A Block Programming Environment Enabling Online Peer Assessment for Promoting Collaborative Learning

Masanori FUKUI[a,b], Jo HAGIKURA[c], Tomoya BANSHO[c], Yuji SASAKI[d], Masakatsu KURODA[e], Jun MORIYAMA[e], & Tsukasa HIRASHIMA[a]
[a]*Graduate School of Engineering, Hiroshima University, Japan*
[b]*Japan Society for the Promotion of Science, Research Fellowship for Young Scientists, Japan*
[c]*School of Science & Technology, Kwansei Gakuin University, Japan*
[d]*Faculty of Environment & Information Studies, Keio University, Japan*
[e]*Graduate School of Education, Hyogo University of Teacher Education, Japan*
*fukui@lel.hiroshima-u.ac.jp

Abstract: This paper reports a development of a block programming environment enabling online peer assessment for promoting Collaborative Learning (CL). Many investigations about CL have indicated that peer assessment plays an important role for activation of CL. In this research, we develop a block programming environment. This environment consists of three systems, (1) a block assembly system, (2) a browsing system, and (3) a server system. Client side systems (the block assembly system and the browsing system) has been implemented by Blockly for novice programming learners. A part of the server system has been implemented by Firebase. This environment helps novice programming learners to program through CL.

Keywords: Collaborative Learning, Peer Assessment, Automatic Evaluation of a Code

1. Introduction

Targeting novice level learners in programming education, this paper reports a development of a block programming environment enabling online peer assessment for promoting Collaborative Learning (CL). CL has attracted attention in recent years. Laal, M. & Ghodsi, S. M. (2012) pointed out that CL is a promising educational approach for teaching and learning that involves groups of learners working together to solve a problem, complete a task, or create a product. They summarized CL into four major categories: social, psychological, academic and assessment. Various practical strategies, then, have been proposed to effectively conduct CL. For example, Problem Based Learning, Project Based Learning, Knowledge Constructive Jigsaw and so on. In group peer assessment, where students grade each other is one of the most effective methods to activate CL (Kollar, I. & Fischer, F. 2009). Sitthiworachart, J & Joy, M. (2004) pointed out that activities of peer assessment are useful for programming education. Therefore, several investigations also have been considered to develop a function that allows peer assessment. However, these previous researches are targeted for programming learning as coding, that is, code writing, and are not targeted for novice programming learners. This research is targeting novice programming learners by using block programing. Block programming environments such as Scratch have been commonly used for programing education for novices. For novices, it is also important to gain a deep understanding of programming by understanding where in code is wrong and then understanding how to program. Evaluation of code, however, is very difficult in common programming languages with various expressions such as Java, C and so on. In contrast, because block programming is only using several predetermined blocks composed by markup language such as xml and json, it is possible to examine all possible combinations of the components, that is, it is possible to realize automatic evaluation of a code. Therefore, block programming is expected to be more useful than common programming language for novice programming learners. However, there are no previous researches that realized

peer assessment in a block programming environment. Moreover, automatic evaluation of a code is not implemented in block programming environment either. Therefore, it is important to develop the block programming system implemented peer assessment and the automatic evaluation of code.

Based on these backgrounds, as the first step, this paper reports a development of a block programming environment enabling online peer assessment for promoting CL.

2. Development of Block Programming Environment with Peer Assessment

This system is composed of client side two programs (first program is called "block assembly system", and the second one is called "browsing system") and a server side program (it is called "server system"). The server system allows a learner to view how other learners are programming and learn programming from various perspectives through network. Therefore, the two client systems using the browsing system can promote deeper learning by recognizing the difference with them through A's programming by block assembly system. Also, it can be expected to promote CL using this system. An overview of this environment is shown in Figure 1.
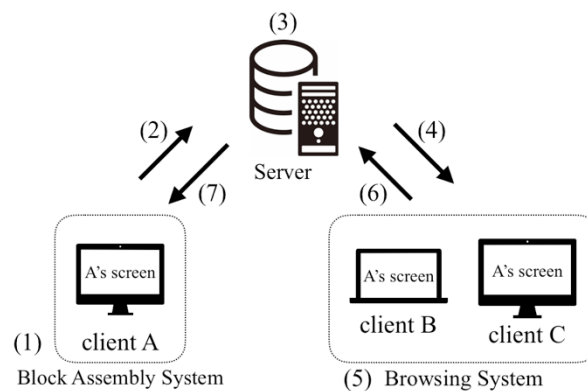

Figure 1. An Overview of This Environment

The server system is implemented based on Firebase and the browsing system to peer to peer connection by using WebSocket communication. The server system plays a role of mutually transmitting xml data of client A using the block assembly system and another client PCs using a browsing system. By the function of the server, the data coded by client A is displayed on another client PCs. The assessment results for client A's code displayed on a browsing system are sent to the block assembly system of client A. The server stores all data in Firebase Realtime Database. This allows students to see their own coding histories and feedback data on the browsing system (Those functions correspond to (3) in Figure 1.).

The block assembly system and browsing system are implemented based on Blockly. Those systems are also running on Electron. Blockly is made by Google for Education, and we customized this library. Blockly is able to make blocks for programming in the same way with Scratch etc. This library is able to customize programming blocks and create new blocks easily, and the blocks can be used on web, Android and iOS devices (Google for Education., n.d.). This library has been already used to make many block programming systems. Therefore, a student who has learned block programming such as Scratch can handle it easily and can expect to utilize it in many schools by preparing a block suitable for the stage of student's learning contents. Electron is made by GitHub, and this library allows us to develop a cross platform desktop application with JavaScript, HTML, and CSS (GitHub., n.d.). Electron application is cross platform, so it can be running on various Environment such as Macintosh, Windows etc. without installing application. Electron also has a built-in browser Chromium inside, and it is possible to avoid the problem that the display differs depending on the browser. Therefore, it is possible to run this system on many classrooms.

The block assembly system is able to make a program using blocks, and send the code, that is, assembled blocks to server system when send button is clicked at block assembly system ((1) in Figure 1.). After programming using the block assembly system, this system converts the results to xml, and then send the xml description to the server ((2) in Figure 1.). When the server sends

feedback data to the block assembly system, the data is displayed on it ((1,7) in Figure 1.). On the other hand, the browsing system receives the xml data transmitted from the block assembly system via the server system ((2,3,4) in Figure 1.), converts the xml data into block representation, and displays it on the screen ((5) in Figure 1.). Then, feedback on the data of the block is performed, and when the send button is clicked, any feedback data is transmitted to the server ((5,6) in Figure 1.). A prototype of block assembly system is shown in Figure 2. This system has two areas. First, the left side is an area where programming is performed, and the second, the right side is an area displaying the evaluation from browsing systems. Next, a prototype of the browsing system is shown in Figure 3. The browsing system has also two areas. The difference points from the block assembly system is that the browsing system has an area where programming results from the block assembly system is only displayed, and an area where evaluation is performed. Also, the browsing system is not able to program, but able to browse peer's codes assembled on the block assembly system. A student is able to evaluate peer's codes by the above function. The browsing system's screen is almost same with the block assembly system's one. This is, if one system can be used by a learner, other systems can also be used. Therefore, we expect that this system makes novice programming learners to use this system easily.
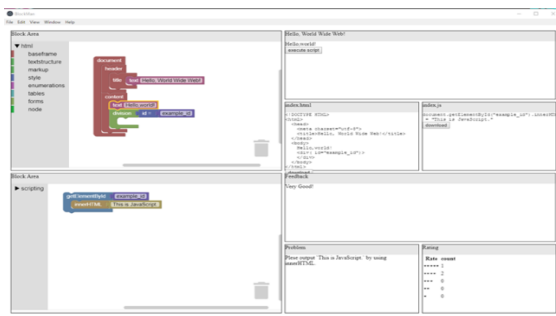

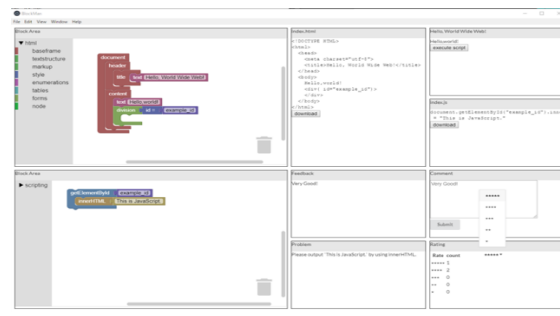*Figure 2.* A prototype of block assembly system.

Figure 3. A prototype of browsing system.

3. Conclusions and Future Works

In this research, we described the Block Programming Environment which implements the peer assessment system using Blockly and Firebase so that the novice programming learners are able to practice easily. Currently, we are developing the environment and planning evaluation through practical use. As future work, it is necessary to further improve the environment to promote CL by extending functions of peer assessment and evaluation method.

Acknowledgements

References

GitHub. (n.d.), Electron. Retrieved August 1, 2018 from https://electronjs.org

Google for Education. (n.d.), Blockly. Retrieved May 14, 2018 from https://developers.google.com/blockly/

Google Developers. (n.d.) Firebase, Retrieved August 1, 2018 from https://Firebase.google.com

Kollar,I. & Fischer,F. (2009). Peer assessment as collaborative learning: A cognitive perspective, *Learning and Instruction*, *20*(4), 344-348.

Laal, M. & Ghodsi, S. M. (2012). Benefits of collaborative learning, *Procedia - Social and Behavioral Sciences*, *31*, 486-490.

Sitthiworachart, J & Joy, M. (2004). Effective peer assessment for learning computer programming, *ACM SIGCSE Bulletin*, *36*(3), 122-126.