Journal of Information Technology Education Volume 4, 2005

Enhancing Student Learning across Disciplines: A Case Example using a Systems Analysis and Design Course for MIS and ACS Majors

Ken Surendran
Southeast Missouri State University, Cape Girardeau, MO, USA

Ike C. Ehie
Kansas State University Manhattan, KS, USA

Chellappan Somarajan
Southeast Missouri State University, Cape Girardeau, MO, USA

ksurendran@semo.edu iehie@ksu.edu csomarajan@semo.edu

Executive Summary

This paper illustrates an approach used to enhance student learning outcomes in a combined cross-listed Systems Analysis and Design (SA&D) course and examines benefits perceived by students through analysis of assessment and students feedback. The SA&D course is a required course in both the Management Information Systems (MIS) major and the Applied Computer Science (ACS) major. The SA&D course was taught to a combined cross-listed class of MIS and ACS students over a period of two years. Two strategies were adopted to make the course a worthwhile learning experience for students in both majors. The first was to extend the scope of the course within the System Development Life Cycle spectrum to include planning before analysis and implementation (prototype) after the design. The second strategy was to have a running group project as the main assessment (accounting for 50% of the course grade) where each group had at least one student from each of the two majors. These groups carried out a system development project with four phased deliverables: system proposal, requirements specifications, design specifications and a working prototype with emphasis on user interfaces. This paper provides a comprehensive overview of how the combined cross-listed course was designed, delivered and refined for future offerings. It also examines the value of teamwork using students' feedback.

The students' experiences were studied over a two-year period. Two different instruments were used to gather feedback and to analyze the effectiveness of the combined cross-listed strategy: a qualitative study and a quantitative study concerning student perception on learning enhancement. Initially a qualitative study with open-ended questions was carried out to identify areas for improvement and to examine how well these strategies had worked. The three problems identified were lack of sufficient time for the last phase (working prototype), lack of time for team meetings, and lack of a comprehensive example case. These problems were addressed in subsequent course offerings. The study also revealed that about 80 percent of the students liked working in a mixed group setting on the extended course project, and 75 percent of the students indicated that working on the mixed group project offered them real-world experience. Encouraged by such positive observa-

Editor: Ramesh Sharma

tions, a quantitative study was conducted on students' perceptions concerning specific learning outcomes for carrying out the various system development tasks and the development of skills (including soft skills) among and between the two majors. The results indicated that the students from both majors perceived more than average learning outcomes and skills development. It also indicated that while the ACS students claimed to have learned relatively more on feasibility analysis and information gathering, the MIS students claimed to have learned relatively more on user interface design and architectural design. However, the results indicated that the perceived differences in the learning outcomes between the two majors were not significant. The analysis confirms an enhanced learning outcome for both the ACS and the MIS majors due to knowledge sharing made available through teamwork.

Keywords: systems analysis and design, teaching combined majors, group project / teamwork, perceived learning outcomes, soft-skills development.

Background Information

Systems analysis and design is a course that focuses on the development and maintenance of new and existing systems in an enterprise (Misic & Russo, 1999). This course is usually taught in an MIS program in a business school or in a computer science program in the liberal arts or engineering school. The MIS curriculum includes a course in Systems Analysis and Design (SA&D) followed by a project course (capstone course) involving system development and implementation. The course sequence is in accordance with the 1997 and 2002 Model IS Curriculum Guidelines (Davis, Gorgone, Couger, Feinstein, & Longenecker, 1997; Gorgone, et al., 2003). The SA&D course focuses on the earlier phases of the System Development Life Cycle (SDLC), and the course is typically delivered using either or both the procedure-centric and object-oriented paradigms. In the Applied Computer Science (ACS) Curriculum, a two-part Software Engineering (SE) course is usually offered, which consists of analysis and design in the first semester and a system development project (a capstone course) in the second semester.

The College of Business established the MIS program that is discussed in this study in 1999 following an extensive review by both the faculty and IS practitioners. As part of the development of the MIS program, a focus group comprising of MIS faculty and IS practitioners was conducted to review a proposed curriculum for MIS. The idea to leverage the expertise of the well established computer science program to support the newly developed MIS program was echoed by the IS practitioners during the focused group discussion (Ehie, 2002). This viewpoint was made known to the provost of the institution by a consultant hired to review the new MIS curriculum. Based on the review, a new joint faculty position was approved for the MIS and the Computer Science departments. One-quarter of the new position was devoted to the MIS program and the new faculty member would be resident in the computer science department. The joint appointment will allow two programming courses to be taught by the ACS faculty for the MIS program. The department heads from the ACS and MIS programs, upon consultation with their respective faculty members and students, decided to offer a cross-listed course in SA&D in which both MIS and ACS students will be combined in one class. Such an unusual collaboration between two competing departments offers both challenges and opportunities. The main challenges were to ensure that the course design, bearing in mind the respective prerequisites, meets the curricula requirements to prepare both majors adequately for their subsequent system development and implementation courses and to diffuse the possible cultural tension between the two differently focused majors (Business and IS) during the class sessions. The main opportunity lies in constructively using the diversity to simulate the real-world situation, where students with different academic background work together to achieve the common course objectives. The SA&D course under the MIS program is followed by a capstone project course called *Systems Implementation and Practice*, which essentially caters to the curriculum requirements specified for implementing databases and distributed applications. The Applied Computer Science majors specializing in information systems in the University have a core course titled *Information Systems Analysis and Design* (ISA&D) in addition to *Software Engineering*, a project-based capstone course required by all ACS majors. Just as in SA&D, primary workflows in software development form the core of ISA&D. A second-level Information System course emphasizing the use of databases serves as the prerequisite for the ISA&D course, along with an in-depth four-semester programming knowledge.

The ACS department, with input from the MIS department, hired a new faculty member that had an extensive industry experience. The new faculty member was charged with developing the combined cross-listed course. The success of the combined cross-listed course relied in most part on the instructor, whose extensive industry experience coupled with his technical experience made it possible for him to relate well to both the MIS and ACS students. The MIS students brought to the stage their knowledge of business processes, whereas the ACS students brought their extensive technical and programming skills. To offer the combined cross-listed course, the following issues were at stake:

1) Feasibility of having a combined cross-listed SA&D course for MIS students with their business process experience and ACS students with their knowledge of programming skills.
2) Extent to which student learning will be facilitated through the combined cross-listed SA&D course.
3) Extent to which the combined cross-listed course adequately prepares students for real-life experience
4) Degree of student satisfaction in taking the combined cross-listed course with students from a difference background.

This paper, in addressing the above issues, attempts to offer methods that have been found to be effective in interdisciplinary teaching in two departments that have been protective of their respective turfs. The interdisciplinary teaching approach espoused in this paper follows the trend in industry where professionals from different backgrounds are required to work together to accomplish a common business objective. In this paper, the authors examine, qualitatively and quantitatively, the merits of teaching the combined cross-listed SA&D course to both MIS and the Applied Computer Science (ACS) majors using cross-disciplinary teamwork for enhancing the simulated real-world experience.

The remainder of this paper is organized as follows. In the second section, we present literature review of past studies. The third section contains the details of the combined course, elaborating on customization of the course to meet the two curricula requirements. The fourth section deals with the feedback from students for determining the effectiveness of this collaborative project approach. The results of both qualitative and quantitative studies concerning students' perception on the extent to which the integrated teamwork helped in enhancing student learning are presented. In the fifth and final section, the authors relate their findings with some of the reported results in the literature and discuss a few tactics useful for further enhancing learning outcomes in such a combined offering of SA&D course to ACS and MIS majors.

Literature Review

Information systems (IS) education in general has been criticized for not producing graduates with the right set of skills, knowledge and attitude to meet the needs of the global and technological evolving workforce (Archer, 1983, Cardinali, 1988). A large part of the problem stems from the inability of universities to produce graduates who are equipped with both human relations and discipline-specific skills. In a recent survey of recruiters of business graduates by Wall Street Journal/Harris Interactive Survey ("Ranking the Attributes ," 2004) 89 percent of recruiters indicated that communication and interpersonal skills are very important for hiring new graduates and 87 percent of those surveys claimed that the ability to work well within a team was very crucial in their hiring process. Content of the core curriculum and faculty expertise were mentioned by only 34 percent and 25 percent respectively by recruiters. In a similar survey of IS professionals on the type of communication skills needed by IS staffers, coherent writing, ability to ask appropriate questions, effective oral communication skills, ability to accomplish assignment and the ability to work cooperatively in a one-on-one and project team environment were found to be essential communication skills to have in present day IS professionals (Miller & Luse, 2004). The university systems continue to operate in "stove pipes," whereas the business practices have moved to a more integrative framework. Universities need to place more emphasis on the integration of technology, applications, data and business functions and less on formal and traditional systems development approach (Trauth, Douglas, & Lee, 1993). Recent studies have found that IS professions are being pulled in opposite directions; one toward the more technical skills required to maintain firm's infrastructure and the other toward business and interpersonal skills. To resolve this impasse, the IS profession should focus on developing graduates with a solid understanding of the business areas and with a broad range of possible technological solutions (Trauth, et al., 1993). In fact, the IS profession has reiterated that it needs graduates that are educated not just in the technical skills but in interpersonal, communication and team-oriented skills. The need to strike the right balance between technical and business knowledge has become a key concern for most IS curricular designers (Ehie, 2002, Cardinali, 1988).). Avison (1991) extols the virtues of pragmatic or action-oriented teaching that would help both students and researchers gain an understanding of the problems when developing information systems for the real-world.

One of the objectives of courses like SA&D and Capstone (where students design and prototype a product) is to prepare the students for real-world challenges in system development. The use of role-play has emerged as an effective teaching pedagogy in information systems courses. Sullivan (1993) discusses how role-play in team projects could be used for enhancing communication skills in a software engineering course. Use of case studies that offer role-play opportunities have been successfully used in teaching SA&D concepts to first year undergraduate students (Cope & Horan, 1996). In a more recent study, Chen, Muthitacharoem, and Frolick (2003) show the effectiveness of using role-play case studies in a SA&D course for improving communication skills. Team projects have the potential for enhancing both the technical and soft skills since they require the students to take on different roles in applying a methodology to create a product. Studies reported the use of client-sponsored projects for simulating real-world experience in capstone courses (Fox, 2002; Lynn, et al., 2002; Polack-Wahl, 1999,). While use of client-sponsored projects may be appropriate for capstone courses, instructor-specified projects offer a safe environment for students in the SA&D course (Surendran & Young 2001), which is usually a prerequisite for the capstone course. Little and Margetson (1989) argued for the integration of technical and organizational (business) skills through the use of project-based approach in information systems design courses.

While many of the reported studies concerning combined classes of graduate and undergraduate students seem to have focused on students' learning and performance in general terms, Etzkorn, Weisskop, and Gholston (2004) look specifically at the relative performance between graduate and undergraduate students. According to this study, the undergraduate students, in view of these courses' strong prerequisite, performed as well as the graduate students. However, they did not use strategies, such as mixed (undergraduate and graduate) team projects, for balanced learning. Although combined course offerings and use of team projects have been studied independently, there are very few studies that consider both issues concurrently. Van Der Vyver and Lane (2003) discussed the value of team-based project activities in a database course with students from Computer Software Development (CSD) and Information Technology Management (ITM) majors. Specifically, they assigned development projects to teams of four - with a competent lead programmer in each team - to improve the performance of the ITM students, who are perceived not to be as competent in coding. This study is somewhat similar to what is being presented in this paper in that both deal with mixed group of majors and use strategies to enhance overall learning through group work. The major differences deal with the intensity of teamwork, the breadth of skills development through group work, and the variety of analysis used.

Phenomenographic research has been used to investigate student perception of learning associated with role play (Cope & Horan, 1996). The methodology is highly desirable in a learning environment in which students attempt to understand the material being presented. Research based on phenomenographic approach has been conducted in a number of studies (Bruce, et al., 2004; Marton & Saljo, 1984). The most important element of this kind of research is that data is collected directly from the learners themselves through self-reports and interviews. The methodology contributes to the understanding of students' learning from the viewpoint of both the students and the lecturers from a range of disciplines (Bruce & Gerber, 1994). The phenomenographic research is based on the principle of seeking understanding of the phenomenon of learning through examining the students' experiences. This is conducted in a naturalistic setting involving the actual content and settings in which students learn (Entwistle & Marton, 1994). Concurring with phenomenographic research, Trigwell and Prosser (1991) identified relationships between perception/evaluations of the learning environment and the quality of learning outcomes. The current study relies on students' perception and feedback in assessing student learning outcomes. This study adopts a phenomenographic technique to investigate students' perception of the learning outcomes in a combined cross-listed course that involved role-playing.

Combined Courses and Team Projects

The SA&D course emphasizes such topics as systems development life cycle, communication with both users and developers, and a variety of standard tools, techniques and heuristics relevant in preparing requirements and design specifications. The knowledge elements of the course reflect the principles and techniques used in the analysis and design aspects of software development. Specifically, in this course the students apply the techniques and tools of the procedure-centric structured methodology for producing the intermediary system artifacts (from inception to design) of software development.

Course Customisation

The SA&D course normally is concerned with requirements specification and logical and physical design. However, by stretching the course on either side, both the *business* focus and the *IS solution* focus required for the two majors are presented in the course (see Figure 1). The business focus includes feasibility analysis, evaluation of business processes (simple mechanization, improvement or re-engineering) and preparing the organization for realizing the value of the new system.

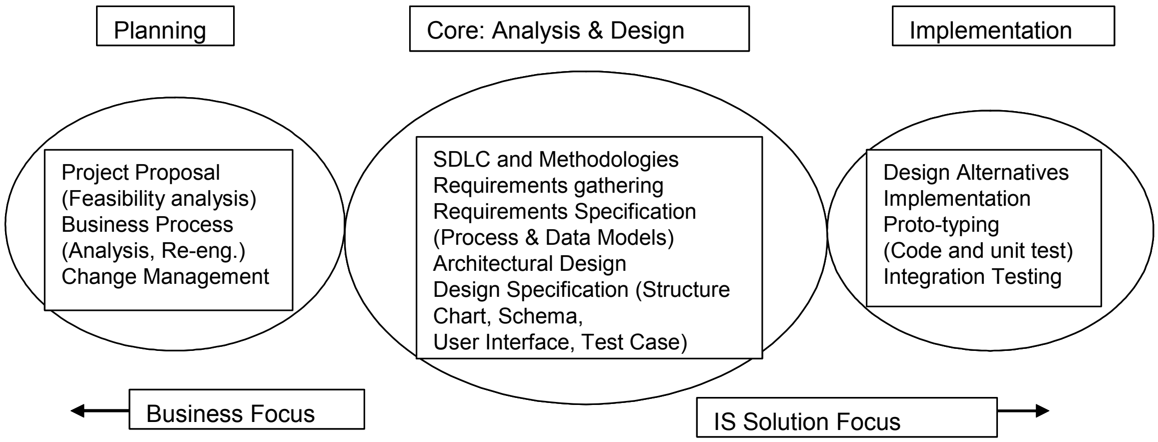

Figure 1. Business-cum-IT Solution focused SA&D Course Model

These pre-analysis and evaluation activities are of greater value to the MIS majors. The IS solution, on the other hand, focuses on evaluating design alternatives, implementing (prototyping) a system design and planning for a quality product. These post-analysis activities are of greater value to the ACS majors.

By including the planning and implementation phases in the project, the two majors are able to appreciate the value their counterparts provide to the overall development of an IS-based business solution. Thus, this 16-week course addresses four main areas:

1. System development project planning (with focus on preparing a proposal and feasibility analysis)
2. Requirements specification (with focus on information gathering and building process & entity models)
3. Design specification (with focus on system architecture, Input/Output design, database schema and structure chart)
4. Implementation (with focus on creating user interfaces and prototypes, documentation – user and system)

The business value of an information system is emphasized in the planning phase. Techniques for information gathering and models like Data Flow Diagrams (DFD) and Entity Relationship (ER) diagrams are introduced in the analysis phase. The distinction between architectural and detailed design is then emphasized. During the architectural design, network models are discussed, and alternative development plans are examined. In the detailed design, schema for the optimized database, structure chart (guided by the concepts of cohesion and coupling) for the modules, test cases and user interface designs are covered. The implementation phase is concerned with test plans, creation of user interfaces and installation plans.

Strategy for Soft-skills (Value Skills) Development

The learning strategy used in the course aims at developing student communication skills and ability to work in multi-disciplinary teams. A group project consisting of four phases was chosen to serve both the objectives. The deliverables involve considerable writing and a managerial presentation. This project assessment accounts for 50% toward the final grade, while several individual assessments (quizzes, exams) account for the balance. The team size for the project is limited to minimum of 4 and a maximum of 5 to match the workload of the assignment (each student is expected to put in about 50 hours of effort for all the four phases). Team selection is left to the students; however, each team is required to have at least one ACS and one MIS major. To become acquainted with team members within a group, mini cases from the textbook *Systems Analysis and Design: An Applied Approach* (Dennis & Wixom, 2003) were assigned at the initial stage of the course to facilitate the process of team-building.

The project with four phases provides an opportunity for students to take up several roles in the system development process. Such role-plays help hone the soft-skills (Chen, et al., 2003; Little & Margetson, 1989; Surendran & Young, 2001). For instance, under each phase, one student acts as a team coordinator directing the activities in a team for that phase. This role requires allocating workload (delegating), negotiating and communicating with team members, organizing meetings, and conducting reviews. Even the other roles during the design phase require coordination and consultation so that the product developed is both coherent and cohesive. The coordinator gets sufficient coaching from the instructor to handle problems in dealing with the team members. Further, the role rotations help the composite teams function synergistically, pooling their skills in realizing the project objectives. For instance, in the implementation phase, which is more demanding, some teams find it very difficult to complete the tasks without proper workload sharing. This provides an opportunity to apply their skills in negotiation, delegation, and time management. In addition, it offers an opportunity to work towards an objective and maintain self-discipline to meet the project deliverables.

Project Deliverables

Using team projects is common in software engineering courses (Surendran, Hays & Macfarlane, 2002) since this offers considerable opportunities for simulating real-world experience. In the SA&D course all teams worked on a single project with the instructor as the client in order to provide a safe environment. During class periods, time was allocated for the teams to interview the client. Also, e-mail was used extensively for gathering information. Each team had a team leader for each of the four phases and the members met outside the class to assess and allocate work.

The four deliverables were: Project Proposal (10%), Requirements Specification (15%), Design Specification (15%), and Prototype for selected subsystems (10%). In the last phase the students submit a project report (which compiles all the earlier documentation, user notes and their reflection) and present their prototype to the class. In view of the greater level of effort required for analysis and design, higher weightings were given to requirements specifications and design specifications. A detailed list of contents for the deliverables in each phase is provided in Table 1. The selection of items was based on the instructor's extensive industry experience which was adapted to the class project. In phase I, tasks such as *system objectives* and *economic feasibility* require business knowledge whereas, tasks such as *solution description* and *technical feasibility* require technical knowledge.

Table-1: Details of Project Deliverables for Four Phases

I. Proposal:
Executive Summary (10 or less lines describing the context, content summary, highlights)
Current Situation (background, business needs; constraints)
System Objectives (functionalities)
Solution Description (approaches and alternatives)
Resources needed (people, training, equipment)
Cost estimates, anticipated benefits leading to economical feasibility
Technical and Operational (organizational) feasibilities
Schedule (major activities and timelines; use MS Project to draw a Gantt chart)

Conclusion (risks, if any; additional notes) Appendix (if any, providing additional supporting details)
II. Requirements Specification:
Executive Summary (summary of report content and highlights) Revised Cost (no need to examine the benefits) Revised Schedule (for design, implementation – in some detail) Process Description of the System (using context and DFDs) Data Description of the System (using ERDs) Conclusion (any identified risks, your plans on further work) Appendices (needed only if you have any additional pertinent material)
III. Design Speciation:
Executive Summary (design report summary) Revisions to Schedule (refine schedule for the remaining activities) General System Design (system architecture, network model and Physical DFDs and ERDs (for the four phases) User Interface Designs (for both input and output) Process Design (structure chart; apply design concepts) Database Design (3rd normal form and optimization -using ERDs) Test Plans (integration and system testing along with test cases)
IV. Prototype for Selected subsystems:
Executive Summary (what is being delivered; limitations if any) Working Code / System (source on a CD / floppy – do not print program listing) Quality Assurance: Test cases, Test Data and Test Results User Manual Systems Manual (analysis and design artifacts - compilation) Reflections (learning – both technical and development of *soft-skills*)

Since the teams collectively had members with both business and technical background, they were able to learn from each other and work collaboratively to complete the tasks. The extensive programming skills that the ACS students brought to the team helped considerably in process modelling and design, while the greater knowledge of business database management that the MIS students brought to the team helped significantly in the data modelling and design. Opportunities for such knowledge sharing were made available throughout the project, and more significantly in the first and the last phases of the project.

Student Feedback and Analysis

Students' feedback was assessed using both a qualitative and a quantitative approach. The qualitative approach was used to gauge students' initial reactions to the combined cross-listed SA&D course. These reactions and feedback were used to re-evaluate and revise the course. The quantitative approach was conducted to assess student learning outcomes and student satisfaction level of the course. These pre and post assessments allowed the instructor to obtain feedback from students that was later incorporated into the course to enhance overall student learning outcomes.

Preliminary Questionnaire

This cross-listed combined SA&D course was offered for the first time in fall 2001 semester. At the end of the semester, a simple questionnaire with 12 open-ended questions was used as the main instrument for gathering feedback from students. The questionnaire addressed the following: project phasing, team composition, role playing, workload balance, skills balance, functioning of the team, real-world experience simulation, and overall experience. The purpose of this initial student feedback was to gather information on how well the course was received and to identifying areas for improvement. The preliminary questionnaire was given to the class of 40 MIS/ACS students enrolled in the combined SA&D course. The students were told that completion of the questionnaire was optional and 20 students (10 ACS and 10 MIS majors) responded. The analysis indicated that more than 80% liked the idea of working in the multidisciplinary teams and felt that the various backgrounds complemented each other. Most of the students were satisfied with the idea of phasing with the exception of a few that complained about lack of time for the final phase. About 75% indicated that the project offered them real-world experience based on the fact that they worked on a single project with phased delivery, worked with people from a different academic background and levels of commitment, experienced the stresses and strains of working in a multifunctional group, delivered something tangible at the end of the project, and dealt with actual client's requirements. During the project, every one had taken on at least three different roles from among the following: business analyst, systems analyst, process designer, database designer, interface designer, programmer, and team coordinator. Some of the problems identified in the questionnaire included lack of time for the last phase, lack of good comprehensive examples for the deliverables and schedule conflicts for team meetings outside class.

In addition to the initial qualitative assessment, the department heads of both the ACS and MIS took a random sample of students in the combined cross-listed course to gauge their experience with the course. About 90% of the MIS students who took the combined cross-listed course commented positively on their experience working with the ACS majors in their teams. The following three sample individual comments summarize the views held by most of the MIS students:

> (1) *This was a good project that helped me feel more assured about my degree choice; I have been saying for the past two semesters that I am a MIS major and I really didn't know what it meant until I took the combined SA&D course. I know now that I want to be a business analyst.*
> (2) *The project was a way to learn how to work as a team and break down a project between members so that each member was able to do tasks that are best suited to him / her.*
> (3) *We learned teamwork, time management, how to compromise, and hopefully taught each other things along the way.*

Encouraged by such comments, the combined SA&D course was offered in spring 2002 and fall 2002 semesters with some schedule revisions to allow more time for the last phase of the project. At the end of fall 2002 semester, a quantitative study was conducted to find out the learning outcomes for both the MIS and the ACS students. The analysis on student learning outcomes draws from the work of Entwistle and Marton (1994) on phenomenographic research. This approach accesses student learning based on students' perception of learning.

Students' Perceptions on Learning Outcomes

Encouraged by the positive feedback of the initial survey, a quantitative study was carried out during subsequent course offerings. The questionnaire (shown in the Appendix) addressed learning outcomes in terms of tasks and skills relevant to systems analysis and design due to the combined ACS and MIS majors. These tasks and skills are the standard ones and are derived from the deliverables. The lists of tasks and skills used in the questionnaire are comprehensive, reflect both industry and academic perceptions, and match those discussed in literature (Misic & Russo, 1996). The tasks considered include preparation of a proposal, feasibility analysis, preparing a project plan, gathering information, preparing requirements specifications, preparing design specifications (architectural design, input/output design, program design, database schema) and developing a prototype. The soft-skills (also called value skills) considered are interpersonal, communication, team building, and leadership. The third component in the study dealt with the roles the individuals played for completing the tasks in the project. These roles also provided opportunities to develop their soft-skills. The students indicated their perception concerning learning outcomes in all the above categories on a 5-point progressive scale. The survey was conducted in spring 2002 and fall 2002 and was given to a combined student group of 49 students. A total of 34 students responded (21 MIS majors and 13 ACS majors) for a response rate of 69.4 percent. The survey was completed at the end of the semester and some of the student had already left town and could not complete the survey.

The results of this survey on student perceptions are presented in Tables 2 and 3. The results show more than average scores in both task and skill related learning outcomes for both MIS and ACS majors. On the task related outcomes (Table 2), the ACS students perceived they learned more in areas such as feasibility analysis and gathering information. The MIS students, on the other hand, perceived they learned more in preparing user interface (input /output) design and architectural design. These relative gains reflect the differing backgrounds (the initial knowledge with which they came to the course) of the two majors and their pre-existing foci (business focus for MIS and technical focus for ACS). The learning outcomes are positive and fairly close to each other in the areas of preparing the requirements specifications and system design (process and database) specifications.

Table 3 shows that MIS majors seem to have acquired more soft-skills than their ACS counterparts (in particular communications, team building and planning skills). While the combined course offering helped both majors in achieving above average learning outcomes, the interesting question would then be to examine if there were any significant differences in the learning outcomes between the two majors. A statistical analysis using t-distribution assuming equal means was carried out between the MIS and ACS majors. The results indicated no statistically significant differences between the learning outcomes of the two majors apart from the task *carry out feasibility analysis*. The results show that both MIS and ACS students acquired value skills as they engage in the course. This implies that both ACS and MIS students learned equally well in the combined SA&D course offering.

Role-plays are important since they offer the students greater opportunities to apply and develop their soft skills. The question on role-play was included to get a feel for how many different roles this approach helped the students to take on during the course of the project. The analysis of responses indicated that the overall, the course averaged 6.0625 roles per student. The ACS student averaged 6.476 roles per student while the MIS students averaged 5.40 roles per student. Of the 10 roles identified, database designer and programmers were the two most common roles among the ACS students. For the MIS students, business analyst and systems analyst were the two most selected roles. The combined course allowed students to play more roles than would be expected in a traditional SA&D course. The students were asked to indicate the extent to which the combined curse facilitated role play in a number of roles. Table 4 summarizes the results of the facilitated role. Although the results show no statistical difference between the MIS and ACS students in terms of their role facilitation, it appears the MIS students have higher means in non-technical roles whereas the ACS students generally have higher means in the technical role. In the role of a programmer, the results show statistical difference between the ACS and the MIS students.

Based on their strong technical skills, the ACS students had higher propensity to serve as programmers than the MIS students.

Table 2. Learning Outcomes for Task-Related Objectives in the SA&D Course

Task	MIS			ACS			Combined		
	Mean	Std Dev	Std Error	Mean	Std Dev	Std Error	Mean	Std Dev	*p-value**
Prepare a Proposal	4.05	0.740	0.161	4.31	0.751	0.208	4.00	0.888	0.329
Carry Out Feasibility Analysis	3.76	0.944	0.206	4.38	0.650	0.180	4.15	0.774	0.045
Prepare Program Plan	3.90	0.768	0.168	4.00	1.080	0.300	3.94	0.886	0.766
Gather Information	3.38	0.865	0.189	3.85	0.987	0.274	3.56	0.927	0.158
Prepare Requirement Specifications	4.14	0.854	0.186	4.08	0.862	0.239	4.12	0.894	0.829
Prepare System Architectural Design	4.10	0.889	0.194	3.69	1.109	0.308	3.94	0.903	0.251
Prepare input/output design (user interface)	4.14	0.655	0.143	3.62	1.121	0.311	3.94	0.886	0.092
Prepare Detailed Design Specifications	3.95	0.740	0.161	3.85	1.214	0.337	3.91	0.993	0.752
Develop Prototype	3.81	0.928	0.203	3.62	1.261	0.350	3.74	1.053	0.609
Develop System User Manual	3.71	0.644	0.140	3.92	0.760	0.211	3.79	0.687	0.397
Average of Means	3.89			3.93			3.91		

**p-value based on a two-tail t-test with α = 0.05*

Table 3. Learning Outcomes on Skill-related Objectives of the SA&D Course

Description	MIS			ACS			Combined		
	Mean	Std Dev	Std Error	Mean	Std Dev	Std Error	Mean	Std Dev	*p-value**
Interpersonal Skills	3.91	0.644	0.14	3.92	0.76	0.211	3.91	0.687	0.397
Technical Skills	3.71	1.056	0.23	3.54	1.664	0.462	3.65	1.3	0.708
Analytical Skills	3.38	1.071	0.234	3.54	1.506	0.418	3.44	1.236	0.724
Communication Skills	3.40	1.231	0.275	2.92	1.115	0.309	3.21	1.193	0.268
Team Building Skills	3.43	1.287	0.281	3.15	1.405	0.39	3.32	1.319	0.563
System Wide Knowledge	3.71	1.231	0.269	3.62	1.557	0.432	3.68	1.342	0.838
Planning Skills	3.57	1.121	0.245	3.23	1.235	0.343	3.44	1.16	0.414
Leadership Skills	3.48	1.167	0.255	3.33	1.435	0.444	3.42	1.251	0.758
Average of Mean		3.573			3.406			3.494	

**p-value based on a two-tail t-test with α = 0.05*

Table 4. Degree of Role Facilitation in the Combined SA&D Course

	MIS		ACS		Combined		
	Mean	Std Dev	Std Dev	Mean	Mean	Std Dev	p-value*
Project Leader	3.667	0.707	0.793	3.685	3.675	0.804	0.439
Business Analyst	3.700	0.949	0.692	3.579	3.600	0.770	0.332
Systems Analyst	4.100	0.876	0.765	3.842	3.900	0.803	0.255
Process Designer	3.600	0.966	0.688	3.842	3.667	0.922	0.148
Database Designer	3.200	1.033	1.033	3.067	3.088	1.077	0.095
GUI/Report Designer	3.000	1.414	1.125	3.250	3.120	1.201	0.186
Programmer	2.444	1.130	1.071	3.421	3.034	1.209	0.031
Tester	3.333	1.225	0.683	3.750	3.615	0.898	0.167
Researcher	4.000	0.707	1.147	3.375	3.595	1.023	0.175
Documentation Specialist	3.625	0.744	0.786	3.257	3.605	0.752	0.110
Other	2.750	1.258	0.866	3.333	3.154	0.987	0.122

**p-value based on a two-tail t-test with $\alpha = 0.05$*

Discussions and Conclusions

The approach used in teaching SA&D to cross-disciplinary student teams with both business and technical backgrounds enhances their learning and simulates what these students will encounter when they join the workplace. The project-driven approach used in the combined SA&D course offers additional learning opportunities for the MIS and ACS majors. The ACS majors learn about business value of the IS solution, and the MIS majors appreciate the challenges of implementing an IS solution. This method allowed students to work in cross-disciplinary teams and helped them develop interpersonal and communication skills, which are often lacking in many student group settings. The students also developed mutual trust and respect for both disciplines represented in the course. Although some students were initially concerned about the combined cross-listing approach to teaching SA&D, they quickly realized the value of such a course offering in preparing them for challenges and opportunities that they are bound to encounter in the real-world.

Value of Mixed Team Projects

The use of mixed-team projects enhanced student learning in a combined SA&D course comprising of MIS and ACS students. Student perceptions (both qualitative and quantitative) were used to study the value of mixed-team projects. The initial qualitative analysis helped considerably in reassuring the value of this approach to students. At ISECON 2002, the authors presented a few tactics that contributed in creating a valuable experience to students in this combined cross-listed SA&D course (Surendran, Ehie, & Somarajan, 2002). These are summarized below:

- Ensure, as far as possible, that teams have balanced skills
- Allocate class time for project meetings
- Scope the requirements for each phase
- Give the best solution for a phase to be used as a basis for the next phase
- Liaise with the team coordinator to ensure good teamwork
- Ask for who-did-what statements (objective is to have overall workload balance)
- Post a complete sample of project with deliverables for all the phases.
- Give a new project every time (not later than second week)

- Relate course material to project at every opportunity.

The authors designed a new instrument, based on a 5-point scale, the contents of which reflect the academic and industry perceptions of the knowledge and skills required in SA&D for use in the quantitative analysis on students' perceptions concerning learning outcomes in both technical and value-skills development. The results clearly indicated more than average learning enhancement due to working in mixed-teams. These confirmed the earlier informal feedback from the students and highlighted the role-play opportunities the project had provided. The findings reported in earlier studies (Chen, el. al., 2003; Sullivan, 1993) concerning the value of role-play are confirmed in this study. However the role-play and the mixed-team approach facilitated the development and enhancement of both technical and soft-skills, including communication and interpersonal skills. There was no evidence of differences in the learning outcomes between the MIS and ACS students.

In a similar study on a combined database course using data from two cohorts, Van der Vyver and Lane (2003) show that the Information Technology Management students benefited more by working with Computer Software Development students. The results, based on student perceptions, presented in this paper do not indicate such significant differences in learning enhancements between the two majors. Perhaps this could be due to the differences between the approaches taken for teamwork in the two studies. Our study involved a higher level of teamwork for a longer duration (semester-long group project involving four phases as opposed to part-semester pair programming), the style of teamwork (opportunities for different role-plays useful for the development of soft-skills instead of using lead-programmers), wider objectives of teamwork (the whole spectrum of system development rather than just coding), and the variety and depth of analysis used (additional qualitative and a quantitative study on student perception instead of one quantitative study on two different cohorts). Thus, the study presented in this paper provides a comprehensive analysis on the value of mixed-team projects in a combined course.

Value of Combined Cross-Listed Courses

Academia has been accused of functional silos and this integrated approach attempts to address this problem. Although both the ACS and MIS departments traditionally offer the SA&D course, there had been little or no collaboration between the two departments. The argument often presented is that ACS approaches the SA&D course from a technical perspective, whereas MIS teaches the course from a business perspective. Besides the aforementioned curricula issues, a greater part of the problem lies in turf battles that traditionally prevail between two departments. These turf battles completely ignore the real-world issues that students face when they enter the workforce, where working in cross- functional and cross-disciplinary teams is the norm. Students are expected to look beyond their disciplines and work with team members from another discipline to develop systems that will meet client requirements. Considering the value of the combined cross-listed SA&D offerings, both departments have collaborated on both research and curricula issues. When department collaborate, the students gain from the synergistic effect explained in this paper. A recent survey study of 10 schools in US found that although there were differences between programs in information technology, information systems and computer science, the similarities among them are more significant (Reichgelt, et al., 2004). This is evident in the average number of roles played by the students in the combined course offerings. Although the ACS students saw themselves in slightly more roles than the MIS students during the period of the course, the results show no significant difference in all the roles with the exception of the role of a programmer. ACS students have stronger preference for the role of programmers than their MIS counterpart. Such results reinforce the value of the cross-listing approach described in this paper, especially when the competing curricula permit.

Although the study focused on the collaboration between ACS and MIS, the findings of the study could have applications in other disciplines. The underpinning assumption of the study is a course delivery mechanism that involves multidisciplinary student teams. One area that is currently being considered is to offer a senior capstone course that involves business and engineering students. The business strategy courses will be combined with the senior engineering design course and delivered using the methods described in this paper. The engineering students will work on the engineering design of the product and the business students will concurrently work on developing a business plan and strategy for the product and how to bring the product to market in a timely manner. This approach resembles cross-functional teams, which are used frequently in the business world. Having students work on company-sponsored projects should further enhance the value of the integrated mixed-team approach as it simulates the real-world experience more closely.

References

Archer, C. B. (1983). What does business and industry expect from computer science graduates today? *ACM SIGCSE Bulletin*, *15* (1), 83-84.

Avison, D. E. (1991). Action programmes for teaching and researching in information systems. *The Australian Computer Journal*, *23* (2), 66-72.

Bruce, C., Buckingham, L., Hynd, J., McMahon, C., Roggenkamp, M., & Stoodley, I. (2004). Ways of experiencing the act of learning to program: A Phenomenographic study of introductory programming students at university. *Journal of Information Technology Education*, *3*, 143-160.

Bruce, C. & Gerber, R. (1994). Towards university lecturers conceptions of student learning. In R. Ballantyne & C. Bruce (eds.), *Phenomenography: Philosophy and practice,* QUT, Brisbane, Australia, pp. 57-70.

Cardinali, R. (1988). Business school graduates – Do they meet the needs of MIS professional? *Word*, *16* (5), 33-35.

Chen, L., Muthitacharoen, A. & Frolick, M. N. (2003). Investigating the use of role play training to improve the communication skills of IS professionals: Some empirical evidence. *Journal of Computer Information Systems*, *43* (3), 67-74.

Cope, C. J., & Horan, P. (1996). The role-played case: An experiential approach to teaching introductory information systems education. *Journal of Information Systems Education, 8* (2&3), 33-39.

Davis, G.B., Gorgone, J., Couger, D., Feinstein, D, & Longenecker, H., Jr. (1997). IS 1997 model curriculum and guidelines for undergraduate degree in information systems. ACM, AIS and AITP. Association for Computing Machinery.

Dennis, A. & Wixom, B. H. (2003). *Systems analysis and design: An applied approach*. John Wiley & Sons.

Ehie, I. C. (2002). Developing a management information systems (MIS) curriculum: Perspectives from MIS practitioners. *Journal of Education for Business*, *77* (3), 151-158.

Entwistle, N. & Marton, F. (1994). Knowledge objects: understandings constituted through intensive academic study. *British Journal of Educational Psychology*, *64*, 161-178.

Etzkorn, L.H., Weisskop, M. E., & Gholston, S. (2004). A study of students' performance in combined courses. *Journal of Information Systems Education*, *15*(2), 163-169.

Fox, T. L. (2002). A case of real-world systems development experiences of CIS students. *Journal of Information Systems Education*, *13*(4), 343-350.

Gorgone, J., Davis, G., Valacich, J., Topi, H., Feinstein, D., & Longenecker, H., Jr. (2003). IS 2002 model curriculum and guidelines for undergraduate degree programs in information systems. *Communications of the Association for Information Systems, 2*(1)]

Little, S. E. & Margetson, D. B. (1989). A project-based approach to information systems design for undergraduates. *The Australian Computer Journal*, *21*(2), 130-137.

Lynn, T., Baughn, J., Hays, A., Langlitz, E. H., Mirly, H., & Surendran K. (2002). Object orientation in action: A system for designing speech articulation deficiency treatment. *ISECON-2002 Conference Proceedings*, Paper 222a.

Marton, F. & Saljo, R. (1984). Approaches to learning. In F. Marton, D. Hounsell & N. Entwistle (Eds.), *The experience of learning.* Edinburgh: Scottish Academic Press.

Miller, R. A. & Luse, D. W. (2004). Advancing the IS curricula: The identification of important communication skills needed by IS staff during systems development. *Journal of Information Technology Education*, *3*, 118-131.

Misic, M. M. & Russo, N. (1996). Educating systems analysts: A comparison of educators' and practitioners' opinions concerning the relative importance of systems analyst tasks and skills, The *Journal of Computer Information Systems*, *36*(4), 86-91.

Misic, M. M. & Russo, N. L. (1999). An assessment of systems analysis and design courses. *The Journal of Systems and Software*, *45*(3), 197-202.

Polack-Wahl, J. (1999). Incorporating the client's role in a software engineering course. *Proceedings of the 30th SIGCSE Technical Symposium on Computer Science Education,* New Orleans, ACM Press, pp. 73-77.

Ranking the Attributes. (2004, September 22). College Journal from the *Wall Street Journal*. Retrieved February 2005 from http://www.collegejournal.com/reports/bschool05/glance/20040922-attributes.html

Reichgelt, H., Lunt, B., Ashford, T., Phelps, A., Slazinski, E., & Willis, C. (2004). A comparison of baccalaureate programs in information technology with baccalaureate programs in computer science and information systems. *Journal of Information Technology Education*, *3*, 19 -34.

Sullivan, S. (1993). A software project management course role-play-team project approach emphasizing written and oral communication skills. *SIGCSE Bulletin*, *25*(1), 283-287.

Surendran, K., Ehie, I., & Somarajan, C. (2002). Teaching Systems analysis and design in a practical way: A collaborative effort between computer science and business school. *Proceedings of the 2002 Information Systems Education Conference*, San Antonio, Texas, paper# 254C on CD

Surendran, K., Hays, H. M. & Macfarlane, A. (2002). Simulating software engineering apprenticeship, *IEEE Software*, *19*(5), 49-56.

Surendran, K. & Young, F. H. (2001). Teaching software engineering in a practical way. *The New Zealand Journal of Applied Computing and Information Technology*, *5*(2), 75-79.

Trauth, E. M., Douglas, W. F. & Lee, D., (1993). The IS expectation gap: Industry expectation versus academic preparation. *MIS Quarterly, 17*(3), 293-307.

Trigwell, K. & Prosser, M., (1991). Improving the quality of student learning: The influence of learning context and student approaches to learning on learning outcomes. *Higher Education*, *22*, 251-266.

Van der Vyver, G., & Lane, M. (2003). Using a ream-based approach in an IS Course: An empirical study, *Journal of Information Technology Education*, *2*, 396-406.

Appendix: Systems Analysis and Design Course Survey

This survey is designed to evaluate students learning in the cross-listed Systems Analysis & Design course intended for both for Computer Sciences (CS) and Management Information Systems (MIS) students.

Please complete the following questions.

Indicate your major: ☐ CS ☐ MIS ☐ Other (pls. specify)__________________

Student Status: ☐ Junior ☐ Senior ☐ Others (pls. specify)__________________

IT related work experience: ☐ 0-1 years ☐ 2-3 years ☐ 3-5 years ☐ over 5 years

Type of IT related work: ____

Circle the number that indicates the extent to which the joint class between ACS and MIS students facilitated your learning on the tasks below

TASKS	Definitely Disagree	Disagree	Not Sure	Agree	Definitely Agree
Prepare a proposal for an IT- based business solution	1	2	3	4	5
Carry out a feasibility analysis for system solution (economic, technical and operational feasibilities)	1	2	3	4	5
Prepare a project plan (objectives, work breakdown structure, resources and schedule) for developing a system.	1	2	3	4	5
Gather information (interview, questionnaire) from users for the intended system	1	2	3	4	5
Prepare requirements specification and develop logical system models (Example: ERD, Context, DFDs)	1	2	3	4	5
Prepare system architectural design and choose appropriate design strategies (Example: Client-server model, Physical DFD and ERD)	1	2	3	4	5
Prepare input/output designs (prototype user interfaces)	1	2	3	4	5
Prepare detailed design specifications for programs and database (Database Schema, Structure Chart)	1	2	3	4	5
Develop a prototype for selected system facilities (use of language tools)	1	2	3	4	5
Prepare system and user manuals	1	2	3	4	5

SKILLS CATEGORIES	Very little	Some-what	Notice able	Good	Very Good
Interpersonal skills (Get along well; work objectively)	1	2	3	4	5
Technical Skills (Know the concepts, apply appropriate tools)	1	2	3	4	5
Analytical Skills (Abstraction, unbiased situation analysis and scoping, cost-benefit analysis)	1	2	3	4	5
Communication Skills (report writing, discussion and presentation at meetings)	1	2	3	4	5
Team-Building Skills (Negotiation, organizing / managing meetings, brainstorming)	1	2	3	4	5
Knowledge of systems wide concept (Abstraction, portioning, Scaling, System interface, SDLC)	1	2	3	4	5
Planning Skills (plan activities, assign tasks, estimate resource)	1	2	3	4	5
Leadership skills (lead by example, coach, resolve resource early on, delegate responsibilities)	1	2	3	4	5

Tick **all** the roles you played during the entire project duration (even for a short period):

☐ Team coordinator ☐ Business analyst ☐ Systems analyst ☐ Process designer

☐ Database designer ☐ GUI / report designer ☐ Programmer ☐ Tester

☐ Researcher ☐ Documentation specialist

☐ Any other role (specify): ____

Circle the number that indicates the extent to which this cross-listed course facilitated you in playing any of above the roles.					
Role	Very little	Some-what	Notice able	Good	Very Good
Project Leader (or even a coordinator)	1	2	3	4	5
Business Analyst	1	2	3	4	5
Systems Analyst	1	2	3	4	5
Process Designer	1	2	3	4	5

Database Designer	1	2	3	4	5
GUI / report designer	1	2	3	4	5
Programmer	1	2	3	4	5
Tester	1	2	3	4	5
Researcher	1	2	3	4	5
Documentation specialist	1	2	3	4	5
Others	1	2	3	4	5

Biographies

Ken Surendran is an Associate Professor in the Department of Computer Science at Southeast Missouri State University. His research interests include Software Engineering, Systems Analysis and Design, and Security Management Education.. Surendran received a B.E. in Electrical Engineering from University of Madras, India, M. Tech. in Electrical Engineering from Indian Institute of Technology, Madras, India, and Ph. D. in Applied Analysis from State University of New York at Stony Brook. He is a senior member of IEEE and a member of ACM.

Ike C. Ehie is the Associate Dean of the College of Business Administration at Kansas State University, Manhattan, Kansas, USA . He holds a Ph.D. degree in Engineering Management from the University of Missouri, Rolla, USA. He teaches in the area of supply chain management, operations management and quantitative analysis in business. He research is focused on supply chain management, manufacturing strategies, and ERP implementation. Dr. Ehie has published over 60 journal articles, conference proceedings and book chapters. He is a fellow of the Certified Production and Inventory Management (CFPIM).

Chellappan Somarajan is a professor in the Department of Accounting and Management Information Systems at the Donald L. Harrison College of Business, Southeast Missouri State University, Cape Girardeau, Missouri, USA. He received his Ph.D. in Management Science from Oklahoma State University, Stillwater, Oklahoma. He has more than 25 publications, refereed conference proceedings, and abstracts published by Computers and Operations Research, Journal of Operational Research Society, European Journal of Operational Research Society, International Journal of Systems Science, O. R. Insight, and others. He teaches in areas of Production/Operations Management, Business Data Base Systems, ERP Systems (SAP), and Project Management.