ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 8(9), 828–833, May 2015*
DOI: 10.17485/ijst/2015/v8i9/53274

Extreme Pedagogy: An Agile Teaching-Learning Methodology for Engineering Education

Manoj Joseph D'Souza* and Paul Rodrigues

Department of Computer Science and Engineering, DMI College of Engineering, Chennai – 600123, Tamil Nadu, India; frmanoj@gmail.com, drpaulprof@gmail.com

Abstract

Traditional instructor-centered, lecture-based teaching methods in engineering education have been criticized for being too linear, dogmatic, systematic and constraining. This paper proposes 'Extreme Pedagogy', a student-centered teaching-learning conceptual framework to improve quality of engineering education which is built on four core values: students and teachers and their interactions, working knowledge, collaboration with students and responding to change. Extreme Pedagogy derives its philosophy from Extreme Programming, an agile software methodology. Extreme Pedagogy aims at continuous improvement of student learning, keeping students' needs and satisfaction as its focus.

Keywords: Engineering Education, Extreme Pedagogy, Extreme Programming, Teaching and Learning

1. Introduction

Over the last years, there has been mad rush in India among science students to join engineering colleges. As a result many engineering colleges have been started in many states to cater to the demand of the students. There has been outflow of thousands of engineering graduates each year. Many graduates who approach industries for placement are simply turned down because employers feel these graduates lack employability skills. The skill shortage is still one of the major constraints in most industries in India[1]. As per the industry reports only about 15% of college graduates have the needed skills to become employable in India. As per the National Assessment and Accreditation Council (NAAC) report of 2010, 62% of the Indian Universities and 90% of the colleges are average or below average[2].

The research findings suggest that engineering education institutions should: (i) seek to improve the skill set of graduates; (ii) emphasize soft skills, (iii) refocus the assessments, teaching-learning process, and curricula away from lower-order thinking skills, such as remembering and understanding towards higher order skills such as analysis and creativity; and iv) interact more with employers to understand the particular demand for skills in that region and sector[1]. Traditionally, instructional design methods have adopted the linear sequential ADDIE model which constitutes five phases:

1.1 Analyze

Identify instructional goals and tasks, analyzing learner characteristics; formative evaluation.

1.2 Design

Develop learning objectives, choose an instructional approach, define performance objectives, develop assessment instruments, develop instructional strategy.

1.3 Develop

Choose materials; design formative evaluation.

1.4 Implement

Deliver instructional materials; apply instructional activities; formative evaluation.

1.5 Evaluate

**Author for correspondence*

Summative evaluation.

The above model follows the similar steps of waterfall model of software development. Just as the waterfall model is linear, rigid and inflexible, ADDDIE model too was criticised for being too linear, dogmatic, systematic and constraining.

In response to overcome the drawbacks of the waterfall model, during the last few years software development methodologies have started embracing agile principles. All these agile software methodologies follow the same principles as defined in the Agile Manifesto[3] which basically states that for a project to be successful we should value: individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation and responding to change over following a plan. Extreme Programming (XP)[4] which is one of the most popular agile software development methodologies has become very successful in the software industry developing high quality software in response to the changing requirements of the customer. It follows twelve best practices to deliver software to the customer incrementally. It can be said that, the core practices of XP can very well be adapted to teaching and learning in engineering education. Extreme Pedagogy is a conceptual framework developed based on XP philosophy to improve the quality of engineering education by overcoming the many limitations of traditional pedagogy. The Extreme Pedagogy values: students and teachers and their interactions over administrative processes, working knowledge over grades, collaboration with students over fixed syllabi and responding to change over following a plan.

2. Extreme Programming Practices

2.1 Onsite Customer

In XP, customer is a part of development team and kept in the loop throughout the developmental process. Customer is always available to address business needs directly and accurately. The closer the customer to the team, the better it is for the developers to evaluate user stories and proceed development in the right direction. Presence of customer adds confidence to the team and also enhances customer satisfaction.

2.2 Small Releases

The purpose of small releases is to return customer's investment regularly. After every short iteration, a working system is delivered to the customer regularly and incrementally unlike in waterfall model where the product is released to the customer only at the end. This will help the customer evaluate the system and offer feedback for further development.

2.3 Planning Game

It is the meeting of customer and developers that takes place before the project is launched and after every successive iteration to negotiate between business requirements and technical feasibility. The planning game becomes very important as customer plays a vital role in the actual evaluation of the product.

2.4 Metaphor

As architectural designs are important for constructing buildings, so are metaphors for software products. Metaphors are nothing but simple analogies which can be easily understood by both customer and developers.

2.5 Simple Design

The purpose of this practice is to keep design as simple as possible to make it easy for future changes. This practice strongly opposes the inherent need of the programmers to solve future needs at present. The developer needs to ask always what is the simplest thing that could probably work.

2.6 Pair Programming

The purpose of pair programming is to share ideas, knowledge, experience and expertise. In XP, programming is always done in pairs, in which one takes the role of 'driver' who actually solves the problem with keyboard and mouse and the other partner takes the role of a 'navigator' who makes strategic decisions and keeps the big picture in mind.

2.7 Collective Code Ownership

The goal of this practice is to make the entire team responsible for the product. Everyone is the owner of the code. Any programmer can change any part of the code when it is needed. The changes done need to be integrated frequently.

2.8 Continuous Integration

The goal of this practice is to keep every team up to date with every other team as many teams are working in

parallel. For this, code from each team is frequently integrated in the central repository so that everyone is current with everyone's work. By integrating often the teams get frequent feedback to check whether changes actually work or not.

2.9 Coding Standards

When everyone speaks the same language it becomes easy to understand each other, similarly when all teams follow similar coding practices it becomes easy to understand the other and communicate effectively. This practice advocates uniform coding principles and practices.

2.10 Test-First Development

In this practice, test cases are written before the code is written. Tests are used as benchmarks to test the code. The principle behind this practice is to know what the task is before trying to figure out *how* to do it.

2.11 Refactoring

The purpose of refactoring is to maintain the code and design simple. By refactoring the code, all the complexities are removed and code is simplified keeping safe however the behaviour of the code. Refactoring is used to leave the code in a cleaner state.

2.12 40-Hour Week

The goal of this practice is to work hard but not too long. Working strenuously overtime kills creativity and as a result productivity is adversely affected. In a software development scenario this could result in high error rates in the code. Maintaining a schedule of eight hours for five days, helps the programmers to wake up refreshed every morning and retire to bed fulfilled.

3. From XP to Extreme Pedagogy

While moving from extreme programming to extreme pedagogy, we need to translate the fundamental concepts of software process such as *product, customer and developer*. The product of education is the learning that student acquires which can be understood in terms of knowledge, attitudes and skills. The primary customer of education is the student only because it is the student who pays for the education and gets the benefits of education. However we can also find many other secondary customers of education who are all stakeholders in education such as parents, sponsors, government etc. In software industry developers are the programmers who develop the product. To find out the developers in academia, the question that could be asked is: who are the developers of *learning*? Since teachers design instruction, assessment they can be called as developers. However, these days learning is very much student-centered in the sense that students themselves construct learning and teachers act as facilitators only. So students themselves are the developers of learning.

3.1 Best Practices of Extreme Pedagogy

From the twelve practices of XP, we take four of them and translate them into educational practices of extreme pedagogy. They are: 1. Onsite customer - Student involvement[2]. Test-driven development - Goal oriented teaching 3. Pair programming - Pair learning[4]. Small releases - Continuous assessment. In the following section, these practices are described in detail.

3.1.1 Student Involvement

In extreme pedagogy framework, student is a valuable resource and a customer of education. As customers of education, students need education that can impart employability as well as life-long learning skills. Student involvement is continuous and takes place at two different levels:

3.1.1.1 Formal Meeting for Course Design and Evaluation

Generally this meeting is conducted between faculty members and subject matter experts to make formal changes into an existing course or to design a new course. In this framework, students too become part of the meeting because it is the actual students who need to decide what is good or not good for them because they are the primary customers of education. Students' suggestions and feedback will be considered in making formal changes to the course.

3.1.1.2 Informal Meeting between Faculty and Students for Course Evaluation

This meeting between the faculty and students takes place during the course to evaluate the current course. Faculty members take the role of developers and students take the role of customers. In the meeting, students will assess whether course goals and objectives have been achieved

offer suggestions for improvement. Faculty members in turn, estimate the feasibility of implementing student suggestions within the framework of available resources, scope, quality and time schedule. This planning meeting is a periodic activity which can help to bring about informal changes in the course.

3.2 Goal-Oriented Teaching

In extreme pedagogy, goal oriented teaching is spiral and incremental. It follows four steps. First learning goals and objectives are explained to students before every lecture. Second, instructional content is carefully designed to bring out learning goals and objectives. Third, instruction is delivered making use of variety of methods such that students find lecture interesting, motivating and useful. Fourth, it is very necessary for the instructor to assess whether students have attained the objectives of the lecture, so a short evaluation in the form of formative assessment is conducted at the end of the class. From the formative assessment, the instructor can make out the areas where a student has failed to achieve these objectives and the areas where the student has achieved success and accordingly revise the instruction[5]. The structure of goal oriented teaching is explained the in the Figure 1.

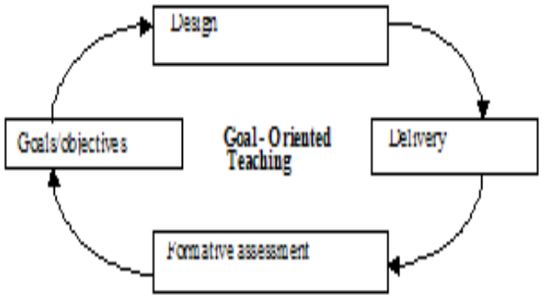

Figure 1. Goal-Oriented Teaching.

3.3 Pair Learning

In XP, all the coding is done by two programmers sitting side by side using keyboard and monitor. In extreme pedagogy, all academic tasks such as programming, laboratory experiments, projects, solving mathematical problems etc. are done by pairs of students working together. One takes the role of a 'driver' who actually solves the problem and other takes the role of a 'navigator', who carefully observes his partner, offers suggestions, corrections, provides ideas etc. The roles of the driver and navigator are swapped periodically. Pair learning is special kind of cooperative learning. The usefulness of cooperative learning in engineering education is well documented by the research studies. Johnson et al[6] who did much research on the usefulness of cooperative learning, identified five important elements of co-operative learning: positive interdependence, promotive face-to-face interaction, individual accountability, social skills and group processing. As pair learning is a special kind of cooperative learning, all the above elements of cooperative learning will also be applicable to pair learning.

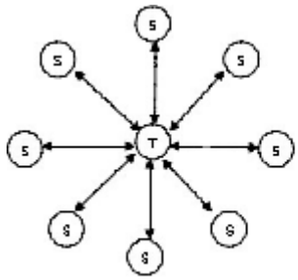

Figure 2. Teaching-learning in a traditional classroom (T = Teacher, S = Student).

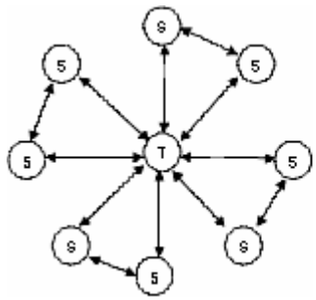

Figure 3. Teaching-learning in an Extreme Pedagogy classroom.

3.4 Continuous Assessment

Educationalists and others would agree with Brown and Knight[7] when they affirm that assessment is central to the student experience. Likewise, Gibbs[8] states that assessment frames learning. The study[9] shows that continuous assessment has the potential to support student learning through feedback and to increase students' motivation for learning.

As in XP, the product is delivered to the customer incrementally in short cycles, student learning is assessed periodically for grading. Unlike having one long durational exam at the end of the course, many summative

assessments are conducted throughout the course. Continuous assessments can take various forms such as tests, coursework, quizzes, project work, seminar presentation etc. Having frequent assessments reduces learner stress and anxiety and also helps in retention of subject matter by constant reinforcement.

4. Extreme Pedagogy Characteristics

Extreme pedagogy is governed by three characteristics. They are: 1. Learning by continuous doing. 2. Learning by continuous collaboration and 3. Learning by continuous testing.

4.1 Learning by Continuous Doing

Traditionally instructional methods have been heavily lecture-based and students have to listen to lectures passively. Such methods have been criticised for failing in their effectiveness by educational researchers[10,16,17]. To sustain interest and motivation, students have to be actively engaged in the classroom. Educational researchers have realised importance of teaching and learning for understanding than for mere knowledge retention[11]. Instructional activities must help the students to move from rote memorisation of facts known as "surface learning" toward "deep learning" where knowledge is constructed through active and constructive processes[11]. Learning by continuous doing involves all active learning methods which make the students to actively participate in the class. Active learning is an umbrella term for methods focusing on student activities and student engagement in the learning process[12]. According to Bonwell[13], active learning methods are instructional activities involving students in doing things and thinking about what they are doing. By continuous, it is meant that the active learning methods should be implemented in every class and not occasionally. Even when there is lecture component, it is necessary to conduct short and meaningful activities in between where students are actively engaged in learning process. Extreme pedagogy has pair learning practice where students work in pairs on an academic task or activity designed by the instructor.

4.2 Learning by Continuous Collaboration

Collaborative and cooperative learning are well researched methods in education. Both these methods describe situations in which two or more people learn together toward a common goal. Unlike learning individually, in a collaborative learning environment, students continually interact with each other, share ideas, knowledge, skills and contribute to the success of everyone in a group. The research has suggested that collaborative learning promotes deep learning, encourage self-esteem and the acceptance of others, and improves interpersonal effectiveness. According to promoters of collaborative learning, when students actively exchange, debate and negotiate ideas within their groups increases students' interest in learning. Importantly, by engaging in discussion and taking responsibility for their learning, students are encouraged to become critical thinkers[14]. Many researchers have reported that students working in small groups tend to learn more of what is being taught. Moreover, they retain the information longer and also appear more satisfied with their classes[15]. Collaborative method of extreme pedagogy is quite unique and called 'pair learning'. In most of the collaborative learning methods, groups are generally made of size three or more. When there are many members in a group, it can happen that only a few dominate and do much of the work and others remain passive. In pair learning, since there are only two members, the responsibility for learning is equally shared between each other. The unique feature of learning by continuous collaboration is that there is continuous collaboration between student and student as well as teacher and students. The interactions are continuous in the sense that they take place throughout the course. The collaborative learning promotes teamwork and communication. It is found in the research that, employers perceive soft skills such as teamwork and communication more important than professional skills[1].

4.3 Learning by Continuous Testing

Testing is a widely used method in education to audit learning. Most educators and students consider testing a tool for assessing student learning and providing feedback to guide future activities. The most used methods of testing in education are called formative assessment and summative assessment. According to Black and William formative assessment includes, "all those activities undertaken by teachers, and/or by students, which provide information to be used as feedback to modify the teaching and learning activities in which they are engaged".

In contrast, summative assessment, seeks to monitor educational outcomes, often for purposes of external accountability.

What is unique in learning by continuous testing in Extreme Pedagogy is that testing is very frequent and can take place even on a daily basis. It will give valuable feedback to the faculty to revise their teaching and also help the students to monitor their own learning by taking self responsibility. Extreme pedagogy uses both formative and summative assessments and takes them to extreme level.

5. Conclusion

Much of the teaching in engineering colleges is lecture-based and teacher-centered. Faculty members are over concerned with completing the pre-designed syllabus, without giving sufficient attention to students' needs, learning styles, learning pace, learning outcomes etc. Students turn out to be passive listeners, memorise the subject content to get good grades in the exams without actually developing skills needed.

Extreme pedagogy shifts focus from teacher-centered instruction to student-centered instruction, from teaching to learning. As a learner-centered pedagogy, its focus is on the students, student learning, student satisfaction, student motivation. Unlike traditional pedagogies, which are rigid and inflexible, extreme pedagogy is flexible and adaptable to cater to students' needs always.

Extreme pedagogy is a novel framework for teaching and learning in engineering education. Its concepts and practices can be easily implemented in classroom teaching in engineering colleges. Its core practices are drawn from XP and mapped to the best practices in existing teaching-learning methodologies in education. Extreme pedagogy framework intends to bring about innovation in the classroom teaching. Its main purpose is to improve the quality of learning in engineering education and address many of the limitations of the traditional teaching-learning methods.

6. References

1. Blom A, Saeki H. Employability and skill set of newly graduated engineers in India. Policy research working paper 5640. Washington, DC: World Bank; 2011.
2. Federation of Indian Chambers of Commerce and Industry and Ernst and Young. Higher education in India: Twelth Five Year Plan (2012–2017) and beyond. 2012. Available from: http://www.ey.com/Publication/vwLUAssets/Higher_Education_in_India/$FILE/EY-FICC_Higher_Education_Report_Nov12.pdf
3. Agile Alliance. Agile manifesto. 2001. Available from: http://www. agilemanifesto.org.
4. Beck K. Extreme programming explained: Embrace change. Addison-Wesley; 1999.
5. Black P, William D. Assessment and classroom learning. Assess Educ Princ Pol Pract. 1998; 5(1):7–73.
6. Johnson DW, Johnson RT, Smith KA. Active learning: Cooperation in the college classroom. Edina, MN: Interaction; 1991.
7. Brown S, Knight P. Assessing learners in higher education. New York, NY: Psychology Press; 1994.
8. Gibbs G. How assessment frames student learning. In: Bryan , et al., editors. Innovative assessment in higher education. New York: Routledge; 2006.
9. Hernandez R. Does continuous assessment in higher education support student learning? Higher Education. 2012; 64(4):489–502.
10. Barr RB, Tagg J. From teaching to learning: A new paradigm for undergraduate education. Change. 1995; 27(6): 12–25.
11. Ritchhart R, Church M, Morrison K. Making thinking visible: How to promote engagement, understanding, and independence for all learners. San Fransisco, CA: Jossey-Bass; 2011.
12. Prince M. Does active learning work? A review of the research. J Eng Educ. 2004; 93(3):223–31.
13. Bonwell CC, Eison JA. Active learning: Creating excitement in the classroom. Washington, DC: School of Education and Human Development, George Washington University; 1991.
14. Totten S, Sills T, Digby A, Russ P. Cooperative learning: A guide to research. New York: Garland; 1991.
15. Chickering AW, Gamson ZF. Applying the seven principles for good practice in undergraduate education. San Francisco: Jossey-Bass; 1991.
16. Janagam D, Suresh B, Nagarathinam S. Efficiency of task based learning and traditional teaching on self-regulated education. Indian Journal of Science and Technology. 2011; 4(3):308–12.
17. Srikumar C, John PJ, Priya V. An evaluative study on comparison of problem based learning and lecture based pedagogy on self directed learning in undergraduate medical education. Indian Journal of Science and Technology. 2009; 2(12):59–67.