

A Project-Centric Curriculum Design

Dennis Drinka, (Email: afded@uaa.alaska.edu), University of Alaska Anchorage
Minnie Yi-Miin Yen, (Email: afmyy@uaa.alaska.edu), University of Alaska Anchorage

ABSTRACT

Student success was the motivation for evolving an individual project-based course into a project-centric curriculum. A one semester project was first extended across a sequence of three interrelated courses tied together through their focus on the success of small team projects that spanned those courses. This sequence was then targeted as the core of a redesign of the entire program curriculum focused on project and student success. Currently, the department is in the process of introducing the measurement of project success as a tool for assessment and control of the department's learning objectives. An overview of the design of this curriculum, lessons learned from developing it, and benefits of this type of curriculum in quality of student learning, community engagement, and reputation of the university, will be discussed.

BACKGROUND

The MIS curriculum at our university was originally designed based on the IS'97 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems developed collaboratively by ACM (Association for Computing Machinery), AIS (Association for Information Systems) and AITP (Association of Information Technology Professionals) (AITP, 1997). According to these guidelines, the scope of Information Systems as an academic field encompasses two broad areas: (1) acquisition, deployment, and management of information technology resources and services (the information systems function) and (2) development and evolution of infrastructure and systems for use in organizational processes (system development).

As a result of a curriculum redesign, our program has evolved so that a community-based project is the focal point in three core courses in the curricula: systems analysis, project management, and system design and development. At the same time, most of the other courses in the program support the success of this core and its project. The relationships between courses in the new curriculum can be illustrated as shown in Figure 1.

Like most programs based on the model curricula for MIS, our program is built around five interrelated categories of courses (networks, database, programming, web development, and systems analysis/project management). These courses build upon each other, can be integrated conceptually through the system development life cycle, and can be taken in different combinations to provide flexibility for student career choices.

THEORIES SUPPORTING AND ENABLING THE CURRICULUM DESIGN

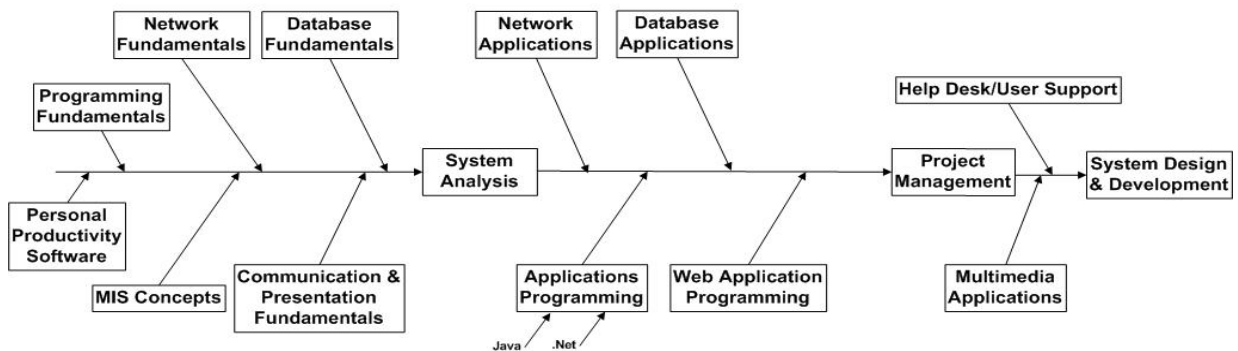
The benefits of this project-centric curriculum are based on firmly established learning theories including those related to engaged, team-based, community-based, and project-based learning.

Engaged Learning

This curriculum is consistent with the engagement theory for technology-based teaching and learning (Kearsley and Shneiderman, 1999). This theory has emerged from Kearsley and Shneiderman's experiences teaching in electronic and distance education environments and has much in common with many of the well-known community and project-based learning theories. The fundamental idea underlying this theory is that students must be meaningfully engaged in learning activities through interaction with others and with worthwhile tasks. Through engaged learning, all student activities involve active cognitive processes such as creating, problem-solving, reasoning, decision-making,

and evaluation. Students are intrinsically motivated to learn due to the meaningful nature of the learning environment and activities.

Figure 1: Project-Centric Curriculum
Community Project-Based Curriculum



Team-Based Learning

A pedagogic model of Relate-Create-Donate (Shneiderman, 1998) proposes that students need to work in teams and work collaboratively to produce ambitious projects that are meaningful for someone outside the classroom. These three components, Relate-Create-Donate, imply that learning activities: occur in a group context (i.e. collaborative teams); are project-based; and have an outside (authentic) focus.

The first principle (the "Relate" component) emphasizes team efforts that involve communication, planning, management and social skills. The second principle (the "Create" component) makes learning a creative, purposeful activity. Students have to define the project (problem domain) and focus their efforts on application of ideas to a specific context. Project orientation is the essence of problem-based learning approaches that are often used in medical and others types of professional education (Barrows and Tamblyn, 1980). The third principle (the "Donate" component) stresses the value of making a useful contribution while learning. Ideally each project has an outside "customer" for whom the project is being completed. In many cases, the projects can be work-related, i.e., an activity that fits into a team's occupational or career interests. The authentic learning context of the project increases student motivation and satisfaction. This principle is consistent with the emphasis on school-to-work programs in many school systems and colleges, as well as the "service" philosophy of contemporary corporate training efforts (Jacoby and Associates, 1996).

Community-Based Learning

Community-based learning is a pedagogy that embeds academic study into community service projects. Local community groups are in need of assistance while students are in need of real-world experiences that relate to their course material so that they might gain a better understanding of real issues and practices by working on actual problems and with nonacademic groups (Lazar and Preece, 1999). While benefiting the community through work performed, this type of learning also increases students' understanding of their academic subjects by putting them directly in touch with what is being studied, as differentiated from learning in the abstract (Keeton and Tate, 1978).

The student in an experiential learning situation learns from actually, directly, and actively working with the objects of learning, usually under controlled conditions and under the instructor's supervision (Venkatesh and Small, 2002). Lewin, Dewey, and Piaget (Lewin, 1951, Dewey, 1938, and Piaget, 1970) have viewed learning as a continuous, adaptive process driven by experiences where the student is active and engaged with the subject matter at hand. Furthermore, in the technology field, computing is often viewed as a social technology (Iacono and Kling, 1988). Individuals must learn to consider technology not only within the context of work practices, but also with that of organizational culture and power. Community-based learning is a productive way to introduce students into the social context of their studies. Learning becomes the product of participation in the actual practice, interactions and relationships that constitute the community-based project (Lave and Wenger, 1991).

Project-Based Learning

Project-based collaborative learning refocuses learning from short, isolated, teacher-directed activities, to activities that are student-directed, long-term, and interdisciplinary--all within a team-based, collaborative framework. Collaborative learning refers to those methods of instruction where students work together to reach a common goal. Within the context of a community project, this collaboration is extended to include the owners and users of the project deliverables. The instructor is included within the collaboration in a supportive role. With collaborative learning, students increase their understanding and knowledge by sharing ideas with their team members and are dependent upon each other for project success. Project team members must work together in learning and knowledge-building communities, exploring each other's skills while providing social support and modeling and observing the contribution of each member (Jonassen, 1995). According to Vygotsky's zone of proximal development theory, student learning is extended by the difference between what a student can do alone and what he/she can do with supportive collaboration (Vygotsky, 1978). Extending the collaboration pool by including the sharing of ideas, dependencies and supportive collaboration, and the existing and widely based knowledge of system owners and users extends these benefits well beyond the scope of student-only collaboration, much less the traditional teacher-centered approach.

Projects provide an ideal organizational structure for supporting collaborative learning because with both projects and collaborative learning, it is necessary that the activity is directed toward solving a problem or creating or discovering something (Schrage, 1991). Project collaboration also contributes to students' social learning through their observation of behaviors, attitudes, and emotional reactions of teammates, owners and users (Bandura, 1977), (LeJeune and Richardson, 1998). From the perspective of social constructivism theory, members of the community serve as active agents in the construction of outcomes and activities that produce a development cycle in the social setting (Shaw and Shaw, 1999). Therefore, combining project-based learning with community-based learning integrates the benefits of both.

THE EVOLUTION OF THE CURRICULUM

In our original curriculum, projects were initialized, designed and developed within a single semester. The scope of these projects was necessarily narrow because teams of students were expected to initiate, analyze, design, construct, and implement a system, all within the fifteen weeks of a single semester. This time was insufficient for a major project, especially since it would be the students' first attempt at such an endeavor. As a result, the analysis and planning phases were usually neglected; the quality of testing was minimal and inconsistent. Planning was nearly nonexistent, development was based on a code-like-hell lifecycle, risks were high, and project success was based almost exclusively on team heroics. The development processes varied from team to team and were therefore very unpredictable.

Project Management And System Development Courses Integrated

The first major change in the curriculum was the joining of the project management course to the system design and development course. With the integration of these two courses, the project management course could be used to establish requirements that insured that good management practices were in place before any given project would be allowed to proceed to development. An outline for project plans was developed by the instructor to guide

students in the development of plans. In addition to typical planning items such as schedules, budgets, and feature sets, the outline also requires life-cycle assessment, risk assessment and risk management plans, uncertainty assessment, change management plans, motivational plans, and communication plans. Complete project plans from previous projects were made available for review by students. The best prepared sections from among all previous plans evolved into models of best practice and were incorporated into outlines and lectures so they could be used in subsequent plans. Students tailored and refined each section to meet the requirements of their particular projects; while all sections were required to conform to the framework established in the outline. Each section of the plan was graded separately and weighted equally so as to insure that the plans were comprehensive, and consistent with the outline. Quality improvement within individual plans was motivated by assigning above average grades to sections that extended and refined the existing models, were particularly innovative, or in some other way, indicated original thought and outstanding quality. Improvement across all plans was achieved by modifying the outline based on the observation of overall project development successes and problems. As a result, over time, the quality and consistency of the plans increased while the time for preparing the plans, while quite high, decreased.

The linking of the project management course with the development course, and the establishment of the outline for the project plan moved the curriculum design into a new stage. The outline formed the seed of a comprehensive methodology that would continue to develop. In this stage, policies for managing project development and procedures for implementing those policies were established and formalized through the plan outline. Students could, and were encouraged to, repeat practices that were successful on previous projects through access to previous plans and through class lectures. At this stage in the curriculum evolution, a planning and development process was initially put into place. This process started being used. It was documented, enforced, taught, measured, and provided for improvement. Controls for managing this process were built into the work review and the grading schemes of the courses.

Systems Analysis Course Included

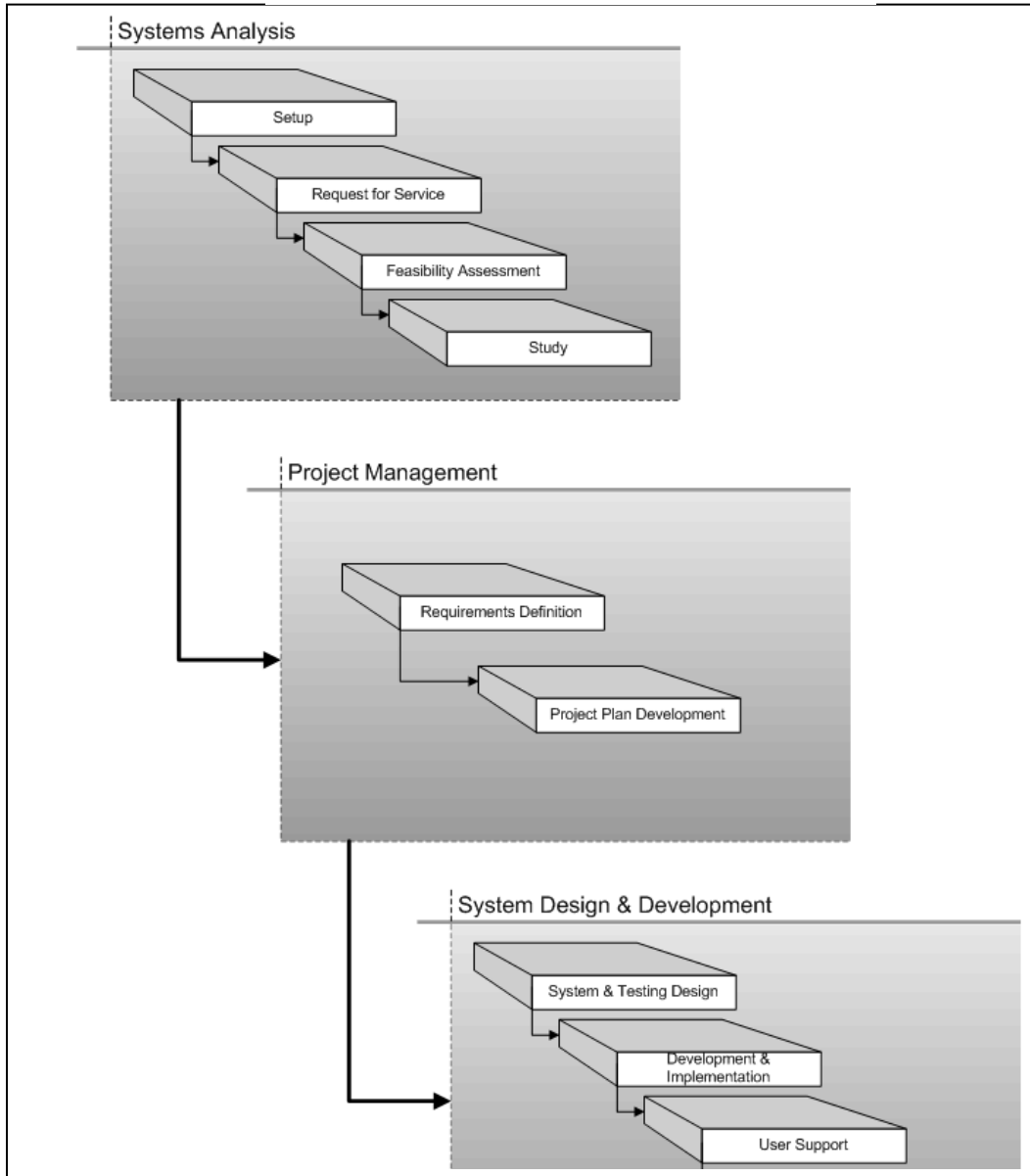
The second major change in the curriculum was the incorporation of the systems analysis course into the sequence. At this point in the evolution, templates for a feasibility report and a systems analysis report were added to the methodology in order to extend the span of student involvement in the planning of the projects throughout the entire system development lifecycle. These templates were used to guide and motivate students in their role of communicating with the sponsors and users; they were used to impose a higher level of consistency on the entire process running from project initiation to final project acceptance; they served as outlines for the feasibility and analysis reports that were the major portion of student grades; and they insured that policies and controls were in place.

The linking of the project management and development courses extended the span of the development process to two consecutive semesters. The use of projects as linkages between these courses was obvious and tangible to the students: a project plan was developed, and then the plan was executed. However, when the systems analysis course was added, the process was extended to three courses which span 16-21 months. Students in the systems analysis course were just beginning their MIS courses; they were not yet able to visualize the entire development process: its purpose, benefits, or outcomes. They had difficulty relating and integrating what they learned in the systems analysis class to the project management and development courses. As a result, further efforts were needed to help the students understand the project development life cycle and how the three semester sequence of courses was related to it.

As the three courses were integrated, the need for better coordination between them increased. In addition to plans being required for each project, some framework for the teaching, preparing, transition between courses, and various levels of communication needed to be systematized. Originally, the requirements for each course were specified, but were couched within the traditional framework of systems analysis, project management and system design and development courses. Gaps in coverage occurred between classes. Students understood that there was a sequence, but were unable to understand the relationships between the content in the three courses in relation to the sequence. In addition, the instructors were constrained by the material in specific textbooks; that is, the courses were consistent with the text, but were not readily transferable to the sequence. The solution to these problems was the

development of a methodology that could be used to guide and organize the entire experience of students across the courses; one that was independent of particular textbooks and cases. What was needed was an overriding framework under which each course had specific objectives to accomplish, and successful completion of those objectives would lead to successful project completion. An overview of the deliverables in this methodology is presented in Figure 2.

Figure 2: Deliverables



With this methodology, not only were the projects being planned and managed within a known framework, but also, the knowledge, skills and abilities of the students were being delivered in a consistent manner, well-understood by both the students going through the sequence and by each of the instructors participating in the courses. Furthermore, with the distribution of this methodology available to all faculty in the department, instructors who taught courses that fed into the sequence but were not part of it, can better understand how their courses fit into

an integrated learning process. Figure 1 above illustrates the relationship between other department courses and the sequence. With this methodology defined, each course could be viewed by both students and faculty as a part of a coherent process directed towards student success. The development process could be integrated across the entire development lifecycle, department-wide understanding of the process and the roles of instructors in the process became apparent, and all needed performance criteria were in place.

Subsequent to the development of the methodology, the next step was to revise the entire curriculum around that methodology. This revision focused on changing the sequencing of courses in the program, not on changing their content. For example, before the revision, the managerial presentations course and the general MIS course were allowed to be taken anytime in the junior or senior year. A majority of MIS students put off taking these courses until after they had completed their required major courses and their upper-division electives. However, both of these courses were seen as contributing to the success of the projects only if they were required to be taken prior to the start of the 3-semester sequence of courses. The managerial presentations course focused on developing professional level oral and written communications. These were skills that contributed to success in the writing of the systems analysis reports, the final project reports, and the final project presentation. The general MIS course introduced students to the role of information systems in organizations and included a section on emerging technologies. This course was important for success in the systems analysis course, and the emerging technology section of this course was particularly important for system design. Therefore, through the use of pre- and co-requisites, the Managerial Presentations course and the general MIS course were moved to the beginning of the curriculum in support of project improvement and students success.

As another example, the success of any project depended on the skills brought to that project by its team members. Before the curriculum revision, students often postponed taking the more challenging upper-division electives until they had completed their required courses, including the project management and design and development courses. This meant that some team members were not bringing sufficient skills to their projects, and from a learning standpoint, were not applying skills they had learned in a meaningful way. Therefore, a prerequisite that required students to have completed at least one of the upper-division electives courses was added as to the development course. This requirement forced each student to bring some high-level skill to his or her project so that teams can then expect to have three members, each of whom brings some advanced application expertise in databases, web development, networking, or multimedia.

Grading Metrics Developed

In the next stage of improvement, we developed metrics for project deliverables that served as lower control limits for evaluation of acceptable performance. Projects were evaluated with respect to these metrics and attempts made to reduce the occurrence of results outside these limits. These metrics were presented to students early in the Project Management course so that they could be aware of the expectations of the evaluators early in the process. The final report prepared by students was the primary source of information concerning success. Students must provide proof of successful completion of the original system requirements (or those as modified through change requests) , quality assurance testing, user satisfaction, training and/or help documentation, the use of systems analysis, including modeling, the application of project management, independent research, and integration of technologies. At the end of the semester for the design and development course, each project was evaluated against these criteria by a panel composed of all department faculty, by alumni of the program, and by IT professionals from the community. Students presented their project and their written report to the evaluation panel during a one-hour review held in a computer lab. Fifteen minutes are provided for questions. Most reviewers tended to focus on reviewing the details in the areas of the project they are most expert in, for example, database design, web development, systems analysis, project management, or managerial presentations. Other reviews focus on testing the system during the presentation.

Included in these standards were the minimum requirements for a project to be considered a success, that is, whether or not it would receive a passing grade. These minimums reflected both the needs of the sponsoring organizations (including satisfying system requirements, quality assurance, and timely delivery) and the learning needs defined by the department (including independent research, integration of courses, and professional quality presentations). Throughout the Systems Analysis and Project Management courses, students were provided with

feedback on the likelihood of their project achieving these standards. Projects plans were evaluated at the end of the Project Management course, and projects were selected for implementation based on their likelihood of success in terms of meeting the minimum standards and on having the potential for being an “A” level project.

Subjective evaluations of quality were also used. These estimates were weighted on a scale determined by the instructor, then normalized and totaled. Upon completion of all presentations, faculty met to reach a consensus on the final project grades. The projects accounted for 70% of each student’s grade, and each student on a project team receives the same score. The remaining 30% was based on the individual performance of the students as determined by the project manager, team members, and the instructor.

Formally, the final project report includes a section on lessons learned and a post implementation evaluation. These sections were used by faculty and students to learn from the successes and failures of past projects. Informally, students who have completed their projects often passed on tips to new students while faculty communicated among themselves and with project sponsors about strengths and weaknesses of the three-semester development process. Courses, project selections, sponsor selection, and the development process were revised based on these formal and informal reviews. Faculty were also willing to meet with teams after the final grades had been decided to provide feedback and suggestions to project teams.

LESSONS LEARNED AND CONCLUSION

This paper has described the redesign of a traditional IT-oriented curriculum into a community-based project-centric curriculum. The paper concludes with the lessons learned and potential future research.

Benefits And Risks

The benefits associated with a curriculum built around substantial community-based projects are significant to the students, the university and the community. However, the risks associated with planning and coordinating these projects cannot be ignored.

The benefits to the students are best understood through the context of the learning theories discussed above. The ideas of engaged, team-based, community-based, and project-based learning have been integrated to garner the benefits of each. By directly interacting with sponsoring organizations, the university benefits directly through cultivation of its engagement with the community, through its interaction with the sponsoring organizations and its graduates, and through recognition of its commitment to the community. The community benefits from the systems developed for the sponsoring organizations, from the acquisition of well-trained employees who become positive contributors, and from exposure to a variety of system development concepts and methods used by the student project teams.

The risks involved in the implementation of this curriculum cannot be ignored. Conducting large projects across three semesters, then integrating these projects across all courses within a curriculum, introduces planning and coordination problems not experienced in more typical curriculum designs. The increased complexity of interrelationships between courses, instructors and students requires new approaches for managing them.

Problems Encountered

Lack Of Potential Qualified Project

In the first few years of implementing this curriculum design, there were problems finding suitable projects. The broader the set of potential projects, the more likely that those projects selected for development will be successful for both the students and the project sponsors. In the initial years, it is likely that there will be a need to proactively seek out projects. However, over time, as recognition of the benefits of the projects grows throughout a community, and as former sponsors return for new projects, the number of qualifying projects are likely to exceed the number of teams available to develop them.

Incompatibility Between Teammates

In our particular academic environment, large differences between the motivation levels of teammates caused more problems than do differences in skill levels or personality conflicts. The latter two problems can be dealt with adequately through good project management, and can result in a positive learning experience for all members of a team. However, different levels of motivation are more difficult to handle because of limited flexibility of actions and other restrictions. A good approach for avoiding this problem is to reduce the likelihood of it occurring by promoting discussions among the students concerning their individual expectations of levels of contribution. These discussions should be held prior to the formulation of teams. One alternative is to then form teams with students sharing similar expectations. This is likely to result in a few lower quality project results, which might be unavoidable within the academic environment. Another alternative is to develop grading schemes that reward students based on their contribution to the projects success. We are still working towards find a satisfactory resolution for this problem.

Unresponsive, Unsupportive, Or Unknowledgeable Sponsors/Users

From the perspective of students, problems with sponsor and user participation have been one of the most common types of problems encountered. These projects are usually the students' first experience with the real-world problems of obtaining feedback over a sustained period. Without timely and informative feedback from sponsors and users, students have difficulty completing their projects within their inflexible deadlines. Three approaches might be used to minimize this problem: 1) screen projects based on indications of sponsor support, 2) initiate faculty intervention with the sponsor to encourage responsiveness, 3) remove the sponsor and users from the project by moving to a simulated projects environment.

From the perspective of learning, lack of sponsor and user participation is actually one of the benefits of having students undertake real-world problems. It corresponds to the same situation faced by many developers. It requires students to dedicate more time towards understanding the business problems and communicating with the users to determine beneficial system design, thereby increasing their systems analysis experience and problem solving capabilities.

In projects where the sponsors/users are not familiar with IT in general, and are unable to define their requirements or provide useful feedback in particular, students may need to assume the role of both analyst/developer and user. Although this results in some difficulties for the students, the success of the projects will not be jeopardized, and the resulting experience will contribute an extra dimension of learning.

A related concern is that of sponsors having low expectations of the student teams. In our experience, for most projects completed for first-time project sponsors, the sponsors have frequently underestimated the capability of the students. However, since the results have far exceeded their expectations, the sponsors have expressed their surprise and praised the student team accomplishments. Low expectations should only become a problem if projects with very narrow scope are accepted for development and hence project teams are not sufficiently challenging. Careful screening of the potential projects should avoid this problem.

Lessons Learned

In addition to the issues related to the curriculum redesign, course delivery and course management issues discussed above, two additional issues that have been learned from this experience have been identified.

Student Advising Issue

In courses having projects that span more than one semester, the relationships between courses, the expectations (i.e. project deliverables) of each course, and the rationale for requiring a sequence, should be introduced early in the first course of the sequence. This helps students recognize the importance and implications of their current tasks within the larger framework of their projects and within the system development life cycle. It also helps them to

design their course of study with an understanding of the interrelationships between courses and the contributions each course makes on their project.

Accepting Challenging Projects

One of the objectives of engaged learning, project-based learning and team-based learning is to stimulate students' desire to learn by providing them opportunities for collaboration and engagement on substantial community projects. We have noticed in our experience that student abilities and motivations have often been underestimated by sponsoring organizations as described above and, in some cases, by instructors. However, we have found that students are willing to take on challenging projects such as those that are proof-of-concept, research-oriented, or outside of their comfort zone, as long as those projects either include the use of emerging techniques or have significant public recognition. It is important on these types of projects, however, that their evaluations be weighted more heavily on the students' learning and research contributions, and less heavily on the functionality delivered, than is done with other types of projects.

In our experience, these challenging projects have often resulted in students achieving the greatest level of success from an educational perspective, and in students developing the most self-confidence, so that upon completion of their projects, they have felt that they would be able to handle any challenge they may encounter in their future careers.

Conclusion And Future Research

We have found that this redesigned curriculum has been successful in developing high achieving, confident students who are ready to immediately contribute in their career fields. It has also resulted in recognition by both the local and national communities for outstanding service by the department, contributing, in part, to our university's recognition in The Princeton Review as one of "81 Great Schools with Outstanding Community Involvement," (Princeton Review, 2005).

Currently, with the trend towards developing course assessment for measuring student and program success, some standard framework for determining the current quality level of instructional processes, and for guiding improvement in those processes, could be useful within the context of continuous improvement. Future research might be conducted based on ideas presented in this paper to explore the development of a framework which would be specifically designed for assessing and improving processes in educational programs. In our program, we have just become to use the quality of the students' performance on their projects as one of the assessment measures of our program. Some framework that indicates the level of maturity of an educational process in terms of the consistency of student success over time, and a prescription for moving towards higher levels might be an extension of this.

REFERENCES

1. Association of Information Technology Professionals. (1997). Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. Retrieved August 19, 2005 from http://www.cis.usouthal.edu/faculty/feinstein/IS97/document/is97_title.htm.
2. Bandura, A. (1977). *General Theory: Social Learning Theory*, Englewood Cliffs, NJ: Prentice-Hall.
3. Barrows, H. & Tamblyn, R. (1980). *Problem based learning: An approach to medical education*. New York, NY: Springer.
4. Dewey, J. (1938). *Experience and Education*, Kappa Delta Pi.
5. Iacono, S. & Kling, R. (1988). Computer systems as institutions: Social dimensions of computing in organizations. In J.I. Degross & M.H. Olson (Eds.), *Proceedings of the Ninth International Conference on Information Systems (ICIS)*, Minneapolis, MN.
6. Jacoby, B. & Associates (1996). *Service-Learning in Higher Education*, San Francisco, CA: Jossey-Bass.
7. Jonassen, D.H. (1995). Supporting communities of learners with technology: A vision for integrating technology with learning in schools. *Educational Technology*, 35(2), 60-63.

8. Kearsley G. & Shneiderman B. (1999). Engagement Theory: A Framework for Technology-based Teaching and Learning. Retrieved from August 19, 2005 from <http://home.sprynet.com/~gkearsley/engage.htm>.
9. Keeton, M. & Tate, P. (1978). *Learning by Experience – What, Why, How*. San Francisco, CA: Jossey-Bass.
10. Lave, J. & Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*, New York, NY: Cambridge University Press.
11. Lazar, J. & Preece, J. (1999). Implementing service learning in an online community's course. Proceedings of the International Academy for Information Management Conference, 22-27.
12. LeJeune, N. & Richardson, K. (1998). Learning Theories Applied to Web-Based Instruction. University of Colorado at Denver.
13. Lewin, K. (1951) *Field Theory in Social Sciences*. New York, NY: Harper & Row.
14. Piaget, J. (1970) *Genetic Epistemology*. New York, NY: Columbia University Press.
15. *The Princeton Review Colleges with a Conscience* (2005). New York, NY: Random House. Schrage, M. (1991). *Shared Minds: The New Technologies of Collaboration*. New York, NY: Random House.
16. Shaw, A. & Shaw, M. (1999). Social Empowerment through Community Networks. In D.A. Schon, B. Sanyal, & W.J. Mitchell (Eds.), *High Technology in Low-Income Communities: Prospects for the Positive Use of Advanced Information Technology*. Cambridge, MA: The MIT Press.
17. Shneiderman, B. (1998). Relate-Create-Donate: A teaching/learning philosophy for the cyber-generation. *Computers & Education* 31 (1) 25-39.
18. Venkatesh, M & Small, R. (2002). Active Learning in Higher Education: A Model and Roadmap. In J. Lazar (Ed.), *Managing IT/Community Partnerships in the 21st Century*, Hershey, PA: Idea Group Publishing.
19. Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.