ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

Antecedents of Software Quality Assurance Framework: An Exploratory Study

Dr. Debashree Souvik Jana, Assistant Professor, Dr. D. Y. Patil School of Management, Lohegaon, Pune debashree.aims@gmail.com

Abstract

The world is transforming very fast digitally today. Every IT solutions and service providers want to keep a pace with the increasing demands of right first time and early launch to the market. The market being fluid poses the challenge of changing business requirements and yet expect the IT project teams to deliver in elevation, the quality product or service. Assurance of product and service quality has become focal point of every IT project managers and the project teams. Today, the verification, validation and testing is not confined only to the software solution being developed, but it has been adopted even from the requirement refinements and formalisation. A framework covering various factors to enable quality assurance aspects is hypothesised to bring benefits in terms of fail fast, savings and test effectiveness.

On one side, the customers are the final authority to gauge the reliability of the software being built. While on the other side, it is a prerequisite for successful project management to execute within the project budget and accommodated higher volume of changes in the volatile business scenario, yet managing the higher product quality for customer delight. Manging the product budget can be achieved through savings and reusability, still balancing the quality aspects of the product through right mix of verification, validation and testing.

The researcher thus carried out the exploratory study to understand and analyse antecedents of quality assurance framework which corroborate with the multiple benefits. A digital transformation program was chosen which has been delivering digital products and services for the citizens of a geographical region in Europe. Multiple projects delivered last five years were studied to understand how engaging customers early , automation and multi-level testing mix enabled fail fast to fix fast, savings and reusability through automation and gain test effectiveness through right mix of levels of software testing.

While shift left of customers helped the project team identify the issues and defects early in the project life cycle helping to fix them early, the adaptation and reusability of test automation helped in reducing the cost of testing year on year. The study also revealed that issues and defects revealed post product launch was higher when customers were not involved from project initiation stage. The cost of testing increased substantially when automation was not implemented in the projects under study. Higher Test effectiveness of 70% and above was achieved only when the right mix of levels of testing was used.

Conclusions relate to the benefits realised in implementation of the three factored framework for quality assurance. Reduction of live issues and defects with reduction on cost imperatives through automation encodes the direct benefit realisation of the project management team. The third dimension of testing right and measure of test effectiveness is achieved through the right proportion of levels of testing.

Key Words Project Management, Quality Assurance, Project Management, Digital Transformation projects, Software Testing, Verification and Validation**.**

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

Introduction

The project managers globally face challenges in terms of engaging customers in the project life cycle i.e. when to engage to get more benefit, budget allotted for entire project and attaining savings, effectiveness of quality assurance are one of the key challenges. Assurance of quality which is commonly known as quality assurance of software being developed by project team, is critical to ensure reliability of the software being developed. Software quality assurance plays and important role in the journey of digital transformation (Sunil Shegal, 2016). The ultimate key stakeholder of any project undertaken for digital transformation is the customer. The customer is ultimately who decides when a project or a feature is finished and when the quality is acceptable (Olausson M., Rossberg J., Ehn J., Sköld M. 2013). To rollout the software in the market early, the project manager and the team will need to continuously build the software and continuously verify, validate and test the features and functionalities.

In digital transformation programs and projects, continuous verification, validation and testing is the process of executing automated tests. This forms the part of the software delivery pipeline which helps in immediate feedback of the quality of the software (Adam Auerbach, 2015). Implementing test automation as one of the factors of quality assurance framework and gaining how much savings as a benefit is a question every project manager will ask himself. This study reveals the answer to this ask on return on investment as savings through automation. The quest for ensuring effectiveness of quality assurance is another challenge the project manager faces today in digital transformation project. Project team not only needs to verify the features and functionalities but also need to assure the reliability of the software being developed. Project managers thus introduces various types of testing such as functional testing, non-functional testing, accessibility testing, cross browser and device testing. Each of these levels of testing helps the project team to assure the quality of software being developed. Functionality testing will verify the business requirements with respect to specifications met by software under test (Prasad, Dr. K.V.K.K., 2008). Non-functional testing is conducted to assure that the non-functional requirements such as performance of the software developed is met as expected (Anon, 2019). Tim Berners Lee, W3C Director and inventor of the World Wide Web, has stated that "The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect" (Mary-Luz Sánchez-Gordóna, Lourdes Morenoa, 2014). It is thus prudent to verify the accessibility aspects of the software being developed by the project team, before it is launched to the market. The other

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

aspect that a project manager is challenged is to ensure that the software built is compatible across multiple browsers and devices, especially when the software product is web based. Cross browser compatibility of the software is one of the major challenge among the developers (J. Rode1 and M. A. P.-Q. Mary Beth Rosson, 2002) and hardly gets importance during software development. The project manager will need to consider the compatibility aspects while building the software so that it can be used with same behaviour across 100 browsers available in the market today (Anon, 2021). However, how much percentage of using each of the levels of testing to gain higher test effectiveness is another challenge for the project manager. It was thus required to study and understand how the three factors of shift left customers, automation and multi-level testing mix influences the higher degree of quality assurance.

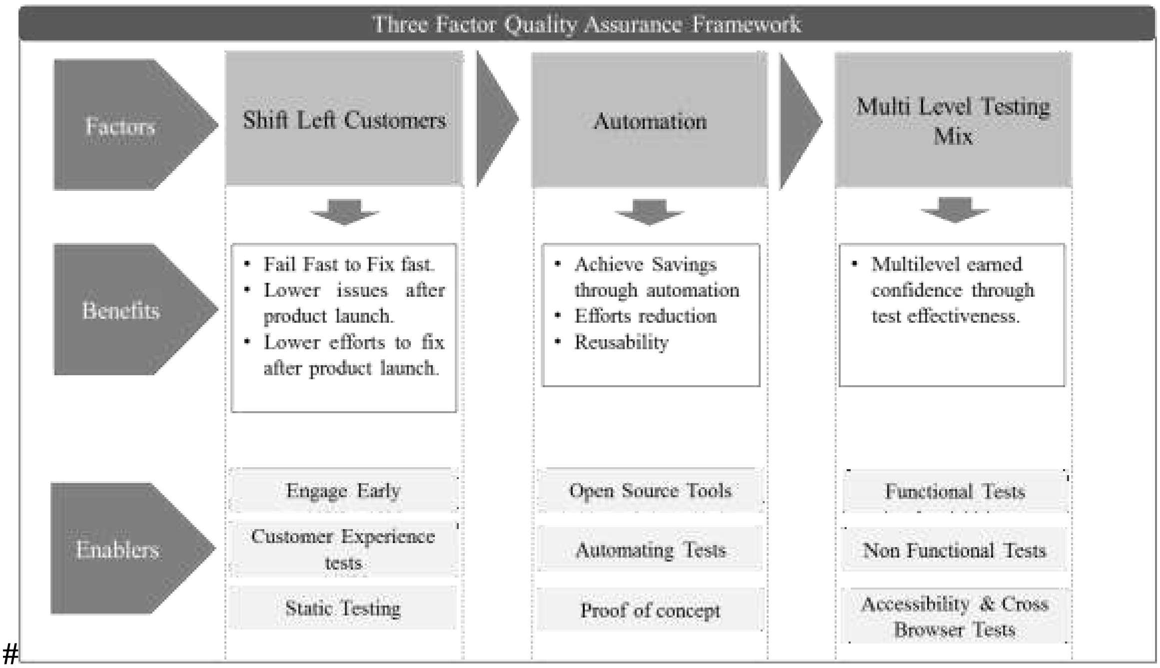
Antecedents

The digital transformation across the globe has pushed the organisations to bring their digital product and services faster to market. Many IT firms and project managers approach the process of testing differently, while many focus on product features and functionalities in addition to hunting issues and defects (Glenford J. Myers,1980)**.**The IT Project managers need to bring the best breed processes, technologies, tools and methodologies to build these digital enabled products and services with right first time. Testing can be used to improve quality (Iacob, Constantinescu, 2008) as suggested also gives an insight that verification, validation and testing does not guarantee quality. However, failing to produce the product or service with highest quality, will entice loss in business in the long run. Hence, this study is conducted to verify if the three factors forms the right quality assurance framework, to bring in multiple benefits.

Three factor quality assurance framework is an amalgamation of critical aspects that proved vital for the scope undertaken for the study. The framework as depicted in the figure 1, was built based on experience and experimental initiatives by the author.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022



#

Source: Author

Figure 1: The three factor quality assurance framework for building right first time solution.

The framework is based on three main factors, shift left customers and users, automation - aligning the right tool box, multi-dimensional verification, validation and testing. Each of these factors brought in various benefits, compounded to achieve a higher standards of digital product and services quality.

Shift Left customers:

Conceptualisation and ideation of the digital product and services always takes place from the problem statement and business scenario that can be well defined by the customers and then by users. As echoed by (Frank Philip Seth, et al , 2014), the project team often faces the challenge of not able to interpret the business environment the customer is in and other scenarios that are not fully described in often inadequate and changing requirements (N. Ibrahim et al., 2009)**.** It is thus important to engage the target segment of the products and services from requirements capture stage of the project. Agile delivery framework has enabled quicker development and delivering value to customers in an incremental way (Richard Knaster - et al., 2019). To leverage from the agile principles of engaging the customer early, the project manager with the help of program board need to on-board customers early. This can give multiple benefits which can be proved through historical year on year data for a particular digital transformation program covering multiple projects of digital transformation initiatives.

IJFANS INTERNATIONAL JOURNAL OF FOOD AND NUTRITIONAL SCIENCES

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

For the purpose of our study and to test the hypothesis of shift left customers results in multiple benefits (Fail Fast to Fix fast, lower issues after product launch, lower efforts to fix after product launch), author collected and analysed the year on year data for all the projects completed for the program. The data and analysis as summarised in the table 1, revealed that whenever customers were involved at early stages i.e. shift left in the project life cycle, the issues and defects in the requirements, design and software in itself were detected early. These issues found early enabled fail fast which in turn enabled development team to fix early in project life cycle. Doing this the issues and defects found after product or service is launched, drastically reduces since there is rigour in quality checks at early stage itself.

			Initiation Stage			Build & Implementation	Post Implementation	
Year	Count of Projects	Customer's involvement in Qulity Assurance from project stage	Project Complexity	Total Requirements (user stories)	Weighted Issues/Defect Density (%)	Weighted Issues/Defect Density (%)	Weighted Issues/Defect Density (%)	
2015-16	2	Not Involved	Low	63	--	--	8.60%	**
2015-16	3	Initiation	High	379	9.00%	13.00%	1.75%	
2016-17	1	Not Involved	Low	42	--	--	5.00%	**
2016-17	1	Initiation (late involvement)	Medium	54	1.20%	9.50%	2.50%	
2016-17	4	Initiation	Medium	137	13.00%	12.00%	0.45%	
2017-18	1	Initiation (late involvement)	Low	29	--	7.50%	1.60%	
2017-18	4	Initiation	Low	121	8.00%	14.00%	0.20%	
2017-18	2	Build & Implementation	Medium	111	2.20%	8.00%	1.20%	
2018-19	4	Involved	High	567	14.50%	13.00%	0.25%	
2019-20	1	Not Involved	Medium	93	--	--	7.50%	**
2019-20	1	Build & Implementation	Low	30	--	6.00%	2.00%	
2020-21	2	Initiation	Medium	132	9.00%	14.00%	0.45%	
2020-21	8	Build & Implementation	High	1423	6.50%	13.00%	0.20%	

Source: Author

*** indicates higher weighted defect density post implementation of product or service*

Table 1: Weighted defect density at each stage of project with respect to shift left

The weighted defect density denoted as 'd' is an industry standard is used as one of the metric to measure the effectiveness of quality assurance. It can be manifested through the data analysis as depicted in table 1 is that, when customers and users were involved from early stage of the project life cycle (initiation stage), the issues/defects are found early for fix. The average weighted defect density across the projects when customers were involved at

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

initiation stage and later in build – implementation stage is d=9%. Such involvement has detected a weighted defect density of 7% for overall issues and defects from the total issues starting from initiation to post implementation. This indicates that due to customer's involvement early in the project life cycle the issues and defects found after implementation has drastically gone down.

The data analysis (as in Table 1), revealed that whenever customers engaged late in the project life cycle the average weighted defect density has gone up. This indicates that more the issues and defects found at later stage of the project life cycle. This is evident from the figures in table 1 where in the years 2015-16, 2016-17 and 2019-20 , customers were not involved in conducting quality assurance of the projects from initiation phase or even did not conduct assurance activities during software build and implementation stage. Due to this an average defect density d = 7.03% is found at post implementation stage which is quite high as compared to when customers involved in quality assurance of the project at some stage initially, where average weighted defect density d = 1.06% after product or service is launched into the market. Another way to look at the data analysis is that when customers are involved early iin project life cycle for quality assurance, they contributed in detection of issues with average weighted defect density of d=8% (at initiation stage) and d=11% (at build and implementation stage). This indicates that quality of the product or service has already improved due to early involvement of the customers.

The project manager and the project team spends a substantial time and efforts in developing the software based on the set of requirements. These efforts also encompasses the issue and defect resolution and fix efforts. The detection of issues at later stage of project life cycle may impact the project schedule and become challenging to go to market early. This is can be manifested through the data represented in table 2.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

					Initiation Stage	**Build & Implementation**	**Post Implementation**	
Year	Count of Projects	Customer's involvement in Qulity Assurance from project stage	Project Complexity	Total Requirements (user stories)	% of efforts spend to fix issue at design stage [De]	% of efforts spend to fix issue or defect at software development stage [Be]	% of efforts spend to fix issue or defect after software is delivered to market [Pe]	*
2015-16	2	Not Involved	Low	63	--	--	13.00%	**
2015-16	3	Initiation	High	379	3.00%	8.00%	2.00%	
2016-17	1	Not Involved	Low	42	--	--	7.50%	**
2016-17	1	Initiation (late involvement)	Medium	54	2.50%	11.00%	3.50%	
2016-17	4	Initiation	Medium	137	4.50%	5.60%	1.20%	
2017-18	1	Initiation (late involvement)	Low	29	--	9.00%	0.75%	
2017-18	4	Initiation	Low	121	4.00%	8.00%	0.15%	
2017-18	2	Build & Implementation	Medium	111	3.30%	9.20%	1.75%	
2018-19	4	Involved	High	567	7.00%	10.00%	0.32%	
2019-20	1	Not Involved	Medium	93	--	--	8.50%	**
2019-20	1	Build & Implementation	Low	30	--	5.50%	1.45%	
2020-21	2	Initiation	Medium	132	7.00%	12.00%	1.10%	
2020-21	8	Build & Implementation	High	1423	9.00%	11.50%	0.65%	

Source: Author

Notations: [De] indicates % of efforts spent to fix issue at design stage by the project team (software developers), [Be] indicates % of efforts spent to fix the issues or defects by project team (software developers) at software development stage, [Pe] indicates % of efforts spent to fix the issues or defects after software is delivered to market by the project team (software developers).

The data represented in table 2 indicates that efforts spent in fixing the software issues and defects after it is launched in the market is high for the projects undertaken in the years 2015-16 , 2016-17 and 2019-20. This happened since the customers were not involved in assurance of quality from initiation stage of the product life cycle. The number of issues found after launching the product or service is high (as indicated in table 1) for these instances. Hence, the efforts by project team (software developers) were high after product was launched. The project team spent substantial efforts to identify the root cause and fix these issues and defects. In this case as indicated in table 2, the average efforts spent by the project team to fix post implementation was Pe=9.67% as compared to Pe=1.29% which is average efforts to fix post implementation when customers were involved in quality assurance before product launch. On the other hand as indicated in table 2, average % efforts spent by the project team is De = 5% and Be = 9% as compared to Pe = 1.29%. This indicates that project team (software developers) have spent more efforts in fixing issues and defects during software design and building stage than after product was launched in the market. The more average

IJFANS International Journal of Food And Nutritional Sciences Official Publication of International Association of Food and Nutrition Scientists

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

efforts spent during software design and building it, would yield higher quality and less issues and defects as well as less efforts to fix issues and defects after product is launched into the market. This can be well manifested through the data presented in table 3.

					Initiation Stage		Build & Implementation		Post Implementation		
Year	Count of Projects	Customer's involvement in Qulity Assurance from project stage	Project Complexity	Total Requirements (user stories)	Weighted Issues/Defect Density (%)	% of efforts spend to fix issue at design stage [De]	Weighted Issues/Defect Density (%)	% of efforts spend to fix issue or defect at software development stage [Be]	Weighted Issues/Defect Density (%)	% of efforts spend to fix issue or defect after software is delivered to market [Pe]	
2015-16	2	Not Involved	Low	63	--	--	--	--	8.60%	13.00%	**
2015-16	3	Initiation	High	379	9.00%	3.00%	13.00%	8.00%	1.75%	2.00%	
2016-17	1	Not Involved	Low	42	--	--	--	--	5.00%	7.50%	**
2016-17	1	Initiation (late involvement)	Medium	54	1.20%	2.50%	9.50%	11.00%	2.50%	3.50%	
2016-17	4	Initiation	Medium	137	13.00%	4.50%	12.00%	5.60%	0.45%	1.20%	
2017-18	1	Initiation (late involvement)	Low	29	--	--	7.50%	9.00%	1.60%	0.75%	
2017-18	4	Initiation	Low	121	8.00%	4.00%	14.00%	8.00%	0.20%	0.15%	
2017-18	2	Build & Implementation	Medium	111	2.20%	3.30%	8.00%	9.20%	1.20%	1.75%	
2018-19	4	Involved	High	567	14.50%	7.00%	13.00%	10.00%	0.25%	0.32%	
2019-20	1	Not Involved	Medium	93	--	--	--	--	7.50%	8.50%	**
2019-20	1	Build & Implementation	Low	30	--	--	6.00%	5.50%	2.00%	1.45%	
2020-21	2	Initiation	Medium	132	9.00%	7.00%	14.00%	12.00%	0.45%	1.10%	
2020-21	8	Build & Implementation	High	1423	6.50%	9.00%	13.00%	11.50%	0.20%	0.65%	

Source : Author

The cumulative average of weighted defect density as customer's were involved early in the project life cycle is d=10% and the cumulative efforts spent by project team (software developers) to fix the identified issues and defects is 7% (average of cumulative De and Be). On the other hand, for the projects where customers were not involved in quality assurance activities , the cumulative average of weighted defect density is 7% and average cumulative efforts to fix the issues and defects is 10%. This comparison clearly indicates that when customers are involved early in project life cycle , efforts to fix issues and defects is less than post launch of the product.

The data analysis across the projects delivered between years 2015-2021 revealed that to enable early quality assurance of the digital product and services, organisation has to engage customers early which is paramount important for the project managers. The issue identification can happen in the early prototype, requirements captured by project team or even during system demo. Static testing methods in addition to witness testing are traditionally adapted by the customers as well to identify issues and defects early. The complete customer engagement early termed as 'shift left', with attribution to quality activities is encapsulated in the figure 2.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

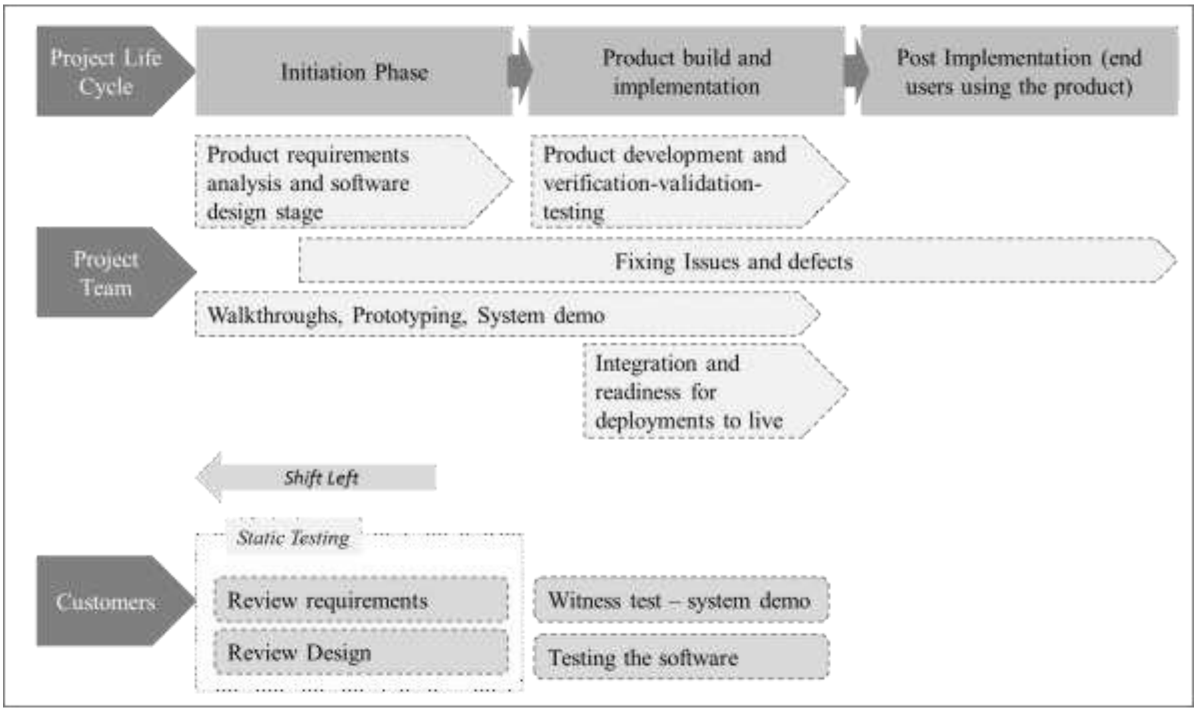


Figure 2 – Project life cycle vis – a – vis customer quality assurance activities

For each of these projects considered for study, customers were involved either from initiation phase or later during software build and implementation stage. When the customers were involved at initiation stage, they were involved in reviewing and formalising the requirements captured earlier which falls under the category of static testing. The test cases written by project team were reviewed which added to the assurance of ‘verification and validation right’. As seen in from the table 1,2 and 3, at every stage of build and implementation, where the customers were involved early in of project life cycle, the issue and defect after implementation was lower. It is thus critical that customers give feedback on improvisation of the digital product that will go to market as early as possible before the launch. Such feedback helps the project manager to get fix of the issues and defects early. Another key aspect of customer’s engagement is to carry out real verification and validation of the working software as it is built incrementally. This attributed to the highest percentage of issue and defect identification early as part of quality assurance as well, indicated in table 1 and 2. The customers were involved in actual testing of the software (executing customer experience tests), early which attributed in quality enhancement of the product enabling ‘fail fast to fix fast’. The data analysis and results mentioned earlier proved our hypothesis that shift left of customers in the project life cycle will provide multiple benefits.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

Automation – alignment of tight tool box:

Testing is considered as a critical aspect of quality assurance, aims to appraise the quality and also to ascertain issues and defects (Jin, H., & Zeng, F., 2011). As a part of project management, the project manager's challenge is to not only to build the software product which is robust but also reliable but also to take it to the market early. The reliability of the software is ensured through rigor in quality assurance processes such as verification, validation and testing. The need of hour today is to deliver software products right first time and early to the market. To enable the right first time project managers are required to pick up tools to automate testing which would help to achieve velocity of testing, reuse the test scripts, reuse the automation framework. As highlighted by M. Fewster in the book, 'software test automation – effective use of test execution tool, the test automation can enable some testing tasks to be performed far more efficiently than could ever be done by testing manually (Fewster, M., & Graham, D., 1999). There are several test automation tools available in the market either open source or licensed which one has to pay and buy. The open source automation testing tool are non-proprietary which is available under GNU GPL (general public license), distributed freely (Singh, I., & Tarika, B., 2014).For the purpose of our study and to test the hypothesis of automation to achieve multiple benefits such as achieve savings, efforts reduction and reusability; the author collected and analysed the year on year data for all the projects executed under the digital transformation program. Table 4 provides a view of the year on year data representing the volume of testing carried out by the project team (software testers) and percentage of automation and non-automation tests.

			Test Case Volume					**Efforts**		**Cost**		If Automation
Year	Count of Projects	Total Requirements (user stories)	Total	Automated	% Automated	Non Automated	% Not Automated	Automated (In person days)	Non Automated (in Person days)	Automated ($) in thousands	Non Automated ($) in thousands	was not done , corresponding cost ($) in thousands
2015-16	2	63	504	---	---	504	---	---	63	---	$ 17.01	
2015-16	3	379	3032	---	---	3032	---	---	379	---	$ 102.33	
2016-17	1	42	336	34	10%	302	90%	2	38	$ 0.45	$ 10.21	$ 1.13
2016-17	4	54	432	52	12%	380	88%	3	48	$ 0.70	$ 12.83	$ 1.75
2017-18	1	137	1096	164	15%	932	85%	8	116	$ 2.22	$ 31.44	$ 5.55
2017-18	4	29	232	35	15%	197	85%	2	25	$ 0.47	$ 6.66	$ 1.17
2017-18	2	121	968	213	22%	755	78%	11	94	$ 2.87	$ 25.48	$ 7.19
2018-19	4	111	888	222	25%	666	75%	11	83	$ 3.00	$ 22.48	$ 7.49
2019-20	1	567	4536	2087	46%	2449	54%	104	306	$ 28.17	$ 82.67	$ 70.42
2019-20	1	93	744	394	53%	350	47%	20	44	$ 5.32	$ 11.80	$ 13.31
2020-21	2	30	240	134	56%	106	44%	7	13	$ 1.81	$ 3.56	$ 4.54
2020-21	8	132	1056	602	57%	454	43%	30	57	$ 8.13	$ 15.33	$ 20.31

Source: Author

Table 4 – Year on year automated test cases and non-automated tests.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

The Project team had no test automation during the year 2015-16. The testing spend this year due to non-automation was $119 thousand as depicted in the table 4. Project team started automating the test cases for first project with 10% progress which also included framework development and proof of concept. To avoid cost overrun, the project team selected testing tool called selenium which is widely used across the IT companies today. This is quite evident from the studies done by researcher in their studies which highlighted that selenium is the popular testing tool used by many project teams (Lotto, Linda S., et al.,1986) and (Raulamo-Jurvanen, et al.,2017).

Selenium integrated development environment (IDE) is open source software which is freely downloadable for use. Open-source software (OSS) is computer software that is released under a license in which the copyright holder grants users the rights to use, study, change, and distribute the software and its source code to anyone and for any purpose (St. Laurent, Andrew M., 2008).

The test automation was implemented incrementally project after project and year on year. Table 4 reveals that by 2020-21, 57% of the test cases were executed using test automation, while 43% was non-automated. As the test automation incrementally progresses year over year , it is observed that the percentage of non-automated test case volume has reduced. This has reduced the testing spend. It is evident from the data that a cumulative of $53.15 thousand was spent on building and testing through test automation, while $341.79 thousand was spent on non-automated testing. in total by year 2020-21 around 3937 test cases were automated which if was not automated would have costed $132.87 thousand. Thus, test automation gave a straight savings of $79.72 thousand by end of year 2020-21.

The test automation is not just carried out for one time activity. If not all the test cases automated, some portion of test case volume is reused when they are re-executed as part of regression testing during issues or defect fixes are re-tested. Table 5 depicts the reusability of test automation and benefits inform of savings.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

			Test Case Volume				Cost	
Year	Count of Projects	Total Requirements (user stories)	Total	Automated	% Automated Test cases reused	Count of automated test cases reused	Maintenance cost for automation Reusability ($) in thousands	If Automation was not done , corresponding cost ($) in thousands
2016-17	1	42	336	34	35%	12	$ 1.18	$ 3.18
2016-17	4	54	432	52	40%	21	$ 2.07	$ 5.60
2017-18	1	137	1096	164	26%	43	$ 4.27	$ 11.54
2017-18	4	29	232	35	50%	17	$ 1.74	$ 4.70
2017-18	2	121	968	213	50%	106	$ 10.65	$ 28.75
2018-19	4	111	888	222	40%	89	$ 8.88	$ 23.98
2019-20	1	567	4536	2087	35%	730	$ 73.03	$ 197.18
2019-20	1	93	744	394	32%	126	$ 12.62	$ 34.07
2020-21	2	30	240	134	30%	40	$ 4.03	$ 10.89
2020-21	8	132	1056	602	30%	181	$ 18.06	$ 48.76

Source : Author

Table 5 – Automated test case reusability

The built up automation test cases are reused year on year in portion as depicted in the table. Reuse describes the level of the reuse in the development and the testing (K. Karhu, 2009). The reusability is dependent upon the testing requirements at an instance of time. Table 5 depicts such portion in percentage of automated test cases reused and corresponding maintenance cost associated to execute the set of reusable automated test cases. The total cost for maintenance the set of automated test cases between year 2016 and 2020-21 is $135.53 thousand as compared to $368.63 thousand, which is the cost if there was no automated test cases for reuse. A total savings of $232.16 thousand is saved through reusing the automated test cases. Hence from data analysis of table 4 and table 5, it is evident that test automation gives savings benefits for new test automation initiatives and even during reusability. The efforts required for testing also gets reduced drastically due to introduction of test automation by the project management team. These data sets prove the hypothesis of benefits through automation.

Multidimensional Validation and Testing:

The third factor which is also critical for project management team to consider during building and implementing the digital software product is introducing multidimensional verification and validation. This means various levels of testing is required by project team to execute for achieving higher quality assurance. These levels of testing comprises of functional testing, non-functional testing, accessibility and cross browser testing. Functional

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

testing is conducted to evaluate the compliance of a system or component with specified functional requirements (IEEE, 2010, p.140). Non-functional testing will verify how in a certain condition a system behaves like performance with a load, scalability, resilience and how system is vulnerable to threats from hackers. Non-functional testing such as performance testing, reliability testing, security testing is important aspect of the project team (Crispin & Gregory 2009) .For all digital products which are developed for end users, accessibility is one of the other aspects which needs to be taken care off. The objective of accessibility testing is to assure that the software product being built is usable across wide population, irrespective of the disabilities. According to (Freire, A. P., et al., 2008) , had surveyed respondents who were involved in web based development projects highlighted that only 20% understand the importance of accessibility testing. Cross browser compatibility and multi devise compatibility is vital too when it comes to digital products. The end users will use the digital software from iphone to android or using any browser on the computer. The project team has to ensure that software product developed is compatible across browsers and multiple devices. As highlighted by (Bartlett, J. ,2016), verification of cross browser compatibility is to verify by comparing a website or web application's functionality and design. This will help project teams to ensure software product behaves consistently across platforms and web browsers. For the purpose of our study and to test the hypothesis of multidimensional verification and validation to achieve multilevel earned confidence through test effectiveness; the author collected and analysed the year on year data for all the projects executed under the digital transformation program. Test effectiveness is one of the key metric to assess the effectiveness of testing carried out project team with respect to issues and defects found after launch of the software product. As highlighted in the blog by (Tricentis, 2016), test effectiveness is the difference between the number of issues or defects found by project team and found after software product is launched in the market.

The table 6 indicates the issues found by the project team for the program under study and issues and defects reported by users post deployment to live. It also highlights the percentage mix of testing conducted by the project team which also contributes to test effectiveness.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

				Milti Level Testing Mix					
Year	Count of Projects	Project Complexity	Total Issues found before launch by Project team	Functional Testing	Non-Functional testing	Accessibility Testing	Cross-browser testing	Total Issues found after Product Launch	Test Effectiveness
2015-16	2	Low	101	56%	33%	--	11%	96	51%
2015-16	3	High	606	48%	38%	--	14%	234	72%
2016-17	1	Low	67	68%	22%	--	10%	21	76%
2016-17	1	Medium	86	60%	35%	--	5%	46	65%
2016-17	4	Medium	219	50%	35%	--	15%	86	72%
2017-18	1	Low	46	50%	32%	--	18%	12	79%
2017-18	4	Low	194	48%	32%	--	14%	85	69%
2017-18	2	Medium	178	50%	32%	8%	10%	76	70%
2018-19	4	High	907	52%	30%	8%	10%	297	75%
2019-20	1	Medium	149	50%	32%	8%	10%	56	73%
2019-20	1	Low	48	45%	25%	10%	20%	23	68%
2020-21	2	Medium	211	52%	30%	10%	8%	79	73%
2020-21	8	High	467	53%	30%	10%	7%	143	77%

Source: Author

Table 6 – Multi Level testing mix and test effectiveness

The analysis of data as given in table 6 , depicts that project team's test effectiveness is >70% , when the testing mix has functional testing coverage of 48% or more, non-functional testing coverage is 30% or more and cross browser testing has coverage of 7% or more. However when accessibility testing was not introduced i.e until year 2017-18 irrespective of right mix the test effectiveness has dropped below 70%. This is possible the case when users have reported issues on usability and accessibility aspects of the software. Test effectiveness compares the issues or defects reported by users after product launch against number of issues or defects reported by project team before launch. The multi level testing mix though plays an important role in proving the hypothesis, it is also imperative that practically project team has mix of team who has right skills and appetite to discover more issues and defects during software development (build and implementation phase).

Conclusion

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

In this study, researcher have investigated the factors that would enable quality assurance of all software products. The project management team finds it challenging to enhance quality assurance and struggle in the real world to identify issues and defects early in the project life cycle. Thus quality assurance methods become a vital aspect to build a reliable software product in the digital transformation era. To achieve this project managers will need to introduce a framework which will help to identify issues and defects early and also help to achieve savings. The findings of this study revealed that there are three factors which would be critical for ever project manager to consider as part of implementing effective quality assurance framework. The first factor focused on move the customer engagement early in the project life cycle. It was evident from the data analysis that the shift left of customers starting from initiation phase of the project until build and implementation provided benefits. These benefits helped the project to understand the issues and defects early. Early issues and defects detection helped to fix them early in the project life cycle. Where the customers did not participate early , the number of issues and defects after product launch was higher, evident through the weighted defect density. The second factor is using automation which is choosing the right automation tools and automating the testing. Through the study and data analysis it was evident that automation brought substantial savings with year on year built up and reuse. The data and analysis made it clear that though there is a maintenance cost to manage the automation test scripts but cost benefits year on year is high. There was a cost optimisation as seen due to usage of open source tool by the project team. The final factor, revealed project management also need to focus on using a multilevel testing mix to achieve higher test effectiveness. It was evident from the data and analysis that when a testing right mix was used the test effectiveness went beyond 70%. Thus, the three factor quality assurance framework helped the program in achieving various benefits as highlighted in the earlier section.

This study is conducted considering projects which were delivered as part of single program for digital transformation. The ‘shift left customers’ factor considered the customer from one single organisation entity as all the products were developed under the contract of the same customer organisation. The user issues and defects were collected as reported from the geographical areas these digital products were launched. This framework is not trial across other similar programs covering global market. The study also did not consider the skill level of project team who are developing and testing the products. The skill level as an attribute may be one of the factors inducing higher issues and defects after product launch.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

It is recommended that further study is conducted considering the three factors of quality assurance framework, already identified and defined through this study. This framework is required to be applied for multiple projects and program to get an indication of benefits manifested shift left for fail fast, savings through automation and higher test effectiveness.

References

64 Essential Testing Metrics for Measuring Quality Assurance Success. (2016, January 27). Tricentis.https://www.tricentis.com/blog/64-essential-testing-metrics-for-measuring-quality-assurance-success

Adam Auerbach (2015), Part of the Pipeline: Why Continuous Testing Is Essential, https://www.techwell.com/techwell-insights/2015/08/

Anon (2019), A Complete Non-Functional Testing Guide for Beginners. Softwaretestinghelp.com

Anon. "List of Web Browsers." Wikipedia, 18 Nov. 2021,en.wikipedia.org/wiki/List_of_web_browsers. Accessed 27 Nov. 2021.

Bartlett, J. (2016, July 21). Why Cross Browser Testing is Important. TestLodge Blog. https://blog.testlodge.com/why-cross-browser-testing-is-important

Crispin, Lisa, and Janet Gregory. Agile Testing : A Practical Guide for Testers and Agile Teams. Upper Saddle River, Nj, Addison-Wesley, 2009.

Fewster, M., & Graham, D. (1999). Software test automation (pp. 211-219). Reading: Addison-Wesley.

Freire, A. P., et al. "The Perception of Accessibility in Web Development by Academy, Industry and Government: A Survey of the Brazilian Scenario." New Review of Hypermedia and Multimedia, vol. 14, no. 2, Dec. 2008, pp. 149–175, 10.1080/13614560802624241. Accessed 28 Nov. 2021.

Glenford J. Myers,"Review of 'the Art of Software Testing, , Wiley-Interscience, 1979." ACM SIGMIS Database: The DATABASE for Advances in Information Systems, vol. 11, no. 4, July 1980, pp. 23–23, 10.1145/1113469.1113478. Accessed 22 June 2021.

IEEE. (2010). ISO/IEC/IEEE International Standard - Systems and software engineering. P.104

J. Rode1 and M. A. P.-Q. Mary Beth Rosson. The Challenges of Web Engineering and Requirements for Better Tool Support. Technical Report TR-05-01, Computer Science Department, Virginia Tech., 2002.

Jin, H., & Zeng, F. Research on the definition and model of software testing quality. The Proceedings of 2011 9th International Conference on Reliability, Maintainability and Safety, 639-644.

ISSN PRINT 2319 1775 Online 2320 7876

 UGC CARE Listed (Group -I) Journal Volume 11, Iss 11, 2022

K. Karhu, T. Repo, O. Taipale and K. Smolander, "Empirical Observations on Software Testing Automation," 2009 International Conference on Software Testing Verification and Validation, 2009, pp. 201-209, doi: 10.1109/ICST.2009.16.

Lotto, Linda S., et al. "Qualitative Data Analysis: A Sourcebook of New Methods." Educational Evaluation and Policy Analysis, vol. 8, no. 3, 1986, p. 329, 10.2307/1163741. Accessed 30 June 2019.

M. Iacob and R. Constantinescu, "Testing: First Step Towards Software Quality", Journal of Applied Quantitative Methods, Vol.3. Issue 3, 2008.

Mary-Luz Sánchez-Gordóna,Lourdes Morenoa, (2014). Toward an integration of Web accessibility into testing processes,5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, DSAI 2013.

N. Ibrahim, W. M. N. W. Kadir and S. Deris, "Propagating Requirement Change into Software High Level Designs towards Resilient Software Evolution", Asia-Pacific Software Engineering Conference (IEEE,APSEC '09), Penang, pp. 347 – 354, 2009

Olausson M., Rossberg J., Ehn J., Sköld M. (2013) Engaging the Customer.P101-113.

Prasad, Dr. K.V.K.K. (2008). ISTQB Certification Study Guide, Wiley, p. vi

Raulamo-Jurvanen, P., Mäntylä, M., & Garousi, V. (2017, June). Choosing the right test automation tool: a grey literature review of practitioner sources. In Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (pp. 21-30).

Richard Knaster - et al., https://www.scaledagileframework.com/

Seth, Frank Philip, et al. "Software Quality Construction in 11 Companies: An Empirical Study Using the Grounded Theory." Software Quality Journal, vol. 23, no. 4, 18 Aug. 2014, pp. 627–660, 10.1007/s11219-014-9246-2. Accessed 24 Apr. 2020.

Singh, I., & Tarika, B. (2014). Comparative analysis of open source automated software testing tools: Selenium, sikuli and watir. International Journal of Information & Computation Technology, 4(15), 1507-1518.

St. Laurent, Andrew M. (2008). Understanding Open Source and Free Software Licensing. O'Reilly Media. p. 4.

Sunil Sehgal (2016). The Essential Role of QA in Digital Transformation, https://www.techwell.com/techwell-insights/2016/