

Systems Development Projects – Gaining Practical Experience while Meeting Community Needs: A Win-Win State of Affairs

Herb Rebhun
Rebhun@uhd.edu

Shohreh Hashemi
hashemis@uhd.edu

Computer Information Systems Department, University of Houston-Downtown, Houston, Texas 77002, USA

Abstract

During their senior year, students in the Computer Information Systems Bachelor degree program at the University of Houston-Downtown are required to complete a capstone Senior Project course under the guidance of the Computer Information Systems faculty. To meet this requirement, either online or in a face-to-face setting, students design and implement systems for various Houston-area businesses at no cost. The students benefit by gaining the needed “real world” experience, and local businesses benefit by having a new system at no cost -- a win-win situation.

After describing the capstone Senior Projects course at a four-year public institution, this paper focuses on the pros and cons of the Senior Project capstone course and addresses such questions as whether this course should be required or elective, who should be allowed to enroll in it, what the expected outcomes from the course are, what its prerequisites should be, what topics should be covered in the course, what methods should be used to deliver course materials, what projects are acceptable, and whether group projects should be allowed. In addition, the authors will share their experiences teaching this course, exploring the pitfalls that can be avoided to ensure successful delivery of the course.

Keywords: Senior Projects, Capstone Course, Systems Analysis and Design, Computer Information Systems

1. INTRODUCTION

The University of Houston-Downtown (UHD at <http://www.uhd.edu>) is an open-admission urban university that offers both undergraduate and graduate studies, is located in the heart of Houston, Texas. As such, it is intimately tied to the business district of downtown Houston, and expectations of students and administration alike are that faculty and courseware will reflect that tie. Thus, the university offers a wide variety of academic degrees to prepare students for entry into the workplace or to pursue gradu-

ate work. According to the UHD 2004-2005 Fact Book (http://www.uhd.edu/about/irp/documents/2004_2005_000.pdf), 1,522 of the student body of 11,408 are seeking their degree through the College of Business, which is accredited by the Association to Advance Collegiate Schools of Business (AACSB International at <http://www.aacsb.edu/>). Given the university's thrust to serve business needs of the region and to prepare its students for careers in Houston's business district, the COB has established close ties with local businesses.

It has become standard business practice for the college to seek input from local busi-

nesses through degree program Advisory Boards, where mid-level managers who typically interview and hire from the student population serve by guiding curricular choices that impact businesses in the local community. In addition to drawing the business community into college decision making, the university also encourages faculty outreach to the local business community. Thus, COB faculty members actively support local and regional professional organizations by attending their meetings and conferences, serving on committees, speaking at local gatherings and serving as a resource for local businesses. In addition, COB faculty members set up and supervise student projects to serve local businesses.

Within the COB are two departments, one of which is the Finance, Accounting and Computer Information Systems Department (FACIS), the department through which the Computer Information Systems (CIS) capstone Systems Development Project course discussed in this paper are administered. The FACIS department focuses on delivering first-rate education to students via both face-to-face and distance education classes, and it maintains several laboratories including a CIS computer laboratory where students enjoy working with state-of-art computer hardware and software. Most of the FACIS faculty members have established close ties to local businesses, several of which routinely arrange for speakers and tours of their facilities and provide students with scholarships, internships, job interviews, and employment. While student projects are utilized in most courses, this paper discusses how computer projects in the Systems Development Project course offered by the CIS degree program can help maintain and improve the department's close ties with local businesses while providing students a rare, real-world hands-on experience.

2. THE COMPUTER INFORMATION SYSTEMS (CIS) DEGREE PROGRAM

An understanding of how the capstone Systems Development Project course fits into the CIS degree program will help clarify the following descriptions of the coursework, problems and benefits of the projects. The CIS degree program, which in 1992 was selected by the DPMA Education Foundation (Now called Association for Information Technology Professionals (AITP) <http://aitp.org>) as the outstanding undergraduate CIS program in USA, has been de-

vised using the AITP and ACM model curricula and in consultation with the local computer businesses via the CIS Advisory Board. In order to earn a BBA degree in CIS, students must complete 126 hours of college work, among them 21 hours of CIS required courses which include Computer Algorithms & Problem Solving using C++ (CIS 2301), programming languages (CIS 3xxx), Systems Analysis and Design (CIS 3301), Computer Hardware, Systems Software and Architecture (CIS 3303), Computer Networking and Data Communications (CIS 3305), and Database Management System (CIS 3306) courses, and an additional six hours of CIS upper-level elective courses in one of the CIS degree concentrations areas -- Electronic Commerce, Database Management, Application Development, Computer Networking, and Information Technology. Additionally, since graduates of the CIS degree program are expected to be able to design and implement computer systems, they are also required to take a three-hour upper-level Systems Development Project course (CIS 4312), called the Senior Project course, which is a capstone CIS course that combines materials covered in various CIS courses. Although most CIS courses involve hands-on exercises and group projects to ensure students receive full exposure to the Information Technology (IT) field and its practices, the entire focus of the Systems Development Project course is on hands-on practice where students are required to build and implement computer systems for local businesses, free of charge. These courses benefit all involved. Local businesses benefit from the students' computer knowledge and technical skills by gaining new or enhanced computer-based systems at no charge; the students benefit by gaining much needed real-world experience; and the supervising faculty member cultivates close ties with local businesses which may lead to publications and consulting opportunities. Thus, when projects are brought to a successful conclusion and business clients are satisfied with their gains, the entire community is in a contented, win-win state of mind.

3. THE SYSTEMS DEVELOPMENT PROJECT COURSE

The Systems Development Project course is a three-credit-hour, required senior-level CIS course that is usually taken during the last semester of the students' undergraduate CIS studies. To accommodate those students who are employed full-time, an online

section of this course is also offered during each semester in addition to the several face-to-face sections of this course. The purpose of this course is to provide the student with the opportunity to utilize previously learned CIS topics including the Systems Development Life Cycle (SDLC) methodologies and techniques through developing or enhancing computer-based business application systems for various local businesses. In addition to programming languages, students eligible to enroll in this course must have earned a grade of C or better in Systems Analysis and Design (CIS 3301), Computer Hardware Systems Software Architecture (CIS 3303), Computer Networking and Data Communications (CIS 3305), and Database Management System (CIS 3306) courses. Successful completion of CIS 3301 prepares students in the design of systems and provides them a clear model to use when developing new systems. They acquire knowledge of the internals of computers in CIS 3303. Because most businesses use networking, students must have a full understanding of the role that computer networks and data communications play in IT, which they acquire in CIS 3305. Finally, students gain database management system knowledge in CIS 3306. These required courses give students the foundation they need to tackle the complexity of the senior project. In addition to these prerequisites for taking the senior project course, students must also have senior standing and have successfully completed the Junior Writing Proficiency Exam. This assures instructors that students have the writing ability to take on the written reporting demands of the course.

4. SENIOR PROJECT CAPSTONE COURSE DESCRIPTION

CIS senior projects can take a variety of forms and complexities. They can involve single students or groups of students, and they can vary from enhancing existing systems to creating new systems. However, there are some shared aspects of all projects. First, the CIS project course requires students to review and use previously learned theories, skills, and technologies, learn new concepts and technologies, and apply the life cycle methodology to devise, develop, and deploy complete applications. Second, students learn to utilize project management tools and methods for resource scheduling and project control. Typically there are two oral presentations during

which students or student groups formally present their projects. Although students are encouraged to use a series of optional books including Information Technology Project Management; Systems Analysis and Design with UML Version 2.0 – An Object-Oriented Approach; and Needs Assessment and Project Planning, there is no required textbook for the course. Whether students are required to review, analyze, and evaluate existing systems with a focus on correcting problems or adding functionality, or instead to research, create and deploy a new system, they of necessity bring all of their CIS skills to the table and achieve an in-depth, intimate working knowledge of those theories they studied so diligently in past semesters.

5. COURSE ACTIVITIES

The course activities are divided into three major parts – prior to the beginning of the semester, during the semester, and post semester tasks.

Pre-semester Activities

Before the beginning of the semester, the supervising faculty member contacts students by sending them a copy of the preliminary course syllabus along with a *Welcome Letter* describing the course and types of appropriate and acceptable projects. The Project Proposal Specification states that only "real" projects for "real" users that have "real" needs are acceptable. Furthermore, it excludes projects for self, spouse, and other immediate family members. These requirements were established because in the past a few students had tried to build bogus systems that had no user and no use. The *Welcome Letter* serves as a reminder that as soon as possible students need to look for and secure at least two alternative projects that they may be interested in pursuing. Also before the beginning of the semester, the supervising faculty member contacts local businesses to obtain information on whether or not they may have suitable projects with the hope that these projects could be assigned to students who cannot secure acceptable projects.

Semester Activities

The second part of the course starts on the first day of class and consumes the rest of the semester (15 weeks) during which students engage in systems development life cycle activities. The first section of this part

is the completion of the Initiation Phase that started prior to the beginning of the semester. On the first day of class students are given a copy of the finalized syllabus, handouts on general information about course requirements, systems development life cycle review materials, and project timelines along with project proposal specifications. The students are then given a maximum of two weeks to secure projects and both prepare and turn in project proposals along with their resumes. At this stage, student proposals are informal and contain a brief description of the business and its needs and/or problems, a proposed solution and a preliminary list of required resources.

This is the first step towards the win-win scenario. Since students are required to update their resumes and contact businesses to secure projects, they are provided, by default, an opportunity to search the local IT job market, get familiar with the requirements of open positions, evaluate their strengths and weaknesses, and practice their job search and interviewing skills. These activities in essence prepare senior students for job search and give them time to refine their knowledge and skills to map into open IT positions before graduation. Furthermore, since local businesses are contacted for possible projects, they may use the opportunity to assess their IT needs, identify projects that would need the manpower not available to them otherwise, and assign them to students for implementation. The university, its degree program and its faculty benefit by having the indirect publicity. At this junction, the major challenge facing students and faculty is finding worthy projects that can wait until the beginning of the semester and which must be completed in less than 150 hours within a semester's timeframe. One solution could perhaps be soliciting small businesses and specifically looking for small projects; another would be to assign complex projects to groups of students. With much effort most students prevail and find suitable projects; however, a few students fail to find projects. So, as the project proposal due date approaches, the expertise of students who have not been able to secure projects and/or have asked to be assigned projects is matched with the needs of businesses secured by the faculty, and these students are accordingly assigned a project. In the past, while most project proposals for developing databases (e. g. client, employee) and web e-commerce sites

have been approved, proposals to build computers, install hardware or software, or networks have been generally rejected.

Once the Initiation Phase is completed and the project proposals have been approved, students are directed to proceed to the rest of the semester's assignment which includes four phases – Phase I: system concept development, requirements analysis, and system planning; Phase II: system design utilizing joint application development (JAD) and rapid application development (RRAD); Phase III: system development, integration, and testing; and Phase IV: system implementation and delivery.

Phase I Activities: Once project proposals are approved, students are given three weeks to complete Phase I activities and generate a comprehensive report along with associated diagrams and charts including an organizational chart and a high-level context diagram. During this phase, emphasis is placed on system planning to identify the scope of the problems and develop a plan of action for problem resolution by determining the user's needs, to perform an in-depth initial investigation, to carry out a feasibility analysis, to outline various possible solutions along with time and cost constraints, and to define system performance. The results of the thorough investigation are then compiled to generate a comprehensive report. Students are required to discuss this report and their findings with the business client and obtain approval to proceed with the project.

Although for the most part business clients approve the projects, a few businesses reject the preliminary report on the basis of lack of resources; as a result, occasionally a few students lose their projects in mid-stream and have to be assigned to other projects, some of which could come from the university community. Although this is a setback, with additional faculty support most students are able to make up for the time lost and complete a newly assigned project on time. In case students are unable to complete the project by the end of semester, they could be given incomplete grades to buy them time to continue their work on the project and complete it after the semester is ended. Although such an experience might appear negative, it, too, provides for a win-win opportunity. Here, in addition to learning from the system analysis step, students also experience first-hand that re-

quirements analysis may occasionally result in postponing system development. The business clients benefit from the exercise as the analysis step provides them a list of requirements that could be used at a later time.

Phase II Activities: Phase II activities begin only after the student has completed Phase I and has obtained user authorization and faculty approval to proceed with the project. The students are given four weeks to fully understand the proposed system requirements, perform detailed analysis of these requirements, further evaluate the alternatives, select the optimal system and system components, finalize a list of all needed resources such as expenses -- including computer hardware and software costs and server, host, and shopping cart costs -- and generate a report accordingly. To gain this full understanding, students are required to participate in a series of Joint Application Development (JAD) and Rapid Application Development (RAD) sessions. At the end of this phase, students are required to discuss the content of the report and their findings with the business client and obtain approval to proceed with the project.

Once again, occasionally a business that lacks funds to purchase the needed resources may decide to postpone or cancel the project, and students may be left without a project well into the middle of semester. So, once again, with the help of the faculty member they are assigned to other projects and, since so much time has been lost, most often the students are unable to complete the project by the end of semester and either drop the course or opt for a grade of *Incomplete* to buy time to complete the work. The non-approval of the project by the business clients at this stage is a problem. One solution could be to reject system requests from business clients who do not have a computer or the funds to purchase one at the onset. This would mean that the Initiation Phase requirements could be modified to include an inventory list of computer resources that the business client has to ensure that at least they own a computer and to notify the business client who does not have a computer system that a system must be purchased before students select the project.

Oral Presentation: At the end of Phase II, students are asked to make oral presentations of their work in Phase I and Phase II. They describe the problem they have uncovered, propose their solution and then stand through a critique and suggestion period from faculty and fellow students.

Phase III Activities: Phase III activities begin only after the completion of Phase II and with the user and the supervising faculty's authorization to proceed with the project. In this phase students are given three weeks to purchase, install and test the required hardware and software including server and host, and to develop and test the proposed system by coding and testing the individual processes/programs that make up the system.

While most students are fully capable of developing and testing the system and need very little supervision and guidance, a few students struggle with their own ineptitude in programming languages, too slavish a reliance upon materials from other websites, or use of software packages that they do not completely understand. Concerning coding, most students use Access, VB, or a combination of the two to develop database systems. Some students may lack an understanding of the code generated by Access or VB and be unable to develop code for functions such as search routines, the creation of switchboards or the hiding of Access windows, buttons and tools. One solution would be to include coverage of coding for Access in CIS 3306 or offer a lower-level programming course to exclusively cover Access and VB code. Concerning creativity and energetic e-com design, a few student projects may lack these necessary elements and fall short of an acceptable performance. Some students' web sites are merely copies of other web sites or links to other sites with little creativity and limited knowledge of forms and database connection. (Most forms are emailed to the business versus populating the business client's databases.) The instructor must find ways to motivate these students to approach the project with greater creative energy. Those students who choose to do e-com design would also benefit from a previously required course in e-com, but, if no such course is available, then the Project Specifications should clearly state that previous knowledge of e-com is required for any projects that involve e-com development.

Phase IV Activities: Phase IV activities begin only after the completion of Phase III and the procurement of user authorization to proceed with the project. By now all system components must have been tested individually and as a whole. In this phase, students are given three weeks to fully install the system, perform a thorough total system check, complete any necessary system fine-tuning, deliver the system to the business client, and conduct user training.

While, in the majority of cases, students deliver flawless systems for which the business clients are grateful, occasionally system flaws are not identified until well after the end of the semester, leaving the faculty members with a ticklish problem if an ongoing relationship with the local business is desired. The student is no longer available, the business expects the faculty member to solve the problem, and the faculty member generally does not have time to debug a system that should have been debugged prior to assigning final grades. At this point, the faculty member may have to fall back on reassignment of the project to a new student in the upcoming semester. A problem similar to system bugs also occurs with the need for simple maintenance and operation of the system. The student completes the project and disappears, but the business desires help with system operation and maintenance. Again, the faculty member does not have time for this activity and, at best, can only throw more student power from upcoming semesters at the problem.

Final Oral Presentation and System Demonstration: The last activity of the semester is the Project Presentation, which is scheduled after completion of Phase IV at the end of semester. Students orally present their semester project through system description and a live demo. Some of the users, fellow students, and faculty may be invited to the presentations.

The most common problem that occurs at this stage is file transfer problems during system installation. Some students must install and test their database system on a university machine in order to demonstrate their product. However, even when the system has worked at the business or on the student's computer, it may not function fully on a university computer and require extensive debugging before the demo can be

shown. Short of the faculty member going to each one of these businesses to verify that all is working, the demo is a necessary proof of functionality. Therefore, it is a step that cannot be missed. Thus, the students should be given a long period of access to the computer in the presentation room with sufficient warning that these problems do occur, or the university should provide laptops for students to install their projects on and demo them to faculty and audience. In addition, electronic copies of systems can be required so that the faculty member can check any that may require additional investigation.

Once the projects are turned in, the supervising faculty member checks the validity and functionality of projects by 1) calling the business client to verify that the system is in place and is functioning, 2) visiting the business client's site to check the actual system, or 3) installing the system on a university machine and testing it there.

Here lie additional issues worth noting. Depending on number of students, number of business clients, and number of projects, it could be time consuming to call every business client and almost impossible to visit each within the short time between the presentations and the last day to turn in grades. Clearly, the deadline for turning in grades cannot be changed, and scheduling the presentation date for an earlier date would not solve the problem as most students need the time to finish their work. The third alternative would also place a demand on the supervising faculty member's time as each project (at least database projects) must be installed and tested on the supervising faculty member's machine, graded, and then uninstalled. Unfortunately, short of the above-mentioned ways to check system validity and functionality, the supervising faculty member may have to rely on the documentation or system demonstration to accept the project. One other possible alternative is to identify problematic projects as soon as possible, pay special attention to them, work with students to resolve problems before final oral presentation, and use a multi-pronged approach to verifying their validity and functionality after the project is turned in.

Post-semester Activities

Once the semester is over and grades have been turned in, the supervising faculty

member contacts business clients by sending them a *Thank You* letter along with a short and simple survey to solicit information on the completed work and also to see if the client has other projects and is willing to go through the same exercise. Although the gathered information regarding the completed work cannot be used to evaluate student's work, the survey results could be used as a source for new projects for the upcoming semesters.

6. Discussion

As previously mentioned, UHD CIS students must complete 24 CIS required courses and must also complete six hours of upper-level CIS elective courses (see section 3, above). The upper-level courses are grouped so that students can concentrate in a CIS topic area by taking two courses (six credit hours) in one of the following topic areas: database management, e-commerce, application development, computer networking, or information technology. In practice, a student who chooses to concentrate in one of these areas would complete nine hours of studies in the chosen area and therefore would have a specialty in that topic area.

Although, by requiring the Systems Development capstone course, CIS students are forced to review and utilize previously learned CIS topics, many argue that this course should be offered as an upper-level elective course to give students who wish to further specialize in one of the aforementioned areas the option of using the three credit hours towards yet another advanced course in that topic area. They contend that materials covered in the Systems Development capstone course are merely a repetition of earlier materials and the hands-on practice should be deeply embedded in all previous courses. Furthermore, they argue that entry-level CIS positions seldom require experience in total systems development and therefore an additional course in systems development is not crucial to job placement nor of great benefit in the early years of employment as, by the time CIS graduates are in positions requiring total systems development, they have gained additional knowledge and have had ample on-the-job training and experience to perform the required tasks. They reason that by allowing students to take another advanced course in a topic area, students would be further qualified for job entry positions in the topic areas. There needs to be a study as

well as a full discussion of this issue amongst the discipline to validate either position.

Another issue that needs further discussion is what the prerequisite to the Systems Development capstone course should be. Are the previously stated CIS core requirements sufficient prerequisites? Are students prepared to tackle any type of project? For example, should a student who has not taken any courses in web development be allowed to work on a web development project? Similarly, should a student who has taken only one course in data base management be allowed to work on developing a system using data bases? Where would these students learn the additional knowledge necessary to complete their projects? In many cases, in addition to supervising many projects, the supervising faculty member must also take on the responsibility of teaching individual students the needed materials not covered in their previous courses and partake in developing these systems. Clearly one option would be to require that students solicit types of projects that they have expertise in, but this may be an unreasonable requirement as taking one course in an area does not make the student an expert in that area.

Moreover, as mentioned earlier, if a project is complex and requires more than 150 hours, it is usually assigned to a group of students with complementing expertise. Since group cohesiveness and group dynamics play major roles in successful completion of the work, group composition becomes a major challenge for the supervising faculty member. In most cases, the supervising faculty member can only rely on students' GPAs when assigning them to groups. Usually the student with the highest GPA is assigned the role of project manager and is asked to divide the tasks and delegate them to group members. If all members of the group participate and deliver high quality work on time, the group succeeds and is able to develop the system with minimum supervision. On the other hand, if a member falls behind or cannot deliver high quality work, the rest of the group may not report the misconduct, but instead choose to complete the weak student's work rather than sacrificing their grades. Unfortunately most times such misconduct is not reported and the transgressor completes the course undeservedly. This raises the question of the value of group projects. The authors

contend that the gained experience from group projects is valuable only when a group is functional; and, unless students are assigned to groups with more than their GPAs in mind and individual gradable components are designed into the course, group projects should be avoided.

Finally, since the purpose of the Systems Development capstone course is to provide the student with the opportunity to utilize previously learned CIS topics, other than brief reviews of the systems development life cycle, no new material is introduced in this course. While some argue that students should be given most of the 45 contact hours to work on their projects, others contend that contact hours should also be used for teaching new materials such as project management concepts and practices, as well as topics that students have problems with including programming or web design. There are pros and cons for each school of thought. On the one hand, using class time to teach new materials would enhance students' CIS knowledge and resolve a few of the abovementioned problems (see Section 5). On the other hand, replacing class time during which students work on their projects with lectures on new topics that would require additional study time would decrease total available time to work on projects and increase study time, resulting in an unreasonable and extensively time consuming course that could result in student failure. A healthy debate of this issue amongst the CIS faculty could result in a paradigm that would benefit all.

7. Conclusion

The authors' many years of experience teaching the Systems Development capstone course have been by far positive and rewarding. During the authors' tenures at UHD, students have deployed many well-designed and functional systems for local businesses. In fact, many happy "repeat" business clients are eager to start new projects, and some have hired former students. Many CIS alumni have reported that their prospective employers were so impressed with their projects that, in addition to offering them jobs, they have also shown interest in partaking in the adventure. On a number of occasions, many of the business clients have approached the supervising faculty member for help with their IT problems and have offered them short-term and part-time consulting positions. In summary, the busi-

ness clients win by achieving some of their computerization goals at no charge, the students win by gaining the needed IT experience and securing jobs, and the supervising faculty member wins by gaining support of the local community. Everyone benefits from the Systems Development capstone course joint venture, resulting in a win-win state of affairs, indeed.

8. References

- Anonymous, Systems Development Life Cycle Guidance Document, The Department of Justice, January 2003, <http://www.usdoj.gov/jmd/irm/lifecycle/ch1.htm#para1.2.1>
- Anonymous, RAD: Ganttthead.com the community for IT project managers: <http://www.ganttthead.com/process/processMain.cfm?ID=2-19516-2>.
- Anonymous, Project Management Institute: <http://www.pmi.org/info/default.asp>.
- Gorgone, John T., Gordon B. Davis, Joseph S. Valacich, Heikki Topi, David L. Feinstein, Herbert E. Longenecker, Jr. (2002), "IS 2002: Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems" *The DATABASE for Advances in Information Systems*, Vol. 33:3.
- Janicki, T. N., D. Kline, J. A. Gowan, & R. Konopaske (2004) "Matching employer needs with IS curriculum: An exploratory study" *Information Systems Education Journal*, 2(21).
- Kay, Russell, (2002) "QuickStudy: System Development Life Cycle" *Computerworld*, MAY 14 <http://www.computerworld.com/developmenttop-ics/development/story/0,10801,71151,00.html>.
- Wood, J. and D. Silver (1995) *Joint Application Development*, 2nd ed., New York: Wiley.
- Woratschek, C. R. & T. L. Lenox (2002) "Information Systems Entry-level Job Skills: A Survey of Employers." *Proceedings of ISECON 2002*, October 31 - November 3.