ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

Effort Estimation for Performance Testing of Software Applications

Mohammed Abdul Razack Maniyar[1], Mohd Abdul Hakeem[2], Dr. Mohd Khalid Mubashir Uz Zafar[3]

Ph.D Research Scholar, Department of Physics, Dravidian University, Kuppam, India[1]

Team Lead, Accenture, Hyderabad, India[2]

Associate Professor, Maulana Azad National Urdu University, Hyderabad, India[3]

ABSTRACT: Performance testing is one of the important Non-Functional testing in the Software development life cycle. For a successful delivery of the software product a sound and accurate planning is needed for the software testing activities. Good effort estimation is also needed for the Performance testing activity. Software testing effort estimation helps in forecasting quality delivery of products. Estimations based on guesses and conjectures are going to drop the performance testing projects into jeopardy.To improve realism in software Performance testing projects, it is better to have knowledge about the testing effort estimation. A lot of studies were carried on to explore the techniques and methods for estimating the efforts for the software functional testing. Very little have been discussed on the effort estimation of Non-Functional testing, especially Performance testing. This paper mainly focuses on the effort estimation of Performance testing activities which is needed for a successful delivery of quality product. Various challenges, methods and approaches for estimating the efforts are discussed here in this paper.

KEYWORDS: Software Testing Effort Estimation, Performance Testing Estimation, Performance testing life cycle, Performance testing estimation techniques.

I. INTRODUCTION

The successful delivery of a Quality software application or product requires a good planning and at the same time an accurate estimation is always needed. Software Performance testing, being an important type of testing under non-functional testing area has to be given more importance as performance bottlenecks identification helps a lot to the organization by saving a lot of cost and in achieving a quality product. The Performance testing life cycle includes a lot of activities, end-to-end. To achieve the performance objectives and goals and to maintain the SLA's (Service Level Agreements), the performance testing has to be conducted more effectively, which requires a good planning. Also, a good estimation of the performance testing efforts has to be done as it is a sound foundation for the performance testing projects. A good and more accurate estimate always helps in forecasting quality delivery. Planning and estimation done on guesses, assumptions, suppositions and presumptions which are based on conjectures always create a lot of chaotic and pandemonic situation and drops the project into jeopardy and leads to project failure.

A lot of studies were conducted so far to explore the estimation techniques and methods for the Project or functional testing efforts. The study presents a generic estimation method which is independent of technology and platform. This paper is an effort to explore further in the area of Non-Functional testing, exclusively for Performance testing activities. Different estimation techniques and methods for performance testing are highlighted in this study. Various challenges and pain points are also mentioned in this paper to give a broader knowledge on performance testing efforts estimation.

Lord Kelvin had said: "If you cannot measure it, you cannot improve it". It is important to measure time and other factors for testing effort from the beginning in the performance testing end-to-end life cycle. Different activities

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

are identified through work breakdown structure technique. Many conventional techniques like Delphi technique, techniques based on software size and test case enumeration were also used for estimating software testing activities. The success delivery of quality software application or product depends on many variables and factors. For a successful project delivery, the degree of uncertainty has to be studied upon and has to be planned for high level of accuracy. A good estimate often helps in steering the project towards success. The Software Performance test specialist or architecture or the Manager has to explore his or her skills to study the project requirements, the environment and the human resource provided to identify the appropriate method, technique or model for estimating the performance testing efforts. Tools can also be used for estimating the efforts. The prepared estimation has to be reviewed again and again for any adjustments or updates.

This paper highlights the challenges which the Performance test Specialist has to face while estimating the efforts. The three point (PERT) estimation technique has been used in a case study presented in this paper. The formula generated excel effort estimator is used in the case study for a Medium level size project with complex technology involved. The estimated figures were seen to be accurate for the project studied.

II. RELATED WORK

Lava Prasad Kafle [1] has conducted a case study of 5 companies and reviewed over 50 papers spanning 30 years' time period. Based on these studies it was suggested that testing data estimations need to be recorded, revised and historical evidence has to be taken into account for avoiding the underruns or over runs in testing. The study found that practically the companies were using the expert judgement and empirical evidence based models for estimatingthe verification and validation testing cost efforts.The observations which were summarized as main findings of the case study are- (a) The test effort are calculated similarly as Total Project effort estimation using expert judgement method and (b) Estimation error of testing effort seems to correlate closely to the estimation error of Total project.Study concluded saying that the companies can further improve the cost effort estimation errors of the verification and validation testing and projects with detailed analysis of their processes and by taking the help of consultants.

Kamala Ramasubramani and Alain Abran [2] have presented an overview of software test estimation techniques from their survey and also some of the challenges that needs to overcome to improve the foundations of the software testing estimation techniques. To analyse test estimation techniques, the criteria points included are, Customer view of requirements, Functional size, Mathematical validity, Verifiability and Benchmarking. The estimation techniques were classified into different groups which are based on the judgement and rules of thumb, Analogy and work breakdown, Factors and Weights, the software size and fuzzy inference and other models. The main focus of the survey presented here is the estimation of effort for software testing and it did not discuss anything about the estimation of cost or schedule.The estimation techniques presented here in this paper as a part of survey is limited to the scope of application in outsourced software testing projects, and it does not attempt to tackle estimates for all types of testing.

ChintalaAbhishek et al. [3] have proposed software testing effort estimation models. This study covered the brief details of the various traditional methods for effort estimation both in the pre-coding phase and the post-coding phase. To improve the accuracy, the proposed model applies the Neural network technique for the pre-coding phase based on usecase point and soft computing technique.The Neural network structure for Pre-coding effort estimation includes, designing the network, Training the network and testing the network. During the Post coding effort estimation phase, the Project Manager uses the coding document to make an estimation of the test effort. The proposed method is based on the fact that the testing effort is based on the number of inputs, outputs and the complexity and criticality of the code. The conventional methods for pre and post coding effort estimations were also discussed in this study. The architecture of the proposed model consists of three main components, Actor components, Usecase component and Technical and Environmental Factors (TEF) components.

Kamala Ramasubramani and Alain Abran [4] have presented the estimation models for the Software Functional testing effort. The study used International Software Benchmarking and Standards Group(ISBSG) database for building software functional testing effort models. In this study, three Project groups were found to be statistically significant which were characterized by domain, team size, elapsed time and rigor of verification and

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

validation.Apart from the functional size, the variations in the test effort within each project group was explained by (a) Processes executed during development and (b) Processes adopted for the functional testing.The study identified two independent variables- Development Process Quality Rating (DevQ) and Test process Quality Rating (TestQ) which were quiet influential in the estimation of Software functional testing effort.Total of 24 models were built using combinations of variables like, Size, DevQ and TestQ. It was found that the estimation models using projects measured in COSMIC (Common Software Measurement International Consortium) function point exhibited a better Quality with more accurate estimates as compared to the projects measured in IFPUG (International Function Point Users Group) function points.

Praveen Ranjan S et.al [5] have proposed a model based on cuckoo search for estimation of Software testing efforts. Cuckoo search is a metaheuristic technique used for testing estimation with a greater accuracy. This technique uses estimation based on use cases and actors of the system by rating them into different categories. In this technique constant weights are assigned to each factor. One of the drawbacks of this cuckoo search technique is that, it can prevent the system from adapting with time when the perceived weight changes which changes the value of effort.In this study, a cuckoo search algorithm is applied for the estimation of software testing efforts. This cuckoo search algorithm approach optimises the parameters involved in effort estimation using use case point analysis. This approach was found to give better results as compared with the other metaheuristic techniques for estimation of testing efforts.

Praveen Ranjan S et.al [6] havedone an assessment through a Fuzzy criteria approach for estimating the software testing effort. This fuzzy based model uses a fuzzy logic, which is a powerful tool for solving the real world problems with uncertain and imprecise information in dealing with the semantic knowledge. This model is capable of incorporating uncertainty which is created due to the difficulty in exact quantification of certain efforts. This approach has some difficulties in using this. One of the difficulties is in determining and fine-tuning of fuzzy rules which depends on the experience and exposure of the decision maker. For analysis and deriving fuzzy rules, fixed triangular membership functions have been considered in this model. The fuzzy rule derived expresses the information for the interpretation of the nature of the software testing efforts. This proposed process can thus be helpful in evaluating the software testing efforts more accurately which in turn helps the developers to a greater extent. This model can further be extended with a number of extensions and applications by using techniques like artificial neural networks and evolutionary computation and combination of neurons-fuzzy approach for effective and accurate estimation of Software testing efforts.

Aloka S et.al [7] have presented a Particle Swarm Optimization technique using which software testing effort estimation can be optimized to a greater extent. This optimization technique was applied on the Use case points and the Test point analysis techniques and the results were compared with the existing methods. It was found that the results were very much closer to the actual testing efforts. This study can be further extended to optimizer the other estimation techniques. The final results of this study have shown that the magnitude of relative error has decreased to a greater extent and the predicted results are very much closer to the actual results. The Particle Swarm Optimization technique can also be applied for estimating the software development activity. There is a possibility of further optimization by applying other variants of this technique and can be explored further.

Aranha E and Borba P [8] have proposed and evaluated a measure called Execution Points. The manual test execution efforts for functional tests especially for embedded systems and other domains are a problematic one which demands a high manual testing execution effort. The study proposes a new approach for estimating the manual testing efforts for these type of systems which are more complex. Execution points are presented as a measure for the size and the complexity of the functional tests which is calculated based on the test specifications. Study also presented the approach for estimating the manual testing efforts and capacity based on the execution pointsmeasure. Estimation models were also presented for manual test execution efforts and capacity based on the Execution points. The results from the empirical study suggest the viability of estimating manual testing execution efforts based on the execution points thereby supporting the research hypothesis. The study further says that this approach can be used to estimate the efforts for other testing activities test set up effort, designing the test specifications, regression testing and others.

PriyaChaudhary and C S Yadav [9] have presented an approach for estimating the software testing effort and size using test case point for the software testing projects. All major factors that affect the testing projects are outlined and covered in this proposed model which helps for an accurate and fair estimation. The proposal is a technology

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

independent one and supports the need for estimation of software testing efforts and helps in forecasting quality deliveries. This proposal is more reliable as it is derived from the SRS of the software to be developed which makes the estimation more precise and accurate. The test case point estimates helps the organization in a better project or product management as this approach provides and early estimation. This early estimation helps to gain knowledge about the project length and schedule which helps in early planning to reduce cost and save time. Thus, this proposal on the whole helps in facilitating early detections and process improvements.

Fernando Oliveira de A and Luciana V M [10] have provided a literature review of some main academic publications under circulation which focuses on the study of how to estimate the software testing activity. The review provided both characteristics of software testing estimation and survey of some main existing techniques. A bibliographic survey was done from the indexed periodicals in the databases Scopus and Web of science, accessed during the period from 2013 to 2014 through CAPES periodicals portal. Eight different themes were identified for discussion about software testing estimation.Estimates produced from (a) Software development Project effort distribution (b) Percentage of time for development (c) Function Points (d) Use Case Points (e) Test Case Points (f) Test Point Analysis (g) Historical bases of Test Projects (h) Simple Estimate Test (SET). In order to make a comparative assessment of the degree of accuracy, the study further recommended for selecting some existing techniques from the literature and their application in actual corporate projects.

PriyaChaudhary and C S Yadav [11] have presented a comparative analysis of two software testing effort estimations and thereby finding the optimum way for testing effort estimation using Absolute Error and Relative Error calculation techniques. For the comparison, the analysis was done using two models- Parametric effort estimates and Test Case point effort estimate analysis. To find the better testing estimation model and for optimization of the test effort estimations, the values of Absolute Error and Relative Error were utilized. The proposal supports the need for the optimization of the testing effort estimation and it is also technology independent one. Of the two models, the study finally recommends the Test Case Point Estimation model as this seems to be more robust than the Parametric effort estimation model. This model can therefore be used for large projects as this would be very much helpful in saving cost and resources as compared to the parametric effort estimation model which has more number of errors.

III. PERFORMANCE TESTING EFFORT ESTIMATION CHALLENGES

The Performance Test Specialist often faces few challenges while estimating the efforts for the software Performance Testing projects. Few important ones are mentioned below.

(1) *Challenges with Scope:*
Projects with scope not defined properly will drop the project at risk. Especially, for effort estimation, the Test specialist must be very much careful with this factor. The Performance Test Specialist has to check regularly and be vigilant to avoid Scope Creep. The requirements have to be analysed properly and the scope has to be defined and regularly checked to avoid Scope Creep and to achieve high degree of accuracy.

(2) *Challenges with Incomplete or poor requirements:*
Effort estimations based on incomplete or poor requirements cannot be helpful in achieving the accuracy and at the same time would not be helpful in forecasting the quality of delivery of the project. For a meaningful and accurate estimate, the Test Specialist has to take care of the requirements. A thorough check has to be made for the completeness and correctness of the requirements.

(3) *Challenges with Task breakdown:*
If the tasks are not broken down enough, it lacks clarity and may thus drop the degree of accuracy of estimations. For Agile methodology, each performance testing task unit should be equivalent to an Agile story point else it would slip out into a subjective guess. A good Work Breakdown Structure would help the Test Specialist in estimating the efforts properly to a greater level of accuracy.

(4) *Challenges with change in Application &Environment:*
A frequent change in environment would be a biggest challenge for the Performance test Specialist. Frequent changes in the software application too are going to impact the initial planned efforts. Changes are to be

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

evaluated and analysed for its impact and the effort estimated values have to be revisited. The degree of uncertainty has to be removed by incorporating the changes to the initial planned estimates.

(5) *Challenges with the Risks:*
For a good estimation, the Project Risks have to be considered. Often risks crop up during the Performance test life cycle. The project has to be evaluated for the known risks and mitigation plan has to be prepared and accordingly the estimations have to be prepared by considering these risks. Ignoring the risks often proves to be an obstacle for Project success. So, it is always suggested to do a Risk Planning while estimating the work. This would help in maintaining accuracy.

IV. CASE STUDY

Study was conducted on a Mid-size complex software application Performance testing project. The application is developed in .NET and Java Platforms. The architecture consists of Web, Application and Database servers. End-to-End Performance testing activities were included for estimating the efforts. All the areas of Performance testing management are covered while estimating for the project. Basic activities included for estimation include- Requirements gathering and analysis, Performance test planning, Load test tools assessment, Load test environment set-up, Load Test designing, Load Test scripting, Test Data creation, Test executions, Resource Monitoring, Load Test script customizations or maintenance, Load test results analysis, Test report generation, Performance bottlenecks analysis and fine-tuning.

The Non-functional requirements have to captured before proceeding for the planning activity. The business and system critical scenarios have to be identified and analyses before confirming for the test. This gives the number of scenarios to be considered for scripting and designing of the test. The Performance test specialist has to segregate the number of Simple, Medium and Complex scenarios from the identified list. Based on the type and complexity of the script, the effort can be easily estimated. A total of 35 business critical and system critical scenarios were identified for the Load and Performance testing for which effort estimation was done. ThePERT estimation technique was applied for estimating the efforts for these scenarios based on the complexity of the scripts. The Optimistic, Pessimistic and Most likely estimate valuesare taken into consideration. The estimates are found out by applying these values to the formula given below.

$$Estimate = [Optimistic\ estimate + 4(Most\ likely\ estimate) + Pessimistic\ estimate)] / 6$$

The Parametric and Bottom-up estimation technique was also applied in few cases for few selected activities.

Based on the project requirement and SLA's, the type and number of performance tests has alsoto be planned for the project. Following types of tests were planned for this study- Safe Load, Load Test for the targeted load of users, Stress test and Stability (or) Endurance test. Efforts were calculated for these planned tests accordingly. Three different load test scenarios were added to the estimated list.

For a successful delivery of Performance testing project, it is very much needed to do a tool evaluation before proceeding for the test planning and execution. The right tool has to be evaluated and identified as per the requirement and technology stack of the software application. Some efforts have to be put in for this activity as well. 8-10 hours of effort were allocated for this activity. This activity is important as the right tool always helps in generating good and accurate results. Test Specialist has to do a thorough analysis and evaluation to see that the identified tools are well suited in the planned framework. The 10 hours of effort invested here in this study was helpful in identifying the different tools used for Scripting, load generation, diagnostics tools, sniffing tools, database profiling tools, resource (server) monitoring tools, log and results analysis tools and report generation tools. Fig.1. gives the list of identified tools duringtool evaluation activity. The servers available in the architecture were included in the estimation for server monitoring, which included Web, Application and database server. Efforts were evaluated separately for these servers for their health check during the test execution.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

TOOLS IDENTIFIED DURING TOOL EVALUATION	
	HP LoadRunner
	HP Sitescope
	Perfmon
	SQL Profiler
	HTTP Watch
	PAL Tool
	HP Web Page Diagnostics Tool
	WebPageTest
	Grafana
	OpenNMS

Fig.1. List of Software Performance Testing tools identified during tool evaluation.

An Excel sheet based estimator is used in the study for estimating the efforts. Fig.2. represents the tabulated figures in hours for each type of effort identified for the selected software performance testing project. Contingency hours are also added to the estimated hour's list as a buffer. The efforts were identified through the Work Breakdown Structure.

PERFORMANCE TESTING EFFORT ESTIMATION

Softare Application Complexity	M

Targetted No. Of Users	15000

TASK	ESTIMATED TIME (Hrs)
Requirements Analysis	16
Test Planning	32
Load Test Tools assessment	8
Test Designing	28
Test Environment Set up	16
Load Test Script Development	390
Load Test Data Set up	40
Load Test Execution	52
Server Monitoring & Maintenance	35
Load Test Scripts Maintenance (Modifications/Customizations)	35
Load Test Analysis	28
Bottleneck Analysis	20
Load Test Reporting	28
Contingency	16
Total Number of Hours	744

Type of Script	Simple	Medium	Complex
Number of Scripts	8	15	12

Type of Server	App Server	Web Server	DB Server
Number of Servers	1	1	2

Type of Test	Safe Load	Load	Stress	Stability
Number of Test Executions	2	3	1	1

Fig.2. Estimated Effort hours for Performance Testing Project.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 12, December 2018

V. CONCLUSION

For a successful delivery of a software Performance testing project, estimations play a vital role. Performance testing projects often fail if they are based on guesses, conjectures and assumptions. Henceforth, the effort estimation has to be taken seriously for the Non-functional testing projects as well, especially for Performance testing.The more accurate the estimates are, the more it would be better for forecasting a quality delivery. Based on the size and complexity of the project, the estimation techniques have to be applied. For a good estimation all the activities obtained from the work break down structure has to be taken into consideration. The presented approach is technology independent and can be applied to any technology stack and for large size projects as well. Effort estimations can also be done for Mobile Applications performance testing using this approach. The case study for Mid-size project has shown high degree of accuracy. The study was conducted on Software Web Applications with complex architecture. Similarly, a study can also be conducted on Mobile applications (Web, Native and Hybrid) and the estimated values can be compared with the actual values. Other estimation techniques like Parametric and Bottom-up approach can also be used for estimating the project efforts based on the requirement and feasibility. For good estimates, it is always suggested to consider the challenges and risks related to the estimations and Performance testing project management. The degree of uncertainty can be removed to a greater extent by considering these factors.

REFERENCES

[1] Lava Prasad Kafle, "An Empirical Study on Software Test Effort Estimation", International Journal of Soft Computing and Artificial Intelligence, Vol. 02, Issue 02, pp. 96-106, 2014.
[2] Kamala Ramasubramani J and Alain Abran, "A Survey of Software Test Estimation Techniques", Journal of Software Engineering and Applications, Vol. 06, pp. 47-52, 2013.
[3] ChintalaAbhishek, Pavan Kumar V, Harish V and Praveen Ranjan S, "Test Effort Estimation using Neural Network", Journal of Software Engineering and Applications, Vol. 04, pp.331-340, 2010.
[4] Kamala Ramasubramani J and Alain Abran, "Estimation Models for Software Functional Test Effort", Journal of Software Engineering and Applications, Vol. 10, pp. 338-353, 2017.
[5] Praveen Ranjan S, Abhishek V and Priyanka N, "Software Test effort Estimation: a model based on cuckoo search", International Journal of Bio-Inspired Computation, Vol. 04, Issue 5, pp. 278-284, 2012.
[6] Praveen Ranjan S, Sirish Kumar, A.P Singh and G Raghurama, "Software Testing Effort: An Assessment through Fuzzy Criteria Approach", Journal of Uncertain Systems, Vol. 05, Issue 03, pp. 183-201, 2011.
[7] Aloka S, Peenu Singh, Geetanjali R and Praveen R S, "Test Effort Estimation-Particle Swarm Optimization Based Approach", International Conference on Contemporary Computing IC3", Vol. 168, pp. 463-474, 2011.
[8] Aranha E and Borba P, "Estimating Manual Test Execution and Capacity based on Execution Points", International Journal of Computers and Applications", Vol. 31, Issue 03, pp. 167-172, 2009.
[9] PriyaChaudhary and C S Yadav, "An Approach for Calculating the Effort Needed on Testing Projects", International Journal of Advanced Research in Computer Engineering & Technology", Vol. 01, Issue 01, pp. 35-40, 2012.
[10] Fernando Oliveira de A and Luciana V M, "Software Testing Estimation: Bibliographic Survey in Brazilian and International Environments", Brazilian Journal of Operations & Production Management, Vol. 14, Issue 01, pp. 10-18, 2017.
[11] PriyaChaudhary and C S Yadav, "Optimizing Test Effort Estimation-A Comparative Analysis", International Journal of Scientific Research Engineering & Technology", Vol. 01, Issue 02, pp. 18-20, 2012