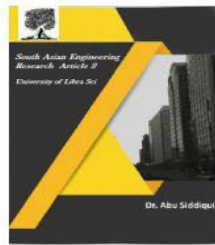# IMPROVING SOFTWARE QUALITY & SOFTWARE MEASUREMENTS STATISTICAL ANALYSIS

[1]DR VUPPU PADMAKAR , [2]DR. B V RAMANA MURTHY

[1]Associate Professor, Department of CSE, Methodist College of Engineering & Technology, Hyderabad

[2]Professor, Department of CSE, Stanley College of Engineering & Technology for Women, Hyderabad

[1]drvuppu@gmail.com, [2]drbvrm@gmail.com

ABSTRACT

Programming item quality can be characterized as the highlights and attributes of the item that meet the client needs. The nature of any product can be accomplished by following a very much characterized programming process. These product procedure results into different measurements like Project measurements, Product measurements and Process measurements. Programming quality relies upon the procedure which is completed to outline and create programming. Despite the fact that the procedure can be completed with most extreme consideration, still it can present some mistake and deformities. Process measurements are extremely helpful from administration perspective. Process measurements can be utilized for enhancing the product improvement and support process for deformity evacuation and furthermore to diminish the reaction time.This paper depicts the significance of catching the Process measurements amid the quality review process and furthermore endeavors to order them dependent on the idea of blunder caught. To decrease such mistakes and deformities discovered, strides for remedial activities are prescribed.

KEYWORDS

Software measurement, Quality improvement, Corrective actions

1.INTRODUCTION

The nature of programming is of most extreme significance in the field of programming designing. Programming quality likewise relies upon the procedure which is completed to plan and build up the product. Indeed, even after the procedure is pursued with ulmost care, the mistakes and imperfections may in any case exist. The nature of a product item is for the most part dictated by the nature of the procedure used to construct it. Estimation and investigation will help in deciding the status of the product procedure as far as whether the procedure is pursued and the working is as expected. Check is the comparable sort of control from the administration point of view. To meet such objectives, quality review for programming process are directed time to time. By estimating the blunders and deformities, we can find a way to enhance the process.The change of process will rely upon measurements caught in the lifecycle of programming. Programming measurements can be arranged into Project measurements, Product measurements and Process metrics[1].Project measurements are those
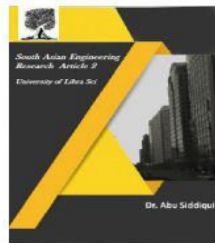
that portray the venture attributes and aid execution arranging. Item measurements catch the properties of programming like interim to disappointment. Process measurements are administration measurements which are utilized for enhancing the product improvement and support process for imperfection evacuation and lessening reaction time of the procedure. Process measurements are priceless device for an association who needs to enhance their procedure. Normally these procedure measurements are not utilized generally as a result of vulnerability about which measurements to utilize, how to perform estimations and how to defeat such defects.For programming process change, there are numerous models which are accessible for instance Capability Maturity Model (CMM) [2], Bootstrap, Personal Software Process (PSP) [3], IT Infrastructure Library (ITIL) [4], IEEE [5], Six Sigma [6] and ISO 9000 quality administration framework [7]. These models assess the product item, quality and their downside. In addition privately planned activities can be started in regions where change is required. The product procedure must be characterized and recorded. Notwithstanding the procedures, guidelines for the diverse work items to be characterized, e.g. coding and record standards.The rest of this paper is sorted out as pursues. In area 2, we have introduced our methodology and goals. In segment 3, programming process change models are depicted. In area 4, we have exhibited the writing audit which is premise of our work. In segment 5, quality practices are clarified, in next segment classification of mistakes

and imperfections are exhibited. In area 7, we have introduced restorative activities. In segment 8, information gathering philosophy is clarified. In area 9,results and the examination are talked about. Future work in the equivalent is proposed in area 9. At last we have given end in segment 10.

2.APPROACH AND OBJECTIVE

In this paper we have connected measurable quality affirmation to the blunders and imperfections revealed amid the quality review for the year 2015 and 2016 in our association. This has been done in view to enhance the nature of programming advancement process and henceforth the product items. We are displaying that, by estimating the blunders and imperfections we can take activities to enhance them. We are likewise introducing how every last mistake and imperfection are gathered. There after every one of them is ordered dependent on effect of seriousness like minor, moderate or genuine. The information gathered over a time of two years has been dissected and exhibited. The examination likewise portrays prescribed activities for the remedial activity. The thought has been roused from the product building experts Roger S Pressman and Bruce R Maxim [8].

Extensively we are attempting to address 3 targets specifically quality change, classifying of mistakes and suggestion of corretive activities.

3.SOFTWARE PROCESS IMPROVEMENT MODELS

Few software process models commonly followed worldwide are CMM, PSP, ITIL, IEEE and Six Sigma. Capability Maturity Model (CMM) [2], is a reference model for
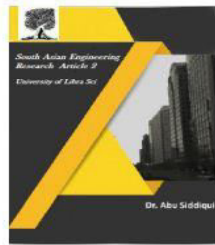
advising the product procedure development into different levels [10]. The distinctive levels of Software Engineering Institute CMM have been thought with the goal that it is simple for an association to manufacture its quality framework. CMMI planned to propel the convenience of development models by coordinating a wide range of models into one casing work.Personal Software Process (PSP) [3], advocates that architects should rack the manner in which they apply time. The quality and yield of an architect is to an incredible degree dependent on the procedure being pursued. PSP is a system that causes designers to measure and advancement. It helps in creating individual abilities and methodologies by assessing, arranging, and following execution against plans, and conveys a characterized procedure which can be changed by architects [9]. IT Infrastructure Library (ITIL) [4] depicts forms, methods, errands, and agendas which are not association particular, but rather can be connected by an association for setting up combination with the association's methodology, conveying esteem, and protecting a base level of competency. It enables the association to set up a standard from which it can plan, execute, and measure. It is utilized to show consistence and to quantify advance.

IEEE [5], gauges affiliation is a gathering inside IEEE that creates worldwide models in an expansive scope of businesses including, control, vitality, biomedical, social insurance, data innovation, mechanical autonomy, media transmission, home mechanization, transportation, nanotechnology, data confirmation, and numerous more [10]Six Sigma [6] can be utilized for any action that is worried about cost, auspiciousness and nature of results. A definitive goal of six sigma hone is the usage of an estimation constructed procedure that concentrates in light of process upgrade [9].

4.RELATED WORKS

In [11] the creators from Laboratoire de génie logiciel École Polytechnique de Montréal, Canada has displayed quality assessment procedures for particular areas or particular strategies. Regularly the product item designers select a pre-characterized show, redo the highlights, characterize the measurements and gauge the nature of the product item. However, in this paper the creators introduces a base up strategy for the quality gauge process. Likewise a technique planned and proposed for expected quality profile. Fundamentally, the initial step is tuning in to the clients, and afterward recovering the most vital quality factors and making a model to assess the normal nature of the undertaking. The profile is shaped by creating the normal clients' quality desires, and after that measuring the evoked factors by applying them to our quality assessment demonstrate and the ISO/IEC 25000 standard.

In [12] the creators have introduced the instrument of how programming building capacities identify with the business execution. They have proposed an auxiliary model including the Software Engineering Excellence pointer which comprised of expectations, venture administration, quality affirmation, process change, innovative work, human asset improvement and client contact.
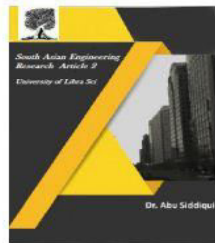
In [13] the segment based programming advancement approach is talked about and illustrated. Creators have proposed quality confirmation demonstrate for segment based programming improvement which incorporates prerequisite elicitation, plan advancement, affirmation, customization, joining, testing and support.

In[14] the creator has shared how NASA's Johnson Space Center built up a 'factual strategy' to decide test estimate for the quantity of process errands to be inspected by SQA. The objective of this work is to deliver a top notch item which is financially savvy.

In [15] this paper the oppressions among prerequisites and engineering parts are talked about with the goal that product deformities can be moderated.

In [16] creators have said that innovative options are crucial for task arranging, asset portion, and nature of the last programming item. For investigation they have taken open source web applications accessible in SourceForge. Creators intend to give devices to help venture administrators. They have said that there is have to set thumb manage to direct mechanical decisions to build the nature of programming ancient rarities.

In [17], the creators have presented the development of programming quality model benchmarks and the actualities of SQuaRE arrangement principles. The lacks of ISO/IEC 2502n programming quality estimation arrangement guidelines were dissected and a guide of new reference demonstrate is proposed.

This paper [18] is identified with programming item quality displaying and estimation. The result of the exploration is gathered as framework level programming quality models, source code component level programming quality models and utilizations of the proposed quality models.

Our work accentuations on applying factual quality affirmation to propel the nature of programming items.

5.QUALITY PRACTICES

The International body, ISO is resolved to give prerequisites, rules, particulars so on which can be utilized for creating quality structures for items and administrations of little and huge associations for any sort of undertakings. ISO International principles guarantee that items and administrations are solid and of good quality. Its specialized advisory groups involves significant industry specialists, buyer affiliation, the scholarly world, NGOs and government [7].

ISO 9001:2008 models set out the criteria for a quality administration framework. The standard features quality standards like client center, top administration inspiration and constant change based process approach. It tends to be utilized by any association, extensive or little, paying little respect to its field of action. Truth be told is executed by more than one million organizations and associations in more than 170 nations. This standard depends on various quality administration standards including a solid client center, the inspiration and ramifications of best administration, the procedure approach and nonstop change [7].

Our association is ISO 9001:2008 ensured. ISO 9001 process is pursued for the advancement of programming items. The ISO related exercises are for the most part
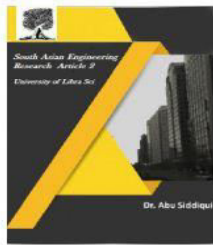
done by the quality confirmation group. The primary job of value confirmation group is guaranteeing quality administration framework conformance, advancing client center, and covering quality administration framework execution. A quality administrator is a representative who has been given this obligation. Observing the quality goals that have been built up and announcing this to 'Administration Review' is another job of the quality supervisor. Administration survey concentrates more on the product procedure instead of the product work products.Quality chief is additionally in charge of interior review arranging and administration. Inner review is the taught way to deal with assess and enhance the viability of programming quality procedures. The extent of inside review is primarily chance administration, control and administration of programming forms. Interior reviews are finished by the quality affirmation group to check the accessibility of the records and to guarantee that all the critical and fundamental parameters were canvassed or not as far as non-conformance points.The center segments of programming improvement are Software Requirement elicitation, Design stage, Implementation stage and Testing stage. IEEE Std 1074 is Standard for Software Lifecycle which essentially covers the above recorded stages. Necessities Engineering Process catches the prerequisites tending to the usefulness, execution, qualities, limitations, HR, equipment and programming interfaces. The characteristics like compactness, viability, security are additionally tended to. Required gauges and working condition are recorded. Process additionally catches limit of the

framework, expected clients, add up to clients, greatest clients at any one time, sort of clients et cetera. Expectations like framework encourage records, manuals, documentation, source code, preparing and bolster angles are additionally mentioned.Overall the necessity procedure endeavors to bring the lucidity, culmination, consistency, traceability and practicality viewpoints. Any adjustment in prerequisites is managed change administration by organizing them. Any change is assessed dependent on the achievability and dangers in accomplishing the newrequirement. The Software Requirement Specification report is a brief archive catching the above viewpoints which experiences a companion audit process and proposed changes are acknowledged or dismissed after the discourse. At long last the endorsement of the archive by the undertaking director turns into the pattern for the whole lifecycle of the task.

Configuration stage catches the outline detail. It gives an abnormal state outline of how the usefulness and obligations of the framework were divided and after that appointed to subsystems or parts. A depiction of all information structures including interior, worldwide, and transitory information structures are recorded. Reference to information lexicon and information stream outlines (DFD) are made amid prerequisites investigation. A definite depiction of every product segment contained inside the design is displayed. Reports all the plan qualities like execution contemplations, dependability, compactness, UI, subtle elements for the safeguarding of
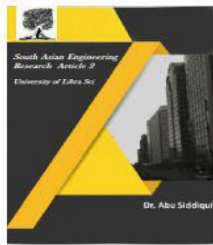
items and so on. Plan confirmation is brought through the Technical Review or Design Walkthrough. Unit test cases or System test cases are set up for the Design approval. On the off chance that any extra highlights have been included the Design stage, the equivalent is reflected in the System Requirement Specification. It additionally catches the traceability grid of prerequisites building. The finished result of the outline stage is the plan report which experiences the specialized associate survey and affirmed for further execution.

In the execution stage, coding begins according to the task. Coding is done according to the coding rules. Record header is incorporated with legitimate name, way, form, no., portrayal, capacity, and technique names. Variable naming tradition is as indicated by benchmarks. Inline remarks are available wherever essential, portraying the current code squares. Code is indented and lucid. Capacities utilized in excess of one program units are placed in the library records. The coding standard shifts from the decision of programming dialect. In the event that the venture has embraced possess standard or rules check are recorded. The deviations from the standard/rules are justified.Testing process covers the testing exercises did at different periods of programming advancement. Testing exercises incorporate, test arranging, outlining experiments, executing the experiments, assessing the product dependent on test outcomes, estimating and investigating test information. Experiments are intended for confirming every prerequisite. Experiments for unit tests are related to the info and yield information.

Joining testing recognizes of the earth required for coordination basic modules and calendars of testing. Framework testing is done to approve the product item against the prerequisite particular. Here characteristics, for example, outside interfaces, execution, security, design affectability, conjunction, recuperation and unwavering quality are approved amid this stage. A progression of tests are performed to guarantee that the framework fulfills all its practical and non-utilitarian necessities.

6.   ERROR AND DEFECT CLASSIFICATION

Information accumulation of different programming parameters and estimation gives bits of knowledge to extend administration group and administrators. The estimation is conceivable by first gathering quality information and afterward it very well may be contrasted and past information and assess whether changes has happened. The product can be estimated dependent on Project, Product and Process and consequently can be named Project measurements, Product measurements and Process measurements [1]. Undertaking measurements catch deserts, cost, calendar, profitability and estimation of task assets and expectations. Item measurements measure cost, quality and time to showcase. Process measurements are identified with quality process pursued for programming advancement. They measure the productivity and adequacy of different procedures. Process measurements can be methodicallly caught from the product quality reviews. Programming quality review is an autonomous and deliberate examination for deciding any deviation from
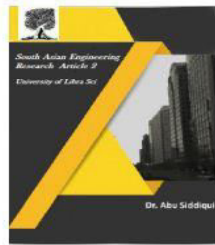
the arranged exercises. A review is the examination of the work items and related data to evaluate whether the standard procedure was pursued or not. The information for our examination is collectedfrom the "Examiner Note Sheet". The gathered information is dissected dependent on its temperament and characterized into different kinds like mistaken particular, error or deficient or off base documentation and so forth.

1.Incomplete or incorrect details - Any particular inadequacy is caught in this class. Any deviations from the procedure manual or particular like endorsement missing, incomplete usage and so on are incorporated. On the off chance that any missing measurements in the determination/format is additionally considered as mistake under this classification.

2.Violation of programming guidelines - Any deviation from benchmarks or presentation or alteration can be tallied in this class.

3.Error in information portrayal - Any deviation from information designs as pronounced in particular.

4.Inconsistent equipped interface - Any deviation from prescribed interface related mistakes.

5.Error in outline rationale - Any deviation from submitted rationale eg DFDs, UML or ER graph.

6.Incomplete or incorrect testing - Any blunders and imperfections revealed in testing by partner/client/outsider client and so on after fulfillment of testing.

7.Intentional deviation from determination -

It identifies with deviation from process manual, programming necessity particular and so on because of absence of appropriate reasons.

8.Inaccurate or deficient documentation - Any missing sub areas of process manual or inadequate documentation.

9.Assorted mistake compose - Any different blunders and deformity not caught in previously mentioned classifications.

The majority of the above classifications are additionally arranged dependent on the seriousness of the mistake/abandons. They are named as minor, moderate and genuine. It is delegated minor if the blunder/imperfection not basic to affect the procedure. Essentially, the deformity is delegated moderate if the procedure is seen to be pursued yet can't be confirm. On the off chance that the blunder or imperfection is seen to have significant deviation from process then it is sorted as genuine.

7.CORRECTIVE ACTION

For every one of the blunder and imperfection ordered over, a restorative activity is suggested as talked about underneath;

1.Incomplete or wrong determinations - Effective Peer Review to be led.

2.Violation of programming norms - Reason to be caught for purposeful infringement and same to be looked into.

3.Error in information portrayal - Recommend to utilize devices for information displaying additionally perform more stringent information configuration audits.

4.Inconsistent able interface - Recommend more proper specialized audits and trainings.
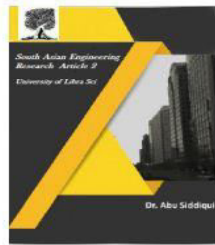
5.Error in outline rationale - Recommend

more suitable specialized surveys and trainings.

6.Incomplete or mistaken testing - Recommend to embrace more suitable testing approachs with legitimate test designs.

7.Intentional deviation from detail - Reasons to be caught for purposeful deviation and same to be surveyed.

8.Inaccurate or fragmented documentation - Recommend to utilize devices for documentation and audits.

8.DATA COLLECTION

At C-DAC, [19] the product quality review is directed quarterly. Review is directed for each venture which is in configuration stage, advancement stage and support stage. Quality confirmation group reveals the timetable with date, time, venture name, auditee, evaluator, and scene. With this auditee will keep prepared all report and points of interest required for review. After the review inspector will submit "Reviewer Note Sheet" to quality affirmation group. Evaluator note sheet contains review blunders and imperfections, assuming any. Quality confirmation group distributes the whole "Examiner Note Sheet" in ISO related intranet site where all C-DAC individuals approach these Note Sheets

Table 1- Error Categorization for year 2016

Error Type	Serious Errors	Moderate Errors	Minor Errors
Violation of programming standards	0	0	0
Incomplete or erroneous specifications	1	2	11
Error in data representation	0	0	0
Inconsistent competent interface	0	0	0
Error in design logic	0	0	0
Incomplete or erroneous testing	0	0	0
Intentional deviation from specification	2	1	0
Inaccurate or incomplete documentation	0	0	0
Assorted error type	0	0	0
Total	3	3	11

Table 2 - Error Categorization for year 2015

Error Type	Serious Errors	Moderate Errors	Minor Errors
Violation of programming standards	0	0	0
Incomplete or erroneous specifications	1	2	6
Error in data representation	0	0	0
Inconsistent competent interface	0	0	0
Error in design logic	0	0	0
Incomplete or erroneous testing	0	0	0
Intentional deviation from specification	3	1	0
Inaccurate or incomplete documentation	0	0	0
Assorted error type	0	0	0
Total	4	3	6

For our experiment we have taken 2 years data namely Year 2016 and Year 2015. Based on our quality assurance guidelines of our organization these errors and defects are grouped as serious, moderate and minor which is described in section 6. Also based on its nature every error or defect is categorized as per section 6, same is recorded in the Table 1 and Table 2. Figure1 and Figure2 capture the severity of the errors thus categorized.
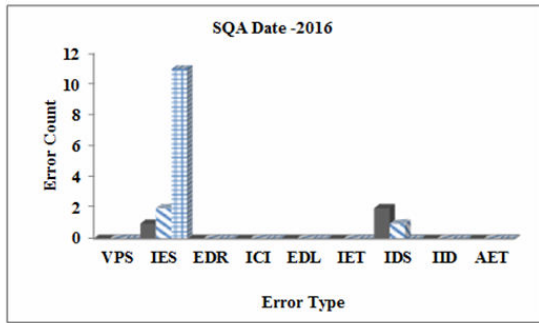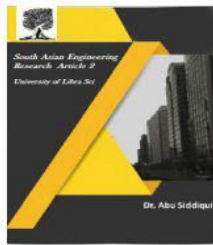
Figure 1- Severity of errors captured for year 2016

Figure 2- Severity of errors captured for year 2015

Table 4 – Cumulative errors for 2 years

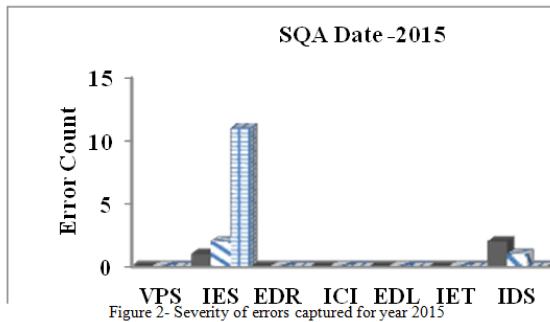| Sl No	Year	Total errors
1	2016	17
2	2015	13
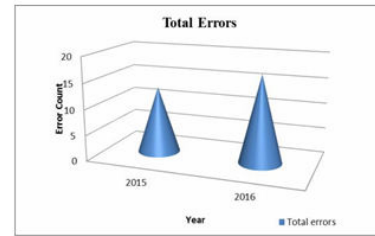
Figure 3 – Projection of errors
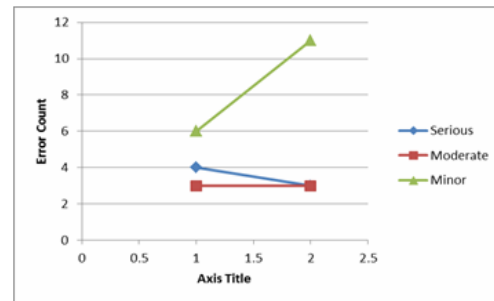
Figure 4 – Cumulative Projection of Severity errors

9.ANALYSIS AND RESULTS

Consistently three inward reviews and one outer review's are led. Interior review is directed by Software Quality Assurance group of C-DAC, outer review is led by outside experts. Amid the review, examiners will recode their perception, blunders and deviations. This is named as "Non Conformity-(NC)" in "Inspector Note Sheet" proclamation. We have gathered all the NC's accounted for, same is arranged according to area 6 and assembled as genuine, moderate and minor. The aggregate genuine, moderate and minor blunders of both the years are spoken to in Table 3 and Table 4. Figure3 and Figure4 ventures the aggregate mistakes for a long time.

Table 3 – Severity of Cumulative Errors

Type of errors	Year 2015	Year 2016
Serious	4	3
Moderate	3	3
Minor	6	11

Every one of the mistakes and deformities are sorted and gathered fundamentally to know the measurements of programming nature of tasks. The information spoke to in Table 3 is gathered from 9 ventures. The tasks are either in plan, usage or upkeep states. The ventures have a place with different areas, for example, disseminated processing, cryptography, elite figuring, Internet of things, versatile applications and so on. These tasks are actualized in programming dialects java, C, python and other scripting dialects. A portion of these are utilizing databases.

In Table 3, it is recorded that in year 2016 by and large mistake revealed was 17. Out of which 3 are not kidding, 3 are moderate and 11 are minor compose. The one genuine blunder was because of Incomplete or wrong
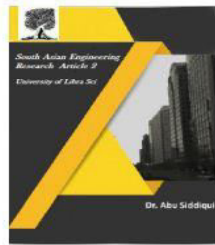
determinations viable 'peer audit process' was suggested. Remaining 2 genuine blunders was because of Intentional deviation from particular – reason was Work Breakdown Structure was not refreshed, endorsement was not taken in time and so on. Every one of the reasons for blunders were examined and preparing given on quality process. Likewise, there were 2 moderate and 11 minor mistakes because of 'Deficient or mistaken determinations' and one more was because of 'Purposeful deviation from detail'. In both the case powerful associate audit process and preparing on quality process was suggested. Comparative investigation was conveyed for the year 2015.The target of the paper is to gauge the blunders and deformities (non congruity) of the considerable number of activities, audit it and suggest the suitable remedial activity. With the goal that the task cost won't over shoot, it very well may be conveyed in time additionally the nature of the undertaking will increment. Henceforth programming nature of items conveyed by association progresses.

10.FUTURE WORK

Here we are describing the work of error categorization. After collecting error and defect information, error index can be calculated. In future, we intend to calculate the phase index and error Index, which is an overall indication of improvement in software quality.

11.CONCLUSIONS

To enhance the product quality, we gathered programming Process measurements. Our center was chiefly towards gathering measurements acquired through the quality control process. The mistakes and imperfections found through the product quality reviews was the base information. These deformities were in this manner ordered into nine sorts. Deformities are investigated, proposals for enhancing such imperfection are suggested.The change process was proposed which primarily comprise of short preparing on commonality with Software Process, prescribed specialized surveys and gathering dialogs for accomplishing the higher quality. The means were investigated where imperfection happened, recognized and expounded for stepwise remedies. Effective utilize case was exhibited through a change program. The powerless regions of deformities were recognized and master help was bestowed to determine them. Wonderful change saw in nature of programming item in the wake of actualizing the proposal to the mistakes and imperfections found.

REFERENCES

[1]Ashwin Tomar and V. M. Thakare, "The Survey of Metrices on Software Quality Assurance and Reuse", National Conference on Innovative Paradigms in Engineering & Technology (NCIPET-2013)

[2]Capability Maturity Model, http://cmmiinstitute.com/

[3]Software Engineering Institute, https://resources.sei.cmu.edu/library/

[4]Information Technology Infrastructure Library, https://en.wikipedia.org/wiki/ITIL

[5]IEEE Standards Association, http://standards.ieee.org/

[6]Six Sigma measurement standards, https://www.isixsigma.com/

[7]International Organization for Standardization, www.iso.org

[8]Roger S. Pressman, Bruce R. Maxim,

"Software Engineering: A Practitioners Approach, Eighth Edition", 2015, McGraw-Hill Education.

[9] Rajib Mall, "Fundamentals of Software Engineering", 4th Edition, PHI Learning Private Limited.

[10]https://en.wikipedia.org/wiki/IEEE_Standards_Association

[11]Reza Mirsalari, Pierre N. Robillard, "Expected Software Quality Profile: A methodology and a case study", 978-1-5090-0996-1/16/$31.00 ©2016 IEEE

[12]Yasuo Kadono, Hiroe Tsubaki, Seishiro Tsuruho. 2008. "A Study on Management of Software Engineering in Japanese Enterprise IT Industry", 978-1-4244-3397-1/08/$25.00 ©2008 IEEE.

[13]Ravi Kumar Sharma & Parul Gandhi, "Quality Assurance of Component Based Software Systems", 2016 International Conference on Computing for Sustainable Global Development (INDIACom), 978-9-3805-4421-2/16$31.00@2016 IEEE

[14]Neera Bansal Talbert, Paramax Space System, Texas. "Representative Sampling within Software Quality Assurance", 1063-677393 $3.00 0 1993 IEEE.

[15]Bashar Nassar and Riccardo Scandariato, University of Gothenburg, Sweden, "Traceability Metrics as Early Predictors of Software Defects?", 2017 IEEE International Conference on Software Architecture, 978-1-5090-5729-0/17 $31.00 © 2017 IEEE, DOI 10.1109/ICSA.2017.12

[16]Valentino Sartori, Birhanu Mekuria Eshete, Adolfo Villafiorita. 2011. "Measuring the Impact of Different Metrics on Software Quality: a Case Study in the Open Source Domain", ICDS 2011 : The Fifth International Conference on Digital Society, ISBN: 978-1-61208-116-8.

[17]Jiayu Gong, Jiawen Lu and Lizhi Cai "An Induction to the Development of Software Quality Model Standards", 2016 Third International Conference on Trustworthy Systems and Their Applications, 978-1-5090-3539-7/16 $31.00 © 2016 IEEE, DOI 10.1109/TSA.2016.28

[18]Peter Hegedus, University of Szeged, Hungary. 2015. "Advances in Software Product Quality Measurement and Its Applications in Software Evolution", ICSME 2015, Bremen, Germany.

[19]Centre for Development of Advanced Computing, www.cdac.in

[20]Chandramouli Subramanian, Chandramouli Seetharaman, B. G Geetha Saikat Dutt, "Software Engineering", 2015, Pearson India Education Service.