

# Induction of Interactive Methods to Teach Software Engineering Course

**Eman S. Alahmadi and M. Rizwan Jameel Qureshi**

Faculty of Computing and Information Technology, King Abdulaziz University, Saudi Arabia  
Email: eman\_alahmadi@hotmail.com, anriz@hotmail.com

**Abstract**—Software engineering (SE) field is considered as a backbone of software industry. Students suffer many problems and difficulties when they approach to industry due to the old contents of SE course, lack of practice and inability in the subject area. The main reason of the SE course problems is the lack of coordination or gap between software industry and academia. This paper proposed a novel solution to enhance SE curriculum to bridge the gap between industry and academia. The proposed solution will assist students at optimal level in their professional careers to achieve goals of SE course. Survey research methodology is used to evaluate the proposed solution and the results of data analysis are highly encouraging. It is anticipated that the universities will find this research helpful to improve the quality of SE course.

**Index Terms**—Software engineering, education, industry, curriculum, mental-map

## I. INTRODUCTION

Software engineering (SE) course is aimed to design and develop high quality software [1]. Software development needs enormous time, cost, efforts and resources. Software industry faces a problem to select suitable candidates in the Saudi Arabia that are equipped with the required software development skills to be hired. The reason is the gap between SE course curriculum and software industry in the Saudi Arabia.

Computer Science devises theories and methodologies to establish the basis of software systems. SE focuses to address main problems of software engineers in development environment. Many of the computer science theories stand powerless to solve the problems of complex software engineering. Computer aided software engineering (CASE) tools are used to implement system development life cycle phases to analyze, design, develop and implement high quality software.

It is essential for the universities and researchers to develop and suggest renovated and useful teaching approaches to equip students' with required skills. The approaches are toward build skills of students to become expert in the world of software engineering. The approaches have to simulate about the activities that are actually practiced in software companies. The aims of new learning approaches are facilitating to get real time experiences for students. This expertise from university

can make a student a better professional for his future career. The current way of teaching and learning software engineering in many universities showed a major gap between software industry and academia.

Rest of the paper is organized as follows. Section 2 covers related work. Section 3 describes the research problem. Section 4 illustrates the proposed solution. Validation of the proposed solution is discussed in section 5.

## II. RELATED WORK

The main challenge for universities during the last several years is how to teach software engineering course to equip students well to face situations in their professional careers. Nowadays, software engineering fresh graduates are facing a lot of difficulties when they approach to the software industry. The graduated students of software engineering courses are suffering from inadequate experiences in real time scenarios. This problem arises during last several years when software development has become an industry like manufacturing industry. The instructors and researchers proposed many approaches to enhance the student's outcomes of the software engineering course. These approaches renovate teaching methods to develop student's expertise to become a real software development professional.

A framework is presented based on experiences and lessons learned from the pilot projects [1]. It includes ideas to build an effective collaborative infrastructure, design a feasible collaboration and evaluate procedures. The objective is to continuously improve the efficiency and effectiveness of teaching software engineering course [1].

A novel methodology to teach and learn software engineering course is presented [2]. The methodology is based on integrating various activities into SE course such as gaming and role plays, demo of SE tools and workshops. The methodology helps the students to improve their knowledge and quality to complete capstone projects.

According to the curriculum design in Nanjing University [3], students learn about software processes by studying three courses related to computing and software engineering. These courses enable the students that how to apply software processes at individual and team level. It is very difficult for students to understand concepts and realize the value of software processes without studying

software projects management course.

An approach is proposed to assess group work performance of students to improve teamwork skills [4]. More specifically, the proposed approach is focused on effective assessment and it is appropriate for software

engineering curriculum. It can be tailored to specific courses or requirements in group work skills development. The proposed approach is applied on several courses and the results are found positive and promising.

Table 1. Summary of the Related Work

Title	Summary
A framework for global collaboration in teaching software engineering [1].	There is a need to extend the opportunity to more students in several countries to standardize the collaboration infrastructure.
Activity based teaching learning in software engineering-An experience [2].	The assessment of students learning focused on their understanding and application more than it is on memory.
An introductory software engineering course for software engineering program [3].	Attendees of this course come from grade one and two. Students without experience in software projects management will have difficulties to understand concepts and values of software process. Usage of waterfall model which has many disadvantages.
Assessing Teamwork Performance in Software Engineering Education: A Case in a Software Engineering Undergraduate Course [4].	Inability to assess the development and improvement of teamwork skills.
Educating students by "real-world" software engineering practice—Designing software engineering practice course [5].	There is a need to add more interesting functions for students to satisfy their true needs.
Educational software engineering: Where software engineering, education, and gaming meet [6].	Education based on coding requires extensive skills and logic.
Enhancing the software engineering curriculums: A case study of the Jordanian Universities [7].	No covering for reducing the gap between SE and CS graduates and the software industry.
Software Engineering Education in Jordan [8] .	Knowledge areas proposed by the 2013 SWEBOK were measured.
Teaching Software Engineering in Developing Countries: A Position Paper [9].	The perspective of this paper is based on the experience in teaching software engineering course in a developing country.
The challenge and practice of creating software engineering curriculum [10].	Students still cannot make "real-world" software engineering practice.
The necessary software measurement knowledge in software engineering education from the practitioners' point of view [11].	Participants must be selected according to certain criteria to assure their level of expertise.
The Research of Software Engineering Curriculum Reform [12].	The methods of teaching reform should be implemented on wide experiment.

The proposed approach [5] focuses on both design and delivery of software engineering curriculum to educate students using industrial software engineering exercises, expertise and learnt lessons. A web-based educational gaming environment is developed to teach and learn software engineering courses [6]. The web-based educational gaming environment is called Pex4Fun. An analysis on the software engineering (SE) curricula of 13 Jordanian universities is made to compare with the software engineering knowledge areas [7]. It was found that many knowledge areas are not covered by the SE curricula.

A case study is conducted to examine the SE study plan and curriculum of bachelor's program at Isra'a University (IU) in Jordan [8]. The objective of this study is to measure the conformance with the Higher Education Accreditation Commission HEAC guidelines.

The rationale is to deliver a realistic learning experience connecting international methods and processes with local context and software engineering development challenges and realities. Osman [9] specifies the necessity to incorporate contextualized and localized examples to teach software engineering course in developing countries. It is difficult to teach software engineering curriculum in one course [10]. Introductory SE courses cover theory instead of practice. Graduating students are weak to apply suitable software engineering techniques to solve industrial problems due to lack of practice and training [12].

Villavicencio and Abran [11] discuss the significance of using empirical SE techniques to teach software engineering courses to improve innovative skills of students. A new approach is introduced to use virtual environments to teach SE courses [12].

A number of approaches [13-17] have been introduced to deal with this problem like divide students into teams to work on a semester project. The team approach will give students a feeling to understand that what are going on in software companies to develop software projects. Blake & M.B. [13] suggests providing two courses in software engineering. The two courses of software engineering will be offered to cover the basics of software engineering concepts in the first course and second one is offered to develop a software project for the whole semester. In [16-17], both papers provide approach of simulating real company environment where large teams of students need to work together on the software engineering projects. Each company has a project manager, team leads and team members. Furthermore, an idea of competition also advised between different companies to encourage the student to produce the best result.

Many proposals are introduced to improve teaching methods according to [18-19]. These proposals [18-19] are based on idea of working a group of 6-8 students in a major software project. The aim of this is to prepare them for large team in industry. In [18], the approach depends on incremental development and delivery after every two

weeks. Also, an instructor falls into multiple roles even also simulated customers like Amazon Company. In addition, student's knowledge built from previous and present one that will be gained in doing their projects [19]. Final projects will be analyzed using key performance areas (KPA's).

With reference to [20], the developed approach focusing on many skills such as managerial, engineering, team working and personal skills. Software engineering student must have adequate different skills rather than engineering skills. Students have to choose at least two skills from the managerial, three skills from engineering and Students must be achieved all personal skills.

Rusu and Swenson [21] present a new idea to a graduate student by appointing two instructor supervisors for the software engineering project. One of the instructors will be a full time faculty time and other is part time industry time. This will give the student a combination of skills from both sectors.

Hadjerrouit [22] introduces a new approach is learner-centered Web-based instruction. This approach is based on three ideas. First, the software engineering education must become more realistic, which mean provide more real examples. Second, software engineering education moves closer to the learner. Third, learning process has to be more to Web technology.

According to [23], it produces an approach to the software engineering course based on the Problem-Based Learning Principles. PBL centers on the problem based to make students focus in solving it. PBL has three principles. First, every week, students solve problems. Second, students work together to solve problems. Third, the role of instructor should be helping the students rather than instruct them. Table 1 shows summary of the related work.

### III. PROBLEM DEFINATION

There is a gap between software industry and academia due to current methods of teaching software engineering in many universities of world. Following is the problem identified from the literature review to address in this research [1][4][8].

“How to improve and enhance curriculum of software engineering courses to bridge the gap between software industry and academia?”

### IV. THE PROPOSED SOLUTION

Instructors should induct interactive methods to teach SE courses in addition to the regular classroom teaching. Most of the course material is theoretical and there is no incorporation of real projects or case studies from global software market. A numbers of instructors teaching SE course does not have software industry exposure. These faculty members teach weak knowledge to students. The weak knowledge prepares students inappropriate for the

software industry. There is no clear understanding to students about the course of SE, course learning outcomes and objectives due to no corporate training to complete degree projects. The involvement of software industry, while performing corporate training during the degree program, will improve the skills of SE students to bridge the gap between industry and academia. The proposed solution is based on the following four goals.

#### A. Goal 1- Identify The Difficulties And Measure The Required Skills Of The Se Course

We can compare the skills require in SE industry with current curriculum of Universities. This will help the instructors to teach SE courses more practically than theoretically. This comparison will also assist students to feel more comfortable, confident and increase their understanding about the subject matter. This goal will also help the educationists to add skills to curriculum to cater difficulties as major concerns in an efficient manner. Corporate training is proposed to improve the skills of students that are required in the software industry. Universities will select the software companies and students to be trained. Corporate training will be treated as equivalent to one regular semester in terms of credit hours. There will be evaluation after the training to grade the students. It is proposed to have two mentors for the students of corporate training. One mentor will be the employee of software company and second mentor will be instructor of university. Goal 1 can be used as a standard to make sure that students enrolling into SE courses have necessary background knowledge and enrich the contents with the industry compatible.

#### B. Goal 2- Teach The Software Engineering Course Using Mental-Map Technique

Mental map technique is a very power skill to visualize something that one can see only in his/her head. It is used to express ideas and schemes using branches, images and colors rather than just words. Mental map relies on visual memory illustration to ease review and remember the rules and instructions. This method is used by human brain to think that is composed of neurons. This is exactly the way of the human mind in thinking and style of building mental maps. Mental map technique is also called cognitive mapping. Mental map technique can help SE students as follows.

- More creative.
- Save time.
- Solve problems.
- Increase the speed and efficiency of the study.
- Ease the understanding.
- Increase focusing.
- Organize ideas.
- Pass the exams successfully.
- Remember things better.
- Planning in efficient manner.
- Improve communication with others.

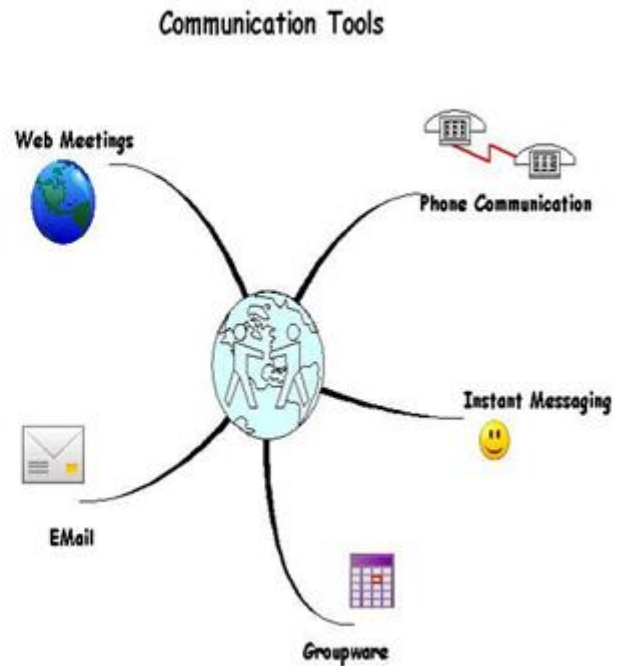


Fig. 2. Virtual Methods to Teach SE Course

Mental map helps the students to understand and focus larger and deeper and it results in saving time and improving test scores.

*C. Goal3- Induct The Inquiry Strategy To Improve Quality And Excellence*

Instructors can use the inquiry strategy (Inquiry Method) that is based on the discovery and curiosity approaches. Students use their abilities to discover scientific knowledge to grasp concepts and generalizations. It is proposed for the instructors of SE course to induct self-review mechanism assessing students' knowledge and understanding with respect to quality and excellence. SE Students review their understanding following the basis of quality to achieve excellence and it is one of main advantages of this method. Fig. 1 shows the mechanism of self-reviewing.

*D. Goal 4- Effect Of Using Virtual Classrooms To Teach Se Course*

Effective communication between students and teachers has a key role to achieve success in teaching discipline. This research proposes to teach SE courses through virtual class rooms to achieve the benefits of globalization. Communication problems can be solved by integrating managerial and technical methods. Fig. 2 shows the usage of virtual class rooms. It is easier and comfortable for both professors and students. All the required information about SE course will be available over the Internet to facilitate communication among faculty and students. It is a quick method to update, upload and download course related materials. Virtual education is cost saving for the students. Flexible timing is another major advantage of the virtual classroom.

Stimulated and quick learners are permitted to move ahead at a much faster rate than the traditional classroom learning. Overall communication among students and between instructor and students is significantly improved via emails and instant messages.



Fig. 1. Mechanism of self-reviewing

V. VALIDATION

Questionnaire technique is used as a research method to validate the proposed solution. It is used to examine the responses of a relatively large number of participants. Questionnaire is composed of fifteen questions to cover the four goals of proposed solution. Goal 1 is identifying the difficulties and measuring required skills of the SE course. Goal 2 is to teach the software engineering course using mental-map technique. Goal 3 is to induct the inquiry strategy to improve quality and excellence. Goal 4 is set to measure the effects of using virtual classrooms to teach SE course. The questions should be answered in likert scale of five as shown in Table 2.

Table 2. Likert scale used to evaluate questionnaire

5	Strongly Agreed
4	Agreed
3	Neither Agreed Nor Disagreed
2	Disagreed
1	Strongly Disagreed

**A. Goal 1- Identify The Difficulties And Required Skills Of The Se**

The questions in this goal represent the difficulties and measure the required skills to attend the SE course. As shown in Table 3, 31.3% are agreed to establish FCIT center whereas 12.5% are strongly agreed. Furthermore, 29.4% are neither agreed nor disagreed. However, 14.7% are disagreed while other 12.2% are strongly disagreed. Fig. 3 depicts the same result graphically.

Table 3. Cumulative analysis of Goal 1

Q. No.	Strongly Disagreed	Disagreed	Neutral	Agreed	Strongly Agreed
1	9	10	28	25	8
2	8	19	30	18	5
3	9	13	19	30	9
4	13	5	17	27	18
Total	39	47	94	100	40
Avg.	12.2	14.7	29.4	31.3	12.5

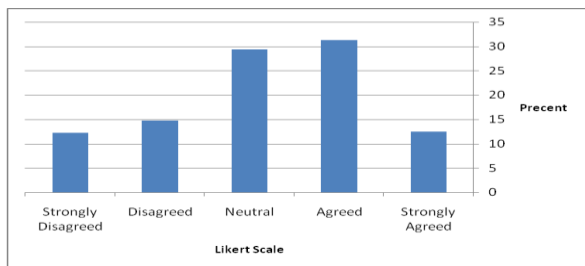


Fig. 3. Cumulative Analysis of Goal 1

Table 4. Cumulative analysis of Goal 2

Q. No.	Strongly Disagreed	Disagreed	Neutral	Agreed	Strongly Agreed
1	12	5	15	24	24
2	11	5	28	25	11
3	13	8	30	25	4
Total	36	18	73	74	39
Avg.	15	7.5	30.4	30.8	16.3

**B. Goal 2- Teach The Se Course Using Mental-Map Technique**

The questions in this goal measure how much mental-map can play significant role on teaching and understanding. Table 4 shows that 30.8% are agreed and 16.3% strongly agreed to use mental-map. In another hand, 30.4% are responding neutral. However, 7.5% of responders are disagreed and 15% are strongly disagreed. Fig. 4 displays the results of Table 4 graphically.

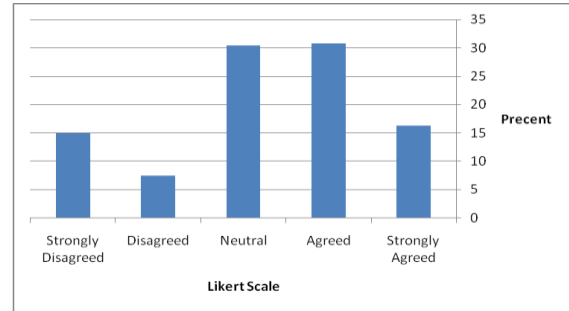


Fig. 4. Cumulative Analysis of Goal 2

**C. Goal3- Induct The Inquiry Strategy To Improve Quality And Excellence**

The questions set in this goal to evaluate the induction of inquiry strategy using self-review mechanism to improve quality and excellence. Table 5 shows that 32.2% of the respondents are agreed with this way and 20.9% are strongly agreed. Whereas 9.1% of the responders are disagreed and 13.1% are strongly disagreed. Also, 24.7% are neither agreed nor disagreed. Table 5 results are presented in fig. 5.

Table 5. Cumulative analysis of goal 3

Q. No.	Strongly Disagreed	Disagreed	Neutral	Agreed	Strongly Agreed
1	14	7	26	18	15
2	7	7	13	35	18
3	12	6	20	25	17
4	9	9	20	25	17
Total	42	29	79	103	67
Avg.	13.1	9.1	24.7	32.2	20.9

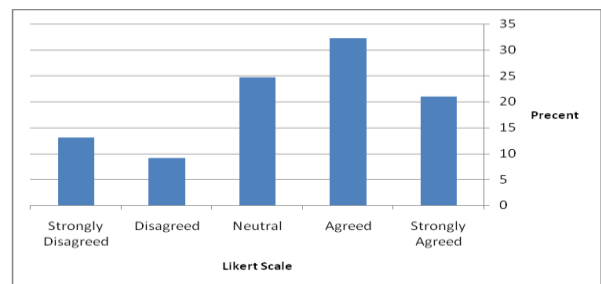


Fig. 5. Cumulative Analysis of Goal 3

Table 6. Cumulative analysis of Goal 4

Q. No.	Strongly Disagreed	Disagreed	Neutral	Agreed	Strongly Agreed
1	8	8	21	25	18
2	7	8	24	27	14
3	9	13	26	20	12
4	7	13	22	26	12
Total	31	42	93	98	56
Avg.	9.7	13.1	29.1	30.6	17.5

**D. Goal 4- Effect Of Using Virtual Classrooms To Teach Se Course**

Four questions are asked to evaluate the effect of teaching SE course using virtual class rooms. The results are displayed in the Table 6. Table 6 shows that 30.6% of the respondents are agreed and 17.5% the participants are strongly agreed about the profound effects of using virtual classrooms to teach SE course. 29.1% of the respondents are neither agreed nor disagreed to use virtual class rooms. However, 13.1% of the participants are disagreed and 9.7% of the respondents are strongly disagreed. Fig. 6 depicts the cumulative results of goal 4.

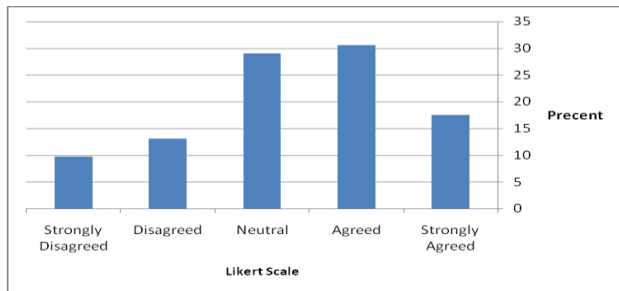


Fig. 6. Cumulative Analysis of Goal 4

## VI. CONCLUSION

One of the difficult tasks is to teach SE course in such a way that prepares the students to be equipped with necessary skills require in the software industry. SE track courses are one of the pillars of the computer science and information technology degree programs offered in all the top universities of world. This paper is written in support of inducting interactive methods to teach software engineering courses to equip students with the skills that are required in the software industry. The current methods of teaching SE courses are more theoretical than the practical due to a gap between industry and academia. Majority of students face difficulties in the software industry lacking in the required skills to be hired in good companies. A solution is proposed in this paper to address this problem that is divided into four goals. Goal 1 is used to identify the difficulties and required skills to attend the SE course. This goal will also help the educationists to update the contents of SE courses to be industry compatible and it is also proposed to introduce corporate training in terms of credit hours as equivalent to regular semester courses. The proposed corporate training will enhance the expertise of students and prepare them more familiar and comfortable to achieve desired outcomes of SE industry. Goal 2 is proposed to teach SE courses using metal map technique. It is proposed to induct the inquiry strategy to improve quality and excellence in goal 3. It is proposed to teach SE courses using virtual classrooms in goal 4. A survey is conducted to evaluate the proposed goals and the results are found encouraging in enhancing and improving SE curriculum to bridge the gap between software industry and academia.

## ACKNOWLEDGEMENTS

First of all, I am thankful to Allah to complete this research. I am also thankful to my parents to support me again and again. Special regards and thanks to my supervisor Dr. M. Rizwan Jameel Qureshi. He guided and helped me to learn step by step throughout this research to complete this paper.

## REFERENCES

- [1] J. Ding, "A framework for global collaboration in teaching software engineering," in *Collaborative Teaching of Globally Distributed Software Development (CTGDSD), 2013 3rd International Workshop on*, 2013, pp. 30-34.
- [2] P. Desai and G. Joshi, "Activity based teaching learning in software engineering-An experience," in *Engineering Education: Innovative Practices and Future Trends (AICERA), 2012 IEEE International Conference on*, 2012, pp. 1-6.
- [3] D. Shao, B. Luo, E. Ding, and Q. Liu, "An introductory software engineering course for software engineering program," in *Software Engineering Education and Training (CSEE&T), 2011 24th IEEE-CS Conference on*, 2011, pp. 411-415.
- [4] J. Chen, G. Qiu, L. Yuan, L. Zhang, and G. Lu, "Assessing Teamwork Performance in Software Engineering Education: A Case in a Software Engineering Undergraduate Course," in *Software Engineering Conference (APSEC), 2011 18th Asia Pacific*, 2011, pp. 17-24.
- [5] Y. Sun and X. Liu, "Educating students by "real-world" software engineering practice—Designing software engineering practice course," in *Computer Science & Education (ICCSE), 2012 7th International Conference on*, 2012, pp. 1356-1359.
- [6] T. Xie, N. Tillmann, and J. De Halleux, "Educational software engineering: Where software engineering, education, and gaming meet," in *Games and Software Engineering (GAS), 2013 3rd International Workshop on*, 2013, pp. 36-39.
- [7] S. Hanna, H. Jaber, F. A. Jaber, T. A. Shalaby, and A. Almasalmeh, "Enhancing the software engineering curriculums: A case study of the Jordanian Universities," in *Software Engineering Education and Training (CSEE&T), 2014 IEEE 27th Conference on*, 2014, pp. 84-93.
- [8] F. Al-Zaghoul, A. Hudaib, and M. Ahed, "Software Engineering Education in Jordan," in *Computer Science and Information Technology (CSIT), 2014 6th International Conference on*, 2014, pp. 127-132.
- [9] R. Osman, "Teaching Software Engineering in Developing Countries: A Position Paper," in *Computer Software and Applications Conference (COMPSAC), 2012 IEEE 36th Annual*, 2012, pp. 648-653.
- [10] Y. Sun, "The challenge and practice of creating Software Engineering curriculum," in *Software Engineering Education and Training (CSEE&T), 2011 24th IEEE-CS Conference on*, 2011, pp. 497-501.
- [11] M. Villavicencio and A. Abran, "The necessary software measurement knowledge in software engineering education from the practitioners' point of view," in *Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference on*, 2012, pp. 1-5.
- [12] A. Abulrub, A. N. Attridge, and M. A. Williams, "Virtual reality in engineering education: The future of creative

- learning," in *Global Engineering Education Conference (EDUCON), 2011 IEEE*, 2011, pp. 751-757.
- [13] Blake, M. B., "A student-enacted simulation approach to software engineering education," *Education, IEEE Transactions on*, vol.46, no.1, pp. 124-132, 2003.
- [14] Broman D., Sandahl K., and Abu Baker M., "The Company Approach to Software Engineering Project Courses," *Education, IEEE Transactions on*, vol.55, no.4, pp.445-452, 2012.
- [15] Chaczko Z., Davis D., and Mahadevan V., "New perspectives on teaching and learning software systems development in large groups," *Information Technology Based Higher Education and Training, 2004. ITHET 2004. Proceedings of the Fifth International Conference on*, pp. 409- 414, 2004.
- [16] Coppit D., and Haddox-Schatz J. M., "Large team projects in software engineering courses," in *Proc. 36th SIGSE Tech. Symp. Comput. Sci. Educ.*, New York, NY, pp. 137-141, 2005.
- [17] Hadjerrout S., "Learner-centered web-based instruction in software engineering," *Education, IEEE Transactions on*, vol.48, no.1, pp. 99- 104, 2005.
- [18] Karunasekera S., and Bedse K., "Preparing Software Engineering Graduates for an Industry Career," *Software Engineering Education & Training, 2007. CSEET '07. 20th Conference on*, pp.97-106, 2007.
- [19] Meawad F., "The virtual agile enterprise: Making the most of a software engineering course," *Software Engineering Education and Training (CSEE&T), 2011 24th IEEE-CS Conference on*, pp.324-332, 2011.
- [20] Odeh, M. "A Reflective Approach to Improve Learning and Teaching of Software Engineering in Large Groups," *The International Arab Journal of Information Technology*, vol. 1, no. 0, 2003.
- [21] Razmov V., "Effective pedagogical principles and practices in teaching software engineering through projects," *Frontiers In Education Conference - Global Engineering: Knowledge Without Borders, Opportunities Without Passports, 2007. FIE '07. 37th Annual*, pp.S4E-21-S4E-26, 2007.
- [22] Rusu A., and Swenson M., "An industry-academia team-teaching case study for software engineering capstone courses," *Frontiers in Education Conference, 2008. FIE 2008. 38th Annual*, pp.F4C-18-F4C-23, 2008.
- [23] Tian K., Cooper K., and Zhang, K., "Improving Software Engineering Education through Enhanced Practical Experiences," In *Computer and Information Science (ICIS), 2011 IEEE/ACIS 10th International Conference on*, pp.292-297, 2011.

### Authors' Profiles

**Eman Saleh Alahmadi** received her undergraduate degree from Taibah University. She is currently working as a teaching assistant in the Department of Management Information System, Taibah University. She is a post graduate student in the Department of Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia.



**DM. Rizwan Jameel Qureshi** received his Ph.D. degree from National College of Business Administration & Economics, Pakistan 2009. This author is the best researcher awardees from the Department of Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia in 2013 and the Department of Computer Science, COMSATS Institute of Information Technology, Lahore, Pakistan in 2008.

**How to cite this paper:** Eman S. Alahmadi, M. Rizwan Jameel Qureshi, "Induction of Interactive Methods to Teach Software Engineering Course", *IJMECS*, vol.7, no.6, pp.43-49, 2015.DOI: 10.5815/ijmecs.2015.06.07