

Use of Service-Learning Projects in Graduate Software Engineering Curricula

Jeff Gilles
*Computer Science & Software
Engineering
Seattle University
Seattle, WA, 98122-1090, U.S.A
gillesj@seattleu.edu*

Roshanak Roshandel
*Computer Science & Software
Engineering
Seattle University
Seattle, WA, 98122-1090, U.S.A
roshanak@seattleu.edu*

1. Background

Seattle University's College of Science and Engineering has been involved in year-long, industry sponsored capstone projects in undergraduate and graduate programs in engineering and computer science for over 20 years. In our experience, industry-sponsored capstone projects provide clear pedagogical benefits, leading us to incorporate project-based learning into our curriculum. Three years ago we adapted our Masters of Software Engineering program to leverage real projects as case studies in various courses.

This was done through the Academic Service-Learning (ASL) approach that utilizes community service as a means of helping students gain a deeper understanding of course objectives, acquire new knowledge, and engage in civic activity. In this paper, we discuss our experience with incorporating academic service learning into our graduate curriculum.

Service-learning is a teaching and learning strategy that integrates meaningful community service with instruction and reflection to enrich the learning experience, teach civic responsibility, and strengthen communities [1]. While the projects are real projects they are solely sponsored by non-profit organizations and all the work is done on a pro bono basis. The Center for Service and Community Engagement at Seattle University [2] and the College of Science and Engineering Project center work together to facilitate communication and interaction between the campus and the greater community in conjunction with Seattle University's broader mission. Consequently, all the projects have a strong social justice and community benefit component which makes the experience more fulfilling for the students.

2. Our experience

In our Master of Software Engineering program, four courses extensively use ASL projects and several other classes (Software Security, Software Architecture, Testing, Project Management) may choose to use project artifacts as part of the instruction. Each academic year, the students begin working on an ASL project in the Software *Requirements* class. While learning techniques for requirements gathering, verification and validation and documentation, they meet with our project sponsors. Teams of four students organize requirements gathering workshops and take turns through the elicitation process. Each team both gathers requirements and develops critiques of the other team's interviewing skills. By the end of the quarter, each team develops a complete Vision and Scope and Requirements Specification for the project. Students' learning is reinforced by interacting with actual clients, and by being able to compare their work products with the other teams.

In 2007 and 2008, the requirements artifacts generated in this class served as a starting point for a second set of courses: *Software Modeling* and *Data Modeling*. The former is focused on

application design issues while the latter discusses data design and modeling. Each class took a team based approach to the project and at the end of the course the students created artifacts detailing application design and related data models. These artifacts then flowed into a course focusing on *Software Construction* with the intention of developing a functioning software application by the end of this class.

Our experience demonstrated that this approach posed some serious challenges. Using a real customer for the requirements class has proven invaluable to the students learning and has been very successful from the client's perspective. Moving those requirements into a data modeling class also seems to work well since creating a data model for these projects has not been an overwhelmingly large problem allowing the instructor time to focus on other important topics not covered by ALS project. However, in the Software Modeling class we faced some challenges that primarily relate to the type and volume of work necessary to create a complete object oriented design of the system. The volume of work can overwhelm this class with two negative impacts: (1) the instructor may not have time to cover other important topics, and (2) producing a complete design of the system may anchor the students learning to practice the same topics in several areas while other topics remain untouched. Finally, the Software Construction class has difficulty covering all required topics while completing the implementation of the product within the quarter time frame. These challenges can lead to disappointment for the community partner.

Given our mixed experience with this approach, we have decided to leave the Software Requirements and Data Modeling classes unchanged, while modifying the expectation for the Software Modeling and Construction classes for next year. For the latter two courses, we will begin creating a repository of application design case studies and reference models and assign targeted portions of these to students. This will enable us to keep the challenges fresh, while ensuring sufficient coverage of all necessary topics. Community partners' expectations are being reset accordingly.

3. Benefits to our students and curriculum

First, our classes are offered in the evenings to cater to our professional students who bring several years of work experience into classroom. While faculty provide required readings and some lectures, a large portion of our class time is spent on facilitating discussion on the related topics and applying them to the project. This allows the students to apply what they are learning immediately to a real project which forces them to understand the trade offs between academic approaches to software development and practical constraints.

Second, by carrying a case study through a sequence of classes, students understand the intra-class synergies and how the topics of various classes work together for successful projects.

Third, students meet with real members of the community who often passionately describe their programs providing an intense motivating factor for many of the students.

Although we have experienced some issues, as noted, we have had some very interesting projects and this experience has been a tremendous success. We believe in the years to come, this type of integrated project-based learning will continue as an integral part of our curriculum.

4. References

- [1] Learn and Serve America's National Service Learning Clearinghouse <http://www.servicelearning.org>
- [2] Seattle University, Center for Service and Community Engagement <http://www.seattleu.edu/csce>