ABSTRACT

In this report, we present a summary and a few reflections on the one day workshop on Software Engineering Education held on February 22, 2012 at Indian Institute of Technology, Kanpur, India collocated with the 5th India Software Engineering Conference. We identify a gap and believe there is a need for creating an annual discussion forum that serves the need of having a regular workshop for software engineering education in India and also benefit the global software engineering education community by sharing the workshop insights and results by a publication process. The workshop consists of two keynotes, one from academia and one from industry, two subgroups discussions and presentations by the subgroups on their discussions. Three systematic techniques, invitation of position statements, set up of a Google group and an online survey, were employed before the workshop to estimate number of participants, subgroups and size of each subgroup for effective discussions. Twenty participants attended the workshop. The keynotes were on Using Collaborative Learning and Divergent Thinking to Teach Software Engineering and on Software Engineering Competency Development Model. Three topics were selected for subgroups discussions by the participants: use of various methods, such as learning while playing and project-based software engineering, over Power Point lecture, requirements and needs of undergraduate software engineering degree program from the perspective of Indian software industry and curriculum content, coverage, and impact of software engineering courses. The workshop was a successful endeavor and the response in terms of the contributions by participants is a clear indicator and confirmation of the need of having a focused discussion forum for brainstorming on software engineering education in India.

Categories and Subject Descriptors
K.3.2 [Computers and Education]: Computer and Information Science Education.

Keywords
Software engineering education.

1. MOTIVATION AND OBJECTIVES

The growing software industry in India has created a huge demand for software engineering education (SEE) and training. The increasing demand for software engineers in the society and in particular by the Indian Information Technology (IT) Industry has created a strong need to create a platform where software engineering (SE) educators and researchers can come together and discuss issues, challenges and opportunities in SEE. The topic of SEE has attracted several researchers’ attention and is an area on which a substantial body of knowledge is available. While there are prestigious conferences and workshops held on a regular basis in several countries extending the state-of-the-art in the field of SEE, there is no such focused workshop or conference in India. Few conferences and workshops on SEE are: annual Conference on Software Engineering Education and Training (CSEET) since 1987 [1], Software Engineering Education Workshop [2] collocated with the 21st Australian Software Engineering Conference (ASWEC) 2010, 1st International Workshop on Software Engineering Education based on Real-World Experiences (EduRex) [3] collocated with the 34th International Conference on Software Engineering (ICSE) 2012, Software Engineering Education Workshop [4] collocated with the 14th Asia Pacific Software Engineering Conference (APSEC) 2007 and International Workshop on Software Engineering Education (IWSEE) [5]. We believe there is a need for creating an annual discussion forum that serves the need of having a regular workshop for SEE in India. In an attempt to fill the gap, we organised Software Engineering Education Workshop (SEEW) which was held on February 22, 2012 at Indian Institute of Technology, Kanpur, India. The SEEW 2012 was collocated with the 5th India Software Engineering Conference (ISEC) [6]. The ISEC is an annual conference on SE promoted by the Special Interest Group on Software Engineering (SIGSE) under the umbrella of the Computer Society of India since 2008. One full day was allocated for the workshop. In this report, we present a summary and a few reflections on the SEEW 2012.

The main objectives of the workshop were to bring together educators, researchers and practitioners from academia and industry to discuss and advance current state of SEE and to facilitate and promote group-subgroups discussions on various topics of SEE. A set of recommendations is expected at the end of the workshop or the group-subgroups can plan and outline draft papers to complete later for publication in the benefit of global SEE community.

2. PROCEEDINGS

The SEEW 2012 was organized by the authors of this report with support from ISEC 2012 conference organizers, student volunteers and contributions from several colleagues of the organizers. The link to the workshop website is https://sites.google.com/site/softwareengageducation/.

The format and structure of the workshop is inspired by the Software Engineering Education Workshop [2] collocated with the 21st Australian Software Engineering Conference (ASWEC) held in Auckland, New Zealand on April 6th 2010. Position statements were invited from participants, which were reviewed by the organizers before the workshop. The SEEW 2012 was divided into two sessions, each session with a keynote for one hour, a subgroups discussions for one hour followed by presentations of the subgroups to the group of all participants, consists of two keynotes, one from academia and one from industry, two subgroups discussions followed by presentation by each subgroup. After a brief welcome and introduction to the workshop by Rakesh Shukla, the first speaker, Jeff Offutt, was introduced, by Rushikesh Joshi, to deliver a keynote talk. After the first keynote, a list of proposed topics was displayed on the projector screen so that participants select topics of their interest for subgroups discussions. Participants were requested to introduce themselves with their affiliation and provide brief reasons why they choose particular topics for discussion. Three topics were selected for subgroups discussions after a scoring from each participant’s topics selection. Participants were requested to form three subgroups to discuss their selected topic for one hour and then one/two representative(s) of each group presented summary and highlights of their discussions to the group. Then the second session was repeated with the same kind of activities. The second keynote, Subraya B M was introduced, by Rushikesh Joshi, to deliver a talk followed by discussions within the three subgroups on the selected topics and their presentations to the group. At the end, vote of thanks was delivered by Rakesh Shukla.
Table 1. List of participants

<table>
<thead>
<tr>
<th>Sr. #</th>
<th>Name</th>
<th>Affiliation</th>
<th>Participated in</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Anand Bajpai</td>
<td>Indira Gandhi National Open University, Lucknow</td>
<td>K1  S1  K2  S2</td>
</tr>
<tr>
<td>2</td>
<td>Ashish Sureka</td>
<td>Indraprastha Institute of Information Technology, Delhi</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>3</td>
<td>Indu Chawla</td>
<td>Jaypee Institute of Information Technology, Noida</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>4</td>
<td>Jeff Offutt</td>
<td>George Mason University, USA</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>5</td>
<td>Rakesh Shukla</td>
<td>Infosys Limited, Bangalore</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>6</td>
<td>Ramrao Wagh</td>
<td>Goa University, Goa</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>7</td>
<td>Richa Sharma</td>
<td>Shiv Nadar University, Noida</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>8</td>
<td>Rushikesh K. Joshi</td>
<td>Indian Institute of Technology, Mumbai</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>9</td>
<td>Subraya B. M.</td>
<td>Infosys Limited, Mysore</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>10</td>
<td>T. S. Mohan</td>
<td>Infosys Limited, Bangalore</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>11</td>
<td>Veena Bansal</td>
<td>Indian Institute of Technology, Kanpur</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>12</td>
<td>Y. Raghu Reddy</td>
<td>International Institute of Information Technology, Hyderabad</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>13</td>
<td>Nenad Medvidovic</td>
<td>University of Southern California, USA</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>14</td>
<td>Dheeraj Sanghi</td>
<td>Indian Institute of Technology, Kanpur</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>15</td>
<td>Keshav Nori</td>
<td>International Institute of Information Technology, Hyderabad</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>16</td>
<td>Philippe Dugerdil</td>
<td>University of Applied Sciences, Switzerland</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>17</td>
<td>Ashish Agrawal</td>
<td>Indian Institute of Technology, Kanpur</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>18</td>
<td>Balwinder Sodhi</td>
<td>Indian Institute of Technology, Kanpur</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>19</td>
<td>Ravi Gorthi</td>
<td>LNM Institute of Information Technology, Jaipur</td>
<td>Y  Y  Y  Y</td>
</tr>
<tr>
<td>20</td>
<td>T. V. Prabhakar</td>
<td>Indian Institute of Technology, Kanpur</td>
<td>Y  Y  Y  Y</td>
</tr>
</tbody>
</table>

Abbreviation :: K1= Keynote 1; S1= Subgroups discussions 1; K2= Keynote 2; S2= Subgroups discussions 2

Table 1 shows the list of participants with their attended sessions. Twenty participants, including three organizers and two keynote speakers, took part in the various activities. Participants are either faculty members or researchers from academia and industry. Most of them are from India, two from the US and one from Switzerland. None of them is practitioner working in software development industry with or without formal degree in SE or related discipline and/or have experience to deal with recent graduates. Practitioners’ viewpoints were missed in the discussions.

Fourteen participants have participated in subgroups discussions, fifteen have attended the first keynote and sixteen attended the second keynote.

3. KEYNOTES

Two keynote speakers, one from international academia and one from industry talked on two different topics and illustrated recent practices of SEE in academia and in industry.

3.1 Using Collaborative Learning and Divergent Thinking to Teach Software Engineering

Prof. Jeff Offutt, Professor of Software Engineering from the Volgenau School of Information and Technology, George Mason University, USA, shared his experience of using divergent thinking, collaborative learning and multi-faceted evaluation to teach SE. He started his talk with definitions, differences and goals of science and engineering. Then he described how computing is different than engineering. He discussed separation of various engineering disciplines from physics to illustrate how SE is similarly separating from mother computer science (CS) and becoming established as a separate, independent, discipline. He also defined CS and SE.

Educators should differentiate SE from CS in three specific ways: by (1) divergent thinking; (2) collaborative learning; and (3) multi-faceted evaluation. Divergent thinking is a creative thought process that generates more than one solution to the same problem. Collaborative learning is two or more people try to learn something together and multi-faceted evaluation is using two or more criteria to evaluate a solution or idea. CS courses emphasize single solutions, developed by students working alone and evaluate primarily by efficiency. Divergent thinking and collaborative learning are essential abilities for practicing engineers. Divergent thinking is encouraged by assigning problems that have many solutions, and evaluating solutions with multiple quality attributes. Collaborative learning is promoted by having students work in teams and by openly discussing technical problems and possible solutions.

He presented three case studies that illustrate how divergent thinking, collaborative learning and multi-faceted evaluation are integrated into a second year course, a fourth year course and a graduate seminar course at GMU. Divergent thinking is introduced into SEE projects early in the second year by teaching usability. The students are encouraged to develop solutions that have different user interfaces, then evaluate their projects’ usability, reliability, maintainability, extensibility, and to a lesser extent, efficiency. The divergent thinking makes true cheating practically obvious and allows more collaboration. In the fourth year course, three topics, usability, building and designing web applications, are taught with six or more assignments and with a very active discussion board. Then he shared his ongoing experience of the seminar course, teaching a new kind of distributed class, a unique and innovative international collaborative learning experience. The topic is SE experimentation in which lectures and discussions are asynchronous using Piazza [7] and the class virtually merges students from three universities, one in the US and two in Sweden.
SE classes should assign open-ended problems to: (1) allow diverse solutions; (2) encourage top students to go beyond the minimum; and (3) reward creativity as the software industry needs creative problem solvers. SE classes should encourage students to work, learn and solve together as software engineers in the industry work in teams. SE classes should evaluate student programs by many criteria, such as reliability, usability, maintainability, scalability, availability and efficiency, as all important quality attributes are essential to build high quality software.

3.2 Software Engineering Competency Development Model (An innovative Staged Framework for Software Engineering Education)

Dr. Subraya B M, Vice President and Head, Global Education Center from Infosys Limited, Mysore, India shared his framework for SE competency development model. He explained importance of SEE and training in IT industry. The demand for right professionals with right skill set in ICT is increasing exponentially. Hence, IT companies are facing bigger challenges in recruitment and retaining professionals. Most of the companies recruit fresh graduates from various universities across the country and develop them on the skill set required and have their own training plan to make them industry ready professionals. Skill set related to SE is one of the basic competencies mandatedly required by the whole IT industry. Accordingly, SEE must be an integral part of both fresh entrants training program and competency development program for the experienced ones.

He discussed multifold challenges in SEE. SE field is vast and boundless encompassing tiny embedded applications to enterprise systems, addressing programming practices, design methodologies and architectural strategies across many technologies. The field includes project management along with engineering encompassing both product quality and process quality. SE practices differ across organizations. It is difficult to appreciate certain concepts by the in-experienced community due to lack of experience and maturity in the industry. SE field is relatively young and is in the process of getting standardized. Some topics are based on heuristics, best practices, legacy practices, time tested concepts. The field is changing very fast.

He discussed his four stages proposed model framework to address the SEE challenges. Each stage represents the maturity of the educational system aspiring to evolve in SE knowledge building and skill development. Each stage of the model would involve four essential knowledge dimensions. He proposed the spiral approach for the SEE to evolve through the four stages of the model, addressing all the four dimensions in each evolution of the spiral.

4. GROUP DISCUSSIONS

The major emphasis of the workshop was group-subgroups discussions and brainstorming oriented sessions. Identification of topics for discussions as well as forming of subgroups for each identified topic is a time consuming process and difficult to complete the process democratic way in one day when most participants are meeting each other first time in the workshop. Hence, we planned and tried to execute some activities before the workshop, particularly identification of topics for discussions and forming of subgroups.

4.1 Pre-workshop

For better planning, organization and to facilitate subgroups discussions, it was essential to estimate number of participants, number of subgroups for discussions and size of each subgroup for effective discussions. To estimate these statistics, we employed three systematic techniques: (1) invited position statements; (2) set up a Google group; and (3) arranged an online survey.

4.1.1 Position Statements

Potential participants were requested via call for participation (CIP) to submit one to two page position statements and a 100 to 150 word summary of the participant’s SE teaching experience and/or experience on dealing with recent graduates in the area. The aims for position statements were to: (1) select participants; (2) restrict maximum twenty participants for effective group discussions; (3) ensure a good mix of views, opinions and experiences rather than any quality measure of the statements; and (4) identify common area of interest to determine subgroups and their topics for discussions. The CIP was published through the workshop website, the conference website [8], WikiCFP [9] and we requested PC members of ISEC 2012 and the organizers’ networks by mass email to spread it to their networks.

The following five position statements were received.

3. Focus-shift Need in Software Engineering Education, Richa Sharma, School of IT, Indian Institute of Technology, Delhi.

None of the aims were fulfilled by the position statements as three participants submitted the statements by due date against nineteen participants did online registration on the conference website. Even after two late submissions nothing changed much due to limited number of the statements. However, the organizers were encouraged, from sincere interest of the five authors, to make more effort on spreading awareness about the workshop. Then the CIP sent to the authors of selected papers at the conference by mass email, keynotes and invited speakers of the conference by individual email, faculty and postgraduate students of Indian Institute of Technology Kanpur and academicians within India who are interested in SEE.

The authors of the last three position statements were attended the workshop. Although maximum one minute per position statement was proposed to present summary of the statement the authors used that time during justification for selection of particular topic.

4.1.2 Google Group

We have created a Google group so that participants, as most of them registered online without position statements, can share their thoughts and opinions through online discussions which would lead to identification of topics for subgroups discussions as well as forming of subgroups. Online communication between participants can be possible over email by sending email to seew2012@googlegroups.com and on the web by signing in at http://groups.google.com/group/seew2012. To initiate discussions, an organizer posted a message with subject “Learning team work and a development process while playing” which is shown in Figure 1.
Several times I have applied learning while playing method to undergraduate as well as postgraduate classes as part of Software Engineering course to teach them a product development process and to determine their team harmony. The students had to do a simple exercise from a childhood play of origami, preparation of three types of paper aeroplanes in a team of 4 to 5 students. An aim of this class exercise was to teach them team work and a development process: set a quality and quantity target; do planning and scheduling; carry out work with harmony to achieve the target; check quality of the product; and review the process to find out lessons learned.

In the first 15 minutes each team has to: read the instructions for making paper aeroplanes; prepare a sample plane of each type to estimate time for preparation; check quality of planes by flying them; plan team work for real production; and set target statement, such as the team will produce 20 planes of each type and all of them will be able to fly more than 3 meters high from one end to another end of the class hall. The next 15 minutes for production followed by 15 minutes to check quality of the planes and the last 15 minutes to discuss lessons learned in this exercise.

Although software development is different than this toy example of product development, awareness about development process could be possible by this exercise in a small amount of time. Overall students’ informal feedbacks were they have enjoyed the exercise. Particularly, the teams were able to determine that their teams were productive or not by working in a specific way. The teams tried team work in different ways: some teams divided their teams in three sub-teams and assigned one paper plane to each team; some teams divided plane making work, such as each member pass plane to next member after doing one/two paper fold; some teams did combination of the above two; some teams even combined production and quality check, and so on. However, an empirical evaluation is important to determine tangible and intangible learning from the exercise.

During this time, online registration was increased rapidly. Forty four, including organizers, keynotes and invited, participants were invited to join the Google group for online discussions. Eighteen have accepted the invitation and rest were passive email receivers. The Google group was started with thirty two registered participants on January 18 out of which fourteen have accepted the invitation. After three weeks, five new registered participants were added in the group out of which one has accepted the invitation. Seven more new registered participants were added on February 16 out of which three have accepted the invitation. The post in Figure 1 was unable to motivate participants for online discussions and we were unable to determine potential topics and subgroups.

4.1.3 Online Survey
We designed an online survey using SurveyMonkey [10] with thirteen proposed topics to: (1) determine topics for discussions; (2) prepare participants for subgroup discussions; and (3) save time for forming of subgroups on the workshop day. We requested all the participants to provide a maximum 3 simple preferences, without ranking, from the list and even they can suggest additional topics to be considered for discussions. We gave less than 5.97 days, an average response speed determined for web surveys [11], for response due to approaching the workshop day. The response rate was 27% as twelve responses, including two organizers, received. Table 2 shows the list of topics, in descending order on percentage of responses, proposed by the organizers as potential discussion topics with the percentage of responses favoured the topic. The result of the survey indicated some topics were preferred than others.

<table>
<thead>
<tr>
<th>Sr. #</th>
<th>Proposed topic</th>
<th>Chosen by</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Use of various methods, such as learning while playing and Project-based SE, over Power-Point lecture</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>Requirements and needs of undergraduate SE degree program from the perspective of Indian software industry</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>Curriculum content, coverage and impact of software engineering courses</td>
<td>41.6%</td>
</tr>
<tr>
<td></td>
<td>Challenges to teach mega courses/classes of SE</td>
<td>41.6%</td>
</tr>
<tr>
<td></td>
<td>Evaluation of SE course projects: self-assessment, peer-assessment, measuring individual performance and contributions in project teams</td>
<td>41.6%</td>
</tr>
<tr>
<td></td>
<td>How much do students learn SE while they are in college?</td>
<td>25%</td>
</tr>
<tr>
<td></td>
<td>How to implement anti-plagiarism or prevent plagiarism in SE courses without putting teachers in the role of police officers?</td>
<td>16.6%</td>
</tr>
<tr>
<td></td>
<td>Differences between PG level, UG level and Industry level courses</td>
<td>8.3%</td>
</tr>
<tr>
<td></td>
<td>Assessing how students learn can help to make decisions about how to improve teaching and learning</td>
<td>8.3%</td>
</tr>
<tr>
<td></td>
<td>Addressing today’s classroom challenges</td>
<td>8.3%</td>
</tr>
</tbody>
</table>
The first two topics received 50% of the votes in which 33% of the voters were common in both topics and the last two topics received none. Discussions on the first two topics could be impossible at the same time with involvement of all the interested participants as some participants are interested in both the topics. Analysis of the responses, such as identification of common voters of different topics, was little time taking even with only twelve responses, may be because of option of simple preferences.

We need an analysis method which will help us to divide N potential participants into two sets of subgroups based on participants’ preferences from M topics. Each set contains N participants divided into S subgroups and each subgroup contains minimum three to maximum six participants. We also need a preferences collection technique to gather preferences from the participants which would provide inputs to the analysis method.

4.2 The Workshop Day
Table 2 was guideline before the workshop. 33.33% respondents of the survey were absent and 42.85% participants were failed to respond the survey. On the workshop day the participants were requested to choose their topics for discussions without showing responses of the survey. The first three topics were selected for subgroups discussions by the participants. The on time selection of the topic was the same as the survey result as 57.15% respondents of the survey attended the workshop.

One subgroup discussed the first topic listed in Table 2 in pre-lunch session and the second topic in post-lunch session. Another two subgroups discussed one topic each, the second and third topic respectively, in both, pre-lunch and post-lunch, sessions. Summary of each subgroup discussions are under as supplied by the each subgroup.

4.2.1 Use of Various Methods, such as Learning while Playing and Project-based Software Engineering, over Power-Point Lecture
The focus of discussion was on various non-lecture based methods of teaching SE to graduate and undergraduate students. The subgroup discussed on how we can bring active learning and experiential learning in SE courses. A member shared his experience on how active learning can be practiced while teaching programming lessons to a large class. The respective faculty member employed a successful strategy in which students are asked to make changes to a program presented by the instructor. During the lecture, the students write, compile and execute the program and learn by doing. Another member shared his experience about a course in which students are asked to develop applications for social media platform and deploy it. The objective was not only to teach specific programming skills but to give an environment in which students can deploy their applications and learn from real user responses. This gives student an exposure to the issues involved while dealing with a real client and gives a different perspective. The subgroup agreed that problem-based learning, active learning and experiential learning should be employed as teaching strategies to improve the overall learning experience of students.

4.2.2 Requirements and Needs of Undergraduate Software Engineering Degree Program from the Perspective of Indian Software Industry
The topic was discussed in three sessions by two subgroups. A subgroup has discussed it in post-lunch sessions and another subgroup has discussed it in pre and post lunch sessions.

The subgroup discussed the merits and needs of designing a SE program that offers an undergraduate and graduate degree in SE (for example, a BTech degree in SE). In India currently there are no, to the best of the participants’ knowledge, noticeable programs where students can earn a bachelor degree in SE. The group debated that SE is a core course in CS bachelor degree program but the discipline of SE is so vast and in demand so that an undergraduate degree program can be possible. During the discussion, a consensus emerged between the participants that SE is an engineering discipline; hence, a focused program is desirable. The subgroup discussion was led by Dr. Kesav Vithal Nori, Distinguished Professor at International Institute of Information Technology Hyderabad, who provided his perspective on integrating SE into the mature traditions of engineering discipline.

In another subgroup, the discussion was varied between many topics-subtopics, such as requirements and needs of undergraduate SE program, how to determine need of the industry, need to involve industry representatives, interaction between industry and academia and potential topics, subjects, curriculum of the program, before lunch. The subgroup has tried to discuss many subtopics with many views but unable to contribute in any one area. Hence, it was decided to discuss points which could be useful to perform SWOT (strengths, weaknesses/limitations, opportunities and threats) analysis of the current state of affairs in SE education in India and identify the need of a SE program at undergraduate level.

In post-lunch session, following points were discussed which could lead to a systematic SWOT analysis.

- The Indian software and IT industry is growing fast and is recruiting about 0.1 million engineering graduates each year. In the absence of qualified software engineer, the industry is recruiting graduates of related disciplines who have little or no formal education in SE. An established industry is available for employment and entrepreneurship for qualified software engineer.
- Various universities in the US, Australia and UK offer undergraduate degree program in SE considering the demand of software engineers. According to Department of Labor, US, fresh graduates in SE are earning high pay compared to fresh graduates of other engineering discipline. Similarly, fresh qualified software engineers in India are more likely to earn high salary than their peers.
- Infrastructure requirements are less for establishing undergraduate degree programs in SE than traditional engineering discipline.
- Employability of a computer science and engineering (CSE) graduate as a “software engineer” is a major concern, as often CSE graduate do not have satisfactory knowledge of even a single SE course.
- There is a prevalent huge gap between conceptual knowledge and its application on various problems using tools and technologies.
- Indian culture prevents the ability to question what is being taught/informed due to subservient tendencies.
- There is a potential for product development in addition to India’s current focus service industry. The curriculum can
focus on product and process engineering as product engineering is also important for software engineers.

- Problem solving should be focus rather than programming.
- A software systems engineering perspective can also be considered.
- India has many government universities, universities established under public private partnership (PPP) as well as private universities. There is a potential for private and PPP universities to start undergraduate degree program in SE.

4.2.3 Curriculum Content, Coverage, and Impact of Software Engineering Courses

The subgroup discussed the topic in both sessions and discussed following points in post-lunch session.

- Industry must help academia.
- We need context-oriented case studies.
- Must make sense to local students.
- We need new and different foundation courses.
- Curriculum is too packed—leaving no room for innovation or new topics.
- Courses need to connect theory and case studies.
- Benchmark against certification courses from IEEE and ACM.

5. CONCLUSIONS

Organising such workshop, in which selection through position statements and main emphasis on group-subgroups discussions, was anticipated a challenging work as: (1) most of the workshops in the area of software engineering and computer science in India are mini-conferences with paper presentations; and (2) it is believed that in general most members of Indian academic community ignores open discussions on academic practices. Although call for submission attracted less position statements the workshop was a successful attempt as twenty participants attended the workshop from which fourteen engaged in full day active discussions.

In addition to group discussions and keynotes, we planned various icebreaking activities to ease the initial discomfort of first time meeting between participants and to energise group for effective subgroups discussions. However, we used only introduction and justification for choosing topics as icebreaking when most attendees were focused participants. A detailed study is required to utilise potential of position statements and online discussions as other workshops and many groups have already enjoyed them effectively. The online survey was helpful in group formation and context setting on the day of the workshop. We can use a better survey technique with an appropriate analysis method to analysis the survey and improve. Practitioners’ perspectives are more important as SEE need to match with the real world requirements which was missed in this workshop. Extra effort may be required to involve practitioners in the next workshop.

The response in terms of the contributions by attendees is a clear indicator and confirmation of the need of having a focused discussion forum for brainstorming on the topics of SEE in India. Most participants felt that one day is limited for team building as well as for profound discussion on a topic to achieve the aim to create a set of recommendations and/or draft papers. Although nearly 50% participants were for the workshop remaining participants participated in the workshop as they were also interested to attend the conference. It is more likely to get fewer participants if the workshop is organized independently without a suitable collocated conference and with a conference only one day workshop is possible. An independent workshop may attract focused participants but may miss participants with different backgrounds which may possible with a collocated conference. A focused group of participants with different backgrounds to bring diverse perspectives with a good team bonding would be possible after a few editions of the workshop; hence, we will retain the same format of the workshop in our next edition.

6. ACKNOWLEDGMENTS

The workshop organizers would like to acknowledge the contributions from all the participants, ISEC 2012 conference organizers, student volunteers and several of our colleagues who provided their valuable inputs and time in making this workshop happen. The organisers express their deep gratitude to Prof. Jeff Offutt and to Dr. Subraya B M for their keynotes. We are also thankful to Dr. Srinivas Padmanabhan, Infosys Limited for his continuous support.

7. REFERENCES


