VOL. 10, NO. 3, FEBRUARY 2015 ISSN 1819-6608

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

A SDLC DEVELOPED SOFTWARE TESTING PROCESS USING DMAIC MODEL

Oythip Onsuk[1], Pongpisit Wuttidittachotti[1], Somchai Prakancharoen[2] and Sakda Arj-ong Vallipakorn[3]
[1]Faculty of Information Technology, King Mongkut's University of Technology North Bangkok, Thailand
[2]Faculty of Applied Science, King Mongkut's University of Technology North Bangkok, Thailand
[3]Faculty of Medicine, Ramathibodi Hospital, Mahidol University, Bangkok, Thailand
E-mail: oythip.onsuk.99@gmail.com,

ABSTRACT

This study aim to resolve the software testing processes by create reliance of a system. With traditional testing processes, there are many issues of unacceptable defects found after the end of testing processes. To solve this problem, we applied quality management according to Six Sigmas quality improvements. From the principles of DMAIC, they found the most of mistakes came from runtime error, logical error and syntax error at 3.83%, 2.83% and 5.50%, respectively. This research consists of five stages of problem identification, the root cause analysis to find out the problems, drawn tree and fishbone diagrams help to analyze and solve problems. The quality improvement concepts were implement by using experiment designed techniques which controlled by standard software testing in the final step to ensure that the problems will not occur again. The results show that using quality management with the principles of DMAIC integration can reduce defects referring to Run Time error from 3.83%, 2.83%, 5.50% to 2.67%, 1.33%, 3.83%. This benefit will improve the confidence level, and raise the good image of the company.

Keywords: software testing, CMMI, six sigma, DMAIC, information system.

INTRODUCTION

This research focuses on software testing processes using technical knowledge to identified errors and mistakes (Ng, 2005). A case study selected one of the software house organization that had local and international business operations in the field of software development services. The software configuration-management (SCM) was one component in the process of creating a software quality standard of CMMI (Capability Maturity Model: CMMI). Because of SCM and CMMI were appropriated approach to software development. The recently report showed increases ability of the organization to manage the software development projects (Brayton, 2009). That means this prototype help to provide quality and promote the trusted image of an organization to customers. The CMMI processes addressed the steps of development. It began with the conceptual designed, development, final implement, and then through to the maintenance steps. Disadvantageous of CMMI process integration usually affects to each artifact, each item and each mistake in the processes caused a lot of work in the document generation area. This research looks for the methodology to solve this problem to reduce errors and loss of earnings from quality of products and services.

Development of the organization used elements of a Six Sigma Quality Improvement which focused on reducing errors and eliminating problems to provide an efficient guideline processes– DMAIC (Define, Measure, Analyze, Control, and Improvement) (Kaur, 2005). As part of the method, defining the problems was the goal of the project management, especially measuring characteristics of the current process and collect relevant data, then analyzed in order to verify and confirm the relationship between causes and effects. Once verified, all relationships and factors deliberated to find out the causes of defects under investigation. The updating or enhancing of the current process were depended on the data analysis process, which used error-checking techniques to ensure the standardization and creates a new process in the future.

These problems must be minimize the error in the manufacturing process. This objective aimed to provide a DMAIC process to solve this problem, increases customer trust, reduces cost, time consumption, and barriers of software testing as in Figure-1.

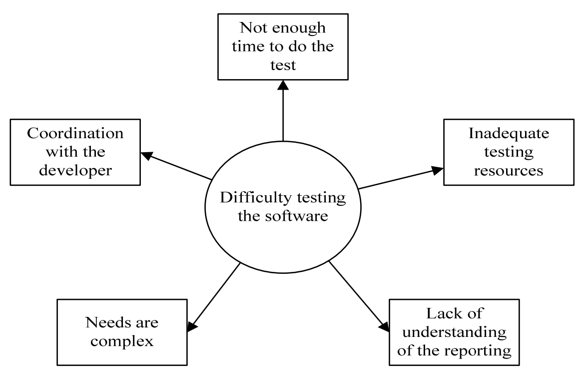

Figure-1. Barriers of the software testing.

RESEARCH SIGNIFICANCE

Principle of Six Sigma was an important quality improvement process based on customer satisfaction or customer-based center (Yaacov, 2000). The concept of Six Sigma management attempted to reduce the defects that could not be met the customer requirements. That means, the need to know the requirements and expectations of the

ARPN Journal of Engineering and Applied Sciences

customer were very important. This research addressed to the first stage of development by using the actual requirement and facts from customer. The management of Six Sigma complied on appropriated statistical model to create the decision support based on factual evidences to take more advantage. In other words, the Six Sigma was an improvement process that aims to improve the quality by re-arrangement of management and processes to generate the best of products and minimize errors (Suratkar, 2002).

The Six Sigma contained an analytic process and evaluation to continuity improvement, which consists of the proactive management that focused on dealing with the problems, emphases on the important of customer, and finds a problem by exploring directed causes and eliminates the root causes of the burden problem (Santorin, 2003). There were three major goals to successfully archive when trying to satisfy customers and reduced the cycle to minimize the defects. The three major process elements of Six Sigma were as follows (Mandl, 1985). The first element processes of improvement used to search the problems, approached to improve the existing processes. Further steps applied to get rid of the existing problems, and the last steps of the first element discovered ways to control the permanent of best results. The second element is the design process (Process design/redesign) in cases when the organization have chosen a new product to develop, this process used instead of the correcting the previous defects or added a new service/product rather than tried to update from previous mistaken. Because the original process involved the improving of the previous process, this was not enough to beat other competitors or achieve customer requirements (Tatsumi and Keizo, 1987). This struggles brought to develop new concept, which designed to achieve maximum customer satisfaction with minimized defects. This new concept aimed to achieve the highest quality by applying designs of Six Sigma quality improvement in to the new concept (Design for six sigma-DFSS) (Brownlie, 1992).

The third element interested in process management (Process management) (Bernstein, 1993). The Six Sigma process could not be fully creative with sustainable results without the participation of quality and process management appropriately (Tatsumi and Keizo, 1987). This means that the management had to determine the direction and strategy of the organization, using leadership strategy to create a quality culture in the development of the Six Sigma. This included finding out customer needs, seeking development opportunity, quality monitoring, as well as trying to control sustainable results of development in the organization. Tatsumi, et al. highly emphases the third element as the leadership quality of Six Sigma.

PROCESS SIX SIGMA

D-Define was the first step of Six Sigma to define the topics and scopes of the project (Lyu, 1996). This project implemented to improve or change the objectives beginning with the search for the true customer (Apichat, 2012). Followed the customer needs, which met the customer's specification, or what could compete in the same business to draw the target of project (Von, 1993). In addition, it needed to define the scope of the project to ensure the project had an appropriate size and direction within a timeframe. Usually, we marked steps by writing a process map to clarify the work process involved in each step from the beginning to the project's completion. Commonly, each project takes at least 3 months to complete, there needs joint of projects among several involved people who come from different departments. These needed to define and understood the framework of the project to ensure effective collaboration.

M-Measure was a collection of information of theirs output (Gokhale, 1997). The services of the process, started from a defined data input plans, data formats, method of data storage in which appropriated to requirements. After that, data evaluation process was done to reflecting effectiveness and performance of the process compared to the target, which relevant with customer needs and specification (Michael, 2008). In Six Sigma, anything that does not conform to the target counted as defects and Sigma Level reflecting the occurred chance of defects.

A-Analyze was the analytical process to make the assumption why the output did not met the goal of customer needs (Michael, 2008). This was the same as the cause of defect (Xs): the mathematical equation was Y = f (Xs). If the targets were not met in Six Sigma, they considered defect (Y), so this gathering of statistical data statistically analyzed what factors affect the defects and then arranged in order of precedence to determine the causes, and secondary causes (X1, X2, X3, ...) (Bubevski, 2008). The Six Sigma working steps must be verified and clarified, and not rely on beliefs or feelings for the final decision. There was a variety of statistical tools that must be chosen correctly fit with the data and the process worked to provide precision analysis. Then the results can be trusted (Bubevski, 2009).

I-Improve, could analyze until the main causes were knew (X1) (Gokhale, 1997). This step directly addresses the improvement process by focusing on eliminating or reducing main cause of problems in Six Sigma that must to evaluate (Paul, 1983). Each X was able to deliver results to improve as many of Y values especially useful in the field of academic and cost-benefits. Because some changes may require more investment, it necessary to study the ways of improved work. The guidelines applied evaluate the most appropriate guidelines or sorted option on before-after, thus appropriated according to the real situation and their expense (Paul, 1983).

C-Control was the final stage of the Six Sigma project. It was an important step, especially after an updated or after changed to improve procedures of work, it necessary to place the control system to maintain the results for long-term. If one needed control, it required

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

both creating an accepted new process with continuous evaluation of risks which needed to be analyze to ensure no adverse effects.

PRELIMINARY INVESTIGATIONS

I. Problem defining processes (Define Phase)

a) The overviews of software testing process studies from case study of the software companies. The authors started with system testing referring to education, studied the process to test system, assigning personnel, a device used for the test, to prepare the data in testing and testing case's document prepared as a basic knowledge of research to understand before making analyses and process improvements (Thomas, 2005). By gathering information from a document in the system, the quality of each of the relevant procedures, the data collected will be used to display information in tables, graphs and flow charts to show in each step to make more understand (Schaef, 1999) (Burke, 2002).

b) Select a sample in a study using the method of group selection (Cluster Sampling) those were divided into 3 groups of 12 systems based on the relationship of each group and selected the 3 systems to study (Wang, 2006) (Petrovic, 2004). Those were System Security Module, Card Interchange System Module and System link Dispute Manager Module. After already made a selection to collect classified data from the fault testing system from retrospective considered to System Security Module Card system link Module and System Manager Module Interchange Dispute to choose problems that occurred and affected the most quality for improvement (Ming-Hsien, 2008). The data collected has made the process of successful end of each process, and then have collected the fault information from all qualifying issues that were significant. Then, using the principle of pareto to prioritize of these problems.

c) Identification of the research problem, sorted from most fault information primarily to find defects as possible into the analysis made in order to find the problem that caused most of these faults. The information provided on each of the issues that arose for finding fault percentage and cumulative percentage used to qualify the significant problems (Anite SAS, 1999).

d) Team preparation of related security problems were a system of systems, three modules system, link Manager, and system Module Card Interchange Dispute Module, divided into 3 groups of works by testing the old system and to test by the Six Sigma. The audit team used in the same series (Biehl, 2004), (Deming, 1975).

II. The measurement procedure (Phase Measure)

a) Measurement procedure, an error value as an introduced to improve was started from creating a flow chart (Mapping Process) of the test system in the production of software. Awareness of the factors and the relation of each process then leaded the associated factors of knowledge problems and created a flow chart (Mapping Process), a conceptual chart, tree or fishbone diagram, respectively. To use the analyses of the problems and questions why these problems occurred (Tree Why-Why) In order to show causes and effects related to the problems that resulting from this process (Vriendt, 2002), (Florac, 1999).

b) Determining the process flow chart of the Process Mapping to study the software test process map (Graham, 1999). The procedure consisted of several working steps. The first step of studied of quality and development of production will be aware of the factors and relationships in each step of the process. The team must have an understanding of the level that was capable of providing more details of their duties; responsibilities reside in the production process in order to be able to identify the problem. That may be the cause of the fault (Harry, 1998). The result of this step, noted which opportunity step to pose problems and aware of the severity of the problems that arose from the process to arrange the order in which to consider the information and to edit it-have to do further diagnostic analysis.

c) Provide a reason using the Fishbone chart (Cause and Effect Diagram) and tree maps (Why-Why Tree Diagram) were consider to the relationship of causes and effects of the issues. The affected factor and distinguish according to the characteristics of this type of research selected by using Fishbone chart, tree chart, and that analyzed the question why the problem occurred (Tree Why-Why Diagram) (Pettichord, 1996), (Humphrey, 1998). Because of they wanted to make the team aware of themes and approached to analyze problems systematically. By showing the relationship of the problem with the map in the form of a reason that it was easier to understand and be able to identify the cause of the problem was clear. The initial reason given by the Tree diagram in the last slot will be the determining factors in making the analyses with the statistical principles (Rahnema, 1993).

III. The analysis phase (Analyze Phase)

a. Experimental method to find new faults proportion of trials required to find a reason to verify if the reason was not able to confirm the faith of needed to accept the alternative (Graham, 1999). We chose it because of there was no reason of fail to reject alternative. In contrary, if the reason was to confirm the belief of the acceptable test (Harry, 1998). The reasons for the beginning of each treatment was analyzed by trials to find out how a new type of defect and measure the rate of crashes by controlling the other constant factors, then the method used to compare the original method of hypothesis testing to demonstrate the significantly differences (Humphrey, 1998).

ARPN Journal of Engineering and Applied Sciences

IV.The Process improvement (Improvement Phase)

a. The design factors that affect the process as much as possible for the experiment because it made work more efficient. By studying the old software and system test to test the Six Sigma for development (Siebra, 2007).

b) The present guidelines to revise the fault issues when we had the most appropriate values of each of the individual processes that were fault. Get the value of the response variables from the best process. The next step will be proposed to revised guidelines, the error problems, and some of fault issues that were not difference but can be better improved (Walsh, 2004).

c) The original test, see Figure-2.

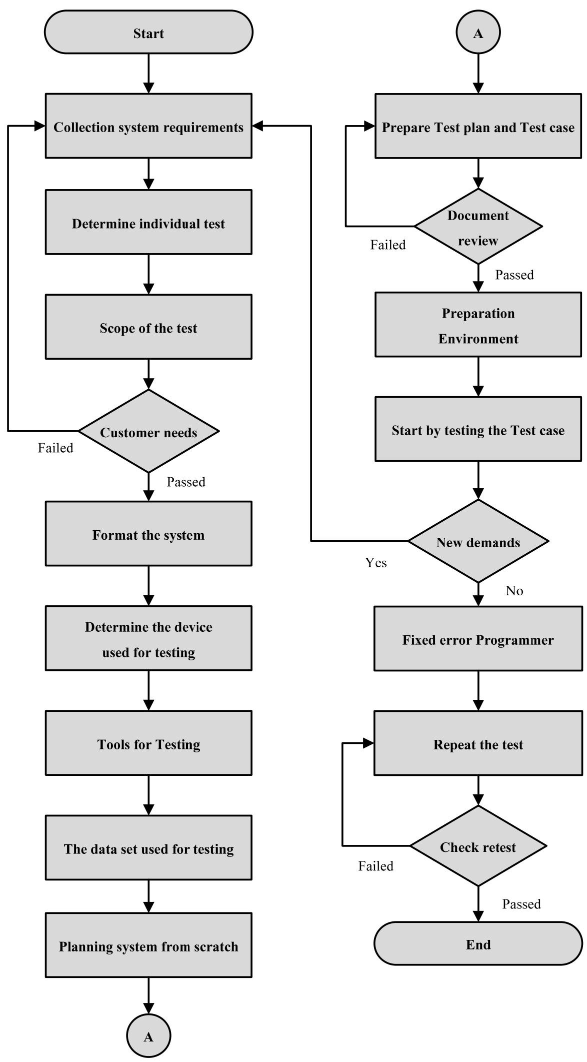

Figure-2. Traditional software testing process.

d) The new testing process diagram of the Six Sigma testing, see Figure-3.

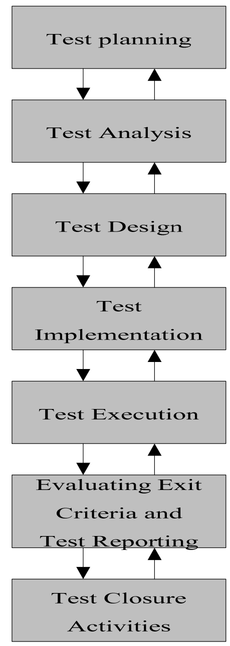

Figure-3. The process of software testing in Six Sigma.

- Test analysis to define test scenario that corresponded to the Test Requirement, Functional Specification, Specification System and Business Process Specification for use in the design, development, Test Case and Test Script, as well as to prioritize testing of the subject (Tortoise, 2008).
- Test Case and Test Design by prepared the Test Script that were used in the test, which was a detailed Checklist covers every Feature or Function of the software. Under test process when the testing was completed to identify the results (pass/fail), and any other details that were necessary evidence to be used as a Checklists for delivering or receiving grants (Raghuraman, 2001), (Simon, 2007).
- Test Implementation activities under the implementation process such as;
- Produce a table in the Test Execution Schedule Test
- Prepare Environment and data for use in testing.
- Prepare the process that used to keep track of test results.
- Prepare the process that used to track the defect caused by the tests.
- Prepare tools
- Test execution and test execution schedule
- Do saving, editing, defect tracking that occurs.
- Track the progress of testing.
- Test controls according to plan.
- Evaluating Exit Criteria and Reporting to check the test results (Alexandre, 2006). For example, the testing reports of all Test Execution Schedule or Defect has been corrected that caused all of the testing

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

and whether or not and also the test progress report to know, such as;

- The number of test case, pass/fail test, or in the process of being tested.
- The number of defect found by severity (Severity) (Status)
- The number of change requests (CR) which were occurred, etc. test closure activities an activity that was subject to closure activities such as test events.
- Check the completeness of the test that accordance with the test execution schedule, and whether the defect were detected, or not found (Allen, 2003).
- Continue to deliver the work, incident data, test results that generated by the tests and reports processes, those involved the acknowledgement (Bae, 2007).
- Storing important documents and test results information incident arising from the various reports on the test and its storage (Configuration Management System)

V. The revision procedure (Phase Control)

The monitoring and evaluation of work performed by comparing the experimental results (Design of Experiment) were used to find out the optimal conditions of improvement (David, 2005). Because of the improved performance was satisfied. Thus, controls (Control) system worked by creating a standard of work (Operation Standard) for the process to avoid those problems repeated.

APPLICATION MODULES

I. Process of selecting problem (Define Phase)

Storage in proportion to the problem of software testing from April to May period 2014. Approximately six weeks, the number of tested cases, 300 test was divided into 50 cases a week tested in phase 1 (Phase 1) had encountered a problem Error Run-time in case of defects (errors during execution, caused unwittingly: Run-time Error) The operations review was a 3.83%, 5.50%, 2.83 respectively can be seen in Figure-4.

Appearance defects	Test case (Security Module)							
	April				March			
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Average	SD
Run-time Error	9.00	7.00	5.00	1.00	1.00	0.00	3.8333	3.7103
Logical Error	8.00	7.00	4.00	1.00	0.00	1.00	3.5000	3.3912
Syntax Error	2.00	2.00	2.00	0.00	1.00	2.00	1.5000	0.8367
Total	19.00	16.00	11.00	2.00	2.00	3.00	8.8333	7.5741
Appearance defects	Test case (Interchange Module)							
	April				March			
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Average	SD
Run-time Error	5.00	5.00	4.00	2.00	1.00	0.00	2.8333	2.1370
Logical Error	3.00	2.00	2.00	0.00	1.00	2.00	1.6667	1.0328
Syntax Error	2.00	1.00	0.00	1.00	1.00	1.00	1.0000	0.6325
Total	10.00	8.00	6.00	3.00	3.00	3.00	5.5000	3.0166
Appearance defects	Test case (Card link Dispute Manager Module)							
	April				March			
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Average	SD
Run-time Error	10.00	10.00	5.00	3.00	3.00	2.00	5.5000	3.6194
Logical Error	10.00	8.00	5.00	1.00	0.00	1.00	4.1667	4.1673
Syntax Error	1.00	1.00	0.00	0.00	0.00	2.00	0.6667	0.8165
Total	21.00	19.00	10.00	4.00	3.00	5.00	10.3333	7.8909

Figure-4. The number of flaws of each system.

e. The problem analysis can be done using the 3W2H question as follows: (Brodman, 1994)

- What: found a defect in a test case Where Error Run-time: error during operation did not understand the test data.
- When: April to October
- How: a report from the quality inspection
- How much: 3.83%, 2.83 %, 5.50%

Above information showed the main problems that occurred and affected the most quality problems Error Run-time by selecting this issue to continuity develop and improve in the next step.

II. The measurement procedure (Phase Measure)

The study on workflow processes (Mapping Process) was the study of the process and workflow processes. It possible to show the process of running a test case called the Run-time Error. That was an error that was not a Syntax error, but had a logical error is caused by a zero-divisor, or errors that occur while the program was operated by either due to unexpected conditions (Dorling, 1993) (Patcharin, 2005). The unexpected conditions have shown in Figure-5.

VOL. 10, NO. 3, FEBRUARY 2015 ISSN 1819-6608

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

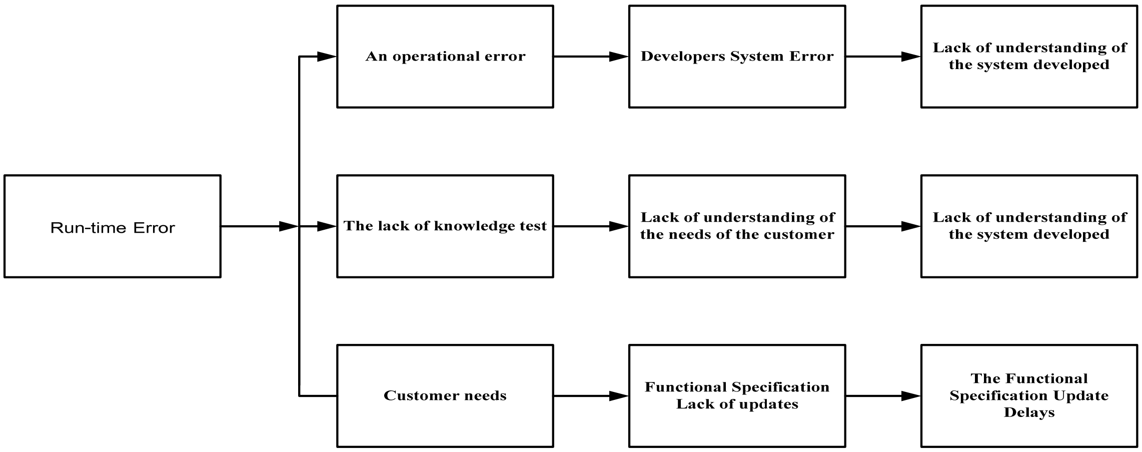

Figure-5. Shows the factors that affect the occurrence of problems.

Measurement (Measure Phase) was found to cause a major impact study that caused the three problems, which consisted of operational program error due to lack of customer knowledge tests. The analysis was done by three reasons why, find out the average of each system and compare your system's reliability (Garcia, 2003).

III. The analysis phase (Analyze Phase)

See Figure-6.

IV. The process improvement (Improve Phase)

Experimental analysis from the process mapping was examined to identify the defect factors. The causes of defect were found and adjusted to three main factors. By the proposed ways to improve, the scaling procedure by standard six sigma was applied to improve from the original test process, monitor the process of the adjusted settings and planning and execution process of working.

Appearance defects	Test case (Security Module)							
	April				March			
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Average	SD
Run-time Error	7.00	5.00	3.00	1.00	0.00	0.00	2.6667	2.8752
Logical Error	6.00	6.00	3.00	0.00	0.00	1.00	2.6667	2.8048
Syntax Error	1.00	1.00	0.00	0.00	0.00	0.00	0.3333	0.5164
Total	14.00	12.00	6.00	1.00	0.00	1.00	5.6667	6.0882
Appearance defects	Test case (Interchange Module)							
	April				March			
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Average	SD
Run-time Error	3.00	3.00	1.00	1.00	0.00	0.00	1.3333	1.3663
Logical Error	2.00	2.00	1.00	0.00	0.00	0.00	0.8333	0.9832
Syntax Error	3.00	1.00	0.00	0.00	0.00	1.00	0.8333	1.1690
Total	8.00	6.00	2.00	1.00	0.00	1.00	3.0000	3.2249
Appearance defects	Test case (Card link Dispute Manager Module)							
	April				March			
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Average	SD
Run-time Error	8.00	8.00	5.00	1.00	0.00	1.00	3.8333	3.6560
Logical Error	7.00	5.00	3.00	0.00	0.00	0.00	2.5000	3.0166
Syntax Error	0.00	0.00	0.00	0.00	0.00	0.00	0.0000	0.0000
Total	15.00	13.00	8.00	1.00	0.00	1.00	6.3333	6.6232

Figure-6. Six Sigma is the number of defects.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

V.The process control (Phase Control)

Control of defects, which earned both direct and indirect results. This was needed to control and prevented the problem through monitoring problems, and shown abnormalities of the process. This phase needed to control both the internal and external factors by doing the design and establish follow-up methods and quality control operations (Garcia, 2003), (Goldenson, 1995).

a. Stages control to test the software division of most Runtime Error. Control at this stage has made modifications to the details of the software test process.

b. Results of observation after defected of control defects. From the purpose of this research aimed to reduce the problem of software test process by applying principles of DMAIC. After an operation to correct the problem, perform the check new data capture results show dancing after adjustment on the part of the operating result to comparison the effect of Error Runtime problems before and after to making the adjustment, as well as control factors.

CONCLUSIONS

This study aims to solve the problem of the software testing process by use the example case of Accellence Company, Thailand by conducting a study on the problems and Run-time Error, Logical Syntax Error, consisting of a system of work by using the DMAIC process. Mostly of the Six Sigma approach performance are as follows:

a) Factors that affect Run-time Error problems, other errors, Logical, and Syntax Errors.
b) Operational error
c) The lack of knowledge test and customer needs

This research also illustrates the principle of DMAIC can be used as a tool to reducing many error in software testing processes such as Run-time problem, Bug, Error, Logical and Syntax Error that mostly occurred in the software testing process. Because of they can analyze to find the causes of the problem with brainstorming, collecting all of causes as well as to find out the cause of the problems, defect work processes using charts from Fishbone (Fishbone Diagram) and Tree (Tree Diagram). Moreover, we can determine the problem, consideration, execute on the experimental design to see whether appropriate factors that are causing the problem. Then the run-time error, syntax error, logical error was solved until reduction of defect reflected the use of Six Sigma, standard and quality improvement can reduce the number of defects, and enhancing more quality of work.

REFERENCS

Apichat S. (2012). Process Improvement Using Six Sigma Concept: Case Study of Hard Disk Manufacturing by DMAIC, Rajamangala University of Technology Thanyaburi.

Allen P., Ramachandran M. and Abushama H. (2003). PRISMS: An approach to software process improvement for small to medium enterprises. ICQS '03: Proceedings of the 3rd International Conference on Quality Software.

Alexandre S., Renault A. and Habra N. (2006). OWPL:A Gradual Approach for Software Process Improvement In SMEs. InEUROMICRO-SEAA.

Anite S. (1999). 12447D/UMOOI GSM-9OO/DCSI800 /PCS-1900. Stand Alone Simulator User Manual, rel. 3.0, AniteTelecoms Ltd, Fleet, Hampshire, UK.

Bae D H. (2007). Software process improvement for small organizations. In COMPSAC '07: Proceedings of the 31st Annual International Computer Software and Applications Conference - Vol. 1, 2007.Washington, DC, USA. IEEE Computer Society.

Bernstein L. and Christine M. Y. (1993). Testing Network Management Software. Journal of Network and System Management.

Biehl R. (2004). Six Sigma for Software, IEEE Software.

Brayton R. and Cong J. (2009). Electronic Design Automation Past, Present, and Future. NSF Workshop Report. July.

Brodman G. and Johnson D L. (1994). What small business and small organizations say about the cmm: experience report. In ICSE '94: Proceedings of the 16th International Conference on Software Engineering, Los Alamitos, CA, USA.

Brownlie R., James P. and Madhav S. P. (1992). Robust Testing of AT&T PMX/StarMAIL Using OATS. AT&T Technical Journal, Vol. 71. No.3, May/June.

Bubevski V. (2008). An Application of Simulation in Software Reliability Prediction. Palisade™ Risk & Decision Analysis Conference, New York City, 13-14 November.

Bubevski V. (2009). A Simulation Approach to Six Sigma in Software Development. International Simulation Multiconference, Istanbul, 13-16 July.

Burke E. K. and Petrovic S. (2002). Recent Research Directions in Automated Timetabling. EJOR. 140(2).

David W. (2005). CMMI and Six Sigma Synergy. In SEPG '05: Software Engineering Process Group, Washington State Convention Center, Seattle, WA USA, March. Carnegie Mellon University, Software Engineering Institute.

ARPN Journal of Engineering and Applied Sciences

Deming W. (1975). On probability as a basis for action. The American Statistician.

Dorling, A. (1993). Software Process Improvement and Capability Determination. Software Quality Journal.

Florac W. and Carleton. A. (1999). Measuring the software process: statistical process control for software process improvement. The SEI Series in Software Engineering, Addison-Wesley.

Frederick P., Brooks Jr. (1995). The Mythical Man-Month(Essays on Software Engineering, Anniversary Edition). Addison-Wesley.

Garcia, S., Miluk, G., Cepeda, S L. and Staley, M. (2003). CMMI FOR SMALL BUSINESS PILOT project. In 3rd Annual CMMI Technology Conference and User Gsroup, November.

Gokhale S.S., Lyu M.R. and Trivedi K.S. (1997), Reliability Simulation of Fault-Tolerant Softwareand Systems. Proceedings of Pacific Rim International Symposium on Fault-Tolerant Systems.

Goldenson D. R. and Herbsleb J. D. (1995). After the Appraisal. A Systematic Survey of Process Improvement, Technical Report 95tr009, Software Engineering Institute, Pittsburgh, August.

Graham D. and Fewster M. (1999). Software Test Automation:Effective Use of Test Execution Tools. Addison Wesley.

Harry M. (1998). Six Sigma: A Breakthrough Strategy for Profitability. Quality Progress.

Humphrey W. (1998). Characterizing the Software Process. IEEE Software.

Kaur J. (2005). A Balanced Scorecard for Systemic Quality in Electronic Design Automation: An Implementation Method for an EDA Company. ISQED.

Leveille R. (2011). Test Strategy for High Quality EDA Software. WCSQ.

Lyu Michael R. (1996). Handbook of Software Reliability Engineering. IEEE Computer Society Press

Robert M. (1985). Orthogonal Latin Squares: AnApplication of Experiment Design to Compiler Testing. Communications of the ACM. Vol. 128, No.10, October.

Robert C.T., and Michael R. L. (2008). Software Reliability Simulation. Chapter 16, Handbook ofSoftware Reliability Engineering, IEEE ComputerSociety Press.

Li Ming-Hsien C., Al-rafaie A. and Cheng-Yu Y. (2008). DMAIC Approach to Imprve the Capability of SMT Solder printing Process. IEEE Transactions on Electronics Packaging, Manufacturing. Vol 31, No. 2, April.

Ng A. I., and Markov L. (2005). Toward Quality EDA Tools and Tool Flows through High-Performance Computing. ISQED.

Paul, B., Bennett, L., Fox., Linus, E., and Schrage. (1983) . A Guide to Simulation. Springer-Verlag, New York.

Patcharin O. (2005). Integrating Lean Six Sigma and CMMI into Enterprise by System Dynamics: Case Study: Spansion (Thailand) Limited. Industrial Engineering, King Mongkut's Institute of Technology, North Bangkok.

Pettichord B. (1996). Success with Test Automation, Proceedings of the Ninth International Software Quality Week. San Francisco, California, USA.

Petrovic S. and Burke E. K. (2004). University timetabling. Ch. 45in the Handbook of Scheduling: Algorithms, Models, and Performance Analysis (eds. J. Leung), Chapman Hall/CRCPress.

Raghuraman R. (2001). EDA software: quality is not optional. Ubiquity. June.

Rahnema M. (1993). Overview of the GSM system and protocol architecture. IEEE Communications Magazine.

Reuven Y. R. and Dirk P. K. (2008). Simulation and the Monte Carlo Method, John Wiley & Sons, New Jersey.

Santarini M. (2003). Panelists find EDA tool quality lacking. EETimes.

Schaef A. (1999). A Survey of Automated Timetabling. Artificial Intelligence Review.

Siebra C., Freitas A., Costa P., Freitas R., Silva F. and Santos. (2007). A. An Investigation into the Use of AI Planning for Handsets Network Test Automation, Proceedings. International Conference on Artificial Intelligence and Applications, Innsbruck, Austria.

Simon K. (2007). DMAIC versus DMADV, Six Sigma WebPage,2007.Available.in:http://www.isixsigma.com/library/content/c001211a.asp

Tatsumi. and Keizo. (1987). Test Case Design Support System. Proceedings of ICQC, Tokyo.

Thomas P., (McGraw-Hill). (2005). The Six Sigma Handbook.

ARPN Journal of Engineering and Applied Sciences

TortoiseSVN. (2008). http://tortoisesvn.tigris.org/, June.

Von Mayrhauser, A, et al. (1993). On the need for simulationfor better characterization of software reliability. Proceedings. Fourth International Symposium on Software Reliability Engineering.

Vriendt De., Laine J., Lerouge P. and Xiaofeng, X. (2002). Mobile network evolution: a revolution on the move. IEEE Communications Magazine.

Wang Z. and Sun J. (2006). Application of DMAIC on Service Improvement of Bank Counter. IEEE.

Walsh J. (2004). combining the Power of DMAIC with Testing Process, Six Sigma WebPage, 2004. Available in: http://finance.isixsigma.com/library/content/c04040.

Yaacov G. B., Bjork L. and Stone E. P. (2000). Advancing Customer-Perceived Quality of the EDA Industry. ISQED.

Yaacov G. B., Suratkar P., Holliday M. and Bartleson K. (2002). Advancing Quality of EDA Software. ISQE.