Capstone Project Experiences: Integrating Computer Science & the Liberal Arts

Gail Miles
milesg@lrc.edu
Computer Science & Mathematics Department
Lenoir-Rhyne College
Hickory, N.C.

Kathleen M. Kelm
kkelm@edgewood.edu
Computer Science Department
Edgewood College
Madison, WI

Abstract

This paper provides an overview and comparison of a one-year capstone experience using a two-semester software engineering project-based model, offered by two small, liberal-arts colleges, Lenoir-Rhyne College and Edgewood College. The capstone experience has been offered at both colleges for the past 15 years. The paper details the structure of the capstone project experience, concentrating on the integration of the liberal arts with the technical education of CS within the framework of project and team work. A discussion on the benefits and challenges of integrating liberal arts follows, with a specific focus on student preparedness for employment as IS/IT professionals. Recommendations for others teaching a professional program within a liberal arts environment conclude the paper.

Keywords: IS Education, Capstone Project, Liberal Arts, IS2002 Curriculum ACM Curriculum, Software Engineering, SDLC

1. THE INSTITUTIONS

Lenoir-Rhyne College (located in Hickory, N.C.) is a small, private, liberal arts institution with a student body of 1600. The Computing Science and Mathematics Department has approximately 50 majors in CS, IT, and Math. The computing program has followed a curriculum in the spirit of the ACM-approved curricula.

Edgewood College (located in Madison, WI) is a small, private, Sinsinawan Dominican sponsored liberal arts institution, which bases its curriculum on the Dominican *studium* of study, reflection and action. With a student body of 2500, the Computer Science department has approximately 55 majors in CIS and CS Teaching, of which 40 are returning adults. The CIS curriculum follows the IS2002 Model, endorsed by ACM, AIS and AITP.

2. THE CAPSTONE EXPERIENCE AT LENOIR-RHYNE COLLEGE

There are numerous challenges in maintaining a strong program in Computer Science or Information Technology in a small college but even more so in a college that takes its liberal arts commitment seriously.

Students are required to take a large core of general-education courses and the courses in a major are limited, thus, each course must be carefully chosen to successfully prepare the student. At Lenoir-Rhyne, great thought was put into the capstone experience. A capstone experience can involve many activities. According to ACM Curriculum 2001 the "culmination of an undergraduate computer science degree should include a final-year project that requires students to use a range of practices and techniques in solving a substantial problem" (ACM, 2001).

Lenoir-Rhyne has chosen a Software Engineering approach with a 2-semester project-intensive sequence. This approach challenges students to develop an extended project which solves a problem for a customer, with limited help from the instructor. The capstone experience requires that students apply their knowledge, not only of their technical skills they have learned, but also the knowledge and skills of the liberal arts such as Ethics, Communication, Psychology, Sociology, and Mathematics. The courses address the issues that face professionals in the field today at the same time they are building a project "Not only are the students using their professional skills, but the strong liberal arts education is necessary for the students to complete the sequence. They are expected to demonstrate a high level of written and oral communication skills. They are required to develop strong external documentation (Project Notebook, User Manual, and Technical Manual) since they are producing a real product that will be maintained by the customer after they hand it over. They are also required to present their work multiple times to the customer, the faculty, and other students in presentation format. They are part of several peer reviews so they need the ability to critique the work of others in a constructive view and professional manner" (Miles, 2004). The Learning Outcomes for the experience are shown in Table 1.

Table 1. Learning Outcomes
Understand the ethical implications of software development.
Understand social interactions and motivations in customer relations
Learn to work effectively with colleagues in a team environment.
Demonstrate Advanced Critical Thinking: --Assess the feasibly of the project --Analyze the requirements for a software system to produce a software design. --Research alternative implementations --Use software metrics to determine cost and time estimates. --Assess project risks --Peer Evaluations
Demonstrate professional communications skills
Demonstrate professional presentation skills
Demonstrate project management skills.
Understand the dynamics of the Human-Computer Interface

Students are required to design the real-world project in modified chief programmer teams (Brooks, 1974) with a real "customer" in the fall and implement the project in the spring. The teams remain the same for the full year. The first course follows a Software Engineering template for designing a software product. Students do not write code in the first course although they use a prototyping tool for design for a "proof of concept". The second course is a Capstone project which implements the project designed in the first course. Many topics of CS culminate in this sequence. Everything that they have learned comes together to allow them to prepare for either the world of work or graduate school (Miles, 2004)

First Course

The Software Engineering course is a 4 credit hour course that explores all aspects of the software engineering process including identification, definition, design, analysis, verification and management of basic requirements. It is taken by students in the first semester of their senior year. Students are taught to use the software engineering model similar to other engineering paradigms where students use tools and skills to allow them to manage a programming project, understanding the technical, management, economic and social implications. Each of the team members is assigned a project role – principal architect, project manager, chief programmer, quality assurance manager, and librarian.

The projects are solicited by the department from non-profit groups in the community. Over the years, several approaches have been used to identify appropriate projects. These include:

a. commercial projects in the community
b. college customers (such as the Dean's office, Admissions, etc)
c. Different simulated projects with simulated customers (usually faculty).
d. One simulated project done by all groups.

The non-profit projects seem to be the best solution. The students' work has the potential to make a significant contribution to the customer organization. The customers are very enthusiastic about being chosen and are much better customers. They meet regularly with the student groups and are happy to follow through with any project requirements (meetings, testing, walkthroughs). It helps Lenoir-Rhyne because it gives back to the community and goes a long way in the Public Relations arena. It helps the students because they are able to have a little more flexibility in implementing the project the way it should be done, rather than a dictate from the institution.

The students are responsible for managing the customer relationship and work closely with their customers. This allows them to develop key social skills that are not often addressed in the "clean-room" environment of undergraduate education. One area where this is invaluable is in the area of defining and assessing non-functional requirements. These often only appear when the "human" factor is involved. This gives students the opportunity to learn how to deal with less concrete issues in software development.

Second Course

The *Capstone Project course*, taken the last semester of the student's senior year, has two separate goals. First, student teams implement the project they designed in the first course by coding to the standards, testing, evaluating, and maintaining quality assurance. Students determine the best approach and the best tools to complete the job (CASE, IDE, APIs, Languages) to complete the project. The Chief Programmer of the teams is responsible for providing training on these tools. A second goal is a series of seminars to survey professional topics important for technical professionals. These include Ethical Issues, Legal Issues, Software Metrics and HCI.

Course Assessment

The students benefit considerably from their involvement in working with a single project for almost a year. They learn much about social issues by working with a customer and team members for a full year. The assessment criteria for the first course are divided into assessment of topics of Software Engineering and Project Work.

Table 2: Assessment Criteria
<u>Software Engineering Course</u>

Tests: (3 exams 10%)
Assignments 15%
Final Exam: (One (1) cumulative final @ 15%)
Projects: (Project Deliverables and Walkthroughs- 40%)
Of the 40% of the course assessment allocated to the project, marks will be split among the documents in the following way:

5%	Project Plan
15%	Software Requirements Document
15%	User's Manual
5%	Configuration and Quality Assur ance Plan
15%	Software Specifications Document
15%	Final Design Document
20%	Prototype and Final Demo
10%	Project Notebook

Capstone Course
The project work in the second course has greater weight.

One exam at midterm (10%)
Project (Project Deliverables 40%)
Walkthroughs and Reviews 20%
(5 of these)
Other Project work - 15%
Final Exam 15%

Challenges

When functional, the members of the programming teams provide support and encouragement to each other and foster the belief that help from team members was readily available (supported by Willis, 2000). Students who report a lack of knowledge or skill in some area receive support from other team members. However, there are times when teams do not work as well as planned. Usually, these dysfunctional teams are unsuccessful because of human issues, rather than skill issues, such as a member who is not pulling his load or not completing the work assigned to them on time. At the extreme level, teams can "fire" a team member who is not producing. For this to be done, there is a written process, similar to personnel evaluations in a "real" place of work, and the instructor must approve it.

Another significant challenge can be categorized as "customer" issues. Because students are working with a real customer, there are no artificial relationships set up by an instructor. Thus, all student groups must cope with real customer issues. Some of these are listed in Table 3.

Table 3: Customer Issues
Not able to articulate requirements
Changing requirements after the requirements analysis
Original contact person is replaced
Rescheduling meetings at the last minute
Not meeting deadlines for Beta testing of module units

A final significant issue is how to evaluate individual effort on a group project. Although students are required to complete a peer review after each deliverable, it is still difficult to determine the contribution of each member of the team. Not only is it difficult to identify the individual effort, but, also each member of the team has a unique role. It is difficult to fairly assess the activities of each of the roles. There are significant differences in the effort of a chief programmer and the quality assurance manager.

3. THE CAPSTONE EXPERIENCE AT EDGEWOOD COLLEGE

Similar to Lenoir-Rhyne, Edgewood College students must complete a full set of liberal arts curriculum while integrating a strict rotation of CS courses. Further, Edgewood requires the completion of a Human Issues project, designed to bring together the liberal arts and professional studies programs for the purposes of examining the human experience, through a service learning project. As an integrative and transformative experience, students perform scholarship in three disciplines, appropriate to the project, examine the issue using the Sinsinawan Dominican values of compassion, truth, justice, partnership and community, and present their experience to the college wide community.

At Edgewood, the curriculum sets are regularly (every two years) reviewed by practicing IS/IT professions, external to Edgewood. The review process takes place in many forms: focus groups, survey and examination of student projects. Alumni also provide yearly feedback, in terms of how well the degree has prepared them as IS/IT professionals, in skills and competencies and as lifelong learners.

Edgewood follows the standard System Development Life Cycle (SDLC) approach for its capstone experience. Students have experience in the traditional, waterfall and object-oriented iterative processes, prior to enrolling in the two course set. This allows the faculty, serving as the project sponsor, the most flexibility in choosing the client and project. The incoming class is assessed, in terms of individual student strengths and weaknesses, so as to ensure that the project type is matched appropriately to the project team's overall capabilities. This process has shown to be an important element of the project selection process, so that the project provides the best learning environment for students. It is worth noting that the project does not necessarily have to be a success, *per se*, as the primary goal of the capstone experience is for students to experience, first hand, the challenges of a fully developed system that solves a client problem.

Under the best circumstances, the optimal client would be a small, non-profit organization, located in the greater Madison community. Unfortunately, the time constraints of when the project must be completed and availability of small non-profits do not always coincide appropriately, so occasionally the project is chosen to serve an internal client at Edgewood.

Similar to Lenoir-Rhyne College, students at Edgewood College are expected to utilize knowledge acquired from the other courses completed as part of their degree. Frequently, students resist the need for classes on accounting or on religion, yet remark, upon reflection that they find these courses to be particularly helpful during the capstone project. Further, employers regularly report that, as employee attributes, the areas of business acumen, personal communication, presentation skills, and project management have increased in importance (Galllivan, Truex and Kyasny, 2004). Denning & Dunham (2001) refer to these as value skills, which, when partnered with deep technical skills, position the IS/IT professional to be capable of providing excellent in customer service.

While students begin to develop these skills in prior courses, the capstone experience further refines these skills in the first course, with expectations that all competencies can be demonstrated during the second course. The Learning Outcomes for the entire capstone experience are shown in Table 4.

Table 4. Learning Outcomes
Demonstrate an understanding of SDLC by undertaking and completing a client project
Identify and utilize the core structures and process employed when managing IT Projects
Differentiate the uses of team roles, as they pertain to SDLC
Demonstrate Capabilities of an emerging IS/IT Professional by: --Assessing the project goals in terms of feasibility and client expectations for success and acceptance --Researching and recommending technology and resources --Using project management structures (PMBOK) for overall management of development cycle --Participating as a contributing member of a team, by fulfilling needed roles and responsibilities
Demonstrate professional communication and presentation skills
Demonstrate technical capabilities in: Software Design, Programming, Database Construction, Web oriented End User Interface Design and Implementation strategies

First Course

The Information Systems: Tools for Enterprise Development (CS480) is a 4 credit hour course that explores all aspects of building information systems for enterprise deployment, including problem identification, solution selection, development and implementation using web development tools. It is taken by students in the first semester of their senior year. Students are taught both web development constructs within the SDLC cycle, using Object Oriented Analysis & Design (OOAD) standards, utilizing skills already demonstration in previous coursework. There are two distinct phases of the course: during the first seven weeks, students learn advanced web design, using a variety of scripting approaches to develop a simple, web-enabled database solution, following PMBOK standards. The second seven weeks are spent working as a team to complete the initial project development phases (problem definition, initial analysis and technology recommendations) for what will become the capstone project during the following semester. Team roles are chosen by the class, based on how they perceive to be the best match of student capabilities and skills needed.

While completing this course, students are also required to record learning discoveries, using Pacific Crest's journaling framework (Beyerlein & Apple, 2005) of Strength, Improvement, Insight (SII). As the course progresses, students find that the journaling assists them in finding the interconnections between the seemingly disparate chunks of knowledge, reinforcing the concept that as practicing IS/IT professionals, they will be continuously learning new content and quickly integrate it into existing knowledge. By simulating Wenger's (1998) theory of communities of practice, the instructor hopes to provide the students the opportunities to move from a way of knowing to a way of being (Yorks & Kasl, 2002).

Suggestions for capstone projects are solicited by the department from non-profit groups in the community; throughout the academic year, additional, unsolicited recommendations are received and maintained in a small database. The selection of a project takes into account several factors so as to ensure that the ultimate goal of a quality learning experience will be achieved, including: (1) competencies and capabilities of student project team (2) type of project and timeframe for completion, (3) ability of client to fulfill the time commitment implied by the project plan and (4) that there exists a recognition that the student work is a valuable contribution to the organization being served. This last point is particularly important as students are expected to work within a project budget and are aware of the value of their volunteer work. Further, students often refer to their capstone project when citing work experience on their resume. Finally, from a financial accountability perspective, non-profits, who receive the result of the capstone project, in terms of a tangible product will report this gain on their financial statements. For more information about how the volunteer value is determined, see the following website: http://www.independentsector.org/programs/research/volunteer_time.html. Once the capstone project and client have been determined, the instructor meets with the client to review the overall goals of the project, from a learning perspective and to ensure that there is an appropriate match between students' capabilities and the project requirements.

The students are responsible for managing the initial assessment of client needs and work with the instructor to review their findings so as to complete the initial analysis and recommendations, in what would be considered a 'safe environment'. This insures that before the project moves forward, all initial negotiations of duties and responsibilities are clearly understood and articulated in writing (Block, 2002).

Second Course

The *Capstone Project* course (CS492), taken the last semester of the student's senior year, has two separate goals: (1) The students complete the design of the project identified in CS480, construct, document and implement the system, following the project plan and (2) The students engage in a set of personal reflections whereby they identify connections between liberal arts and their professional study, the results of which are presented to the Edgewood College community at the end of the semester.

Students determine which technology to use, based on the original analysis, often finding that their knowledge previously acquired in other coursework only serves as a springboard for further in-depth exploration of system-specific development tools. The project team roles are assigned as the result of a self selection activity, whereby the class self-identifies key attributes (anonymously) and aligns the attributes around requirements of the six roles of the team: Project Leader, and Lead Analyst, Designer, Programmer, Tester and Documenter. Each student serves in a primary role and backup roles, so as to provide adequate levels of resources during key phases of the project

Course Assessment

The students benefit considerably from their involvement in working with a single project over the course of two semesters. The first, small individual project, completed in CS480 serves as a 'test run' of student's capability to organize and reflect on a collective of knowledge tests and build a schema to use when retrieving the knowledge at a later date (Wenger, 1998). During CS492, the focus of serving a client through construction of a solution that directly solves a problem provides the context within which they experience, first hand, the benefits of good project management. Further, they find that while they can articulate the attributes of a quality project team, by the end of the two semesters, they have specific examples of when and where project teams succeed and fail. Employers (located in the greater Madison area) continually emphasize the need for better project management skills. Lastly students learn that by managing themselves as a team, they can provide better service to the client to ensure that a high quality product is produced and delivered.

The assessment criteria for both courses is loosely based around three goals: (1) demonstration of technical capability and competency, (2) demonstration of project management acumen and (3) student reflection of transformative learning processes, while completing their degree program. Table 5 presents the specific assessment criteria in terms of percentage breakdown.

Table 5: Assessment Criteria

CS480 (Information Systems: Tools for Enterprise Development)
10% Mid-term test
15% Learning Journals
25% Individual Project
30% Collaborative Project
5% Class Participation
CS492 (Capstone Project)
70% Capstone Project
10% Project Peer Evaluation
20% Learning Assessment Report

The students contribute to an overall grade for peer assessment using a rubric that assesses levels of effective communication, equality of work and timeliness of work.

Challenges

Students encounter two types of challenges: (1) Time management misalignment with capabilities and capacities of team roles and (2) lack of continuous validation of client expectations. Often the cycle of the course involves a realization, around the time of the spring semester break (3rd week of March) that the time constraints of working as a team are not easily achieved or have not been adequately addressed. Further, once construction begins, students must face the reality that although they bring to the project a high level of capabilities and technical competencies, the project is not quite as simple as first thought and consequently, requires acquisition of new knowledge. Communication between team members is tested, during the month of April, and each set of individuals assigned to tasks must demonstrate high levels of trust in each other, so as to not derail the production of the final product. Past experience has shown that teams who enjoy higher levels of project success also have high levels of regular communication between the team ***and*** with the client. Students who do report frustration with other team members usually connect it to issues surrounding unclear communication and/or lack of precision about client requirements.

Much like the experiences at Lenoir-Rhyne, students recognize the value of 'sticking together' and work hard at producing a high quality product. Late nights and a sense of camaraderie results in a feeling of 'we did it together' and on the night the team formally presents the project to the client for acceptance, each student, during their presentation, mentions their gratitude for the support of the other team members.

It is also at the formal presentation, that students speak directly to the value of their liberal arts education; in particular they see it as a strong foundation and base for their growing stature as a member of the IS/IT profession.

4. CONCLUSION

The goal of any Capstone experience is to function as a bridge that integrates the theory of Computer Science with the practice of Software Engineering and Systems Development. It allows students to make the transition from student to computer professional. The foundations that students have learned from the liberal arts make the students very marketable with perspective employers. For both Lenoir-Rhyne College and Edgewood College, there is evidence that the Capstone experiences achieves this goal: (1) employers in the community value the experiences students bring to their organization and specifically discuss the group project activities they have completed and (2) the non-profit community provides positive feedback in the success of the student projects that would not have otherwise been available to them.

The authors believe that the continued review and updating of the curriculum that precedes the Capstone project is essential to maintaining it as a high quality experience. Further, such regular reviews assure that those completing degrees in CS, CIS and IT at liberal arts colleges are given the opportunity to demonstrate, in a concrete way, the applications of skills, both technical and interpersonal, thereby strengthening the profession and as well as the connection between theory and practice.

References

ACM. *Computing Curricula 2001, Computer Science Volume, Chapter 13, Institutional Challenges.* Retrieved July 22, 2006, from http://www.sigcse.org/cc2001/cs-challenges.html

Beyerlein, S. & Apple, D. (2005). *Faculty Guidebook*. Chicago: Pacific Crest Publications.

Block, P. (2002). *Flawless Consulting.* New York: John Wiley & Sons.

Brooks, F. (1982). *Mythical Man-Month.* London: Addison Wesley

Computing Curricula 2001*. Computer Science Volume. Chapter 7. Introductory Courses.* Retrieved July 22, 2006, from http://www.acm.org/sigcse/cc2001/cs-changes.html

Denning, P and Dunham, R. (2001) The profession of IT. *Communications of the ACM*, *44*(11), 21-25.

Galllivan, M., Truex, D., and Kyasny, L. (2004) Changing patterns in IT skills sets (1988-2003). *The DATA BASE for Advances in Information Systems, 24*(3), 64-87.

Gomory Ralph E. Sheffield Lecture - Yale University January 11, 2000 Internet learning: Is it real and what does it mean for universities?. *Journal of Asynchronous Learning Network, 5(1),*

Kolb, D. A., (1984). *Experiential Learning*, Upper Saddle River, NJ: Prentice-Hall.

Li, Frances. (2001). *Supporting Collaborative Teams in Engineering Education*. University of Berkley. Research funded by grants from Intel Corporation and the National Science Foundation.

Miles, G. Computing curriculum 2001 in a small liberal arts college. *Journal of Computing Sciences in Colleges, 18*(6),

Wenger, E. (1998). *Communities of Practice*. New York: Cambridge University Press.

Willis, B. (2000). *Distance Education - Strategies and Tools* and *Distance Education - A Practical Guide*.

Yorks, L. & Kasl, E. (2002). Toward a theory and practice for whole-person learning: Reconceptualizing experience and the role of affect. *Adult Education Quarterly, 52*(3), 172-192.