

Systematic Mapping Study: Augmenting Personal Software Process Analysis For Extreme Programming Teams

Shahbaz Ahmed Khan Ghayyur¹, Abdul Razzaq^{2,*}, Asim Khalil¹, Ahmad Soban³, S. A. Hassnain²

¹ Department of Computer Science & Software Engineering, Faculty of Basic and Applied Sciences

² Information Science & Technology, Department of Marine Engineering

³ Department of Software Engineering, Faculty of Information & Communication Technology

*Corresponding Author: Abdul Razzaq, Emails: Dr.razzaq@zju.edu.cn, abdull.razzaq786@gmail.com

Abstract— The Personal Software Process offers individuals with a self-controlled structure for doing a job. To improve individual and team ability is a crucial source of productivity and quality. Measuring an individual's performance is a challenging task in an agile environment as individuals work on several projects at the same time. No specific criteria exist, which gives personal growth in agile XP. This research study is based on an idea to align the personal software process with agile extreme programming and propose a new model for an individual's professional growth measurement. An evidence-based case study is conducted to accumulate knowledge about the measurement of an individual's performance in the agile extreme programming team. In this study, systematic mapping is used to collect issues in existing literature. The reason for systematic mapping is needed to recap the enhancement and need to classify the holes also requirements for upcoming studies related to agile with process improvement. This study supports to realize the variance between SPS and XP. This scientist mapping makes mindfulness for the procedure improvement with a mix of SPS and XP. We also proposed a solution model which we have implemented in our research.

Keywords—*Personal software process, PSP, agile, individual improvement in agile, agile and PSP, collaborative software process vs personal software process, XP practicing, PSP and XP*

I. INTRODUCTION

A. Personal Software Process

The PSPSM delivers engineers with a well-organized individual framework to do work[1]. The PSP delivers a way to software engineers to advance the value, productivity, and probability of their work[2]. PSP aimed to report the enhancement of the needs of different engineers. This encourages designers to comprehend the nature of their work and to welcome the adequacy of the strategies they use [2]. PSP was presented by Watts Humphrey as "A Castigation" for Software Engineering [1]. The PSP process comprises a set of methods, script, and forms that provides a structured approach to individuals to plan and manage their work[2]. The PSP [3] is a controlled structure by which programming designers can design, track, and manage faults earlier to harvest good quality products[3] [1]. The main objective is to produce a quality product with a zero-defect product when individuals use PSP [2]. There are a lot of discrepancies found in the traditional model; a new model "Agile" was presented to cater to traditional model problems [4]. The PSP [3] is a planned context by which individual engineers to plan, track, and accomplish faults earlier to produce high quality products [1]. Data Collection and analysis is evidence of nominal process implementation in PSP[3].

B. Extreme Programming

A novel process called "Agile" and went on to overcome problems with conventional ways [3]. The traditional approach to a different way of adoption. For frequent are the methodologies under the canopy of "Agile" is more than a standard methodology extreme Programming (XP). A set of ideals and XP values and a deep, quickly develop software that offers quality is perfect competition. Russell Oberlin programming (XP) is lightweight, agile methods to the elastic developers 2 to 10 [15]. They are founded on a set of beliefs, XP nearby principles, values, and practices [16]. Principles are concepts defined to a high level, while these practices are that in the steps taken by the values [16]. Value to the troops bears responsibility values practices. The principles have been the gap between the practical values and a set of rules and practices used to describe the value's victory [17]. Value has five XP, communication, leisure, feedback, and a resolution to the resection [16] [17] the twelve practices [18]. Advantages of XP and a faster processor that includes software development, better quality, better customer satisfaction, and highly respected development teams motivated [19].

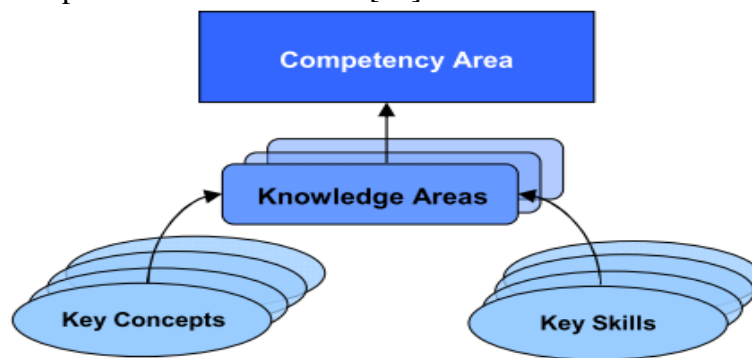


Figure. 1. Knowledge area classification [5]

C. PSP & XP

PSP covers different competency areas, and each competency area further classifies into knowledge areas, as mentioned in Fig. 1. Agile software development can state many specific software development methods, the best known Scrum and XP [5]. XP is a lightweight, agile, rebates, and methodology for developers 2 to 10 [6]. A set of XP has established principles and practices values [7]. The advantages of using the XP process model include faster time to market, form a more in-depth, more motivated development team's customer satisfaction, and honored [8].

D. The Need for Systematical Mapping

In this study, methodical mapping has investigated to discover issues that are looked at during the estimation of a person's exhibition [57]. the quality standards of PSP were investigated during methodical mapping, that can be utilized for a person's exhibition estimation in a coordinated group [58].

High-development improvement forms, utilizing measurements and quantitative strategies, for example, the PSP can create a lot of information to distinguish execution issues [28]. One collective blame of software engineering occupation is the weak value of products that software engineer harvests[37]. The uses of PSP notions and procedures in their effort, engineers in nearly any technical area can increase their planning abilities, the value of their work, and decrease the

number of faults in their yields[41] since PSP trusts profoundly on the group and analysis of private data as evidence of effective procedure execution[49]. The necessary concepts' descriptions and PSP's services can help the software specialists in evaluating their own abilities and services and in classifying areas in which they need of enhancement [4].

II. LITERATURE & BACKGROUND KNOWLEDGE

The PSP was presented by Watts Humphrey as "A Discipline" for Software Engineers [1] and a controlled structure by which programming specialists can plan, follow and oversee deserts prior to deliver top notch products[1]. Enhancements in hierarchical execution could likewise result from process improvement by singular designers. Humphrey perceived that an association arrives at a development level past which hierarchical procedure improvement requires singular procedure improvement.

Level0 is mainly the present process the individual engineer practice to write programs and software and to provide performance measurements. PSP level0 was improved to Level 0.1 by the addition of size measurement, coding standard, and personal process improvement. Level 0.1 also improves package size capacity, individually count procedures and methods. After Level 0.1, Level 1.0 enhances planning to Level 0, and the early step adds size and reserve estimation. In Level 1.1, planning and status tracking are introduced. In Level 2, a wide-ranging and constant emphasis is sited on refining the individual's capability to quality program production. The main objective is to produce quality work in a consistent way. The intent of introducing Level 2.1 is not to provide a designing process but to tell engineers how to state the criteria for design accomplishment. The Level 3 process presents methods for individuals to developing large scale programs[9].

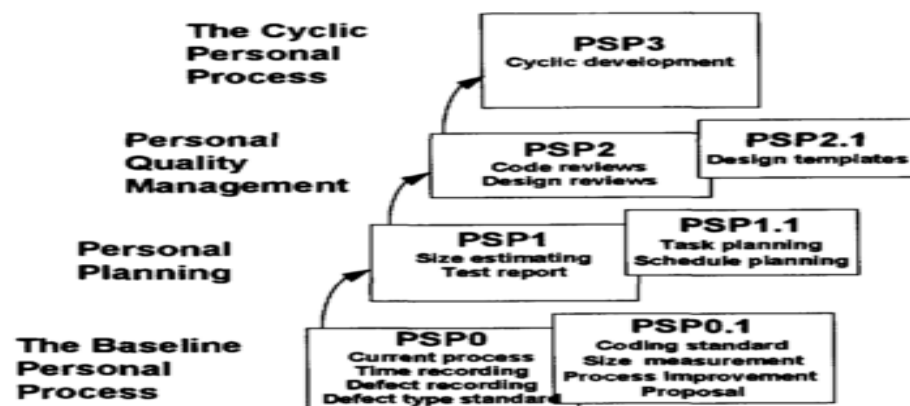


Figure. 2. Personal Software Process Phases[9]

III. RESEARCH METHODOLOGY

A. Mapping planing

This mapping is highlighting the significant problems that arise due to a lack of process for individual developers to measure their growth in the agile XP team. This mapping will assist us in evaluating the advantages [57] that can be accomplished if the personal software process introduced in the agile XP team to measure an individual's performance and growth. This study will also help us to find out problems and issues that arise when aligning PSP with the agile XP process.

B. Research Questions

RQ1. What problems occurred during the integration of the Personal Software Process with an agile extreme programming team? This question has discussed in table 6 and table 8.

RQ2. What challenges have encountered in the literature about XP? This question has discussed in table 10.

RQ3. What are the primary motivations for using XP?

Motivations of XP have discussed in table 9.

RQ4. What are the challenges reported in the literature of PSP? We discussed these challenges in table 12.

RQ5. What are the primary motivations for using PSP?

Motivations of PSP have discussed in table 11.

C. Search Strategy

for searching primary studies, different digital Computing databases are used. In all electronic databases, the Researcher uses search strings, keywords.

D. Keywords

Search string for searching the studies. {Personal software process*}, {PSP*}, {agile}, {individual improvement in agile}, {agile and PSP}, {collaborative software process vs personal software process}, {personal software process problems*}, {XP practicing}, {PSP and XP}

E. Search String(s)

a) Initial Search String:

((Agile AND (XP OR Extreme programming)) AND (personal quality measures OR individual quality measures OR person quality measures)) AND (PSP OR personal software process))

b) Final Search String:

((Agile OR XP OR Extreme programming) AND personal software process) OR individual OR Solo OR single) AND (PSP)

F. Primary study

After the substance getting ready is completed, the study is set up for the arrangement. Identical the arrangement record using the Save As heading, and use the naming show that is required by your gathering for the title name of papers. we select all material and import prepared element report. Use the look down window on the left 50% of the MS Word structuring toolbar.

G. Search Engine

This paper used the advanced search feature of digital libraries for Search strings. Databases that are selected for retrieving relevant articles are shown in below figure 3 and years wise are shown in figure 4.

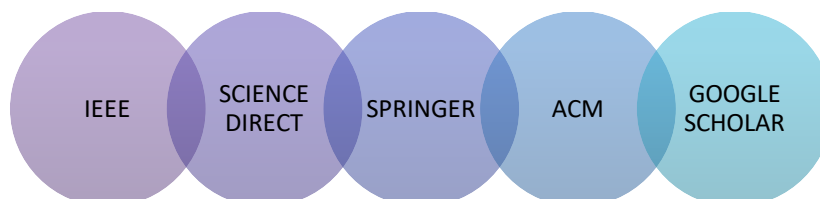


Figure. 3. Selection of research Paper Digital Libraries

H. Criteria of Inclusion

Following inclusion criteria were followed while making a decision.

- Research papers relevant to agile and personal software process (PSP) were included as primary studies
- Relevant research papers to agile XP issues were included
- Research papers relevant to personal and collaborative software process benefits were included
- Experts opinion's research papers were included
- Research papers that are relevant to individual development in agile XP are included.

I. Criteria of Exclusion

Excluded research papers.

- Books excluded
- Research Papers unrelated studies
- Non-English written paper is excluded
- Blogs, reports are excluded

J. Way of Conducting the mapping

With search string(s) in hand and conferences that were selected for search, 114 papers were selected in total that meet the inclusion criteria for primary study [Table 1]. With the further evaluation of these papers, the Researcher has comprised different types of studies that are most relevant to the issues in hand and filtered papers to 18, as mentioned in Table 1. The first study mentioned in Table 2 and the relevant study mentioned in Table 3. The reference description of primary studies is in Table 4, and the reference description of relevant studies is in Table 5. Aligned parameters extraction detail is in Table 7.

Table 1. Publication count

Databases	Publication Count		
	<i>Selected</i>	<i>Primary</i>	<i>Relevant</i>
IEEE	52	12	10
ACM	35	5	8
SD ₁	3	1	1
Springer	14	1	1
Scopus	0	0	1
GS ₂	10	3	0
Total	114	22	18

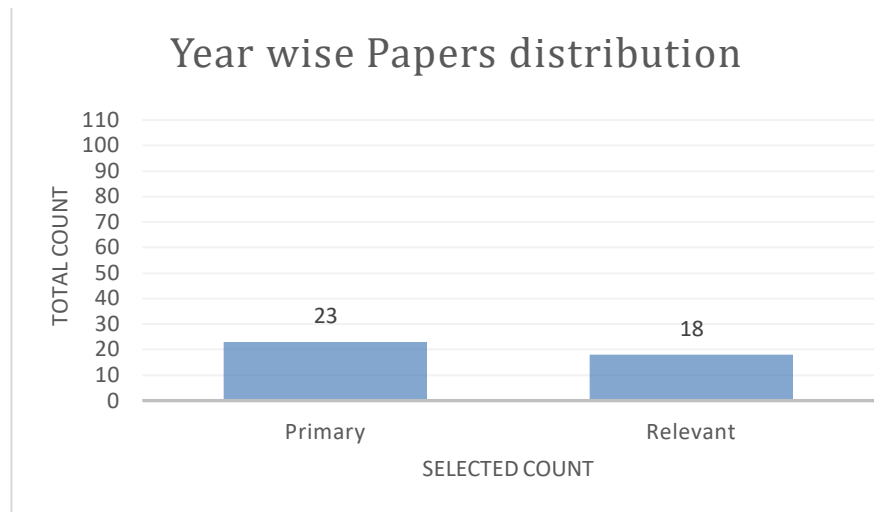


Figure. 4. Year wise paper distribution

TABLE 2. PRIMARY STUDY COUNT

Database	Primary Study Count
IEEE	12
ACM	5
Science Direct	1
Springer	1
Scopus	1
GS	3
Total	23

TABLE 3. RELEVANT STUDY COUNT

Database	Relevant Study Count
IEEE	8
ACM	6
Science Direct	1
Springer	1
Scopus	0
GS	2
Total	18

TABLE 4. PRIMARY STUDY REFERNCES

No	Reference	Primary study
1	[10]	SCRUM-PSP: Embracing Process Agility and Discipline
2	[11]	A technique for individual capability valuation in agile teams by means of individual points
3	[12]	Incorporating Lean Advance Practices into Agile Software Development
4	[13]	Faat – Freelance as a Team
5	[14]	Assessing Individual Presentation in Agile Apprentice Software Engineering Teams
6	[15]	Extreme Programming for a Single Individual Team
7	[16]	Individual Quality Management with the PSP
8	[17]	PSP PAIR: Auto PSP Presentation Investigation and Enhancement Endorsement
9	[18]	Thoughtful Self-arranging Teams in Agile Software process
10	[19]	Agile Methods and PSP are trends in software process

11	[20]	Scrum Solo
12	[21]	Light-Weight Development Method : a Case Study
13	[22]	There are different types of multiple case studies impact pair programming on the eminence of the product.
14	[23]	Comparing the project's result of waterfall and XP
15	[24]	Decomposing Agile models: Would Prearranged Design Be Supportive in XP Projects?
16	[25]	The factor of the Agile development model and their impact on implementation
17	[35]	Agile Models and Traditional/Monolithic Software Development Methodologies' comparison
18	[26]	Personal XP Agile model for autonomous developers
19	[27]	The Agile Samurai
20	[28]	Software Fault Recognition and Process Enhancement using PSP Data M.GOPICHAND
21	[29]	XP versus SCRUM comparative study of Agile models

TABLE 5. RELEVANT STUDY REFERENCES

No	Reference	Relevant study
1	[30]	Effects of PSP training experiment measuring
2	[31]	Application empirical study of PSP with different levels of training
3	[32]	Agile Software Development: The People Factor
4	[33]	Empirical Study for Improving Personal Software Process Education by Pairing
5	[34]	Towards Personalized Software Engineering: Experiential Studies Must Collect Psychometrics
6	[35]	XP development: Analysis of Linear Sequential for a Gaming Application
7	[36]	Tool Support for Personal Software Process
8	[37]	Agile methodologies and plan driver with ccomparison of Student Experiences
9	[38]	TSP overview & Practices' results
10	[39]	Improvement & Research of TSP
11	[40]	Review and Analysis of familiar Agile models approach for software development
12	[41]	A Hybrid Approach of Agile for large scale projects
13	[42]	Secure software developing challenges in literature study
14	[43]	An investigation by the empirical study on effort estimation in Agile GSD
15	[44]	Extreme Programming Explained , Second Edition
16	[45]	Service agile development using XP

17	[46]	Traditional/Monolithic vs development of agile: chaos theory comparison
----	------	---

TABLE 6. ALIGNING AND CONFLICTING PSP'S CONCEPTS WITH AGILE XP TEAM (ALIGNING CONCEPTS)

PSP Key Concepts	Aligned with XP principle
PSP Code reviews and PSP standards	Practices of Coding Standards for XP.
PSP's Software Design's Knowledge area	This training reflects the YAGNI value for efficient application of templates in your system, in accordance with XP's simple design practice.[47]
Right PSP form	This training isn't unequivocally portrayed by XP standards and practices, in spite of the fact that it is lined up with the XP ways of thinking of "on the off chance that you have to do it" and utilizing the most suitable device or procedure for the activity at hand.[48]
Tracking software quality	Reflection on XP's Testing practice with esteem to modeling [59]
PSP's process extension knowledge area	This is displaying explicit practice. XP engineers can plainly deal with a few models, for example, CRC cards, acknowledgment experiments, and outlines
Quality measure and methods	Aligned with the refactoring principle of XP. [48]
Making and tracking project plans	Aligned with the principle of XP's Planning game
User stories, scripts, and guidelines	User stories and requirements [29]
Reviews after every phase	Aligned with the Retrospectives practice of XP, in which after each iteration, the team does a short reflection on what went well during [49]

Table 7. Aligned parameters extraction from Primary Study

Years	Coding standards	Tracking software	PSP Forms	Quality management	Review lists	Software Design	User scripts and guidelines	Software measurement and estimation	Primary studies References
2000	*		*						[1], [2]
2001				*					[16]
2002		*			*				[50]
2003				*					[19]
2004					*				[49]
2005			*	*					[48]
2007				*					[21]
2008	*								[15]
2009	*		*			*	*	*	[26]

2010				*				*	[10]
2011					*		*		[23]
2013						*			[51]
2014								*	[11]
2015							*		[29]
2016		*					*		[52]

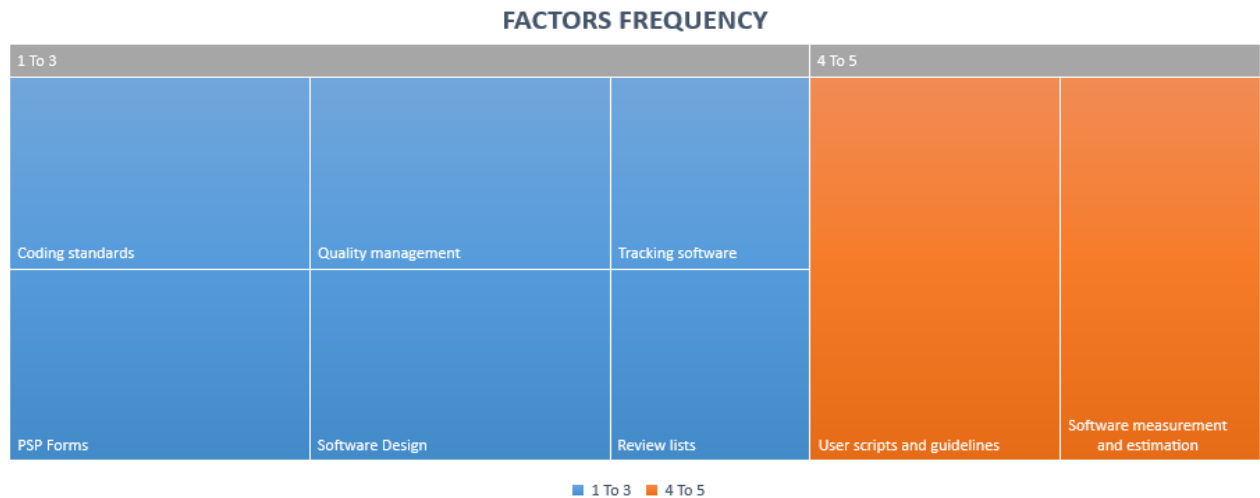


Figure 5. Factors Frequency based on papers

Analysis of Factors

In this segment, the Researcher utilized a Minitab software of statics to find the variation analysis of the elements of XP and PSP. This device makes the distinctive four charts in light of the investigation result [57]. Diagram of scree plot inform us concerning the variety amongst individually factor & demonstration by a tapped mark where the variety is happening and the amount it is. Analyst dissects the aftereffect of components by values; these qualities imply what number of times each factor is utilized as a part of writing that is considered an estimation of the factor for each paper. That is to say; figure 6 will discover the significance of factor and writing center around factor. Analyst connected the consequences for all variables in Appendix 'A' area factor variation result's table. What's more, the Appendix B segment demonstrates the result in various charts.

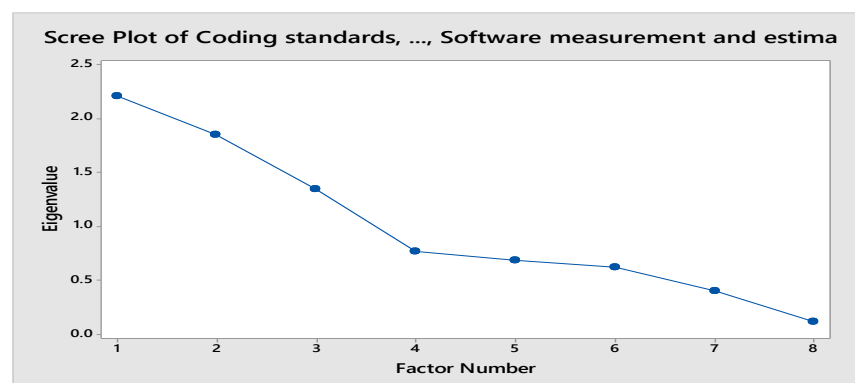


Figure 6. Factors list analysis variation graph

Table 8. Conflicting concepts between PSP and XP

PSP Key Concepts	Conflicted with XP Principles
Single person programming	Conflicted with the principle of XP Pair Programming
Existence of single person ownership	Conflicted with XP's Collective Ownership practice.
The complete finished product in a traditional style development	Conflicted with the XP's increment and iterative approach to development.[49]
Being traditional in nature	Conflicted with being iterative in nature[53]
The complete product is the outcome	Small releases[15]

Table 9. Pros and Cons of XP

Advantages	Disadvantages
<ul style="list-style-type: none"> • XP encourages pair programming. • Eminence planning assistances appropriately skilled teams to harvest high-quality products 	<ul style="list-style-type: none"> • Having two programmers on the same computer • XP programming is not structured • Extreme Programming is rigid to do

Table 10. Challenges in XP

No.	Challenges	Description	Ref.
01	Requires high investment	Organizations need to prepare the entire programming improvement group! Also that the rate of XP preparing is considerably more costly than Scrum preparing.	[56]
02	Infrastructure	To run XP, you require speculation for computerized tests and a nonstop conveyance framework.	[56]
03	Culture	You additionally need to change the association culture by uniting various divisions.	[56]
04	XP Coaches	Employing XP mentors is costly that is on the grounds that they are programming craftsman. A craftsman is costly in light of the fact that they think about quality, and they are optimists.	[56]
05	Irrational	Non-specialized individuals lead most organizations, even IT organizations. Numerous practices in XP is viewed as silly from representatives point of view.	[56]
06	Pair programming	Having two programming engineers on a comparable PC wearing down a comparable segment is extraordinary to pros. For what reason would you waste money for two designers finishing a specific something.	[56]

Table 11. PSP Pros & Cons

Advantages	Disadvantages
<ul style="list-style-type: none"> • PSP helps engineers how to make better size and resource estimates using statistical techniques and historical data [54]. 	<ul style="list-style-type: none"> • Gathering measurements is a period of the costly undertaking. This is an issue since programming ventures are regularly late, and there is no opportunity to spend in exercises that don't create prompt advantages. [54] • Manual information accumulation is a temperamental movement. An excessive number of blunders or missing information seriously influence the investigation procedure. [54]

Table 12. Challenges in PSP

No.	Challenges	Ref.
01	Problems of Data Quality in PSP	[55]
02	Metrics collection and analysis	[54]
03	Metrics data exchange	[54]

Aligned parameters extraction from Primary Study

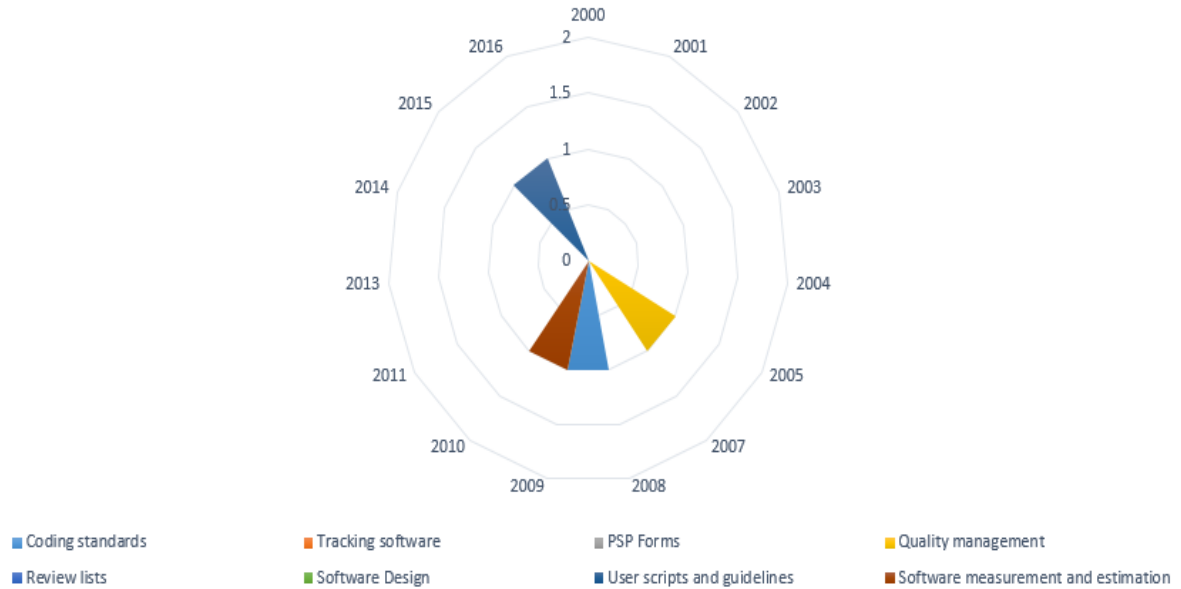


Figure 7. Parameters according to years

Aligned parameters extraction from Primary Study

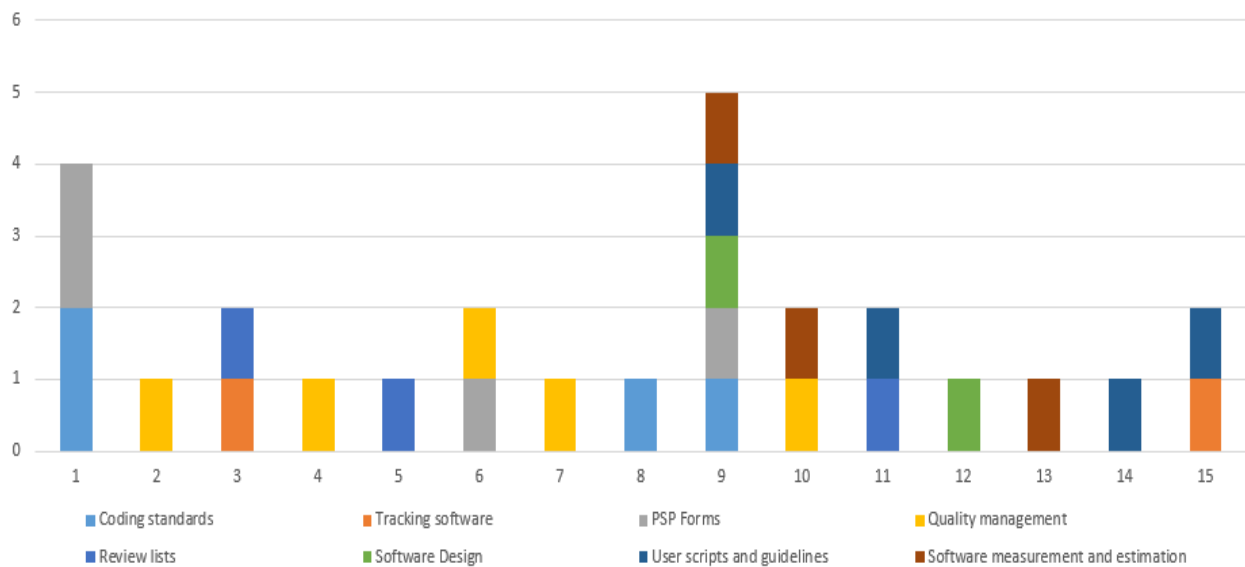


Figure 8: Parameters according to papers in years

IV. PROPOSED SOLUTION

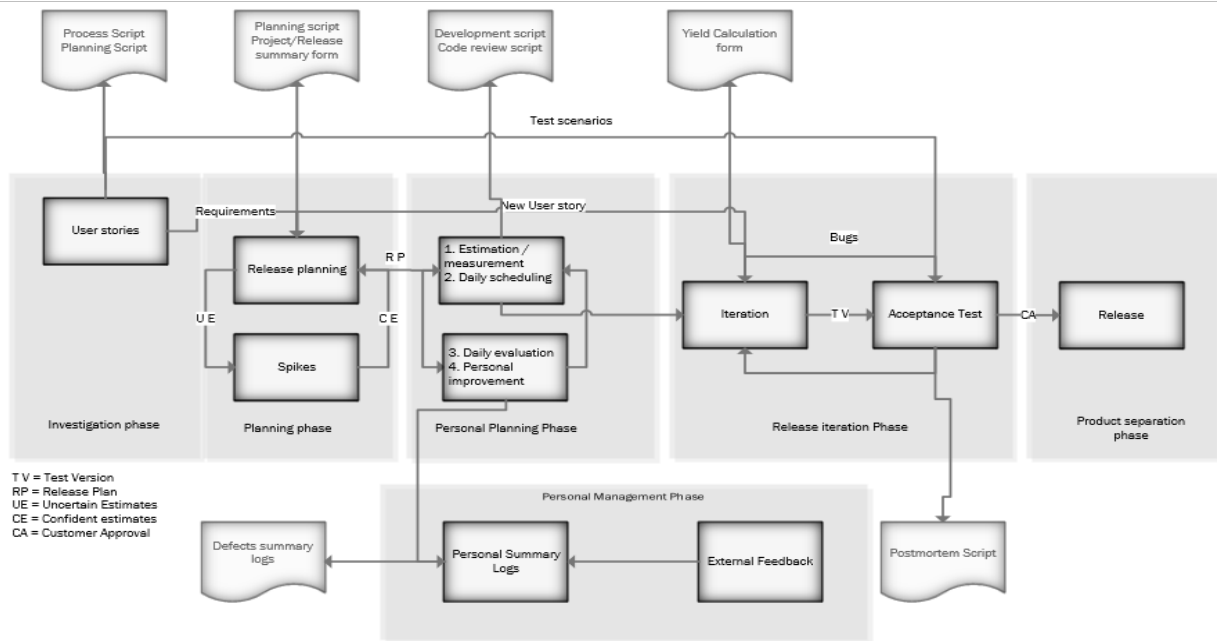


Figure 9. Proposed Model

Advantages of Combining PSP with Agile. Figure 9 It essentially finishes up six focuses on the advantages of utilizing PSP to supplement the Agile. These six advantages are recorded as pursues:

1. PSP gives the measurements and information assortment systems.
2. PSP gives help to make it an increasingly significant estimation.
3. PSP gives the advantages to upgrade plans.
4. PSP backings to deliver the product items up to quality objectives.
5. PSP gives extra documentation for coordinated techniques.
6. PSP encourages singular designers to improve their own presentation steadily.

The quality PSP rules that are utilized for singular development estimation are;

- Defect thickness
- Review rate
- Development time proportions
- Defect ratios
- Yield

V. CONCLUSIONS

This basic assessment and mapping study has inspected painstakingly the given investigations on PSP for XP and the essential engineering challenges revealed in writing. The analyst has talked about in insights about PSP with XP. Compose the arranging of mapping concentrate to deliver the outcomes of how mapping will have appeared in this investigation and the real catchphrases that help to discover the writing identified with PSP and XP. Research Questions is the real piece of this examination. These are affected by after effect of this investigation. RQs have discussed through the table for data. Every engineer has a different nature, and every engineer plans its own

work, and make plans for personal data. In this systematic mapping study first, the Researcher collects relevant publications from electronic databases. Then, apply the inclusion and exclusion criteria to extract the main and relevant study. After the final selection, the Researcher applies the methodology of systematic mapping. In this study, we explained the perceptions of alignment and conflict between the PSP and the agile XP team and discussed all the concepts from 2000 to 2017. Figure 2 shows all phases of PSP. Figure 5 shows the frequency among all factors. Minitab statics software is used to find the variation of the factors of PSP & XP which are mentioned in figure 6 and table 7, furthermore, make the outcome so as to quantitative qualities and diverse view that diagrams are and the noteworthy chart is Scree plot figure 6 of this investigation, Other charts and results are appeared in Appendix segment which are the figure 9, figure 10, and figure 11. Extracted parameters are shown in figure 7. Year wise aligning of parameters are shown in figure 8.

VI. FUTURE WORK

Assessed material depends on the best in class inquire about mined for process improvement and productivity with quality of PSP & XP. The researcher will use the proposed model to process improvement that proposes a better than ever programming advancement model for small and medium scale programming industry. The proposed model methodology endeavor between the light and generous technique to improves the product designer profitability and quality. This improved accepts the procedure in XP, which targets singular execution; in any case, it will keep the vast acts of XP with the goal that it tends to be fit for a little advancement group.

A. *Rationale*

The specialist attempts the investigation to improve/assess the custom-made XP process by coordinating the individual programming process (PSP) bolster that was at first a piece of customary programming advancement. Since the individual programming process is a challenging action, which is against the deft center standards so a lightweight rendition of PSP has proposed, and assessment will be made on this custom-made XP process. There are very few studies available that validate and measure the impact of PSP on the agile XP process and discussed in table 6.

B. *Study objective*

This investigation is to assess the effect of coordinating and aligning PSP with agile XP methodology to measure the person's presentation and growth in an agile XP team

REFERENCES

- [1] R. L. Cannon, "Putting the Personal Software ProcessSM into practice," Proc. 12th Conf. Softw. Eng. Educ. Train. (Cat. No.PR00131), pp. 34–37, 1999.
- [2] W. S. Humphrey, "The Personal Software Process (PSP) The Personal Software Process SM (PSP SM)," Engineering, no. November, 2000.
- [3] M. Pomeroy-Huff and T. A. Chick, "USE BY SOFTWARE DEVELOPMENT PROFESSIONALS The," Area, vol. 2009, no. August 2009, p. 92, 2010.
- [4] N. Iqbal, M. Hassan, and A. R. Osman, "A framework for partial implementation of PSP in Extreme programming," vol. 3, no. 2, pp. 604–607, 2013.
- [5] E. Woods, "Aligning Architecture Work with Agile Teams," IEEE Softw., pp. 24–26, 2015.
- [6] R. Juric and T. Interfaces, "WestminsterResearch Extreme programming and its development practices .," 2000.
- [7] A. Jansen and J. Bosch, "Software Architecture as a Set of Architectural Design Decisions," 5th Work. IEEE/IFIP Conf. Softw. Archit., pp. 109–120, 2005.
- [8] E. Abdullah and E. T. B. Abdelsatir, "Extreme programming applied in a large-scale distributed system," Proc. - 2013 Int. Conf. Comput. Electr. Electron. Eng. 'Research Makes a Differ. ICCEEE 2013, pp. 442–446, 2013.
- [9] W. S. Humphrey, "The personal process in software engineering," Proc. Third Int. Conf. Softw. Process. Appl. Softw. Process, no. c, pp. 69–77, 1994.
- [10] G. Rong, D. Shao, and H. Zhang, "SCRUM-PSP: Embracing process agility and discipline," Proc. - Asia-Pacific Softw. Eng. Conf. APSEC, pp. 316–325, 2010.
- [11] S. Celar, M. Turic, and L. Vickovic, "Method for personal capability assessment in agile teams using personal points," Telecommun. Forum Telfor (TELFOR), 2014 22nd, pp. 1134–1137, 2014.
- [12] J. Sutherland and J. Tabaka, "Incorporating lean development practices into agile software development," Proc. Annu. Hawaii Int. Conf. Syst. Sci., p. 7695, 2007.
- [13] R. B. Bernabé, I. Á. Navia, and F. J. García-Peñalvo, "Faas," Proc. 3rd Int. Conf. Technol. Ecosyst. Enhancing Multicult. - TEEM '15, pp. 687–694, 2015.
- [14] R. F. Gamble and M. L. Hale, "Assessing individual performance in Agile undergraduate software engineering teams," Proc. - Front. Educ. Conf. FIE, pp. 1678–1684, 2013.
- [15] R. Agarwal and D. Umphress, "Extreme programming for a single person team," Proc. 46th Annu. Southeast Reg. Conf. XX - ACM-SE 46, no. March, p. 82, 2008.
- [16] A. S. Koch, "Personal quality management with the personal software process," Softw. Eng. Educ. Conf. Proc., p. 43, 2001.
- [17] C. B. Duarte, J. P. Faria, and M. Raza, "PSP PAIR: Automated personal software process performance analysis and improvement recommendation," Proc. - 2012 8th Int. Conf. Qual. Inf. Commun. Technol. QUATIC 2012, pp. 131–136, 2012.
- [18] N. B. Moe, T. Dingsøyr, and T. Dybå, "Understanding self-organizing teams in agile software development," Proc. Aust. Softw. Eng. Conf. ASWEC, no. 3, pp. 76–85, 2008.
- [19] J. A. Stark and R. Crocker, "Trends in software process: The PSP and agile methods," IEEE Softw., vol. 20, no. 3, pp. 89–91, 2003.
- [20] J. A. Fabri and A. L. Erario, "Scrum Solo."
- [21] J. Q. Chen, D. Phan, B. Wang, and D. R. Vogel, "Light-Weight Development Method : a Case Study," 2007.
- [22] H. Hulkko, "S8a ~ SUR |."
- [23] T. Sedano, "Comparing Extreme Programming and Waterfall Project Results," pp. 482–486, 2011.
- [24] R. Noël, G. Valdes, M. Visconti, and H. Astudillo, "Deconstructing Agile Processes : Would Planned Design Be Helpful in XP Projects ?," pp. 42–51, 2008.
- [25] J. A. Livermore, "Factors that Impact Implementing an Agile Software Development Methodology," pp. 82–86, 2007.
- [26] Y. Dzhurov, I. Krasteva, and S. Ilieva, "Personal Extreme Programming—An Agile Process for Autonomous Developers," Int. Conf. SOFTWARE, Serv. Semant. Technol., no. May 2016, p. 252, 2009.
- [27] F. Breu, S. Guggenbichler, and J. Wollmann, The Agile Samurai. 2008.
- [28] M. Gopichand, V. Swetha, and a. Ananda Rao, "Software defect detection and process improvement using personal software process data," 2010 Int. Conf. Commun. Control Comput. Technol., pp. 794–799, 2010.
- [29] Mohammad Almseidin, Khaled Alrfou, Nidal Alnidami, and Ahmed Tarawneh, "A Comparative Study of Agile Methods: XP versus SCRUM," Int. J. Comput. Sci. Softw. Eng., vol. 4, no. 5, pp. 126–129, 2015.
- [30] L. Prechelt and B. Unger, "An experiment measuring the effects of Personal Software Process (PSP) training," IEEE Trans. Softw. Eng., vol. 27, no. 5, pp. 465–472, 2001.
- [31] J. F. Niño Manrique and R. Anaya, "An empirical study of application of PSP methodology with students of a Systems Technology program with different levels of training," 38th Lat. Am. Conf. Informatics, CLEI 2012 - Conf. Proc., 2012.
- [32] A. Cockburn, "Development .," no. November, pp. 131–133, 2001.
- [33] G. Rong, H. Zhang, M. Xie, and D. Shao, "Improving PSP education by pairing: An empirical study," Proc. - Int. Conf. Softw. Eng., pp. 1245–1254, 2012.
- [34] R. Feldt, L. Angelis, and M. Samuelsson, "Towards individualized software engineering: empirical studies should collect psychometrics," Proc. 2008 Int. Work. Coop. Hum. Asp. Softw. Eng., pp. 49–52, 2008.

- [35] P. Sharma and N. Hasteer, "Analysis of Linear Sequential and Extreme Programming Development Methodology for a Gaming Application," pp. 1916–1920, 2016.
- [36] J. Lappalainen, "LNCS 3547 - Tool Support for Personal Software Process," pp. 545–559, 2005.
- [37] C. A. Wellington, T. Briggs, and C. D. Girard, "Comparison of Student Experiences with Plan-Driven and Agile Methodologies," pp. 18–23, 2005.
- [38] T. Report, "Process (TSP): An Overview and Preliminary Results Practices," vol. 2000, no. October, 2001.
- [39] P. Ri, R. Q. J. X. Q. X, L. Dqj, H. S. W. Ri, V. Qjlqhhulqj, R. I. Qjlqhhulqj, P. K. B. X. Frp, D. Vfkphpv, F. Dqg, H. Wr, L. Wkh, F. Phwkrq, F. Wkh, L. Ri, D. Q. G. Sudfwlfh, and V. Ri, "Shvhdufk dqg ,psuryhphqw ri 7hdp 6riwzduh 3urfhv," 2009.
- [40] A. Mahdi, M. Hamed, and H. Abushama, "Popular Agile Approaches in Software Development : Review and Analysis," pp. 160–166, 2013.
- [41] M. Tanveer, "Agile For Large Scale Projects – A Hybrid Approach," no. Nsec, pp. 14–18, 2015.
- [42] H. . Oueslati, M. M. . Rahman, and L. B. . Othmane, "Literature review of the challenges of developing secure software using the agile approach," Proc. - 10th Int. Conf. Availability, Reliab. Secur. ARES 2015, pp. 540–547, 2015.
- [43] R. Britto, E. Mendes, and J. Borstler, "An Empirical Investigation on Effort Estimation in Agile Global Software Development," Proc. - 2015 IEEE 10th Int. Conf. Glob. Softw. Eng. ICGSE 2015, pp. 38–45, 2015.
- [44] K. Beck, *Extreme Programming Explained*, Second Edition. .
- [45] F. Carvalho and L. G. Azevedo, "Service agile development using XP," Proc. - 2013 IEEE 7th Int. Symp. Serv. Syst. Eng. SOSE 2013, pp. 254–259, 2013.
- [46] D. M. Shawky, "Traditional vs agile development: A comparison using chaos theory," ICSOFT-PT 2014 - Proc. 9th Int. Conf. Softw. Paradig. Trends, pp. 109–114, 2014.
- [47] C. Lan, K. Mohan, X. Peng, and B. Ramesh, "How extreme does extreme programming have to be? adapting XP practices to large-scale projects," 37th Annu. Hawaii Int. Conf. Syst. Sci. 2004. Proc., vol. 0, no. C, pp. 83–92, 2004.
- [48] M. A. Awad, "A Comparison between Agile and Traditional Software Development Methodologies," Univ. West. Aust., vol. 1, pp. 1–300, 2005.
- [49] L. Lindstrom and R. Jeffries, "Extreme programming and agile software development methodologies," Inf. Syst. Manag., vol. 21, pp. 41–52, 2004.
- [50] S. Ambler, *Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process*. 2002.
- [51] Y. Chaiyo and S. Ramingwong, "The development of a design tool for personal software process (PSP)," 2013 10th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2013, 2013.
- [52] C. A. D. L. Salge and N. Berente, "Pair programming vs. solo programming: What do we know after 15 years of research?," Proc. Annu. Hawaii Int. Conf. Syst. Sci., vol. 2016–March, pp. 5398–5406, 2016.
- [53] B. Boehm and R. Turner, "Management challenges to implementing agile processes in traditional development organizations," IEEE Softw., vol. 22, no. 5, pp. 30–39, 2005.
- [54] Sillitti, Janes, "Collecting, integrating and analyzing software metrics and personal software process data," IEEE, 2003.
- [55] Sillitti, Janes, "Collecting, integrating and analyzing software metrics and personal software process data," *Data Quality Problems in the Personal Software Process*, researchgate, Oct, 2017.
- [56] "XP CHALLENGES," *WEBLINK*, OCT, 2017
- [57] Shahbaz Ahmed, Abdul Razzaq, S. Ullah, S. Ahmed, Matrix Clustering Based Migration of System Application to Microservices Architecture, *ijacsa*, Jan, 2018.
- [58] Shahbaz Ahmed Khan Gahyyur, Abdul Razzaq, Syed Zeeshan Hasan, Salman Ahmed and Rafi Ullah, "Evaluation for Feature Driven Development Paradigm in Context of Architecture Design Augmentation and Perspective Implications" *International Journal of Advanced Computer Science and Applications(Ijacsa)*, 9(3), 2018.
- [59] Shaik Mohammad Shahabuddin, Dr. Prasanth Yalla, "Impact of lean software development into agile process model with integration testing prior to unit testing" *Journal of Theoretical and Applied Information Technology, JATIT*, 2017.

Result report of Factor analysis

Section A: Appendix

Analysis of Factors: Coding stand, Tracking software, PSP Forms, Quality mana, Review lists, Software measurement & estimation

Factor Analysis Principal Component of the Correlation Matrix

Factor Communalities and Loadings Un-rotated

Variable	Factor1	Factor2	Factor3	Factor4	Factor5	Factor6
Coding standards	0.816	-0.229	-0.449	0.001	-0.093	-0.019
Tracking software	-0.466	-0.509	-0.147	-0.452	-0.314	0.436
PSP Forms	0.800	-0.096	-0.446	-0.094	-0.073	-0.030
Quality management	-0.126	0.884	0.012	-0.219	0.039	-0.018
Review lists	-0.542	-0.470	-0.244	0.541	-0.137	-0.142
Software Design	0.483	-0.284	0.553	0.219	0.324	0.440
User scripts and guidelines	0.014	-0.664	0.361	-0.400	0.222	-0.448
Software measurement and estima	0.383	0.076	0.656	0.077	-0.631	-0.112
Variance	2.2133	1.8510	1.3474	0.7668	0.6857	0.6188
% Var	0.277	0.231	0.168	0.096	0.086	0.077
Variable	Factor7	Factor8	Communality			
Coding standards	0.132	0.233	1.000			
Tracking software	-0.091	0.031	1.000			
PSP Forms	-0.322	-0.180	1.000			
Quality management	-0.363	0.143	1.000			
Review lists	-0.298	0.068	1.000			
Software Design	-0.178	0.044	1.000			
User scripts and guidelines	-0.130	0.047	1.000			
Software measurement and estima	-0.032	0.004	1.000			
Variance	0.3998	0.1173	8.0000			
% Var	0.050	0.015	1.000			

Factor Score Coefficients

Variable	Factor1	Factor2	Factor3	Factor4	Factor5	Factor6
Coding standards	0.369	-0.123	-0.333	0.002	-0.136	-0.031
Tracking software	-0.211	-0.275	-0.109	-0.589	-0.458	0.704
PSP Forms	0.362	-0.052	-0.331	-0.123	-0.106	-0.048
Quality management	-0.057	0.478	0.009	-0.285	0.058	-0.030
Review lists	-0.245	-0.254	-0.181	0.706	-0.200	-0.229
Software Design	0.218	-0.153	0.410	0.285	0.473	0.711
User scripts and guidelines	0.006	-0.359	0.268	-0.521	0.323	-0.724
Software measurement and estima	0.173	0.041	0.487	0.100	-0.920	-0.182
Variable	Factor7	Factor8				
Coding standards	0.331	1.990				
Tracking software	-0.227	0.264				
PSP Forms	-0.806	-1.539				
Quality management	-0.909	1.222				
Review lists	-0.746	0.579				
Software Design	-0.444	0.372				
User scripts and guidelines	-0.324	0.398				
Software measurement and estima	-0.080	0.035				

Appendix B

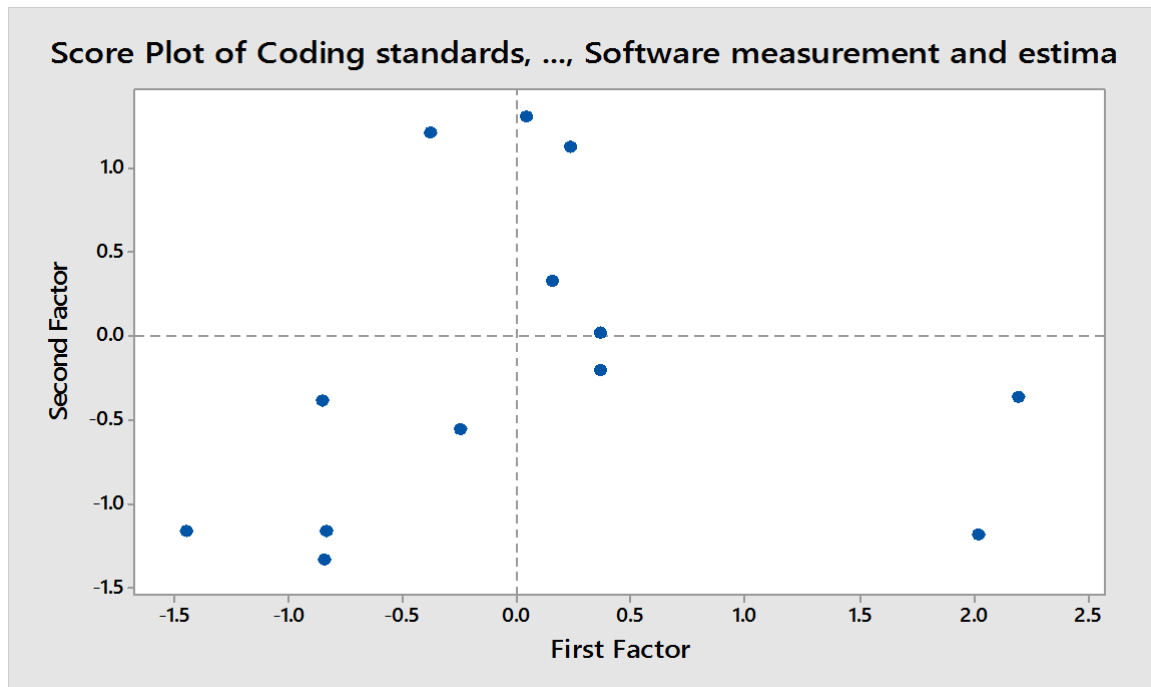


Figure 9. Score Plot

This displays a connection among factors in the type of gatherings. It advises the distinctive connections keeping in mind the end goal to positive and negative. This chart enables us to find comparative perceptions quickly.

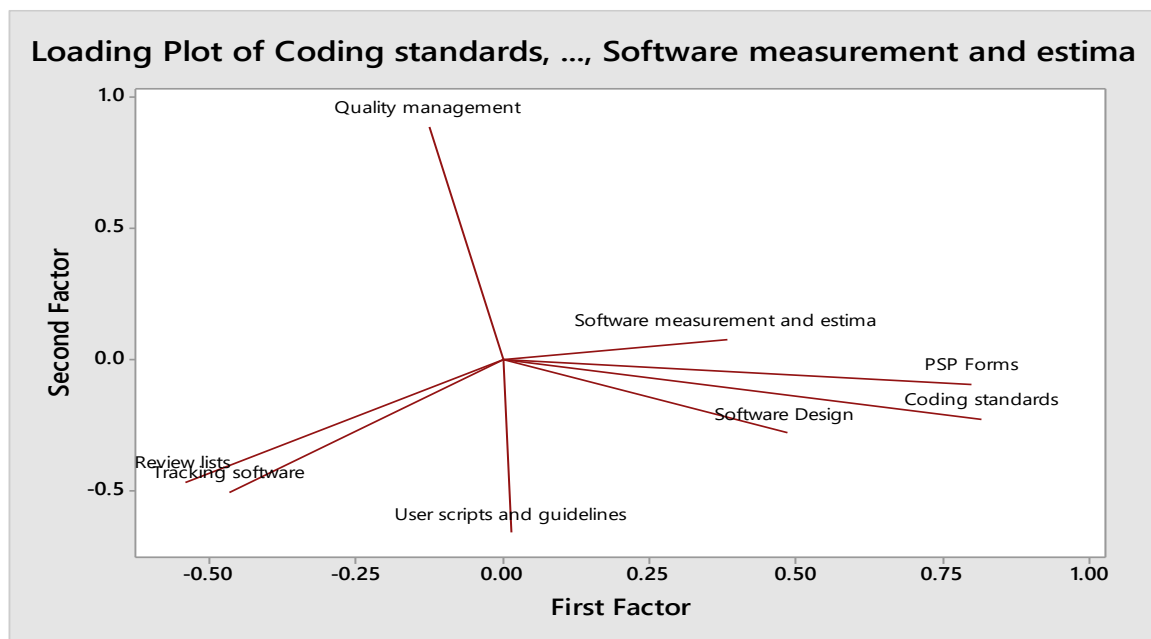


Figure 10. Loading Plot This diagram reveals to us co-connections in two routes among factors evenly and vertically. It discloses to us the connection between two segments.

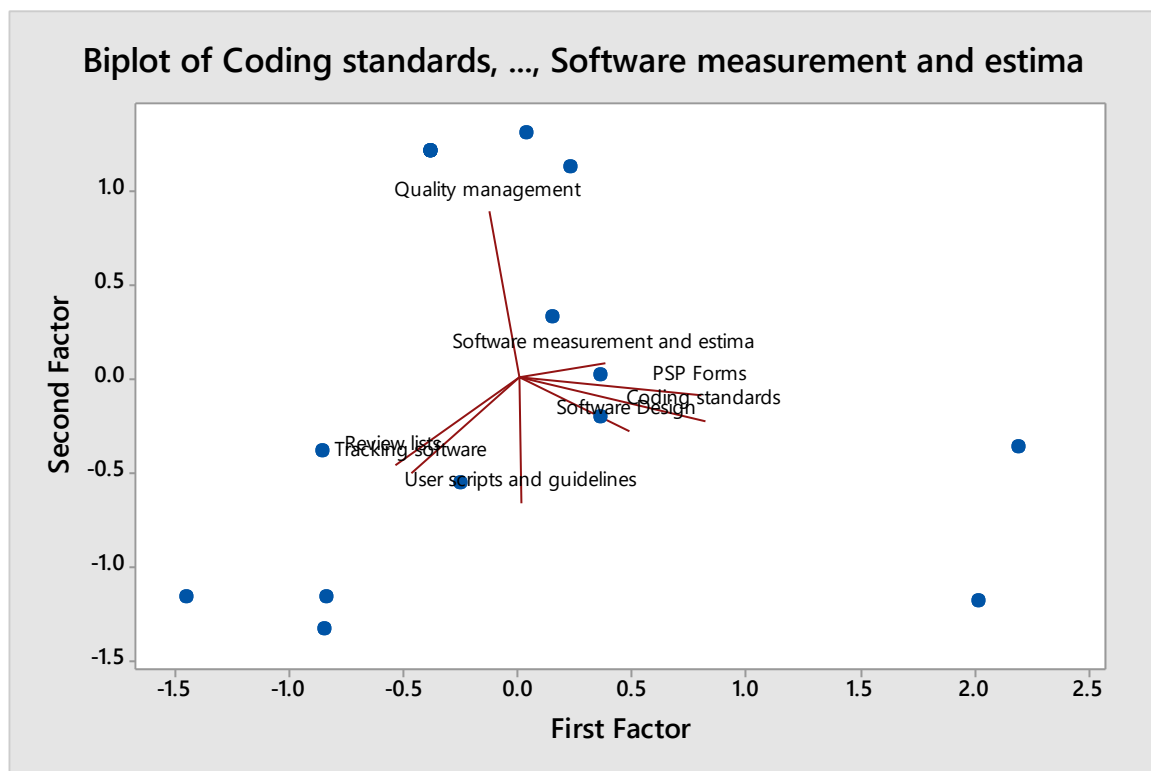


Figure 11. Biplot

This chart indicates us connections among factors as sets.

Acknowledgement

Conflict of Interest

All authors have no conflict of interest relevant to this article.