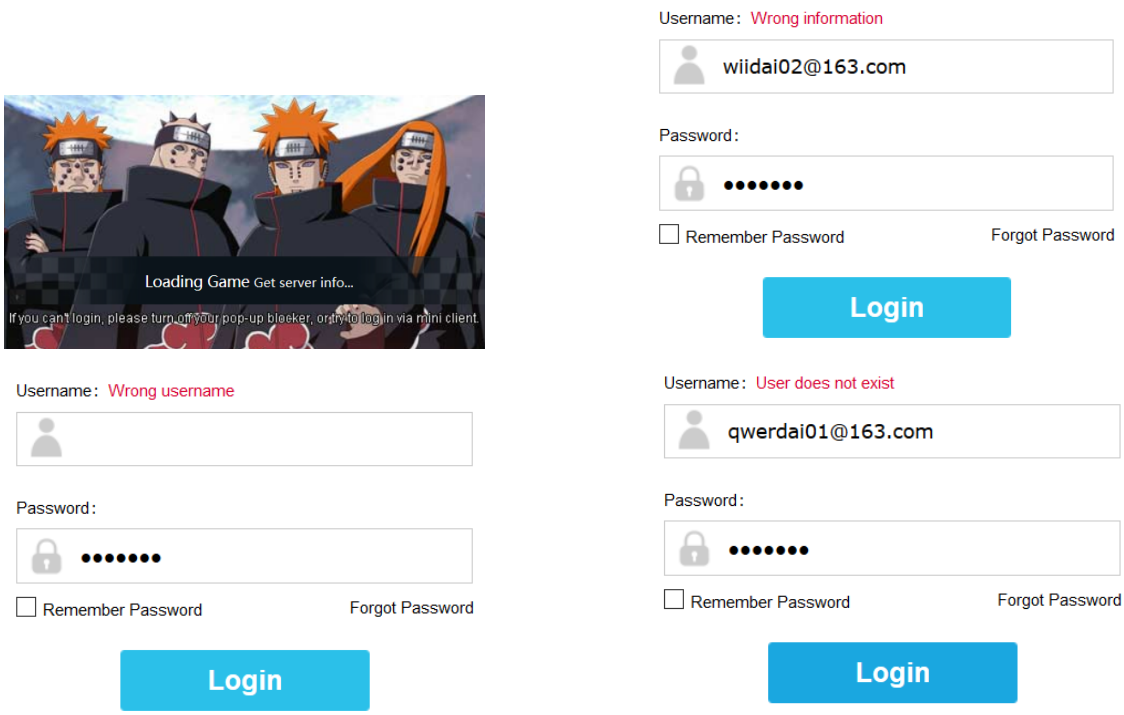

Figure 11: Different test cases

Figure 12: Result of test

3.2 SoapUI

3.2.1 Introduction

SoapUI is an open source testing tool, which covers a variety of tests for web services. It has an easy-to-use graphical interface and supports all standards of protocols and technologies of web.

3.2.2 Description

The attributes of SoapUI would be demonstrated in the following table. The following content is referenced from the official SoapUI website.

Attribute	Contents		
Name	SoapUI		
Purpose	SoapUI is used for web services testing for SOA and REST. Main functionalities cover exploring, mocking, load, functional and security testing for web services.		
Website	`https://www.soapui.org`		
Version	5.5.0		
Date	2019/02/12		
Features	1. Multiple web service tests 2. Web services mocking and inspection 3. Data-driven tests and test automation 4. API monitoring, HTTP recording		
Requirements/ Compatibility	Windows	Windows XP or later Java 7	
	UNIX	Ubuntu, Red Hat, Fedora, etc. Java 7	
Cost	SoapUI Open Source	free	
	SoapUI Pro	Fixed License	€640/year
		Floating License	€4,100/year
Notes	-		

Table 6: SoapUI description

3.2.3 Sample

The sample to practice is a test with several steps, which finally get the TV program information from a certain channel on a set date. At first a SOAP or REST project would be created by inputted URL. Then run the requests to find out the parameters which are necessary for the test suite. After several processes a test suite with the test case was generated, that to search TV program from Shanghai East TV Channel. (see figure 13)

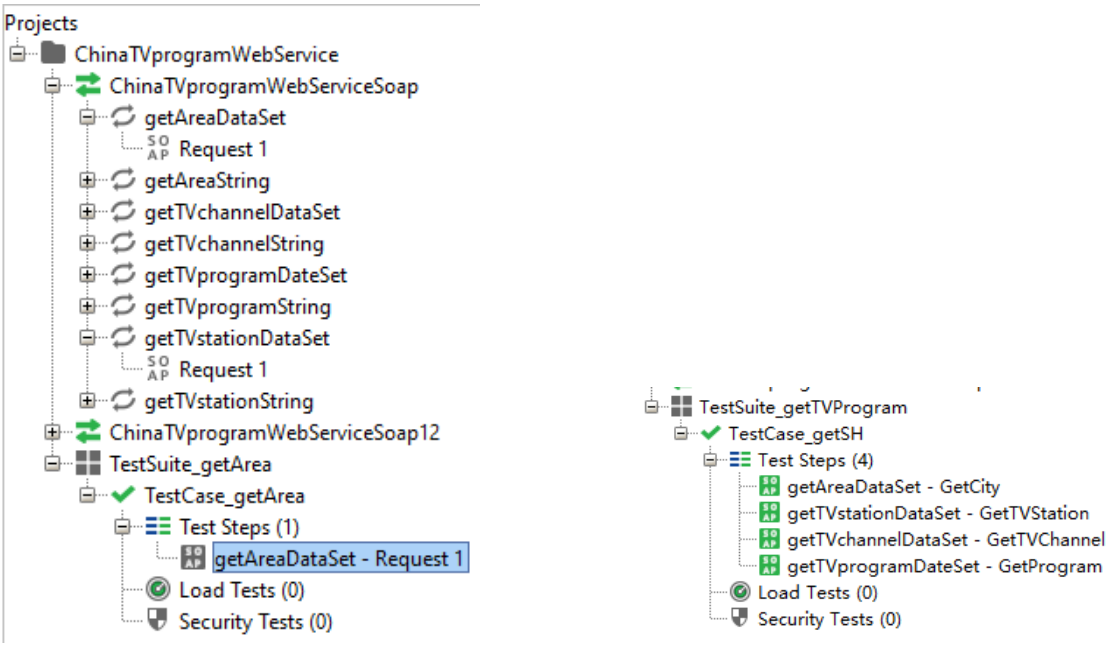

Figure 13: Create a completed test case

Then the xml file of each request must be added by parameter. (see listing 2)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://
    WebXml.com.cn/">
   <soapenv:Header/>
   <soapenv:Body>
      <web:getTVchannelDataSet>
         <web:theTVstationID>44</web:theTVstationID> <!--44: Shanghai East TV Channel-->
      </web:getTVchannelDataSet>
   </soapenv:Body>
</soapenv:Envelope>
```

Listing 2: Request with parameter

Furthermore a important step is to generate a assertion to each request. Through the assertion the result could be detected easier. (see figure 14)

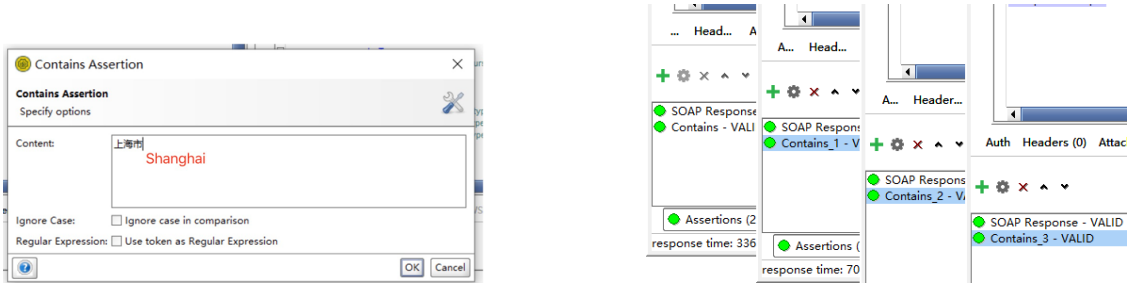

Figure 14: Set the assertion and results

3.3 Conclusion

It is established that web services are called by programs, which generally do not provide interfaces for end user or tester directly. Underlying, call relationships as well as protocols should be taken into consideration. SoapUI is a such a tool so that user can complete tests in SoapUI with simple operations while it reduces workload. On the other hand TestNG provides developers multiple test functions while they are developing. The program could be tested for different purposes. On the one hand it is supported by variety of plugins and tools, on the other hand TestNG will be integrated in different projects as well. It could be a good element for a DevOps toolchain.

List of References

[Aza13] Irma Azarian. *A Review of Software Version Control: Systems, Benefits, and Why it Matters*. June 14, 2013. URL: https://www.seguetech.com/a-review-of-software-version-control-systems-benefits-and-why-it-matters/.

[Bro19] Korbin Brown. *What Is GitHub, and What Is It Used For?* Nov. 13, 2019. URL: https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/.

[Che20] Xuyuan Cheng. *Introduction to SVN and Git, Differences, Advantages and Disadvantages*. May 15, 2020. URL: https://zhuanlan.zhihu.com/p/141260729.

[Fin12] Klint Finley. *What Exactly Is GitHub Anyway?* July 14, 2012. URL: https://techcrunch.com/2012/07/14/what-exactly-is-github-anyway/.

[SC14] Ben Straub Scott Chacon. *Getting Started - About Version Control*. 2014. URL: https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control.

[SL18] He Zhang Shanshan Li Shenghui Gu. *Automated Support Tools of DevOps*. June 10, 2018. URL: http://softeng.nju.edu.cn/tech_reports/Tech-Report-devops4tools-2018.pdf.