ISSN: 1545-679X

Information Systems Education Journal

Volume 4, Number 80 http://isedj.org/4/80/ September 21, 2006

In this issue:

Group Projects in In-Ground Undergraduate and On-Line Graduate Degree Programs: Guidelines for Success

Kathleen Kelm
Edgewood College
Madison, WI, USA

Gail Miles
Lenoir-Rhyne College
Hickory, NC 28603 USA

Abstract: Professional organizations and employers continue to recommend the use of team based projects for undergraduate and graduate degree programs in Information Systems and Technology. The delivery model of in-ground and on-line degree programs are different in terms of physical versus virtual space. Yet, the use of group projects reveals similarities in challenges in both paradigms. Effective teamwork requires collaborative work space, accountability for contributions, and effective time management. This paper discusses how these are integrated in both in-ground undergraduate programs and on-line graduate programs, in particular, in courses that produce software programs and research papers. Recommendations for best practices are included at the end of the discussion.

Keywords: online graduate group projects, undergraduate group projects, collaborative work, course in software engineering, operating systems, project management

---

Recommended Citation: Kelm and Miles (2006). Group Projects in In-Ground Undergraduate and On-Line Graduate Degree Programs: Guidelines for Success. *Information Systems Education Journal, 4* (80). http://isedj.org/4/80/. ISSN: 1545-679X. (Also appears in *The Proceedings of ISECON 2005:* §2342. ISSN: 1542-7382.)

This issue is on the Internet at **http://isedj.org/4/80/**

The **Information Systems Education Journal** (ISEDJ) is a peer-reviewed academic journal published by the Education Special Interest Group (EDSIG) of the Association of Information Technology Professionals (AITP, Chicago, Illinois). • ISSN: 1545-679X. • First issue: 8 Sep 2003. • Title: Information Systems Education Journal. Variants: IS Education Journal; ISEDJ. • Physical format: online. • Publishing frequency: irregular; as each article is approved, it is published immediately and constitutes a complete separate issue of the current volume. • Single issue price: free. • Subscription address: subscribe@isedj.org. • Subscription price: free. • Electronic access: http://isedj.org/ • Contact person: Don Colton (editor@isedj.org)

EDSIG activities include the publication of ISEDJ, the organization and execution of the annual ISECON conference held each fall, the publication of the Journal of Information Systems Education (JISE), and the designation and honoring of an IS Educator of the Year. • The Foundation for Information Technology Education has been the key sponsor of ISECON over the years. • The Association for Information Technology Professionals (AITP) provides the corporate umbrella under which EDSIG operates.

Group Projects in In-Ground Undergraduate and On-Line Graduate Degree Programs: Guidelines for Success

Kathleen Kelm
kkelm@edgewood.edu
Math & Computer Science
Edgewood College
1000 Edgewood College Dr.
Madison, Wisconsin, USA
and
Laureate Online Education/
University of Liverpool
Liverpool, England, UK

Gail Miles
milesg@lrc.edu
Computing Sciences & Mathematics
Lenoir-Rhyne College
Hickory, North Carolina 28603 USA
and
Laureate Online Education/
University of Liverpool
Liverpool, England, UK

Abstract

Professional organizations and employers continue to recommend the use of team based projects for undergraduate and graduate degree programs in Information Systems and Technology. The delivery model of in-ground and on-line degree programs are different in terms of physical versus virtual space. Yet, the use of group projects reveals similarities in challenges in both paradigms. Effective teamwork requires collaborative work space, accountability for contributions, and effective time management. This paper discusses how these are integrated in both in-ground undergraduate programs and on-line graduate programs, in particular, in courses that produce software programs and research papers. Recommendations for best practices are included at the end of the discussion.

Keywords: online graduate group projects, undergraduate group projects, collaborative work, course in software engineering, operating systems, project management

1. INTRODUCTION

The importance of group work in computing and technology courses has long been acknowledged. It was formally standardized by a joint IEEE Computer Society/ACM Task Force in 2001 with the "Model Curricula for Computing" (CC).(ACM, 2001) and in the IS2002 Model Curriculum (Gorgone et al, 2002). The need to include 'Professional Practice' in the academic study of computing was clearly defined to include those skills

which are expected in real-world situations One of the key areas of professional practice is the need to include the study and practice of teamwork skills in models of work environments (Deming & Durnhan, 2001). Working in teams is not always a natural way of learning for many students. In fact, some students often choose to omit a course because of the team work requirements.

However, experience in teamwork is one of the most important tools students must have to be successful professionals in today's complex computing environments. Skills which students learn by participating in teams include, not only how to manage a team or even how to be led by a team manager, but also such skills as "planning, budgeting, organizational, and interpersonal skills" (ACM (2), 2001). Project experiences with required teamwork significantly improve the students' ability to problem solve difficult technical problems. This paper discusses the use of group projects in both in-ground undergraduate programs and in an on-line graduate, degree program.

2. DEGREE PROGRAMS IN INFORMATION SYSTEMS AND TECHNOLOGY

Both authors teach various courses on information systems and technology: (1) separately in two different in-ground, undergraduate liberal arts colleges, and (2) together in an online graduate program in Information Technology. All three programs are structured around content courses, with each course representing a specific skill that the student will find useful to his or her career. The size of the classroom (either physical or virtual) is designed to encourage interactivity both inside and away from the classroom. Typically, the interaction is intense, networking takes place, and friendships are established. The undergraduate programs make use of a capstone experience as a way for students to demonstrate the integration of their skills in a manner required by the information systems profession. The graduate program requires the completion of a research project, in the topic area of the student's choosing or a software project (depending on the course objectives).

Both authors teach in undergraduate programs in Information Systems and Technology. These programs make extensive use of projects so as to enhance student skills in areas of project and time management and peer communication. For example, the Software Engineering course at Lenoir-Rhyne College, offered for about 15 years, follows the common practices of the Software Engineering Institute. Each student is required to be a member of a team organized around the Chief Programmer Team approach (Brooks, 1975). Students are assigned to a team, and the team members determine the various roles in the team. Each team is assigned a 2-semester project for a 'real' customer. The course has been very successful in preparing students for the workplace. Students who have completed a 'large' team-based project are more heavily recruited, and employers have consistently reported positive feedback. The project is completed in a senior capstone course where students are required to demonstrate effective project management skills. Similarly, at Edgewood College, the Information Systems Project course is designed so that all students work together as members of a consulting team for the entire semester. Their project typically involves providing systems development services to a local, not-for-profit organization. Experiences during the capstone project reinforce the benefits of effective project management and teamwork.

On-line learning, as an educational delivery system, is not new, but the ease in which non-traditional programs can be delivered to students has increased. The Internet allows students to geographically study beyond the boundaries of a campus, institution, or country. Research studies measuring the global impact of on-line higher education programs have found that, with appropriate course structure determined by course content, students can be successful (Wilson, 2001). Many on-line Masters' programs demonstrate the success of collaborative learning and sound pedagogy, reflecting the findings by both Bocchi, Eastman & Swift (2004) and O'Brien & Renner (2002) who report that the use of teams and collaborative projects increases retention rates in online programs

Courses that once were not considered candidates for on-line learning by virtue of their content or presentation type now exist in completely on-line virtual classrooms. Engineering and Computer Science courses were often included in this group. Li (2001)

provided evidence that not only can project-based courses be offered totally on-line, but they facilitate student success in the on-line classroom. Other studies have also found that the on-line environment is well suited for collaborative learning (learning in groups) because it emphasizes group interaction (Gunawardena, Nolla, Wilson, Lopez-Islas, 2001).

English is the most widely used language in delivery of information systems and technology degrees, using the on-line distance learning paradigm. Thus, students must possess a reasonable command of the language to be successful in the learning environment (Bates, 2001). This was critical for collaborative teams because, as with undergraduate program, group project work requires strong communication skills.

The characteristics of graduate students in MSc. on-line courses are somewhat different and have an impact on the success of the course work. Most students in these types of classes have been working computer professionals. This background is a significant contributor in building a sense of community very early in the class. Although students work in a different area of the profession and have different cultural backgrounds, they continue to have similar interests, experiences, goals, values or vision. For Software Engineering courses, a typical project team might include a strong programmer who helps other team members learn programming skills, or a team member who has an excellent command of the English language helping other team members improve their language skills. For Operating Systems and Project Management courses, very often the leader of the team is one who either has expertise in the subject area (such as in an operating system) or has experience from the work environment (such as holding a position of Project Manager in an organization).

For teamwork to be successful in the on-line environment, each team must be assigned its own private virtual place where team members can communicate and share files. It is also helpful for the course to have a virtual project space where the instructor acts as the customer, and students can request information related to the project. All students should have access to the project space allowing all students to 'hear' answers to any project-related questions.

Common configurations for group projects include: (1) entire semester or term length, (2) sequential weeks of a limited timeline (such as two-three weeks), (3) non-sequential weeks, with multiple stages (to be completed at different timeframes, such as once every two weeks, (4) non-sequential weeks, non-related single stages (to be completed at different timeframes) and (5) single stage, single timeframe (such as one week in duration). In the next section, two different group projects configurations will be presented, showing commonalities and differences between design and assessment of activities.

3. GROUP PROJECTS IN SOFTWARE ENGINEERING COURSES

Implementing groups into Software Engineering courses in traditional in-ground courses has become common. Research has consistently shown that the use of team collaborative activities significantly enhance the students' ability to work in teams in the workplace. There are many project management books and articles, specifically designed to help successfully implement teams in a traditional classroom. One of the major recurring problems in using team work is the assessment of individual effort. Since much of the work is done outside the classroom, it is often difficult to determine who did what. Students are not often forthcoming in how much work each student has contributed to each deliverable. Most of the work is hidden from the instructor.

When developing a Software Engineering module for the graduate program, the author was mindful of the conclusions reached by Brereton, Lees, Gumbley, Boldyreff, Layzell, Macaulay and Young (1998). One of the factors they found for success was a common place for each group to store and share their documents. They also found on-line chat and shared whiteboards to be useful as well. Thus, the important criteria for success appear to be solid communication and management within the teams.

The Software Engineering course, developed as a totally on-line environment, is assigned a virtual class space with a certified online instructor who has expertise and education

in the Software Engineering topics. For an on-line course to be successful there must be structured methods for promoting discussions to all class members as well as private group space. The Software Engineering course developed for the on-line MSc, was divided into individual class times which are set aside for specific content for that time period. Students participate in moderated discussions in an asynchronous weekly virtual classroom but also have the ability to communicate privately with the instructor through e-mail. All parts of the course are delivered via the Internet. Weekly assignments are based on a course textbook, and assessment is based on individual and group assignments. The course follows a traditional Software Engineering course covering a good part of the Sommerville Software Engineering (Sommerville, 2004) book.

The first week in an on-line software engineering course is a time when the students become acquainted with each other especially in terms of their attitudes toward the course and their strengths as they relate to a team project. Each student must become a member of a three or four person project team. They are expected to form their own teams as soon as possible – during the first few days. This can be accomplished by getting to know each other in a asynchronous or synchronous chat environment. These activities helped solidify the team process (Brown, 2001).

Many students in an in-ground team environment are somewhat ambivalent to the team project work. However, students in the on-line environment, are more likely to rate the experience as high. Students consistently rated the course high.

The team project for a Software Engineering course can be any application as long as it can be assessed by the instructor – one example for a software engineering course is a simple web-enabled database application. Students are given web development space on a class server. The project work follows a traditional cyclic approach. The first project assignment is given as soon as the teams are formed. Each team produces deliverables that are found in the traditional software engineering curriculum -- a Project Plan, a Requirements document, Final Design Document, Prototype, Quality Plan and a Test Plan.

4. GROUP PROJECTS IN OPERATING SYSTEMS AND PROJECT MANAGEMENT COURSES

In undergraduate programs, that entail research projects, typical challenges identified by students include (1) difficulty in finding common time for discussion of the project requirements, (2) differences in levels of research and writing competencies (3) differences in styles of time management for deadlines and (4) inequality of contribution of work The goals of research projects in Operating Systems and Project Management courses, in particular, are to not only increase content knowledge, but also to promote practice of collaborative writing for effective communication and project management. In the past three years, employers have identified these areas of expertise of increasing value for the Information Systems and Technology professions (Galllivan, Truex & Kyasny, 2004).

In an online environment, group projects for the Operating Systems and Project Management courses takes the form of a paper as the 'product' Both courses require groups to collaboratively author a research paper. In the case of the Operating Systems course, the students complete the paper in two stages, with a two week break between stages. The Project Management course has the groups analyze and report on a set of project case studies. Neither of these two courses uses a precise model for development of the deliverables. The students must use effective time management techniques developed from their own personal experiences. As with the Software Engineering course, collaborative work space is provided for members of the team to post versions of the paper, text from online, synchronous chats and final production of combined contributions.

Without the imposed deadlines of a structure (such as ones found in a Software Engineering course), group projects typically take on a tone of 'we are out of time, we must get moving on this project'. Analysis of a recently completed Operating Systems course is typical: The group project for this course is a two-stage case study report, the first due in week four, the second in week seven. Observations of group communication in their individual folders revealed that, during the first stage, only one group used the en-

tire week to research and collaboratively develop the final report. In this case, the 'leader' of the group was 'elected' at the very beginning of the group formation, and as leader, guided the other members of the group, using constructed task lists, assigning roles and responsibilities at the outset. The remaining groups (four) showed poor use of time and were forced to submit their reports after working collaboratively for only the last two days of the week During the second stage, the first group continued using good time management, one group improved in that it began the project sooner (in day three) but the remaining (three) groups showed no improvement in how they employed their time. Statistics gathered from three other Operating Systems courses, completed within the past twelve months show that only 22% of groups used all seven days for project work, while 46% used three-five days and the remaining 32% used two days or less.

5. OBSERVATIONS

Allowing the teams to self-select their members, forces the selection to be done based on strengths in a variety of proficiencies. A team member may be chosen because he or she possesses programming skills, but almost as important may be the ability to manage projects or write English documentation. Team members provided support and encouragement to each other (supported by Willis, 2000). They clearly understand the project deliverables can not be completed without interactive group work. Students who report a lack of knowledge or skill in some area receives support from other team members.

The retention rate in all three courses is high. On average, less than 10% do not complete the course once the project work begins. For the Software Engineering course, every project team must work as a team to produce the project. It is human nature that some teams work better together than others, but, all teams appear to function together. Each team chooses its method of work. There were several rules they are required to follow. All teams should have a team leader for the assigned task. Some teams may choose to have the same team leader throughout the process. Others teams may change team leaders each time based on the skill level for the assigned weekly task. Occasionally, the team leader may be assigned based on the time constraints of other team members.

By monitoring the discussions, the instructor has an excellent picture of each of the team member's contributions. This is an important point to make concerning this process. Each deliverable should be graded as a group grade. In 90% of the cases, the student's individual weekly project grade will be the grade given to the group for the deliverable of the week. Occasionally, a team member may receive a different grade based on the amount of work the student completed that week.

For Operating Systems and Project Management courses, no leader is required. Typically the groups form based on location in world, so as to facilitate easy communication using synchronous chats and to ensure that team members work relatively at the same point in time during the day. However, unlike the Software Engineering course, group projects in Operating Systems and Project Management courses, do not naturally have the persistence in communication, in that there are breaks between group activities. This is clearly evident when monitoring the postings in the folders. In the previously cited Operating Systems courses, very few continued to communicate after the week's work is completed, and in every case, those continuing the process were groups who had submitted excellent reports. The postings contain complementary comments, reflecting feelings of excitement over the achievement of excellence by the group as a whole.

It is interesting to see the evolution of team dynamics with such different nationalities and backgrounds. It is remarkable that every group is able to work effectively to produce the documents. They soon realize that managed communication and time management were critical to get the work done (as supported in Hogan and Thomas 2005). One point that has not been mentioned is global time difference. Because students come from all over the globe in an on-line course, a team might consist of a member from Sweden, one from Canada, and one from China and would develop methods of communication that worked for them. Some teams may choose synchronous chat times (available in the class envi-

ronment). The quality of the deliverables is often exceptional in these on-line environments. Documents produced by these teams can stand up to any professional quality documents of their kind. This is often done with significant time constraints, multinational and multi-lingual teams, and team members who had never met each other face-to-face. As noted in the previous section, students who have participated in these courses have consistently rated them high in terms of quality and the amount of benefit that they drew from it. In many cases, the satisfaction was directly tied to the team work in which they participated.

6. CONCLUSIONS

What is common in both undergraduate and graduate on-line degree programs is the challenge of time management. The courses in Software Engineering show evidence that the more structured the process, the better the collaboration. However, group work in courses that have projects not requiring weekly communication (such as ones in the Operating Systems and Project Management) do produce excellent results, even with a higher incidence of poor time management and uneven contributions of work.

Both in-ground and on-line group projects have proven to be successful in developing a sound educational environment and providing successful team collaboration in a project-based pedagogy that enhance the learning outcomes of the course. The synergy within the various courses and modules assists the students' ability to effectively learn collaboratively. The structure of communication and course content facilitates this process. In the undergraduate programs, collaborative work is enhanced through the use of shared folders and when discussions are captured and recorded by each team. For the online program, the use of group folders to deposit project documents transition them to becoming collaborative documents in which each team member takes ownership. The ability to communicate using both synchronous and asynchronous supports group collaboration in the virtual space of online programs.

Last but not least, as these students progress in their careers as IS and IT professionals, the benefits of working collaboratively on a variety of projects continue long after the course is completed.

7. REFERENCES

ACM, 2001 joint IEEE Computer Society/ACM Task Force http://www.computer.org /education/cc2001/

ACM(2) 2001 Professional Practice . Joint IEEE Computer Society/ACM Task Force http://www.computer.org/education/cc2001/final/chapter10.htm

Bates, Tony. (2001). "International distance education: Cultural and ethical issues". Distance Education, pp. 122-137.

Bourne, J.R. (1998). "Net-learning: Strategies for On-Campus and Off-Campus Network-enabled Learning", Journal of Asynchronous Learning Networks (JALN), 2(2), pp. 70-88.

Bocchi, J., J. Eastman and C. O. Swift, (2004). "Retaining the online learner: profile of students in an online MBA program and implications for teaching them." Journal of Education for Business. 79(4), pp. 245-253.

Brereton, P., S. Lees, M. Gumbley, C. Boldyreff, S. Drummond P. Layzell, L. Macaulay and R. Young. (1998). "Distributed Group Working in Software Engineering Education." The work was undertaken as part of the UK JTAP project number JTAP-2/140 'Developing a Virtual Community for Student Groupware'. The project web site is http://cssec.umist.ac.uk 1998.

Brooks, Fred. (1975). Mythical Man-Month. Addison-Wesley, New York.

Brown, Ruth. (2001). "The Process of Community-building in Distance Learning Classes." Journal of Asynchronous Learning Networks. 5,(2), pp. 1-17.

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

Galllivan, M., D. Truex, , and L. Kyasny (2004) " Changing Patterns in IT Skills Sets" (1988-2003). The DATA BASE for Advances in Information Systems. 35,(3), pp. 64-87.

Gogan, James. and Richard Thomas. (2005). "Developing the Software Engineering

Team." Australasiam Computer Education Conference, 2005, Newcastle, Australia

Gomory Ralph E. (2001). Sheffield Lecture - Yale University January 11, 2000 "Internet Learning: Is It Real and What Does it Mean For Universities?. ". Journal of Asynchronous Learning Networks. 5(1), pp. 139-146.

Gorgone, John, Gordon Davis, Joseph Valacich, Heikki Topi, David Feinstein & Herbert Longnecker. (2002). IS2002 Curriculum Model. Association for Information Systems, Atlanta.

Gunawardena, Charlotte, Ana. Nolla, Penne Wilson. Lopez-Islas, Jose. (2001). "A cross-cultural study of group process and development in on-line conferences". Distance Education. 22(1), pp. 85-121.

Kember, David and Greg Harper. (1987). "Approaches to Studying Research and Its Implicationsfor the Quality of Learning From Distance Education", Journal of Distance Education/ Revue de l'enseignement à distance. 2,(2).

Kolb, David. A.. (1994). Experiential Learning, Prentice-Hall. New York.

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

O'Brien, B. S. & A. L Renner. (2002)."Online student retention: can it be done? "*Proceedings of ED-Media, 2002 World Conference on Multimedia, Hypermedia and Telecommunications.* Norfolk, VA: Association for the Advancement of Computing in Education. (ERIC Document Reproduction Service No.ED477076).

Parker, Drew and Andrew Gemino. (2001). "Inside Online Learning: Comparing Conceptual and Technique Learning Performance in Place-based and ALN Formats." Journal of Asynchronous Learning Networks. 5(2), pp. 64-74.

Paterson, Kurtis. (1999). "Student perceptions of Internet-based learning tools in environmental engineering". Journal of Engineering Education, 88(3), pg 295-395.

Pimentel, Juan R. (1999). "Design of Net-learning Systems Based on Experiential Learning,". Journal of Asynchronous Learning Networks, 3,(2), pp. 65-90.

Spodick, Edward. (1995). "The Evolution of Distance Learning." Hong Kong University of Science & Technology Paper Presented August, 1995.

Sommerville, Ian. (2004). Software Engineering. Seventh Edition. Addison Wesley. Mew York.

Willis, Barry (2002). Distance Education - Strategies and Tools and Distance Education - A Practical Guide.

Wilson, Michael. (2001). "Cultural considerations in online instruction and learning." Distance Education, 22,(1), pp. 52-64.