A Study of the Effectiveness of Case Study approach in Software Engineering Education

by

Kirti Garg, Vasudeva Varma

in


IEEE Press.

Report No: IIIT/TR/2008/61

Centre for Software Engineering Research Lab
International Institute of Information Technology
Hyderabad - 500 032, INDIA
June 2008
A Study of the Effectiveness of Case Study approach in Software Engineering Education

Kirti Garg, Vasudeva Varma
International Institute of Information Technology, Hyderabad, India
kirti@research.iiit.ac.in, vv@iiit.ac.in

Abstract

Software Engineering (SE) educators have been advocating the use of non-conventional approaches for SE education since long. In this context, we conducted action-research to compare the effectiveness of a case study approach with conventional lecture based approach.

We designed and taught a first course in SE, that involved case study approach as well as the traditional lecture based approach. We recorded and analyzed student's perception of learning over using well defined parameters that reconciled with cognitive, skills and meta-cognitive goals of SE education.

Results corroborated that case study approach is more effective and interesting for learning SE than the lecture based approach. These results indicate that academia and industry should further explore learning-by-doing paradigm, especially the case studies. Benefits of approach include bridging of the industry-academia gap and creation of professionals who are well versed with theory and practice and have experienced the intricacies of real software development even before entering the industry.

This paper provides empirical data to support that case study approach is very effective in SE education and hence useful for curriculum designers. This work is useful for SE educators and researchers as it describes methodology for rigorous scientific educational research.

1. Introduction

Software Engineering Education (SEEd) is witnessing a positive change. Innovations and improvements in curriculum, instruction and assessment are being directed towards bridging the academia-industry gap by projecting the true nature of software development and facilitate the student in nurturing essential knowledge, skills and attitude, that are actually needed by the industry[2, 16, 17, 19].

Software Engineers often face decision dilemmas and need to analyze the problems, evaluate various alternatives and apply their knowledge and skills (computer science, Software Engineering and domain related) to develop an effective solution [12, 19,20]. They use the proven practices (aka best practices) to improve the effectiveness of their solutions. They usually work in collaboration and should exhibit good communication skills and professionalism. Software Engineers need to acquire new knowledge and skills at various points in their career and hence the ability to learn on their own (self-directed learning) is important.

Considering the nature of SE, we feel that a SE educational endeavor should focus only on SE concepts, but also on higher order cognitive skills of application, analysis, evaluation and synthesis [5] as well as other abilities such as communication skills and self-learning. Students should understand the complex and evolving nature of SE and get exposed to professional practices during the instruction itself.

Many SE educators, both academicians and industrial trainers are moving away from the conventional class room or lecture based instruction and showing interest in non-
traditional pedagogies such as projects, e-learning, industrial projects and internships, teaching through case studies etc [8, 12, 17, 10, 19]. Though the approaches differ, the target is same i.e. imparting SEEd in effective manner.

This research paper documents results from our action research with a first course in Software Engineering for Computer Science majors. In addition to lectures, we used case studies for instruction and assessment. We will discuss only the instruction related aspects in this paper. The goal was to impart students with the right set of knowledge and skills required for a Software professional. This systematic research gave us useful insights into the suitability and effectiveness of case studies for SEEd as well as the nature of Software Engineering education.

The further sections we will discuss the current approaches for SEEd, the case study approach and its potential advantages. We will explain the action research done at International Institute of Information Technology, Hyderabad, and will discuss the research method, results and observations. This was initial efforts and hence we have several ideas for further exploration, which are discussed in the future research section.

2. Current Approaches

The conventional and the most prevalent approach to Software Engineering Education is a lecture based paradigm, sometimes accompanied by course projects. A major problem with this paradigm is that a student is usually a passive listener during lectures and doesn't get actively involved in the learning process.

Research done by learning scientists and cognitive psychologists has proven that this kind of learning is not effective [14]. Lectures suggest a reactive approach where the students are expected to react to a solution presented as opposed to proactively thinking about the problem on hand. This typically results in their settling for short term goals such as getting a good grade in the course instead of leveraging on long term goals of learning.

Our past experiences with use of academic projects to teach SE to computer Science majors show that students focus more on programming issues and very little on the development process and the associated SE issues. We also received the feedback that the students felt that SE is a theoretical and unexciting subject and is perhaps of very little use in future. Many a times students do not experience the intricacies related with a project [15,19]. This disconnection from the industry another great worry for the academia as well as the industry [3, 11, 16].

Owing to the unsuitability of these paradigms for SEEd, educators are looking for alternative methods, conventional and non- conventional, that can make SEEd more effective and interesting. Bonwell, and later Sivan, strongly suggested that students can learn more effectively when they get actively involved in the learning process [6, 18, 13].

3. Case Study Approach

Bromley defined case study as a "systematic inquiry into an event or a set of related events which aims to describe and explain the phenomenon of interest" [7, p. 302]. We define a Software Engineering case as "an account of a Software Engineering (development) activity, event or problem containing background and complexities actually encountered by a software engineer".

In a case study based approach, students are given a hypothetical or a real problem. The case includes background of people and organizations involved, their nature of
work, challenges they face and decisions they make. Usually some challenge(s) is presented as an exercise that needs to be ‘solved’. Finding solution to these challenges involve understanding the people, organization, past events and the challenges, and living through the experience of the protagonist and other actors in the case study. This further involves analysis of problems, application of concepts, tools, techniques and skills, evaluation of alternatives and decision making.

Usually students work in teams and different teams may come up with different solutions. These solutions are discussed by the entire class. Students analyze various solutions and present their views. Discussions facilitate learning and communication skills. Sometimes students may be asked to submit a detailed report of their solutions, which can be used for assessment.

Thus Case studies as learning instruments facilitate “thinking and acting forward from first principles”. [19, 20]. Hilburn strongly says that “Case studies are of special value in problem-based learning, which concentrates on the development of problem-solving skills, self-directed learning skills, and team skills”[12]. These are the capabilities that we look for in a Software Engineer, and hence we can say that a case study approach suits y=our purpose very well. Thus we strongly suggest that case studies can be a powerful learning and teaching instrument for SEEEd.

4. Software Engineering Course using Case Studies

We offered a first course in SE to Computer Science majors (undergraduate, graduate students and some industry participants). For most undergraduate students, this course is the only opportunity to learn SE before they start their career as Software Engineers.

This course used case studies for instruction and assessment. All case studies were drawn from real life projects executed in industry. Typically each case study had a co-author from the industry who actually worked on the project being described in the case study.

4.1. Learning Goals

Like every educational endeavor, we had well defined learning goals for the course, distributed as knowledge, skills and dispositions. These goals map with cognitive goals for education [5]. Some of the major learning goals of this one semester course were as follows:

- **Knowledge and understanding of SE discipline, its’ nature and scope:** Understanding processes, models & SDLC, Requirements Engineering, Design & architecture, verification & validation (testing and reviews), implementation, security, release & configuration management, software quality, project management & maintenance. For each topic and its sub-topics, the related concepts, scope, major issues, best practices, practical considerations also needs considered. These contents were heavily guided by Software Engineering Body of Knowledge (SWEBOK) [1].

- **Skills:** The major skills include UML, documentation, others related to the knowledge and understanding goals. team-work, self-directed learning and communication, problem solving skills namely analysis, evaluation and application.

- **Dispositions:** Problem solving attitude, understanding that there can be more than one correct solution to same problem, and there are wrong answers.
Table 1. Effect of Learning Activities on Learning Goals

<table>
<thead>
<tr>
<th>Activity</th>
<th>Learning Goals Served</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attending Lectures and tutorials</td>
<td>Concepts, knowledge and skills, active listening, integrating factual knowledge, holistic picture of SE (synthesis)</td>
</tr>
<tr>
<td>Doing Assignments (mostly written)</td>
<td>reinforcing concepts, skills (example: UML diagrams, writing test cases, analyzing requirement specification documents etc), communication skills</td>
</tr>
<tr>
<td>Preparing Case Study solution</td>
<td>Acquiring concepts and knowledge (domain, SE and problem solving), applying concepts, building and applying skills, critical thinking and analytical skills, evaluation, seeing relationships between ideas (synthesis)</td>
</tr>
<tr>
<td>Discussing case solution in class</td>
<td>Reinforcement of concepts and knowledge, Critical thinking and analytical skills, seeing relationships between ideas (synthesis), evaluation, communication skills</td>
</tr>
<tr>
<td>Writing detailed reports</td>
<td>Written communication skills</td>
</tr>
<tr>
<td>Writing Reflections Report</td>
<td>Communication skills, analytical skills</td>
</tr>
<tr>
<td>Presenting Case study solutions</td>
<td>Communication skills</td>
</tr>
</tbody>
</table>

4.2. Operational Details

Both lectures and case studies were used for teaching. One lecture and at least one case study was devoted to each major topic. A few cases covered multiple topics. For such cases, participating students were asked to focus on one particular aspect. It is essential to remind that a single case study cannot cover entire syllabus. A case study usually has a primary focus or a goal and multiple secondary goals. This reflects the complex nature of real world situations where issues do not occur in isolation.

A case study was posted a week in advance so that students can go through it before coming to the class. The lecture focused on the relevant concepts and skills for the topic. The case study was also presented. Two or three teams were randomly selected to solve the case and present their solutions to the class. Each team had 4-5 students, a mix of undergraduate and graduate students.

Select teams did an extensive study for a week to come up with an appropriate solution. They could meet the instructor/TAs during the tutorial to obtain clarifications. Teams presented their solutions the following week to the entire class whole class discussed the solution. The instructor facilitated the discussion and gave his comments, and views on each solution. The instructor refrained from giving an ideal answer or the ‘teacher’s solution’.

This solution presentation and discussion was a 30 minutes exercise for each team. Once discussed, teams were given a week to improve/modify their solutions in light of the class discussion, and further submit a detailed solution report. Rest of the class submitted a ‘Reflections’ report that recorded their learning, criticism, appreciations, alternate solutions for the case study solutions. The Reflections and the detailed solution reports were used for assessment. The case solutions and reflections weigh to 40% of the total grades. Written exams and assignments weigh another 50% and class
participation, judged by the questions asked and issues raised during various discussions, weigh the rest.

4.3 Learning Activities

The learning activities involved in the course can be classified as conventional lecture based approach activities (Attending lectures and tutorials, doing reading and assignments) and case study activities (preparing case study solution, presenting case study solutions, participating in class discussions, writing detailed solution reports and writing Reflection reports). Table 1 lists these activities and their purpose in light of course learning goal given in section 4.1.

5. Research Model

This course was conducted as part of the action research and was designed to gauge and compare the effectiveness of lectures and case studies for SEEd in a scientific and systematic manner.

The measure of effectiveness we used was “Achievement of learning goals”. We used triangulation of assessment measures for this. The assessment measures were student’s perception of learning and actual performance. For this paper we will present the results from the student’s perception of learning only. Studies have found correlation between the student’s perception and actual performance and hence we can use the measure as well without risking the internal validity. This perceived learning is an important measure for a SE course as many computer science students considered SE as ‘one of the necessary evils’. They believed that a SE course is focused on theory and may not be useful or exciting as other core Computer Science courses such as programming, algorithms or data structures. Thus unless students understand that a SE course included more than just theory, it would be have been difficult to carry the benefits of learning in future.

Hiltz describes a number of parameters that record student’s perception of learning [adapted from 9, 14, 21]. These parameters overlap with Bloom’s taxonomy of education goals for cognitive dimension. We use a version of parameters, modified as per SE domain and derived from course learning goals.

5.1. Method

Students filled a questionnaire at the end of the course where we asked their opinion about various learning activities (corresponding to lectures and case study approach) they performed during the course helped them achieve various learning goals. A five point Likert scale was used i.e. 5 = Significantly, 4 = Above average, 3= Average, 2 = Below average, 1= Well below average.

186 students responses from two course offerings were collected and analyzed. Mean data imputation was used to replace missing values. A Wilcoxon Signed Rank Test [4] was applied to check if the difference between the perceived learning for two approaches is significant. We selected Wilcoxon Signed Rank test as it is equivalent to paired t-test, but does not assume that the data is normally distributed.

5.2 Hypothesis

We expected that the students will find the case study approach more effective as compared to conventional lectures. Thus the null hypothesis was as follows: There is no significant difference between the students perception of learning when following a
traditional lecture approach and when following a case study approach. Group Lectures learning activities represent the traditional lecture approach and group Case Studies represent the case study approach.

5.3 Results and Inferences

Table 2 summarizes the results we got by applying the Wilcoxon’s signed rank test to compare the two approaches across all learning parameters:

<table>
<thead>
<tr>
<th>Learning Parameters</th>
<th>Lectures</th>
<th>Case Study Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Helped me to gather and understand SE concepts</td>
<td>3.23</td>
<td>3.55</td>
</tr>
<tr>
<td>Improve my communication skills</td>
<td>2.57</td>
<td>3.17</td>
</tr>
<tr>
<td>Improved my ability to critically analyze</td>
<td>2.78</td>
<td>3.35</td>
</tr>
<tr>
<td>Improve my ability to apply the concepts and skills learned in course</td>
<td>2.36</td>
<td>3.16</td>
</tr>
<tr>
<td>Helped me to see a holistic picture of SE</td>
<td>2.69</td>
<td>2.94</td>
</tr>
<tr>
<td>I found interesting</td>
<td>2.53</td>
<td>3.25</td>
</tr>
<tr>
<td>My overall experience</td>
<td>3.24</td>
<td>3.63</td>
</tr>
</tbody>
</table>
We see that the case studies score higher than the lectures for the various learning parameters, as well as for interestingness and overall experience of the students. Thus we can conclude that case study approach faired to be more effective as compared to the conventional lecture based approach.

5.4 Other Observations

Students reported that they could relate to many of the case studies. For example, after discussing software processes case study, where the case protagonist moves from a small software development firm to a CMM level company and gets overwhelmed by the processes, one of the industry student commented “This is my story”.

The graduate students, based on their undergraduate experience and undergraduate students, with impressions transferred by their seniors, had initial perception that Software Engineering is purely theoretical or “hands-off” course. We observed a dilution in this perception. Some students started using UML and a few of the Software Engineering best practices for project work in other courses. These included the use of coding standards & guidelines and designing the systems prior to coding.

Initially we collaborated with a couple of companies for case study development. As these organizations observed the course in execution and the subsequent results, they showed a lot more interest and were more forth coming in contributing to our case study initiative. They shared information about their other projects and initiated their own case study based training programs.

Typically case study approach or any non-conventional approaches have been found most effective in smaller classes. Even though we had large classes (ranging between 100-130 students), we got compelling results indicating the effectiveness of case study approach.

An interesting issue when designing a SE course with case study approach was the alignment of goals, instruction and assessment. Here not only the course curriculum, but even the instruction driven by case studies is supporting the program goals.

6. Future Directions

This experiment was conducted under several constraints like being a first course in Software engineering, large number of students, presence of both undergraduate and graduate students in same class, pre-defined course objectives etc. We plan to handle some of these constraints in subsequent experiments by ensuring more support from administration. We also plan to compare case studies with other approaches such as project based and e-learning through controlled experiments. The experiment is now being repeated for an advance level software engineering course as well as in a corporate training environment. Further we are in the process of developing a concrete and comprehensive measure of effectiveness of SE education.

7. Conclusions

The Case study approach provides a platform for learning the concepts, skills, tools and techniques in presence of a context where the instructor and student get engaged in a meaningful manner. In this approach, learning is student centric and active. This experiment gives compelling indications that the problems identified in several conventional and non-conventional learning models can be effectively addressed by the case study approach. We also believe that the case study approach will be the right tool for creating an effective feedback and assessment method in such active learning
environments. We will further research and evaluate the ways to extend the application of this approach to create more effective and practical learning environments. In conclusion, we advocate creation of learning environments that expose to and prepare the students for issues and challenges in the real world in a proactive and effective manner by using problem based learning and case studies in particular.

8. References