

PAPER • OPEN ACCESS

Augmenting SCA project management and automation framework

To cite this article: M Iyapparaja and Bhanupriya Sharma 2017 *IOP Conf. Ser.: Mater. Sci. Eng.* **263** 042018

View the [article online](#) for updates and enhancements.

Related content

- [Manual and automation testing and verification of TEQ \(ECI PROPRIETRY\)](#)
Ravi Abhichandra and M Jasmine Pemeena Priyadarsini
- [DEVICES FOR OBSERVATORY AUTOMATION](#)
R. K. Honeycutt, B. R. Adams, D. J. Swearingen et al.
- [AUTOMATION OF A COMMERCIAL ASTRONOMICAL TELESCOPE](#)
Anthony Mallama

Recent citations

- [M. Iyapparaja et al](#)
- [An Effective Detection Mechanism for Localizing Macular Region and Grading Maculopathy](#)
C. R. Dhivyaa and M. Vijayakumar

Augmenting SCA project management and automation framework

Iyapparaja M, Bhanupriya Sharma

School of Information Technology and Engineering, VIT University, Vellore-632014, Tamil Nadu, India.

E-mail: iyapparaja85@gmail.com

Abstract. In our daily life we need to keep the records of the things in order to manage it in more efficient and proper way. Our Company manufactures semiconductor chips and sale it to the buyer. Sometimes it manufactures the entire product and sometimes partially and sometimes it sales the intermediary product obtained during manufacturing, so for the better management of the entire process there is a need to keep the track record of all the entity involved in it. Materials and Methods: Therefore to overcome with the problem the need raised to develop the framework for the maintenance of the project and for the automation testing. Project management framework provides an architecture which supports in managing the project by marinating the records of entire requirements, the test cases that were created for testing each unit of the software, defect raised from the past years. So through this the quality of the project can be maintained. Results: Automation framework provides the architecture which supports the development and implementation of the automation test script for the software testing process. Conclusion: For implementing project management framework the product of HP that is Application Lifecycle management is used which provides central repository to maintain the project.

1. Introduction

It is important to provide and interface to implement the supply chain technology which helps in collecting data from all areas of your chain. Now the functioning of the application is the challenging task as it should meet with the requirements or expectations which lead the development of the application. There is a great possibility of making errors that create faults or the possibility to miss any of the requirements while developing the application which causes failure of the application.

So in order to make the application perfect, test strategies are deployed which help in accessing the faults and the missing requirements. Through this a bug free application along with the fulfilled requirements will be obtained with the optimized risk for the end users.

Many times after the use of the application, new requirements comes into the picture which need to be deployed in the application? So on adding new requirements there is need to test what all things have been affected by this which can be done by the reuse of test cases.

The entire above described thing can be done through the use of application lifecycle management which is used for managing the project and the unified functional testing is used for the automation testing. Project management architecture comprises of the following modules requirement recording, testing details, defects, management and dashboard[1].

Management comprises of release and libraries. Release is created when a new version is launched and for each release its scopes has to be defined which tells the requirements which are going to be cover in the release and its milestones are always defined. In each release cycle are created which

describes that in how many cycle the release meets the user requirements as many a times even after the keen scrutiny of the project some defects penetrates in and appears after the delivery which are fixed in the next cycle[2].

Requirement module is used to track and record the requirements of the application or for each release of the application. First we have to define the requirements of the application from the provided document and if there is no document present for the existing system, then we have to identify the requirements by analysing the code which is a crucial task. After defining the requirements we need to connect them with the appropriate release so that the track can be maintained which helps in determining that in which release which requirements are fulfilled. In further process, testing is performed for each requirements mention here in order to verify that it is functioning properly.

Testing comprises of four components that are test resource, plan, lab and runs. Test resource component is used when we are introducing the test cases from other testing tool which are supported by this tool[4]. In this the things related to the automated test scripts which help in the successful execution of the test scripts are stored here. The things which help in successful execution of the test scripts are object repository, user defined functional library and data sheets. Now in the test plan module test cases are created either manually or automatically or both. Manual test case are under design steps and automated test cases are under test script. After the creation the test cases the test cases are need to be linked with the requirements which specify that the following test case is fulfilled through this test case. Multiple test cases can be linked with one single requirements and also multiple requirements can be covered through a single test case. Test lab components help in grouping or collecting the test cases in form of test suit which help in exploring the application as changes done in the test lab will not affect the parent test present in test plan and due to this reason only it is named as test lab. This test suits are need to be connected with the specified cycle of the specified release in order to know for which release this test suit has been executed. In test lab we can run the test cases for different inputs, different situation in order to explore more and more of the application and also to verify the application keenly. Test run component is used for analysing the result obtained from the execution of the test case in the test lab [3].

Defects are a component used for managing the defects occurred while testing the application. The defects can be created by taking the snapshot of the error occurred and can mail the error occurred during the execution to the specified developer in order to fix it. When defects occur there is a need to log it and while logging or creating the defect it is necessary to provide the complete description of the defect. Then this defect should be connected with the script and duplicate defects should not be present.

Dashboard component is embraced of analysis view and dashboard view. Analysis view is used for analysing the project. It helps the user for creating, managing and viewing analysis items for example graphs, excel reports and project reports.

Test automation framework provides us with the environment which helps in planning, building and executing the test cases. It provides the environment which consists of the set of protocols, standards for coding, procedures, notion, hierarchies of the project, test data injection etc. Testing is a process to identify any bug or error is present in the developed application or not. In other words, it is the process of verifying the functionality of the application that it is working correctly or not. Testing can be done in two ways that are manually or by automation [5].

Manual testing is done by the tester manually prohibiting the use of any testing tool. In this large man power is needed as the test has to keep his eye over the application and should have a domain expertise knowledge of the application which helps in identifying the unanticipated behaviour or bug. Manual testing is used when you need to find mainly the bugs while automation testing is done for positive testing only. Manual testing consist of different stages which are as follows- Unit testing, Integration testing, System testing, and User Acceptance testing. As tester investigates the software in order to find the bugs so exploratory testing is also included in this.

On the other hand automation testing, test the software with the use of software through the scripts developed. These scripts are the test scripts which guide the steps to the testing tool for moving future and achieve the completeness of the testing. Manual process is automated through this process. Re-usage of the test scenario is possible through automation testing and decrease the no of test scenario and enlarge the test coverage, enhance the accuracy and retain time and money in contrast to manual testing.

2. Development of the proposed system

For the Subcontractor Collaborative Application, the project management framework is developed using Application Lifecycle Management. It is also known by the name quality centre which was its old version name. By managing each and every release of the application corresponding with its requirements and testing scenarios and also management of the defects helps in gaining the quality of the product along with its enhanced performance.

Now automation testing framework for the Subcontractor Collaborative Application uses two methods for generating the test scripts. Those methods are object repository based test design and descriptive programming. First, in the object repository based test design, recording is used for the generation of the test script. In recording each and every user action is recorded and framed into a statement which is framed using visual basic scripting language standards. In this procedure the object is identified while recording and stored in the local object repository. If the object misses from the object repository then an error will be thrown while testing and the test remains uncompleted [6].

Second is the Descriptive programming through which dynamic objects are handled. Through this methodology we can develop the test script manually using visual basic scripting language standards. In this methodology there is no need of developing object repository as the description of the object is directly mentioned in the steps or the statement. As the object repository is not used in this methodology so execution is faster here in contrast to object repository based testing [7].

3. System preliminary design

Project management architecture consists of five phases. They are

1. Requirement gathering and defining phase
2. Release management phase
3. Test planning phase
4. Defect management phase
5. Analysis phase

Test automation architecture consists of three phases. They are

1. Identifying objects and developing object repository phase
2. Test script development phase
3. Result analysis phase

The complete structure is illustrated in the following figure 1.

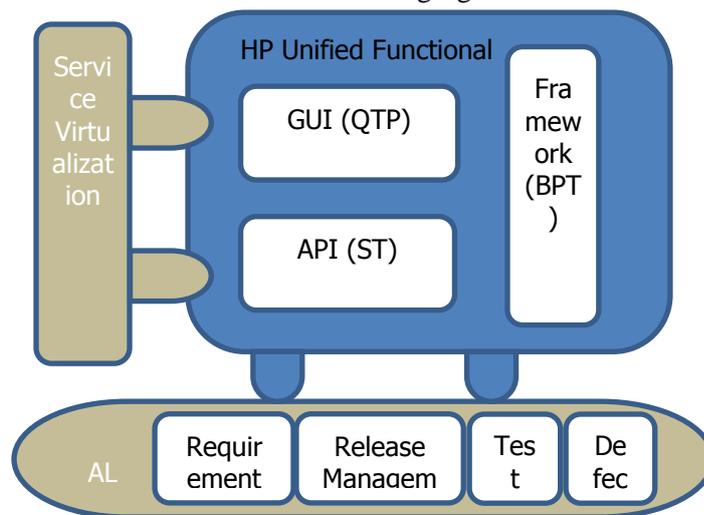


Figure 1. Architecture of integrated HP ALM and HP UFT

4. System planning

The proposed system is implemented in the five phases or modules as depicted below [8]. These modules are according to the reflections of architecting and implementing which depicts the way through which we can proceed towards the project. The five modules are described below in pictorial representation in figure 2.

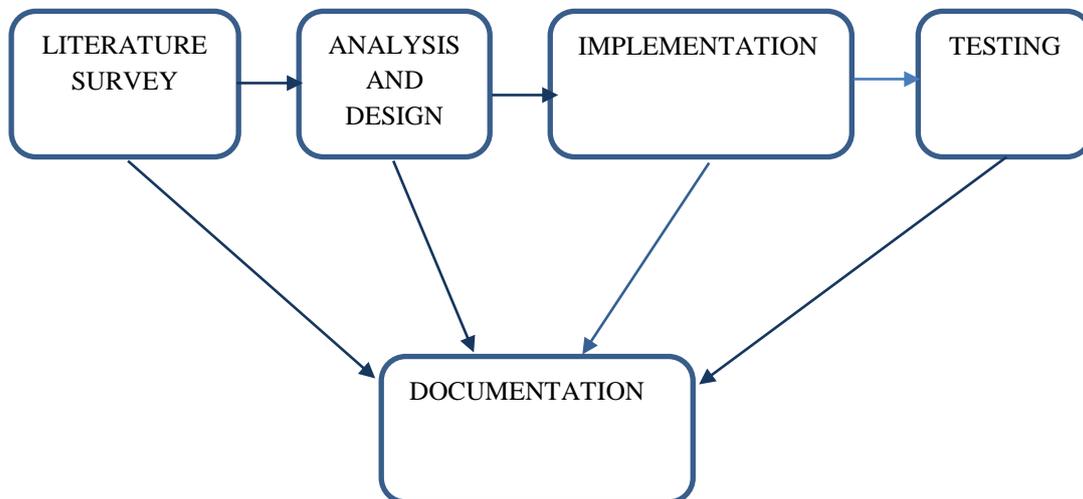


Figure 2. System Planning Diagram

5. System requirements

As consider hardware requirement of HP Application Lifecycle Management, Focusing Server-side system needs Quad Core AMD64 processor or equivalent x86-compatible processor, requires minimum 8GB Memory (RAM) & free disk space. Focusing Client-side system requirement, needs Core duo 1.6Ghz (or higher) or equivalent compatible processor, requires minimum 2GB Memory (RAM) & free disk space. For HP UFT, requires 1.6Ghz (or higher) processor, needs minimum 2GB Memory (RAM) & free disk space, Graphics card with 64 MB video memory & 5400 RPM Hard disk drive.

As consider Software Requirement for HP Application Lifecycle Management, Focusing Server-side system needs MS windows server 2012 R2 standard, Red Hat Enterprise Linux 6.3 , requires MS SQL Server 2012 SP2, Oracle 11.2.03 database & Microsoft IIS8.5 web server . Focusing Client-side system requirement, needs MS windows 7 SP1 OS, requires Microsoft Internet Explorer 11 browser, Microsoft office 2010 SP2 suite & MS .NET framework 4.0/ 4.5x ; MS office 2010/2013 prerequisites. In Web –client Supported environment requires MS windows 8 SP1 32 bit/64 bit; Red hat Enterprise Linux 6.5 OS & Microsoft Internet Explorer 11; Google Chrome 35; Microsoft Firefox 29 browser. In HPUFT needs Windows 7 Service Pack 1 (32 bit or 64 bit) OS & Microsoft Internet Explorer 9+ browser.

6. Other Non-functional requirements

6.1. Performance requirement

For best performance of the system, the following above mentioned hardware requirements should be fulfilled. Maintenance of the data constraints should be properly done. Based on the functionality of the system, the inputs and outputs by the user to the system should be clear. Through the help of secure methods different data malfunctions are detected and rectified. For better performance following requirements should be achieved that are hardware, environment, software and interface [9].

6.2. Safety requirement

Verification of entering of improper or incorrect data is done and if it is detected than an error message should be thrown. For preventing the system from the attack different safety programs should be run by the system. While performing the operation when the user goes wrong then the user is guided through the instructions. At each foot of the framework includes the well arrangement of safeguards are archived and conveyed to the client.

6.3. Security requirements

To suppress the uncertified access to the system, user authentication is performed. When uncertified account details are detected then the predefined actions are followed. It is strictly prohibited to take copyrights over software which already copyrighted. The programs which are used for investigating the illegal actions for breaking down the system should be highly prioritized [10].

6.4. Software quality attributes:

Only certified user can access the system. It should be flexible so that the instructions become clear and easily usable. Maintenance of fault tolerance should be done to an extent. At any stage maintenance can be done. For easier reusability the desired software should be packaged together [9].

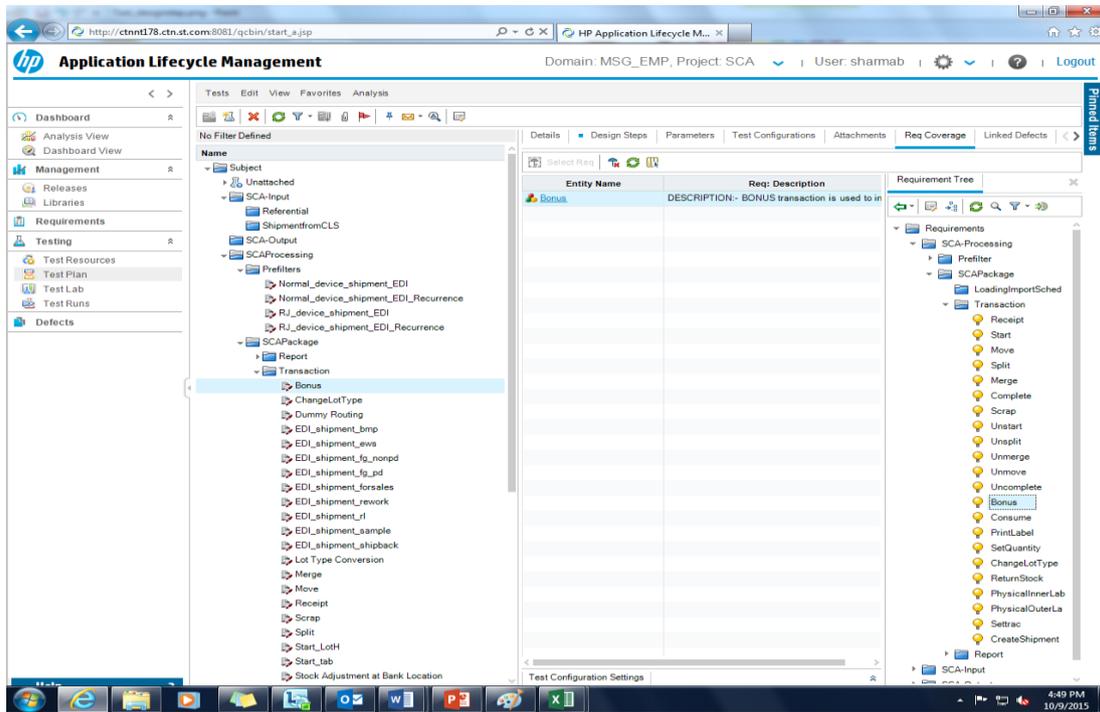
7. Results & Discussion

In this, we focused the automation testing framework in Calculator desktop application, “**Test Automation of Desktop application Calculator**” of which concentrates following sequence of action for executing the test cases,

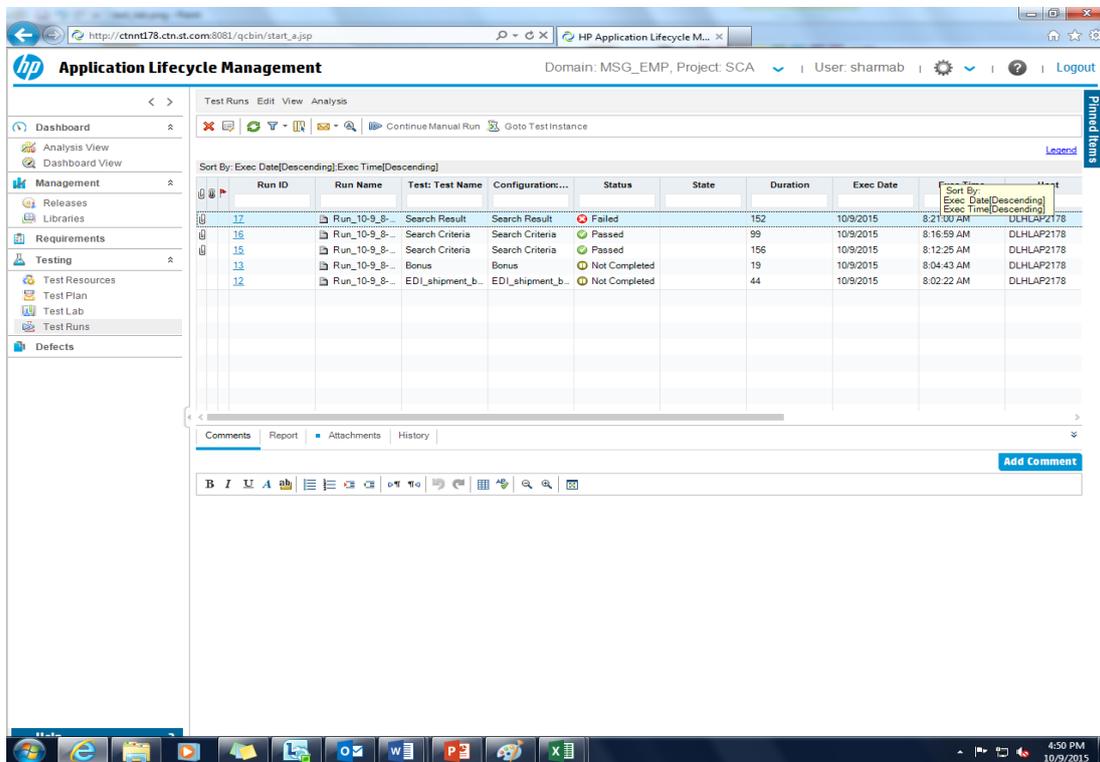
Navigate twice the Calculator Application Window which is found in two different modes. Name the mode as Calci_std_mode : for its standard mode. Calci_scientific_mode: for its scientific mode. Assimilates Help topics windows of the calculator. Assimilates about the calculator window help in the calculator. (Warning: In default name blank space should be eliminated). Assimilates Statistics Box. In the Project DB the following Testsuite, TPRs and TestCases are developed. The easier practise would be to perform the task from bottom to the up. First perform testcases, then perform TPRs, and lastly perform testsuites.

- a. RegressionTest_Testsuite
- i. SmokeTest_TRP - To perform smoke test from the following just select any couple and from the test case raise one question.
 ModeSwitch_TestCase, Keybrd_TestCase, Buttons_TestCase, SimpleMath_TestCase, HexOctBin_TestCase, SimpleTrig_TestCase, SimpleLogic_TestCase, SimpleLog_TestCase, SimpleStatistics_TestCase, SimpleMemory_TestCase, DivideByZero_TestCase, SimpleHelp_TestCase
- ii. Help_TPR
 VerifyVersion_TestCase, VerifyHelpContentsTab_TestCase, VerifyHelpIndexTab_TestCase, VerifyHelpSearchTab_TestCase, VerifyHelpNavigation_TestCase
- iii. StdModeSqrt_TPR
 VerifySqrt_TestCase - use 10 numbers stored in Data Table
- iv. NegativeTesting_TRP
 VerifyMaxDigits_TestCase, VerifyNonNumbers_TestCase, DivideByZero_TestCase, Log0_TestCase

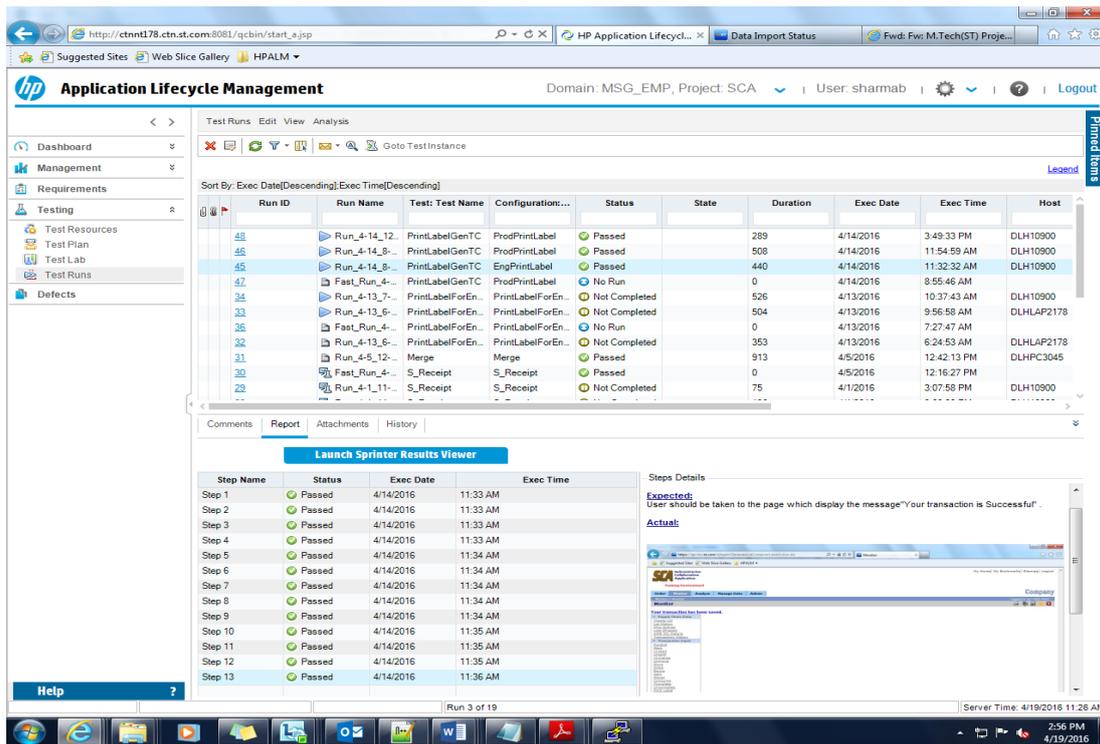
7.1 Run and Debug



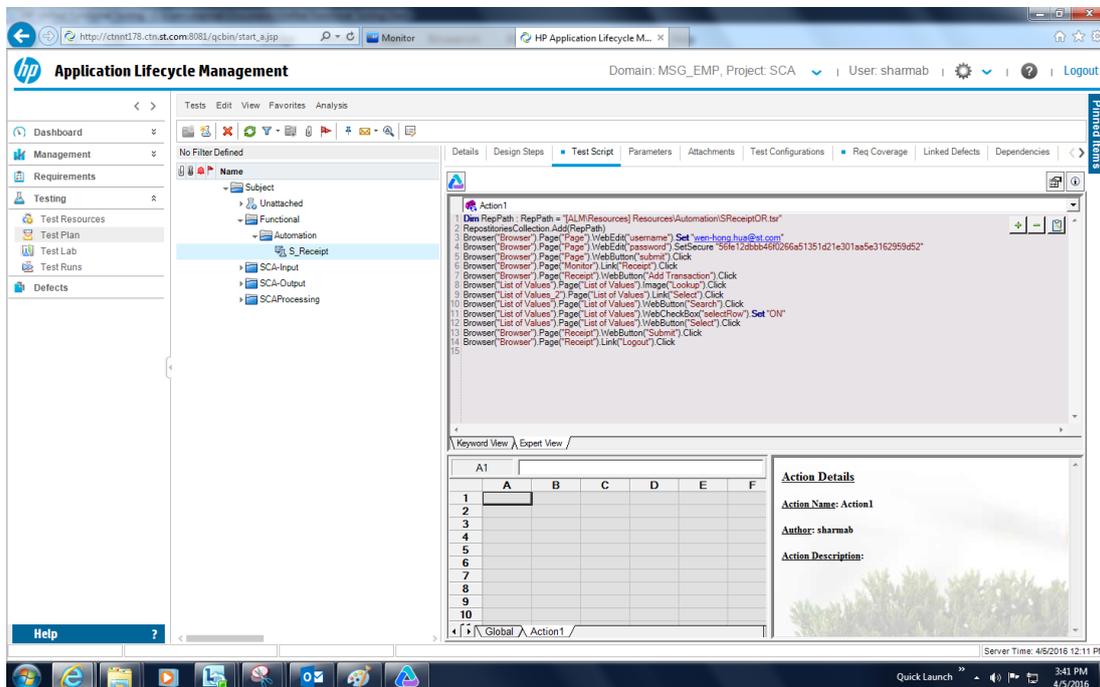
Snapshot 1: Requirement Coverage in Test case in ALM



Snapshot 2: Test Run in ALM



Snapshot 3: Test Run Report in ALM



Snapshot 4: Test Script in ALM

8. Conclusion

Project management framework develops the platform and the environment through which we can easily manage the project under development and testing. This framework depends on the management suite through which you can keep an eye on the project and can perform the task within the defined time and can deliver the correct product to the customer. For the correct product delivery

testing also plays a vital role and for performing this we need to develop the automation framework. This automation framework provides the automation platform for performing the testing process and it rely on automation suit also on the testing application.

References

- [1] Alex Cervantes Exploring the use of a Test automation framework *IEEE AC* **009 2** 1477.
- [2] Boehm B 2005 Value Based Software Engineering: Overview and Agenda Springer
- [3] Dustin E Automated 1999 Software Testing *Addison – Wesley*
- [4] Fewster M and Graham D 1999 Software Test Automation: Effective use of test execution tools *Addison-Wesley*
- [5] Greehanik M and Chen F U 2009 Maintaining and Evolving GUI Directed Test scripts *IEEE* 16-24
- [6] Iyapparaja M et.al. 2012 Coupling and Cohesion Metrics in Java for Adaptive Reusability Risk Reduction *IET Chennai 3rd International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2012)* 52-57.
- [7] Leckraj Nagowah and Purmanand Roopnah 2010 AsT – A Somple Automated system testing Tool *IEEE* **9(10)** 978
- [8] Leshin A, Gerenko M and Dmitriev V 2009 Test Automation - Flexible way *IEEE* 978
- [9] Marinov D, Khurshid and TestEra S A 2001 Novel Framework for Automated Testing of Java Programs *IEEE International conference on Automated Software Engineering* 22-34
- [10] Pressman R S Software Engineering: A Practitioner’s Approach *Mcgraw-Hill International Edition* ISBN: 007-124083-7
- [11] Rameler R and Biffel S 2005 Value based Management Software Testing Springer