Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2017, 4(4):266-272**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Investigating Training Program and Intergroup Coordination in relation to Peer Review in Nigerian Software Companies

Aregbesola, Moses Kehinde

Department of Mathematics and Computer Science, Elizade University, Ilara-Mokin, Ondo State, Nigeria

Abstract Training Program (TP), Intergroup Coordination (IC) and Peer Reviews (PR) are key process areas (KPA) at the defined level of maturity (i.e. maturity level 3) of the Capability Maturity Model Integration (CMMI). TPs are typically carried out to ensure that the personnel are well equipped with the requisite skills necessary to perform their duties. IC on the other hand is aimed at establishing a means for the software engineering team to work actively with other teams so that projects are able to better meet the needs of customers with effectiveness and efficiency. PR is targeted at removing defects from the software work products early and efficiently. This paper is focused on investigating the performance of TP, IC and PR in the Nigerian software industry and the possible relationship between them. The survey and action research methods were employed in the study. The study involved the participation of twenty-six out of the thirty randomly selected Nigerian software companies. The study showed that the performance of TP, IC and PR is relative strong in the selected companies with the highest performance recorded in the PR KPA. The study equally showed a positive relationship between the three KPAs. The performance of these KPAs within the country can however be improved. These KPAs should therefore be given the required consideration to further strengthen them.

Keywords Investigation, Training Program, Intergroup Coordination, Peer Review, Nigerian Software Companies, Software Process, Software Industry, Nigeria

Introduction

Training Program (TP), Intergroup Coordination (IC) and Peer Reviews (PR) are key process areas (KPA) at the defined level of maturity (i.e. maturity level 3) of the Capability Maturity Model Integration (CMMI). The CMMI is made up of 5 maturity levels namely Initial, Managed, Defined, Quantitatively Managed, and Optimizing. Each maturity level consists of Key Process Areas (KPA) which in turn consists of key practices. [1-3].

Paulk *et al.* (1995; 1993) [4-5] and Mead *et al.* (1996) [6] described the concept of Training Program (TP) as consisting of the collection of related entities that focus on meeting an organization's training requirements. This generally consist of an organization's training plan, training facilities, training materials, development of training, conduct of training, evaluation of training and maintenance of records. The aim of the TP key process area (KPA) is to nurture the skills sets of team members so they can perform their roles with effectiveness and efficiency. TP typically starts by the initial identification of the training needs of the individuals, projects, and organization. The current and future skill requirement for each software project is evaluated. The necessary training is then subsequently developed or procured. However, the concept of training as used in the Capability Maturity Model Integration (CMMI) context is slightly broader than might usually be considered when using the term. Training is provided to make an individual proficient with specialized instruction and practice. This training may consist of informal as well as formal mediums for transmitting skills and knowledge to the personnel in the organization. Certain skills are imparted with efficiency and effectiveness through informal

mediums such as on-the-job training and informal mentoring, whereas other skills need more formal training mediums such as classroom training and guided self-study using formal mentoring and apprenticeship programs, facilitated video and computer aided instruction. The appropriate mediums are identified and adopted. The TP KPA describes the specific practices related to these training vehicles.

Intergroup Coordination (IC) is focused on establishing a means for the software engineering group to be in active participation with other engineering groups so that the project is more able to efficiently and effectively satisfy the needs of the customers. Paulk *et al.* (1995; 1993) [4-5] described IC as involving the active participation of the software engineering team with project engineering groups to address system-level requirements, objectives, and issues. The project engineering groups' representatives participate in the establishment of the system-level requirements, objectives, and plans by working with other stakeholders such as the customer and end users, as required. The established requirements, objectives, and plans then become the basis for all other engineering activities. Furthermore, the technical working interfaces and contacts between groups are planned and managed to ensure the quality and integrity of the entire system. Technical reviews and transactions are frequently conducted with representatives of the project's engineering groups to ascertain that all engineering groups are conscious of the status and plans of all the groups, and that system and intergroup concerns obtain appropriate attention. The software-specific practices related to these engineering tasks are described in the Requirements Management and Software Product Engineering key process areas.

Peer Review (PR) is aimed at early, efficient and effective elimination of defects from the software work products with the significant consequent effect of build a better understanding of the software work products and of the defects that can be prevented [4-5] . PR has equally been describes as a valuable and highly effective engineering technique that is drawn from Software Product Engineering and can be implemented by the use of structured walkthroughs, Fagan-style inspections, or a number of other collegial review methods [7-8]. PR entails a systematic inspection of software work products by the producers' peers to detect defects and parts where modifications are required. A number of studies have discussed the concept of defect detection and prevention as involving the analysis of defects that were encountered in the past and taking specific actions to prevent the occurrence of these types of defects in the future [9-13]. The defects may have been identified on other projects as well as in earlier stages or tasks of the current project through Peer Review. Both the project and the organization take specific actions to prevent occurrence or recurrence of such defects. The particular products that will go through a peer review are identified in the project's defined software process and programmed as part of the software project planning activities. The practices identifying the particular software work products that are subjected to PR are contained in the KPAs that define the development and maintenance of each software work product.

The present paper is aimed at projecting three key process areas at the Software Engineering Institute (SEI) Capability Maturity Model Integration (CMMI) maturity level 3. These key process areas, namely TP, IC and PR, were studied within the context of the Nigerian software industry. Other studies that have explored CMMI implementation in Nigeria include Aregbesola and Akinkunmi (2010a; 2010b) [14-15], Aregbesola *et al.* (2011) [10], Aregbesola and Onwudebelu (2011) [16], Aregbesola and Oluwade (2014) [9], and Aregbesola (2017) [17].

It should be noted that there is a difference between review and peer review. A review typically occurs at the end of a task and involves presenting a software work product, or set of work products, to stakeholders (such as managers, the customer, and end users) for their comments or approval. At a peer review however, a software work product, or set of work products, is presented to the producer's colleagues to identify defects. The other stakeholders (Managers, the customer, and end users) are typically not present in a peer review. Peer reviews are an essential, in-process part of a task. They are implemented so that defects can be eliminated early, resulting in increased productivity and high-quality products. While some software work products will be reviewed, others will undergo peer review or both reviews and peer reviews.

In the present paper, a total of thirty Nigerian software companies were randomly selected for study using survey research and action research methods, out of which twenty six companies actually participated. The study showed that the performance of TP, IC and PR is relative strong in the selected companies with the

highest performance recorded in the PR KPA. The study equally showed a positive relationship between the three KPAs in the Nigerian software industry.

Research Methodology

The methodology employed in the current study is presented in this section. The survey research and action research were the two major research methods applied in the study. Using survey research, the software practices adopted by many of the Nigerian software companies were examined. A total of 30 Nigerian software companies were studied. Most of the companies, 27 of them, were based in Lagos, South-Western Nigeria. Three (3) were based in Asaba, in the South-South geo-political region of Nigeria. A total of Twenty six (26) (i.e. 86.67%) of the 30 selected companies eventually participated in the study. The sampling method was stratified from the perspective that the majority of Nigeria's software companies are based in Lagos. Lagos is commonly considered the commercial nerve centre of the country. The SEI Maturity Questionnaire [18] was used to gather information about software process implementation within the companies covered. This instrument served as the key data collection tool for the survey.

Some of the companies were considered for more detailed investigation using the action research approach. A direct observation of their activities and environment was carried out. Measurement of process-related phenomena was also performed. The companies affected were visited and observed over a period of time to see how they actually implement their software development process. Both structured and unstructured interviews were used to acquire information. Print and electronic documentation were equally explored as sources of useful details about the companies and their operations.

Research Outcomes

The results of the investigation carried out on the Nigerian software industry with regards Training Program (TP), Intergroup Coordination (IC) and Peer Review (PR) are summarized in Tables 1, 2 and 3 as well as Figures 1, 2 and 3.

Table 1: Outcomes for key practices in the Training Program (TP) KPA

Questions (Key Practices)	**Responses**			
	Yes%	**No%**	**NA%**	**DK%**
1. Are training activities planned?	58	35	0	8
2. Is training provided for developing the skills and knowledge needed to perform software managerial and technical roles?	73	12	0	15
3. Do members of the software engineering group and other software-related groups receive the training necessary to perform their roles?	38	27	31	4
Does your organization follow a written organizational policy to meet its training needs?	8	81	4	8
4. Are adequate resources provided to implement the organization's training program (e.g., funding, software tools, and appropriate training facilities)?	19	77	4	0
5. Are measurements used to determine the quality of the training program?	19	73	0	8
6. Are training program activities reviewed with senior management on a periodic basis?	15	73	0	12
Average	33	54	5	8

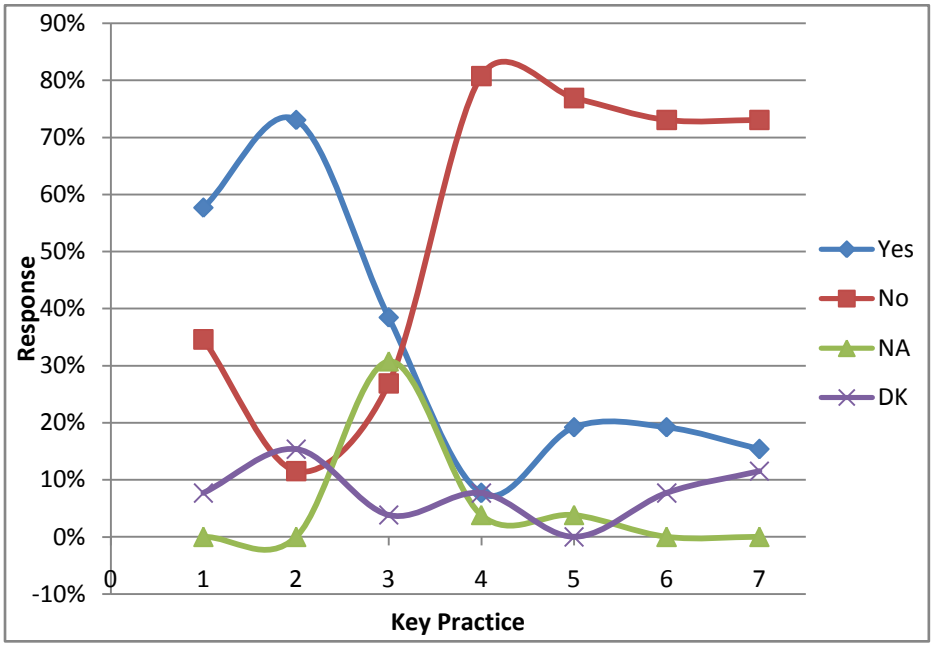

Figure 1: Chart of responses to key practices in the Training Program (TP) KPA

Table 2: Outcomes for key practices in the Intergroup Coordination (IC) KPA

Questions (Key Practices)	**Responses**			
	Yes %	**No %**	**NA %**	**DK %**
1. On the project, do the software engineering group and other engineering groups collaborate with the customer to establish the system requirements?	77	23	0	0
2. Do the engineering groups agree to the commitments as represented in the overall project plan?	62	19	4	15
3. Do the engineering groups identify, track, and resolve intergroup issues (e.g., incompatible schedules, technical risks, or system-level problems)?	19	65	8	8
4. Is there a written organizational policy that guides the establishment of interdisciplinary engineering teams?	4	77	8	12
5. Do the support tools used by different engineering groups enable effective communication and coordination (e.g., compatible word processing systems, database systems, and problem tracking systems)?	73	8	8	12
6. Are measures used to determine the status of the intergroup coordination activities (e.g., effort expended by the software engineering group to support other groups)?	15	85	0	0
7. Are the activities for intergroup coordination reviewed with the project manager on both a periodic and event-driven basis?	19	35	19	27
Average	**38**	**45**	**7**	**10**

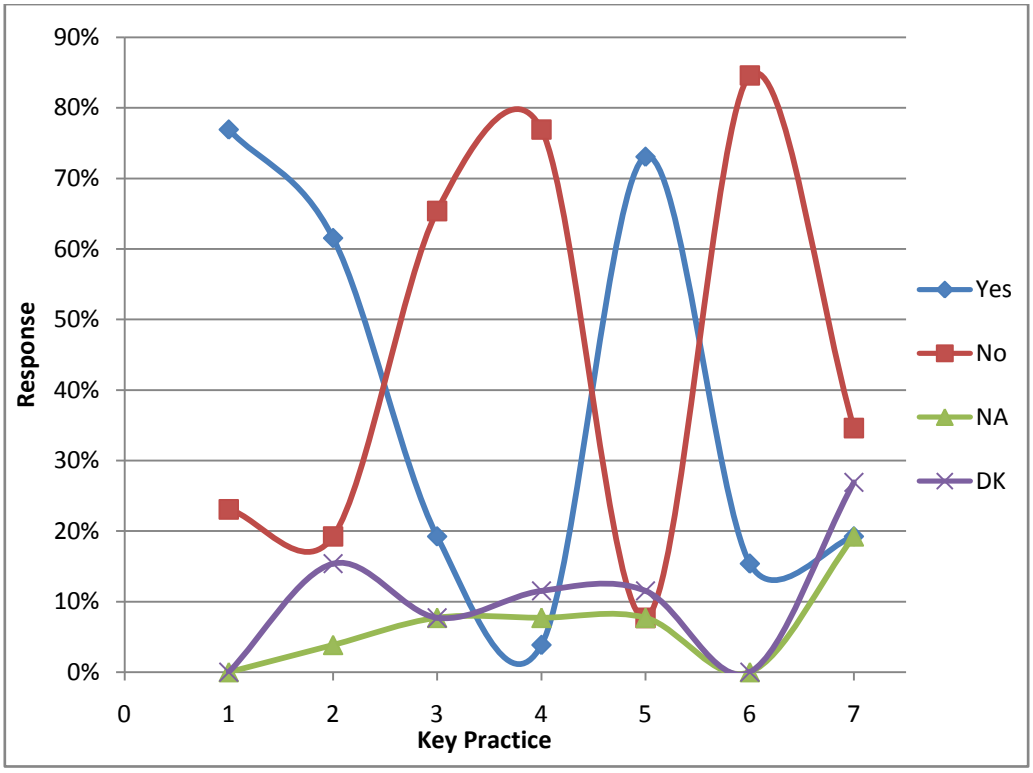

Figure 2: Chart of responses to key practices in the Intergroup Coordination (IC) KPA

Table 3: Outcomes for key practices in the Peer Reviews (PR) KPA

Questions (Key Practices)	**Responses**			
	Yes %	**No %**	**NA %**	**DK %**
1. Are peer reviews planned?	77	8	4	12
2. Are actions associated with defects that are identified during peer reviews tracked until they are resolved?	73	12	0	15
3. Does the project follow a written organizational policy for performing peer reviews?	15	77	4	4
4. Do participants of peer reviews receive the training required to perform their roles?	69	15	8	8
5. Are measurements used to determine the status of peer review activities (e.g., number of peer reviews performed, effort expended on peer reviews, and number of work products reviewed compared to the plan)?	77	15	8	0
6. Are peer review activities and work products subjected to SQA review and audit (e.g., planned reviews are conducted and follow-up actions are tracked)?	15	73	8	4
Average	**55**	**33**	**5**	**7**

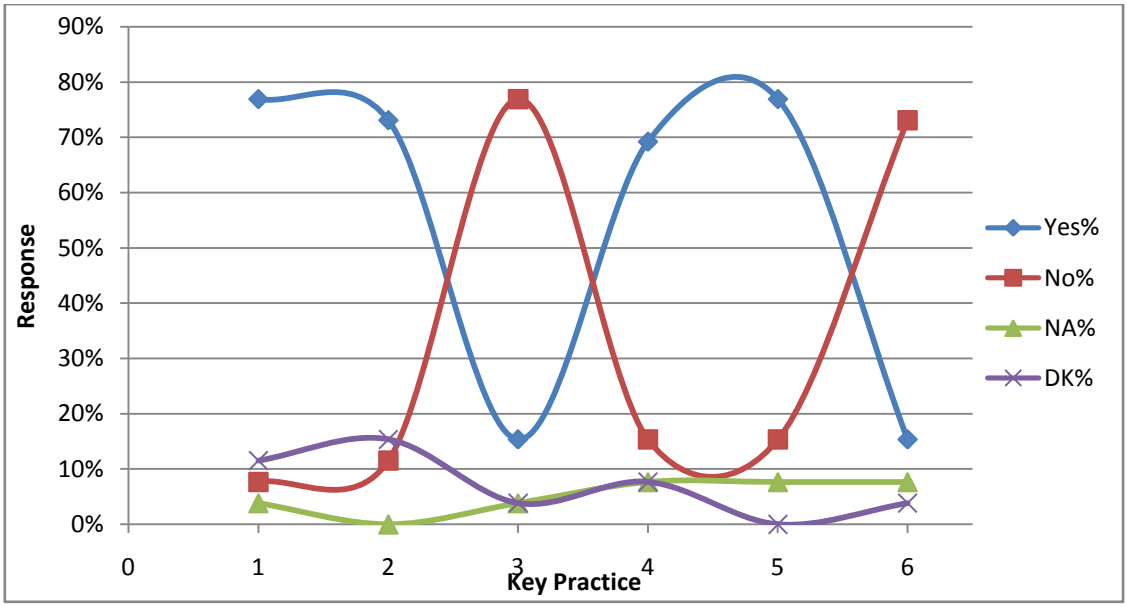

Figure 3: Chart of responses to key practices in the Peer Reviews (PR) KPA

Discussion

Extracts of the employed SEI maturity questionnaire [18] consists of two major sections. The first section is made up of questions regarding software process key practices within the KPA in organisation. The second section which is the response section consists of four response options namely "Yes", "No", "NA" for Not Applicable and "DK" for Don't Know. These four were the response options available to each respondent with regards to the organizations performance of the respective key practices in the questions section.

As depicted in Tables 1, 2 and 3, three KPAs (Training Program (TP), Intergroup Coordination (IC), and Peer Review (PR)) associated with the SEI CMMI Maturity level 3 as discussed in this paper have relatively good level of positive performance. The positive performance is as depicted by the Yes% responses to the performance of key practices within each KPA. Figures 1, 2, and 3 are graphical depictions of Tables 1, 2, and 3 respectively. The level of informedness about the KPAs is equally seen to be high as shown by the low DK% responses, which is an indicator of unawareness. The research outcomes equally showed an overall positive interaction among the three KPAs. The research outcomes show that the positive performance of TP and IC resulted in a much higher level of positive performance in PR.

Conclusion

This paper has focused on the investigation of the performance of three key areas of software process definition, namely, Training Program (TP), Intergroup Coordination (IC), and Peer Review (PR). By using survey and action research methods, it has been shown that the performance of these key process areas is relatively high in the Nigerian software industry. There however still exists plenty of room for better performance. It is therefore recommended that these KPAs be given the required attention so as to strengthen them and improve the overall maturity level of the individual companies and the entire industry with the country.

References

[1]. CMMI Product Team (2006). CMMI for Development, Version 1.2 - CMMI-DEV, V1.2. Software Engineering Institute, Carnegie Mellon University.

[2]. Glover M. T. and Dennie D. (2017). CMMI–Agile Process Combo: How to be Agile with CMMI. Excellence in Measurement Technology.

[3]. Hurst J. (2017). The Capability Maturity Model and Its Applications. SANS Software Security with Frank Kim.

[4]. Paulk M. C., Weber C. V., Curtis B., & Chrissis M. B. (1995). The Capability Maturity Model: Guidelines for Improving the Software Process. Addison – Wesley, Boston.
[5]. Paulk M. C., Weber C. V., Garcia S. M., Chrissis M. B., and Bush M. (1993). Key Practices of the Capability Maturity Model, Version 1.1. Technical Report CMU/SEI-93-TR-025 ESC-TR-93-178, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania.
[6]. Mead N., Tobin L., Couturiaux S. (1996). Best Training Practices within the Software Engineering Industry. Technical Report, CMU/SEI-96-TR-034, ESC-TR-96-134, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213. November.
[7]. Freedman D.P. and Weinberg G.M. (1990). Handbook of Walkthroughs, Inspections, and Technical Reviews, Third Edition. Dorset House, New York, NY.
[8]. Fagan M. E. (1986). Advances in Software Inspections. IEEE Transactions on Software Engineering. Vol. 12, No. 7, pp. 744-751.
[9]. Aregbesola M. K. and Oluwade B. A. (2014). An Experimental Evaluation of Defect Prevention and Change Management in Software Process Optimization in the Nigerian Software Industry. ARPN Journal of Systems and Software Vol.4, No.1, pp. 5-11.
[10]. Aregbesola M. K., Akinkunmi B. O., and Akinola O. S. (2011). Process Maturity Assessment of the Nigerian Software Industry. International Journal of Advances in Engineering and Technology (IJAET), Vol.1, Issue 4, pp. 10-25.
[11]. Snipes W., Robinson B., Yuepu Guo, Seaman C. (2012). Defining the Decision Factors for Managing Defects: A Technical Debt Perspective. Proceedings of 2012 Third International Workshop on Managing Technical Debt (MTD), pp. 54-60.
[12]. Huizinga D., Kolawa A. (2007). Principles of Automated Defect Prevention. In Automated Defect Prevention: Best Practices in Software Management, Wiley-IEEE Press Ebook Chapters, pp.19-51.
[13]. Eickelmann N. S. (2011). Empirical Studies to Identify Defect Prevention Opportunities Using Process Simulation Technologies. Proceedings of 26th Annual NASA Goddard Software Engineering Workshop, pp. 22-25.
[14]. Aregbesola M. K. and Akinkunmi B. O. (2010a). Software Process Implementation – A focus on the Nigerian Software Industry. Journal of Research in Physical Sciences, Vol. 6, No. 2, pp. 9 – 14.
[15]. Aregbesola M. K. and Akinkunmi B. O. (2010b). Software Process Implementation – A focus on the Nigerian Software Industry. International Research and Development Institute (IRDI), World Congress on Research and Development, Conference Center, University of Ibadan, 5th - 8th October. Vol. 5, No. 6, pg.111-116.
[16]. Aregbesola M. K. and Onwudebelu U. (2011). Typical Software Quality Assurance and Quality Management Issues in the Nigerian Software Industry. National Association for Science, Humanities & Education Research, 8th National Conference, University of Ado Ekiti, Ado Ekiti, September 14-17.
[17]. Aregbesola M.K. (2017). Experiential Appraisal of Organizational Process Focus and Process Definition in Nigerian Software Companies. Manuscript submitted for publication.
[18]. Zubrow D., William H., Jane S. and Dennis G. (1994). Maturity Questionnaire. Special Report CMU/SEI-94-SR-7.