The Needs of Collaborative Tool for Practicing Pair Programming in Educational Setting

https://doi.org/10.3991/ijim.v13i07.10722

Ani Liza Asnawi (✉), Amalina Ahmad, Nor Fadhillah Mohamed Azmin, Kamsiah Ismail, Ahmad Zamani Jusoh, Siti Noorjannah Ibrahim, Huda Adibah Mohd Ramli
International Islamic University Malaysia (IIUM), Kuala Lumpur, Malaysia
aniliza@iium.edu.my

Abstract—Pair programming is a technique which is introduced for helping programmers to develop high quality codes. Beside technical aspects, collaboration and working in team are the important skills needed in practicing the technique. These skills are essential to be instilled among our engineering students. Pair programming (known as PP) is one of Agile software development technique which introduces collaborative ways for programmers to develop software. Many advantages have been reported when one practiced the technique, which include improve team communication and productiveness, reduction in defect counts, and at the same time lessen development time. Among students, the technique was also reported to enhance students' learning and satisfaction, as well as students' learning skills. In order to get the full benefits of pair programming technique, however, correct guidance must be provided to the students. In this study, we proposed a collaborative pair programming tool for students in practicing the technique. The tool offers all features required to allow the collaboration for the pair programming technique to work. The tool is designed and developed using Visual Basic Studio software. A database is incorporated to store students' information. An experiment was conducted to evaluate the students' performance with and without using the tool. Results showed that the tool can help improving students' performance in terms of quality of codes, and lessen their time in completing their coding. These are the consequences of collaboration and team work skills that they have been practicing when using the tool.

Keywords—Pair Programming technique; pair programming tool, collaborative tool, engineering students, educational tool; engineering education

1 Introduction

Pair programming (known as PP) is one of Agile software development technique under Extreme Programming [1, 2]. It is a technique in which two programmers work together [3]. PP is when two programmers work collaboratively on the same algorithm, design and programming tasks [3, 4]. One person is called a 'driver' which controls the mouse or the keyboard and is developing design or code [3, 4]. The other person is called an 'observer', continuously and actively examines the work of the 'driver', watching for defects and thinking of solutions for any problems found in the coding [4]. In pair programming, the two programmers will actively switching their roles [4]. There are two types of pair programming in software development, which are:

- Traditional pair programming
- Distributed pair programming.

The traditional PP consists of two persons sitting side by side on the same computer, in the same place while the distributed PP is where the two programmers work on the same programs but using different computers and can be in different locations [5, 6]. The practice of distributed pair programming can be automated with a tool that providing all features for the collaboration to occur, even though the programmers are in different location.

Most of an engineer's time in software industry is spent working with other programmers [7] and programmers can be from different location and even in different time region. Furthermore, they need to develop the ability to comprehend the programs developed by other programmers or software engineers [8]. Therefore the need to learn PP is essential not only for software developers but also to engineering students as the benefits they can get from practicing PP can help them when working in the industry later. These students need the skill of team work which can be obtained by practicing pair programming [9]. Knowing the benefits PP can offer, educators were also expressing their interests in applying PP in educational institutions [4]. In this paper, we present the development of collaborative tool for practicing pair programming technique. Our focus is on distributed pair programming (DPP) which allowing two persons to work at different computer while working on the same coding. The tool is very essential to be introduced in educational setting as it allows and automates the process of performing the technique correctly. The tool, at the same time can provide the right environment for practicing PP to students.

2 Background and Motivation

Many have reported benefits of practicing pair programming in educational setting [10, 11]. The benefits outweighed the disadvantages when students practicing the technique. A study conducted in computer science classroom showed that students were found to be more confident in their work [11]. Knowledge is consistently being shared between partners [11]. The students also had minimal questions for the teaching staff when practicing the technique. Survey showed that 74% of the students were satisfied working with their partners as they could figure out almost everything [11], while 84% of the class agreed that they have performed better when using the pair programming[11]. There was also one study published on the benefits of implementing pair programming in classroom [12]. The implementation which was well monitored and regulated by instructors at a given time has proved that pair programming enhanced students problem solving skills, improved quality of their works and increased teamwork [12].

At educational setting, introduction can be provided by educators prior to its practice. The practice can be automated with a tool which incorporating all the features needed for practicing the techniques. This is important as it can assist to provide the students with correct environment to practice the technique. Although there are tools available to be used in industry, however from our investigation, none of the tools provide complete features needed in practicing the pair programming particularly in educational setting. In this study, we proposed and developed a pair programming tool with the important features to practice the technique. The tool includes screen sharing application, collaborative work support, floor control, communication channel and platform. Workspace window is for programmers to write their codes whereas screen sharing applications is mainly to support the collaborative activities required in the pair programming. Besides, in order to enhance the collaboration, several communication channels such as text-message, voice call and video call are also included in the tool. The features and functions are needed to provide the right environment to meet the educational setting. Since the focus of this tool is for students, therefore a database is also incorporated. The database stores students' information and tasks (assignments) for educators to check and comments.

3 The Needs for the Tool

Correct guidance to use the technique must be introduced to students. It is essential to have a tool which can automate collaboration among students to the practice PP. This can help students to be independent and having minimum supervision from their instructors. Currently they are several tools available (i.e Gobby, COPPER, VNC) however the practice requires an effective tool support to address new challenges like communication, distributed collaboration and data exchange [12]. Some of the basic requirements for pair programming particularly distributed pair programming (DPP) are workspace control and awareness, screen sharing enables, floor control, gesturing and various communication channels. Furthermore, although most of the tools mentioned are currently being used in industry [12] but none of the tools investigated are suitable to be used by students. To the best of our knowledge, we could not find any tools that provide all the features which are required by the students. For example, VNC does not have collaborative works support which is needed by students when doing their tasks. In addition medium for communication is also not supported by the VNC. In practicing the technique, students need to share and exchange information between them. While for other tools such as ACE, MoonEdit, GrewPEdit and Gobby, they do not have the screen sharing applications like VNC [12]. Screen sharing applications is crucial to be included in the tool as students need them in order to communicate (between partners) when doing the programming tasks separately or virtually. Table 1 shows the summary of the features provided by the existing pair programming tools.

Table 1. Difference in Features for Existing Tool

Tools' features	Existing PP Tools				
	ACE	*MoonEdit*	*GrewpEdit*	*Gobby*	*VNC*
Screen Sharing Application	NO	NO	NO	NO	YES
Collaborative Work Support	YES	YES	YES	YES	NO
Floor Control	YES	YES	NO	NO	YES
Communication	YES	NO	YES	YES	NO
Platform	YES	YES	YES	YES	YES

4 Methodology

The project applied an evolutionary process model; a prototyping methodology. It is a software process model which allows development of increasingly more complete version of the software [13] and can help to obtain the requirements for the system [14]. This section presents the development of the tool. Flow of the tool is shown as in Figure 1.

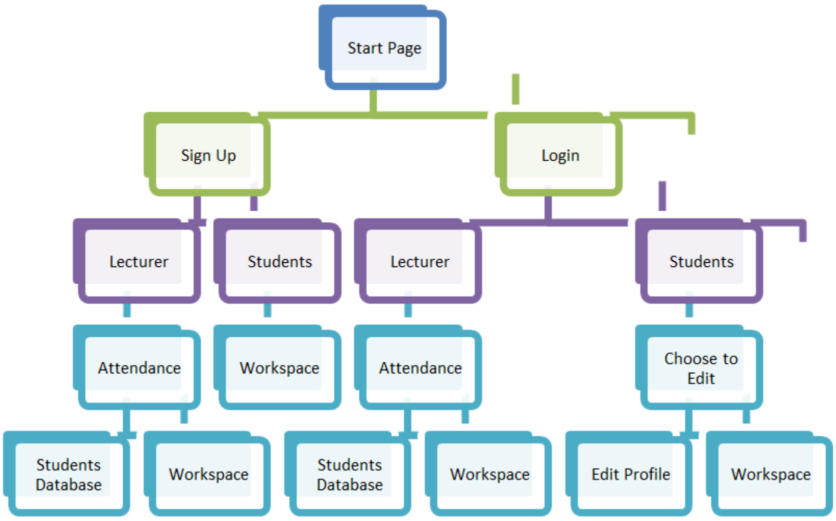


Fig. 1. Flow Design of Pair Programming Tool

As in Figure 1, the tool is designed with signup and login button for both students and lecturers. The tool has features for students' and lecturers' usage. A workspace is provided for students to do coding and there is a platform for lecturers to check students' attendance (from students' database) and their students' coding (workspace). The tool is designed and developed using Visual Basic Studio (VBS) and a MySQL database is incorporated together using C# programming language. The start page for the tool is shown as in Figure 2.

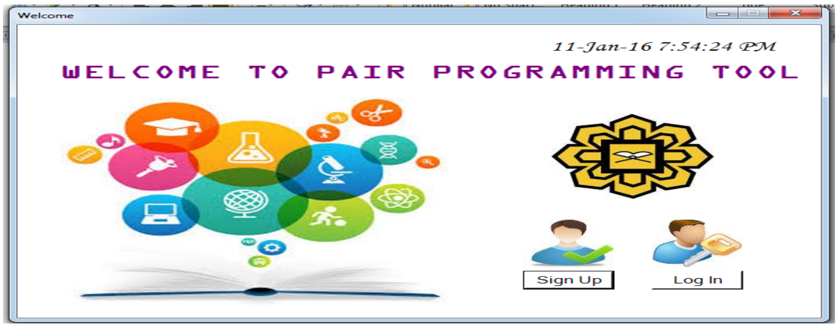
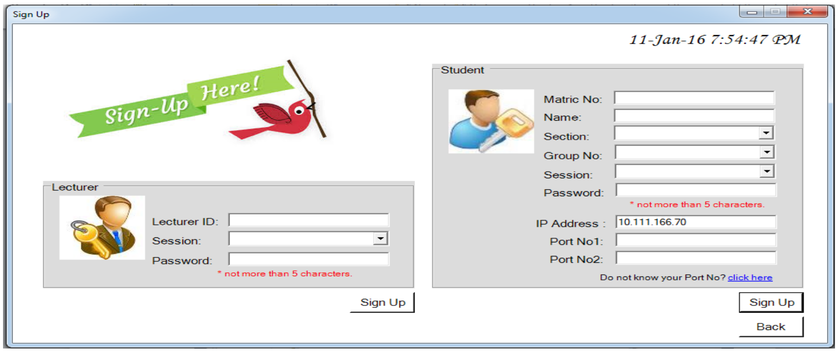


Fig. 2. Start Up Page for the Tool

In the start page, there are two buttons to choose; (i) signup button; if they are using the tool for the first time and (ii) login button; if they have already stored their information. In the signup page, students need to key in all of their information as shown in Figure 3 below. The data is stored in the database for their lecturers' reference.



IP Address : 10.111.166.70

Fig. 3. Sign Up Page

Next, is the login page (Figure 4), which students can enter their matric number and password to start using the tool. The login page will retrieve data from the database. Then students will be directed to the workspace window. From here, they can choose development software to start working on the programming.

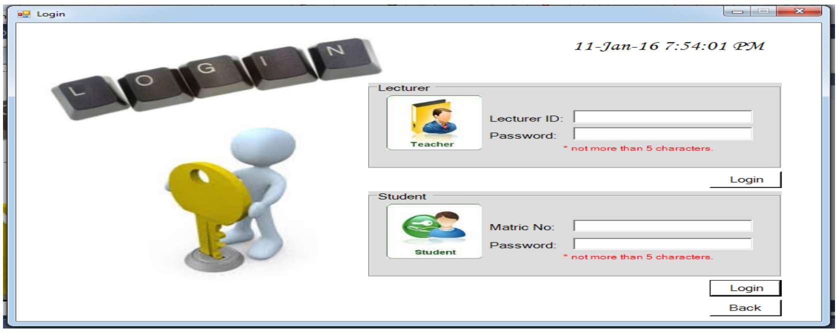
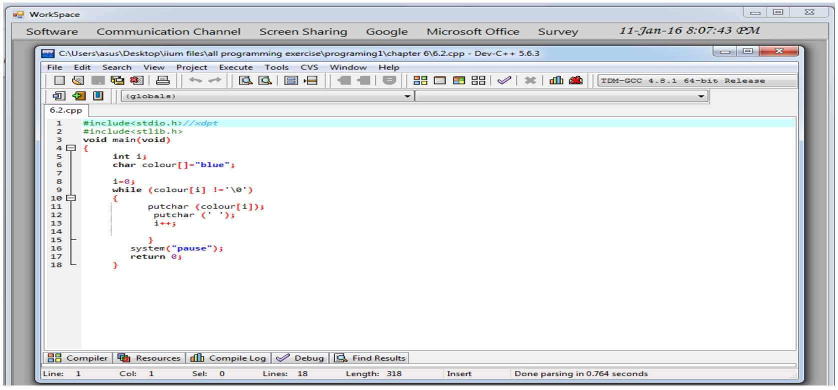


Fig. 4. Login Page

The students' workspace is shown in Figure 5 below. In this page, there are six buttons at the upper row. The buttons include 'software', 'communication channel', 'screen sharing', 'google', 'microsoft office' and 'survey'. These buttons have their own functions that will be discussed in this section. The workspace shows the page of the programming software for students to write their coding in completing their programming assignments.



```
#include<stdio.h>//xdpt
#include<stlib.h>
void main(void)
{
        int i;
        char colour[]="blue";

        i=0;
        while (colour[i] !='\0')
        {
              putchar (colour[i]);
               putchar (' ');
               i++;

              }
           system("pause");
           return 0;
        }
```

Line: 1 Col: 1 Sel: 0 Lines: 18 Length: 318 Insert Done parsing in 0.764 seconds

Fig. 5. Workspace

Together in this tool, there are four types of communication channels attached to the workspace. They are text message, voice call, video call, and email. Students can select their preference for communication channel when using the tool. Figure 6 shows a text message channel i.e one of the communication channels included in this tool. Students can communicate with their partners while doing their programming tasks using the communication channel. Students just need to enter their matric number and their partner's matric number before sending the message. The IP address and port number will be retrieved from the information given earlier when they signed up (from Fig 3). Then, they need to click on 'connect' button to be linked to their partner (on different computer).

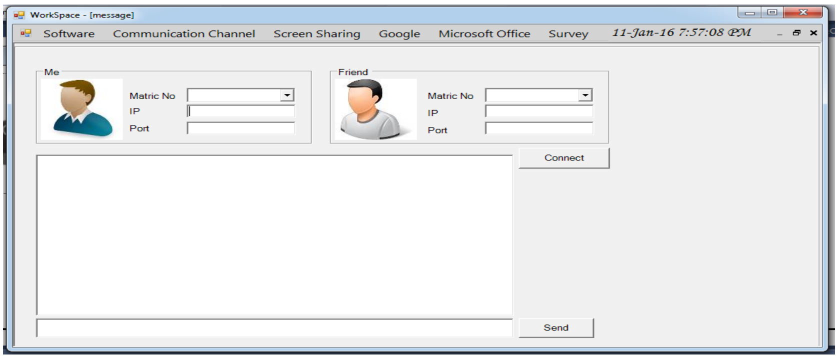


Fig. 6. Communication Channel Workspace (Text Message)

Figure 7 is the voice call application where students can communicate with their partners. Same as before, they only need to enter their matric numbers and their partner's matric number.

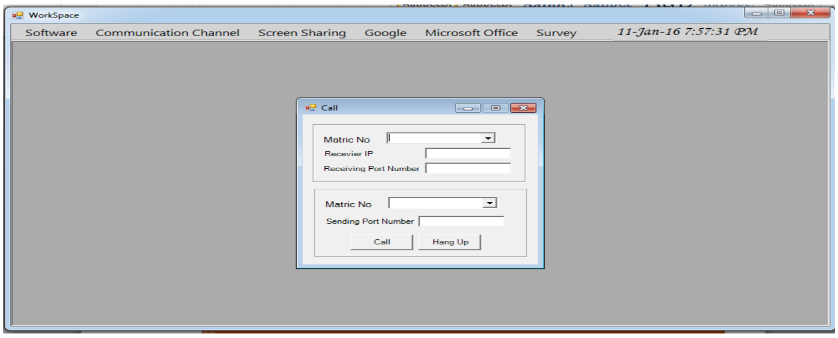


Fig. 7. Communication Channel Workspace (Voice Call)

The video application is shown in Figure 8. Using this page, students can communicate with their partners through video call/webcam.

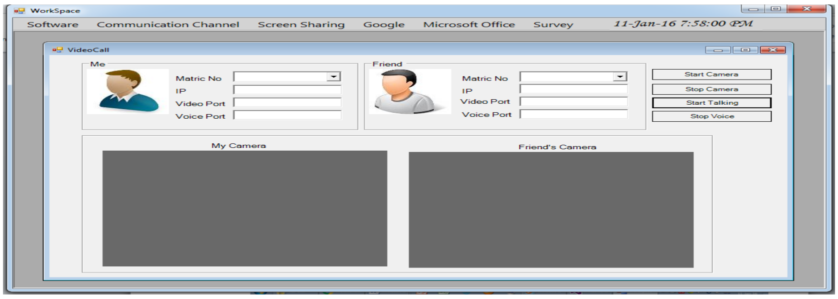


Fig. 8. Communication Channel Workspace (Video Call)

At the same time, students using the tool can also communicate through email. They can submit their completed assignments to their partners or instructor (as shown in Figure 9).

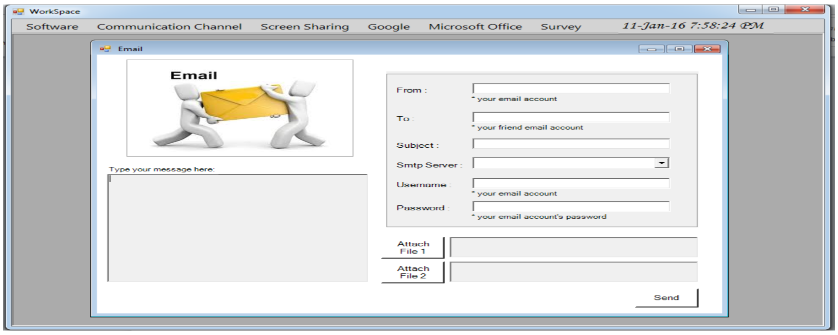
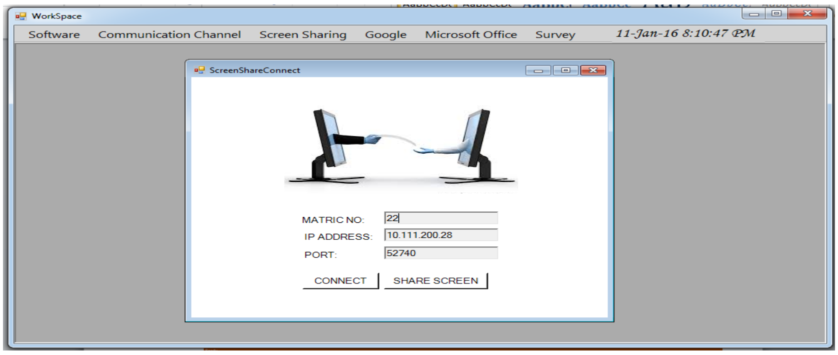


Fig. 9. Communication Channel Workspace (Email)

Figure 10 shows screen sharing page, which direct students to share their desktop screen with their partner to perform the pair programming. This way, students can see their partner's coding activities. Practicing PP, the person who acts as a navigator can observe any errors that occur in the coding, while his/her partner is developing his/her code.



MATRIC NO: 22
IP ADDRESS: 10.111.200.28
PORT: 52740

Fig. 10. Screen Sharing Page

In addition, the tool includes Google tab thus allowing students to search any information they require in completing their assignment. This can save students' time as everything can be obtained from this tool.

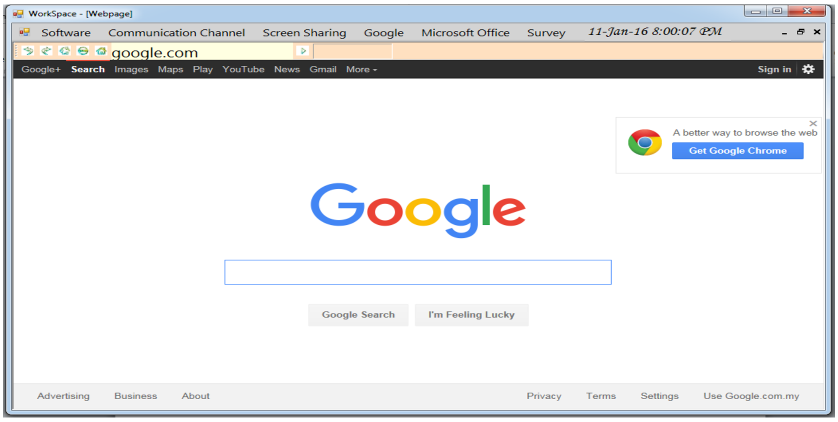


Fig. 11. Google

Furthermore, students can have access to Microsoft Office directly when using the tool. Figure 12 presents one example of directing student to Microsoft Office (Power Point), when using the tool. Microsoft Office Excel and Word are included together in this tab.

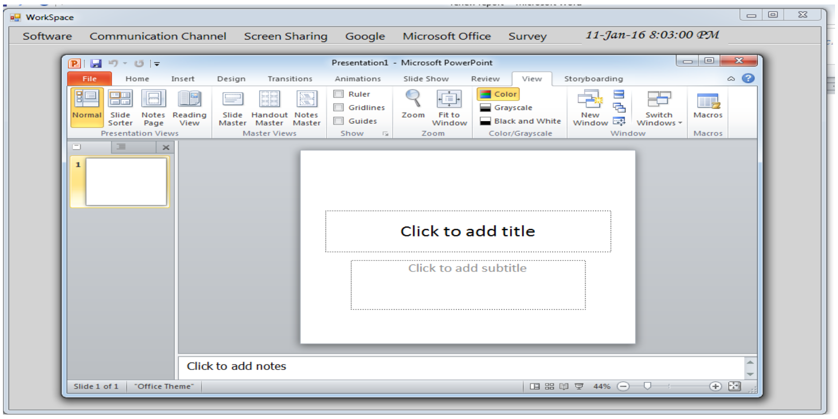

WorkSpace
Software
Communication Channel
Screen Sharing
Google
Microsoft Office
Survey
11-Jan-16 8:03:00 PM
Presentation1 - Microsoft PowerPoint
File Home Insert Design Transitions Animations Slide Show Review View Storyboarding
Normal Slide Sorter Notes Page Reading View
Presentation Views
Slide Master Handout Master Notes Master
Master Views
Ruler Gridlines Guides
Show
Zoom Fit to Window
Zoom
Color Grayscale Black and White
Color/Grayscale
New Window Switch Windows
Window
Macros
Macros
Click to add title
Click to add subtitle
Click to add notes
Slide 1 of 1 "Office Theme"
44%

Fig. 12. Microsoft Office

Finally, after finishing their work using the tool, students can give feedback by clicking on the survey tab. Our proposed tool provides all features required to support collaboration activities when practicing the pair programming technique. The tool assists the instructor in focusing to the programming subject whereas the collaboration and pair programming activities are automated by the tool.

5 Results and Discussion

In order to evaluate the effectiveness of the developed tool, an experiment was conducted with first year students from programming class, ECE 1322. In this experiment, nine students were chosen by their lecturer. Five of them were moderate students and another four were excellent students. They were chosen based from their midterm examination's results. A programming question was given to them. Duration of half an hour was allocated for them to complete the program. Prior to the experiment, an introduction about pair programming and the developed tool were given. It is important to provide understanding about the pair programming technique to students and how the tool can help them in performing the PP technique and completing their programming tasks.

They were three categories for the experiment:

- Two groups using the tool (different computer)
- Two groups of traditional partnered but without using the tool (TPP)
- Solo (only one student- without the tool).

Each group (excluding solo) consisted of 2 students (pairing).

In the first category, we had two pairs which were:

- Moderate and moderate
- Excellent and moderate

The second group consisted of:

- Excellent-excellent
- Moderate-moderate students

While in the solo category we placed only one excellent student in it. Summary for the category is shown as in figure 13.

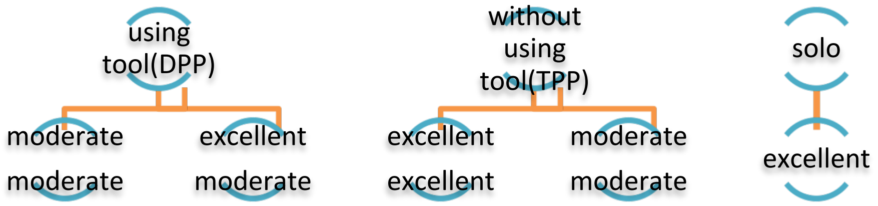


Fig. 13. Categories for the Experiment

The experiment was conducted in order to investigate students' performance when using and not using the tool. The performance was measured in terms of

- The difference in students programming marks between groups.
- The difference in the amount of time spent by the students to complete the programming.

Based on the experiment and from the descriptive analysis, students that used our proposed tool were identified to receive higher marks than those who were doing the programming traditionally partnered (without tool) or solo (without tool). Figure 14 displays the difference of marks among the groups (full marks is 20).

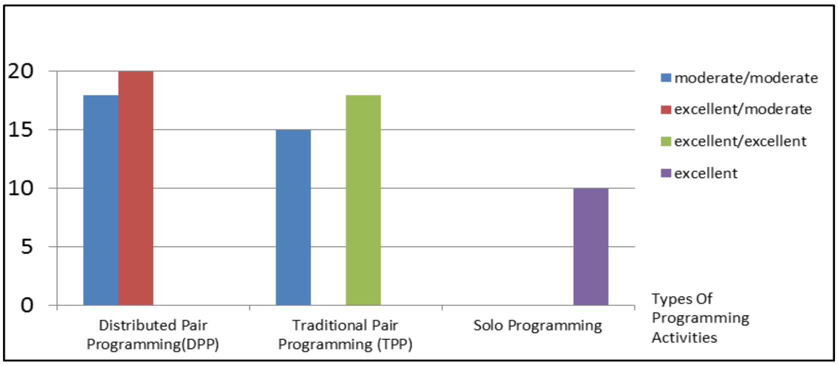


Fig. 14. Difference in Students Programming Marks between the Group

The students using the tool obtained the expected output for the problem given. In addition, their codes were neat and tidy and also they only had few lines of coding when compared to the students without using the tool. From the graph, we can see that in general students using the tool obtained higher marks. It is also observed that the pair of ‘excellent/moderate’ students received full marks compared to the pair of ‘moderate/moderate’ students. Their marks (using tool) were higher than the groups without using tool (TPP) and in solo.

Figure 15 shows the time spent by the group using the tool (i.e DPP) was shorter (30minutes) than the solo students (40 minutes). From the results, we can say that the students with tool performed their work quicker than the solo student (without using the tool). Although we assigned an excellent student in the ‘solo’ category, he did not perform his work well as in comparison to the students working in pairs (with tool and even without tool). The results for using TPP and DPP were obtained from the group of moderate/moderate students (same level).

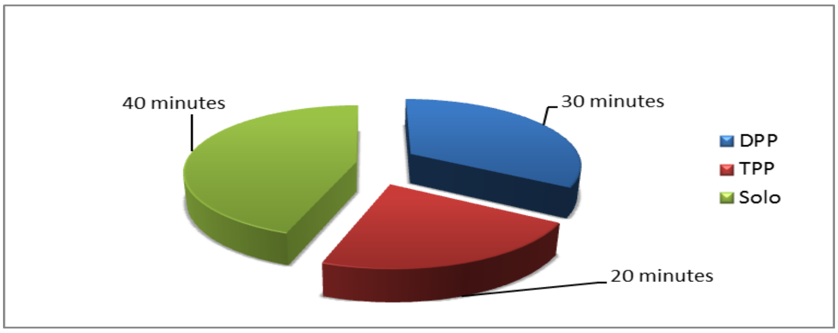


Fig. 15. The amount of Time Spent by the Students to Complete the Programming

Interestingly, the time spent when using the tool (DPP) was longer than the group that was traditionally partnered (TPP- without tool), as shown in Figure 15. From our observation, this might due to students' familiarity with the tool. They were (group of DPP) still new with the tool, which required more time for them to get used to it, resulting to longer time compared to traditionally partnered students (without tool). From the experiment, we did not see any drawbacks from the students practicing pair programming. This is because the students were selected from the same level of studies. Even though there was a group of student having 'excellent-moderate' partner, however the excellent student was not being dominant towards the moderate student. The experiment also showed that pair programming has helped students in getting better marks. This is the result from teamwork activities and collaboration they had when practicing the pair programming technique.

At the end of the experiment, the students were asked to answer the survey questions related to the tool (included as the tab button in the tool). As a result, overall, students were very satisfied with the easiness when using the tool. They found that it was easy to find any information needed through the tool. The interface was pleasant. They also agreed that the tool had all the functions and capabilities that they needed. Furthermore, the tool has assisted them to complete their programming tasks. Finally they have recommended for the tool to be used in programming class. The tool can be introduced to students in completing their programming assignment. (Please email the main author to get the survey questions and detailed results).

6 Conclusion

Educators and engineering educators particularly are always looking for ways to incorporate activities that would increase students' learning and at the same time students' collaborative skill. The main objective of this project was to develop a collaborative pair programming tool for students. The tool was developed to help students practicing the pair programming technique in completing their tasks. Pair Programming was chosen because of the reported benefits it can delivered [15-17] and the same time it can help to expose students to the real working environment. The features we developed in the tool were found to help students to achieve better communication with their partner when doing the programming tasks. Furthermore, the tool has been assisting students in practicing the technique of pair programming with less supervision from their lecturers. The tool provides benefits and has impact in educational setting, for both students and educators. As we all realize, besides the academic aspects, it is very important to instill collaborative skills, team work and communication skills to our engineering students. We want to prepare our engineering students not only to be good in technical aspects, but also in soft skills. From the experiment, the findings showed that the tool can help to improve students' performance in terms of time (reduced) and also their grades (higher). When compared to solo, students practicing pair programming (traditionally partnered, without tool) has shown to receive better marks while when pair programming is practiced with tool, the performance of the students is better than those without using tool. The findings showed that students with tool received almost the full benefits PP technique can offer. For the future, we recommend educators to familiarize students with the tool prior to its usage and increase number of participants in the study. This is to ensure fair results to be produced when evaluating the effectiveness of the tool. We also suggest educators to conduct the evaluation in a full semester; hence can provide inferential analysis from the results.

7 References

[1] T. Dyba and T. Dingsoyr, "Empirical studies of agile software development: A systematic review,"*Information and Software Technology,* vol. 50, pp. 833-859, Aug 2008. https://doi.org/10.1016/j.infsof.2008.01.006

[2] J. A. Livermore, "What Elements of XP are being Adopted by Industry Practitioners?," in *SoutheastCon, 2006. Proceedings of the IEEE*, 2006, pp. 149-152. https://doi.org/10.1109/second.2006.1629340

[3] J. Nawrocki*, et al.*, "Pair programming vs. side-by side programming," *Software process improvement,* pp. 28-38, 2005. https://doi.org/10.1007/11586012_4

[4] L. Williams*, et al.*, "In support of pair programming in the introductory computer science course," *Computer Science Education,* vol. 12, pp.197-212, 2002. https://doi.org/10.1076/csed.12.3.197.8618

[5] B. Hanks, "Virtual Pair Programming," in *Doctoral Symposium at the International Conference on Software Engineering (ICSE 2003)*, 2003.

[6] K. M. Lui and K. C. Chan, "Software process fusion by combining pair and solo programming," *IET software,* vol. 2, pp. 379-390, 2008. https://doi.org/10.1049/iet-sen:20070035

[7] R. M. Poston and M. P. Sexton, "Evaluating and selecting testing tools," in *Assessment of Quality Software Development Tools, 1992., Proceedings of the Second Symposium on*, 1992, pp. 55-64. https://doi.org/10.1109/aqsdt.1992.205836

[8] T. VanDeGrift, "Coupling pair programming and writing: learning about students' perceptions and processes," in *ACM SIGCSE Bulletin*, 2004, pp. 2-6. https://doi.org/10.1145/1028174.971306

[9] S. Ambler, *Agile modeling: effective practices for extreme programming and the unified process*: John Wiley & Sons, 2002.

[10] C. Bravo*, et al.*, "A groupware system for distributed collaborative programming: usability issues and lessons learned," in *Proceedings for*, 2007, p. 6.

[11] A. Cockburn and L. Williams, "The costs and benefits of pair programming," *Extreme programming examined,* pp. 223-247, 2000.

[12] S. Goel and V. Kathuria, "A novel approach for collaborative pair programming," *Journal of Information Technology Education: Research,* vol.9, pp. 183-196, 2010. https://doi.org/10.28945/1290

[13] R. S. Pressman, *Software engineering: a practitioner's approach*: Mc Graw Hill Education, 2015.

[14] A. L. Asnawi*, et al.*, "Empirical Investigation on Agile Methods Usage: Issues Identified from Early Adopters in Malaysia," *Agile Processes in Software Engineering and Extreme Programming,* pp. 192-207, 2011. https://doi.org/10.1007/978-3-642-20677-1_14

[15] S. Salinger*, et al.*, "Saros: an eclipse plug-in for distributed party programming," in *Proceedings of the 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering*, 2010, pp. 48-55. https://doi.org/10.1145/1833310.1833319

[16] P. Brereton*, et al.*, "Pair programming as a teaching tool: a student review of empirical studies," in *Software Engineering Education and Training, 2009. CSEET'09. 22nd Conference on*, 2009, pp.240-247. https://doi.org/10.1109/cseet.2009.11

[17] B. F. Hanks, "Tool support for distributed pair programming," in *Workshop on Distributed Pair Programming. Extreme Programming and Agile Methods-XP/Agile Universe*, 2002. https://doi.org/10.1007/978-3-540-27777-4_9

8 Authors

Ani Liza Asnawi is an engineering educator in Electrical and Computer Dept, Faculty of Engineering, International Islamic University Malaysia. She received her PhD from School of Electronics and Computer Science, University of Southampton, United Kingdom (2012), Master Degree in Communication and Computer Engineering from University Kebangsaan Malaysia (UKM), and Bachelor Degree (Computer and Information Engineering) from International Islamic University Malaysia (IIUM). She is currently looking for productive ways in educating engineering students. Her other research interests include wireless communication, software defined radio, software engineering, empirical software engineering, Agile methods and software processes. She is an active Senior Member of IEEE (The Institution of Electrical and Electronic Engineers), IEEE Computer Society, and a registered member for BEM (Board of Engineers Malaysia) and IEM (The Institutions of Engineers Malaysia).

Amalina Ahmad, Nor Fadhillah Mohamed Azmin, Kamsiah Ismail, Ahmad Zamani Jusoh, Siti Noorjannah Ibrahim, and **Huda Adibah Mohd Ramli** are with the Electrical and Computer Dept, Faculty of Engineering, International Islamic University Malaysia, Kuala Lumpur, Malaysia.

Article submitted 2019-04-22. Resubmitted 2019-05-27. Final acceptance 2019-06-04. Final version published as submitted by the authors.