

LAMP: A Framework for Large-scale Addressing of Muddy Points

Rwitajit Majumdar

Inter-Disciplinary Program in Educational Technology
Indian Institute of Technology Bombay
rwitajit@iitb.ac.in

Sridhar Iyer

Department of Computer Science and Engineering
Indian Institute of Technology Bombay
sri@iitb.ac.in

Abstract— Muddy Points (MP) is a strategy to elicit and address individual students' doubts. While this can be effectively implemented in small classes, it is a challenge to do so in a large class. In this paper we propose LAMP, a framework for Large-scale Addressing of Muddy Points, as a mechanism for instructors to ensure that every individual student's doubts are addressed even in large classes. LAMP has three phases: Collection, Addressal, and Closure. In the collection phase, MPs are systematically collected through four different modes. In the addressal phase, MPs are categorized into six categories and addressed accordingly. In the closure phase, the discussions on MPs are summarized. We investigated the effectiveness of LAMP in an introductory computer science course having 450 students. We found that 68% of students confirmed they were able to pose their questions and 57% of students confirmed that there was closure to their questions.

Keywords- Instructional strategies; Large classes; Muddy points; Framework.

I. INTRODUCTION

In undergraduate engineering education, large-enrollment classes are common for introductory courses in various disciplines like physics [1], chemistry [2], biology [3], mathematics, and computer science. Instructors of such courses continuously deal with the diversity of the students that is inherent in such a class [4]. Students vary in pre-exposure to subject knowledge, approach to learning, and cultural background. To get students to be engaged with the content, instructors use various active learning strategies, such as Peer instruction [5][6], Just-in-Time-Teaching [7], and Inverted Classroom [8]. However, students with prior exposure may respond more actively to the instructor, thereby increasing the pace or the depth of the course. Those with no prior knowledge often keep quiet, even if they are not able to keep up with the course. Hence the instructor needs to elicit and address individual student's doubts.

Muddy Points is a strategy to elicit and address individual students' doubts. In this strategy, the instructor typically asks the question 'what was least clear to you?' at the end of a class. A student's response to this question is called a muddy point. The instructor collects muddy points from every student, usually by having them write it anonymously. Then the instructor goes through the muddy points and addresses them in the next class. While muddy points can be effectively implemented in small classes, it is a challenge to do so in a large class.

To address this challenge, we have developed a framework, LAMP (Large-scale Addressing of Muddy Points). LAMP has three phases: Collection, Addressal and Closure. In the collection phase, the instructor follows a systematic mechanism to elicit students' muddy points (MPs) through traditional and technology assisted modes.

One traditional mode is writing MPs on slips of paper, and one technology-assisted mode is the posting MPs on a Moodle forum. In the addressal phase, the course TAs (teaching assistants), categorize the collected MPs into one of six pre-defined categories, such as clarification question, advanced question, and so on. The TAs address the muddy points according to specific action guidelines for each category. In the closure phase, the instructor highlights the addressal of important MPs, for the benefit of the whole class. The LAMP framework also helps an instructor to determine trends of muddy points after a topic, and to accordingly modify content or instructional strategies for subsequent classes.

In this paper we present the LAMP framework, along with the results of its effectiveness study done in a large enrollment (450 students) introductory computer science course (CS1). Our research questions (RQs) were:

RQ1: How effective is the LAMP framework?

(a) For collection of muddy points from students.

(b) To address the muddy points of students.

RQ2: How does the nature of muddy points change as the semester progresses?

We found that 68% of students felt that they were able to pose a query to the instructor (RQ1a) and 57% were satisfied with the addressal of their queries (RQ1b). We also found that the percentage of clarification questions increases as the semester progresses, while the percentage of conceptual questions remains nearly constant (RQ2).

We believe that the LAMP framework also has affective benefits, since it enables students to shun their inhibition to clarify doubts, including seemingly trivial ones. This may boost their confidence, and once they receive timely answers to their queries, they may develop a more positive attitude towards learning the subject.

II. RELATED WORK

Mosteller [9] used the "Muddiest-point-in-the-lecture" technique as a feedback for his undergraduate statistics course. He found increased level of participation in class. The technique also helped to cater to different students who had varied levels of interest. In addition, analysis of MPs is a valuable aid for instructors to modify content or instructional strategies for subsequent classes.

A. Classification of muddy points

Graesser and Person [10] studied the nature of queries in a tutorial class. They classified the queries by (a) degree of specification, (b) content, and (c) question-generation mechanism, to analyze their quality. For the study they collected queries that were asked either by students or tutors in tutoring sessions on research methods (college students) and algebra (7th graders). Trained judges classified the queries into 18 categories based on

content of information sought. They reported statistics and compared with in-classroom studies. They found that most queries (92%) were from lower order cognitive levels, such as recall level of the Blooms taxonomy [11]. Similar analysis of questions collected in domains like Physics [12] and Biology [13] are also reported in the literature.

B. Technology assisted mechanism for scaling

Moodle is a open-source Learning Management System (LMS) platforms [14]. Costa et. al. [15] present a table in which 15 modules in Moodle are shown to facilitate 7 different teaching learning activities. The Forum module represents “a communication tool where students and teachers can exchange ideas by posting comments” [15]. This feature is suitable to setup a Muddy Point collection and addressal portal. It ensures scalability to reach the entire student community enrolled in a course, as well as provide a collaborative platform for listing queries and having discussions moderated by the teaching assistant.

To the best of our knowledge there is no prior work on mechanisms for soliciting and addressing muddy points, efficiently and effectively in large classroom scenarios.

III. LAMP – THE FRAMEWORK

The LAMP framework utilizes the advantages of both the anonymity of submitting muddy points on slip of paper and the scalability of multiuser LMS like Moodle. It has three phases as illustrated in Figure 1. The arrow direction signifies the flow of communication.

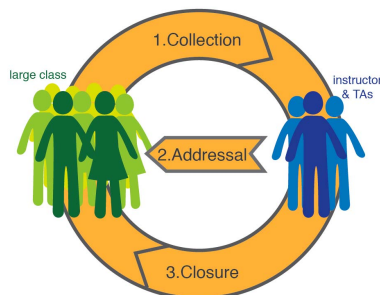


Figure 1. Phases of LAMP

In the Collection phase, instructor and TAs (teaching assistants) collect MPs through different modes. In the Addressal phase, depending on the mode of collection and type of MP, they take specific action to address the MP. In the Closure phase, the instructor highlights important MPs to the whole class. The TAs summarize the discussions on Moodle, thereby archiving the MPs along with solutions. We illustrate the details of the LAMP framework using a running example of a CS1 course.

A. Collection Phase

The first phase of the LAMP framework focuses on collecting students muddy points through 4 different modes. The modes and the corresponding collection mechanisms are:

1. *Queries raised in class:* TAs present in class noted the queries arising during the lecture, and the corresponding response from the instructor.

2. *Queries to instructor outside class:* The instructor addressed such queries briefly and then requested students to note their query on a slip of paper and give it to him/her.
3. *Queries posted on Moodle:* The course used Moodle to archive instructional materials, handle submissions, and communicate with students. The instructor created a forum called “Muddy Points” specifically for students to post their queries, distinct from News and other forums.
4. *Systematic collection of muddy points:* After every major topic, the instructor asked every student in the class to write down one muddy point on a slip of paper (anonymously) and drop it in a box. This mechanism is similar to the approach in Mosteller’s research [9].

TAs periodically listed the queries collected through all the above modes, onto the Moodle forum.

The Collection phase harnesses the advantages of each mode as follows: Queries in class and Queries to the instructor outside class have the advantage of immediate addressal. Queries posted on Moodle provides for scaling, peer discussion, summarizing and archiving. Systematic collection of muddy points has the advantages of anonymity and encouraging every student to communicate their queries to the instructor.

B. Addressal Phase

The second phase of the LAMP framework focuses on categorizing and addressing the collected muddy points.

1) Categories of muddy points

The instructor and TAs sort the collected muddy points into the following six categories:

1. *Clarification* – These MPs pertain to clarification of points explicitly discussed in class. e.g.: What is a loop control variable?
2. *Core* – These MPs relate to understanding of core (fundamental) concepts in the course. e.g.: Will two threads (in Scratch) that receive the same event, execute simultaneously?
3. *Deep* – These MPs lead to deeper conceptual inquiry on a course related topic. e.g.: Can we make our own blocks in Scratch?
4. *Advanced* – These MPs arise from discussions on course related topics, but their depth is beyond scope of the course. e.g.: How are threads scheduled in multi-core processors?
5. *Technical skill* – These MPs relate to operational issues or skill-oriented knowledge of tools usage. e.g.: How to use a C++ string class variable in a printf statement?
6. *Off-Topic* – These queries are not related to the course topic at all; e.g.: Is P=NP?

The first two categories are aligned with the Recall and Understand levels in revised Bloom’s taxonomy [11], while the next two categories are aligned with the Apply and higher levels. Technical skill is a separate category to capture MPs related to use of domain specific tools.

LAMP		ADDRESSAL matrix			
mode of COLLECTION		A in class	B outside class	C MP posts on moodle	D systematic collection of MPs
1 clarification		answer on the spot	answer on the spot	give pointers to class slides and notes	post compiled list and answers on Moodle
2 core			(collect as chits)	give pointers to class slides and notes and additional reference	
3 deep		answer on the spot to scope	answer on the spot to scope (collect as chits)	lead discussion and post answers on Moodle	
4 advanced		mention out of scope and forward query to Moodle discussion		participate in discussion and give reference or resource	
5 technical skill					
addressed by		instructor		teaching assistants	

Figure 2. Addressal Matrix of the LAMP framework

2) Mechanism to address muddy points

The steps followed for addressing the collected muddy points are as follows:

Step 1: Sort the collected muddy points into one of the six categories.

Step 2: For queries that are raised in class or outside class, the instructor immediately addresses them by: (i) Answering Clarification and Core MPs, (ii) Answering Deep MPs to the extent that they are within the scope of the course, (iii) Answering Advanced or Technical skill MPs briefly, and encouraging students to post their MP on Moodle for further discussion.

Step 3: TAs post all the MPs collected on the 'Muddy Points' discussion forum and address them as follows:

Step 3a: For Clarification and Core MPs, TAs give pointers, like slide-number or notes of class discussion.

Step 3b: For Deep MPs, TAs initiate a discussion, encourage students to participate, and provide a summary.

Step 3c: For Advanced MPs, TAs wait for interested students to respond, and then provide references for further self-study.

Step 3d: For Technical skill questions, TAs provide references as well as links to skill-oriented websites, for interested students to follow up on.

These steps are summarized as an Addressal matrix for this phase, shown in Figure 2.

C. Closure Phase

The third phase of the LAMP framework focuses on highlighting important MPs or common/recurring MPs to the whole class. Depending on the category of the MP, closure is done systematically at specific intervals. The stepwise mechanism is given below:

Step 1a: Instructor selects important Core MPs and creates instructional activities, such as multiple-choice questions for peer-instruction, to be done in the next class.

Step 1b: Instructor selects important Deep MPs and creates bridging activities so that students with no prior exposure can also understand the answer to the MP.

Step 2a: TAs compile Clarification MPs on a weekly basis and post their answers.

Step 2b: TAs summarize the discussions on Core and Deep MPs, and post the summary, after each topic.

Step 2c: TAs track discussions on Advanced and Technical skill MPs and then summarize the answers.

These steps are summarized as a Closure matrix shown in Figure 3.

LAMP		CLOSURE matrix		timeline
mode of COLLECTION		A in class	B outside class	
1 clarification			compile and post answers	weekly
2 core		select and prepare multiple choice questions	post summarized discussions	end of topic
3 deep		select and prepare bridging activity.		end of topic
4 advanced		next class	track discussion and summarize answers	2 days after post
5 technical skill				2 days after post
closure by		instructor		TA

Figure 3. Closure matrix for the LAMP framework

The following section describes the implementation of the LAMP framework in the CS1 course.

IV. IMPLEMENTATION

A. Course format

The setting for our study was a large enrollment CS1 class. The instructional goal was to teach programming concepts and skills to a population of 450 first year undergraduate students, across various engineering disciplines. The topics addressed were: control structures such as conditionals, iteration, functions and recursion, data structures such as arrays, matrices, strings and queues, object-oriented structures such as classes and inheritance. Scratch and C++ programming languages were used as the medium to develop programming skills in the students. The course was conducted over 14 weeks during Spring 2013 semester, and was organized into lectures, labs, and exams.

B. Integration of the LAMP framework

Before the start of the course, the instructor introduced the LAMP framework to all the 45 TAs associated with the course. A senior TA was assigned the responsibility as moderator of the Moodle muddy points forum. Other TAs were give responsibility to participate in discussions and answer queries. Some TAs were assigned to attend the classes, to note down queries raised during the class, for later upload into Moodle. Another senior TA was given the responsibility of ensuring that all muddy points were addressed within the time limit specified.

In the first class, the instructor introduced the concept of Muddy Points to students and informed them of the advantages, stressing on the anonymity. The instructor carried a Muddy Points Box and small slips of paper to class. The instructor encouraged students to write their muddy points and drop into the box, before/after class.

There were two sessions of systematic collection of muddy points, one before mid-term and one before end-term. These were conducted in-class as follows: Ten minutes was specifically set aside for students to write muddy points on the topics covered till then. Each student had to write a query (anonymous) and drop it in the Muddy Points Box while leaving.

The addressal and closure phases were as per the LAMP framework described in Section III.

C. Example Queries

The MPs that were collected across the semester resulted in several distinct queries. Some examples of the queries in each category are given below:

Clarification

- i. *Can we read & write on the same file simultaneously?*
- ii. *What do you mean by 'system call'?*

Core

- i. *How to return multiple values from a function?*
- ii. *What is difference between loop & recursion?*

Deep

- i. *How can I change CPU time of a thread?*
- ii. *How to create a file that will be automatically deleted on program termination?*

Advanced

- i. *What happens in multi-core processors?*
- ii. *Can we create commands by ourselves?*

Technical Skill

- i. *How to access system clock through C++?*
- ii. *What is the difference between scanf and cin?*

V. EFFECTIVENESS & PATTERN STUDY

We carried out a research study to evaluate the effectiveness of the LAMP framework. Our research questions, methodology and analysis technique, are discussed in this section.

A. Research Questions (RQ)

RQ1: How effective is the LAMP framework?

- a. For collection of muddy point from students.
- b. To address the muddy points of students.

RQ2: How does the pattern of muddy points change as the semester progresses?

B. Methodology

1) Instruments

To answer RQ1 we administered three questions as an online survey to the students. One question measured their perception of effectiveness of the collection phase, while another measured their perception of effectiveness of the addressal phase, on a Likert scale. The third was a rank order question on their preference of mode for MPs. The specific questions are given below:

Answer on a Likert scale:

1. I had enough opportunities to put forward my queries to the instructor.
2. I got answers to my queries either in class or through Moodle.
3. Rank the following modes of asking queries
 - i. Raising question in the class
 - ii. Putting a Muddy Point chit
 - iii. Posting a Muddy Point discussion thread on Moodle
 - iv. End of class discussion with instructor.

To answer RQ2 we tracked the logs on Moodle forum and analyzed the queries collected through different modes. This helped in detecting patterns in the categories of MPs at any given point of time.

2) Sampling

Our sample consisted of 450 first year students registered in the course. Survey was offered to the whole class and 340 responses were received. There were 274 completed data points (80% of total number of response), i.e., responses which had uniquely given preferences of mode of queries and answered the Likert scale questions.

To answer RQ2 we analyzed a total 343 student queries (195 before mid-sem + 148 before end-sem) across the whole semester.

C. Analysis

To answer RQ1 on effectiveness, we analyzed survey data to get student perception of the framework. The Likert's scale data was segregated in three groups Agree (combining rating 4 and 5), Neutral (rating 3), and Disagree (combining rating 1 and 2).

For both the questions in the survey, we studied the distributions of responses. We analyzed effectiveness of collection phase (RQ1a) using the percentage of students who agreed that they had scope to ask their queries. We analyzed effectiveness of addressal and closure phase (RQ1b) using the percentage of students who agreed that their queries were answered satisfactorily.

We carried out a chi-square (χ^2) test to determine whether the perception of receiving a satisfactory answer was influenced by the perception of whether they could put forward queries to the instructor at the first place.

To answer RQ2 we considered percentage composition of each category of MPs, from the 343 MPs collected.

VI. RESULTS

The frequency distribution of students' perception on Likert's scale for the two questions in the survey is plotted in Figure 4.

68% of students agreed (4 or above in Likert scale) that they were able to pose their queries to the instructor. 57% of students agreed (4 or above in Likert scale) that they received answers.

The χ^2 test on the data set shows that the perception of getting a satisfactory answer depends on the perception of whether they had enough opportunities to ask a query ($\chi^2 = 77.26$, $\text{dof}=4$, $P < 0.001$).

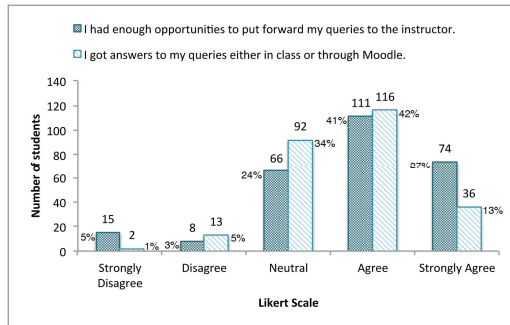


Figure 4. Students' perception (N=274)

To quantify the perception, we collected the frequencies of students who prefer one mode over the other, from their ranking of the four modes. TABLE I shows this distribution. There does not seem to be any absolute ranking of the modes for the given sample, and all modes seem equally likely to be chosen for queries.

TABLE I. RANK DISTRIBUTION OF EACH MODE OF ASKING QUERIES

N=276	Raise Q. in class	Muddy point chit	Post on Moodle	Discuss with instructor
1st	80	85	38	73
2nd	51	77	72	76
3rd	43	59	109	65
4th	102	55	57	62

A transition diagram of how students' perception on whether they could ask queries, mapped to their perception of receiving an answer, is shown in Figure 5.

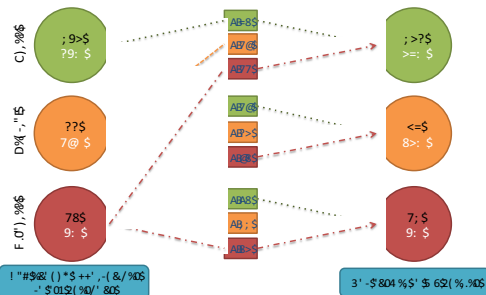


Figure 5. Distribution and transition of students' perception over the survey questions 1 to 2.

We observe from Figure 5. that nearly three out of four students who agreed they had enough opportunities to pose queries were satisfied that they received the answer. For these students, we analyzed their preferred mode of asking the query. This is shown in TABLE II. Further discussion is presented in the next section.

TABLE II. DISTRIBUTION OF QUERY MODES USED BY SATISFIED STUDENTS

Had opportunities to ask query – Received answer to query	Raise Q. in class	Muddy point chit	Post on Moodle	Discuss with instructor
Agree - Agree	34%	27%	12%	27%

To answer RQ2, the distributions of queries, from the Moodle logs and systematic collection of muddy points, are shown in 0along with the topics covered.

Before the mid-sem, 44% were Clarification MPs and 21% were Deep MPs. As the semester progressed and newer topics were discussed, Clarification MPs increased to 71%, while Deep MPs dropped to 7%, before end-sem.

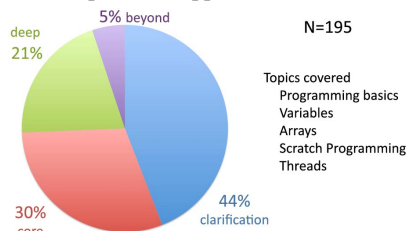


Figure 6a. Distribution of muddy points before midterm

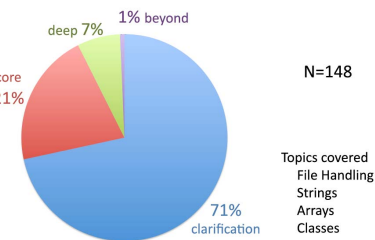


Figure 6b. Distribution of muddy points before endterm

VII. DISCUSSION

The answers to our research questions are as follows:

For RQ1a - How effective is the LAMP framework for collection of muddy points from students? – we found that 68% of students agree that the LAMP framework provided them with satisfactory means to pose their MPs.

For RQ1b - How effective is the LAMP framework for addressal of muddy points collected? – we found that 57% of students agreed that their MPs were addressed satisfactorily, either in-class or on the forum.

For RQ2 - How does the pattern of muddy points change as the semester progresses? – we found that the percentage of Clarification MPs increased from 44% to 71% while that of Deep MPs dropped from 21% to 7%.

It is worth noting that the results of RQ1a and RQ1b include responses from students who did not pose any MPs. A deeper analysis (Figure 5) shows that 73% of

students who agreed that they had opportunity to pose an MP were also satisfied that their MP was answered.

The overall analysis of the ranking across modes does not highlight a unique preference for any one mode over the other (Table I). However there is preference in ranking of modes among the students who agreed that they could pose a MP and got an answer (TABLE II). A possible reason for higher preference of 'in-class' mode could be that students are familiar with this mode and it can also address their query immediately. Even though the Moodle forum showed the lowest preference, it is beneficial from the perspective of addressal. It can be used to convey all the MPs and all the responses to all the students, as well as archive the responses for later reference.

We believe that the mix of the four modes that are provided in the LAMP framework caters to differences in the nature of students. Queries in-class are typically from students who are able to cope with the course. Those having difficulty in coping with the course, typically assume that others may regard their questions as trivial, and do not speak up in-class. This hampers their learning. Muddy points slips provides anonymity to such students and the addressal phase enables them to realize that there are others who have similar difficulties.

The overall results of RQ2 are not surprising. Many students had prior knowledge on the initial topics in the course. So the number of queries in the Core and Deep category were higher during that time. Gradually as the course evolved with new topics for everyone, the nature of queries shifted towards Clarification category. However we note that more than 40% of the questions (core+deep+advanced) are in Understand to Analyze level of Bloom's taxonomy. This percentage of cognitively higher-level questions is about five times more than the results reported in [10]. We believe that this is because the LAMP framework provides for multiple collection modes and timely addressal mechanism.

The LAMP framework also helped the instructor to moderate the pace and depth of the course by analyzing the MPs. The instructor mentioned that after one systematic collection of MPs, he found the class had considerable core MPs in the topics of Arrays, which was contrary to his expectation. So, in subsequent classes he revisited the topic before proceeding with the course.

Since the LAMP framework provides mechanisms to efficiently collect large number of MPs from a large class, a frequency analysis on the collected MPs can be used by instructors to assess students' difficulties and modify instructional strategies and duration of topics accordingly.

One limitation of this study is that all the students at the institute come from a highly competitive national exam. So the results may not generalize to other institutes. However, the categories of MPs in the LAMP framework and the addressal mechanisms could be directly adopted in any undergraduate engineering course at any institute.

VIII. CONCLUSION

The LAMP framework is a systematic mechanism for instructors of large classes to ensure that every individual

student's doubts are addressed. The framework integrates the advantages of face to face interaction, anonymous muddy point slips, and online forum, to elicit and address muddy points in a large class. Our pilot study found that the framework is useful for students to get clarifications and for the instructor to assess students' difficulties. Since the framework requires considerable involvement of TAs, we now plan to optimize it for smoother implementation.

REFERENCES

- [1] Deslauriers, L., Schelew, E., Wieman, C. "Improved Learning in a Large-Enrollment Physics Class", *Science* 332, 862 (2011), DOI: 10.1126/science.1201783
- [2] Woelk, K. "Optimizing the Use of Personal Response Devices (Clickers) in Large-Enrollment Introductory Courses.", *J. Chem. Educ.*, 85 (10), pp. 1400-1405, 2008.
- [3] Allen, D., & Tanner, K. "Infusing active learning into the large-enrollment biology class: seven strategies, from the simple to complex." *Cell Biology Education*, Vol. 4, No. 4, pp. 262-268. 2005
- [4] Gordon S., Reid A. & Petocz P., "Educators' conceptions of student diversity in their classes", *Studies in Higher Education*, Vol. 35, No. 8, pp. 961-974, 2010. DOI: 10.1080/03075070903414305
- [5] Simon, B., Parris, J., and Spacco, J. "How we teach impacts student learning: peer instruction vs. lecture in CS0.", In *Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13)*. ACM, New York, NY, USA, pp. 41-46. 2013. DOI=10.1145/2445196.2445215
- [6] Porter, L., Lee, C. B., Simon, B., and Zingaro, D. "Peer instruction: do students really learn from peer discussion in computing?", In *Proceedings of the seventh international workshop on Computing education research (ICER '11)*. ACM, New York, NY, USA, pp. 45-52. 2011. DOI=10.1145/2016911.2016923
- [7] Martinez, A. "Using JITT in a database course.", In *Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12)*. ACM, New York, NY, USA, pp. 367-372. 2011. DOI=10.1145/2157136.2157245
- [8] Lockwood, K. and Esselstein, R. "The inverted classroom and the CS curriculum.", In *Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13)*. ACM, New York, NY, USA, pp. 113- 118. 2013. DOI=10.1145/2445196.2445236.
- [9] Mosteller, F. "The 'Muddiest Point in the Lecture' as a Feedback Device.", *The Journal of the Harvard-Danforth Center. On Teaching and Learning*. Vol. 3, pp. 10-21, 1989
- [10] Graesser, A. C., Person, N. K., "Question Asking During Tutoring", *American Education Research Journal*, Spring 1994, Vol. 31, No. 1, pp. 104-137, 1994
- [11] Bloom, B.S. (Ed.) "Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain", New York ; Toronto: Longmans, Green, (1956).
- [12] Harper, K. A., Etkina, E., & Lin, Y., "Encouraging and analyzing student questions in a large physics course: Meaningful patterns for instructors.", *Journal of Research in Science Teaching*, Vol. 40, No. 8, pp. 776-791. 2003
- [13] Colbert, J.T., Olson, J.K. and Clough, M.P. "Using the Web to Encourage Student-generated Questions in Large-Format Introductory Biology Classes" *CBE—Life Sciences Education* Vol. 6, 42-48, Spring 2007
- [14] Martin-Blas, T., Serrano-Fernández, A. "The role of new technologies in the learning process: Moodle as a teaching tool in Physics", *Computers & Education* Vol. 52, pp. 35-44, 2009
- [15] Costa, C., Alvelos, H., & Teixeira, L. "The Use of Moodle e-learning Platform: A Study in a Portuguese University.", *Procedia Technology*, Vol. 5, pp. 334-343, 2012