A Brief Review of using LMS in Computer Science Learning

Fadoua Koucham, Younès El Bouzekri El Idrissi and Ayoub Ait Lahcen
Engineering sciences laboratory, National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco

Keywords: E-learning, Computer Science Learning, LMS, Intelligent Tutoring System

Abstract: The world of technology is expanding at a breakneck pace, and the ability of humans to learn and progress has a significant impact on attaining a high degree of knowledge. Therefore, humans seek to change and value the systematic tools used by having a good learning acquisition. Thus far, various strategies have been utilized in the educational system to facilitate and maintain a strong knowledge of student learning behaviour. Especially in the e-learning system, which appreciates considerable importance to the student community and has been the subject of an extensive investigation by developers and researchers to make it more resilient, accessible, and motivating for students to learn. In light of this, this paper aims to critically review different approaches to implementing the development of the LMS system for computer science (CS) students.

1 INTRODUCTION

In the history of the development education system, e-learning has been critically impacting the improvement of student knowledge by getting courses in the distance. Thus, some institutes adopt blended learning (Patel et al., 2013), which combines face-to-face with distance learning, to offer the student the most ability to comprehend the subject.

Therefore, one of the most popular systems used in e-learning education is the Learning Management System (LMS): a software application responsible for all learning areas for instructors and students; it offers essential tools such as downloading courses, submit and return assignments to acquire the theoretical concept of the subject (Yulianto et al., 2016) also it provides directly the creation and delivery of content by the teachers, as well as the monitoring and evaluation of student engagement and performance. Such as Moodle, the widespread LMS use by most education institutes that adopt distance learning, is open-source and free. Still, unfortunately, this type of system not offering the student the interactive part to practice the new concept learned (Al-Khanjari and Al-Roshdi, 2015).

This issue considers a gap for student learning, who have the habit of getting the basic concept of subject and practising it to have a good comprehension of the course, and this impact all subject mainly a computer science student which they must have a practising part, considering a computer science must adopt three basic steps to acquire knowledge: cognitive, constructive, and socially situated learning(Yulianto et al., 2016).

- Cognitive: Understanding different theoretical programming languages with prerequire knowledge to facilitate the comprehension of the new concept (Yulianto et al., 2016).
- Constructive: Making an actual application to improve the excellent learning theoretical programming language (Yulianto et al., 2016).
- Socially situated learning: Considering as social interaction, by using different online tools such as forum discussion, email, social networks, or even a real-life face-to-face sharing knowledge, which allows exchanging cognition to solve a problem and lead to improving knowledge (Muhisn et al., 2019).

This paper surveys recent studies on the employment of learning management systems in computer science learning. The first section of this paper will examine the literature review by introducing two different ways of ensuring education for computer science students using the standard and the adaptive LMS, then a conclusion with a mention of our future research goal.

Koucham, F., El Bouzekri El Idrissi, Y. and Ait Lahcen, A.
A Brief Review of using LMS in Computer Science Learning.
DOI: 10.5220/0010730900003101
In *Proceedings of the 2nd International Conference on Big Data, Modelling and Machine Learning (BML 2021)*, pages 196-200
ISBN: 978-989-758-559-3

2 LITERATURE REVIEW

More recent attention has focused on providing the perfect approach of implementing a learning system that improves online education for a computer science student.

It considered many different methods to transform traditional LMS learning into a system that gives a student interaction possible, powerful, and personalized.

2.1 Standard Learning Management System in Computer Science Learning

Researchers (Al-Khanjari and Al-Roshdi, 2015) have introduced two approaches implemented to the standard LMS to ensure practical usage. So that the students would be able to solve online exercise problems, watch a video course, compile code, and display the results into different CS courses; the study is based on Moodle LMS since a large community use it, namely:

- Connect extern tools, an existing application, or a new application developed by any language with the LMS system by a button link integrated inside the platform, facilitating users to access the extern desired tools inside Moodle. This approach is realizable by a technology combined with Moodle, the learning tool Interoperability LTI.

  This approach has multiple advantages, such as reducing the time of execution of the platform since different manipulation will be outside, and the disadvantage of this method is limited functionality.
- Make an SOA communication between differents tools developed as service (SAAS), registering into the service registry UDDI, and implemented into Moodle using LTI technology.

Furthermore, motivation is an essential aspect of improving the active engagement of students learning a programming language into standard LMS in a computer science study.

Thus, an early example of research into (Almeida et al., 2017) has implemented a new methodology of learning programming using a robot. The main idea is to make an active part of controlling a robot in the distance; by executing code inside the LMS system, moving it from one destination to another. The students can access the robotic experiments according to their scheduling time on a specific day; the program offers five challenges—the robot has been situating in the laboratory, connected to the server by Bluetooth. A camera associated with a server by IP address shows the robot's movement in the LMS interface. When the user sends the code to the server, the robot agent looks for the programming code and generates the execution file using the NXT-Python program. If there is no error, the running program controls the robot remotely.

Another way to improve student motivation is using the Multimedia Educational Resource for Learning and Online Teaching concept (MERLOT). An online repository that englobes digital resources; learning objects (LOs): like images, tutorials, simulation, quiz, test, presentation and all other tools helping to create a complete unit of learning regroups by categories, and which allow the registered user to rats in each LOs, and making a private collection. That aid improves the MERLOT repository by the user and has visualization into digital resources that have less interaction by the student, which will help the organization make it enhance in the future (Alharbi et al., 2011). The MERLOT concept can easily combine with the Open Educational Resources (OER) projects such as LMS (Gunarathne et al., 2020).

2.2 Personalized Content in Learning Management System

A new approach proposed by (Thamarai Selvi and Panneerselvam, 2012) was integrating tools and material into the LMS system that improves self-regulated learning within the student's knowledge level, based on adaptive learning object (ALOB), that allows generating personalized content.

The implementation of this approach focused on learning the C language. The main idea is to classify the course into six-level; each part provides appropriate content based on the student's level, relying on different student information collected.

The architecture focus on creating multiple web service-oriented within (SOAP) protocols. Each web service has a mean objective such as:

- Self-monitoring: Register the accomplished courses and the time spent at each level.
- Self-assessment: Render the teacher questions to the student and save the return response.
- Feedback: Automatically proposed feedback to the student, which reduces the teacher effort, based on the response assessment and the registration of activities.

2.3 Learning Management System with Intelligent Tutoring System

2.3.1 Standard Intelligent Tutoring System

Among the outgrowth technic that has been evolving of the personalized learning system, since the appearance of Artificial Intelligence (AI) in the educational domain in the 1970s, is the birth of a new concept, the Intelligent Tutoring System (ITS)(Almasri et al., 2019).

Intelligent tutoring systems are computer programs that combine three crucial disciplines: education or theorists discipline, intelligent artificial, and psychology or cognition discipline. The ITS strives to provide learners with instant and personalized education or feedback, usually without the need for the teacher's intervention. All the procedurals began in the 1950s with developing the first model of this system known as Computer-Aided Instruction. The fundamental phase to be recognized in systems has been the teacher's experience, memory in a pre-stored structure element named frames, produced by the specialist instructor and exposed to the student following specific conditions. But such absolute representation of cognition has since been acknowledged as ineffectual; they could not afford valuable feedback or individualization. It led in the 1970s to combine Computer-Aided Instruction with Artificial Intelligence and consider different information about a student represented as a psychologist discipline (Nwana, 1990).

There was a significant accord in the literature that ITSs consists of at least four primary models: knowledge model, also known as a domain or expert model, pedagogical model, student model and user interface model (Almasri et al., 2019), Each of those models has a meaningful role and implementation.

- Knowledge model: Acts as the reference of knowledge to present to the student (Nwana, 1990).
- Student model: It consists of a dynamic display of student's knowledge and abilities (Nwana, 1990).
- Pedagogical model: It strictly links to the student model, deciding which educational activities to offer based on information about the student and its tutorial intent structure (Nwana, 1990).
- User interface model: Allow interaction between ITS system and the student (Almasri et al., 2019).

Despite the various changes and updates of ITSs architecture, they act in the same way. According to (Santos and Jorge, 2013), ITS design support two primary task loops: The inner and outer loop:

- The inner loop assures the student's assistance in the process of direct solving problems by proposing hints and providing rapid and automatization of the adaptive feedback, which lead a positive satisfaction returning in the interactive learning system (Riaz et al., 2019), through evaluating student competence registered on the student model.
- The outer loop executes the task that the student should practice for the next step; the decision made is according to the student's knowledge.

The most important responsibility of an ITS was to assist a student in the problem-solving process. The required knowledge has expected to learn outside of the system. With the advancement of computer capabilities, many ITS developers have found it feasible to combine an ITS with instructional content in an electronic format.

According to (Alkhatlan and Kalita, 2018), Several influential and successful ITSs realised in recent years, including:

- Affective Tutoring Systems (ATS): ATS are the ITSs that are capable of recognizing human emotions in real-time. These devices are responsible for identifying physical signals such as facial pictures, sounds, heart speed, pressure and pressure levels. However, recent learning theories have proved a connection between emotions and education, claiming that knowledge, sentiment and motivation, are the three components of education.
- Cultural Awareness in Education: It considered the various changes applied to the ITS in the optics of learning technologies systems thinking existence of multiple cultures.
- Game-based Tutoring Systems: Students play an instructional game that successfully mixes gaming methods with curriculum-based materials rather than learning a subject traditionally. Students learn better when they are having fun and are involved in the learning process.
- Adaptive Intelligent Web Based Educational System (AIWBES): It is advanced research on adaptive educational hypermedia, which combines ITSs with educational hypermedia.
- Data mining: Data mining is the process of examining large volumes of data to extract and identify valuable information. Data mining integrate into ITSs for various goals, such as catching student effects and automatically determining a partial problem space from logged user interactions.

- Collaborative Learning: They incorporate collaborative learning into tutoring systems to demonstrate the benefits of student engagement during problem-solving. So that students could study in groups by asking questions, clarifying and defending their opinions, and presenting their information.
- Authoring Tools: ITS research organisations have been interested in creating ITSs easier for designers and teachers by making ITS authorship more accessible. Authoring tools in the ITS area is classifying into several categories, including the required programming skills and those that do not, as well as pedagogy-oriented and performance-oriented tools.

2.3.2 Intelligent Tutoring System Architecture in Computer Science Learning

Implementing the ITS approach to assure adaptive learning in computer science significantly impacts analyzing learning within a student. Thus, ITS architecture has known several updates to solve different needs indoors to improve the computer science study. Namely, (Wang et al., 2020) has proposed a novel approach to help students generate a good response program over a specific problem. A classical method was to introduce the Online Judge (OJ). This system executes a response code of a problem and gives two output feedback: if the response code is correct or incorrect, it did not give more details or suggestions. Although, this approach did not help the beginner programmer student enough, which needs more override proposition within the output feedback. Thence, the novel ITS consist of integrating the following component: error repair, Frontend using VueJs Framework, Backend implementing Django Framework, Postgres database, code classification by using CLARA engine allows classifying the similar proposing response of the same problem, knowledge tracing to identify level knowledge student. It categorized into four principal parameters: two parameters according to the knowledge parameters and the two others from the performance. Therefore, the new implementations of the ITS have allowed having hinted feedback about error codes sent by students within the given problem or even a repair answer, which will help students correctly introduce a new response and acquire analyses behaviour of the programming.

Moreover, unless the availability of multiple projects that support integrating personalized learning into LMS, such as GRAPPLE and T-MAESTRO projects. Hence, they use a nonstandard external database and setup to assure adaptation (Santos and Jorge, 2013). Therefore, a researcher (Santos and Jorge, 2013) has investigated into implementing two novel concepts into ITS architecture to allow standardization and facilitation of implementation ITS into an open-source project such as OERs, personalized, and intelligent LOs, namely:

- Atomic tutors (AT): Equivalent to a small educator system, representing a specific task within a topic, which held only on performing the inner loop.
- Molecular tutors (MT): Implement the outer loop, which picks up a specific AT that should be allowed to a learner at a distinct moment by using the student model to select the appropriate task. Thus one particular MT represents a complete ITS for a specific topic.

Foremost existing researchers on ITS system for computer science focus on teaching programming languages to the students (Crow et al., 2018); by apprenticeship standard programming language based on the one-one learning such as C++, JAVA, C sharp, database, PHP web programming, while other ITSs systems have supported collaborative learning, known by the computer-supported collaborative learning (CSCL), to apprenticeship UML modelling within groups community to motivate the learning processing within a student by implementing the constraint-Based Modeling (CBM) (Almasri et al., 2019).

Although ITS support various features, which are classifying within six essential approaches: example, simulation, dialogue, program, feedback, collaboration (Nesbit et al., 2015):

- Pedagogical assistant: The character appearing on the platform is usually presented as an avatar animated, leading to show notifications. It allows the student to have cognition about different messages delivered by the ITS (Nesbit et al., 2015).
- Feedback: Proposing feedbacks depending on student response(Crow et al., 2018).
- Hints: Automatically helps generated by the ITS to the students in the processing of solving specific problems. The hint developed following to the degree of student motivation (Nesbit et al., 2015).

The lack of ITS design in computer science is the reusability of the same architecture in different ITS contexts; each design applies to one objective (Crow et al., 2018).

3 CONCLUSIONS

This brief review reveals the different ways of manipulating LMS to achieve learning for computer science students. To point out that, researchers have investigated extravagant efforts of improving distance learning based on LMS by introducing intelligent technics, to assure a high degree of apprenticeship, even for: providing courses, feedbacks, hints, according to the level of each student. That leads to enhancing student learning engagement and tying the stage of apprenticeship a new concept from one student to another to balance the elevation of learning inverse of students in the distance study.

Therefore, this research demonstrates each technique's weaknesses and effectiveness, which will help us proposing our model to ensure adaptive learning by using LMS to provide online education for computer science learners for Moroccan students.

ACKNOWLEDGEMENTS

This work was supported by the Al-Khawarizmi Program funding by Morocco's Ministry of Education, Ministry of Industry and the Digital Development Agency (ADD) under Project No. 451/2020 (Smart Learning).

REFERENCES

Al-Khanjari, Z. and Al-Roshdi, Y. (2015). Developing virtual lab to support the Computer Science Education in Moodle. In *Proceedings of 2015 12th International Conference on Remote Engineering and Virtual Instrumentation (REV)*, pages 186–191, Bangkok, Thailand. IEEE.

Alharbi, A., Henskens, F., and Hannaford, M. (2011). Computer science learning objects. In *Proceeding of the International Conference on e-Education, Entertainment and e-Management*, pages 326–328, Bali, Indonesia. IEEE.

Alkhatlan, A. and Kalita, J. (2018). Intelligent Tutoring Systems: A Comprehensive Historical Survey with Recent Developments. *arXiv:1812.09628 [cs]*. arXiv: 1812.09628.

Almasri, A., Ahmed, A., Al-Masri, N., Sultan, Y. A., Mahmoud, A. Y., Zaqout, I., Akkila, A. N., and Abu-Naser, S. S. (2019). Intelligent Tutoring Systems Survey for the Period 2000- 2018. 3(5):17.

Almeida, T. O., de M. Netto, J. F., and Rios, M. L. (2017). Remote robotics laboratory as support to teaching programming. In *2017 IEEE Frontiers in Education Conference (FIE)*, pages 1–6, Indianapolis, IN. IEEE.

Crow, T., Luxton-Reilly, A., and Wuensche, B. (2018). Intelligent tutoring systems for programming education: a systematic review. In *Proceedings of the 20th Australasian Computing Education Conference on - ACE '18*, pages 53–62, Brisbane, Queensland, Australia. ACM Press.

Gunarathne, W. K. T. M., Shih, T. K., Chootong, C., Sommool, W., and Ochirbat, A. (2020). An Automated Learning Content Classification Model for Open Education Repositories: Case of MERLOT II. *Journal of Internet Technology*, 21(5):1277–1288. Number: 5.

Muhisn, Z., Ahmad, M., Omar, M., and Muhisn, S. (2019). The Impact of Socialization on Collaborative Learning Method in E-Learning Management System (eLMS). *International Journal of Emerging Technologies in Learning (IJET)*, 14(20):137–148. Publisher: International Journal of Emerging Technology in Learning.

Nesbit, J., Liu, L., Liu, Q., and Adesope, O. (2015). Work in Progress: Intelligent Tutoring Systems in Computer Science and Software Engineering Education. In *2015 ASEE Annual Conference and Exposition Proceedings*, pages 26.1754.1–26.1754.12, Seattle, Washington. ASEE Conferences.

Nwana, H. S. (1990). Intelligent tutoring systems: an overview. *Springer*, page 27.

Patel, C., Gadhavi, M., and Patel, D. A. (2013). A survey paper on e-learning based learning management Systems (LMS). 4(6):8.

Riaz, S., Mushtaq, A., and Kaur, M. J. (2019). Intelligent Tutoring for Informed Feedback in Interactive Learning Environments. In *2019 Advances in Science and Engineering Technology International Conferences (ASET)*, pages 1–6, Dubai, United Arab Emirates. IEEE.

Santos, G. S. and Jorge, J. (2013). Interoperable Intelligent Tutoring Systems as Open Educational Resources. *IEEE Transactions on Learning Technologies*, 6(3):271–282.

Thamarai Selvi, S. and Panneerselvam, K. (2012). A Self-regulated Learning approach for programming language using cloud-based Learning Management System. In *2012 International Conference on Recent Trends in Information Technology*, pages 191–196, Chennai, Tamil Nadu, India. IEEE.

Wang, M., Wu, W., and Liang, Y. (2020). A Novel Intelligent Tutoring System For Learning Programming. In *2020 International Conference on Development and Application Systems (DAS)*, pages 162–168, Suceava, Romania. IEEE.

Yulianto, B., Prabowo, H., Kosala, R., and Hapsara, M. (2016). MOOC architecture model for computer programming courses. In *2016 International Conference on Information Management and Technology (ICIMTech)*, pages 35–40, Bandung, Indonesia. IEEE.