Journal of Engineering Education Transformations ,
Volume 35 , No. 1 , July 2021 , ISSN 2349-2473, eISSN 2394-1707

Active Learning Strategies and Blended Learning Approach for Teaching Under Graduate Software Engineering Course

Abirami A.M [1], Pudumalar S [2], ThiruchadaiPandeeswari S [3]
Department of Information Technology, Thiagarajar College of Engineering , Madurai, Tamil Nadu
[1] abiramiam@tce.edu
[2] spmit@tce.edu
[3] eshwarimsp@tce.edu

Abstract: Software Engineering is a core theory course offered in Undergraduate Engineering programmes. It is one of the challenging courses for the teaching faculty. The course includes various systematic approaches, and methods that can be employed for designing, developing, testing and maintaining quality software applications. The course also focuses on the latest tools and technologies that are being practiced in the industry. Effective content delivery methods leveraging active learning strategies for the classical topics have been well researched and established. However, improvement in content delivery plan and execution is required for relatively new and evolving topics like DevOps and version controlling. This paper presents an experimental study of application of the various active learning techniques such as discussion forums, tech talks followed by quiz for such topics. Impact of the application of the improved content delivery plan on the course outcomes attainment by the learners has also been observed and presented. The enhanced approach has proved to improve student learning outcomes, which are measured and presented using a standard set of tools and metrics. This paper uses the logistic regression model for the study of impact of blended learning approach on student learning outcomes which produces 60% accuracy.

Keywords: DevOps, Software Maintenance, Version Controlling, Software Engineering, Course Outcome Attainment, Active Learning Strategies

1. Introduction

The Software Engineering course normally aims to present various standard methods, tools and procedures available for quality software development. The course emphasizes systematic and quantifiable approaches available for the development and maintenance of software. This course helps the students to identify a real time problem, analyse and understand the requirements, formulate functional and non-functional requirements, choose a suitable process model for the development of software and incorporate the milestones prescribed in the process model appropriately. Further students get to learn the umbrella activities of project development such as project scheduling, staffing, various standard project and product metrics. Covering a rich set of contents, this course is paramount for Undergraduate Engineering Students to prepare them for their future prospects as a Software Engineering Professional.

With the advent of modern technologies such as IoT, Cloud, artificial intelligence, mobile based apps

Abirami A.M
Department of Information Technology,
Thiagarajar College of Engineering , Madurai, Tamil Nadu
abiramiam@tce.edu

and trends such as automation, fast to market, frequently changing requirements and frequent updates, Software Engineering as a paradigm is evolving very fast. Lots of new technologies, practices and tools are being introduced. For example, DevOps practices and version control systems like Git have become integral parts of contemporary Software Engineering (Chatley, 2016; Bobrov, 2019).

To make the undergraduate engineering course on Software Engineering up-to-date with the industry trends, topics on such new and modern practices and tools are included in the course plan along with the other classical topics (Bobrov, 2019). However, from the previous course runs, it was observed that content delivery of these relatively new topics require enhancements so that the learners will be able to achieve the expected level of course outcomes attainment.

Hence, an enhanced blended learning approach that involves number of in-class, out-of-class (online) and extension activities were evolved and adopted in the content delivery and assessment for newer topics such as DevOps and version controlling that are part of the Software Engineering course. Adopting the suggested practices proved that the majority of students were able to achieve all of the course outcomes (Mason, 2017, Demchenko, 2018).

In this paper, active learning method like peer learning through discussion forums, which is part of the blended learning approach is presented (Cheng, 2018). The impact of the proposed approach has been observed from two major perspectives as follows:

(1) Effectiveness of the activities in leading the learners towards further exploration of futuristic technologies.

(2) Impact of these activities in the course outcomes attainment.

Quantitative analysis has been carried out on the summative assessments and Course Outcomes (COs) attainment data. The Logistic Regression model is used to determine the most important factor among the different learning processes that impacts the CO attainment. The paper is organized as follows: section 2 discusses the research work done on this area, section 3 describes the research questions, section 4 explains the methodology, section 5 discusses the results and section 6 concludes the paper.

2. Background

Effective learning environment and blended mode for teaching learning process suits for all types of courses. Hubscher et. al (2003) used new pedagogical approaches like constructive and collaborative activities and visual demonstration for learning algorithms which improved self learning among the students. Chen et. al. (2009) emphasized that hands-on experience could provide better understanding on the methods and concepts of software engineering and make the students ready for the IT industry. Eison (2010) studied the importance of ALS inside the classroom and to create a learning environment to have enhanced learning. Huan (2012) explored practice-based teaching methods for Software Engineering course by analyzing the shortcomings of traditional methods of teaching. The author suggested that case-based learning and creating a space for students to discuss and practice software methodology concepts in the classroom would really help the students to understand what Software Engineering is all about. Maguire et. al. (2014) adopted a pair programming model to teach Computer Programming, Data Structures and so on for the students who had not studied computer science courses in their school days. Continuous assessment was carried out in different aspects like interdependence and co-ordination between the pair, and individual contribution to problem solving which resulted in drastic reduction of failure rate.

Venson et. al. (2016) presented a framework of academy-industry collaboration that would facilitate students to undergo practical and academic activities in a real world scenario using an IDEAL instructional model (Initiating, Diagnosing, Establishing, Acting, and Learning). Alabbadi et. al. (2016) proposed novel methods by combining cooperative learning and mastery learning strategies to teach software engineering using social media tools. The author emphasized that usage of modern ICT tools and teaching strategies in classrooms would support students to actively learn Software Engineering than the traditional chalk and talk approach. Chatley (2016) proposed different topics in software engineering that are to be taught to students in their four year study period. He was sure that this type of teaching learning methodology would improve the learning experience more effectively and would make the students ready to take up IT jobs immediately. Mason et. al. (2017) proposed novel methodology for teaching Agile Development using DevOps approach for the students of Regis University through the courses Software Engineering and Database Technologies Practicum.

Ramachandran (2017) proposed Active Learning & Teaching model based on specific learning outcome (ALT Model) by enforcing three types of learning practices such as Divergent Thinking, Differential Assessment and Collaborative Learning. The approach was implemented with second year and final year students where the final year students guided the second year students and collaborated with them. Fonseca et. al. (2017) explored Problem-based Learning (PBL) and agile software engineering practices in order to achieve learning outcomes and to improve students' participation during classes. Cummings et. al. (2017) proposed different blended mode strategies for teaching which would advance skill sets of students through learner-centered instructions along with tools and technologies.

Alva et. al. (2018) practiced Problem Based Learning (PBL) and Peer Assisted Learning (PAL) while delivering the Software Engineering content to fourth semester students, which in turn enhanced the learning outcomes of Students. Demchenko et. al. (2018) explained unique aspects of DevOps and how it could be implemented among university students. Ramos et. al. (2018) experimented and presented the impact of using Team based learning for Software Engineering course. Team Based Learning (TBL) along with the IDEAL approach showed an increase in attainment of course objectives. Alramouni et. al. (2018) conducted a study from academic year 2013 to 2015 (2 terms in each year) for Software Engineering course by investigating various strategies for identifying at-risk students (students those who have a higher probability of failing academically or rare class participation), engaging and making positive attention of those students through interesting class activities, appropriate assessment methods and assignments to make connection between their classroom learning and real-world problems. Bobrov et. al (2019) taught DevOps both to students and industry personnel and consolidated their lessons learnt. They proposed to update Software Engineering curriculum with the emerging technology DevOps framework, which in turn support making the role ready engineers.

Lokare et.al (2020) discussed the challenges in reaching out to students with laboratory courses rather than theory courses. Experimental study was performed to the Internet Technology Laboratory course by following different pedagogic paradigms, namely Project Based Learning and Collaborative Learning. Results proved that the course attainment significantly increased around 15% to 20%, thereby enhancing the student learning capacity. Cheng (2018) observed that knowledge sharing and peer learning are important characteristics for learning Software Engineering concepts. Software design course has been experimented to analyse the possibility of adoption of pear learning significantly. Students were grouped into team by at most three ones, so they can choose as individual learning. Both review and summary quizzes were held at the first and the last 15-min in the Moodle every week, respectively. A web-based tool was designed to illustrate the peer learning performance and students can view their scores. This approach grabbed the attention of many students and increased student participation than individual learning. The author demonstrated that pear learning is highly beneficial to student community.

It could be inferred from the literature that many variations of the active learning strategies are successful in enhancing the student learning outcomes. From the literature survey, following observations were made to make the course more effective:

- The course must have well defined learning outcomes, course/session plan and effective assessment methods
- Use of technology must be incorporated into the content delivery and assessment.
- Students must be given suitable platforms for communicate and collaborate among themselves
- The collaborations may be suitably monitored and mended, if needed.

Though the course has well defined COs and customized course or session plan, it would be difficult to achieve certain COs as some students may skip their Continuous Assessment Test (CAT), or perform poorly in their tests. Sometimes, the topic may be new and the instructor himself may find it challenging for the content delivery. This scenario motivates to practice active learning strategies for delivering those two specific topics with the support of stakeholders from industry, alumni, and senior students. Blended content delivery plan is prepared by tailoring the traditional course plan, specifically for the two topics. Number of activities are identified that are to be performed by the students. The motivation for this study is supported by the following research questions:

RQ1: Does 'peer learning' through Discussion Forums enhance the exploration of relatively new technologies?

RQ2: Does participation in tech talks and follow-up Quiz support attainment of Course Outcome related to futuristic technologies?

This research is an experimental study on incorporation of active learning strategies for teaching the topics 'DevOps' and 'Version Controlling' with reference to the cognitive level of course outcomes. Effective assessment techniques have been incorporated to provide constructive feedback to the learners.

3. Methodology

This section describes the blended learning approach for teaching the new topics in the Software Engineering course, as shown in Fig. 1. Various pedagogical practices tuned with suitable instructional models have been adopted by instructors to teach the course. Active learning strategies along with traditional classroom teaching methods have been practiced so as to have higher level of student engagement.

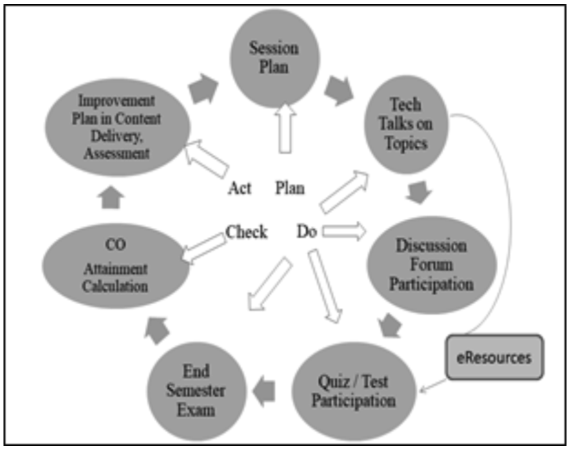

Session Plan
Tech Talks on Topics
Discussion Forum Participation
Quiz / Test Participation
End Semester Exam
CO Attainment Calculation
Improvement Plan in Content Delivery, Assessment
Act
Plan
Check
Do
eResources

Fig.1 : Framework of blended learning practices

A. Course Design

The Software Engineering course has six Course Outcomes (COs) with the Bloom's level of taxonomy 'Understand' and 'Apply' and are mapped to 12 Programme Outcomes (POs). The two COs which are specific to the experimental study are:

CO1 - Demonstrate DevOps life cycle processes (Understand)

CO2 - Illustrate the use of version controlling and tracking mechanisms (Understand)

Table 1 describes the CO-PO mapping for these two COs.

Table 1: COs vs POs Mapping

CO1	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
	M	L			L	
	PO7	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
			S	S		S
CO2	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
	M	L			L	
	PO7	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
			S	S		S

As these two COs are at 'Understand' level, they are mapped to 'medium - M' and 'low – L' to PO1 and PO2 respectively. These two COs deal with the use of tools in the course. Students would be given knowledge of different DevOps and version controlling tools and are motivated to use them in future. Hence they are mapped to PO5 and PO12 with the mapping 'low –L' and 'strong – S' respectively. Students would be asked to participate in discussion forums, seminars, class discussions on these topics and so on which would help them to express their learning among their peers. Hence, these two COs are mapped to PO9 and PO10 'strong –S'. The participation of students in the discussions, seminars, quiz and written test would be used for assessing the knowledge level of each student, in terms of CO attainment.

B. Course Plan

Course plan has been designed in such a way that students would be engaged in different types of activities like discussion forums, seminars, and quiz pertaining to the course outcomes and for the entire duration of the course. As these two COs are particular to industry practices, it would be better if some of the topics related to these two COs are delivered by industry experts also. This practice would definitely add value to the course and to the learners.

C. Content Delivery and Assessment

The industry specific topics can be delivered to the students not only by the Faculty, but by the industry experts too. The following guidelines may be practiced for organizing Guest Lecture from the industry professionals:

- Identify suitable industry expert, and/or alumni for the content delivery, who has adequate knowledge on the topic(s)

- fix the date and submit for approval from the authorities

- Give short lecture or preamble to students about the topic; motivate them by articulating the need of the topic(s)

- Organize the guest lecture event

- Encourage the students for interaction with the expert

- Organize learning materials and share eresources among the students

- Ask the students to participate in the discussion forum; ask them to reflect their learning from the lecture event; ask them to reply to others' posts; ask them to post queries. The forum can be monitored by the instructor and queries can be answered.

- Arrange seminars in teams; student volunteers would go through the learning materials and share their understanding with others

- After a week, quiz can be organized using Google Forms or Canvas or any other platform. Questions have to be framed consciously and should not be very trivial. Quiz should ensure the instructors indirectly, whether the students used learning resources and participated in the forum effectively.

- Students would perform better for the questions related to these topics in the subsequent written test/examinations.

D. Course Outcome Attainment Calculation

Generally both Continuous Assessment Test (CAT) and Terminal Examinations (TE) data considered for CO attainment calculation. Sometimes equal weightage is given for both; sometimes it may be 60% for CAT and 40% for TE or vice versa. If course exit survey data available, the % split may vary like 60% for CAT, 30% for TE and 10% for survey.

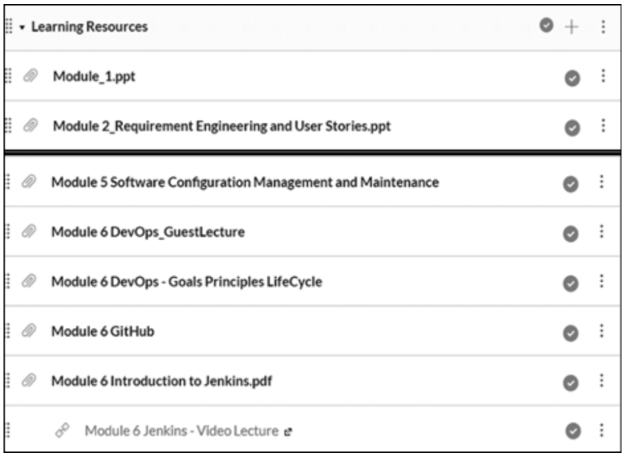
E. Improvement Plan

Based on the actual CO attainment value, when compared with the target value, an improvement plan may be prepared for the next batch of students. More active learning strategies may be identified in the course / session plan to improve the student participation and their learning.

4. Results and Discussions

A. Experimentation Details

A batch of second year students of BTech IT programme were considered for this study, consisting of 131 students. Canvas portal was used for Learning Management System (LMS). All announcements, learning resources, discussions, quiz and so on are posted in Canvas.



Learning Resources

Module_1.ppt

Module 2_Requirement Engineering and User Stories.ppt

Module 5 Software Configuration Management and Maintenance

Module 6 DevOps_GuestLecture

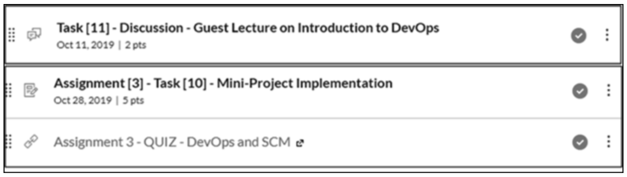
Module 6 DevOps - Goals Principles LifeCycle

Module 6 GitHub

Module 6 Introduction to Jenkins.pdf

Module 6 Jenkins - Video Lecture

Fig 2 : Learning Resources in Canvas



Task [11] - Discussion - Guest Lecture on Introduction to DevOps
Oct 11, 2019 | 2 pts

Assignment [3] - Task [10] - Mini-Project Implementation
Oct 28, 2019 | 5 pts

Assignment 3 - QUIZ - DevOps and SCM

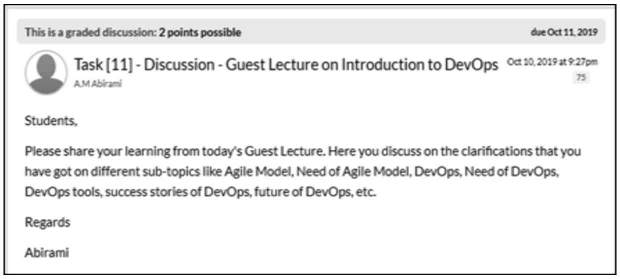
Fig 3 : Discussion Forum and Quiz in Canvas

Figures 2, 3 and 4 show the snapshots of eResources shared, discussions and quiz. Techtalk series has been arranged for these two topics: industry expert delivered lecture on the topic DevOps process framework; alumni presented video lecture on Jenkins framework; senior student demonstrated the use of GitHub repository to the students.

B. Formative Assessment

Discussion forum was initiated in Canvas. Totally

75 students (out of 131) participated and reflected their views. Figures 4 (a), (b), (c) and (d) explain how the Canvas platform has been used effectively in the learning process through discussions.



This is a graded discussion: **2 points possible** due Oct 11, 2019

Task [11] - Discussion - Guest Lecture on Introduction to DevOps Oct 10, 2019 at 9:27pm
A.M Abirami 75

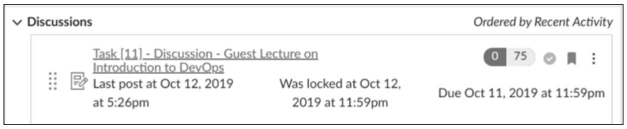
Students,

Please share your learning from today's Guest Lecture. Here you discuss on the clarifications that you have got on different sub-topics like Agile Model, Need of Agile Model, DevOps, Need of DevOps, DevOps tools, success stories of DevOps, future of DevOps, etc.

Regards

Abirami

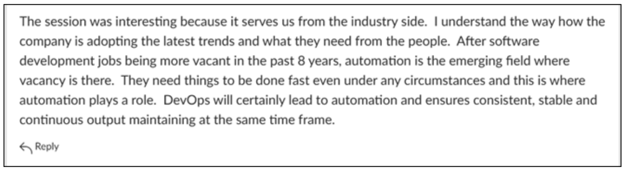
Fig. 4 (a) : Discussion Forum initiation in Canvas



Discussions — *Ordered by Recent Activity*

Task [11] - Discussion - Guest Lecture on Introduction to DevOps
Last post at Oct 12, 2019 at 5:26pm
Was locked at Oct 12, 2019 at 11:59pm
0 75
Due Oct 11, 2019 at 11:59pm

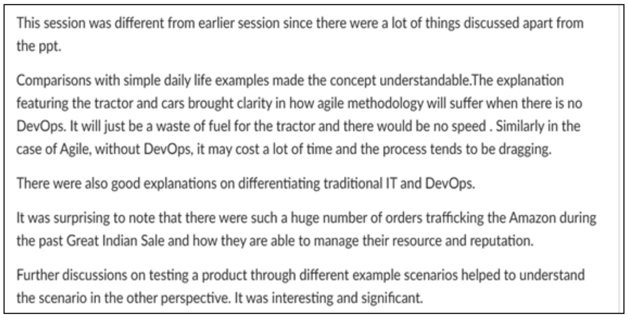
Fig. 4 (b) : Participation of Students in the Forum



The session was interesting because it serves us from the industry side. I understand the way how the company is adopting the latest trends and what they need from the people. After software development jobs being more vacant in the past 8 years, automation is the emerging field where vacancy is there. They need things to be done fast even under any circumstances and this is where automation plays a role. DevOps will certainly lead to automation and ensures consistent, stable and continuous output maintaining at the same time frame.

Fig. 4 (c) :Sample reflection by the student in the Forum

Though the expert differentiated agile and DevOps practices using real time applications, few students got confusion and they were not able to understand the marginal differences between the two concepts. These students raised their doubts in the forum and it was explained by another student, as shown in the Fig. 4 (d). The discussion forum is an effective tool for peer learning.



This session was different from earlier session since there were a lot of things discussed apart from the ppt.

Comparisons with simple daily life examples made the concept understandable.The explanation featuring the tractor and cars brought clarity in how agile methodology will suffer when there is no DevOps. It will just be a waste of fuel for the tractor and there would be no speed . Similarly in the case of Agile, without DevOps, it may cost a lot of time and the process tends to be dragging.

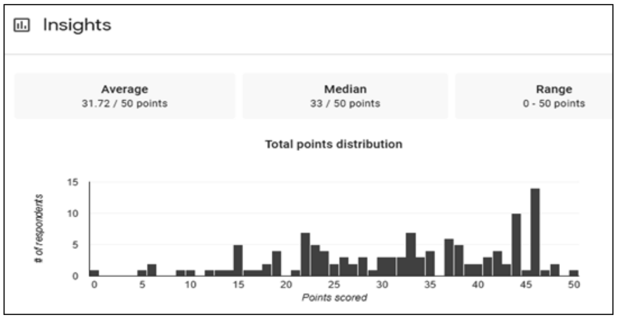
There were also good explanations on differentiating traditional IT and DevOps.

It was surprising to note that there were such a huge number of orders trafficking the Amazon during the past Great Indian Sale and how they are able to manage their resource and reputation.

Further discussions on testing a product through different example scenarios helped to understand the scenario in the other perspective. It was interesting and significant.

Fig.4(d):Reply by the student for a Query in the Forum

Google form was designed for conducting quiz on the topics. Quiz had different types of questions like multiple choice questions (single answers correct, multiple answers correct), fill in the blanks, and matching.



Insights

Average 31.72 / 50 points
Median 33 / 50 points
Range 0 - 50 points

Fig. 5 : Google Form Responses for Quiz

It consisted of 15 questions for 50 marks with the time limit set to 45 minutes. 120 students participated in the quiz. One student scored 100%, 19 students scored > 90%; class average is 62%. Median is 33 and the mode is 46, as shown in Figure 5.

C. Summative Assessment

One continuous assessment test and terminal examination were conducted to assess the student's knowledge. Data was collected from these assessments specific to two COs under study, and it is explained in Table 2.

Table 2 : Details of Summative Assessment

CO	Type	Total Students Attended	Number of Questions	Marks Allotted (out of 100)	Direct CO attainment (in %)
CO1	CAT	67	3	12	34
	TE	131	2	8	66
CO2	CAT	67	5	16	30
	TE	131	1	6	60

Table 2 shows that 66% of students scored above the expected target level for CO1 and 60% for CO2 through Terminal Examinations. CO attainment calculation is explained in the next sub-section.

D. CO Attainment Calculation

Equal weights are given to CAT and TE i.e. 45% and 10% is given to course exit survey, as shown in Table 3. Expected Proficiency (EP) for COs is set to 70% and the Expected Level of Attainment (ELA) is set to 70%. It means that atleast 70% of students (ELA) has to score ≥ 70% (EP) for each CO. These target values are set based on the result data of previous batches.

CO attainment values, shown in Table 2, are multiplied with the corresponding weights, as shown in Table 3, and the actual CO attainment values are calculated using CAT, TE (direct mode) and course exit survey (indirect mode).

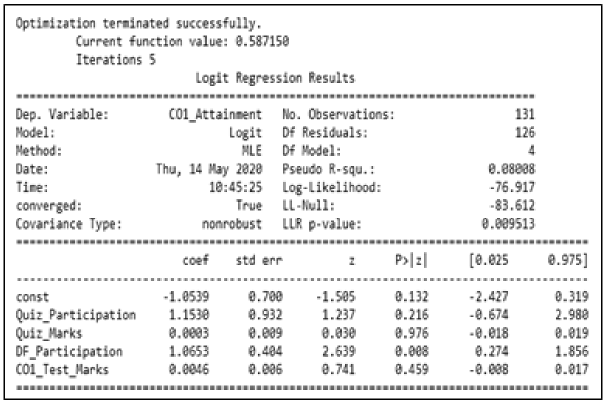
Table 3 : CO Attainment Calculation

CO	CAT (wt. 0.45)	TE (wt. 0.45)	Course Exit Survey - indirect (wt. 0.1)	Actual CO attainment
CO1	15.4	29.7	6.85	51.9%
CO2	13.4	27	6.97	47.3%

From the Table 3, the actual CO attainment values are very low, as there were nearly 50% absentee for CAT. There could be various reasons for more absenteeism like students would have left this test in choice, or he would not have understood the concepts, or he would not have actively participated in the activities given inside/outside classroom. Considering this scenario, the study is extended to identify the significant factors from the set of activities which support for better performance.

E. Prediction of Student Performance

Data analytics is used on these result data to predict the performance of students, given his active participation inside/outside classroom activities. The approach used Logistic Regression model for classifying the CO attainment (yes/no) based on the student activities. The factors like participation in discussion forum (p), participation in quiz (q), marks scored in quiz (r), marks scored in test (s) have been considered as independent variables. The dependant variable is whether student scored greater than the target value or not in the terminal examinations (t). The regression model says the feature 'p' is more significant (p-value is 0.008) than others, as shown in Fig 6.



```
Optimization terminated successfully.
         Current function value: 0.587150
         Iterations 5
                           Logit Regression Results
==============================================================================
Dep. Variable:         CO1_Attainment   No. Observations:                  131
Model:                          Logit   Df Residuals:                      126
Method:                           MLE   Df Model:                            4
Date:                Thu, 14 May 2020   Pseudo R-squ.:                 0.08008
Time:                        10:45:25   Log-Likelihood:                -76.917
converged:                       True   LL-Null:                       -83.612
Covariance Type:            nonrobust   LLR p-value:                  0.009513
=======================================================================================
                         coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------------
const                 -1.0539      0.700     -1.505      0.132      -2.427       0.319
Quiz_Participation     1.1530      0.932      1.237      0.216      -0.674       2.980
Quiz_Marks             0.0003      0.009      0.030      0.976      -0.018       0.019
DF_Participation       1.0653      0.404      2.639      0.008       0.274       1.856
CO1_Test_Marks         0.0046      0.006      0.741      0.459      -0.008       0.017
=======================================================================================
```

Fig. 6 : ROC Curve – Testing Data of CO1

The performance of model is evaluated by ROC Curve (Receiver Operating Characteristics). Fig. 7 shows the ROC curve of train data and Fig. 8 shows the ROC curve for test data.

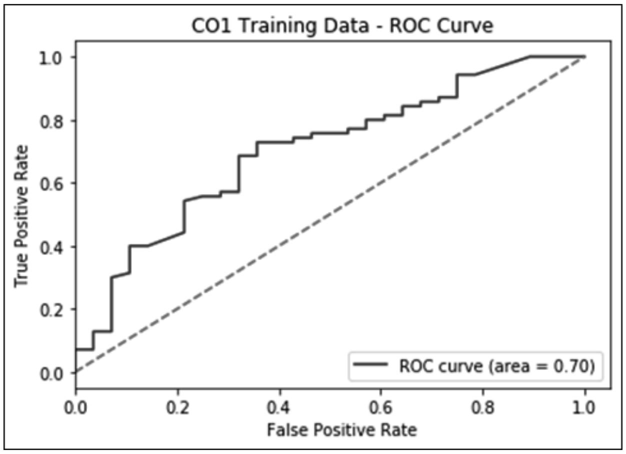


Fig. 7 : ROC Curve – Training Data of CO1

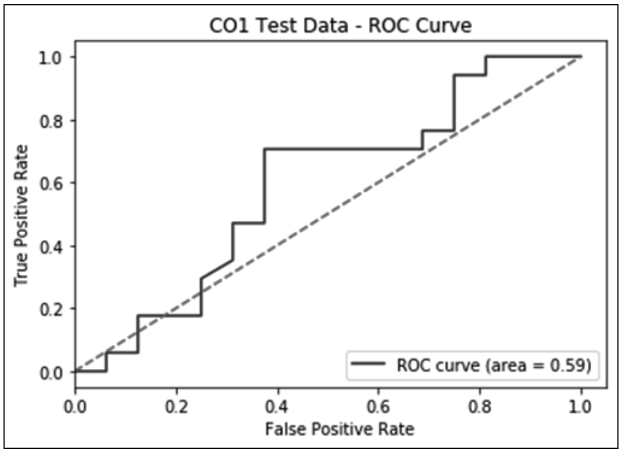


Fig. 8 : ROC Curve – Testing Data of CO1

k-fold cross validation is used. Area occupied by ROC curve for test data is about 60%, as shown in Fig. 8. The model predicts data on an average with 60% accuracy. The optimum threshold values for the logistic regression model is determined by using 'true positive' and '1- false positive' rates. The precision-recall curve is drawn for various values of thresholds, as shown in Fig. 9 and 10.

From the Fig. 9, it is seen that higher precision value (0.67) is obtained when threshold value is between 0.7 - 0.75; and the recall value is 0.71. The precision and recall value is same (0.65) when the threshold value is 0.76, which is the optimum threshold value, for CO1 attainment calculation.

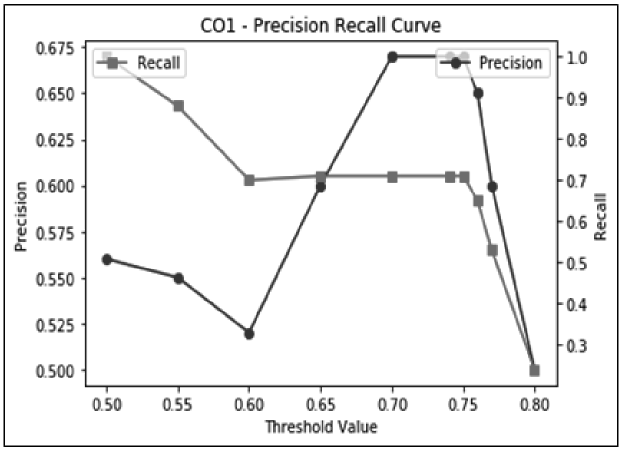


Fig. 9 : Precision – Recall Curve – CO1 Attainment

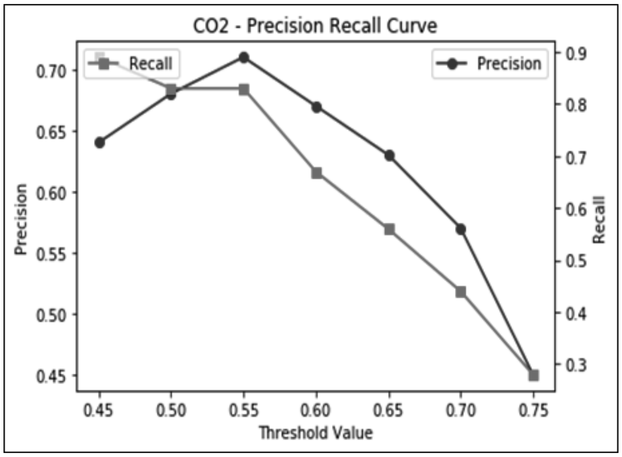


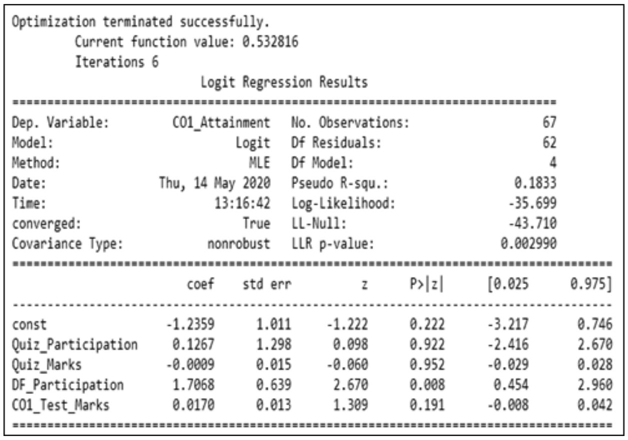
Fig. 10 : Precision – Recall Curve – CO2 Attainment

From the Fig. 10, it is seen that higher precision value (0.71) is obtained when threshold value is between 0.55; and the recall value is 0.83. The precision and recall value is same (0.67) when the threshold value is 0.6, which is the optimum threshold value, for CO2 attainment calculation.

It is evident that the EP target for CO1 can be in the range 70-75%, whereas it can be lowered to 60% for CO2. However, the same study can be done with the next batch of students before revising the target value.

F. Discussion

As there were 50% absentee for the test, the data was filtered by removing the absentee data and the model was re-built with all the four features 'p', 'q', 'r', and 's'. For this scenario too, it shows the feature 'p' as the most significant factor (p-value is 0.008), as shown in Fig. 11.



```
Optimization terminated successfully.
         Current function value: 0.532816
         Iterations 6
                           Logit Regression Results
==============================================================================
Dep. Variable:         CO1_Attainment   No. Observations:                   67
Model:                          Logit   Df Residuals:                       62
Method:                           MLE   Df Model:                            4
Date:                Thu, 14 May 2020   Pseudo R-squ.:                  0.1833
Time:                        13:16:42   Log-Likelihood:                -35.699
converged:                       True   LL-Null:                       -43.710
Covariance Type:            nonrobust   LLR p-value:                  0.002990
=====================================================================================
                        coef    std err          z      P>|z|      [0.025      0.975]
-------------------------------------------------------------------------------------
const                -1.2359      1.011     -1.222      0.222      -3.217       0.746
Quiz_Participation    0.1267      1.298      0.098      0.922      -2.416       2.670
Quiz_Marks           -0.0009      0.015     -0.060      0.952      -0.029       0.028
DF_Participation      1.7068      0.639      2.670      0.008       0.454       2.960
CO1_Test_Marks        0.0170      0.013      1.309      0.191      -0.008       0.042
=====================================================================================
```

Fig. 11 : Precision – Recall Curve – CO1 Attainment

In the set of continuous assessment activities, participation in Discussion Forum supports the learners to perform better in the terminal examinations. In this particular activity, the students have written their understanding and given their point of views on others' ideas. It helped them to write elaborate answers in the examinations. This addresses the research question (RQ1).

The participation in techtalk events and the follow-up quizzes help the students to give answers for 'remember' type questions like fill in the blanks, matching, and so on in the terminal examinations. Sometimes, this 1 or 2 mark is the most important factor which makes a student to make him CO attained or not. CO attainment is recalculated using quiz marks, and shown in Tables 4 and 5.

Table 4 : Details of Summative Assessment - Recalculated

CO	Type	Total students attended	Number of Questions	CO Direct attainment (in %)
CO1	Quiz	120	9	42
	CAT	67	3	34
	TE	131	2	66
CO2	Quiz	120	6	42
	CAT	67	5	30
	TE	131	1	60

Table 4 shows that 42% of students scored ≥70% in quiz. This quiz marks (10 marks) also considered as part of continuous assessment and CO attainment is re-calculated. It shows 1% improvement in the actual CO attainment, when compared with values of Table 2.

Table 5 : CO Attainment Re-calculation

CO	CAT (wt. 0.45)	TE (wt. 0.45)	Course Exit Survey (wt. 0.10)	Actual CO attainment
CO1	16.2%	29.7%	6.85%	52.7%
CO2	14.4%	27%	6.97%	48.3%

The significant improvement is not shown in the result. The various reasons could be that the students would not have considered quiz and test seriously; they would not have actively participated in it. From the results of Tables 3 and 5, it is inferred that participation in the follow-up activities of tech-talks and test has the significant role in CO attainment. This addresses the research question (RQ2). This analytics and results may be shared with the students and motivate them to participate in all activities, which in turn help them to score more marks in the terminal examinations.

5. Conclusion

The experimental work aims at systematic integration of active learning strategies and blended learning approach for the Software Engineering course. Detailed investigation on the performance of the learners clearly indicates that participation in discussion forum, seminars, and quizzes have a significant impact in increasing the learning outcomes. Logistic regression model has been used for classifying the students whether they would attain the expected target level or not, given their active participation in the continuous assessment, discussion forums, and quizzes. It seems that 60% accuracy resulted from the data considered; it may be improved if more participation is involved.

Students need to be motivated to participate in all the activities recommended by the instructor. Students' participation in continuous assessment has to be considered while calculating the CO attainment. Rigorous follow-ups with the students would give win-win situation to both students and the teachers – students would score better and the teacher would show better CO attainment for the course. This approach can trigger students' motive for learning beyond the curriculum and self learning.

Future work may focus on a study to develop a content delivery and project development model that narrows down the Industry–Academia gap in Software Engineering education.

References

[1] Alabbadi, A. & Qureshi, R.J. (2016). The Proposed Methods to Improve Teaching of Software Engineering. International Journal of Modern Education and Computer Science (7), 13–21.

[2] Alramouni, S. & Alkhawar, H. (2018). Strategies to Improve At-risk Students' Learning and Achievement: Undergraduate Software Engineering Course Case Study (pp.25-30). DOI: 10.1007/978-3-319-60011-6_3.

[3] Alva, H.H., Uma, B., Shruthi, V. & Saroja, C. (2018). Enhancing Learning Outcomes in Software Engineering Course through Problem Based Learning and Peer Assisted Learning. Journal of Engineering Education Transformations, DOI: 10.16920/jeet/2018/v0i0/120948 [accessed on April 2020]

[4] Bobrov, E., Buchhiarone, A. & Capozucca, A. (2019). Teaching DevOps in Academic and Industry: Reflections and Vision. DOI: 10.13140/RG.2.2.26837.27362 [accessed on May 2020]

[5] Chatley, R. (2016). Teaching Modern Software Development Techniques at University. https://www.infoq.com/articles/teaching-software-development-university/ [accessed on April 2020]

[6] Chen, J. Lu, H. An, L. & Zhou, Y. (2009). Exploring Teaching Methods in Software Engineering Education. In Proceedings of 4th International Conference on Computer Science and Education (pp. 1733–1738). Nanning.

[7] Cheng, Po-Hsun & Chen, L.W. (2018). Peer learning efficacy analysis on undergraduate software design course. Computer Applications in Engineering Education, 26 (1), pp. 5-16.

[8] Cummings, C., Mason, D., Shelton, K. & Baur, K. (2017). Active Learning Strategies for Online and Blended Learning Environments, USA, IGI Global, pp. 88-114.

[9] Demchenko, Y. Zhao, Z., Surbiryala, J & Shi, Z. (2018). Teaching DevOps and Cloud based

Software Engineering in University Curricula. http://www.uazone.org/demch/papers/dtw2019-devops-bok-course-v04.pdf [accessed on May 2020]

[10] Eison, J. (2010). Using Active Learning Instructional Strategies to Create Excitement and Enhance Learning. Jurnal Pendidikantentang Strategi Pembelajaran Aktif (Active Learning), 2 (1), pp. 1–10.

[11] Fonseca, V.M., & Gómez, J. (2017). Applying Active Methodologies for Teaching Software Engineering in Computer Engineering. In Proceedings of IEEE Revista Iberoamericana de Tecnologias del Aprendizaje, 12 (3), pp. 147-155.

[12] Huan, S. (2012). The Innovation of Software Engineering Course in the Non-Computer Professional. Procedia Environmental Sciences, (12), 1253–1256.

[13] Hubscher, T. & Narayanan, H. (2003). Constructive and Collaborative learning of algorithms. In Proceedings of the 344th SIGCSE Technical Symposium on Computer Science Education, pp.6-10.

[14] Lokare V. T., (2020). A Blended Approach for Teaching Laboratory Course: Internet Technology. Journal of Engineering Education Transformations, 34 (1), pp. 41-51.

[15] Maguire, P., Maguire, R., Hyland, P. & Marshall, P. (2014). Enhancing Collaborative Learning Using Pair Programming: Who Benefits. All Ireland Journal of Teaching and Learning in Higher Education (AISHE-J), 6(2), pp. 1411-1425.

[16] Mason, R.T., Masters, W. & Stark, A. (2017). Teaching Agile Development with DevOps in a Software Engineering and Database Technologies Practicum. In proceedings of Third International Conference on Higher Education Advance, HEAd'17, Valencia.

[17] Ramachandran, M. (2017). Technology Enhanced Active Learning in Software Engineering. In Proceedings of 9th International Conference on Computer Support Education CSEDU 2017 (pp. 242–248). PaulaEscudeiro. SCITEPRESS – Science and Technology Publications.

[18] Ramos, C.S., Kosloski, R. A.D. & Venson, E. (2018). TBL as an active learning-teaching methodology for software engineering courses. In Proceedings of the XXXII Brazilian Symposium on Software Engineering SBES '18 (pp. 289-297). ACM NewYork.

[19] Venson, E., Figueiredo, R., Silva, W. & Júnior, L.C.M.R. (2016). Academy-Industry Collaboration and the Effects of the Involvement of Undergraduate Students in Real World Activities. In Proceedings of IEEE Frontiers in Education Conference FIE 2016 (pp. 1-8).