Usability Evaluation of Metacognitive Support System for Novice Programmers (MSSNP) using SUMI

Siti Nurulain Mohd Rum[1], Maizatul Akmar Ismail[2]

[1,2] Department of Information System,
Faculty of Computer Science and Information Technology, University of Malaya
Kuala Lumpur, Malaysia
[1]*Email: snurulain {at} siswa.um.edu.my*

ABSTRACT— ***To build effective e-learning is the new challenge faced by software designers and human-computer interaction (HCI) researchers. In computer science, usability is part of the human-computer interaction aspect that relates to the clarity and elegance with which the interaction with software is designed. SUMI (Software Usability Measurement Inventory) is a comprehensive solution, and is a well-tested and validated method for determining whether software has achieved its predetermined goals efficiently and effectively, as evaluated by the end users. This paper presents an empirical study by employing the SUMI (Software Usability Measurement Inventory) to evaluate the MSSNP in terms of the affect, efficiency, control, helpfulness and learnability. The results indicate that MSSNP is an ideal system to support the metacognitve activities for learning Introductory Computer Programming.***

Keywords—MSSNP, usability, novice programmers, human-machine system

1. INTRODUCTION

In Computer Science Education, it has been noted that acquiring knowledge in programming is challenging for novice learners [10, 15]. Recently, a number of approaches, methods and tools have been used to teach the complexity of programming concepts. According to [14], per Figure 1, problem solving activities consist of the following steps:

a. Initiate a specific plan;
b. Analogical problem solving – Applying a solution to a known problem for solving a similar problem;
c. To reduce the problem, general plans are applied;
d. The trial-and-error method is applied for a possible solution; and
e. The execution of this combination of approaches

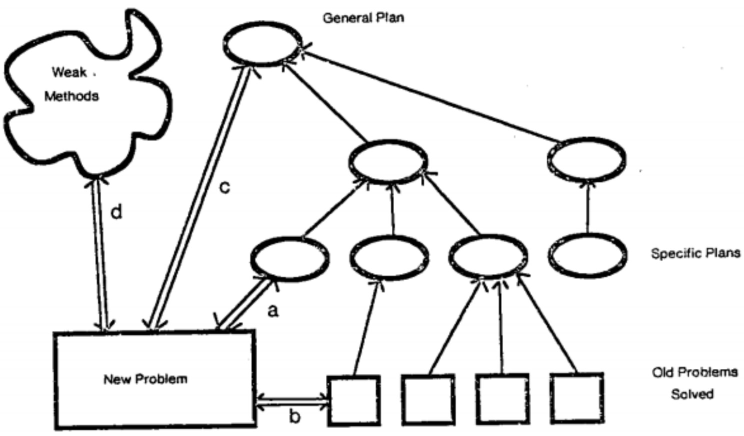

Figure 1. Problem Solving [3]

In problem-solving activities, metacognition plays an important part in directing or monitoring knowledge depending on the context in which it is used. Understanding the importance of metacognition and the related activities to regulate the metacognitive skills will lead to understanding the type of system that would assist a novice programmer in learning computer programming. According to the study by Tobias and Everson [18], metacognition is the blended components of the "knowledge of cognition" and "self-cognitive monitoring". The model of metacognition of Tobias & Everson, as shown in Figure 2, is specifically appropriate for this study, as to define the requirements of the metacognitive activities in learning computer programming. It enables one to concentrate on particular metacognitive skills that are associated with problem-solving. A questionnaire survey on student metacognitive awareness towards the learning success of Computer Programming was conducted using the Metacognitive Awareness Inventory (MAI) developed by [16].

The outcome of the study provides an understanding concerning the relation of metacognitive awareness and learning success in Computer Programming. Five expert lecturers from Universiti Teknologi MARA were also interviewed in identifying the problems in teaching and learning computer programming at the university. The participation was voluntary in nature. The actual requirements of the MSSNP were produced by the results analysis of the interview and survey. These requirements were then translated into the architectural design of the MSSNP. The MSSNP screenshot is presented in Figure 3, Figure 4, Figure 5, and Figure 6. The discussion of the development of the MSSNP can be found in [12, 13, 17]. The usability of the interface design of the software may have a major influence on learning [1]; therefore, measurement of its usability needs to be assessed. Usability testing determines whether the MSSNP meets the pre-defined objectives. In evaluating this product, several measurement tools are available for use, such as UAT, SUMI [7], TAM [2, 3] and others. In this research work, as part of the usability test, the SUMI method is employed to provide an overall judgement and understanding of user satisfaction with the MSSNP.

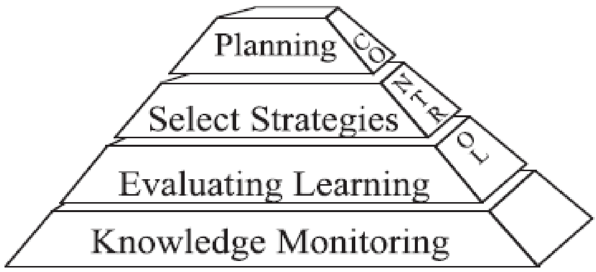

Figure 2. Tobias & Everson's Model of Metacognition

2. USER SATISFACTION ASPECTS

The user satisfaction aspects can be subdivided into five attributes [10] as follows:

- **Efficiency**: how quickly, effectively and economically can users perform tasks using the software
- **Affect**: this psychological term refers to the emotional reaction of users interacting with the software
- **Helpfulness**: the perceptions of users concerning the adequacy of the help support features to assist in solving operational problems
- **Control**: users feel in control of the system and not vice versa
- **Learnability**: the degree of easiness of using the product

The method of evaluating the MSSNP is discussed in the following section in more detail.

3. EVALUATION WITH SUMI

The selection method on user-perceived quality often relies on what is to be measured. The Software Usability Measurement Inventory (SUMI) is a comprehensive solution for measuring the quality of the software from the end user's perspective [12, 13, 17]. The development of SUMI was started in 1990 by the Human Factors Research Group (HFRG) under the project 'Metrics for Usability Standards in Computing' (MUSiC). The objectives of SUMI are to examine the competency scale of CUSI and to expand it further and to standardise the software usability test instrument to be used in a commercial environment. SUMI has successfully achieved its objectives, and, since being published in 1993, it has been widely disseminated internationally. SUMI has also been indicated in the ISO 9241 standard as a technique for assessing the user perception of software [5]. This inventory consists of a set of 50 questions that relate to the affect, efficiency, control, learnability and helpfulness of software [4]. The empirical result of SUMI studies would be very useful for measuring the user satisfaction of the system in terms of its usability. Generally, how long the evaluation process takes using SUMI relies on the system complexity, usually users with no previous experience will need additional time for the introduction and training with the software.

4. MSSNP ENVIRONMENT

The main elements of metacognitive learning instruction involve actively thinking about what the learner knows and how to apply it. The MSSNP system consists of five main activities, namely pre-task, familiarization, production, evaluation and post-task. The pre-task activity happens before the student embarks on the learning process (Figure 3).

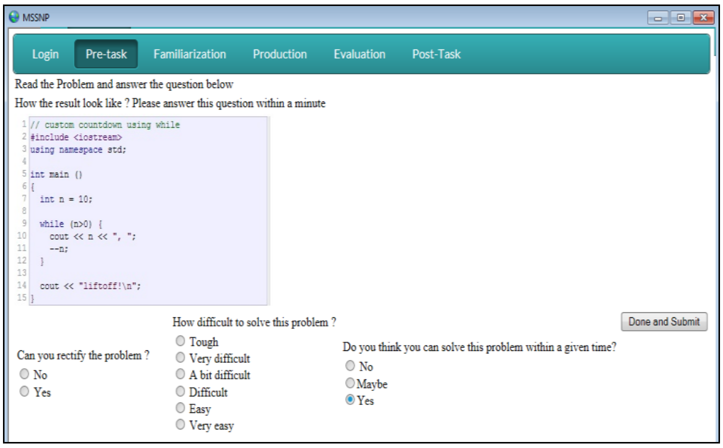

Figure 1. Pre-task stage

In the familiarization activity, a list of possible strategies is presented to the students and they are required to select the appropriate strategies in order to solve a given problem. The strategies are divided into three types, specifically, monitoring strategies, understanding, controlling errors strategies and revising strategies.

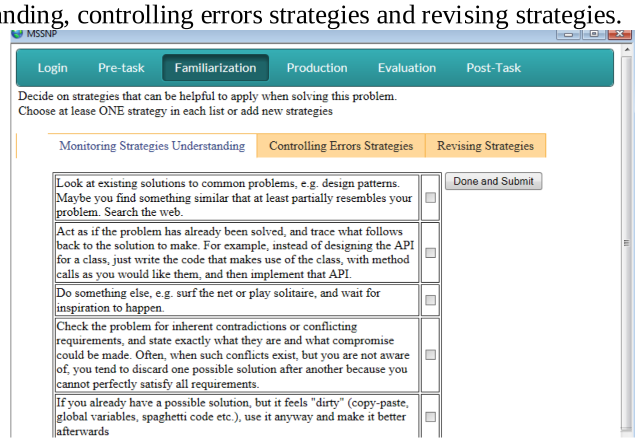

Figure 2. Familiarization Stage

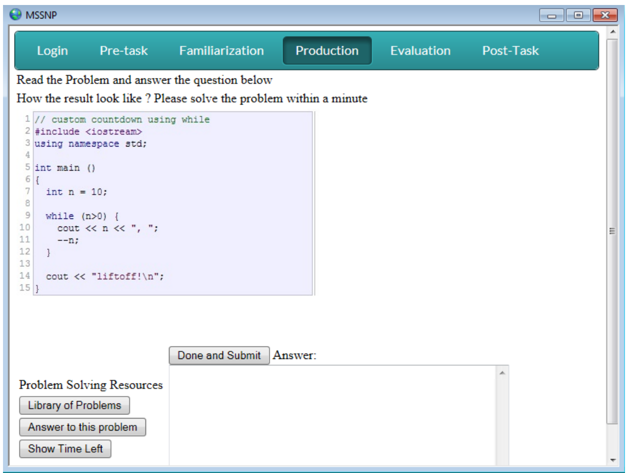

Figure 3. Production Stage

During the production stage, novices are provided with a library of similar problems, where they can view the technique to solve similar problems. The objective is to provide the students with clues and ideas concerning how to solve the given task. Students are also allowed to view the answer of the problem and monitor the remaining time left to complete the task given. Once students are done with a given task, they can submit the answer by clicking the button labeled "Done and Submit". The system will automatically check the answer.

Figure 4. Post-Task Stage

In this stage, the students can review their performance and their most recent experience through the provision of a summary of activities, and will be able to explore what happened during the problem solving activity. The outcome of this stage is to assist the student to identify mistakes related to the problem, the resources used and the issues relating to time management.

5. METHOD

SUMI is an internationally proven method and well tested instrument for determining the software quality from the end user. As already mentioned in the previous section, the SUMI questionnaire includes 50 items to represent the efficiency, affects, helpfulness and learnability attributes of the software. In a normal distribution of SUMI, the average for the global value is 50, which means that definition values beyond this value give an indication that the user satisfaction is higher than average. The statements presented to the users are about their views and behaviour towards the tested software.

5.1 Procedure and Observation

The usability testing of MSSNP was done in the computer-lab room with each participant being provided with a computer. Ten users who were invited via email agreed to take part in the study. The email stated the details of the testing session information (i.e. time, venue and the objective). The experiment was conducted and administered in the computer-lab room. A verbal instruction was given to the respondents at the beginning of the session to present the SUMI method and to help them understand the objective of the evaluation test. They were also required to become well acquainted with the MSSNP environment. The respondents were not given any instruction concerning how to operate the MSSNP. An average of five minutes was given to the participants to try out the system. They were assisted by the experimenters in respect of any difficulties they encountered with the questionnaire. At the end of the test session, the participants were asked to complete the questionnaire. The questionnaire form was divided into three sections consisting of general information, evaluation criteria (SUMI questions) and suggestion or comments. The general information section provided the information about the background of respondents in terms of gender, age and level of study (Diploma or Bachelor Degree), the second section focused on the evaluation criteria of the MSSNP based on the SUMI attributes (affect, efficiency, helpfulness, learnability) and the last section consisted of one open-ended question on suggestions or comments. The sessions lasted about 20 minutes.

6. RESULT

The overall scores for the various SUMI attributes are presented in Table 1. In general, the results indicate that user satisfaction with the system is encouraging in that the rating score was better than average and within the desired range of

40 to 60. The "global" in the SUMI scale is used as the benchmark for determining the overall judgement of usability. As shown in Table 1, the global score is 60, which exceeds the benchmark score.

Table 1. SUMI scores MSSNP

Attr.	Global	Efficiency	Affect	Control	Helpfulness	Learnability
Mean	59	62	60	59	57	59

Table 2 presents the individual score rated by 10 users for various SUMI subscales. Based on the usability ratings, as shown in Table 2, the lowest score was for the "Helpfulness" attribute, which is 56. This might be due to the lack of help facilities and documentation for the software. Overall, the results indicate the desired range of 40 to 60, which shows that users are satisfied with the MSSNP as a supporting tool for learning introductory computer programming. In order to improve the score rating, the design has to emphasize all the attributes of SUMI, especially the "Helpfulness" and "Learnability" attributes.

Table 2. Scores of SUMI subscales per user

User	Global	Efficiency	Affect	Control	Helpfulness	Learnability
User 1	63	67	68	61	57	60
User 2	55	54	58	58	52	51
User 3	59	59	52	52	60	70
User 4	61	68	61	65	49	61
User 5	60	67	57	57	68	51
User 6	58	51	65	49	67	57
User 7	60	65	60	60	51	65
User 8	64	70	56	67	65	60
User 9	58	71	57	61	40	62
User 10	58	51	65	59	59	57

Besides SUMI, as part of the usability test, an open -ended question of comments or suggestions for further improvements to the MSSNP presents varied feedback from the participants:

- It would be nice if the MSSNP provides a forum or platform to enable students and lecturers to communicate and discuss the programming method, strategies and planning, etc.
- I think the MSSNP is a good system to make learners aware and apply their metacognitive skills appropriately in learning computer programming
- From my point of view, the MSSNP can be improved in terms of its look and feel, such as the layout, color as well as the behavior of the dynamic elements, such as menu and buttons.
- It would be better if MSSNP can be integrated with the compiler or syntax error checking library so that students could post their programming code to check syntax error and the system could provide recommendations or advice concerning how to fix the problems.
- A search engine is something that the MSSNP should consider as a help support feature

Out of ten participants, four participants gave suggestions for further improvement of the MSSNP in terms of the functionality and the design aspects. One participant felt that the MSSNP would help novices to be aware of their metacognitive skills and apply it appropriately in learning computer programming.

7. CONCLUSION

The usability approach of SUMI on the Metacognitive Support System for Novice Programmers (MSSNP) presented in this paper for evaluating the affect, efficiency, control, helpfulness and learnability is plausible. The semantic web, which is the underlying technology of the MSSNP, provides a common framework that allows data to be reused across the applications of an enterprise. The findings gave beneficial information for the early detection of usability flaws of the system before deployment. All the attributes of SUMI show a positive result from which it may be concluded that the MSSNP prototype is helpful as a supporting tool for learning introductory computer programming metacognitively and enabling novices to become lifelong learners. However, to address the needs of the institution, faculty and students, "blended learning" has proven to be a successful approach that combines the online training while retaining the benefit of traditional face-to-face instruction [15]. The main contribution of this research work has been identifying the kind of information that needs to be produced by the interaction of the end user with the system. It yields better insight into the cognitive mechanism underlying the observed effects.

8. REFERENCES

[1] Davids, M.R., Chikte, U., Grimmer-Somers, K., and Halperin, M.L.: 'Usability testing of a multimedia e-learning resource for electrolyte and acid-base disorders', British Journal of Educational Technology, 2014, 45, (2), pp. 367-381

[2] Davis, F.D., Bagozzi, R.P., and Warshaw, P.R.: 'User acceptance of computer technology: a comparison of two theoretical models', Management science, 1989, 35, (8), pp. 982-1003

[3] Davis, F.D.: 'User acceptance of information technology: system characteristics, user perceptions and behavioral impacts', International journal of man-machine studies, 1993, 38, (3), pp. 475-487

[4] Dumas, J.S., and Redish, J.: 'A practical guide to usability testing' (Intellect Books, 1999. 1999)

[5] FDIS, I.: '9241-11. Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs), Part 11: Guidance on Usability Specification and Measures', in Editor (Ed.)^(Eds.): 'Book 9241-11. Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs), Part 11: Guidance on Usability Specification and Measures' (Technical report, 1997, edn.), pp.

[6] Johnson, R., and Hegarty, J.R.: 'Websites as educational motivators for adults with learning disability', British Journal of Educational Technology, 2003, 34, (4), pp. 479-486

[7] Kirakowski, J., and Corbett, M.: 'Measuring user satisfaction', in Editor (Ed.)^(Eds.): 'Book Measuring user satisfaction' (Cambridge University Press, 1988, edn.), pp. 329-338

[8] Kirakowski, J., and Corbett, M.: 'SUMI: The software usability measurement inventory', British journal of educational technology, 1993, 24, (3), pp. 210-212

[9] Kirakowski, J.: 'The use of questionnaire methods for usability assessment', Unpublished manuscript. Retreived from http://sumi. ucc. ie/sumipapp. html, 1994

[10] Maheshwari, P.: 'Teaching programming paradigms and languages for qualitative learning', in Editor (Ed.)^(Eds.): 'Book Teaching programming paradigms and languages for qualitative learning' (ACM, 1997, edn.), pp. 32-39

[11] Means, B., Toyama, Y., Murphy, R., Bakia, M., and Jones, K.: 'Evaluation of Evidence-Based Practices in Online Learning: A Meta-Analysis and Review of Online Learning Studies', US Department of Education, 2009

[12] Mohd Rum, S.N and Ismail,M.A (2013), Promoting Metacognitive Learning Environment in Supporting Novice Programmer Using Semantic Web, International Conference on Emerging Trends in Computer Science & Information Technology, Kuala Lumpur ,pp45-48 , 2013

[13] Mohd Rum, S.N and Ismail,M.A Development of a Metacognitive Support System for Novice Programmers (MSSNP) Using the Semantic Web, 3rd International Conference on Computer Science and Computational Mathematics (ICCSCM), 8th 9th May 2014, Langkawi, Malaysia, pp 24 28, 2014

[14] Pearl, J.: 'Heuristics: intelligent search strategies for computer problem solving', 1984

[15] Riley, D.D.: 'Teaching problem solving in an introductory computer science class', ACM SIGCSE Bulletin, 1981, 13, (1), pp. 244-251

[16] Schraw, G., and Sperling Dennison, R.: 'Assessing metacognitive awareness', Contemporary educational psychology, 1994, 19, pp. 460-460

[17] Siti Nurulain Mohd Rum and Ismail M.A Ontology Development of Metacognitive Support System for Novice Programmers (MSSNP),International Symposium on Technology Management and Emerging Technologies (ISTMET), 27th - 29th May 2014, Bandung, Indonesia,IEEE, 2014

[18] Tobias, S., Everson, H.T., and Laitusis, V.: 'Towards a Performance Based Measure of Metacognitive Knowledge Monitoring: Relationships with Self-Reports and Behavior Ratings', 1999