

Student's Opinion on Adopting Pair Programming as a Teaching and Learning Tool

Radhakrishnan. P

Department of Computer Science & Engineering
Pondicherry Engineering College
Pondicherry, India

Kanmani. S

Department of Information Technology
Pondicherry Engineering College
Pondicherry, India

ABSTRACT

Research in the realm of Pair Programming in computer science students has begun to explore the methods for accomplishing the benefits. However, valid measurements of the opinion on adopting a pair programming in the practical sessions are required to be drawn, before concluding. An Attitude Survey Test (AST) was conducted before implementing Pair Programming as a teaching and learning tool among computer science students in Pondicherry Engineering College. The purpose of this investigation is to examine the student's opinion on four vital issues viz. i) General Perception about pair programming; ii) Personality conflicts among pair; iii) Relevant examination system and iv) Female student's attitude. We have engaged 154 B. Tech and MCA students as participants from Pondicherry Engineering College, for the AST, each answered a set of questions relating to Pair Programming. By accurately assessing the students' opinions, our AST results indicated the adequate convergent validity of scores obtained, that there is a room for implementing pair programming as a teaching and learning tool in laboratory course works.

Keywords

Extreme Programming (XP), Pair Programming, Attitude Survey Test, Teaching and Learning

1. INTRODUCTION

Teaching and learning of software programs in graduate level engineering students involves designing, writing, testing, and implementing in the computer laboratories. Learning computer programming is a taxing task for the student's community [8]. The same is the case for software developers in industries. To overcome these difficulties, different software development methodologies have been proposed. Agile software development methodology [2], [36], is unique from traditional software development methodology like a waterfall and spiral models. Some of the contemporary agile methods are Crystal Methodologies; Dynamic Software Development Method (DSDM); Feature-driven development; Lean Software Development; Scrum and Extreme Programming (XP). Extreme Programming is one of the most popular agile software development methodologies [9]. Several reports explain the benefits achieved by adopting XP in a software development organization. The evidence of success of XP in the industry, created an interest in the community of computer science educators to apply XP practices in educational settings for teaching software programs. XP recommends 12 practices for software

Development. Even though all the practices of XP are equally important, the educational researchers were attracted more towards pair programming than the other practices of XP [5]. Pair programming involves two programmers working collaboratively on one computer, one as a driver who operates the keyboard, concentrates on the lower level details of the task and another as a navigator who observes the driver, offering suggestions and corrections on higher level details of the task [2], [4]. In contrary, solo programming is the traditional method, developing software program individually. The programmer decides himself how it's going to be. The solo programmer owns the code, with all the good and bad associated with that program. If the task is straightforward, solo programming can be more efficient. Several previous controlled experiments in programming industries claimed that pair programming is useful and beneficial in numerous facets [26]. Even though, few argue that, pair programming is not as productive as solo programming, many claimed that, pair programming contributed several benefits, especially in educational environments [12]. Some of the widely gained benefits found in the literature are as follows.

- Program correctness: [14], [24].
- Higher software quality: [7], [13], [39].
- Reduced time for program development: [1], [20], [33].
- Increased learning efficiency: [3], [6], [19], [34], [39].
- Increased confidence level: [12], [25], [31].
- Course completion rate: [12], [17], [23].
- Improved personality [27], [28], [29], [35]

Nearly, all of the research work was carried out in the domain of education and have used the students as subjects. The experiments were done using the laboratory course work. Most probably, the students were asked to take part in the experiments without obtaining their concern. If the students are trained using pair programming and asked them to take the final exam individually, then that may cause a concrete issue of self confidence. Furthermore, adopting a pair programming might summon few controversial problems such as pairing and personality conflicts, facing final examination individually. Hence, it is necessary to obtain the attitude of students towards all the above issues, before adopting Pair Programming in teaching the programming exercises in laboratories.

The common problems in adopting pair programming and the possible solutions of the conducted research are discussed in section 2. Section 3 discusses the methodology adopted in this study. Section 4 describes the results and discussion and section 5 concludes this paper.

2. COMMON PROBLEMS IN ADOPTING PAIR PROGRAMMING

As stated in the section 1, before implementing, it is necessary to explore the following issue to investigate the student's attitude about pair programming and its importance incorporating the anticipated common problems faced by the students while adopting a pair programming in the laboratories.

2.1 General Perception

After the introduction of XP, which gained more popularity, several experimental and case studies have been conducted in the educational set ups. These experiments revealed several benefits of pair programming. In spite of the benefits achieved by the pair programming, few researchers have asserted, pair programming as failure and so was not worth undertaking. But this assumption was refuted by Norman Jacobson and Suzanne K. Schaefer [26]. Solo programming is the traditional method being followed in the laboratories for practical courses. In solo programming, students develop and solve the programming assignments individually. The solo programmer owns the code, with all the good and bad, associated with that program. If the task is straightforward, solo programming is more efficient. Since the pair programming is a new methodology, introduction of this method among the students for learning might radically change the way, in which students do the laboratory exercises [38]. But, before introducing, like any other newly inducted methodology, pair programming is also to be assessed by way of obtaining opinions from the students.

2.2 Personality Conflicts

Pair programming methodology permits the students to discuss each other for problem solving. This activity needs the students have to interact closely with each other [18]. In such circumstance, pairing the students plays a major role for the effective learning [4], [10]. The student's choice of pair partner depends on many social factors such as the personality [16], [32], academic achievement [15], gender, and mother tongue of the other partner. The study conducted by Nagappan et al. [25] at North Carolina State University shows that equal or higher skill level students finish the course with better grading. The student's skill level is the most accurate predictor of the partner compatibility [18] in PP. Out of these factors, we believe that gender and academic achievement level of the other partner are significant factors [11]. As per Theodore, V.T. et al [34], the equal personality profile of introverts proved better in programming. But in some occasion, variance in personality factor induces conflicts among pairs of students [37]. Therefore it has become necessary to obtain the students view and their options for selecting their partners for effective learning.

2.3 Attitude of Female Student

Primarily, pair programming is a joint venture of two programmers sitting side by side, using one computer for programming [40]. During this period, mutual understanding and combined efforts between the pairs are essential. Hence, forming of pairs favors compatibility and cultural aspects. According to Ken Beck [2], pair programming is not only an academic issue but is also about social and cultural issues. It is important that men and women both simultaneously acknowledge and admire the quality of pair programming and

apply it to effective learning. In this context, the study should be carried out to analyze the perpetual posture, and to determine the differences of opinion (if so) among men and women regarding pair programming

2.4 Relevant Examination System

Lui and K.C.C.Chan [22] proposed a model of Software Process Fusion (SPF) and suggested that, programmers may design solution patterns in pair and then use the patterns to build sub models on solo. When pair programming is practiced in regular laboratory classes, it is important to find whether the students are ready to take-up the final semester practical examinations without pair partner. But, after practicing with pair during the entire semester, the students may opt for examining them in pair as done in the project works. The traditional system of conducting the final exam is only by individuals. Hence, it is necessary to frame some other suitable examination methods to evaluate the students.

3. METHODOLOGY

3.1 Pilot Study

A pilot, or feasibility study, is a small experiment designed to test and gather information prior to the target study, in order to improve the latter's quality and efficiency. To know the attitude of the students, we asked few volunteer students to take-up pair programming in short duration programming assignments and the opinions were obtained about the pair programming through questionnaires. The assignments for short duration were framed considering program requirements of a lab session. The participants of the pilot study exchanged their experience to their classmates. In addition, prior to the main study, a clear explanation about pair programming and the aim of this study was given to all the students.

3.2 Survey Method

The survey method is the technique of gathering data by asking questions to people who provide desired information. A formal list of questionnaire is prepared. By using the information or response provided by the respondent, users can analyze it using standard statistical techniques.

Subjects: The AST was conducted using the students of Pondicherry Engineering College, who had no experience of pair programming in the laboratories. Hence, we decided to provide an exposure to the students on pair programming before obtaining the opinion from them. We have selected 154 students from three departments viz. Department of computer science and engineering, Department of Information Technology and Master of Computer Applications. They are 66 students each from B. Tech computer science and Information Technology branch respectively and 22 students from Master of Computer Application (M.C.A) course.

Assignments: The selected students were asked to adopt pair programming to do two laboratory exercises, in order to get well versed with different partners. Each of those exercises was about three-hour duration. The students in each pair were assigned randomly except that no pair was repeated with the same members for doing the second exercises. The exercise for each of the laboratory classes was announced only at the beginning of the laboratory session. The students working in the Data Structures lab, were asked to develop programs for the following exercises.

Table 1. Questionnaire

Sl.No	Questions	Scale 1	Scale 2	Scale 3	Scale 4
1.	Do you agree that PP is an effective methodology, and can be adopted in the programming laboratory?	Strongly Agree	Agree	Disagree	Strongly disagree
2.	Which academic level of partner, you would like to choose to learn more through PP?	Better level	Equal level	Less level	Any level
3.	Which gender of partner you would like to prefer for PP?	Same gender	Other gender	Mixed gender	Any gender
4.	After practicing in regular lab with PP methodology, are you willing to take up the final examinations individually?	Only individual mode	Sometimes individual mode	Only partner mode	Any mode

- a). Implement stack and use it to convert infix to postfix expression
- b). Implement a double-ended queue where insertion and deletion operations are possible at both the ends

At the end of the second exercise, questionnaires were distributed to all and were asked to fill the questionnaire individually and not pairwise. The questions listed in Table 2 were framed to collect the opinion of the students on pair programming

Questionnaire: A questionnaire is a list of written query that can be floated among the subjects who are taking part in the experiments. There are two types of questionnaires used in the research work. Close-ended questionnaires are restricted to be answered using yes/no; true/false; or choosing from a list of options provided by the experimenter which yields quantitative data. In contrast, open-ended questionnaire is framed for the respondent to answer in their own words and the resultant answer gives qualitative data. The questionnaire remains a unique instrument with comprehensive multidimensional scales for testing the hypothesis related to the concerned research. In this experiment, four questions listed in table 4 were developed for gathering the answers from the participating students. In addition to answering, students were also asked to write their gender to find “whether any difference of opinion in adopting pair programming based on gender”.

4. Results and Discussion

The data collected were analyzed to find out the opinions of the students.

i) General Perception:

Pair programming creates an environment conducive to more advanced, active learning and social interaction leading to students being less frustrated [41]. When two students are working together, each has their own set of knowledge and skills. A large subset of this knowledge and skills will be common between the two. Allowing them to interact effectively in academic institutions, it was found that the time taken for one student to complete a program would be nearly cut in half and hence allowing the teacher to assign more programs. That would help the students’ knowledge by

allowing them to complete more programs. In case project work, pair pressure makes them to submit it on time. Such kind of pressure does not exist if the student does the assignments individually. In this study, on post pair programming session, our interaction with the participants reveals that the students were comfortable and gained more confidence in solving the assignments; in terms of design and coding.

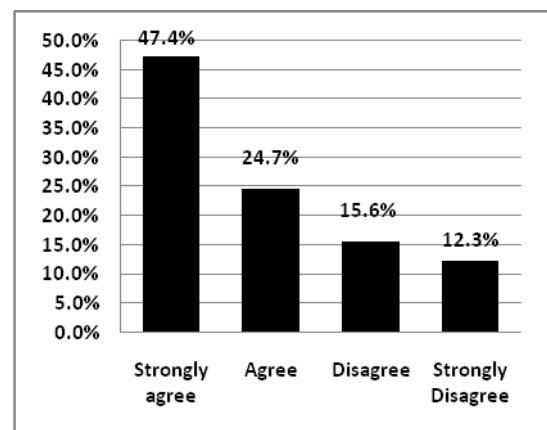


Fig 1: Willingness of the students to adopt pair programming

They also expressed that the pair programming process helped the pair in clearing bugs fast and efficiently. Many students felt that the partner was a kind of immediate resource for learning. In the context of adopting pair programming in the laboratory classes, the question was asked with four options.

Table 2. Willingness for adopting pair programming

Answer	Frequency	Percent
Strongly agree	73	47.4
Agree	38	24.7
Disagree	24	15.6
Strongly disagree	19	12.3
Total	154	100.0

The Fig 1 shows that 47.4% (strongly agree) and 24.7% (agree) students have expressed their willingness to adopt pp in the regular lab course. The table 2 shows that almost 72.1% of the students have expressed their willingness to adopt pp in the regular lab course. This indicates that the majority of the students liked the Pair Programming to be adopted in the regular lab as a learning method

ii) Personality Conflicts

Pair programming accelerates the programming process [39], provides an opportunity for lowering students' dependence on teaching staff [40]. This enables the weaker students to learn more by talking to the stronger. But in this experiment it was noted that the students were not in favor of explaining things to the partner and improving their understanding of the topics. Few students had a negative experience due to the sluggish learning partner.

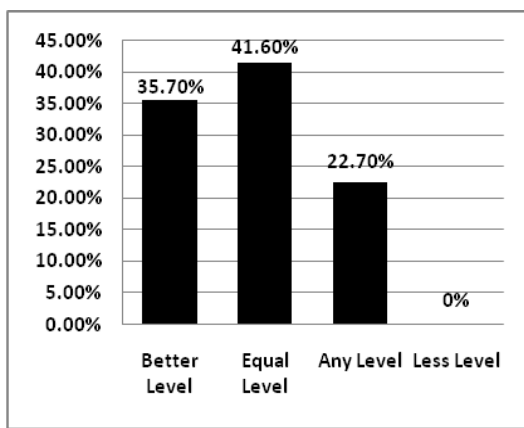


Fig 2. Preferred personality level for partner

The differences in the personality level lead the partners with different approaches to solve the problem. Nevertheless the different approach is a welcoming measure to solve a problem, but sometimes the idea given by the weak student did not influence the higher level partner. In this context, the

Table 3. Preferred personality level for partner

Answer	Frequency	Percent
Better level	55	35.70
Equal level	64	41.60
Any level	35	22.70
Less level	0	0
Total	154	100.0

answer given by the participants (better level 35.7%, equal level 41.6%, any level 22.7% and less level 0% as in Fig 2 and Table 3 shows that the majority of the students are interested in working with better level or equal level partners and none are interested in working with a low level partner. This shows that the majority of the students are preferred to do pair programming with better or equal personality type of students.

iii) Female Students' Attitude for Gender Preference

Although boys and girls differ in their physical, emotional and intellectual development, social and cultural factors are the major reasons leading to gender differences in academic performance. These factors include students' familiarity with the subject, changes of career aspiration, gender perceptions of specific subject, presentation styles and teacher's expectation. Girls generally prefer cooperation, open-ended and organized activities, while boys prefer competition and individualism. Normally girls give more favorable reviews of their learning experiences than boys do. Presently, the gender differences in the developed countries have diminished considerably, especially among students. In many conservative countries, the social and cultural background forces the students to remain confined in orthodox territory.

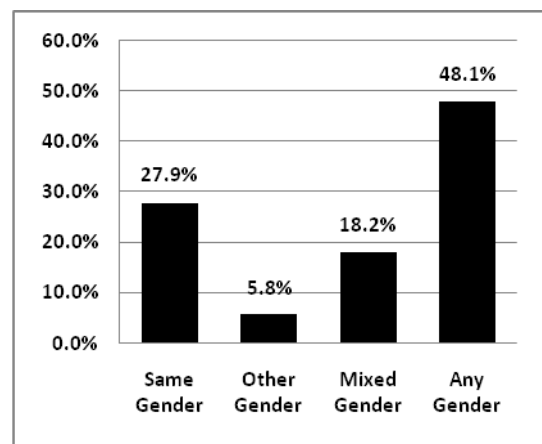


Fig 3. Gender preference

With the result, the students are comfortable when they are working with the same gender. This view is reflected in the analysis of the answer shown in the Table 4. and Fig 3. Although 27.9% of the students preferred to work with same gender, 41.8% students expressed that they can work with any gender. The difference between these two opinions was significantly more. In contrary only 5.8% of students have opted for other gender.

Table 4. Preferred Gender for the partner

Answer	Frequency	Percent
Same Gender	43	27.9
Other Gender	9	5.8
Mixed Gender	28	18.2
Any Gender	74	48.1
Total	154	100.0

This perception is based on the view by both male and female students. But to know the view of female students exclusively, a cross table analysis between gender and preference for the partner was done.

Gender of the students	Gender preference for the partner				Total
	Same Gender	Other Gender	Mixed Gender	Any Gender	
Male	20	6	16	43	85
	46.5%	66.7%	57.1%	58.1%	55.2%
Female	23	3	12	31	69
	53.5%	33.3%	42.9%	41.9%	44.8%
Total	43	9	28	74	154
	100.0%	100.0%	100.0%	100.0%	100.0%

Table 5. Cross tabulation for female student’s gender preference for a partner

Table 5. Cross tabulation for female student’s academic level preference for a partner

Gender of the students	Preferred academic level of the partner			Total
	Better level	Equal level	Any level	
Male	24	38	23	85
	43.6%	59.4%	65.7%	55.2%
Female	31	26	12	69
	56.4%	40.6%	34.3%	44.8%
Total	55	64	35	154
	100.0%	100.0%	100.0%	100.0%

. For this test the following two hypotheses were set.

- Female students prefer to work only with the same gender
- Female students prefer to work with better level students.

There is no much difference between the male and female students in the gender preference in pair programming, the chi-square test also reveals the same (P value = .555). However, the cross table reveals that 53.5% of the female students prefers to work with the same gender only, in the case of male students it was 46.5%. Moreover, it is normal course that male students prefer to work with any gender (58.1%); mixed gender (57.1%) and other gender (66.7%) when compare to female students (41.9%).

Table 6. Chi-Square Tests for hypothesis 1

	Value	df	Asymp. Sig. (2-sided)
Pearson Chi-Square	2.087	3	.555
Likelihood Ratio	2.091	3	.554
N of Valid Cases	154		

Table 7 Chi-Square Tests for hypothesis 2

	Value	df	Asymp. Sig. (2-sided)
Pearson Chi-Square	4.990	2	.083
Likelihood Ratio	5.008	2	.082
N of Valid Cases	154		

Female students more or less prefers to work with better level students (56.4%) when compare to the male students (43.6%). The chi-square test also proves that there is a relationship between the gender and preference of the partner’s academic level. It reflects that, the female students feel convenient in working with students who has better knowledge.

iv. Relevant Examination System

Usually in final examinations, the students are examined individually. This practice is followed both in theory and practical examinations. But, if the pair programming is applied in the regular laboratory course work, the students might be working in pairs during entire session. Consequential effect of this practice might reduce the student’s confidence level of facing examination individually. But the analysis of the answer given by the students, as shown in table 8 and fig 4 reveals that 68.2% of the students are ready to take the final examination individually. In spite of the fact that students are occupying only half of the time at the

computer terminals out of the total allocation time of three hours, the students feel that

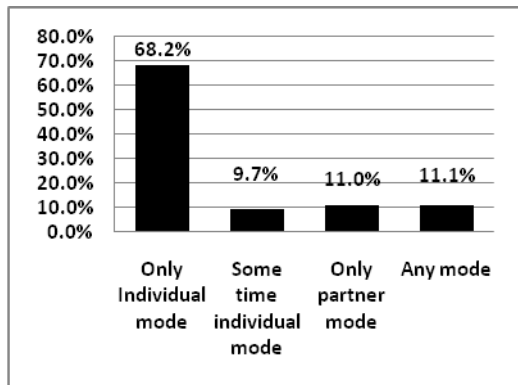


Fig. 4. Willingness to take the examination

they can appear for the examination individually. This indicates that the pair programming increases the knowledge and confidence level of individual persons to face the examination.

Table 8. Preferred examination mode.

Answer	Frequency	Percent
Only individual mode	105	68.2
Sometime individual mode	15	9.7
Only partner mode	17	11.0
Any mode	17	11.0
Total	154	100.0

6. CONCLUSION

A survey was conducted to know about the students' attitude to the common problems in adopting a pair programming as a tool to learn software engineering in practical laboratories. The survey was conducted by floating questionnaire, after implementing pair programming methods among the students in the laboratory course works. The answers collected as data was analyzed through various methods. The results indicate that the students like to adopt pair programming as a learning methodology in the lab course work. They also like to have partners whose academic achievement are same or higher. There was no gender impact for the female students in selecting their partners as a pair. Finally students were in favour of appearing final exams individually, inspite they practice and learn to develop software programs in pairs. The above findings are helpful to study further in applying pair programming as a tool in teaching and learning environment.

7. REFERENCES

[1] Arisholm, E., Gallis, H., Dybå, T., and Sjöberg, D.I.K. 2007. Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise. *IEEE Transaction on Software Engineering*. 33, 2, 65-85.

[2] Beck, K. Extreme programming explained: embrace change. 2000. Addison Wesley.

[3] Beth, S., and Brai, H. 2008. First Year Students Impression of Pair Programming, *ICER'07*. ACM. pp. 73-85.

[4] Brain, H. Problems Encountered by Novice Pair Programmers. 2008. *ACM Journal on Educational Resources in Computing*, vol. 7, No.4, Article 2, pp. 2:1 – 2:13.

[5] Charlie McDowell., Linda Werner., Heather E. Bullock, and Julian Fernald. 2003. The Impact of Pair Programming on Students Performance, Perception and Persistence. *25th International Conference on Software Engineering (ICSE'03)*, IEEE Computer Society. pp. 602-607.

[6] Declue, H.T. 2003. Pair Programming and Pair Trading Effects on Learning and Motivation in a CS2 Course. *Journal of Computing for Small Colleges*. 18 (5). pp. 49-56.

[7] Emilia Mendez, Lubna AL-Fakhri, and Andrew Luxton – Reilly.2006. A Replicated Experiment of Pair-Programming in a 2nd –year Software Development and Design Computer Science Course. *ITiCSE'06, Bologna, Italy*. ACM. pp. 108-117.

[8] Freudenberg, S., Romero, P., and du Boulay, B. 2007. Talking the talk': Is intermediate – level conversation the key to the pair programming success story?. *Proceedings of the AGILE 2007, IEEE Computer Society Washington, DC, USA*. pp.84-91.

[9] Gert, V.V., Tommi, T., Ber de, B. and Douwe, P. 2011. Quantitative release planning in extreme programming', *Journal of Information and Software Technology*, Vol. 53, Issue 11, November 2011. pp. 1227–1235

[10] Grant Braught., L. Marlin Eby., and Tim Wahls. 2008. The Effects of Pair Programming on Individual Programming Skill. *SIGCSE' 08, Portland, Oregon, USA,ACM*. pp.200 – 204.

[11] Grant Braught., John McCormick. and Tim Wahls. 2010. The Benefits of Pairing by Ability. *SIGCSE'10, Milwaukee, Wisconsin, USA,ACM*. pp.

[12] Grant Braught. and Tim Wahls. 2011. The Case for Pair Programming in the Computer Science Classroom. *TOE00041 ACM Transaction on Computing Education*, vol. 11, No. 1, Article 2, pp 2:1 – 2:21.

[13] Gunalan, R., Shereshevsky, M. and Ammar, H.H. 2005. Pseudo Dynamic Metrics, Computer Systems and Applications. *3rd ACS/IEEE International Conference on Digital Object Identifier: 10.1109/AICCSA.2005.1387106*.

[14] Hanna, H. and Pekka, A.2005. A Multiple case study on the Impact of Pair Programming on Product Quality. *ICSE'05. St. Louis , Missouri, USA,ACM*. pp. 495-504.

[15] Ilenia Fronza., Alberto Sillitti. and Giancarlo Succi. 2009. An Interpretation of the Results of the Analysis of Pair Programming during Novices Integration in Team. *Third International Symposium on Empirical Software Engineering and Measurement, IEEE*. pp. 225-235.

[16] Jo E. Hannay., Erik Arisholm., Harald Envik., Dag and I.K Sjøberg. 2010. Effects of Personality on Pair Programming. *IEEE Transactions on Software*

Engineering vol. 36, No. 1, pp. 61-80.

- [17] Jeffrey, C.C., Lisa, H., Lulu, H., Julia, H. and Donna, R. 2007. Increased Retention of Early Computer Science and Software Engineering Students using Pair Programming. *20th Conference on Software Engineering Education & Training (CSEET'0) 7, IEEE Computer Society*. pp.115-122.
- [18] Joshua Sennett and Mark Sherrif. 2010. Compatibility of Partnered students in Computer Science Education. *ACM SIGCSE '10 Milwaukee, Wisconsin, USA*, pp. 244-248.
- [19] Kuppaswami, S. and Vivekanandan, K. 2004. The Effects of Pair Programming on Learning Efficiency in Short Programming Assignments. *Informatics in Education*, vol. 3, pp. 251-266.
- [20] Lui, K.M. and Chang, K. C. C. 2003. When does a Pair Outperform Two individuals?. *Proceedings of the 4th international conference on Extreme programming and agile processes in software engineering*. Springer-Verlag Berlin, Heidelberg pp. 225-233.
- [21] Lui, K.M. and Chang, K. C. C. 2008. Software process fusion by combining pair and solo programming. *Institute of Engineering and Technology Software*. Vol. 2, no. 4, pp. 9-390.
- [22] Laurie Murphy., Sue Fitzgerald., Brain Hanks. and Renee McCauley. 2010. Pair Debugging: A Transitive Discourse Analysis. *ACM ICER'10, Aarhus, Denmark*. pp. 51-58.
- [23] Lee, R.W. and Tara, T.S. 2009. A National Study of Community College Retention Rates Segmented by Institutional Degree of Urbanization. *Academic Leadership: the online journal*. Vol. 7, Issue 1. <http://www.Academi-leadership.org/>
- [24] Militon, F. 2005. Correctness: A Very Important Quality Factor in Programming. *Studia Univ. Babeş-Bolyai, Informatica, Volume L, Number 1*. pp.11-20.
- [25] Nagappan, N., Williams, L., Wiebe, E., Miller, C., Balik, S., FerZil, M. and Petlik, M. 2003. Pair Learning: With an eye Toward Future Success. *Extreme Programming and Agile Methods - XP/Agile Universe 2003. Lecture Notes in Computer Science Volume 2753*. pp. 185-198.
- [26] Norman Jacobson and Suzanne K. Schaefer. 2008. Pair Programming in CS1: Overcoming Objection to its Adoption. *Inroads – SIGCSE Bulletin, Vol. 40. Number 2*. pp 93-96.
- [27] N. Salleh, E. Mendes, J. Grundy, and G.S.J. Burch. 2010. The effects of neuroticism on pair programming: An empirical study in the higher education context. *Proceedings of the 4th ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM 2010)*.
- [28] N. Salleh, E. Mendes, J. Grundy, and G.S.J. Burch. 2010. An empirical study of the effects of conscientiousness in pair programming using the five-factor personality model. *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering (ICSE 2010)*, Vol. 1. pp. 577-586.
- [29] N. Salleh, E. Mendes, and J. Grundy. 2011. Empirical studies of pair programming for CS/SE teaching in higher education: A systematic literature review. *IEEE Transactions on Software Engineering*. Vol. No. 4. pp. 509-525.
- [30] N. Salleh, E. Mendes, and J. Grundy. 2011. The Effects of Openness to Experience on Pair Programming in a Higher Education Context. *IEEE CSEE&T 2011. Waikiki, Honolulu, HI, USA*. pp. 149-158.
- [31] Nuramiah, R., and Shukor, S.M.F. 2008. The Effects of Pair Programming in Programming Language Support. *ITSim 2008. International Symposium on Information Technology*. Vol.1 pp. 1-4.
- [32] Raymond Sison. 2009. Investigating the Effect of Pair Programming and Software Size on Software Quality and Programmer Productivity. *16th Asia-Pacific Software Engineering Conference, IEEE*, pp. 187-193
- [33] Tanja Bipp., Andreas Lepper. And Doris Schmedding. 2008. Pair programming in software development teams- An empirical study of its benefits. *Information and Software Technology, 50, Elsevier*, pp. 231-240.
- [34] Theodore Van Toll III., Roger Lee., and Thomas Ahswede. 2007. Evaluating the Usefulness of Pair Programming in a classroom setting. *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)*, IEEE Computer Society. pp. 302 – 308.
- [35] Thorbjørn Walle. and Jo E. Hannay. 2009. Personality and the Nature of Collaboration in Pair Programming. *Third International Symposium on Software Engineering and Measurement, IEEE*. pp. 203-213.
- [36] Tore Dyba. and Torgeir Dingsoyr. 2008. Empirical studies of agile software development: A systematic review. *Journal of Information. and Software Technolgu., Elsevier*, vol. 50, pp. 833-859.
- [37] Vincent a. Cicirello. 2009. On Self-Selected Pairing in CS1: Who pairs with Whom. *Consortium for Computing Sciences in Colleges. JCSC 24*. pp. 43-49.
- [38] Wallace Chigona. and Michael Pollock. 2008. Pair Programming for Information System Students New to Programming: Students Experiences and Teacher's Challenges. *PICMET 2008 Proceedings, Cape Town, South Africa*, pp. 1587-1594.
- [39] Williams, L., Kessleer., Cunningham, R.R. and Jeffries, R.R. 2000. Strengthening the Case of Pair Programming, *IEEE Software, Vol 17, No 4*, pp 19-25.
- [40] Williams, L., Upchurch, R. 2001. In support of pair programming. *Proceedings of the 32nd SIGCSE Technical symposium on Computer Science Education, SIGCSE Bulletin*. 33(10). PP. 327-331.
- [41] Williams, L., McCrikard, D.S., Layman, L. and Hussein, K. 2008. Eleven Guidelines for Implementing Pair Programming in Classroom. *Agile 2008 Conference , IEEE*. pp. 445-452.