120th ASEE Annual Conference & Exposition
FRANKLY, WE DO GIVE A D*MN
June 23-26, 2013

Paper ID #6912

Industry Involvement in an Undergraduate Software Engineering Project Course: Everybody Wins

Prof. W Eric Wong, University of Texas, Dallas

W. Eric Wong received his Ph.D. in computer science from Purdue University. He is currently a Professor and Director of International Outreach in the Department of Computer Science at the University of Texas, Dallas. Prior to joining UTD, he was with Telcordia (formerly Bellcore) as a Project Manager for Dependable Telecom Software Development. Professor Wong received the Quality Assurance Special Achievement Award from Johnson Space Center, NASA, in 1997. His research focus is on the technology to help practitioners develop high quality software at low cost. In particular, he is doing research in software testing, debugging, security/safety, and reliability at the application and architectural design levels. Professor Wong is the Vice President for Technical Operations of the IEEE Reliability Society and the Secretary of the ACM Special Interest Group on Applied Computing (SIGAPP). He also has an appointment as a Guest Researcher from NIST (National Institute of Standards and Technology), an agency of the U.S. Department of Commerce.

Industry Involvement in an Undergraduate Software Engineering Project Course: Everybody Wins

The "Software Engineering Project" (SE 4485) is a one-semester capstone course in our undergraduate Software Engineering curriculum at the University of Texas at Dallas (hereafter, simply referred to as UTD). It is intended to complement the ***theoretical knowledge*** that students receive in their prior (and ongoing) courses and provide them with an ***in-depth, hands-on experience*** in all aspects of software engineering. They are expected to walk-through the various phases of a software development life-cycle such as requirements analysis, software architecture and design, testing, and implementation, while producing quality deliverables at each phase. By the end of the semester students should have a working knowledge of each individual aspect of software engineering, and also have gained experience in how these aspects are related to, and depend on, one another in the successful development of a software system.

However, the task of designing a project for the students is a major challenge for the instructor. Several factors must be taken into account, such as:

- The project must have some relevance to the course content.
- The students should be expected to spend a reasonable amount of time on the project.
- The project can be completed by the students within a feasible timeframe.
- The successful completion of the project should be within the abilities of the students.
- The project should be designed in such a way that no student in the class is treated unfairly.

A project that satisfies all of the above conditions may still raise complaints that it is too simplistic or not sufficiently realistic. Such criticism is not unwarranted. Often when designing a project, to satisfy all of the constraints the instructor must sacrifice many desired requirements. In doing so, the classroom project often fails to simulate real-life projects. The teaching efficacy of the project is thus greatly reduced.

Difficult as it may be, it is still very important to close the gap between software engineering practices in industry and those taught in an academic setting. It is not enough to select a project that emulates one representative of industry. Rather, what if the project is actually proposed by an industry affiliate, based on their own needs? What if instead of receiving only a project description, students can interact with company representatives acting as customers? Also, what if the industry affiliate is not just restricted to helping design the project; but instead they also play an active role in supervising the development of the software?

With the help of our industry partners, we are able to investigate these questions. This paper serves to document and share the experiences of bringing in industry partners, to not just help propose projects, but also evaluate the deliverables turned in by students, for a software engineering project course. Such a partnership has been very well received by both the students and the industry affiliates.

In addition, a success in the software engineering project course is also critical for ABET accreditation of the Computer Science department and the Software Engineering Program at UTD. Through this course, students have learned, practiced, and improved their skills in communication, teamwork, ethics, and professionalism, as well as received knowledge of contemporary issues and engineering within a real-life industry environment.

Related Work

Broadly speaking, any article or paper that aims for the betterment of computer science or software engineering education is related to the work presented in this paper. However, given that we discuss the collaboration of industry and academia in the context of a software engineering project course, we narrow our focus to other such work.

In his paper titled "*An Industry Approach to the Software Engineering Course*"[2], Clifton emphasized that there are several methods of teaching software engineering to give the course a flavor of the industry. He noted that the most obvious of these is a class project, especially one from the industry itself.

Gehrke, et al.[3] described their four years of experience conducting software development project courses. To achieve practical results, they emphasized that the chosen application and project execution must be as close as possible to real-world operation.

In a recent study[11], Tahmoush, et al. identified that the practical constraints of a semester-long course limited how well it could match realistic software engineering practices and reduced the real-world applicability of the classroom experience. To address the problem, they proposed a shift away from the mentality that student success in the course should be measured in terms of grades. Instead, an industrial perspective should be substituted which emphasizes good software engineering processes and practices.

Klappholz, et al.[8] demonstrated the importance of real projects for real clients courses (RPRCCs) in computing curricula. They provided advice on setting up an effective support infrastructure for such courses, discussed where and how to find clients, and addressed issues related to assessment and evaluation. In addition, benefits of an RPRCC-centric approach to computing curricula were presented.

Christensen, Rundus, and Prodanoff[1] argued for the importance of students learning both hard and soft skills in order to become productive software engineers – abilities such as working well with others and managing uncertainty should particularly be cultivated. Outside the classroom, projects will be encountered with missing or poorly defined requirements and dependencies, and students must be equipped to deal with these uncertainties

Gorka, Miller, and Howe[4] presented the lessons learned from an industry sponsored capstone project such as maintaining industry confidentiality, encouraging active participation from sponsors, and using company's back-stories to motivate students. They also suggested that students might think of things overlooked by faculty mentors and industry sponsors. In addition, they emphasized that faculty and students must be prepared to deal with unexpected incidents and make appropriate adjustments to ensure the success of the projects.

Deviating from the common practice of reporting only success stories, Kantipudi, et al.[7] provided an analysis of factors leading to the failure of software engineering course projects, as well as practical recommendations for increasing the chances of successful projects. They broadly classified the reasons for failures into three categories: *technical* issues, *personal* issues and *management* issues.

In order to stimulate students' interest and maintain their enthusiasm for their assigned projects, Horning and Wortman[6] proposed to use a multiplayer economic game, where all the teams interact with each other and play different roles, such as developer, buyer and seller of software products developed by themselves. Team members will address issues such as delegation of responsibility, communication of decisions, coordination of activities, price negotiation, and resolution of differences in management during the "game".

Wortman described his experiences while teaching a software engineering project with an objective to provide students with some 'real world' experience[13]. The paper discusses a variety of problems that arose in their use of a software project and also presents some guidelines on how such a project may be organized. However, the work in[13] deals exclusively with projects in a graduate level course. A project for a graduate level course may be quite different from its undergraduate counterpart and it is not simply a matter of scaling a graduate level project down to match an undergraduate course.

McKeeman presented an approach to meeting the academic objectives of advanced software engineering project courses (part of the Master of software engineering degree program), including a simulated industrial context[9]. The author (also instructor of the course) emphasizes industrial simulation and to this effect, various supporting roles are played for the project team. Some of the suggested external participants are the *Customer*, the *Technical Editor*, etc. Often the instructor plays such a role by wearing a particular hat (dark blue, checkered, etc). The purpose of the hats is to ensure the team members react to the role rather than to the real person. In our case, instead of simulating an industry setting, we actually have the students work on, and develop, a real project that is to be used in industry.

One of the challenges to teach a capstone project course is to provide students with sufficient motivation and get them invested in the outcome. To achieve this, Horgan, Smith and Thomas in their 2005 ACE paper[5] suggested a problem domain that accurately reflects the concerns and priorities of a real industry client. They also proposed a *Real World Software Process* with four different phases: (1) Phase Zero – developing a project proposal which addresses the client's needs and clearly identifies goals and success criteria, (2) Phase One – requirements gathering, release planning, and the initial development, (3) Phase N – the generic, repeatable cycle in which the functionality of the system is incrementally extended and the requirements reviewed periodically, and (4) Finalization – addressing issues of delivery and installation.

Reichlmayr[10] reported their experiences using commercially sponsored senior capstone projects in the Software Engineering Department at Rochester Institute of Technology. This process includes soliciting sponsors, vetting and revising proposals in terms of collecting high-level requirements and identifying potential issues, allowing students to rank and select their preferences among the proposals, hosting meetings between students and sponsors to develop process plans, and giving final poster presentations.

Although the aforementioned process is similar to the one we used at UTD, there is a major difference. In addition to weekly meeting with industry sponsors at their sites, our students are also required to attend a three-hour lecture every week (as explained later in the paper). These classes are designed to cover special topics such as *ethical considerations for software engineers* and are taught either by faculty at UTD, or experts from other institutions or industry. They also provide opportunities for all the students (numbering in the range of 70 to more than 100, varying from semester to semester) to get together and review challenges that may be *common* to many projects. Many students have indicated that such discussion has saved them much time in many cases.

Why is Industry Collaboration Absent From Many Software Engineering Project Courses?

Although studies (such as those listed above) have shown the importance and the benefits of involving industry partners in software engineering project courses, this model still has not been widely adopted by many Computer Science departments. One of the main reasons (as also discussed in[8]) is that their faculty does not have the appropriate background and the confidence to mentor such projects.

Unlike other classes, teaching courses with real-life industry projects requires skills that cannot easily be learned from a textbook, but rather must be gained from practical experience developing real software in industry. As many faculty members have reached academia directly from graduate school without gaining any industry experience, they likely do not have the tools necessary to teach such a course. Industry professionals with the requisite software skills who make the switch to academia may also encounter challenges with these courses, as they may not possess the teaching skills to create an interesting experience for the students. In this case, special training on effective teaching practices may be called for.

Our Advantage: Intellectually and Graphically

The problem described above does not exist for the Computer Science Department at the University of Texas at Dallas, one of the few institutions in the USA with an accredited *Software Engineering* program in addition to its existing Computer Science program. B.S., M.S. as well as Ph.D. degrees are offered in both Software Engineering and Computer Science. The annual number of Bachelor's degrees in Computer Science we award each year has been among the top in the nation in the last few years. In fact, we are also one of the largest CS departments in the USA with more than 55 tenured or tenure-track faculty and an additional ten teaching faculty. Many of them had working experience in industry before joined us. Furthermore, the UTD campus is located in the center of a technology corridor including many high-tech companies such as Texas Instruments, Raytheon, Lockheed Martin Aeronautics, HP/EDS, Rockwell Collins, etc. This provides UTD with a significant geographic advantage as engineers from nearby companies can conveniently participate in our project-oriented capstone courses.

Starting from 2009, the School of Engineering and Computer Science at UTD has formed a UTDesign team in charge of all the capstone project courses in Computer Science, Software Engineering, Computer Engineering, Electrical Engineering, Telecom Engineering, and Mechanical Engineering. This is a collective effort with strong commitment from all departments, the school, and the university.

With respect to SE 4485, UTDesign connects companies in the Dallas metropolitan area to teams of senior students majoring in software engineering, who have been trained in the latest technologies, to help effectively expand their technical staff and work on engineering concepts or problems that have been languishing on a back burner for lack of manpower or other resources, software projects in need of further development or testing, and something they could handle in-house if only their technical staff had some extra bandwidth.

With a minimal investment ($80K for each project in Fall 2012), companies can have a talented team of 4 to 5 students and a faculty mentor working on a problem of direct relevance and importance to them. They also have an opportunity for early exposure to graduating seniors as potential hires. Moreover, the company has full ownership of all generated IP as all the students and faculty will sign a one-way NDA with the company.

Position of SE 4485 in the Curriculum

We first describe the position that the SE4485 course holds relative to the other courses in the Software Engineering curriculum at UTD. It is typically taken by students in their senior year and allows students to develop a group project while walking through the various phases of software development. The primary pedagogical objective is to instill competence in students in all software engineering activities. It is a capstone course that all undergraduate students who graduate with a degree in Software Engineering are required to take. The course itself has several pre-requisites at UTD such as Software Architecture; Requirements Engineering; Software Testing – Verification, Validation and Quality Assurance; and Data Structures and Algorithms to name a few. A visual representation of the hierarchy of courses, among which SE4485 sits towards the top, is presented in Figure 1. In the figure, an arrow going from course A to course B indicates that A is a pre-requisite for B.

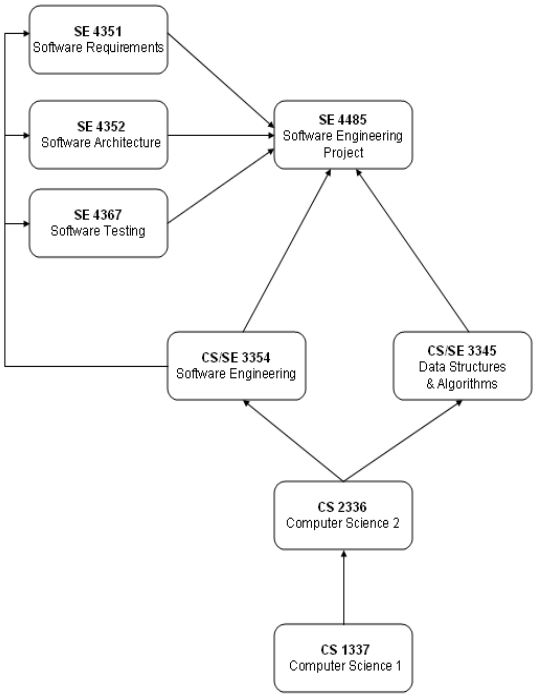

Figure 1: SE4485 in the undergraduate SE curriculum

Based on the relative position of SE4485 in the hierarchy we observe that it essentially integrates the knowledge that is learned and accumulated by students in their prior courses and gives them an opportunity to apply that knowledge collectively. This also implies that students are already supposed to have a strong background in software development models, etc.

Implementation

Figure 2 shows the entire of process flow for projects in SE 4485. Below we explain the implementation in detail.

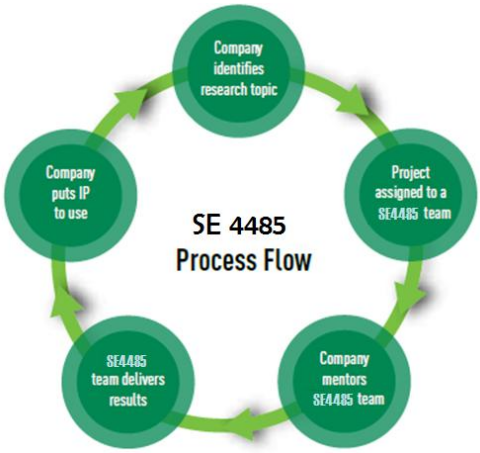

Figure 2. Process flow for SE 4485 at UTD[12]

The UTDesign team (which covers SE 4485) invites potential industry sponsors to submit project proposals two months before the beginning of each semester. Each proposal should have a project description consisting of an overview (with necessary block diagrams, sketches, photos, data, and graphs to help describe it), background and objectives (including a summary of the most important functional and non-functional requirements), deliverables (outlining what the company anticipates the students to design and build), and possible applications of these deliverables. The proposal should also specify the "***must have***" skills/knowledge that students need to have to get started on the project. However, if the company believes students can learn a skill quickly, without impacting the project schedule, then it should be listed under "***necessary skills***." More precisely, the necessary skills include those which students need to master for successful completion of the project. In addition, the proposal includes other information such as the work load expectation – how many hours per week students are expected to spend on the project; how often students will meet with corporate mentor; where these meetings will be held – on campus or at a corporate location.

A *corporate mentor* from the corresponding industry sponsor must be assigned for each project, who will work with a *faculty mentor* from the Computer Science department at UTD to co-supervise the project. To reduce the workload, each faculty mentor can only supervise at most two projects during any given semester. Also, to encourage strong participation from our faculty, they will be paid with an additional compensation of $1,000 for each project they supervise. This model has worked extremely well as the Computer Science Department at UTD has many faculty members with extensive industry

experience in the development of large complicated software systems and can effectively work with corporate mentors to supervise students on their projects.

The UTDesign team will review all the proposals, and select appropriate projects. One important point worth noting is that many companies, after joining UTDesign, continue to return each semester with an extended or entirely new proposal. A good example is Tyler Technologies which has been a sponsor for the last six semesters (of which the last four were mentored by the author of this paper). Each time they proposed a project built on top of the results of the previous semester's project. After a few semesters, a largely complete system which monitors the whereabouts of each inmate in a given prison had been developed by students in SE4485. This prototype was demonstrated to their customers and helped them win a contract. This kind of long-term collaboration provides a very healthy relationship between industry sponsors and the UTDesign team.

During the first lecture, all approved projects are presented by their corporate mentors to students, who will then form teams each with 4 to 5 members and select projects which they want to work on. Students are allowed to choose team members on their own, provided that each team is internally balanced in terms of the skills that the team members possess. This is not hard to regulate because, as mentioned before, enrollment in SE4485 is impossible without having taken certain pre-requisites which guarantee the fundamental knowledge that is required of each student in the course.

After that each project will have its own kick-off meeting with the industry sponsor. In this meeting, students and the faculty mentor are presented with more details. They will also discuss with the corporate mentor the objectives and deliverables, possible solution approaches, timeline and milestones, etc. The location and exact schedule for the weekly meetings will be determined as well. A project management plan is due one week after the meeting.

The following stage is to execute the project plan, which may be revised, if necessary, after a group discussion and approved by the mentors. Students need to submit a weekly status report, including peer-review related comments. They also need to submit documentation on *requirements*, *architecture*, *design*, and *test plan* for their project, using the templates posted at the SE 4485 course website on the specified dates. Late submissions will not be accepted.

An important observation is that many students in SE4485 have either full-time or part-time positions as software developers or testers at nearby industry corporations, which give them first-hand experience in how real-life projects are managed and how to work with other team members to complete their assignments efficiently and effectively. Other students who do not have the working experience are required to follow a professional standards guideline provided by the mentors. Furthermore, to ensure the smooth operation of the project, whenever possible a student with real industry experience is appointed as the team leader to assist the corporate and faculty mentors with project management and monitor the progress of the team.

Faculty mentors are required to give ***mid-term grades***, which is a great way to give an official notice to teams or specific students that they are performing up to expectations or running behind schedule; a false alarm is better than no alarm. Mid-term is a good opportunity to address any issues while there is still time.

In addition to the completion of the project on time with all the required functional and (if any) non-functional requirements specified in the submitted project management plan, each team is also evaluated using the following three criteria

- Cooperation: How open is the team to feedback and adjusting to changes?
- Leadership and teamwork: Are all the members able to work together in a productive way? Does the member assigned to lead a given part of the project adequately perform his/her job? Does the team leader properly supervise the overall effort? Note that at the end of the semester, each group is required to submit a written document describing the relative contribution of each team member. Any conflicts should be reported and peer reviews are to be performed. Unless suggested by the students, each member of a team will receive the same grade for the project.
- Dependability: How frequently does the team attend meetings? How does the team use project management techniques to help them monitor the progress? Does the team use any configuration tool to track all the documents and source code under development?
- Professionalism: How well prepared is the team for each meeting? How much effort is the team willing to put toward the project?

Besides the weekly meeting with their corporate and faculty mentors, students are also required to attend a 3-hour lecture each week during the pre-scheduled class time. Specially arranged presentations on selected topics are given by UTD faculty or external domain experts from industry, government agencies or other educational institutions. These lectures are often co-sponsored by various professional societies such as the IEEE Reliability Society and the IEEE Dallas Section, one of the largest IEEE sections. The purpose is to provide students with the state-of-the-art information, such as how software security and safety is applied in the real-life projects in industry, that is very unlikely to be covered by regular classes which they took previously. Nevertheless, such knowledge may be critical during project development. Another important topic covered relates to the ethical issues faced by software engineers. This is also emphasized by ABET accreditation.

At the end of the semester, we have a dedicated ***Senior Design Day***. All the students in SE 4485, along with those in CS 4485 (the Computer Science version of the capstone project course) must attend. It has two parts. In the morning, each project will give a 5-minute presentation, followed by lunch (provided), and then a 3-hour poster session in the afternoon, where students stand in front of their posters to explain their projects and answer questions from industry sponsors, faculty and students. See the photos in Figure 3. In Fall 2012, we had more than 150 attendees with 30+ from industry. Each team is evaluated for their presentation skills, slide contents, poster design, and project demo, by all the industry participants. The top 3 winners will receive cash prizes and award certificates. The project sponsored by Tyler Technologies and mentored by the author won first place in Spring 2012, while a Texas Instruments-sponsored project also mentored by the author won third place in Fall 2012.

Figure 3. The project sponsored by Tyler Technologies received the first place prize in spring 2012 (The second from the left is the corporate mentor and a manager from Tyler)

We would like to emphasize again that the corporate mentor is vital to the success of each project. Their responsibilities include 1) discussing potential projects with UTDesign faculty as early as possible, 2) presenting final project proposals to students the first week of the semester, 3) hosting kickoff meetings for students and faculty mentor to finalize project deliverables, 4) organizing subsequent weekly meetings to monitor the progress and answering questions related to the projects. The rule of thumb is that corporate sponsors must be available throughout the entire projects to mentor students, ensure progress, and transfer technology.

Benefits

There are two major benefits to the students: exposure to real-world engineering problems and access to potential employers.

Many students brought their resumes on the Senior Design Day which helped them receive interview opportunities and eventually job offers from the company representatives who stopped by their posters for explanations. An excellent example is the Tyler team (winning first place in Spring 2012; see Figure 3) which did such outstanding work that the company extended a job offer to four of the five members (the fifth one had already accepted an offer from a different company).

The benefits to the industry sponsors include

- opportunity to recruit the best graduates from a talented, diverse and creative pool of students
- progress on lower-priority projects without expending staff resources
- full ownership of results (as students and faculty will sign one-way NDAs)

- potential for high return on a small investment (the cost of each project in Fall 2012 is $80K)
- relationships with faculty who are subject matter experts
- networking with students and other corporate sponsors
- branding of their company and products with undergraduates and other sponsors

Below are two testimonials about UTDesign posted at its designated website[12]:

- "This student program is unique because we get a tangible work product out of it," said our sponsor at Rockwell Collins. "One tool the students created helps us in our day-to-day task of developing software and integrating it into our hardware. The time saved with the tool can now be invested in other projects."

- "Their energy, ideas, and motivation helped us a lot," our sponsor at Nth Fusion said about students who helped the company expand their educational applications from iOS to Android platforms.[12]

Furthermore, the major benefit to the faculty mentor is to help them establish a good partnership with industry sponsors which can lead to research funding on projects of mutual interest.

The success of the model described in the paper and the benefits to all the parties (students in SE 4485, industry sponsors, faculty mentors, the Computer Science Department, the School of Engineering and Computer Science, and even the University as a whole) have been proven by the results of the surveys returned by the students, the testimonials such as the above, and more importantly the job offers from our sponsors to the students.

Advice to Universities without Similar Resources

For universities with limited resources compared to UTD (especially those located in the area which does not have a strong industry presence), a good alternative is to start from a small scale by first working with only one industry sponsor, which can be accomplished even at a remote location as long as the designated corporate mentor has a commitment to participate in the regular project meetings via teleconference. A good example is one of the projects mentored by the author of this paper in Spring 2013, sponsored by NTT Data, Inc. in Boston, while the entire SE4485 team for this project resided far away on the UTD campus in Richardson, Texas. Nevertheless, such physical distance never prevented students in any way from establishing a close collaboration with engineers at NTT, and they completed their project successfully.

If it is still difficult to find any industry sponsor, then a non-profit organization can also be a good candidate. One project which the author mentored previously was sponsored by the Trinity Christian Academy in the Dallas area, where the SE4485 team helped design and develop a web-based professional development tracking system to include professional development and in-house course registration, teacher goal setting, and teacher evaluation. Both parties profited from this project, the sponsor in particular as they received a working software solution from the UTD students for a problem which had long been on the back burner due to a lack of manpower and technical expertise.

Instead of spending resources (which may not be available) to solicit many projects, a different approach is to divide all the students in the capstone project class into different teams, but all working on the same project. To make it even more attractive, an announcement can be made at the beginning of the semester indicating that there will be a competition between all the teams and a winner will be selected at the end whose project is deemed to have been the best. If possible, a small cash award can be presented to the winning team. This model was adopted very successfully by the author on a project sponsored by Tektronix before UTDesign was launched (with a strong commitment from the school which devoted significant resources) in 2009. This arrangement was found to be a powerful driver for the students to work even harder, encouraging a healthy competition that motivated students to truly deliver their best work rather than simply complete the project.

Conclusion & Future Work

The traditional SE4485 course at the University of Texas at Dallas was modified to bring in commercial companies as industry sponsors. Students were able to work on real-life projects under the co-supervision of the faculty mentor and the corporate mentor, and thus, the course did more than just to simulate an industrial setting. It actually gave students a chance to directly communicate with practitioners to receive a first-hand account of the work environments and lifestyles in the industry. All teams were able to complete the project successfully and the top three winners (among all the CS and SE projects, ranging from 18 to 25 from semester to semester) received cash prizes and award certificates. Industry sponsors enjoyed the deliverables to the extent that many were directly deployed to their operational environments to improve productivity. Thus, the benefits of such an endeavor have been multi-fold and have benefited all participants.

We are currently in the process of establishing a UTDesign Studio – a place with more than 10,000 square feet of dedicated space, where students and sponsors can create, innovate, design, and build their senior design projects (including those in SE 4485). A computer center, with special advanced software and hardware, will be available for exclusive use by students working on their UTDesign projects. A 70-seat auditorium, videoconference rooms, and a display of selected projects will also be included. We are confident that this state-of-the-art studio will help us move UTDesign to the next level.

Acknowledgments

Our sincere thanks go to the UTDesign team of the Erik Jonsson School of Engineering and Computer Science at the University of Texas at Dallas − especially Dean Mark Spong, who initiated the UTDesign program; Professor Gopal Gupta, Head of the Computer Science Department for his encouragement and support; Rod Wetterskog and Ed Esposito, UTDesign program coordinators, along with Professors Ivor Page, Jey Veerasamy, Miguel Razo, and Marco Tacca, for soliciting industry sponsors. We also wish to thank the students of SE4485 who were always enthusiastic and eager to excel, and made this course a pleasure to teach. Thanks also to all the industry representatives involved as project mentors.

References

1. Christensen, K., Rundus, D., and Prodanoff, Z. G., "Partnering with industry for a computer science and engineering capstone senior design course," in *Proceedings of the 2003 ASEE Southeastern Section Conference*, Macon, Georgia, USA, April 2003
2. Clifton, J. M., "An industry approach to the software engineering course," in *Proceedings of the 22nd Technical Symposium on Computer Science Education* (SIGCSE), pp. 296-299, San Antonio, Texas, USA, March 1991
3. Gehrke, M., Giese, H., Nickel, U. A., Niere, J., Tichy, M., Wadsack, J. P., and Zündorf, A., "Reporting about industrial strength software engineering courses for undergraduates," in *Proceedings of the 24th International Conference on Software Engineering* (ICSE), pp. 395-405. Orlanda, Florida, May 2002
4. Gorka, S., Miller, J. R., and Howe, B. J., "Developing realistic capstone projects in conjunction with industry," in *Proceedings of the 8th ACM Special Interest Group for Information Technology Education Conference* (SIGITE), pp. 27-31, Destin, Florida, USA, October 2007
5. Hogan, J. M., Smith, G., and Thomas, R., "Tight spirals and industry clients: The modern SE education experience," in *Proceedings of the 7th Australasian Computing Education Conference* (ACE), vol. 42, pp. 217-222, Newcastle, Australia, January 2005
6. Horning, J. J., and Wortman, D. B., "Software hut: A computer program engineering project in the form of a game," *IEEE Transactions on Software Engineering*, 3(4):325-330, July 1977
7. Kantipudi, M., Collier, K.W., Collofello, J.S. and Medeiros, S., "Software engineering course projects: failures and recommendations," in Proceedings of the SEI Conference on Software Engineering Education, pp. 324-338, San Diego, California, October 1992.
8. Klappholz, D., Almstrum, V. L., Modesit, K., Owen, C., Johnson, A., and Condly, S. J., "A framework for success in real projects for real clients courses," in *Software Engineering: Effective Teaching and Learning Approaches and Practices*, 1st ed., vol. 9, Ellis, H. J. C., Demurjian, S. A., and Naveda, J. F., Ed., pp. 157-190, Pennsylvania: IGI Global, 2008
9. McKeeman, W. M, "Experience with a software engineering project course," *IEEE Transactions on Software Engineering* 13(11): 1182-1192, November 1987.
10. Reichlmayr, T. J., "Collaborating with industry: Strategies for an undergraduate software engineering program," in *Proceedings of the 2006 International Workshop on Summit on Software Engineering Education* (SSEE), pp. 13-16, Shanghai, China, May 2006
11. Tahmoush, D., Fouché, S., McMaster, S., and Purtilo, J., "Enhancing software project management courses with industry participation," in *Proceedings of the 2009 International Conference on Frontiers in Education: Computer Science and Computer Engineering* (FECS), pp. 135-140, Las Vegas, Nevada, July 2009
12. UTDesign website (http://www.utdallas.edu/dept/eecs/utdesign), Jonsson School of Engineering and Computer Science, University of Texas at Dallas, January 2013
13. Wortman, D.B., "Software projects in an academic environment," *IEEE Transactions on Software Engineering* 13(11): 1176-1181, November 1987.