

Testing Estimation in Agile

Monika Dixit

*Banasthali University, P.O. Banasthali Vidyapith ,
Dist. Tonk ,Rajasthan,India*

Monikadixit.dixit@gmail.com

Karan Yadav

*Research Endeavors
B-38 Sector2 Noida,Uttar Pradesh,India*

Karanyadav87@gmail.com

Abstract-- Most of the software overruns time so they are not tested well and delivered to the customer on time. Because of this sometimes some critical bugs remain in the product and the product goes to the production as it is. When a missed bug appears in the production, then this will have a negative impact on the organization and can lead a heavy loss to the organization .So we a need to make an estimation of the cost and time for the software development. This paper deals with the estimation method for testing activities in agile environment as that we can make a measure of the effort and time to be given to the testing phase. A big challenge for the test estimation in agile environment is the changing requirements (commonly known as Change Request) which is an important feature of the agile.

Keywords-- *Test Estimation, Change Request, Agile, Test Summary Report*

I. INTRODUCTION

Today the demand of the market is to have high quality and bug free software and to have the control on the effort in terms of the money and person hours. The crucial phase in the software development is the testing phase. Testing is that phase of the software development where the developed software/product is tested for the presence of the bugs. When developed software is tested with insufficient time then this can lead to missing of the major critical bugs in the software and this in turn render our entire product useless. So in order to overcome with this situation we must measure the effort required for the testing of the product. Software test cycle is divided into three stages.

A. Start Phase

In this phase we identify which parts of the developed product are to be tested. This stage involves the preparation of the test plan and test design. Test plan is a document that involves Who will test? , When will test? , What will test? , Where will test be

executed? , How will be tested? Test design is a document which includes: a) which features will be tested first? b) Which modules need driver script or which modules need stubs? c) Which modules test cases will be prepared first? Etc.

B. Middle Phase

This phase involves the development of the test cases for the various modules of the project followed by the execution of the developed test cases. Bugs discovered during the execution of the developed test cases are there by logged using the bug tracking tool.

C. End Phase

This phase involves the analysis and reporting the health of the project tested. This involves the analysis of the problems faced during the real time implementation of the project. Test Summary report (TSR) is studied which includes all the issues discovered during the testing. This includes the number of the major issues as well the number of the minor issues of individual modules.

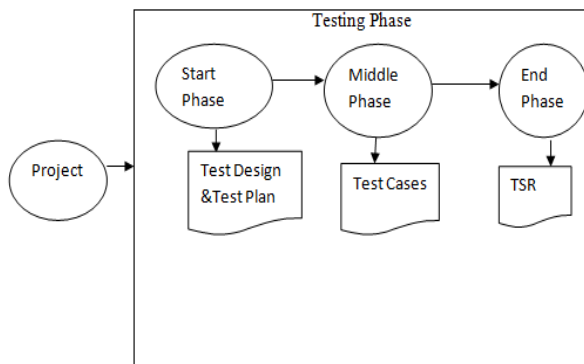


Fig.1 Testing Phases

Sometimes the test estimation is not accurately calculated because of missing some important factors.

Let's have the look on the factors that must be included while making the test estimates for any project.

I) *Size of the project*: When making test estimates considering system size is very important. Testing effort depends upon the system size i.e. Number of modules to be tested, how many processes each module has, whether processes are simple or complex etc.

II) *Testing Types Required*: We need to consider whether the product coming in testing requires smoke testing or sanity or detailed testing, functional etc. Different testing types differ in the time estimates.

III) *Testing Tasks Required*: We should consider the testing tasks that need to be carried out during testing of the product. We should include following things while making the test estimates:

- Preparation of Test plan
- Preparation of Test Design
- Preparation of Test Cases
- Execution of test cases
- Test Reporting
- Defect Tracking
- Test Summary Report

IV) *Past Experience*: Past experience helps a lot in estimating the effort for test estimation. We can infer from the analogous projects and can also have expert opinion while estimating.

Domain Knowledge: An organization has different domain projects and all need to have go through the testing phase. A

tester who has good knowledge of the health domain might not have good knowledge of the accounts domain. So he has to be given some training for the accounts domain so that the project can be tested well without missing any bug. Also each tester has his/her own capability/ability. Also the speed rate differs from tester to tester.

V) *Buffer Time*: This must be included in test estimates to deal with delays. These delays could be due to following:

- Deliverables not on time
- Hardware dependency
- Design documents not available and etc.

VI) *Bug Cycle*: Bugs found during the execution of developed test cases need to be logged and reported to the development using some defect tracking tool so we need to include this in our test estimates. Also the test cases corresponding to previous bugs logged need to be executed and verified whether these bugs are still existing or not. If existing then these bugs need to be reopened and if not existing then these should be closed.

Availability of resources: We must take available resources for the testing team into account while calculating the test estimates. Sometimes the resources were not available during the testing of the product thus delayed the product delivery. Resources also include human labor. We should consider the Holidays and requested leaves of the staff members when making the estimates. We must take care of some of the adjusting factors also while estimating.

- Normal conditions do not apply always.
- Interactions cost because sometimes the testing team member need to interact with the implementation or development.
- Things can go wrong on projects
- Uncertainty associated should be taken care off.

II. EFFORT ESTIMATION

First we have to look the system size which can be defined in terms of the number of modules. For each module we have to calculate the number of processes. To each process we assign a weight depending upon the complexity of the process. For example consider a registration page (process) which captures the user details. Then this process is a simple process.

A. Related Work

Here we are going to perform the estimation for a Hospital Information system (HIS) project which basically provides online support to various hospital operations which are spread across clinical and non-clinical services that are required to be run round the clock throughout the year. Here first we are going to find the effort estimation considering the Billing module of the HIS project. To each process of the Billing module assign whether is Simple or Normal or Complex Process. After that convert all of the process to simple process using the conversion formula:

1 Medium process= 2 Simple Process

1 Complex Process = 3 Simple Process

Billing Process	Complexity
Cash Collection Offline New	Medium
Cash Collection Online New	Medium
Day End	Simple
Ipd Bill Management Auditor	Complex
Ipd Bill Management Clerk	Complex
Online Bill Cancellation	Simple
Online Refund Approval	Simple
Online Request Cancellation	Simple
Receipt Duplicate Print	Simple
Receipt Re-Print	Simple
Refund Approval (Without Request)	Simple
Refund Request	Simple
Setups	Complexity
Advance Master	Simple
Billing Config	Simple
Billing Counter Master	Simple
Charge Master	Medium
Group Master	Simple
Hospital Service Group Master	Medium
Tariff Master	Medium

Fig.2.1 Size Estimation Sheet For Billing Module

Billing Module					
Conversion Formula:					
1 Medium=2 Simple					
1 Complex=3 Simple					
Services	12 Simple	8 Overall	Total	19	Conversion to Simple
	Medium	2	Simple	12	12
	Complex	2	Medium	5	10
Setup	8 Simple	4	Complex	2	6
	Medium	3	Total Simple Processes		28
	Complex	0			
Reports	0 Simple	0			
	Medium	0			
	Complex	0			

Fig.2.2 Size Estimation Sheet For Billing Module

Component	NEW	ASSIGNED	REOPENED	RESOLVED	VERIFIED	CLOSED	Total	Actual Count	Count	Size	Invalid
ADT	16	.	.	44	37	61	158	10	17	43	2
Alert Management	7	.	.	3	.	4	14	0	0	7	0
Appointment	63	12	75	1	1	11	6
BMED	3	.	.	25	18	22	68	5	9	21	8
Billing	.	.	.	10	8	13	31	1	2	28	1

Fig.3 Bugs with status

This is done for the Billing module Refer Fig. 2.1. Then we have estimated the test effort. Actually we can estimate the test effort from process or from test cases refer below fig. We have collected the status of the bugs for each of the modules. We have assumed that the 40% of the test cases is equivalent to the defects. We have taken 3 min for the preparation of each of the test cases.

Component	Planned Effort for System Testing (Staff Hrs) From Processes	Planned Effort for System Testing (Staff Hrs) From Cases	System Testcase Preparation Effort (Staff Hrs)	No. of System Test Cases	System Testcase Review Effort (Hrs)	No. of System Testcase Review Defects	System Testcase Rework Effort (Hrs)	System Testcase execution Effort (Staff Hrs)	No. of System Testing Defects	Retesting Effort	Actual Effort for System Testing Phase	ACTUAL	System Testing Effectiveness
ADT	57.33	197.5	19.75	395	6.58	118.5	5.93	19.75	158	137.6	237	189.61	0.92747
Alert Management	9.33	17.5	1.75	35	0.58	10.5	0.53	1.75	14	22.4	21	27.01	1
Appointment	14.67	93.75	9.375	188	3.13	56.25	2.81	9.38	75	35.2	112.5	59.89	0.9127
BMED	28.00	85	8.5	170	2.83	51	2.55	8.50	68	67.2	102	89.58	0.81744
Billing	37.33	38.75	9.875	78	1.29	23.25	1.16	3.88	31	89.6	46.5	99.80	0.93168

Fig 4 Testing effectiveness

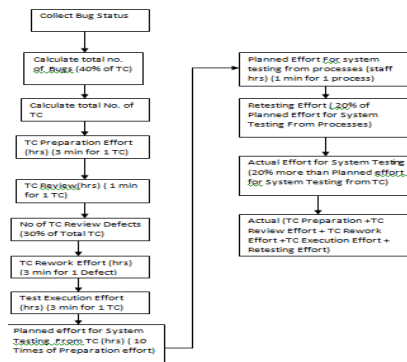


Fig.5 Flow chart for the Estimation

III. CONCLUSION

Test estimation for the project is proved to be beneficial not only for the organization as well for the client as they receive product with high quality and tested well. Often to meet the hard deadline of the client enough time is not devoted for the testing so test effort estimation can also help us to know which areas are critical or major that need to be tested. Our future work we will calculate the deviation of the actual effort for the system testing from the planned effort for the system testing . and comparative study between the Planned Effort For system testing from processes and Planned Effort For system testing from Test Cases.

References:

- [1] Muhammad Danyal Ashraf, Prof. Naeem U. Janjua "Test Execution Effort Estimation (Tee) Model In Extreme Programming," International Journal of Reviews in Computing 31st December 2011. Vol. 8
- [2] Chintala Abhishek , Veginati Pavan Kumar, Harish Vitta, Praveen Ranjan Srivastava " Test Effort Estimation Using Neural Network ," J. Software Engineering & Applications, 2010,
- [3] Jiangping Wan1, Ruoting Wang " Empirical Research on Critical Success Factors of Agile Software Process Improvement* ," J. Software Engineering & Applications, 2010,
- [4] David Talby , Arie Keren ,Orit Hazzan and Yael Dubinsky "Agile Software Testing in a Large-Scale Project" IEEE Computer Society, 2006
- [5] Andreas Schmietendorf, Martin Kunz, Reiner Dumke "Effort estimation for Agile Software Development Projects"
- [6] Edward R Carroll "Estimating Software Based on Use Case Points"
- [7] D. S. Kushwaha and A. K. Misra, "Software Test Effort Estimation," *ACM SIGSOFT Software Engineering Notes*, Vol. 33, No. 3, May 2008.
- [8] M. Chemuturi, "Software Estimation Best Practices, Tools & Techniques: A Complete Guide for Software Project Estimators," J. Ross Publishing, Lauderdale, July 2009.
- [9] R. S. Pressman, "Software Engineering – A Practitioner's Approach," 5th Edition, McGraw Hill, New York, 2002.
- [10] B. T. Rao and B. Sameet, "A Novel Neural Network Approach for Software Cost Estimation Using Functional Link Artificial Neural Network," *International Journal of Computer Science and Network Security*, Vol. 9, No. 6, June 2009, pp. 126-131.
- [11] H. Zeng and D. Rine, "Estimation of Software Defects Fix Effort Using Neural Network," *IEEE 28th Annual International Computer Software and Applications Conference (COMPSAC'04)*, Los Alamitos, 28-30 September 2004, Vol. 2, pp. 20-21.
- [12] K. K. Agarwal, P. Chandra, *et al.*, "Evaluation of Various Training Algorithms in a Neural Network Model for Software Engineering Applications," *ACM SIGSOFT Software Engineering Notes*, Vol. 30, No. 4, July 2005, pp. 1-4.
- [13] S. Nageswaran, "Test Effort Estimation Using Use Case Points (UCP)," *14th International Software/Internet Quality Week*, San Francisco, 29 May-1 June 2001.
- [14] T. E. Hastings and A. S. M. Sajeev, "A Vector-Based Approach to Software Size Measurement and Effort Estimation," *IEEE Transactions on Software Engineering*