Available online at www.sciencedirect.com

ScienceDirect

Procedia - Social and Behavioral Sciences 90 (2013) 293 – 302

Procedia
Social and Behavioral Sciences

6th International Conference on University Learning and Teaching (InCULT 2012)

Designing Prototype Model of an Online Collaborative Learning System for Introductory Computer Programming Course

Mahfudzah Othman[a*], Muhaini Othman[b], Fazlin Marini Hussain[c]

[a]Faculty of Computer & Mathematical Sciences, Universiti Teknologi MARA, 02600 Perlis, Malaysia
[b]Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400 Johor, Malaysia
[c]Faculty of Computer & Mathematical Sciences, Universiti Teknologi MARA, 26400 Pahang, Malaysia

Abstract

This paper discusses the design of the prototype model of the online collaborative learning system for introductory computer programming course. The methodology used involves three phases which are the data collection, analysis and design and the implementation phase. Initially, fifty respondents from the first year students of the Diploma in Computer Science in Universiti Teknologi MARA (UiTM) Perlis, Malaysia have been randomly selected to participate in the data collection phase in order to investigate the students' interests, learning styles as well as their learning preferences. The results have shown the need for the development of online small group discussions that could facilitate online communication and collaboration from dispersed location, hence encouraging distance learning education. A design of a structure model for an online collaborative learning system has been constructed in order to support the online collaborative learning activities in a virtual environment. The logical designs of the Online Collaborative Learning System or OCLS are being designed using the object-oriented models which are the use-case model and class diagram in order to show the concise processes of virtual "Think-Pair-Share" collaborative activities. The "Think-Pair-Share" collaborative learning technique that is being used in the design structures has been chosen because of its simplicity and relatively low-risk. Later, the physical design of the prototype model is being constructed using the Web-based technologies which are the MySQL database, PHP and Apache web server. This paper also discusses the impact of the online collaborative learning system towards the students' performance where analysis has shown that the t-test result had a significant value of 0.01, which is less than 0.05 (sig. 2-tailed).

Selection and/or peer-review under responsibility of the Faculty of Education, University Technology MARA, Malaysia.

Keywords: collaborative learning; introductory computer programming; online collaborative learning system

* Corresponding author. Tel.: +6049882000; fax: +6049882501.
E-mail address: fudzah@perlis.uitm.edu.my

Selection and/or peer-review under responsibility of the Faculty of Education, University Technology MARA, Malaysia.
doi:10.1016/j.sbspro.2013.07.094

1. Introduction

Computer programming is claimed to be a challenging intellectual task. Blackwell, Whitley, Good and Petre (2001) mention that computer programming involves complex activities equivalent to other design and engineering activities. A recent study done by Mohd Norafizal, Mahfudzah & Zainab (2009) from UiTM Pahang, Malaysia had also showed that a significant number of 70% of the Computer Science students agreed that programming courses are considered tough. Due to the nature of the complexity and toughness in learning programming, there were solid evidences showing high failure rates in computer programming courses. Roddan (2002) reported that almost 42.2% of the students have failed in the future programming module and did not move on to the second year in Computing Science course at University of Glasgow. Guibert and Girard (2003) also mention that there were high failure rates between 25% and 80% worldwide in programming introductory courses.

For many years, researchers have been striving to find the right solutions to the problem. Reducing the numbers of failures in programming courses has become their main objectives. Some studies include investigating and comparing the differences in programming languages, paradigms and environments to evaluate the ease of use and appropriateness of the programming languages (Mancy & Reid, 2004). Some studies have also focussed on the differences of the students' learning styles. For instance, by integrating the real-world problems into the classroom or during lab sessions, it will bring relevance to the class that have variety of different personalities (Woszczynski, Guthrie & Sherri, 2005). Furthermore, innovative ideas such as incorporating pair programming or small group discussions in classes have also proven to produce better codes than individuals, making small group discussions a preferred learning style in most of the programming classes (Nagappan, Williams, Ferzli, Wiebe, Yang, Miller & Balik, 2003). Recent studies have shown various ways to use the Web as the mediator to teach and learn computer programming course. SCALE which stands for Supporting Collaboration and Adaptation in a Learning Environment was a web-based system developed to engage students with an active learning environment by providing them with multiple informative and tutoring feedback components (Verginis, Gogoulou, Gouli, Boubouka, & Grigoriadou, 2011). Law, Lee, and Yu (2010) have proposed the used of an e-learning system called the Programming Assignment aSsessment System (PASS) to support the teaching and learning of computer programming. From the findings, they claimed that a well facilitated e-learning environment would be beneficial in enhancing learning motivation and students' self-efficacy. Furthermore, another web-based system named AutoLEP has also been developed to aid novice programmers to attain their programming skills. Through AutoLEP, it has been effectively helped the students to adequately test and evaluate the programs (Wang, Su, Ma, Wang & Wang, 2011).

Although there are a lot of web-based systems that have been developed to support the learning of programming courses, however, the impact of using the web-based systems to the students' performances has still remained elusive. There is a need for further investigations to study how the web-based systems can be used as the medium to teach programming collaboratively and how significant is the effect of online collaboration to the students' performances in programming courses? Therefore, the aim of this study is to introduce the concept of online collaborative learning system to support the teaching and learning of computer programming courses and also to study the impacts that can be analyzed to determine whether it could benefits the students or not in term of their achievements in programming courses.

2. What is collaborative learning?

According to Roschelle and Teasley (1995), collaboration can be described as coordinated, synchronous activity that is the result of a continued attempt to construct and maintain a shared conception of a problem. Varieties of collaborative learning techniques such as the "Round Robin", "Numbered-Heads-Together", "Jigsaw", "Think-Pair-Share" and many others have been widely implemented in a classroom-based environment and they have also been proven effective in enhancing the students' performances. With collaborative learning, it

will focus on helping students to plan and design their problem solving strategy and guide them to evaluate the problem during the learning process (Mahfudzah et al., 2010).

2.1. Computer-Supported Collaborative Learning (CSCL)

An online collaborative system, in order to teach and learn collaboratively, should be supported by specific tools. Therefore, it can be closely related to Computer-Supported Collaborative Learning or CSCL. According to Nik Azlina (2008), CSCL has been seen as the most promising innovations to improve teaching and learning in a virtual environment. With the help of modern information and communication technologies, online collaborative learning is now possible and can be implemented more efficiently. CSCL can also be described as how collaborative activities and their coordination can be supported via computer systems (Mahfudzah, Muhaini, Nurzaid, Atiqqah, Mohd Norafizal, 2010). In previous studies, CSCL has also been referred as a "groupware" or "collaborative software". Gros, Guerra and Sanchez (2005) have also elaborated that the online computer-supported collaborative learning expresses two important ideas which are the idea of learning collaboratively online with other community members and also in a group. In addition, Mahdizadeh (2007) also claims that CSCL can be used to enhance peer interaction and group work where the collaboration and technology used will facilitate the sharing and distribution of knowledge and expertise among community members. Therefore, any online collaborative learning system that adopts the concept of CSCL will provide a more interactive online learning environment that supports and facilitates interaction, negotiation and collaboration amongst and between students and their tutors.

3. The concept of the online collaborative learning system

A virtual-based collaboration differs from the classroom-based collaboration in terms of the interactions among the community members. The main purpose of group collaboration is to bring the team together in order to accomplish a common goal which can be easily achieved within the classroom-based collaborative activities. Therefore, the main challenge for a virtual-based collaboration is; it should be able to facilitate and support the learning activities among the action-oriented teams over the geographic distances and from dispersed locations. The virtual learning system must be able to provide tools that will facilitate communication, collaboration and the process of problem solving. These tools are important in order to provide the team with common means for communicating ideas and brainstorming. In addition, the virtual learning system should also be able to facilitate the project management functions, for instance, the uploading and downloading assignments and time-managements.

Therefore, for this study, by adopting the concept of CSCL, the Online Collaborative Learning System or OCLS will be built using the web-based applications such as MySQL database, PHP and Apache web server where these applications are able to support multi-users simultaneously from dispersed locations, thus providing efficient online communications and discussions. OCLS has been designed with two levels of collaboration tools which have been identified as the communication and the collaborative management tools. Both tools will be used for conferencing purposes and online coordination. The communication tools will involve the Web 2.0 technologies such as online chatting, instant messaging and discussion board. Furthermore, the "Think-Pair-Share" collaborative learning technique is being selected to accommodate the collaborative activities in the virtual environment. It has been found that the "Think-Pair-Share" technique is a relatively low-risk and short collaborative learning technique and therefore, it is ideally suited to be infused in a virtual learning environment.

4. Materials & methods

The research method for this study will be divided into three phases as explained below:

4.1. Data Collection

In order to investigate the students' interest, learning styles and preferences in learning programming subject, a set of questionnaire was given to a sample group consists of 50 first year students from the Department of Computer Science, UiTM Perlis, Malaysia. The analysis derived from the questionnaires will reveal whether there are needs for the development of the online system as an alternative medium for teaching and learning the programming course.

4.2. System Analysis & Design

The Online Collaborative Learning System or OCLS was developed by adopting the concept of Computer-Supported Collaborative Learning (CSCL) with the use of "Think-Pair-Share" collaborative learning technique. There are three main actors designed for the system which are the administrator, lecturer and the student. The collaborative activities involved three main phases which are the "Think", "Pair" and "Share" phases. In the "Think" phase, the students had to access the tasks that have been previously posted by the lecturers. Then, they had to think individually about any possible answers within a given period of time. After that, the "Pair" phase automatically started when the "Think" session ended. During this phase, each of the students was allowed to discuss only with their previously assigned partners using the chat rooms within a pre-set time period. Therefore, the pair had to discuss and submit the possible answers to the lecturer before the timer ended. Lastly, in the "Share" phase, the submitted answers from the pairs were posted on the online discussion board. The answers were graded by the lecturers and can be downloaded by the entire collaborative class.

For the logical design of the online collaborative learning system, the system requirements comprised of two parts which are the functional requirements and data requirements. Both requirements are being depicted using the use-case diagram and class diagram. Finally, the physical design of OCLS is being built using the web-based technologies which are the PHP, Apache web server and MySQL database.

4.3. Implementation of OCLS

OCLS was introduced to a test group of the first year of Diploma in Computer Science students in UiTM Perlis, Malaysia. In order to analyze the effectiveness of using the OCLS in programming classes, another group of the first year students was selected as the control group. Both classes were taking the Fundamentals of Computer Problem Solving course which is compulsory for the first year students. Both groups consisted of 23 students for each class and were taught by the same lecturer with more than 7 years of experience in teaching the fundamental programming courses. Later, in both groups, students' performances in Test 2 were recorded and then compared to see whether the use of OCLS has significant impact on the students' achievements in the programming course or not. For us to see the correlations between OCLS and students' performance in the course, procedure of Bivariate Correlations using Spearman's Rho model was used to measure the strength of association between these two variables. Another test that was conducted was the t-test to further analyze the significant impact of using OCLS in programming classes to students' performances.

5. Results & discussions

5.1. Data collection phase

As discussed in the previous section, the investigation done in this phase consists of questionnaires to investigate the students' interests, learning styles and preferences towards programming subjects. Table 1 below shows the results of the students' interests in learning programming subject where the analysis has showed that about 76% of the respondents (38 students) are more interested working in groups while learning programming. Meanwhile, it has also revealed that about 86% of the respondents (43 students) are interested to use the e-learning portals used by the lecturers for information sharing and knowledge distributions.

Table 1: Students' interests in learning programming subjects

Factors	Percentage
Students are very interested in learning programming	48.0%
Students always update their knowledge about programming	25.0%
Students are expert in programming languages	7.0%
Students are more interested to work in groups while learning programming	76.0%
Students are more interested to solve problems that require logical thinking skills	46.0%
Students are interested to use the i-learn/e-learn portals used by the lecturers for information sharing & knowledge distribution	86.0%

Furthermore, the data collection phase also investigated the students' preferred learning styles and their preferred learning materials. Table 2 below shows the results for the students' preferred learning styles. From the results, it has showed that about 54% of the respondents (27 students) preferred to study programming during exercise sessions in small groups while about 42% of them (21 students) preferred to explore the e-learning portals in order to enhance their knowledge and skills in programming.

Table 2: Students' preferred learning styles

Factors	Percentage
During lectures in classes	56.0%
During exercise sessions in small groups	54.0%
During practical sessions in labs	58.0%
While studying alone	46.0%
While working alone on programming coursework	28.0%
When exploring e-learning portals	42.0%

Meanwhile, Table 3 below depicts the results for students' preferred learning materials. From the analysis, it has showed that about 60% of the respondents (30 students) strongly preferred to use the example programs given by the lecturers to accompany their learning in programming and about 34% of them (17 students) preferred to use the web tutorials, e-learning materials or other Internet resources as their supporting learning materials in learning the programming subject.

Table 3: Students' preferred learning materials

Factors	Percentage
Programming course book	36.0%
Lecture notes/copies of transparencies	56.0%
Exercise questions and answers	58.0%
Example programs given by lecturers	60.0%
Still pictures of programming structures	36.0%
Web tutorials/e-learning portal/other Internet resources	34.0%

All of the analysis derived from the data collection phase had revealed that almost all of the students need more supporting learning materials other than given by the lecturers and they also preferred to work in small groups as an alternative learning strategy to improve their performances. The uses of e-learning portals, web tutorials or other Internet resources have also been the preferred learning materials for the students. Therefore, the findings from this phase have contributed to the analysis and design of the Online Collaborative Learning System (OCLS).

5.2. System Analysis & Design

5.2.1. Logical Design

The logical design involves the system requirements that have been modeled using the use-case diagram and class diagram. As mentioned earlier, the system requirements are divided into two parts which are the functional requirements and data requirements. The functional requirements are being depicted using use-case diagram in Figure 1 where the "Think-Pair-Share" collaborative learning technique is being adopted. Functional requirements include the definition and specification of the set of operations (or use cases) that the OCLS must provide to the actors; who are the lecturers, administrators and the students. These can be divided into administrative, teaching and learning operations:

- Administrative operations and associated use cases are: coordination and management activities such as handling of the users' registrations, creating the classes and most importantly managing the system's administration in whole.
- Teaching operations and associated use cases are: lecturer login, update course content, assigning collaborative classes and partners, engage in collaborative activities, review learning activities, answer questions and requests, provide course assessment results.
- Learning operations and associated use cases are: student login, display course content, perform collaborative learning activities, perform online dialogue and group interaction with collaborative partners, perform course assessment and check course assessment results.

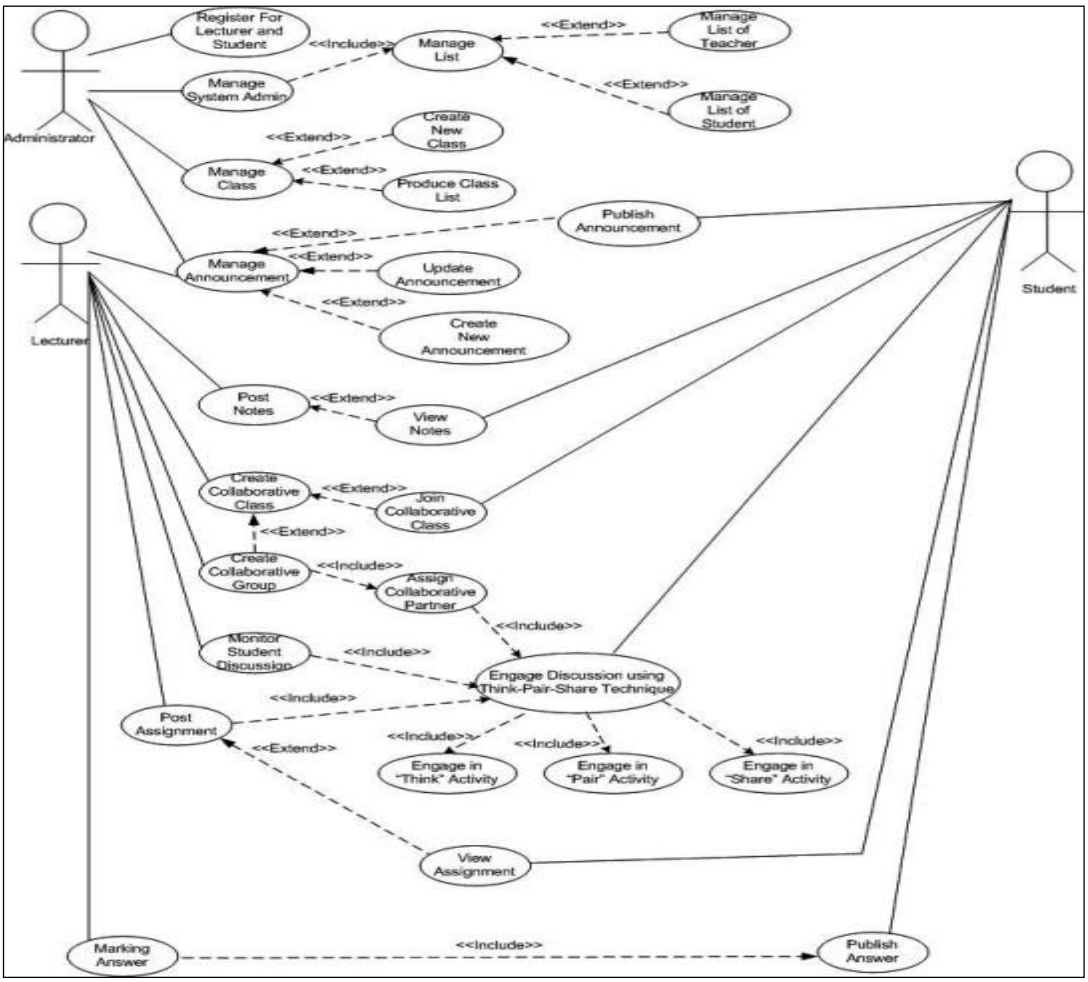

Fig.1 Use-Case Diagram for OCLS

The data requirements for OCLS emphasize on one major model which is the course model. The course model is being presented in Figure 2 using the class diagram. The class diagram for OCLS consists of seven associated classes which are Administrator, Lecturer, Student, Collaborative Class, Collaborative Group, Assignment and Note.

5.2.2. Physical Design

After the logical design has been constructed, the physical design was developed involving the use of MySQL database, PHP and Apache web server. All of these technologies are the open-source technologies where they can be downloaded from the Internet. Figure 3 depicts few of the OCLS interfaces that have been developed such as the interfaces for the Collaborative Group, Think-Pair-Share, Chat Room and Students' Answers. OCLS application can be viewed and accessed by the users via the web browser.

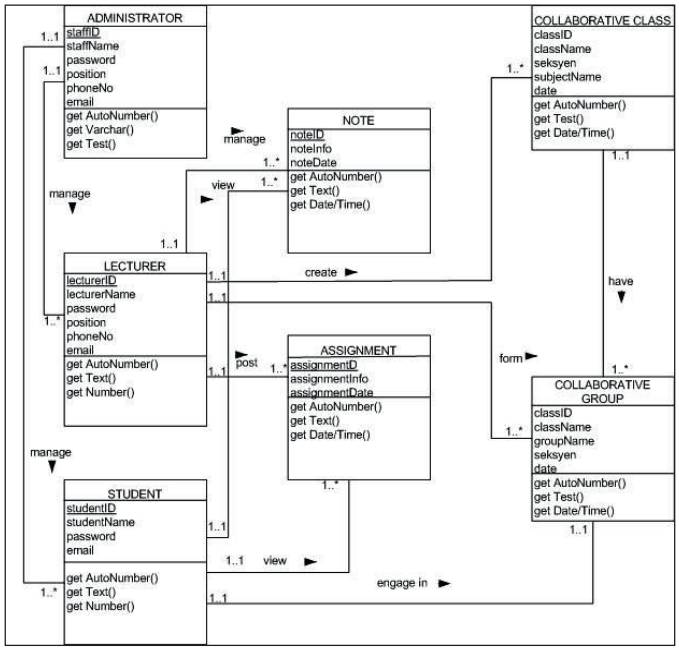

Fig. 2 Class Diagram for OCLS

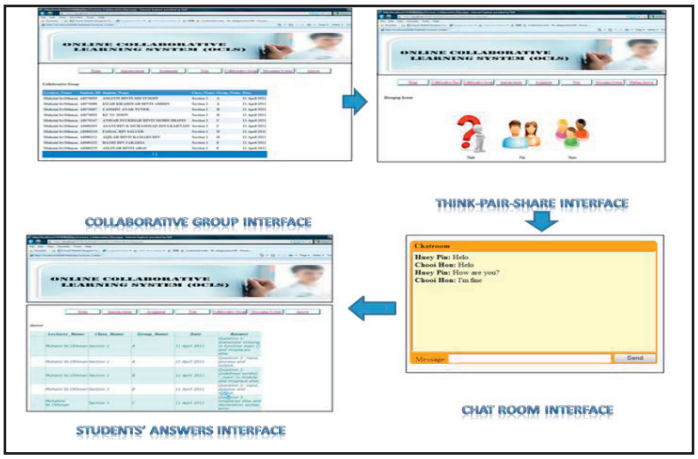

Fig. 3 Interfaces of OCLS application

5.3. Implementation of OCLS

The third part of this study is where the analysis to measure the effectiveness of the OCLS implementation towards the students' performances is being conducted. Table 5 shows the result of the analysis to see the correlation between OCLS and the students' performances. As mentioned earlier, two groups of students, where each group consists of 23 students from the first year of Diploma in Computer Science studies were being selected. The test group is being exposed to OCLS in Fundamentals of Computer Problem-Solving class where

the lecture notes, examples, exercises and tutorials are available via the application. Meanwhile, the control group is being taught using traditional method, where the lecturer will be teaching only by using white board and papers, and examples were given to them in the class without having them to download it via the OCLS application. Before we elaborate more about the impact of the online collaborative system to the students' performance, we first take a look at Table 4 below to see the frequency of students' achievements in their Test 2 results for Fundamentals of Computer Problem-Solving course.

Table 4: Students' results in Fundamental of Computer Problem-Solving course according to sample groups

		Sample groups		
		Test Group	Control Group	Total
Test 2 results	A	1	0	1
	A-	1	1	2
	B+	3	4	7
	B	14	4	18
	B-	3	4	7
	C+	1	6	7
	C	0	1	1
	D+	0	1	1
	F	0	2	2
	Total	23	23	46

From the Table 4 above, as depicted in the test group's results, there were higher achievements in students' performances and there were no students who failed in the Test 2 compared to the control group where there were few students who have scored D+ and F for the Fundamentals of Computer Problem-Solving course. Furthermore, a t-test had been done to find the significant impact of OCLS usage to the students' performance. From the analysis, it shows the significant value is 0.01, which is less than 0.05 (sig. 2-tailed). For further analysis, Table 5 shows that the correlation between OCLS and students' performance is significant at the 0.05 level (2-tailed) with significant value of 0.012. Therefore, we can conclude that the use of the OCLS application can give significant impacts to the students' performance in programming courses.

Table 5: Correlations between OCLS and students' performance in Fundamental of Computer Problem-Solving course

			ProPWeb Tutorial	Students' performance
Spearman's rho	OCLS	Correlation Coefficient	1.000	.367*
		Sig. (2-tailed)	.	.012
		N	46	46
	Students' performance	Correlation Coefficient	.367*	1.000
		Sig. (2-tailed)	.012	.
		N	46	46

6. Conclusion

The classroom-based collaborative learning has been proven effective in enhancing students' performances and individual self-esteems. Such benefits can be easily achieved as the interactions among and between the community members are visible and can be directly monitored and evaluated by the instructors. Meanwhile,

designing the virtual collaborative learning system, on the other hands, involves complex issues and challenging tasks in order to create the virtual environment that suits every community members ideally. The developer needs to consider the selections of the right collaborative learning technique, tools and technologies that can be ideally used in a virtual environment. In this study, series of fact-finding processes have been implemented to further understand the current situations and most importantly to gain users' needs and requirements. The results have showed that the students prefer to work in small groups in order to enhance their understandings in programming and they also prefer to search learning materials from various Internet resources such as from the e-learning portals. Realizing the needs for online materials and online group discussions to accompany students' learning beyond the classroom walls, Online Collaborative Learning System (OCLS) has been designed using object-oriented modeling in order to fulfill the users' requirements as well as the system's functional requirements. A well-known collaborative learning technique known as the "Think-Pair-Share" has been selected to be modeled in OCLS because of its simplicity and suitability to be implemented in a virtual learning environment.

OCLS is being developed using the web-based technologies such as MySQL, PHP and Apache to support the teaching and learning of the introductory computer programming course with the objective to provide supporting virtual learning aids to the students thus promoting active learning in a virtual learning environment. Further analysis have been done to measure the effectiveness of OCLS implementation towards students' performances and therefore, the results have shown significant correlation that proves that the online application is very much needed and had helped the students to achieve better results in programming courses.

References

Blackwell, A.F., Whitley, K.N., Good, J. & Petre, M. (2001). Cognitive Factors in Programming with Diagrams. *Artificial Intelligence Review , 15(1)*, 95-114.

Gros, B., Guerra, V. & Sanchez, J. (2005). The design of computer-supported collaborative learning environments in higher education. *Encounters on Education, vol. 6*, pp 4-23.

Guibert, N. & Girard, P. (2003). Teaching and Learning Programming by Example System. *International Symposium on End User Development.* Retrieved December 11, 2011 from http://www.uni-paderborn.de/cs/ag-engels/Conferences/SE-EUD/Guibert.pdf.

Law, K.M.Y., Lee, V.C.S., & Yu, Y.T. (2010). Learning motivation in e-learning facilitated computer programming courses. *Computers & Education, vol. 55*, pp. 218–228.

Mahdizadeh, H. (2007). *Student collaboration and learning, knowledge construction and participation in an asynchronous computer supported collaborative learning environment in higher education.* PhD-thesis, Socuial Science Group, Wageningen University and Research Centre, The Netherlands.

Mahfudzah O., Muhaini O., Nurzaid M.Z, Atiqqah K.Z. & Mohd Norafizal A. A. (2010). Designing the Online Collaborative Learning System (OCLS) for Introductory Programming Course using Think-Pair-Share Technique, *Proc. 2010 Science, Technology and Social Sciences Regional Seminar*, Pahang, Malaysia, pp. 617-624.

Mancy, R. & Reid, N. (2004). Aspects of cognitive styles and programming. *Proceedings of the 16th Workshop of the Psychology of Programming Interest Group (PPIG 2004), 1-9* Carlow: Ireland.

Mohd Norafizal, A.A., Mahfudzah O., & Zainab, O. (2009). The use of Web tutorial applications to improve students' performance in computer programming language. *Proceedings of International Conference on Languages 2009*, 535-546, Malaysia: Pulau Pinang.

Nagappan, N., Williams, L., Ferzli, M., Wiebe, E., Yang, K., Miller, C., & Balik, S. Improving the CSl Experience with Pair Programming. SIGCSE '03, February 19-23, 2003, 359-362, Reno: NV.

Nik Azlina Nik Ahmad. (2008*). The collaborative teaching environment system using Think-Pair-Share technique*. Masters thesis. University of Malaya, Malaysia.

Roddan, M. (2002). *The Determinants of Student Failure and Attrition in First Year Computing Science.* Undergraduate project report. University of Glasgow. Retrieved November 21, 2011 from www.psy.gla.ac.uk/~steve/localed/roddanpsy.pdf

Roschelle, J. & Teasley, S. D. (1995). *The Construction of Shared Knowledge in Collaborative Problem Solving in Computer Communication and Cognition Vygotskian Perspectives,* pp. 67-197. Cambridge: Cambridge University Press.

Verginis, I., Gogoulou, A., Gouli, E., Boubouka, M., & Grigoriadou, M. (2011). Enhancing Learning in Introductory Computer Sciences Courses Through SCALE: An Empirical Study. *IEEE Trans. Edu., vol. 54, No. 1,* pp. 1-13.

Wang, T.T., Su, X.H., Ma, P.J., Wang, Y.Y. & Wang, K.Q. (2011). Ability-training-oriented automated assessment in introductory programming course. *Computers & Education, vol. 56,* pp. 220–226.

Woszczynski, A.B, Guthrie, T.C. & Sherri, S. (2005). Personality and Programming. *Journal of Information Systems Education*, *16(3)*, 293-300.