Facilitating A Flipped Classroom using Chatbot: A Conceptual Model

Piyanuch Tangkittipon*, Apiwat Sawatdirat,
Phoemporn Lakkhanawannakun and Chaluemwut Noyunsan

Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen Campus, Khon Kaen, 40000, Thailand

piyanuch.ch@rmuti.ac.th*

Abstract. *The flipped classroom concept is one of the effective blended learning models. It utilized both in-class hours and out-of-class hours in the sense that all students are expected to prepare for the classroom in advance. As the student behavioral engagements can be collected by using learning management system, the course report revealed that some students did not prepare for class. At this point, how to boost student's engagements and accomplishments in the flipped classroom is quite a severe issue for the instructors. Even though the eLearning system is twenty-four seven accessible and social tools can support the sharing of ideas between class members, students' engagement did not rise expectedly. The purpose of this paper is to present a conceptual model for the computer programming flipped classroom. This paper also explores the features required of establishing virtual teaching assistance for enhancing the computer programming course. The suggested approach is to implement a chatbot integrating to the blended classroom environment due to increasing motivation among flipped class students is worthwhile in the long term.*

Received by	13 August 2019
Revised by	16 May 2020
Accepted by	18 June 2020

Keywords:

chatbot, flipped classroom, student engagement

1. Introduction

The instructional approach in the 31-407-100-101Computer Programming course has been continually improving. Not only the reason that it is a compulsory course which all Engineering freshman at the Rajamangala University of Technology Isan Khon Kaen Campus are required to complete, but It is also the very beginning course to develop the ICT literacy skills for the 21st-century engineering students. The instructors learned that there were varies ICT skill levels among each different sections of computer students from semester to semester; there is no one-size-fits-all teaching approach for this course. The formal Learning Management System (LMS) was adopted to create blended learning more than ten years ago, following with making an Automated Programming Assessment Systems (APAS) called CLAB [1] it still cannot persuade the Computer Programming students to engage with the course more. Recently, the Flipped classroom approach has been implementing in this course. So far, it seems to work, but this practice created a new issue to be improved on.

2. The Flipped Classroom Concept

For decades, educators and educational researchers have been putting tremendous effort into improving the pedagogies, finding to the superlative suitable approach of teaching. The flipped classroom is a pedagogical model which has gained plenty of attention in recent years. Bishop and Verleger, 2013 described in [2] that the flipped (or reversed) classroom approach is a student-centered learning approach, wherein the traditional instruction is reversed. The students:

(1) Exposed to the class material content outside of the classroom.

(2) Provided with out of class online chances to interact and discuss the material with classmates.

(3) Utilized the inside classroom time to comprehend the knowledge through discussions, problem-based learning, and presentations.

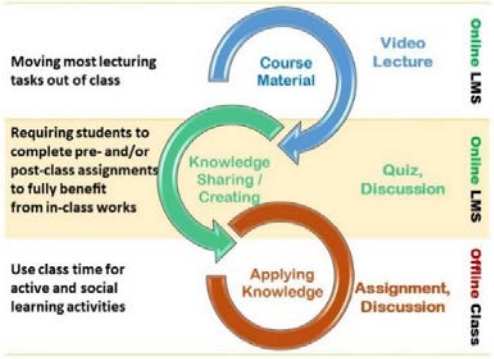

Fig. 1: The flipped classroom concept (derived from [2-5])

Lately, the flipped classroom became quite a buzzword in engineering and education; many models have been adopted. Though the success of the approach might depend on the practice and cultural background of the students, hence, the appropriate implementation process for the different area and culture might not be the same. In some quarters of the education sector, the wisdom of using student and instructor time for face-to-face live lectures is questionable, and complacency might set in with some student who might lead to under-performance of the student in their respective courses.

3. The Flipped Computer Programming Course

The theoretical fundamentals used for justifying what is the flipped classroom typically focus on the reasons for not using classroom time for delivering lectures as well as the approach that has been using in the 31-407-100-101 Computer Programming course. This course has two lecture credits and one laboratory credit, which embodies five hours in class and five hours outside of class each week of the 16-week-long semester. Alongside facing problems regarding the hefty instructor workload due to a vast number of students enrolled each semester, the varied ICT skill level of the students also caused time-consuming delays in every session. The flipped classroom was chosen as a strategy to support to meet the expectations of students while optimizing teaching and classroom resources. The blended learning approach of the flipped could also be leveraged for both individual courses and on the organizational level to improve instructional delivery and enhance satisfaction.

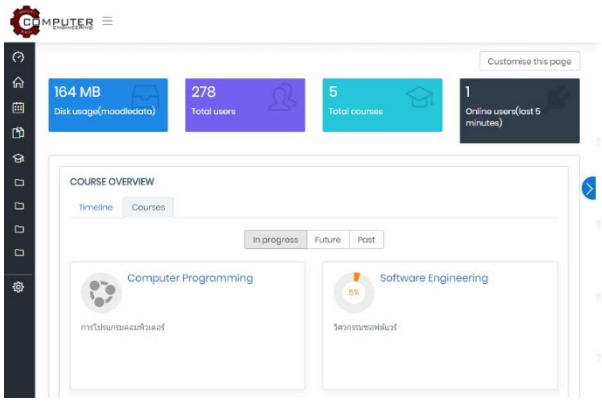

Fig. 2: The Collaborative Dashboard of the recent LMS adapting for the flipped classroom approach

Since the first semester of 2016, the micro video lectures of all units were recorded then uploaded to Faculty's LMS along with other class materials. Since then, the course was flipped, the video lectures have been narrated together with slide-based presentations. The contents published on eLearning course every week. All students were assigned to take a pretest and prepare before the next class. They can watch or rewind the video lecture if they need it. The Computer Programming class time changed into a new structure such that the instructor started with delivering a summary of the important thing points and let the class discuss, after that the students can self-work on CLAB programming exercises.

Throughout three academic years, the lack-of-engagement problem found, so the instructor facilitated the class with the adoption of Social Media Tools.

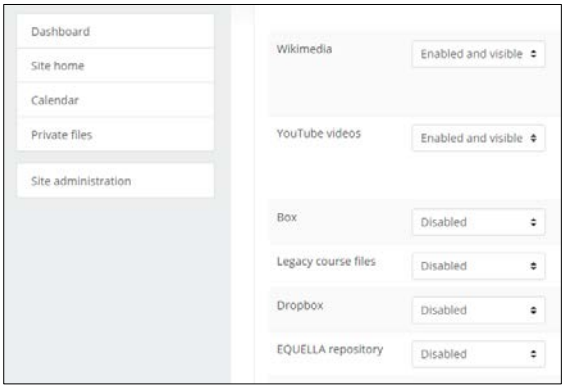

Fig. 3: The social media tools integration

Even the LMS report showed the increase of the behavioral engagement from students in every Learning Objects (LOs); nevertheless, some students designedly skip the material LOs, which they think it is not mandatory. The example of the behavioral engagement of the Computer Programming LOs showed on Fig. 4.

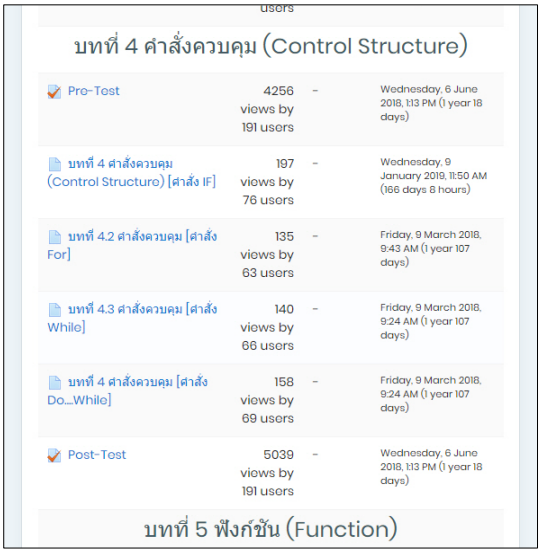

Fig. 4: The example of LOs interactions report

4. Feedback from Students

To propose the Conceptual Model Regarding filling the gap on the new flipped class issue, the instructor firstly surveyed with the requirements of students in order to determine current needs, alluring qualities and attributes of the innovative system. After that observed and took the semi-structured interviews with focus group.

No.	The Reason for Skipping the LOs
1.	Self-understanding the technical content is too difficult
2.	Lack of motivation waiting for instructor's or classmate's explanation
3.	Course assignments workload is correspondingly heavy

Table 1 The top categories of students' feedback

The motivation for preparing before class might increase if the system provides some more helpful. When the student is struggling, they needed advice immediately, but they have to wait for the instructor or classmates reply. The survey result revealed that 83% of frequent questions is basic knowledge, which can increase more instructor workloads while keep answering similar questions like a routine. The sample questions are demonstrated below:

- How can I config the CLAB class path?

- Where can I resend the assignment to increase the score?

- I cannot login. What should I do?

Additionally, some students felt comfortable and can continue the before-class preparation when his or her senior offered technical help through the social media chat platform such as Facebook Messenger or Line. A literature review study revealed that automated conversation could support both students and instructors in the role of teaching assistants.

5. Chatbot Concept

Chatbots are the automated answering programs that are used in various applications for assisting the answering process. These bots are developed through a machine learning algorithm and are used for simple to complex operations. Chatbots have evolved from simple answering tools to complex programs which learn through experiences. These applications replied in an automated format in the real-time and formed the basis of the modern chatbots. Initially, chatbots only replied to specific questions with a specific format and were used in the customer care related tasks.

5.1 Chatbots in formal education

After a period of time, they have been evolved in such a way that they learn from experience to reply to complex problems. Now, they are used apart from chat purposes, for web related applications and other artificial intelligence-related programs.[6]

The most initial use of chatbots dates back to the time when ELIZA and instant messenger by AOL. Since then, applications such as ZoeOnAOL, SmarterChild, Yahoo chat rooms, and MSN, have been created that uses a smart algorithm to improve the user experience. As, ZoeOnAOL used the keywords of users to search from various search engines and provided the results in the chat box, such that user does not have to leave the chat room for the search of words.

There are two basic models upon which chatbots are developed: retrieved based model and generative model [7]. Retrieved based models use a tree structure in which heuristic imitation is used to respond in a manner of human memory. They use predefined databases with .Net or JAVA through which decision trees are created accordingly. As they typically use predefined answers so that the model can get stuck in case, an answer is not available for a particular question. Generative models, on the other hand, are much more advanced as they do not rely on predefined databases. They use machine translation in which output is used to create the input. They use deep learning technology to use the previously recorded answers for the development of new responses. As the response is self-created through prior learning, so it is possible to have grammatical errors. Generative models are being modified for complex and more efficient use of chatbots in modern applications for the development of smart uses. They are now used in education and other daily routine applications like autocorrects, predictive texts, alarms and calendars in chats, and so on. Similarly, Facebook uses it for search engine optimization while other applications now use it for optimized and more related user experience in the applications. All things considered, chatbots in education have moderately a few uses and still have more possible features. The AIML-based chatbot can be both simple and complex to implement, all depending on the effort placed into the implementation process. The tool is assorted and might be used for many different purposes, the only limitation being the creators' creativity and imagination [8]

5.2 A chatbot conceptual framework

The potential utilization and acceptance of the chatbot technology have caused in leading technology hucksters. [9] outlines of a raised conceptual outline for completing flexible chatbots endowed upon agent-oriented abstractions, which are goals, plans, and commitments.

The progressive growing in attractiveness of chatbots as part of the virtual assistants through a conversational interface has led to many enterprises releasing the If-This-Then-That frameworks to mastermind the effective chatbots. Still, these frameworks frequently result in inflexible and difficult-to-maintain chatbots.

The potential applications and renown of chatbots innovation have induced about driving innovation marketers, for example, Google, IBM, Facebook, Microsoft, and so many more to discharging structures to fabricate such chatbots. Focusing on 21st Century Skills development, one significant application area for chatbots is information technology (IT), so Computer Programming is the appropriate course to be applied.

Typically, the chatbot model works like this; a dialogue manager communicates with the user, the knowledge base then interactive engine in the background gives relevant results. The dialogue management obtains meaning from the user's words or sentences. Inference engine receives information from the dialogue manager and then inference it with the knowledge base available. The chatbots may infer so many entities during the process for mapping them to relevant actions. Knowledgebase here is the central database of chatbots in which all the information regarding the system and the users' replies, entities and their relationships. Through the generative process, this

knowledge base gets widened over time based on information that is received through the user's replies and actions. Before-mentioned information is stored in the knowledge base, which is then used in the future to reply according to the scenario.

In conclusion, the planner prepares the execution process that needs to be applied to the required task. The effectiveness of a dialogue manager and the planner depends upon the complexity and smartness of the machine learning algorithm. The more optimized the model would be the capability of chatbots abilities. Consequently, it is preferable to adopt machine learning and artificial intelligence (AI) for chatbot invention.

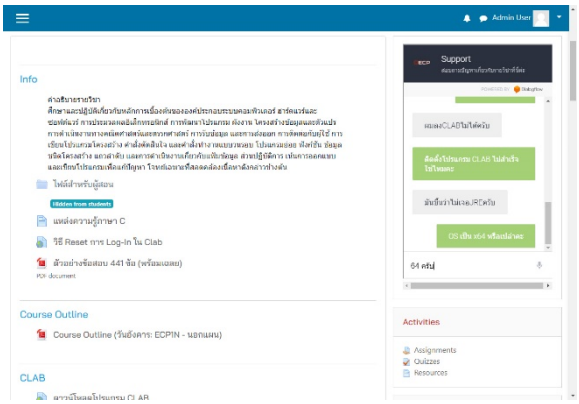

Fig. 5: The new system integrating vTA-chatbot prototype

6. Chatbot Role and Features for Computer Programming Course

The original idea with the initial chatbot was to be able to establish an automated help to students while they are preparing before-class, boost their motivation to meet the course competencies and increasing the behavioral engagement to all Learning Objects (LOs).

The instructors created a prototype of a chatbot using Dialogflow with Firebase Cloud Functions, integrated to the course main dashboard then commenced observation. Also, the focus group discussion and individual interview conducted.

The expected Role of the chatbot is to be virtual teaching assistants (vTA). The required features, according to the exploratory research result, are that the chatbot should be able to:

- help the students 24/7
- answer student's basic questions
- ask the student related questions
- hold a general conversation
- provide the LOs guide
- recommend additional learning material
- provide the contact with the instructor

.

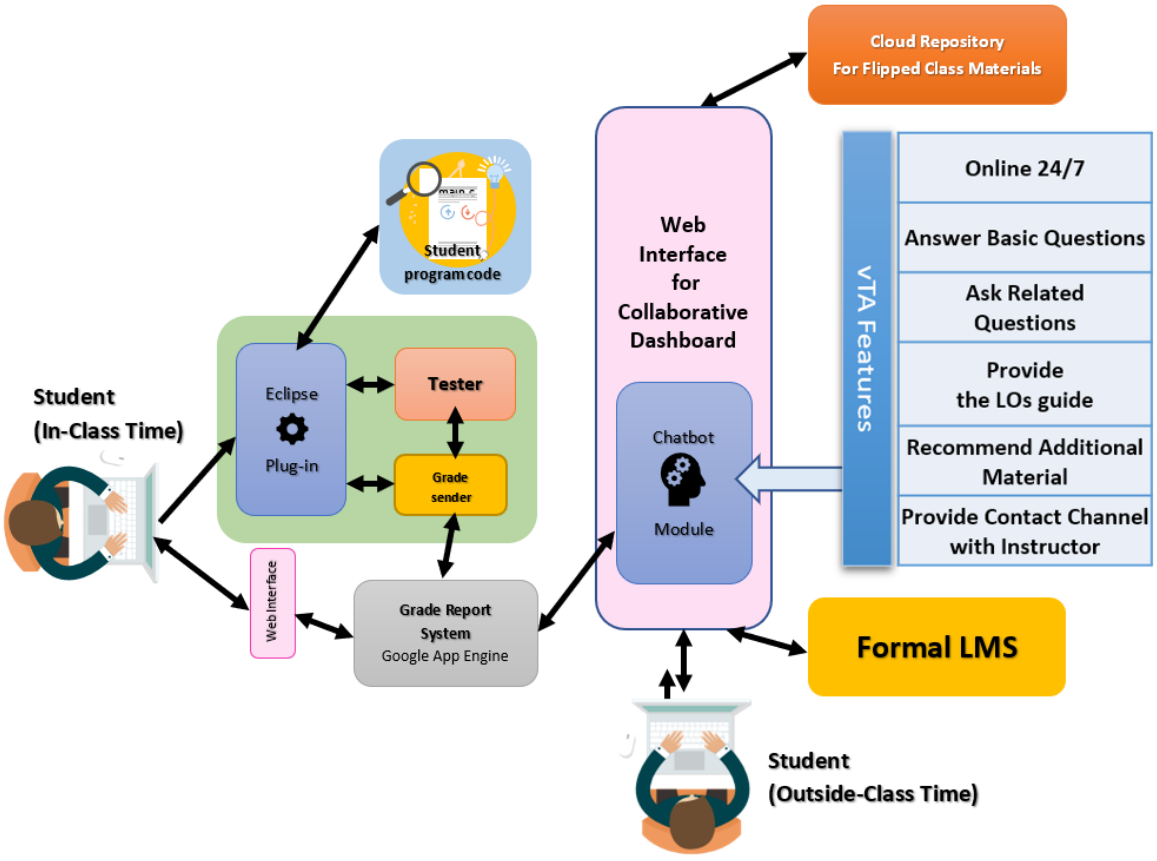

Fig. 6: A Conceptual Model for Computer Programming Flipped Classroom integrating vTA Chatbot

All features above conveyed from the indirect student feedback while the flipped class process was going routinely.

7. Conclusions

This study presented an overview of how to enhance the student's engagement in the 31-407-100-101 Computer Programming course. In order to determine the need of the students, focus group interviews were conducted to gather student feedback regarding their behaviors and flipped classroom engagement. The result showed that students were sometimes struggling with content in the micro-video lecture. They indisputably needed for advice immediately before continuing but have to wait for instructor or classmates delayed-replying; consequently, they skip preparing for the class. The suggested approach is to implement virtual teaching assistants (vTA): a chatbot integrating to the flipped classroom environment (Fig.6) with the intention of helping increase motivation among students and focusing on how to encourage them to be more engaged with all type of Learning Objects

8. Discussion

The possible ways to increase the level of students' engagement may include more dimensions; however, this study has found that the majority of flipped classroom implementation is somewhat specific to how to support the students while they are out of class. The future work can also focus on hands-on usability issues concerning the side effect of the technologies that may distract students while they are in the formal class.

References

[1] Noyunsan, C., Tangkittipon, P., & Ngamwittanon, N. (2014). Online Computer Programming Grading Report System using Eclipse Plug-In, paper presented in The 6th International Conference on Science, Technology and Innovation for Sustainable Well-Being (STISWB VI), pp. 229-234. Apsara Angkor Resort, Cambodia.

[2] Bishop, J. L., & Verleger, M. A. (2013). The flipped classroom: A survey of the research, paper presented in ASEE national conference proceedings, 30(9), pp. 1-18, Atlanta, GA

[3] Diwanji, P., Hinkelmann, K., & Witschel, H. F. (2018). Enhance Classroom Preparation for Flipped Classroom using AI and Analytics, paper presented in the 20th International Conference on Enterprise Information Systems (ICEIS), pp. 477-483, Funchal, Medeira-Portugal.

[4] Wang, F. H. (2017) 'An exploration of online behaviour engagement and achievement in flipped classroom supported by learning management system, Computers & Education. Pergamon, 114, pp. 79–91.

[5] Caligaris, M., Rodríguez, G. and Laugero, L. (2016). A First Experience of Flipped Classroom in Numerical Analysis, Procedia - Social and Behavioral Sciences, vol. 217, pp. 838–845.

[6] Molnár, G., & Szüts, Z. (2018, September). The Role of Chatbots in Formal Education, paper presented in The 16th International Symposium on Intelligent Systems and Informatics (SISY), pp. 197-202, Serbia.

[7] Babar, Z., Lapouchnian, A., & Yu, E. (2011). Chatbot Design-Reasoning about design options using i* and process architecture. paper presented in The Proceedings of the 10th International i* Workshop, pp. 12-13, Washington, DC, USA.

[8] Roos, S. (2018). Chatbots in education: A passing trend or a valuable pedagogical tool?, Master thesis in Information Systems, Uppsala University, Sweden.

[9] Telang, P. R., Kalia, A. K., Vukovic, M., Pandita, R., & Singh, M. P. (2018). A Conceptual Framework for Engineering Chatbots. IEEE Internet Computing, 22(6), 54-59.

Biographies

Piyanuch Tangkittipon hold a PhD degree from Assumption University, Thailand. She received her B.S. and M.S. degrees from Khon Kaen University, Thailand, in 2006 and 2008. She has been at Rajamangala University of Technology Isan, Khon Kaen campus, Thailand, since 2007 as a Lecturer at Department of Computer Engineering. Her primary research interests include software engineering, software quality assurance and virtual learning environments.

Apiwat Sawatdirat has been a lecturer in computer engineering, Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen campus, Thailand. He received his M.Eng. in Electronics Engineering and B.Eng. in Computer Engineering from Rajamangala University of Technology Thanyaburi, Thailand.

Phoemporn Lakkhanawannakun received the B.S. degree in Computer Science, the M.S. degree in Information Technology, and Ph.D. degree in Computer Science from Khon Kaen University, Thailand, in 2005, 2008, and 2016, respectively. In 2016, she joined the Department of Computer Engineering, Faculty of Engineering, Rajamangala University of Technology Khon Kaen Campus, as a lecturer. Her current research interests include Data Mining and Machine Learning.

Chaluemwut Noyunsan was born in Khon Kaen, Thailand, in 1981. He received the B.S. degree in computer engineering from Khon Kaen University, Thailand, in 2003, and the M.S. degrees in computer science from Chulalongkorn University, Thailand, in 2009, respectively. He received PhD degree from the Department of Computer Engineering, Khon Kaen University. His current research interests include social network analysis and information credibility measurement.