Software Testing Estimation using Test Scenarios and Weight age Factor

K. Raghavan, M. Kannan

Abstract— Software testing is an important role in software development life cycle [SDLC]. In order to deliver defect free software to the customer, time lines are a key factor for software testing to ensure quality testing. Hence estimation of software testing is an important factor to test the software within the time frame in software projects. In today's software projects, estimations can be performed in many ways like function-points, COOCO model, and Use case Test point. This research work is going to illustrate about, how testing estimation can be derived based on requirements and weightage factors like simple, medium, complex against each test scenario, which include test planning and test estimation for a given release. The proposed model will be very useful for estimating the time efforts against each requirement within short duration. This model can be modify based on previous release data's to calculate the testing effort estimation in more active approach.

Index Terms— Software testing, Estimation, Software project, Requirements, Weightage factor

I. INTRODUCTION

Software engineering is a development of software product using defined process and methodology. At end results gives an efficient and reliable software product with customer satisfaction. It provided us basic understanding of software product to design, development, quality and software project management and its complexities. In Software engineering testing plays a vital role, to deliver the defect free software to the customer. Software Testing is a process to validation of the software against requirement specifications. Testing is conducted based on Software testing life cycle [STLC]. Software testing consists of Validation and Verification [ISTQB]. Estimating effort for test is one of the major and important vital roles in SDLC. It is the process of finding an estimate or approximation value. Test estimation process is to calculate the testing time line to run the test cases with provided time lines and to ensure defect free software is delivering to the customer. This research work is all about to discuss on testing estimation factor to provide the right time to validate the software and to deliver the defect free software to the customer.

II. LITERATURE REVIEW

In this section, researcher present review of the selected literature on testing estimation techniques and their usage. The key objective is to highlight the strengths and limitation of the techniques. Jyoti G. Borade and Vikas R. Khalkar [2013] [1] dealt with various methods of software project efforts and cost estimation techniques. This paper mainly focuses on software estimation methods. Estimation is a complicated activity that requires lot of key inputs. As per the journal 40% of the efforts are required for software testing in overall software project. Rashmi Popli and Naresh Chauhan [2013] [2] dealt with agile estimation algorithm for software testing for calculating the regression test efforts. This algorithm will calculated the effort and duration of small and medium size projects in efficient way. Roy Clem [2005] [3] said about test estimation with use case points. Use case point is the easiest way to calculate the test efforts not only the test efforts even for overall project efforts also been determined. Adeshkumarand Vikas Beniwal [2012] [4]dealt with test estimation with test point analysis (TPA) with using subs and drivers. It's also delta testing process is an important in software product development and 50% of total cost is expended in testing the software being developed.

Kamala Ramasubramani Jaya kumar and Alain Abran [2013] [5] said about various software test estimation techniques with survey findings and its advantages. As per the discussion there are n number of way to calculate the testing efforts based on various project types where they can be utilized easily to derive the testing efforts.

MuraliChemuturi [2009] [6] dealt with various estimation techniques for software projects. Estimation is an important activity for which time line for any activity in software development phase a crucial.
WasifAfzal [2007] [7] dealt with estimation for cost required for testing, test case effort, human efforts as a metric estimation. There are various check lists to validate and review process for test estimation.

Ashish Sharma and Dharmender Singh Kushwaha [2012] [8] dealt with requirement based complexity for software test estimation. During requirement analysis itself, with help of draft test case and resource utilization estimation can be done for software testing. This helps more in software development life cycle. Priya Chaudhary and C.S. Yadav [2012] [9] dealt with how test case-point estimation model is effective with comparative analysis of their research. Estimation is crucial in project activity and there is no standard form for the same hence this study has come to compare the test case-point estimation and parameter estimation which include development of 30 % for testing. ShaikNafeez Umar [2013] [10] dealt with Software testing estimation with COBB-Douglas functions. This estimation technique is used total no of requirement, no of test case, complexity of release and testing governance. Seyed Morteza Hosseini [2014] [11] dealt with how to model the estimation for test execution efforts by using various techniques like use case points, ad-hoc method and percentage of development time.

III. OBJECTIVES OF THE STUDY

- To provide easy and understandable estimation technique.
- To provide accurate testing time lines with help of test

Revised Version Manuscript Received on January 21, 2016.

K. Raghavan, Research Scholar, Department of CSA, SCSVMV University, Tamilnadu, India.

Dr. M. Kannan, Department of CSA, SCSVMV University, Tamilnadu, India.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

scenario and weightage factor which is predefined based on person per days and same will be re-usable to modify the factors based on project approach.

- To more appropriate technique for testing estimation so that test team can test the application with flexible time line and can deliver the defect free software to the customer.

IV. METHODOLOGY

4.1 Research Design

This Study was conducted to provide an easy effort estimation model for testing projects for requirements and overall release testing projects. This research work is comprised of descriptive using quantitative approach. Researcher also reviews the existing estimations models which are used in the literature review.

4.2 Pilot Study

In pilot study, researcher interviewed 10 individuals, who are project managers, software testing managers, test leads and team members. We came to know that, estimating the 'requirements efforts for testing' is the challenging one. They have suggested to develop a model in such way, where they can calculate the efforts required for testing on individual requirements using the Use case/weightage factor before we arrive at the actual test cases, where this will improve the testing coverage for the requirement and we can also use this at the time of audit to show that how effectively the 'effort estimations for testing time lines' have been calculated and based on our literature review. This work came up with an estimation model with given requirement, how to calculate individual requirements testing efforts and overall testing effort estimations for a release.

4.3 Instrumentation

This work has been carried out the data based on one to one interview, telephonic interview and with the existing project release data.

Primary Data:

Data for this work are collected from previous release testing project, interviewing by project manager, testing manager, test lead and tester with one to one conversation and by telephonic interview.

Secondary Data:

Data for this work are collected from various journals, reviews, testing community and testing articles.

V. DATA PROCESSING AND ANALYSIS

5. Data Processing and Analysis

In this section, it's going to depict the data which is collected are grouped into proposed table format to frame the test planning against every individual requirement with the existing release 1and 2 data. The below tables 1 and 2 shows us how the past releases data are categorized against every individual requirements with our defined parameter.

Table 1: Test Planning Parameter Previous Release I

Individual Requirements	High level Design understanding	BA/ Design walkthrough	Chase up to follow up open queries raised in Design Walkthrough and assessing regression impact	Preparing Traceability Matrix	Preparing TC's	Reviewing UTCs coverage/ Providing feedback	Incorporating BA and Design review comments in CST TC document	Peer review of TCs	Test data set up
R-1	0.38	0.25	0.25	0.125	1.5	0.4	0.2	0.125	1.5
R-2	0.165	0.25	0.25	0.125	1.5	0.4	0.2	0.125	0
R-3	0.305	0.25	0.25	0.125	2.75	0.4	0.2	0.125	1
R-4	0.45	0.25	0.25	0.125	3.0	0.4	0.2	0.125	0.5
R-5	0.4	0.25	0.25	0.125	4.0	0.4	0.2	0.125	1

Note: R denotes Requirements

Table 2: Test Planning Parameter Previous Release II

Individual Requirements	High level Design understanding	BA/ Design walkthrough	Chase up to follow up open queries raised in Design Walkthrough & assessing regression impact	Preparing Traceability Matrix	Preparing CST TC's	Reviewing UTCs coverage/ Providing feedback	Incorporating BA and Design review comments in CST TC document	Peer review of CST TCs	Test data set up
R-1	0.5	0.25	0.25	0.125	0.3	0.4	0.2	0.125	1.5
R-2	0.2	0.25	0.25	0.125	0.5	0.4	0.2	0.125	0
R-3	0.305	0.25	0.25	0.125	0.8	0.4	0.2	0.125	1
R-4	0.45	0.25	0.25	0.125	0.3	0.4	0.2	0.125	0.5
R-5	0.45	0.25	0.25	0.125	0.3	0.4	0.2	0.125	0.5

Above tables 1 and 2 shows some of the data are unchanged. As of now with this as raw data, test team has provided the estimation without a structure. As part of this work, we are going to drive a new test planning model which against each parameter with respect to given raw data.

The above test execution data's are provided by randomly without any techniques on each requirement by test

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

team where they unable to meet their dead line since, they are not proper test estimation.

The below table 3 shows the total number of testing efforts with reference to table 1 and 2 of test plan and test execution and it's not mapping exactly on individual requirements and it's been calculated randomly without any proper estimation model .

Table 3: Random test estimation efforts previous data

Requirement	Testing Efforts
R-1	3 days
R-2	8 days
R-3	4 days
R-4	5 days

Table 4 - Proposed estimation for testing planning

Individual Requirements	High level Design understanding	BA/ Design walkthrough	Chase up to follow up open queries raised in Design Walkthrough & assessing regression impact	Preparing Traceability Matrix	Preparing CST TC's	Reviewing UTCs coverage/ Providing feedback	Incorporating BA/ Design review comments in CST TC doc	Peer review of CST TCs	Test data set up
R-1	0.38	0.25	0.25	0.125	0.5	0.4	0.2	0.125	1.5
R-2	0.165	0.25	0.25	0.125	0.5	0.4	0.2	0.125	0
R-3	0.305	0.25	0.25	0.125	0.5	0.4	0.2	0.125	1
R-4	0.45	0.25	0.25	0.125	0.5	0.4	0.2	0.125	0.5
R-5	0.2625	0.25	0.25	0.125	0.5	0.4	0.2	0.125	1

Therefore, with help above tables 1 and 2, proposed model table 4 defined about how to calculate the testing planning efforts and further it will get added with over all test estimation of the model.

Table5 Proposed Solution Test Estimation for Release

CR Description	High level Design understanding	BA/ Design walkthrough	Chase up to follow up open queries raised in Design Walkthrough & assessing regression impact	Preparing Traceability Matrix	Preparing CST TC's	Reviewing UTCs coverage/ Providing feedback	Incorporating BA/ Design review comments in CST TC doc	Peer review of CST TCs	Test data set up	1st round of Test execution	Defect retest	Final round of Test execution post all defect fixing
R-1	0.38	0.25	0.25	0.125	0.5	0.4	0.2	0.13	1.5	7.6	1.9	3.04
R-2	0.165	0.25	0.25	0.125	0.5	0.4	0.2	0.13	0	3.3	0.8	1.32
R-3	0.305	0.25	0.25	0.125	0.5	0.4	0.2	0.13	1	6.1	1.5	2.44
R-4	0.45	0.25	0.25	0.125	0.5	0.4	0.2	0.13	0.5	9	2.3	3.6
R-5	0.263	0.25	0.25	0.125	0.5	0.4	0.2	0.13	1	5.3	1.3	2.1
R-6	0.28	0.25	0.25	0.125	0.5	0.4	0.2	0.13	1	5.6	1.4	2.24
R-7	0.365	0.25	0.25	0.125	0.5	0.4	0.2	0.13	1	7.3	1.8	2.92

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Test phase wise PD's	2.2 08	1.75	1.75	0.875	3.5	2.8	1.4	0.88	6	44	11	17.66
	21.16									72.85		
	Test planning									Test execution		
Total PD's	94.1											

Test planning → 21.16 Pd's efforts are required for this release.

Test Execution →72.85 Pd'sefforts are required for testing an application.

Over all → 94.1 Pd's effortsrequired to complete the overall testing.

Note: If one person is working they need 94 days to complete the testing.

This Pd's is based on the IT organization only. This can be modified based on the project as well; with above analysis we formed how to calculate the test estimation for a requirement. In table 5 we are going to calculate over all of Pd's for entire release. Hence, this research work has arrived, with use case count and weight factor where we can calculate the test planning and test execution for testing requirements. This estimation will help the organization to provide the overall testing efforts which is required for the project.

VI. CONCLUSION

Software testing is an important stage in any project. In testing project, effective test estimation is required to deliver the quality product to the customer. Since, testing team should have enough time to validate the application based on the use case and other aspects.

In the present work, researcher made an attempt to present a new model for estimating the test execution effort. In our new model, we considered total number of use case by classifying it into simple, medium and complex and its corresponding weightage factor. The formula for the calculation is defined properly and finally we calculated the overall person per day count for individual requirements and overall project.

This model can be editable based on the projects needs by changing the weightage factor and testing estimation across the weightage factor. The main advantage of this model is very easy to calculate the test estimation and it can be reused based on the project needs.

This study is limited to calculate the effort estimation of the requirement and overall release for testing projects, where it can be modified based on the project. This study does not handle the cost factor or any tools to calculate the effort estimation. The effort estimation is calculated by using the weightage factor with use case count based on the experience on the testing projects and only for manual testing projects using excel calculation. The future study of this work can be improved how to calculate the automation efforts estimation and minimizing the efforts and to carry forward the other parameter's like complete end-to-end testing projects. Same techniques can be reused for the performance and load testing to be carried out.

VII. REFERENCES

1. Jyoti G. Borade and Vikas R. Khalkar, "Software Project Effort and Cost Estimation Techniques",International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, August 2013.
2. Rashmi Popli and Naresh Chauhan, "An Agile Software Estimation Technique Based on Regression Testing Efforts", 13th Annual International Software Testing Conference in India, 04 – 05 December 2013, Bangalore, India.
3. RoyClem, "Project Estimation with Use Case Points", http://www.codeproject.com/Articles/9913/Project-Estimation-with-Use-Case-Points,2005 .
4. Adeshkumarand Vikas Beniwal , "Test Effort Estimation With & Without Stub And Driver Using Test Point Analysis (TPA)" , International Journal of Engineering Research & Technology (IJERT) , ISSN: 2278-0181 , Vol. 1 Issue 7, September 2012 .
5. Kamala RamasubramaniJayakumar and Alain Abran , " A Survey of Software Test Estimation Techniques",Journal of Software Engineering and Applications, October 2013.
6. MuraliChemuturi ,"Software Estimation: Best Practices, Tools and Techniques", publisher – J.Ross Publishing, Inc, USA, ISBN 978-1-60427-024-2, Pages=298,2009 .
7. WasifAfzal, "Metrics in Software Test Planning and Test Design Processes", School of Engineering, Blekinge Institute of Technology, Box 520, SE – 372 25 ,Ronneby, Sweden, MSE-2007:02, January 2007.
8. Ashish Sharma and Dharmender Singh Kushwaha, "Applying requirement based complexity for the estimation of software development and testing effort", Department of Computer Engineering & App., GLA University, Mathura, India.
9. Priya Chaudhary and C.S. Yadav "Optimizing Test Effort Estimation-A Comparative Analysis", International Journal of Scientific Research Engineering & Technology (IJSRET) , Volume 1 Issue2 pp 018-020 , ISSN 2278 – 0882, May 2012.
10. Shaik Nafeez Umar, "Software testing effort estimation with cobb-douglas function - a practical application", ISSN: 2319-1163, May-2013.
11. Seyed Morteza Hosseini , "A Model for Determining the Cost of Test Case Execution (or Test Execution Effort Estimation Model)", School of Innovation, Design and Engineering, April 2014 .

Authors Profile

Raghavan Kalyanaraman pursuing M.Phil., (Computer Science) from Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya University, Completed his Master of Computer Applications from Bharthidasan University. Currently, Raghavan is a Test Lead in Tech Mahindra Limited for around 6 years and previously associated with HCL Technologies Limited for around 2.5 years. His areas of interests are Software Engineering, Software Testing, Test Estimation and Test Automation. His domain areas are Telecom (OSS/BSS, VOIP and SIP), Banking and Ecommerce.

Dr. M. Kannan, has been working as an Assistant Professor in the Department of Computer Science & Applications in Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya University, Enathur, Kanchipuram –631561. He received Ph.D(Computer Science) from SCSVMV University in 2014, and M.Phil(Computer Science) from Madurai Kamaraj University in 2005. He has published more than 15 papers in International Journals, 1 in National Journals and 1 in book. His research interest includes Software Engineering & Data Mining.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.