International Engineering Journal For Research & Development
Vol.2
Issue 6

E-ISSN NO:-2349-0721

Impact factor : 6.549

AUTOMATION TESTING IN AGILE METHODOLOGY

Mustafa Shuaieb Sabri
Department of Information Technology

ABSTRACT

Automatic testing and automation in the open are essential. Automation in Agile method has gained popularity since its development and continues to be used by software developers worldwide. The widespread use is mainly due to its speed, unequal return on investment, reliability, similarity, and redundancy, allowing developers to develop high-quality software in a short amount of time and with limited functionality. The most widely used aging methods today are XP (Master Plan) and SCRUM. Others include Kanban, the development of trust, and Crystal's methods.

Keywords: Automation, Agile Methodology, Software Development, Technologies.

INTRODUCTION

According to Mohammad (2016), testing in older projects is an integral part of the development process and is not a separate component of the Software Development Life Cycle. Therefore, Gao et al. (2003) argue that automation testing software defines the functions assigned to perform engineering tasks in the software development phase of the software development process, using well-defined solutions and advanced programs and strategies. Test automation is beneficial for projects with end date windows due to test expansions increasing, and the test performance speed being too high for automatic testing (Dustin et al., 1999). Crispin & Gregory (2015) argued that tests are essential for success in software development and are the heart of liver testing (Taipale & Smolander, 2006). Test automation in projects allows for effective blurring and rapid delivery of value, by project management teams as long as they can find a way to reduce the risks involved in implementing test automation (Mohammad, 2015). This paper's main objectives are to identify the various challenges faced by high-level professionals while adopting experimental skills in applied methods based on results from basic research conducted on 40 older professionals and strategies to overcome those challenges. This paper will provide an insight into the challenges you may face while using the automation test and the best ways to overcome them.

REVIEW LITERATURE

The term "agile" was coined by 2001 by seventeen software developers at a conference discussing the future of software development and incorporating their equipment technicians who had similar symbols into what is now the agile manifesto. The agile manifesto emphasizes people, social interaction, performance software, customer interaction, flexibility, and satisfaction, rather than creating processes, tools, and methods (Kent et al., 2001). A study by Gandomani & Nafchi (2015) presented a framework for rapid change and acceptance using people using the "Strong Participatory Approach," and participants of 49 senior professionals from 13 different countries abroad. Their main goal was to develop a simple and flexible framework for a flexible acquisition

environment. The findings show that the structure is more efficient for companies and organizations due to its slower, slower, more sustainable, and profit-based approach in line with the Agile approach. Papadopoulos (2015), in its study "From old to old software development and to large, distributed projects," highlighted the significant benefits of new software development approaches by building better relationships and relationships between software development teams, thereby improving metrics. Papadopoulos, (2015), used case-control analysis based on metric quality limits and user satisfaction in adopting new methods to combat those who still use traditional methods. These three review reviews provide the basis for what the research topic aims to accomplish. The identification of potentially problematic literature is made by researching the most suitable resources to identify research reports in software engineering. This includes searches in the HU library, IEEE Explore, and on the Web for Science. Combined, these indicators cover almost all relevant scientific journals, magazines, and conferences related to software development.

EVALUATION

To verify the search query and data selection, we used a specific publication number. IP9 was missing in the data search result. The reason may be a development document that focuses on and provides a framework for process development, based on empowering data and dynamic framework testing. Empirical research is presented in the paper, but it is not the main focus of the report.

The journal Testing Software, Verification & Reliability, and Advances in Software Engineering sought out the same papers and inclusion terms. The papers found in that search and those identified by back and forth links are listed in the database and subsequently controlled by the same analytical process as the standard search threads.

The collection from the papers varies with the related variety, type of reading, and quality of science. We have decided to keep strictly non-ethical documents as reports of simple experience. This ensures that the findings and potential trends in the new field of research are not initiated early. We also identify topics and concepts that have not been investigated yet, but that should be included in future research. The analysis of selected books was encouraged by the practice of regular comparisons where the concepts in the middle of the topic emerged through repeated analysis of the revised editions.

There are many different types of software testing, namely, (1) Blackbox testing where the tester can determine the software's internal structure under scrutiny. This approach incorporates techniques such as boundary value analysis, equitable classification, and output results. (2) Whitebox testing where the internal structure of the software under test is known

to the examiner. Includes techniques such as data flow monitoring method, flow control testing, method testing, and branch testing. (3). The software's internal structure is known in part by the tester, and the test is performed at the user level. (4) An Ad-Hoc Examination, which, as the name implies, is a non-compliant examination. (5) The Agile Test we are talking about. Agile Testing was initially described by Brian Marick (Marick, 2003) and later became famous for Janet Gregory and Lisa Crispin (Crispi & Gregory 2009). Traditional testing methods, such as the arrival of water, include two-stage and two-stage processes in which the development team builds software near pre-test and bug fixes in those phases late. Agile testing, however, allows developers to identify and correct errors that appear faster than waiting for a project to be completed. According to the 2016

Worldwide Testing Practices Report, older methods have grown in popularity and use. Other methods used by agile include; flexible software development (ASD), performance enhancement (DevOps) aging service delivery (DAD), dynamic system development (DSDM), feature development (FDD), growth and expansion development (IID), Kanban, Lean SD LeSS, model-driven engineering (MDD), Microsoft solution framework (MSF), personal software system (PSP), rapid application development (RAD), integrated integration process (RUP), framework Comprehensive (SAF), Scrum, team software (TSP), and XP. Many future experts have different approaches to testing, e.g., diagnostic testing, performance testing, and automation testing, which are based on this research. The automatic testing has many benefits, as explained in the next section.

STUDY SUPPORT

The first is cheaper than other testing methods as the cost is only available once compared to other non-existent methods where the money is spent every time a test is required. Second, it is faster than other manual testing methods, especially if the task to be completed is awkward and tense. Third, the automatic checker is free from global errors and, therefore more reliable. Finally, automation testing confirms the concern of free growth software. There is no risk of recurrence involved than non-compliance with manual testing where testers working on those projects could die or leave the client in a very precarious position. However, the whole system comes with challenges, and automation testing is no different, thus creating this research. It is essential to understand those challenges and develop critical solutions to allow developers to increase the cost of developing product plans, help reduce time-consuming code-based involvement, whenever there is an error, and also bring project results in the shortest possible time. Automatic aging testing has the full meaning that with automation you can do the same test as many times as you want, with multi-purpose test details and allow testing to be done in different locations.

RESEARCH DESIGN AND RESEARCH METHODOLOGY

It is a systematic form of research widely used in scientific (social sciences) research that involves the formation of ideas through a robust set of actions and data analysis (Glaser, 2010; Hoda et al., 2010). The great idea is to research not on a particular theory that has ever existed, but, the method says to emerge as the research process progresses. It therefore becomes the final product of the communication of data collected and analyzed data. There are three versions known as GT, i.e., Glaser GT, Straussin GT, and Charmaz's constructivist GT. This study will use Glaser GT as the objective under review based on the respondents' experience based on their interaction with the project management software, and therefore, authorizes us to exclude any previously accepted comments as it may constitute a preference. The aging process is also customer-centered and interactive. The Glaser GT is therefore helpful in studying social interactions and behaviors in humans. Glaser GT is widely used in long-term study, and it happens that the field of automatic testing in a relaxed environment is still new and useless. The final reason why Glaser GT is so active is that it encourages the careful study of older groups as seen in previous research, conducted by Dorairaj et al. (2012), and Corbin & Straus (2015).

CHOICE OF SAMPLE COLLECTION AND TRANSFER OF FAILURE

The study involved 38 participants (P1-P40) from 20 different companies based in the USA, with international relations such as India, New Zealand, Australia, UAE, and the UK and regional linkages to western Europe and Southeast of Asia. Each participant had a group of between 7 and 20 people involved in projects from the following domains; finance, insurance, leading sales, network management services, e-commerce,

telecommunications, health, power solutions, IT infrastructure, IT and Agile training and Banking. In evaluating participants participating in the Agile India International Conference of 2016 (Mohammad, 2016), it is possible to use the findings as a platform where participants can be integrated with them in partnership with the certification of competence XP & SCRUM applied methods. DevOps has been leading the IT sector providing efficiency to the workloads (Mohammad, 2016). The following are the corporate roles expected of those involved; Agile Trainers (ACs), Chief Technical Officials (CTOs), Devops (DOs), Engineers (DEVs), Senior Sponsors (SDs), Product Owners (POs) , Senior Agile Trainers (SACs), Scrum Masters (SMs), and senior leadership such as the company's Vice-President (VP), senior quality analyst (SQA), senior inspectors (STs), Tests (TAs), testers (TES).

ABOUT THE POTENTIAL FINDINGS

Test automation is an essential component of software development projects. First, we expect that this study's findings will empower illegal traffickers around the world to take appropriate automatic action (Collins & Lucera, 2012). They will know the basics of the project before starting it. This will help to avoid wasting time on losses and loss of default and accuracy. Secondly knowing that the challenges associated with changing testation with their solution strategies will help older staff because, they can be solved before they become a problem and thus testing will always yield positive results as long as the proposed strategic guidelines are followed strictly.

Also, cutting off these issues from the name will ensure faster delivery of support applications, which ensures customer satisfaction. That is the main goal we are all striving to achieve. Lastly, I believe that the proposed solutions to those identified challenges will be even more critical when developers face more dynamic projects. This means that changing the requirements will be much more comfortable, and the fact that automatic testing is a big plus. This study's findings will be of great help in project management by providing value from scratch in a continuous manner. Managers will see real progress and any potential risks to the business or organization at the beginning of the phase. Effective decision-making is another critical benefit of project management. The challenges and techniques of automation testing will lead to learning among the developing team involved in the project, allowing them to gain a critical understanding of the process as a preparation process when project management leadership begins. A full knowledge of the results will ensure that they are successful in the projects they are undertaking as they will have a guide on which roads to take and which ones to choose.

DEFINITION OF STRONGER ACTIVITIES

Test automation: In IT management software testing, automation test refers to installing a separate software program that is intended to test the performance of test processes, to easily compare actual results with expected / predicted results.

Code: a summary of crucial point phrases in the data for relationship sharing

Iteration: The time (about two months or less in a period) when the aging team produces a completed software extension.

Sprint: in software development, a speaker is a set time during which specific software development tasks must be completed and presented for review.

CONCLUSION

This paper proposes to research to assess the challenges experienced by older staff while adopting and implementing test automation in earlier projects in the project management phase and other interventions that can help overcome the problems mentioned above. It will use The Ground Theory research process to form the basis of a research framework for research. The GT method's limit is that the findings will be tied to a specific condition studied in the study. We will collect data from two sources to access reliable data, namely, interviews and customized views. The nature of the research is guided by the availability of older professionals who participated in the study, and all the findings. from this study it will be appropriate for earlier projects that use automation tests and may not work globally. The main contribution of this study will be to understand the significant barriers that can be encountered by any high-level employee who wishes to use the automation test tool in the process and provide the best strategies he or she can use to overcome these challenges. The findings will help emerging professionals successfully pass their exams

REFERNCES

1. Collins, E., Dias-Neto, A., & de Lucena Jr, V. F. (2012, July). Strategies for agile software testing automation: An industrial experience. In 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops (pp. 440-445). IEEE.
2. Collins, E. F., & de Lucena, V. F. (2012, June). Software test automation practices in agile development environment: An industry experience report. In 2012 7th International Workshop on Automation of Software Test (AST) (pp. 57-63). IEEE.
3. Corbin, J., Strauss, A.: (2015) Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory, 4th edn. Sage, London (2015)
4. Crispin, L., & Gregory, J. (2009). Agile testing: A practical guide for testers and agile teams. Pearson Education.
5. Dorairaj, S., Noble, J., & Malik, P. (2012, May). Understanding team dynamics in distributed Agile software development. In International Conference on Agile Software Development (pp. 47-61). Springer, Berlin, Heidelberg.
6. Dustin, E., Rashka, J., & Paul, J. (1999). Automated software testing: introduction, management, and performance. Addison-Wesley Professional.
7. Gandomani, T. J., & Nafchi, M. Z. (2015). An empirically-developed framework for Agile transition and adoption: A Grounded Theory approach. Journal of Systems and Software, 107, 204-219.
8. Sikender Mohsienuddin Mohammad, "AN EXPLORATORY STUDY OF DEVOPS AND IT'S FUTURE IN THE UNITED STATES", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.4, Issue 4, pp.114-117, November-2016, Available at :http://www.ijcrt.org/papers/IJCRT1133462.pdf or http://doi.one/10.1729/Journal.24201
9. Gao, J., Tsao, H. S., & Wu, Y. (2003). Testing and quality assurance for component-based software. Artech House.
10. Glaser, B. G. (1992). Basics of grounded theory analysis: Emergence vs forcing. Sociology press.
11. Glaser, B. (2010). Grounded Theory Institute: Methodology of Barney G. Glaser.
12. Gregory, J., & Crispin, L. (2014). More agile testing: learning journeys for the whole team. Addison-Wesley Professional.
13. Hoda, R., Noble, J., & Marshall, S. (2010, June). Agile undercover: when customers don't collaborate. In International Conference on Agile Software Development (pp. 73-87). Springer, Berlin, Heidelberg.
14. Sikender Mohsienuddin Mohammad, "CONTINUOUS INTEGRATION AND AUTOMATION", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.4, Issue 3, pp.938-945, July 2016, Available at :http://www.ijcrt.org/papers/IJCRT1133440.pdf or http://doi.one/10.1729/Journal.24061
15. Manifesto, A. (2001). Principles behind the agile manifesto. Retrieved July, 23, 2005.

16. Marick, B. (2003). Agile testing directions: tests and examples.
17. Papadopoulos, G. (2015). Moving from traditional to agile software development methodologies also on large, distributed projects. Procedia-Social and Behavioral Sciences, 175, 455-463.
18. Taipale, O., & Smolander, K. (2006, September). Improving software testing by observing practice. In Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering (pp. 262-271). ACM.
19. Sikender Mohsienuddin Mohammad, "AUTOMATION TESTING IN INFORMATION TECHNOLOGY", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.3, Issue 3, pp.118-125, August 2015, Available at :http://www.ijcrt.org/papers/IJCRT1133463.pdf or http://doi.one/10.1729/Journal.24200