Intelligent Test Platform for Power Metering System Software

Tao Liu*, Shaocheng Wu, Yue Ma, Jie Zhao and Hefang Jiang
Shenzhen Power Supply Bureau Co, Ltd, Shenzhen, China

*Corresponding author email: 1513629216@qq.com

Abstract. In order to meet the requirements of the operation and maintenance personnel for the power metering system software, when updating the software version, they can independently, quickly and effectively control the quality of the software, an intelligent test platform is designed. Based on the platform, the software operation and maintenance personnel can simply select the corresponding records from the test scripts, test cases and automatic test tools in database debugged by the third-party test team, create the test plan, click the execute button, the platform will automatically connect the selected test tool, intelligently generate a set of executable command lines, to complete the functionality or performance test of the software using the selected test script and test case, and to create the reports of software defects, then a software development team repairs the defects and then the personnel performs regression testing until the new version meets the requirements.

1. Introduction

As more power producers and users trade electricity through market-based mechanisms rather than via the old off-line system, the new features of online Power Metering System (PMS) have been instantly designed and released into the production[1]. The system has become so complex that it is very hard for the software operation and maintenance personnel to control its quality in Shenzhen Power Supply Bureau Co, Ltd. Therefore the Bureau has changed the strategy to improve the situation, i.e. the third-party organizations have been introduced into the business to control the quality of the system at the first in means of series of automatic test tools such as JMeter, LoadRunner and WinRunner for performance and functionality and so on[2-12]. However, as long as the testing task has been fulfilled, the maintenance personnel in the Bureau couldn’t frequently request the people from the third party to test the software over and over again because of the finance or other reasons.

Therefore, an intelligent test platform has been designed and developed to control the quality of the Power Metering System when the software will be updated. Based upon the platform, professional testers from the third-party organization can record, modify, adjust and implement the number of test scripts and test cases using the automatic test tools such as LoadRunner, JMeter and SonarQube for the tests of functionality, performance or security vulnerabilities of the PMS, then the test scripts and cases are saved into the database of a version control by the testers.

As long as the PMS is planned to update a new version, the operation and maintenance personnel of the PMS in the Bureau can do all of the tests by the platform and get the test result analysis reports by “One-Touch Style” without the consideration of both tester’s assistance and the details in test scripts. If the results had revealed any bugs or issues, they could be submitted to software developing team to fix them and regression tests were carried out until it will meet the requirement of the version upgrade. Based on the automatic test platform, the personnel responsible to maintain the software system can sufficiently control the quality of new version of PMS, reduce the daily errors of software maintenance and improve the work efficiency.
2. Logic Architecture of the Platform
The platform is consist of two parts: management and service. The latter includes intelligent test, version control, vulnerability scanning of source code and system administration. The intelligent test as a key part of the platform will implement the series of smart operations for different tests of the PMS according to the test scripts and cases debugged by the professional testers in advance. The version control is responsible for the file version control involving test scripts, test cases, profile settings, remote servers or virtual machines etc. System administration manage users and their roles and privileges (shown in Figure 1).

3. Functionality Description
The intelligent test platform\cite{13,14} is designed for testing and managing subsystem in order to expand the test services in future flexibly. The main functions will be shortly described in this section. A real example will be mentioned in detail in the next section.

3.1. System Administration
According to the duties of users, the different roles and privileges can be setup for the test team of third-party agency, software operation and maintenance personnel and system administrators.

3.1.1. User privilege management. Including the role and user management, it is the smallest unit to assign function permissions to users through roles. The role of management is to give the privilege of roles based upon user’s duty, such as weather. The user gets the permission to open a menu or button in a Web page. The user management will create, modify or delete the users to access the platform.

3.1.2. Data privilege management. It is the smallest unit to assign data privilege to users through roles, for instance, if the data in the same Web page can be given to be displayed or not?

3.1.3. System audit. Record the key information for users to use the platform, which includes the login to the platform, deleting or creating a record and so on. The audit log files in database can be moved to other place for back-up in predefined period that can be set up by a system administrator.

3.2. Facility Management
The functional module creates records in the database for the servers or virtual machines where the automatic test tools such as JMeter, Forerunner, Gunrunner are installed. It will be more easy for the users to select the records when they create test plan. The record could be modified, searched or deleted by system administrator.

3.3. File Management
The testers from the third party are signed for the permission to access the module to put the files of test scripts, test cases and profile settings into Gigabit, a version control software. The files will be used to create a test plan (Figure 2).
3.4. Version Control
The module will control the version of source codes like JSP, test documents and vulnerability scanning results. The system maintenance personnel by the module can directly get the statistical information about source codes and their version, the contents and version of test scripts, test cases, profiles and test reports, the contents and version of vulnerability scanning reports.

3.5. Test Plan Management
The group of test plans is necessary to be created before the intelligent test is implemented. The information of a plan includes test script and test case with proper version from the database of Gigabit. The plan could be modified, searched and deleted.

3.6. Intelligent Test
As long as the test plans exist in database, the operation and maintenance personnel of PMS can simply select a proper plan from drop-down list of the plans, click the “Start Test” button, the intelligent test will be implemented automatically.

3.7. Vulnerability Scanning
Through this module, the system operation and maintenance personnel can obtain and upload all or part of the software source codes from Gigabit repository, and the test platform automatically scans the codes to generate the scanning results, which can be directly viewed by the system operation and maintenance personnel or software development engineers if necessary.

4. Application Case
A powerful server with 4X32GB RAM, 4 as 2.4TB is used to install the seven virtual machines Firmware of Cantos 8 64 bit where the software is separately installed, including the intelligent test platform with Tomcat, PostgreSQL database, Gigabit version control software, automatic test tools including JMeter, Gunrunner, Forerunner and so on.

4.1. Environment Settings
The intelligent test platform is designed based on the secondary development of JMeter, Gigabit, Soberness open-source tools as well as the commercial automatic test tools like Gunrunner, Forerunner and so on. The platform can satisfy the one-touch intelligent test task for the high quality control of PMS updating.

JMeters5.4 service is a compressed file of jar, which is transferred to a directory of the server without decompression. At the same time, there is a configuration file named application.properties, which is placed in the same directory with jar file, and then the service is configured and started. At the same time, you need to install JMeter together, which is installed under / opt. The file structure is as follows:

```
# --- system settings -----------------------------------------------
server.port=port number
spring.MVP.view.suffix=HTML
spring.servlet.path.prefix=class path/templates/
spring.web.resources.static-locations=file:/opt/test/class path/static/
# --- JMeter--------------------------------------------------------
jmeter.base.home=/opt/jmeter5.4
jmeter.out.home=/opt/test/
```

Based on the secondary development of Reasonable 8.6.1, it integrates thousands of intelligent automatic detection, scanning and analysis rules, and has the function of self-learning and automatic identification of new security vulnerabilities. It can continuously detect the security vulnerabilities of static source codes from seven dimensions, and can catch and prompt the errors within the codes in advance, before the quality or security of the code base is threatened, ensure the elimination of code vulnerabilities or defects, improve the security and reliability of the code from many aspects. The configuration file sonar.property of SonarQube is related to database.
Gitlab is a version control system with open source, which is used for multiple people to jointly develop the same project and share resources. Through Java secondary development based on its open source interface program, the source code and configuration files of the measurement system can be uploaded to the data warehouse of GitLab directly through the intelligent security test platform. Postgres13.0 database is used to store data tables and records related to test, security vulnerability scanning, version control and intelligent security testing platform. Maven 3.6 has a set of source code analyzers in Maven plug-ins, which uses the configuration stored in the database to complete the unified development specification and manage the jar package. Generally, SonarQube relies on Maven to run the analysis, and can also analyze non-Maven projects.

4.2. Example

The contents for function and performance tests involves remote measurement system of electric energy in power plant and station, load management system for key customers, monitoring and metering system for electric distribution transformer, centralized reading system in low voltage of smart metering central system. The system is supporting the business of Shenzhen Bureau, which includes remote meter reading, electricity statistics, load management, line loss statistics, power consumption inspection, distribution application, voltage qualification rate statistics, customer outage time statistics and remote inspection of metering devices etc.

At first, the number of test scripts and test cases in the GitLab repository existed, which are recorded, programmed, debugged, implemented successfully in advance by the professional test team from the third-party organization. The team is assigned by the privilege to access the module of the file management and to load the test files into the repository.

Then, the operation and maintenance personnel of the PMS software can login the platform and open the “test plan” Web page (Figure 3), select both test script and case from drop-down lists, items of which come from the GitLab repository. The plan will be created in the database in means of clicking Add button. There is a new record shown in the plan list in the bottom of the page. By the same way a set of test plans will be created depending on the different requirements of the software functionality and performance tests.

When the new version of the PMS software is scheduled to deploy the production environment, it makes sure that the version will be no effect on the parts of current version of the software before doing it. Therefore, for the personnel to operate and maintain, it will be able to complete all of various tests by the access and implementation on the platform.
As a sample of application using the JMeter, an automatic test tool with open sources, the operation and maintenance people can go through the Web page of executing test, select one of the test plans in the drop-down list, such as a plan called UserSearchPerfPlan and choose a server or virtual machine from a list, where the JMeter is running, the series of intelligent test steps described in the following will be automatically implemented by clicking “Start Testing” button (Figure 4).

Step one: Program the secondary development by Java, based on the open-source interfaces of JMeter and get thing down of communication between the intelligent test platform and JMeter, transfer parameters of both the test script and test case to the virtual machine on the remote server.

Step two: Check out the following executable command line from database by Java

```
./jmeter -n -t TestScript.jmx -l TestReport -o -e OutputReport
```

and the parameters of the command will be replaced by the name UserSearchPerfPlan of plan,

```
./doc/UserSearchPerfTestPlan.jmx as the name of test script;
```

```
./doc/UserSearchPerfTestPlan.jtl as the name of test report;
```

```
./doc/UserSearchPerfTestPlan as the directory of test report.
```

Here .\doc refers to a root directory storing the test reports.

Step three: Remove all of files with .jtl and the directory of the test report sub-directory of .\doc to escape from a conflict when running the above command so that new test reports with the same name can’t be created automatically.

Step four: The platform will implement the executable command dynamically created in the Step two, complete the test by transferred script and case and generate the HTML graphical test reports of this test.

Step five: It will be able for the PMS maintenance people to see the test reports on a browser, which include error report, time curves of data throughput, the curve of click rate per a second, time distribution curve of HTTP status code and so on.

Step six: On the base of the result reports, the maintenance people can make a final decision if there exist any bugs to be fixed or not. If it is true, the error report will be submitted to a developing team to solve them. The bugs will be fixed, the people can simply do the regression testing until the test results have satisfied the whole requirements of the software new version.

Step seven: In the final, the new version of the PMS can be deployed on the production servers.

As a result, before the new version of the PMS software is updated, the operation and maintenance people can freely do the different tests for new version and efficiently control the quality of the software new version.

Similarly, the WinRunner, LoadRunner tools can do the automatic test task as well. More and more automatic test tools like the tools of embedded software for smart electricity terminal will be available in future.

5. Conclusion

The intelligent test platform for the power meter system is developed by Java program, executing on Tomcat, database PostgreSQL and version control tool GitLab on virtual machine VMWare with CentOS8, 64bit.

The test team of the third-party organization selects set of proper automatic test tools, and records, writes, debugs implements test scripts and cases in accordance with the requirements of the software functionality and performance. After that, the scripts and cases are loaded into the GitLab repository,
the operation and maintenance personnel of the software will select a group of test scripts and cases from the repository and create a set of test plans. Before the software will be updated with a new version, in order to control the quality of its new version, the maintenance personnel will access the Web page of Executing Test module, select one of the test plans and test tools from database, then click “Start Testing” button, the platform will connect to the test tool on virtual machine, to create a set of executable commands intelligently and dynamically, to implement test script and case, to complete the test task for the software, then to generate a test result report automatically. The personnel can make an adjustment if more regression tests will be done or not depending on the test result until the no bug is found in the tests. Therefore, the operation and maintenance personnel just with the basic skills of software tests can efficiently control the quality of software new version in basis of the test platform. It can reduce the errors of manual tests and improve both the management level and work efficiency.

References

[1] Notice2018 the national development and Reform Commission and the National Energy Administration on actively promoting market-oriented electricity trading and further improving the trading mechanism

[2] Song X Q, Wang P P, Zhou S Y 2014 Overview of software testing technology for smart electric power meter Electrical Measurement and Instrument, vol 51(11), pp18-22

[3] Zhang L P, Wang B H, Hu S S 2019 Research on key test methods of the smart meter software based on failure modes Journal of Physics: Conference Series, vol 1325 pp1-6

[4] Wen S J 2016 Zero cost web performance testing based on Apache JMeter (Electronic Industry Press) p343

[5] Isha A A 2015 Brief survey on Web application performance testing tools literature review International Journal of Latest Trends in Engineering and Technology vol 5(3) pp367-375

[6] Vyatkin V 2013 Software engineering in industrial automation: state-of-the-art review IEEE Transaction on Industrial Information vol 9(3) pp1234-1249

[7] Deepak D and Amit G 2015 Performance testing and evaluation of web applications using wapt pro International Journal of Innovative Research in Computer and Communication Engineering vol 3(7) pp6965-6975.

[8] Nitesh S, Noranjan N, Balaji S, et al 2014 Comparative study of software testing techniques International Journal of Computer Science and Mobile Computing vol 3(10) pp151-158

[9] Sharma M, Shetty A, Subramanian S, et al. 2016 A comparative study on load testing tools International Journal of Innovative Research in Computer and Communication Engineering vol 4(2) pp1906-1912

[10] Serdar D, Aysu B, Vahid G 2014 Web application testing: A systematic literature review Journal of Systems and Software vol 91 pp174-201

[11] Jianf Z and Ahmed H 2015 A survey on load testing of large-scale software systems IEEE Transactions on Software Engineering vol 41(11) pp1-1

[12] Shbha C, Nandhini S, Priyanka R 2017 The analysis of automated testing tools for Web based applications International Journal of Trend in Research and Development vol 4(1) pp496-498

[13] Bures M, Frajralk K, Ahmed B 2018 Automation support of exploratory testing using model reconstruction of the system under test IEEE Transactions on Reliability vol 67(2) pp557-580

[14] Priyanka K, Yashada J 2014 Generating and analyzing test cases from software requirements using nlp and hadoop International Journal of Current Engineering and Technology vol 4(6) pp3934-3937