SOFTWARE AUTOMATION TESTING

Ravinder Sharma[1], Er Sonia Rani[2]
[2]HOD, [1,2]CSE Department Institute of Technology and Science, Haryana, India

Abstract:The perpetual demand for faster and,reduced cost of quality, invariably leads Quality Assuranceteams to test a utomation.However, few companies have gotten far within the field of test automation. This thesis investigates how testing and test automation is conducted in a IT Industry. It has been recognized that low test process maturity is common in customer projects and this has led to equally low system testability and stability. The study started with a literature survey which summarized the current state within the field of automated testing. This was followed by a consulting case study. In the case study it was investigated how the identified test process maturity problems affect the test consulting services. The Automation testing strategy been developed to meet the current identified challenges in the domain. Customer guidelines which aim to increase the test process maturity in the customer organization have also been developed as a support to the strategy. Furthermore, the study has included both industrial and academic validation which has been conducted through interviews with consultant practitioners and researchers.

I. INTRODUCTION

Automation is the only long term solution for reduced cost in software testing and better quality products. But these aims are achieved only when best practices are followed before and while developing the automation suite. Howard Fear has aptly stated that take care with automation when done well it can bring many benefits,but when done wrong it can be very expensive resulting in frustration. More often that not after automating the testing of a product, the automation team finds the automation tool more of a headache because of the unplanned and thoughtless approach adopted while developing the tool. Generally, lots of effort is spent in developing the tool, only to discover that the tool is limited tin scope , lacks user friendliness and requires frequent rework now and then and if sufficient care is exercised and proper practices are followed before while automating the same product, the resulting automation tool not only saves time and effort, but is also a sheer beauty in itself because if the amount of user friendliness flexibility ,reusability and extensibility it ensures ,let us therefore, discuss what all needs to be taken care of before going for test automation and also while actually doing the automation.

II. SOFTWARE TESTING

In every large software development project, there exist several defects in artefacts such as requirements, architecture, design in addition to the source code, each of which decrease the quality of the product. Software testing practises are used to ensure quality of software items by finding these defects. The overall development cost can be decreased by finding these defects early in the development process rather than later. For example, consider performing a bug fix to a set of requirements after the implementation has been completed. When performing such change, the already implemented source code may now be based on an incorrect set of requirements. This means that the existing functionality may not be needed after all, rendering the development effort useless. The longer a defect goes unnoticed, the more software artifacts are being developed in parallel. When the defect finally is discovered, these developed artifacts may need changes as a result which in turn increase the time required for bug fixing. This makes is beneficial to conduct the testing practices continuously throughout all development phases. By finding the defects continuous, this feedback can be delivered to the developer responsible for bug-fix immediately thus limiting the affected artifacts that need to be changed.Agile development methodologies have evolved which accommodate the need for continuous testing. Traditionally, every development phase produces the complete set of artifacts before proceeding to the next phase. The main distinction between the agile approaches and traditional ones is that the agile projects are broken up into several releases which are given to the customer throughout the project. In agile methodologies, large sets of documentation are also avoided in favor of strong communication within the development team. Since it is hard to maintain such close communication in large teams, these approaches are considered to be better suited for smaller project teams. Extreme programming(XP) is an agile methodology which emphasizes test-driven development. This simply means that the tests shall drive the development forward and in the case of XP, the testing practices stress the implementation of executable unit test cases.In many organizations there is a reluctance to adapt testing practices due to a misconception that these practices would increase the cost of development. This is not the case in reality since the maintenance and bug fixing required often produce larger total costs without these practices. The lack of enthusiasm for software testing can decrease when the quality benefits are made more visible to the organizations. Also, in my experience, software developers do not consider writing test cases as productive. This is also a misconception since these tests contribute to the increase in quality while decreasing the total development effort at the same time.

III. AIMS AND OBJECTIVES

This aim of this thesis project was to report on the difficulties within the test consulting domain in regards to the automated test methods and processes used. With this information in mind, an automated testing strategy and customer guidelines has been constructed with the aim of making these methods and processes more adaptable between different customer domains. The objectives which were formed prior to the study are primarily described in the list below:

- Identify which approaches and strategies that are used in the consulting domain.
- To understand the testing stages in the software test life cycle.
- To know the automation testing methods and the tools being used in the industry.
- Validate the adaptation efficiency of the strategy in the consulting domain.
- Validate the feasibility and cost effectiveness of the proposed strategy in the consulting domain.

IV. RESEARCH CONTENT

With the aims and objectives in mind, the following set of research topic was constructed:

Sometimes you just don’t pay proper attention what the company-defined processes are and these are for what purposes. There are some myths in testers that they should only go with company processes even these processes are not applicable for their current testing scenario. This results in incomplete and inappropriate application testing.When project goes on expanding the regression testing work simply becomes uncontrolled. Pressure to handle the current functionality changes, previous working functionality checks and bug tracking.Hey tester, we want to ship this product by this weekend, are you ready for completion? When this order comes from boss, tester simply focuses on task completion and not on the test coverage and quality of work. There is huge list of tasks that you need to complete within specified time. This includes writing, executing, automating and reviewing the test cases. How will you take decision which test cases should be executed and with what priority? Which tests are important over others? This requires good experience to work under pressure.Application development methods are changing rapidly, making it difficult to manage the test tools and test scripts. Test script migration or reuse is very essential but difficult task.testing methods, approaches and strategies for automation testing areconsidered state-of-the-art, automation testing methods, approaches and strategies are currently used bytesting consulting firms, potential for reuse of automated test cases between different testingconsulting clients and domains, problems exists in regards to testability in customer projects, How can the automation testing methods, approaches and strategies be transformedand combined in order to be more flexible in the dynamic environments of consulting firms?

V. TEST CONSULTING

In general, consulting firms employ persons which have strong competence in a specialized knowledge domain and provide services to customers within this domain. The main characteristic of a consultant is that the domain knowledge is generally greater for the consultant than for a regular employee in the customer organization. Contracts are signed by the consulting firm and the customer that describe the services which shall be provided by the consultants. A general fact is that such contract is signed for a particular time period. If there is time left after the task has been completed, it is usual for the consultant to suggest additional tasks to be performed thus filling the gap in the contract. The opposite, however, can be difficult since it can be hard to persuade the customer to accept additional task that would affect the original contract or project deadline when it is discovered that additional services is needed.

VI. ROLE OF THE CONSULTANT

The primary responsibility of the consultant is to deliver the services described by the contract within the given time frames. Before an assignment starts, the consultants often make a plan based on the expected parameters of the particular project. Glass recognizes this as a problem since it is hard to know the exact valid conditions for the particular customer domain [Glass98]. It is important to realize that the plan may need alterations later on due to unexpected issues that can come up at the customer site.In theory it is sometimes appropriate to guide the customer in directions that differ from the original problem statement. In other words, there is sometimes a need to convince the customer that the actual problem is something else than was originally contracted. In practice this can be a sensitive and hard issue to approach as a consultant. Some clients may feel uncomfortable for an external person to tell them what to do. However, these persons are also aware of the fact that problems do exist since the consultant contract has been initiated in the first place.

VII. AUTOMATION TESTING

Not many of the customer organizations have gotten very far in the field of automated software testing. It has been identified that this does not come from lack of developer knowledge of automation but from higher level management that expect shorter development time for each new project. Test automation is not scheduled since this is expected to increase the development time. Creating automated test cases will cause initial development overhead but compared to manual execution, overall execution time will be decreased for each test case regression. This fact is often not taken into consideration when estimating the test execution.One interesting issue related to the automated testing practices at a customer is that the developers had recognized a need for a commercial automated testing tool in several projects but that the project manager did not want to spend project resources on a tool which were to be used over the entire organization. This way, the purchase of the tool was postponed with the intent that it could be bought in the next project where there were more resources. However, the same problem of course occurred here as well. What can be learned from this is that such purchases should be brought up to the organizational level so that the tools are indeed brought into the organization. On the other side, it should be noted that careful tool selection is needed to avoid the bias imposed by commercial tool vendors which only displays how easy it can be to test certain items. Most often, the tools needs to be complemented with manual testing due to missing features such as the inability to test several applications sequentially. For example, a test application may be able to do a system test on an application that adds data to a database while it lacks the possibility to launch a test script that checks the actual database contents in sequence which could be appropriate for the test case to be complete. In such case, a manual effort is needed for the test script. Such chains of interactions are needed in system and acceptance testing since the levels test implemented behavior which can be spread over several applications. This does not mean that the test application should not be purchased but it does mean that it shall be noted that such features is missing in the tool.A typical scenario for automation is that the customer has some existing sets of manual test cases that they want to automate in order to save resources in form of manual testers. Usually, this type of assignment starts with a workshop where the customer and consultant sit down and discuss which of these test cases would be appropriate to automate. Test selection is used and the test cases up for automation are eventually prioritized. Factors such as how tiresome the manual test cases are to do manually, how prone they are to change and if they even can be automated is considered when doing this prioritization.In most cases, it is not possible to automate every of the manual tests and the strategy instead is to write executable test cases to get large system coverage. A technique that is used by the automation consultants is partition testing where similar tests are gathered into collections that corresponds to different parts of the system under test. This way, each part of the system gets some sort of testing which is considered better than to focus the testing efforts to some single component. However, for some customers there are critical components that needs to receive higher priorities and in these cases, partition testing is not the most appropriate way to go. In these test cases, it is usual to include techniques such as boundary checking. Furthermore, a data-driven approach is often attractive for the test cases.The programming language used when doing automated testing depends on the language used in development. Many projects are web based which leads to languages such as C# and Visual Basic. Ruby is another language that is common in web based testing due to its possibilities of testing code written using other languages.It has been recognized by the consultants that it is important for the test automation that testability is designed into the software. These assignments sometimes require the software to provide so called software hooks so that the test cases can interact with through these hooks to verify that a given input gives the correct output. If such issues are not taken into consideration in the design it is hard to automate tests for certain components and application types.It has been recognized that the management at the customer site often require statistics about the progress of the automated testing which is a sign that they are involved in the process to some extent. However, the management tends to view the number of test cases as a good measurement of this progress and not the quality of them which would be a better measurement.

VIII. AUTOMATION TESTING STRATEGY

The strategy consists of three main phases; Preparation, Execution and Post execution phase. Note that the strategy phases are independent of the development methodology and should not be confused with the development phases of the current methodology. Each phase of the strategy has a set of tasks, each of which is responsible for some aspect of the total quality assurance process. The purpose of the strategy is to increase the efficiency of the testing, both within the current project as well as other projects in the consulting organization. The testability and stability are the first targets of the strategy as can be seen in the preparation phase which is necessary in order to facilitate the test automation which is done in the execution phase. The execution phase is where the actual testing is performed and where the test methodologies are adapted to the current situation. Metrics should be collected during execution so that the test results can be documented and reported to the management. The metrics also serve as a means of strategy and guideline improvement which is done in the post execution phase. Note that this phase also includes knowledge reuse which aims to improve the total knowledge within the consulting firm so that the experiences collected by the individual consultants are shared. The figure illustrates that the strategy phases are iterative which means that the actual strategy is iterative even if the current development methodology is sequential.For example, if the project uses the traditional waterfall model where the software is delivered to the consultants in the test phase, the testability and stability focus in the preparation phase should target future releases from the organization since it is too late to affect this for the software under test. When the tool selection step is started, the current testability and stability should be assessed so that it can be determined what related problems the tool needs to circumvent in the execution phase.

IX. PREPARATION PHASE

The phase consists of three steps; Project testability and stability, Customer training and Tool selection each of which will be introduced in this section. Many factors can impact the testability and stability in late phases of development such as complex design and ambiguous requirements. In order for the customer to get full value of the consultant services they need to facilitate testing by considering such factors in early development. The preparation phase is the first step of the strategy which begins with customer preparation since it aims to affect the testability and stability of the customer project in a positive direction. This can be considered the hardest step in most situations since customers seldom realize the importance of early testing activities, testable requirements and testable design which makes it hard but necessary to influence this. In the traditional waterfall model for example, it is hard to affect the testability and stability of the current release when it has been delivered to the testing phase. However, the preparation phase of the strategy may be used to affect the testability and stability of upcoming releases or projects since it is designed to increase the test process maturity of the entire customer organization. The first subsection focuses on the testability and stability of the software development projects and gives advice on how this could be increased. This is followed by the customer training step which may be required if the customer organization has agreed to adopt certain guideline pointers. As can be seen in Figure 8, these two steps are optional since it may not be possible to influence the customer organization for one reason or another. The final subsection deals with issues that should be taken into consideration when choosing among different automated tools for the test execution.

X. EXECUTION PHASE

The execution phase contain four steps; Test selection, Metric selection, Method tailoring and Test execution and measurement. The execution phase contains pointers which should be considered when starting the actual test execution. The test selection is important for quality assurance purposes since it affects the quality and test execution performance if unimportant items are selected for testing. In order to judge the effectiveness and efficiency of the current testing practices, they need to be quantified and measured. To do this, a relevant set of metrics should be collected during test execution. These metrics should later be used as an indication of the current quality of the software product. Furthermore, they also serve as a means of improvement for the customer guidelines. In this phase, execute the tests and measure the results in regards to the selected metrics. It is important to ensure that sufficient data is collected for each relevant metric so that is actually can be used in the metric evaluation step in the post execution phase.

XI. RESULT

This contains discussions regarding the proposed automated testing strategy.

Risk-based testing gives higher priority to testing the elements that are most at risk of failing which also carry the greatest negative consequences if said failure occurs. Here you should take into consideration:

- The financial impact of potential errors
- Probability of failure (here it is a good idea to ask the developers what they think)
- Service level agreements (SLA).
- Is there money or are there lives at stake? (Yes, this may seem dramatic, but there are several systems that deal with such important matters.)

This should give you a good method for prioritizing which test cases to automate.The applicability of an automated testing strategy within the consulting domain was considered to be low by an automated test consultant at test firmy due to the changing parameters in the different customer environments..Customers need to follow the guidelines given by the consultant for better development of the product. However, the industrial validation showed that it would be difficult to use the guidelines in this manner so these were modified in order to be used through a test manager from the consulting firm. This way, the consultant test manager can convince the developers to adopt the pointers and train them accordingly. A regular visit at the customer site is required to convience the client to adopt the guidelines.Furthermore, if the project managers are serious about the quality assurance process the test manager did not see further difficulties in applying the guideline pointers in order to facilitate the system and acceptance test.After consulting the client the guidelines can be understood.

Since system testability and stability was considered to be the main challenges in most situations, the consultant test manager perceived the focus in the preparation phase as credible. Furthermore, it was also mentioned that strong motivations of the guidelines used in this phase was needed in order to convince the developers and managers in the customer organization Testability and stability to be of great importance for the system testing, acceptance testing and overall product quality. Furthermore, the importance of having people in the organization which are test-oriented since the testing practices may not be adopted otherwise

Unfortunately, the use of an automated testing strategy in the consulting domain was perceived as difficult by the automated test consultant due to the changing parameters at different customer sites. However, this has been taken this into consideration in the design of the automation specific parts of the strategy and thereby they have been designed for practitioners who move between different customer domains and development phases. Therefore, it is suggested as future work to include a dynamic validation of the strategy so that the feasibility of the strategy can be addressed from a live industrial perspective.As described throughout the thesis,the primary focus has been to produce an approach suitable for a consulting setting. In this setting, the application domain is expected to change due to the consultant movement between customers. To achieve this goal, the strategy and guideline pointers have been generalized to the extent that they can be adopted independent of the current parameters in the customer domain. Due to this design, the approach may be transferred to ordinary development settings where the application domain is static. However, in this case it could be appropriate to extend the guidelines with more specific pointers for the particular domain since the strategy no longer needs the flexibility that the generalization provides. This kind of extension is possible due to the dynamic structural design of the guidelines. The customer validation showed the importance of having at least one person who is aware of the quality benefits that testing provides. Since the preparation phase in the strategy involves a consultant test manager which motivates the guideline pointers to the software practitioners and managers, a similar person is needed in the traditional organization as well. Without such person, it would be hard to cross the threshold described in Section 7.2.1 and this could infer a problem if the strategy and guidelines is needed in an organization which currently have low test process maturity. Furthermore, since the proposed strategy and guidelines refer to academic studies in the pointer motivation sections it can be hard to transfer the approach to organizations which are not susceptible to the results made by such studies.Also, as previously described, the preparation phase is dependent on a consultant test manager who is responsible for motivating the guideline pointers. If a similar person who can take this responsibility is not available in the organization that is about to adopt the strategy, it could result in a low adoption level of the pointers.As illustrated in Figure 3 (Section 3.1), the study started with a large literature survey which formed the initial point of view of the author regarding the current state-of-the-art within the field of testing and more specifically automated testing. Since the author has limited experience in research evaluation, this can pose a threat to validity because it is possible that the evaluated studies have limited relevance for this particular thesis project. This threat was handled through the academic validation described in section 3.5. As described, some modifications were needed and have been implemented since the discussion with the academic researcher. Furthermore, the concept of the strategy and guidelines was considered feasible. However, further research validation was suggested for the metrics and requirements pointers and these pointers need further confimability to ensure academic relevance.This section will revisit the research content that were initially formed in the early phases of the thesis project. Section 7.3.1 provides a flowchart of how the questions were answered and more elaborated work is provided in Section 7.3.2.As described in Section 3.2, a literature survey was done in order to find research relevant for the consulting domain. Through this survey, eight relevant state-of-the-art approaches were found which was deemed relevant for use in this domain. The primary focus was to find automated testing methods suitable for use in the system and acceptance testing levels. However, since the consultants act in unit and integration testing levels in some situations it was necessary to introduce methods which can be relevant for these situations as well. This led to the summary of three black-box techniques, three black-box/white-box hybrid techniques and two frameworks for unit testing suitable for both agile and plan-driven development settings. More deep going summaries of these can be found in Section 2.5.Since the most common test levels are system and acceptance test, it is important that the automated tools have sufficient support for testing system level requirements. For example, tools such Selenium Webdrive for testing web based application. In general, it is important that the tools are flexible enough to support custom modifications to the test cases. Script languages such as JAVA have also been recognized as valuable due to its capabilities to test modules written using other programming languages. Furthermore, it has been recognized that partition testing is suitable in the cases when full test case automation is not possible since this can be used to measure how much testing each system partition has received.Since the customers pay for the automated test case creation, these test cases are not allowed to leave their organization. This means that there are no possibilities for this kind of reuse. However, it has been recognized that the test methodology is constantly reused by each individual consultant among different customers. The proposed strategy has taken this a step further in order to allow knowledge reuse from an individual consultant to the rest of the available consultants in the consulting firm. Such reuse benefits the consulting firm since the knowledge set and experiences of each individual can be transferred which in turn increases the total knowledge in the organization.

Since there are often problems with low testability and stability due to low test process maturity in the customer projects, these problems must be handled prior to the consultant assignments. This in turn can be solved by introducing customer guidelines which targets the most common test process problems in the organizations. A dynamic structure is needed for the automated testing strategy and customer guidelines for them to stay efficient. The suggested pointers in both the strategy and the guidelines need to evolve when the test process maturity increases in the customer organizations. This can be handled by modifying, removing or adding pointers to the current set due to the dynamic structure. Furthermore, it is important that the pointers of the strategy and customer guidelines are supported by either previous consultant experiences or empirically evaluated studies which prove their worth. Otherwise it can be hard to convince practitioners of their value in their domain.

Most is on management they should have a quality development process in place, a contingency plan in place in case of problems, they should understand that testing is only a evaluation activity.

REFERENCES

[1] [Dijkstra72] Dijkstra, E. W. (1972). Notes on structured programming. In Structured Programming, eds. Dahl, O.J., Dijkstra, E.W., & Hoare, C.A.R. London: Academic Press.

[2] [Miller01] Miller, R. W., & Collins, C. T. (July, 2001). Acceptance Testing, Procs.XPUniverse.

[3] [ObjectMentor01] ObjectMentor, Incorperated. (2001-04). JUnit, Testing Resources for Extreme Programming.

[4] 21st IEEE/ACM International Conference on Automated Software Engineering (ASE'06), 365-368. [Trochim06] Trochim, W. M. (October 20, 2006). The Research Methods Knowledge Base (2nd Edition).