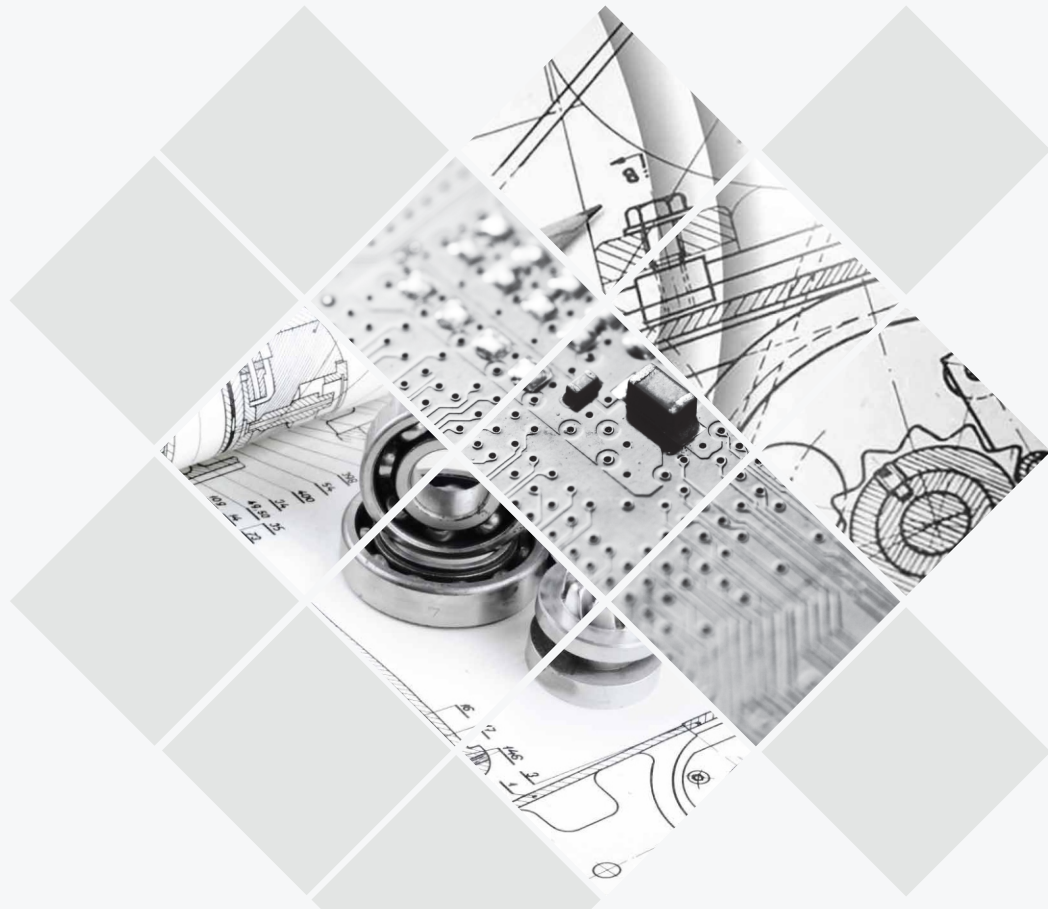


test automation & acceptance with extensibility- TAAAX

In this paper the authors are sharing the experience of usage of a framework, TAAAX which helps the testing community to integrate the test management aspects with the different testing tools available.



contents

0.1	Abstract	01
0.2	Introduction	02
0.3	Industry Practice and Lacunae	03
0.4	Solution	05
0.5	Return on Investment	10
0.6	Conclusion	10
0.7	Author Profile	11



Test Automation & Acceptance with Extensibility, TAAX

Abstract

Today's software industry has seen a rapid growth in all the areas, based on the demands from the market, which has increased the challenge in the testing industry. The days in which the functionality testing has been given the prime importance has gone and in the current industry scenario, along with functionality testing, other types of testing like performance, security, stress, volume etc has gained importance which in turn increased the expectations from the testing community. It is evident that a product which meets the functionality requirements, but fails in any of the non functional tests like performance, security, compliance etc. is likely to be rejected by the customer. The Final quality assessment of the product is a function of how much functionality, performance, security and other non functional requirements are met by the product. This demands to have the transformation from assurance to confirmation.

Even though the scope of testing has been widened; the time to market has been reduced and as a result the schedule for testing has been further reduced. The testing phase needs to accommodate for the slippages of the previous phases too. The testers have been left with the choice that they have to ensure the quality of the product to be improved within the schedule allocated to the testing team – The cycle time reduction being one of the key factors here. This reduction in schedule for testing is mainly addressed by the tester community with the concepts of early testing and usage of tools. As a result the industry has witnessed a tremendous improvement in the usage of tools in testing areana in the last few years. Along with the brighter side of tools the analysis and consolidation of the test results from this multiple tools used inside the project has taken a large effort of the testers and getting these reports from multiple locations has become a hectic task for the managers.

In this paper the authors are sharing the experience of usage of a framework, TAAX which helps the testing community to integrate the test management aspects with the different testing tools available. This framework ensures that the results of the testing are reaching the relevant stakeholders which have made the life of managers easy.



Test Automation & Acceptance with Extensibility, TAAX

1. Introduction

Testing, the gateway phase between the development organization and the customer is chaotic phase were the software is 99% complete, but not ready to use. Years has seen this torrent phase to be disturbing and drift the software projects to failures. It is history that the phase which was initially envisaged as one to improve the confidence of the developer has emerged itself to an engineering discipline which demands for enormous but systematic improvements. They key factor which has driven the change is the wide acceptance and requirements of non functional testing like performance, load, stress, volume etc as well as the usage of test automation tools – mainly in the functional testing area.

From the period when testing has started gaining importance the prime importance has been given to the functionality testing. The rapidly growing software industry witnesses the requirement of non function testing. A banking application which fails in security, a web mail application which fails in meeting the performance expectations, a mobile application which cannot store the expected number of contact details etc. fails to meet the customer expectations and will be thrown out of the competition rich software market. This made the organizations focusing more on non functional aspects of the products which is being released.

Multiple products been released to market at marvelous pace which has emphasized the importance of performance of the products been released. With the availability of multiple products to choose and also with the increased exploitation of the technology the different types of testing like security, performance, stress, volume, load etc are gaining importance while the importance of functionality testing is still continuing.

The industry also has witnessed high usage of testing tools in functional test automation as well as other non functional testing. The test managers need to ply between the results of multiple test tools to make a call on the current quality of the product under test. At this point the importance of porting the different tools like functional test automation tools, performance testing tools, compliance testing tools etc. under one umbrella gains importance. If there had been a system which helps the tester to identify the test cases based on the type, severity and priority, execute these test cases as a group, after the completion of testing the results will be analyzed, check whether the acceptance criteria has been met and inform all the relevant stakeholders on the result of the testing that might have reduced the effort of testing and make the life of the project managers much more comfortable and make the teams more efficient.

TAAX framework, which is being proposed, is a Hybrid Test automation framework which take care of Automation of various types of testing and which consolidate entire test results and check whether the acceptance criteria defined for product through the mentioned framework is met or not.



Test Automation & Acceptance with Extensibility, TAAX

2. Industry Practice and Lacunae

The current industry practice of testing is pretty simple and straight. The test cases shall be identified by the tester based on the requirements specification of the project. The test engineers will identify the type of test case, its severity and priority. These test cases could be kept in different templates and tools, in predefined formats. During the test execution phase of software life cycle, different category of test engineers such as Functional, Performance, Security etc will execute these test cases either manually or using available automation tools. After the completion of the testing the testers will analyze the results given by different tools and consolidate it into a predefined format. The project manager validates whether the acceptance criteria defined in the project is met or not from the test results which has been consolidated by the test engineer.

The different activities in each phase of the testing lifecycle and the level of automation available in each phase are depicted in the Figure 1: Level of automation in different phases of testing lifecycle below.

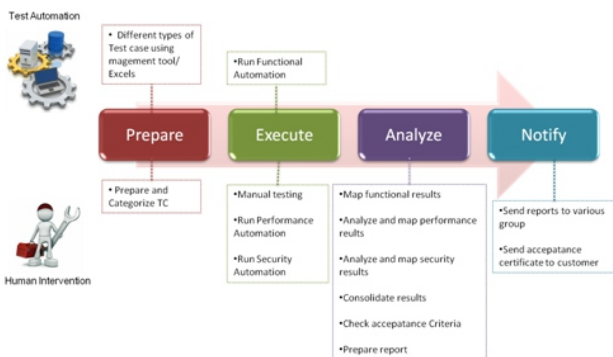


Figure 1 : Level of automation in different phases of testing lifecycle

The current industry practice mentioned above is having the following challenges and drawbacks.

i. No centralized controlling

In the current industry practice the test cases which are classified under different types shall be executed by the test engineers either manually or using tools. For executing each test group the tester has to execute different tools – test automation tools for functional testing, performance testing tools for checking the performance of the system, compliance checking tools for checking the standard compliances of the industry / domain etc. The results from different tools, which are stored in different locations, in different formats, need to be analyzed by the test engineers before consolidating the same into a defined format by the organization. It is indeed a challenge for the test engineers to identify which test results belongs to which test group and sequencing of the execution of these tools. The challenge of the tester with no centralized system to control is as depicted in the .

ii. Manual Consolidation

As depicted in the figure 2, the test results which are available in different locations has to be consolidated manually by the test engineer into a predefined format. The reports which will be generated by these tools will be in different formats. The test engineer should have the capability to analyze these different test results given by these tools. It is indeed a challenge for the test engineers to analyze the test reports from different tools and consolidate the same to a defined format. This challenge has been depicted in the following Figure 3: Challenge of consolidation of the results into a format

iii. Checking the Acceptance criteria

The acceptance criteria for the project shall be defined in the test plan of the project. After the completion of the testing, the test manger/project manager has to verify whether the product has met the defined acceptance criteria or not. From the complex system as depicted in the above figure it is tough for the manager to verify the acceptance criteria compliance of the project.

Test Automation & Acceptance with Extensibility, TAAX

The challenge of the manager is as depicted in the Figure 3: Challenge faced by the test/project manager in checking the acceptance criteria



Figure 2 : Challenge of a tester with no centralized control

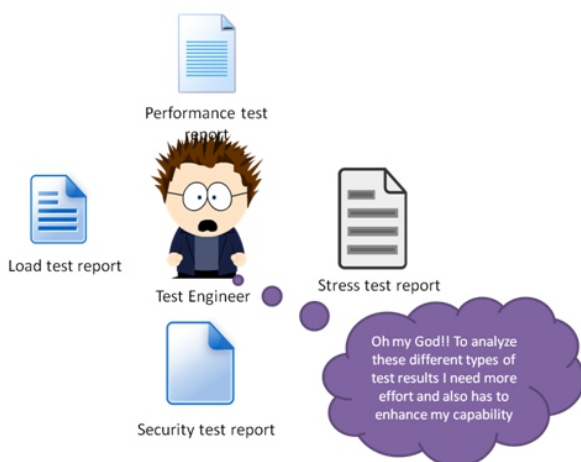


Figure 3 : Challenge of consolidation of the results into a format

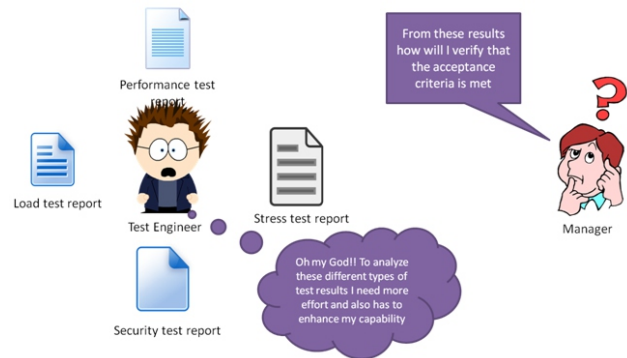


Figure 4 : Challenge faced by the test/project manager in checking the acceptance criteria

iv. Time and Accuracy

It is evident from the scenarios which have been mentioned in the above section that there will be a lot of effort that needs to be put in for consolidating the test results. The testers inside a project will be of having skills in different areas. This makes the analysis which has been done by the testers to be error prone and not effective. These issues might have been rectified if there can be a system which can trigger the execution of different tools and consolidate the results into a defined format after analysis.

v. Notification to the relevant stakeholders

In the current industry practice followed the testers has to send the test results after consolidation shall be send as mail to the relevant stakeholders. The challenge in this scenario is that the stakeholders may vary based on the different types of testing. The stakeholders for functional testing may not be the stakeholders of non functional testing. It will be hard-hitting to identify the relevant stakeholders and ensure that the information has reached only the relevant stakeholder after testing of each test group.



Test Automation & Acceptance with Extensibility, TAAX

3. Solution

The lacunae in the industry system as explained in section 2 triggered the authors to think about a system which is

1. A centralized system from which the testers can configure the test groups and trigger the execution of the testing tools based on the identified test group.
2. Automated consolidation of the test results from different testing tools from the formats produced by the tool to a defined template
3. Automated verification of the acceptance criteria defined for the product and
4. Automated notifications to the relevant stakeholders of the project

The solution suggested by the authors is presented as TAAX – **A Next Generation framework for Test Automation and Acceptance with Extensibility (TAAX)** which will help the testing community by reducing the effort, wiping off the error prone reports and thereby improving the quality of the product. TAAX is a framework used for automating the test execution and management activities inside the project.

TAAX framework can be used for test execution at any stage of the testing lifecycle; regardless of whether it is independent testing, acceptance testing etc. The creation and selection of the test groups and setting up the acceptance criteria is the only effort coming to the tester once the TAAX framework is installed and the test projects are created inside the framework.

TAAX framework interacts with the test management tools and provides the testers the flexibility to create test groups based on the priority and type of test case. The tools which shall be used for executing these test cases will

be identified while creating the test groups itself. These configurations will trigger the execution of the test cases using the tools identified. After completion of testing the test results from these tools will be analyzed and consolidated automatically into a defined format. The acceptance criteria shall be checked automatically by the framework and the results will be communicated to the relevant stakeholders through automatic mails.

3.1 TAAX framework Architecture

There are three main modules in the TAAX framework. They are

1. Interface modules :- These modules helps to interact with the test management tools and the test automation tools
2. Configuration modules :- Configuration of the different factors like test groups, sequencing of test cases, setting up of notification groups and setting up of the acceptance criteria are done using this module.
3. Project Management module:- Different files which will contain the test cases that needs to be executed with the relevant tools and the acceptance criteria for each phase is defined using this module.

The high level architecture of TAAX is as depicted in the below Figure 5: High Level Architecture of TAAX

3.1.1 TAAX Interface Modules

TAAX is having two interface modules – Interface module to interact with the test management tools and interface module to interact with the test automation tools which are depicted as test case tools interface and Tools interface respectively in Figure 5. The Test case tools interface ports the test cases from the test management tool to the test configuration module of TAAX framework. The uploaders in the test case tools interface

Test Automation & Acceptance with Extensibility, TAAX

module accept the test cases in the formats generated by the test management tools and convert it into the defined test case format.

The tools interface module analyzes and converts the test results from the automation tools from their known formats and map the pass/fail test cases in the defined test case document. The acceptance criteria are verified and the mail notification shall be passed to the relevant stakeholders by the test tools layer. The function of uploaders and transformers are mentioned in the Figure 6 : Uploader and Transformers

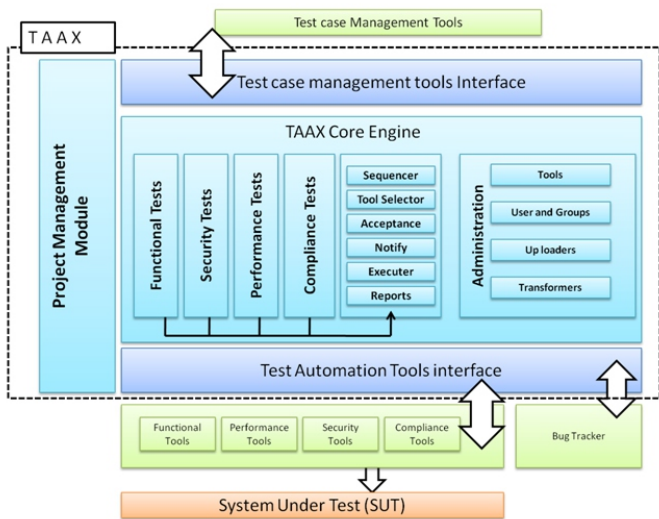


Figure 5 : High Level Architecture of TAAX

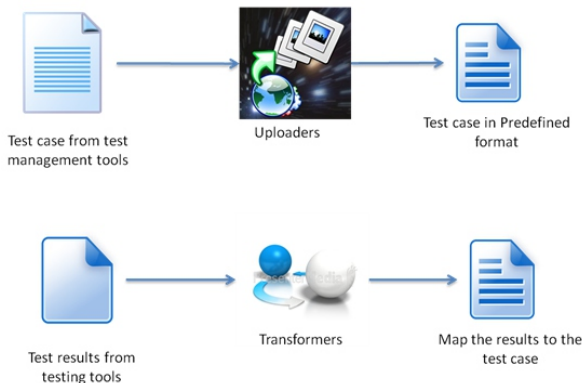


Figure 6 : Uploaders and Transformers

3.1.1 TAAX Core Engine

TAAX core engine is the module in which the entire configuration activities of the framework is been done. The different modules inside the core engine are as depicted in the below Figure 7: TAAX Core Engine

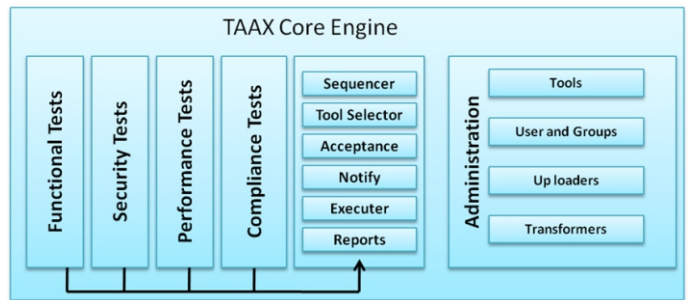


Figure 7 : TAAX Core Engine

Test groups can be generated inside the core engine from the test cases which are ported to the core engine by the uploaders through the test case management interface module. Test groups can be created for different types of test cases and priority in which the test groups to be executed based on the product characteristics and requirements. For e.g. if a projects is having three functionality test groups; the priority of execution of these test groups can be assigned using the sequencer sub module. While executing these test groups those with high priority will be executed first by TAAX framework. The tools which are already configured to the TAAX framework can be assigned to each test group using the tools selector sub module. TAAX framework will automatically execute the test automation tools when the test group is taken for execution. Users have the facility to create new test groups, Load existing test groups which are already created, modify existing test groups and save the modifications in the project created in the TAAX framework.

Acceptance criteria can be selected for the test groups using the acceptance sub module of TAAX framework. Acceptance criteria can be set



Test Automation & Acceptance with Extensibility, TAAX

for any stage of testing like independent testing, acceptance testing etc. TAAX framework provides the option to modify and save existing acceptance criteria by providing the testers the flexibility to select the acceptance criteria as per the stage of testing.

The stakeholders who need to know the results of the functionality testing may not be interested to know the results of performance testing. The testers who are doing the performance testing don't have to hassle about the results of the security testing. TAAX framework provides the facility to set the stakeholders for each test groups using the sub module Notify. Automated notification shall be send to the stakeholders identified in the test group after the completion of the testing.

Testers can select the test groups which need to be executed using the executer sub module of TAAX core engine. The execution sequence can be selected by the tester based on the priority of the test execution. The results after the test execution shall be viewed using the reports sub module of core engine. Result module of the sub module reports will communicate with the interfacing module and acceptance module and corresponding result will supplied to the sub modules. This module communicates with the native test case repository, native Test results repository and acceptance module to generate final consolidated results. Certificate generator sub module of reports communicates with the acceptance module and based on the overall test criteria success this module generates a certificate on PDF format and finally kept in the output result repository.

3.1.1 Project Management Module

The project Management module of TAAX framework helps the users to create projects inside the TAAX framework which can be saved as xml files and can be executed using the framework. The workflow that needs to be

followed for creating a project is as mentioned in the Figure 8: Workflow of TAAX project creation

Administration activities need to be done in the TAAX framework as a first step in the TAAX project creation. As part of the administration activities tools are added to the TAAX framework in order to assign the tools to the test groups. Organizational groups shall be identified so that the notifications shall be sending to these groups automatically after the test execution. User profiles are also created during this stage. The pages in the TAAX framework for adding the tools, organization groups and user profiles are as shown in the below Figure 9: Screens for adding tools, organization group and users in TAAX

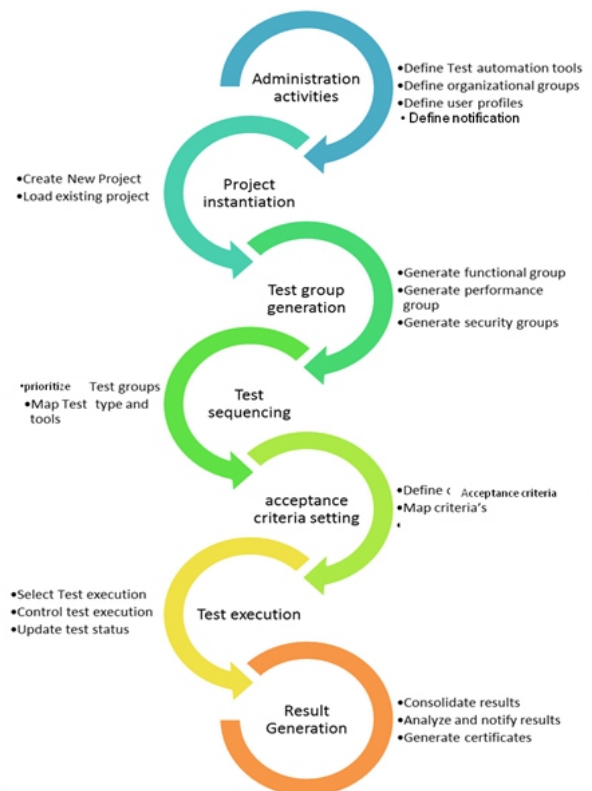


Figure 8 : Workflow of TAAX project creation



Test Automation & Acceptance with Extensibility, TAAX

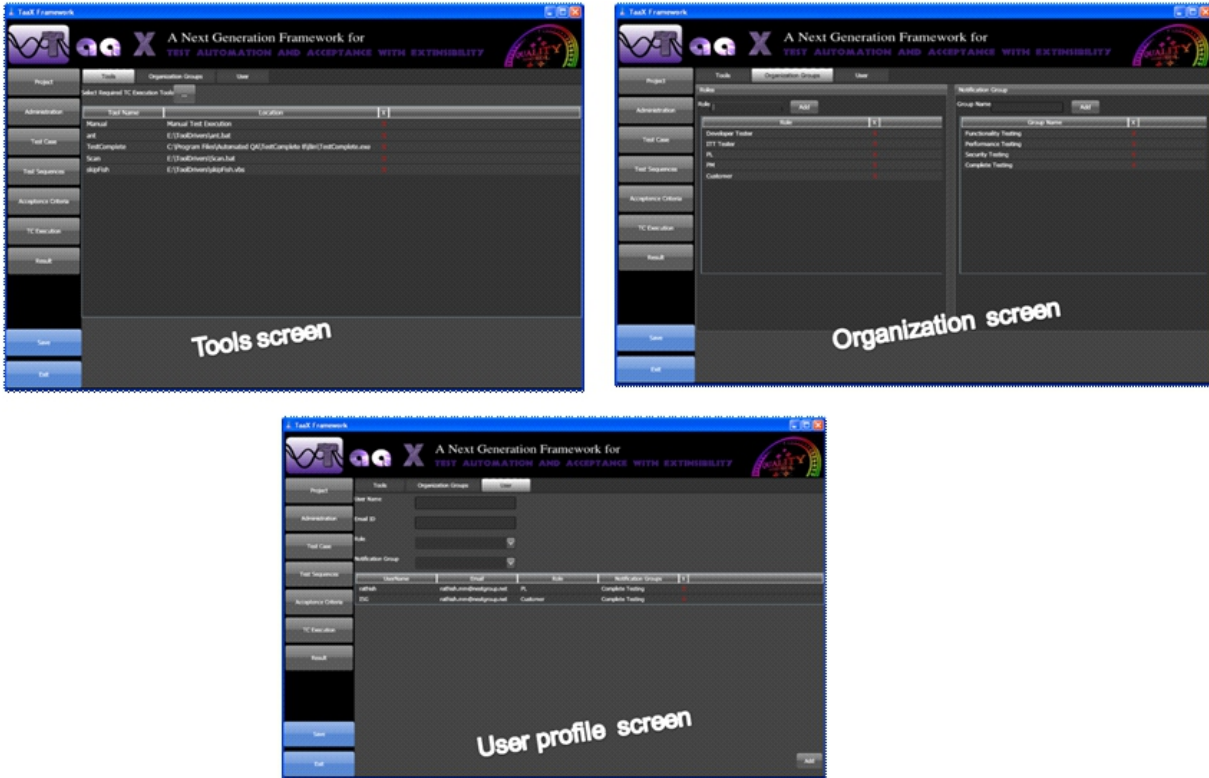


Figure 9 : Screens for adding tools, organization groups and users in TAAX

These tools ensure that the projects shall be created inside the TAAX framework. A project shall contain test groups that needs to be executed which include the priority of execution, acceptance criteria, test results and the group to which the notification of the results needs to be send. The TAAX screen for creating a project is as shown in the Figure 10: Project Screen . New projects can be created or existing projects can be loaded to the TAAX framework. Modifications can be done and saved using the Save button in the project screen.

Test groups are created based on the test case type and the priority of execution, test tools are assigned for each test group, acceptance criteria and the set of stakeholders to which notification needs to be send are selected and assigned to the test groups. The screen of TAAX framework which is used for viewing the test cases which are ported from the test management tool and creating the test groups are as shown in Figure 11: Test case screen

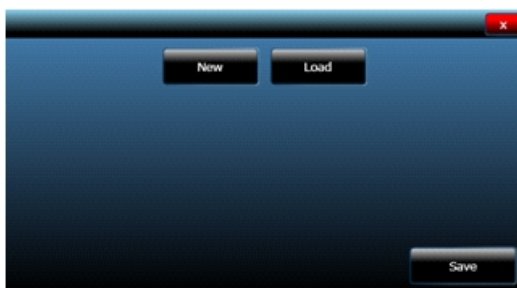


Figure 10 : Project Screen

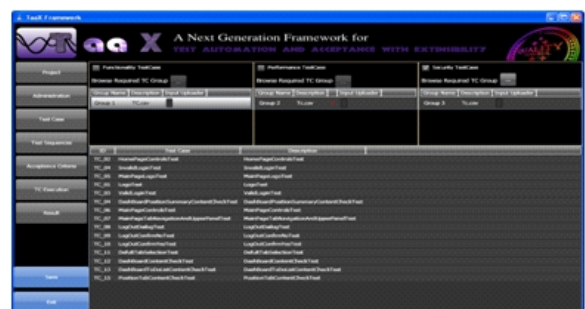


Figure 11 : Test case screen



Test Automation & Acceptance with Extensibility, TAAX

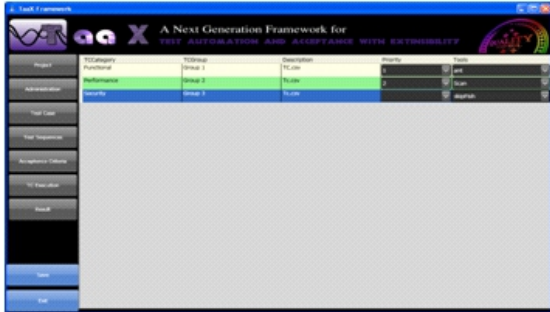


Figure 12 : Test sequencing screen

Test sequencing is the next step in which each test group is mapped to an execution type either as automation or manual and the priority of execution. If it is automation then available automation tools are also mapped. The test sequencing screen of TAAX is as shown in the below Figure 12: Test sequencing screen

The acceptance criteria for each test group can be assigned in the acceptance criteria setting phase. The organization group to which the notification needs to be send for each test group is also assigned to each test group created. During this phase user has the option to select whether a test certificate needs to be generated after the test execution. If user needs a test certificate then a test certificate shall be generated after the test execution completion. The acceptance criteria setting screen of TAAX is as shown in below Figure 13: Acceptance testing screen of TAAX

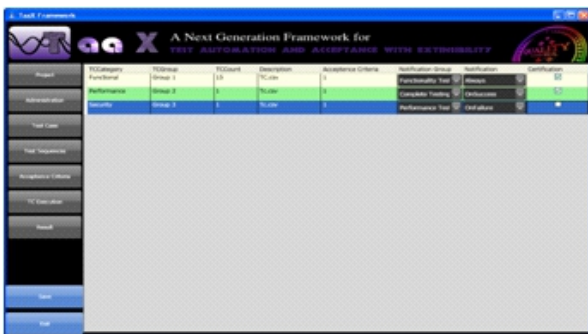


Figure 13 : Acceptance testing screen of TAAX

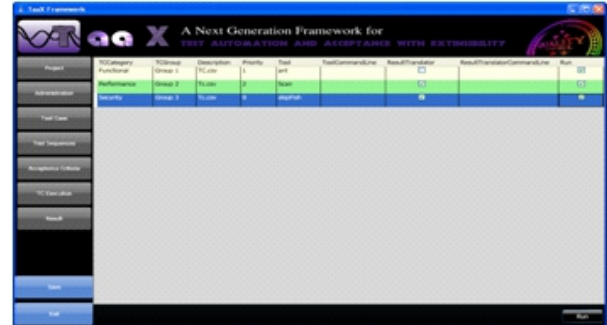


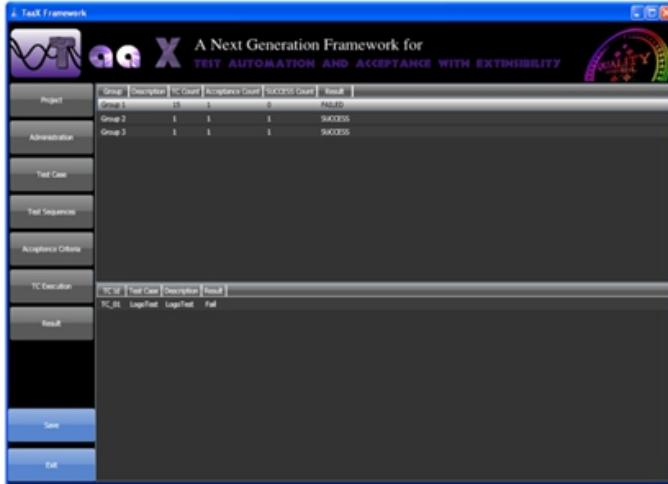
Figure 14 : Test Execution screen of TAAX

Next is the test execution phase of TAAX workflow. During the test execution phase user has to select required test groups and then save the final settings. During this save operation a final framework project configuration file is created in the project output folder. Users can start automated run either from test execution module or from the framework command line option. Each test groups are executed sequentially. During the starting of each test group, tool drivers launch the already mapped automation tool and executes corresponding test cases. If the execution type is selected as manual, then the result of that particular test group needed to place inside the test result output repository. User can view the status of manual execution also through the execution viewer module. The test execution screen of TAAX is as shown in the Figure 14: Test Execution screen of TAAX

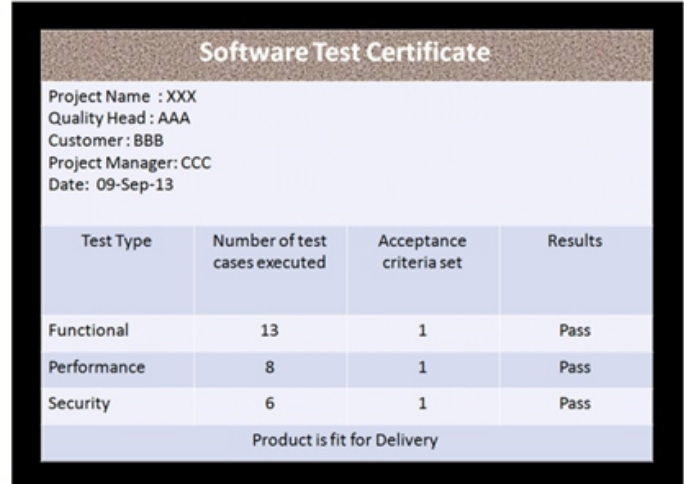
Once the execution completed, the results from various tools are extracted and transformed to the native test result template with the help of Test result transformers. Certificate generator module then analyze the test cases, test results and acceptance criteria of each test group and corresponding status are updated in result viewer module. Result module also generates a Software test certificate in available template. The results viewer screen and the certificate generated are as shown in Figure 15: Result viewer and certificate



Test Automation & Acceptance with Extensibility, TAAX



Result viewer



Software Test Certificate

Figure 15 : Result viewer and certificate

Finally the results and certificate are send to the stakeholders as configured in the notification setting.

4. Return on Investment

By implementing the TAAX framework the authors could see an impressive return on investment. It was observed that an overall saving of 50% in testing was achieved in an year by implementing the TAAX.

TAAX also provides provision to choose the tool for test automation which in turn ensure that optimum usage of test automation license can be done. The test automations are done with freeware tools wherever possible and license tools are used only in slots where it is an absolute necessity. By adapting to such a strategy, a 60% reduction in cost of automation tool has been achieved by reducing the license requirements. Once integrated with Cloud computing, the returns from TAAX are going to increase multifold.

The report generation became automated which in turn removed the possibility of errors which comes as an intangible benefit of TAAX framework.

5. Conclusion

TAXX is a framework for automating the test execution and management which can help the tester to optimize and save the effort, improve the test coverage as the execution of all the test cases are done automatically and thereby improve the quality of the product. The benefits of using the TAAX framework can be consolidated as below

1. A centralized control shall be brought in the test execution process. The framework will perform the test execution by running the test groups based on the priority set and the results shall be verified based on the acceptance criteria set and a test certificate shall be generated. In addition to that all the relevant stakeholders will get the notification on the results in their mailbox.
2. Since the framework consolidates the test results from the automation tools after analysis it saves the effort taken by the testers for doing the manual consolidation. This also helped to reduce the errors which has happened due to human intervention and consolidation done by less capable resources thereby reducing the time and



Test Automation & Acceptance with Extensibility, TAAX

improving the accuracy

3. The acceptance criteria setting feature of the TAAX framework provides the facility to do the test execution in any testing stage and also gives the managers a bird's eye view to the test results.

4. As TAAX framework allows the user to configure the organization groups which needs to be notified after the completion of the test execution only the relevant stakeholders will get the notification.

5. Since the uploaders, which are used for converting the test case formats generated by the test case management tool, and transformers which are used for converting the test results generated by the test automation tool to the predefined test results template can be created for any type of formats; TAAX framework can be used for integrating to any test management tool and test automation tool.

Along with the remarkable benefits which are offered by TAAX framework a setback which can be mentioned is the highly skilled resource required for the implementation of the framework, test case generator scripts and the test results transformers scripts. Another setback of the framework is the effort taking for the first time configuration of the framework. But once the framework is installed it can be operated by less experienced resources and also saves effort inside the project.

6. Author Profile



Pradeep is working as Program Manager at QuEST Trivandrum. He has 10 years of experience in software testing industry and is an Engineering graduate from College of Engineering Trivandrum under Kerala University. He is a Certified Test Manager by STQC and Certified Tester by ISTQB - Advance Level Test Manager. He gives consultation for various testing tools like Selenium, J-Meter, QTP, Rational Functional Tester and Test Complete. He can be contacted at pradeep.pp@quest-global.com



BORN TO ENGINEER

www.quest-global.com