Empowering Students to Actively Learn Systems Analysis and Design: The Success of an Entrepreneurial-Inspired Project in a Hybrid Learning Environment

Wang-chan Wong
wcwong@csudh.edu
Department of Information Systems and Operations Management
California State University, Dominguez Hills
Carson, CA 90747, USA

Abstract

Systems Analysis and Design (SA&D) is the cornerstone course of a traditional information system curriculum. Conventionally, it is a sequence of two courses with the second course dedicated to the completion of a project. However, it has recently become more common to reduce the two-course sequence into one, especially for IS departments that are only concentrations of a business school and not independent departments. Not only has the original sequence been reduced to one course, but the course is also offered to non-technical business students. It is challenging to design a successful SA&D course that is subject to these constraints. In this article, the author showcases a modified SA&D hybrid course enriched by a real-life entrepreneurial-inspired project with a double review development process model. A survey conducted at the conclusion of the course shows that the new approach works well and holds great promise for improving future SA&D courses.

Keywords: Systems Analysis and Design, active learning, hybrid, entrepreneurship, double review process

1. INTRODUCTION

Many studies show that teaching Systems Analysis and Design (SA&D) has become increasingly difficult (Burns, 2011; Burns, 2012; Chen, 2006). For example, in the author's affiliated department, not only was information systems (IS) curriculum expanded to include business intelligence, data analytics, and project management, but the formerly two-course SA&D sequence was also reduced into a single course. The author was thus challenged to design a single project-based SA&D course that would not only cover both theoretical and conceptual topics, but also accommodate non-technical business students. The only prerequisite for the new SA&D course was an Introduction to Information Systems course.

To this end, a new structure was developed based on hybrid learning with the incorporation of a real-life project. Simply put, the traditional series of SA&D courses was implemented as one course: theoretical and concept-based materials were covered online while face-to-face sessions concentrated on the project. The entrepreneurial-inspired project created by the author was structured to follow the online materials. In addition, the author acted as a mentor, investor, and technical advisor to the project teams. When the 16-week course ended, a survey was conducted, and judging from student performance, assessments, and survey comments, it became clear that the entrepreneurial-inspired project was pivotal to the entire course, and greatly excited and motivated students. Overall, results to the new approach are very encouraging and show definite promise. In this paper, the author shares his experiences so other instructors may consider using a similar approach.

The paper is organized as follows: Section 2 elaborates the current challenges SA&D instructors face. A detailed review of the hybrid learning model is given in Section 3. Section 4 discusses the importance of real-world projects in learning SA&D. The modified course structure is elaborated in Section 5. The survey results are summarized and discussed in Section 6. Final remarks are given in Section 7.

2. THE CHALLENGE

Several trends have had a drastic impact on teaching SA&D. First of all, IS departments have been opening SA&D classes to non-technical business majors. They argue that the magnitude of money spent on developing business information systems needs close monitoring. According to one study, companies on average spend up to 5% of their total revenue on systems development and maintenance (Carmen & Tija, 2005). IT has become an enabling technology for functional business units, and managers should have at least a minimal understanding of technologies in order to develop internal scorecards for assessing performance and to source strategies to minimize costs. Therefore, managers must have a basic understanding of the system development process to fully utilize the enormous amounts spent on development and maintenance. If managers are involved in the process of system designs and operations, they will be in the position to develop other business initiatives such as business process re-engineering and social media analytics. A strong and effective relationship between functional business units is a determinant of success in gaining business advantage through IT (Keen 1999; Reich and Benbasat 2000). Mature and system development-savvy managers can be assets during the development process. The 21st century business model is very different from before: market volatility is high because consumer tastes change quickly and the real-time global economy brings competition from around the world. Consequently, both product development and system development cycles have been dramatically shortened. When managers see opportunities to respond to market changes, new systems and applications must be developed quickly in order to seize these opportunities. Technical teams need to work closely with business units and improve communication and encourage the exchange of ideas. It makes sense to train business people in systems analysis and design.

Another trend is the change of SA&D curriculum. Typically, SA&D is a sequence of two undergraduate courses: a theoretical and conceptual introduction to SA&D, followed by a project course. Some schools offer only one course in SA&D but require a capstone project class, essentially following the same two-course model. However, Burns (2011) found that the one course, one semester delivery approach becomes much more common than the two class approach.

Within the author's affiliated college, the two-course undergraduate SA&D sequence has been reduced to simply one course with no capstone project class. SA&D is also being offered as an elective to other business students while still being required for both IS major and minor students. The only prerequisite for SA&D is a general course on IS theories and practices.

Consequently, the author has had to restructure the SA&D sequence into a single course that serves the dual purpose of accommodating business students while also furthering the development of students wanting to pursue careers as systems analysts.

3. HYBRID LEARNING

The main dilemma in having a single course replace a traditional two-course series is that the course now requires a different delivery modality. To accomplish these goals, the author adopted a hybrid modality; a mix of traditional face-to-face and online learning modes. Hybrid learning has been praised as having the best of both worlds. It is also adopted in teaching SA&D. For instance, Bain shows that a hybrid course delivery can produce similar if not better results than traditional delivery methods (Bain, 2012). Tanner and Scott report how a flipped classroom approach actually helps to teach SA&D (Tanner and Scott, 2015). In the flipped classroom approach, students are expected to learn theoretical concepts outside of the classroom setting (e.g. online), and are given the opportunity to apply these concepts in a face-to-face class with the instructor and other students. Griffiths et al. report the success of a lecture-free approach in a hybrid graduate course on SA&D with learning materials and design tools online supplemented by a weekly, one-hour lab-based practical session (Griffiths, et al., 2003).

These studies show a certain degree of success in using a hybrid delivery to teach SA&D, but do not offer any frameworks. A detailed literature review on variations of the hybrid delivery model is given by (Dana 2007). The models being reviewed identify the components and their roles and relationships to each other in the hybrid setting.

For example, Kerres and De Witt (2003) suggest a basic hybrid model combining online and face-to-face meetings based on their 3C model: content, communication, and construction. Kitchenham (2005) identifies three major components of successful hybrid courses: collaboration, strong infrastructure, and student demand. Schatzberg (2002) reports how Bloom and Kolb's experimental learning could potentially be used to teach systems analysis and design. Barnum and Paarmann (2002) identify four components of a hybrid model: web-based delivery, face-to-face processing, creation of deliverables, and collaborative extension of learning. The first component refers to a typical online learning module where students can access necessary learning materials, discussion forums, message exchanges, etc. face-to-face meetings help students have a more comprehensive understanding of materials. Having conceptual knowledge is not enough; through the creation of a "tangible" deliverable, students undergo the process of constructing knowledge and demonstrating their understanding. A collaborative extension of learning encourages students to share their learning experiences, information, and resources in their own ways, whether they be online or offline.

4. IMPORTANCE OF PROJECT TO SA&D

Many of us may have taken it for granted that projects are vital to learning SA&D. The author agrees with Burns that IS is an applied science similar to medicine and engineering (Burns, 2011). SA&D is fundamentally an engineering discipline, where engineering principles are applied by developing information systems to solve business problems.

There are many studies that attest to the benefits of having an SA&D curriculum build towards a project. Chen (2006) argues that a real-world project is better than a simulated project. Helwig (2006) also suggests using a real system development project to enrich SA&D coursework.

There are many ways to incorporate real-world projects to enrich SA&D curriculum. Some instructors might instruct groups to work on different client systems. Logistically, it is difficult for each team to work on a separate client; in this case, students must deliver professional results otherwise it will be difficult to attract other companies for future collaboration. In order to be successful, the project has to leverage on the instructor: the instructor has to mediate and participate in meetings between clients and students. Another method would be to bring in one client project and have each student group work on the project and compete with each other, as Helwig suggests (2006). Harris (2009) also argues that a competitive project method would benefit students more than a stand-alone project.

5. THE MODIFIED COURSE STRUCTURE

After reviewing previous studies on teaching SA&D as discussed above, the Barnum and Paarmann (2002) hybrid model was adopted. The course lasted for 16 weeks, and learning materials were accessible online on Blackboard, including lecture notes, videos produced by the author and from other learning sources, discussion forums, individual chapter assignments, and chapter quizzes. The online portion was essentially identical to a full online SA&D course with both asynchronous learning and interactivities. However, a weekly 1.5 hour face-to-face meeting was mandatory. Since the project was the backbone of the course that required students to create tangible deliverables, students had to collaborate both online and offline. In the face-to-face meetings, the author spent time addressing technical and management issues. The class was given a single project, creating a natural competitive environment among teams. For each deliverable, the author randomly selected the work of one team and walked through the document in class, offering comments and critiques and inviting the class to be actively involved in the open review process.

Class teamwork has inherent management problems; major disputes such as non-performing team members and disagreements among team members regarding concepts and directions were handled separately by appointment or during office hours with the author. Team restructuring did occur during the first phase of the project.

In order to accommodate non-technical business students, the breadth of topics and depth of coverage needed to be changed. The IS 2010 curriculum has been published for many years (Topi, et al, 2010). The new curriculum suggests seven core courses, including a course on Systems Analysis and Design (IS 2010.6). The Task Force also suggests a list of topics to be covered in the SA&D course. A graphical comparison between the topics in IS 2010.6 and the selected textbook by Shelly (Shelly and Rosenblatt, 2010) is illustrated in Appendix 2. The IS 2010.6 curriculum guideline has replaced the technical skills of functional and object-oriented design approaches with business process management. These missing topics closely mirror the actual coding of the system. Wong shows that

students without programming exposure suffer and underperform in learning SA&D if the course's coverage was too technically oriented, for example, by focusing on functional or object-oriented approaches and their skills (Wong, 2015). On the other hand, business process management is not programming-driven, and is capable of capturing business behavior and logic without any technical programming training. Students without programming backgrounds can definitely benefit from it. Conceivably, the new curriculum can accommodate non-technical business students wanting to understand how systems are developed. In this aspect, the new guideline actually fits the current requirements well. However, most SA&D textbooks, including Shelley's, do not cover business process modeling (BPM) in detail. To supplement the text, the author created a complete module with lecture notes, readings, and videos on BPM, including advanced concepts such as business process re-engineering (BPR) and activity-based costing (ABC). In fact, the BPM module also included extensive discussion on the patentability of business processes, and several exemplary patents, including Amazon's 1-click checkout process, were discussed. The inclusion of relevant, popular examples made the students more enthusiastic and willing to join discussions.

Another significant supplement to Shelley's textbook included an emphasis on writing functional and non-functional software specs based on the clausal form, the use case analysis that starts with the use case diagram, and detailed use case descriptions using a standard industrial template.

In addition to the team project and online materials, students had individual homework assignments such as creating Gantt charts for scheduling, and computing NPV and ROI to determine project feasibility. Students were exposed to a variety of subject areas, and were assessed through chapter quizzes and a final exam.

5.1 The Project

As we have seen from prior studies, the project is an integral part of learning SA&D. It is preferable to have a competitive project rather than multiple standalone projects. However, projects taken from textbooks are not effective; for one thing, most textbooks have "solutions" posted online, and students can easily complete projects simply by searching for answers. Since the course was also being offered to non-technical business students, the project was designed to only complete the analysis and design phases without going to implementation.

To entice students, the author used a new strategy: instead of bringing in a client's project to the classroom, the author created a "startup" company that would offer a mobile grocery shopping App called "**B4U"** based on an online-to-offline (O2O) model similar to Uber. A high-level business narrative was given and explained to students in the first face-to-face meeting (see Appendix 3). Students were grouped into teams of 3 to 4 people to form their own startup. As seen in the narrative, they needed to fill in many gaps, for instance, the business model, the payment methods and alternatives, etc. Based on student feedback, the first face-to-face meeting was very inspirational and motivated them to be entrepreneurial. They brainstormed the features and processes of the App with each other, and interviewed dorm-mates, friends, and family members for additional requirements. They were motivated because they believed the project was real and attainable. The author considers the project to be entrepreneurial even though it took place in a classroom setting, because students were encouraged to go beyond the course requirements and approach it as a serious startup project. Several student groups did go beyond the course requirements. They created business plans and funding proposals with assistance and guidance provided by the author outside of class. The author observed that the teams radiated a team spirit similar to that in a fast-paced, high-energy startup.

There were three other considerations in deciding the focus of the project. Firstly, as a practitioner and SA&D educator, the author values the importance of system requirements. There is significant evidence demonstrating that requirements and the management of requirement changes can make or break a project. Both practitioners and educators commonly agree on this belief. Misic and Russo (1999) report the differences between topics prioritized by SA&D practitioners and educators. Both sides agree that defining requirements and the scope of project are the top two tasks that should be taught to SA&D students.

Secondly, a requirements document is essentially a contract between the development team and the system procurer. The author treated the documentation as a writing-intensive exercise similar to the one proposed by (Pomykalski 2006). The difference is that Pomykalski used case studies for students to practice on, but the author used a complete project requirements document

that was written, reviewed, revised incrementally and iteratively.

Thirdly, the author incorporated industry best practices into the project by practicing incremental and iterative development and a rigorous double review process. A major concept in system development such as validation and verification (V&V) may be too abstract for students. The question in system validation: "Are we developing the right system?" can only be answered by the review process. For instance, in developing their software functional specs, one team misunderstood the nature of the project (perhaps they were influenced by examples of grocery shopping Apps found online). However, their review team realized the specs were not an O2O model but a conventional B2C model between grocery markets to consumers. The error was caught early in the development cycle and was corrected before moving forward.

The double review process was conducted as follows: during face-to-face meetings, the author randomly selected a work-in-process deliverable from a team. The author pointed out deficiencies or errors in this document and explained to students how to review and critique the deliverables themselves. The author acted as a mentor, coaching them through their mistakes. Since they were working on the same project, this type of hands-on mentoring and open review bridged the gap between learning concepts and actually applying them.

The deliverable was then assigned to another team to review (the assigned review teams remained the same throughout the project, and project teams were also encouraged to interact with their reviewers.) Once the student reviews were finished, the author reviewed the original deliverable in addition to the comments of the review team.
To enforce the rigor of the review process, two separate scores were given for each deliverable, one to the team that created the deliverable, and one to the review team. After the original team received comments from both their review team and the author, they had to revise the document.

Because deliverables were out of sync with each phase, students truly understood firsthand why the waterfall process model wouldn't work and why incremental and iterative processes were desirable. The review process also focused on consistency and coherency from deliverable to deliverable.

Face-to-face meetings primarily focused on the project and its relationship to textbook concepts, but they also helped clarify problems that students had with the online learning materials. Furthermore, the author was able to demonstrate how to use software tools in class.

The project also became a training ground for students wanting to become project managers. Students were explicitly encouraged to rotate the role of project manager for each deliverable. Project managers had a chance to practice their skills on scheduling, team management (personality conflicts, non-performance, etc.) There was a peer evaluation for each deliverable, and members would evaluate each other on their contributions to the deliverable. At the end of the semester, a blind peer evaluation was conducted, and evaluations had a direct impact on grades.

Students were given a document template that had the following mandatory sections. They could add or expand from the template.

Section 1. Executive Summary	Section 6. Use Case Diagram
Section 2. Business Case and SWOT Analysis	Section 7. Use Case Descriptions
Section 3. High Level Requirements	Section 8. Data Model
Section 4. System Functional Specifications	Section 9. Future Provision
Section 5. Constraints and Non-Functional Specifications	Section 10. References

The first sections form the base of the business plan, in which teams presented their ideas as the startup company offering the B4U app. They needed to create a business case and a SWOT analysis to support the reason for the startup. The author was impressed by several teams that actually created competitive analyses along with three-year cash flow analyses to justify the investments.

Sections 3 through 10 comprise the requirements document that focused on conveying the business requirements to the development team. Both high-level and system functional specs were written in clausal form for precision. In the Use Case Descriptions section, a use case description table template was given to students. For each use case description, a set of user interface screens or reports would follow if the use case needed to interface with people. It was then followed by a business process model capturing the business logic and control flow of the use case. Section 7 forms the bulk of the entire document.

The Phase 1 deliverable was the write-up of sections 1 to 5. Phase 2's deliverable focused on sections 6 to 7, plus the revision of the previous sections based on the double reviews. However, only the use case description tables were required without the UIs and the business process model at this point. Phase 3's deliverable was essentially a revised Phase 2 deliverable with UIs. The Phase 4 deliverable was the revised Phase 3 product with the addition of the business process model implemented in a multifunctional flowchart. Unfortunately, the double review cycle was time consuming, and did not give the class sufficient time to complete the data model for the project.

Note that for each review cycle, the entire document was reviewed for consistency and coherence. A detail evaluation form was given for each deliverable review. When the review was returned, both the team and reviewers would see the comments. The double reviews also helped the reviewers; they would know if their comments were correct and appropriate and whether they had missed out on other issues, and so on.

Students used mostly Microsoft tools, such as Visio to draw the UML use case diagram and the multifunctional flowchart for the business process model, Powerpoint's Storyboarding add-on for user interface design, and MS Project to create the Gantt scheduling diagram. There were both text-based and video-based tutorials on using these software tools in the online modules. The author also demonstrated them in the face-to-face meetings, mostly focusing on the project itself as the example.

6. THE SURVEY RESULTS AND DISCUSSION

The course was offered in two sessions with a total of 47 students. With 3-4 students per team, there were 15 teams in total. Students filled out a survey at the end of the course, and the results are summarized and discussed in this section.

6.1 Student Profiles

The student profile is tabulated in Table 1 in Appendix 1. Among the 47 students, 64% (30) were seniors and 64% (30) claimed either they had taken programming classes or learned programming on their own. 70% of them took the class because it was a requirement for their major or minor. The other 30% took the course because it was listed as an elective, or they had personal interest in learning SA&D.

Only 51% considered themselves as IT-savvy even though 64% claimed they had programming experience. (Note that the percentage henceforth is the sum of the Agree and Strongly Agree percentage of responses unless stated otherwise.)

Prior to taking this course, 57% of students thought application development was simply writing code. This furthers the argument that SA&D should be offered to non-technical business students so they can learn and appreciate the complexity of system development.

40% of students believed that the class gave them more confidence to pursue a career in SA&D, while 34% were indifferent. These indifferent students may not have considered pursuing careers as system analysts anyway. 85% of the students believed that the class did help them understand SA&D and only 26% of them thought learning SA&D was difficult. Overall, the new approach worked well.

While project-related work accounted for 52% of the weighted total, the remaining 48% consisted of individual homework assignments, chapter quizzes, and the final exam. To determine any difference in performance between students with programming experience and those without, two independent t-tests were conducted. The first t-test was based on students' weighted totals that included group project scores of deliverables for each phase (including review reports), as well as individual scores for assignments, chapter quizzes, and the final exam. The second t-test was based solely on the weighted total of students' individual work, including four homework assignments, nine chapter quizzes, and one final exam. The results are shown in Table 2, Appendix 1. As seen in t-test 1, the mean scores of the weighted total between students with programming and without are 80.70 and 79.08, respectively, and the difference is not significant. In the second t-test, the mean scores of students' individual work are 20.83 and 19.80. Again, the difference between them is not significant. Wong reports that, in SA&D classes covering technical topics such as dataflow diagram, etc., students with prior programming experience perform significantly better than those without (Wong, 2015). The current study is based on the IS 2010.6 guideline that eliminates both functional and object-oriented approaches. Instead, the guideline suggests adding business process modeling that does not require much prior programming exposure. The t-tests in this study confirm that the performance gap no longer exists.

6.2 Course Delivery Modality

Students were also asked about their opinions on the course delivery modality. Responses are

summarized in Table 3 in Appendix 1. 68% of students preferred the hybrid format to online classes. Only 40% of them preferred a hybrid format to face-to-face classes, while 26% of them were indifferent. Unexpectedly, only 28% of them thought it would be a good idea to split the class into two courses. It could be because 57% claimed they learned the online materials effectively and only 30% of them thought they needed more face time. During the course, a lingering question for the author was whether the online material workload was too overwhelming, since almost two courses were combined into one. However, 60% of the students said the workload was about right; 28% thought the materials were excessive, while 13% of them demanded even more material. Generally speaking, the hybrid modality was well-received.

6.3 The Project

The project was the main component of the entire course, and the author was anxious to find out what the students thought about the project and the double review process. The survey questions regarding the project are summarized in Table 4 in Appendix 1. The responses are listed in the descending order. Surprisingly, a majority (83%) of students agreed that working as reviewers helped them not only on their project but also in understanding SA&D. 81% of them also agreed that the project helped them practice project management. 79% of them thought that the project was realistic and relevant, with 21% being indifferent and with no one disagreeing that the project was realistic and relevant. This is a significant affirmation of the project's effort. Similarly, 79% agreed the project helped them understand the online learning materials. This is again a significant confirmation of structuring project progress in parallel with the online learning materials. 70% of students agreed that they had positive team experiences, and 70% of them even claimed they would hire their team members in the future. 64% of them believed their team members were technically competent. 64% agreed that reviewer feedback was helpful and only 28% of them thought their reviewers were not qualified to review their document. Interestingly enough, 68% of them admitted to spending more time on the project than studying online materials.

The results affirm the idea that a realistic, and in this case, entrepreneurial, project can highly motivate students. Working on the project did enrich their active learning of SA&D concepts.

6.4 Topics and Other Issues

The next set of questions in the survey was designed to elicit responses from students regarding topics covered in this course, and other issues and concerns they might have.

The author had considered an alternative approach in that perhaps the project could have begun by designing the user interface first. Students were asked in the survey if they agreed that designing the UI first would help them in doing the project. It turns out 49% agreed, while 32% were indifferent. However, in working with students closely, the author noticed that visualizing the end product definitely helped them connect the dots.

When students were asked if they found the software tools difficult to use, only 17% of them said the tools were difficult.

Students were also asked what topics were confusing and difficult, and which topics they would like to see covered more deeply. The responses are summarized in Table 5 in Appendix 1. The rankings for confusing and difficult topics are almost identical. The author was surprised to see that software specs were the top concern. It might have been confusing and difficult to students because they needed to discover, collect, validate, and organize the requirements, and they might be unfamiliar with the technical writing style of the specs in clausal form.

It is interesting to see that 36% of them wanted to learn more about project management. This suggests that the efforts they made while producing this complex assignment showed them the importance of project management.

7. CONCLUDING REMARKS

The new SA&D curriculum, the entrepreneurial project, and the Barnum and Paarmann hybrid model seem to be successful. Another indication of success is that many students chose to include the project document in their job interview portfolios. One may argue that the heavily-documented approach taken in this class does not truly reflect the preferred agile development process typically adopted in startups. However, the author believes that once the students experience such a heavily-documented process, it will be easier for them to transition to an agile process.

There are several takeaways from this report. Firstly, the online module is very similar to other SA&D online programs; the important differentiator is the startup-based project chosen for this course. It instilled a sense of authenticity in the work and fostered open competition among

the teams; in fact, at the end of the course a few teams were working on a full business proposal to raise funds. Secondly, the double review process is vital. As indicated in the survey results, students really appreciated this process, particularly the opportunity to be reviewers themselves.

There are several concerns and issues with the new structure. The downside of this approach is the sustainability of finding interesting projects to motivate students. In the Internet age, it is almost mandatory not to re-use previous projects taken from textbooks, because a simple search will easily reveal the answers. Another weakness is the double review process; while it is extremely useful, it is also time consuming. The review cycle for each deliverable took roughly two weeks to complete, and the class did not finish the data modeling portion of the project because they ran out of time. The author plans to improve this process by using Google Docs instead of printing out documents for review.

Finally, SA&D is, in fact, an applied discipline, and the hands-on mentoring approach helps students see the relevance of learning materials and teaches them how to solve real-world problems. However, it is heavily dependent on the instructor's capability. As shown in (Burns 2012), the industry experience of an instructor does affect the purpose and content of the SA&D course. Clinebell and Clinebell report the centuries-old contention between academic rigor and the relevancy of real-world education (Clinebell and Clinebell 2008). Simply put, the current structure may not be appropriate if an instructor does not have the necessary industry experience.

8. REFERENCES

Bain , Lisa Z. (2012) Behind the Final Grade in Hybrid v. Traditional Courses: Comparing Student Performance by Assessment Type, Core Competency, and Course Objective, Information Systems Education Journal (ISEDJ) 10 (1), February 2012.

Barnum, C., & Paarmann, W. (2002). Bringing induction to the teacher: A blended learning model. THE Journal. Retrieved June 21, 2016. Retrieved from: https://thejournal.com/Articles/2002/09/01/Bringing-Induction-to-the-Teacher-A-Blended-Learning-Model.aspx

Burns, T. J. (2011). Defining the Content of the Undergraduate Systems Analysis and Design Course as Measured by a Survey of Authors. Information Systems Education Journal, 9(5), 4-17.

Burns, T. J. (2012). Does the Author's Experience as a Practitioner Affect the Purpose and Content of the Undergraduate Systems Analysis and Design Course? Information Systems Education Journal, Volume 10, Issue 1, February 2012.

Carmen, Erran and Tija, Paul. (2005). Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce 1st Edition, Cambridge University Press, 2005.

Chen, Brady. (2006). Teaching Systems Analysis and Design: Bringing the Real World into the Classroom. ISEDJ, Volume 4, September 27, 2006.

Clinebell, S.K. and Clinebell, J.M. (2008) "The Tension in Business Education Between Academic Rigor and Real-World Relevance: The Role of Executive Professors", Academy of Management Learning & Education, 7 (1), 99-107.

Gray, Dana, (2007). Uses and Perceptions of Online Learning Components in Hybrid Courses by Full-Time Business Authors at Comprehensive and Regional Public Universities in Oklahoma, Ph.D dissertation, Graduate College of the Oklahoma State University, July, 2007.

Griffiths, Gary and Oates, Briony J (2012) Lecture-free Teaching for Systems Analysis: An Action Research Study, Information Science, InSITE, June, 2003.

Harris, Ranida. (2009). A Systems Analysis and Design Semester Project: A Stand-alone Project vs. a Competitive Project. ISEDJ, Volume 7, Number 11, March 23, 2009

Helwig, Janet. (2006). Using a "Real" Systems Development Project to Enrich a Systems Analysis and Design Course. ISEDJ, Volume 4, Number 62, August 24, 2006

Keen, P.G.W. (1999)."Middle-Out Ideas," Computerworld (56), April 12, 1999.

Kerres, M., & De Witt, C. (2003). A didactical framework for the design of blended learning arrangements. Journal of Educational Media. 28 (2-3), 101-113.

Kitchenham, A. (2005). Adult learning principles, technology and elementary teachers and their students: The perfect blend? *Education, Communication & Information*, 5(3), 285-302.

Kolb, D. A. (1984). Experiential Learning: Experience as the Source of Learning & Development. Englewood Cliffs, NJ, Prentice-Hall.

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

Pomykalski, James J. (2006). Teaching Systems Analysis and Design as a Writing-Intensive Course. ISEDJ, Volume 4, September 6, 2006

Reich, B. H., and Benbasat, I. "Factors that Influence the Social Dimension of Alignment between Business and Information Technology Objectives," MIS Quarterly (24:1), March 2000, pp. 81-111.

Schatzberg, L. (2002). Applying Bloom's and Kolb's theories to teaching systems analysis and design. In The Proceedings of ISECON (Vol. 19).

Shelly and Rosenblatt (2010). Systems Analysis and Design, Eighth Edition, Video Enhanced, Course Technology, 2010.

Tanner, M., & Scott, E. (2015). A flipped classroom approach to teaching systems analysis, design and implementation. Journal of Information Technology Education: Research, 14, 219-241. Retrieved from http://www.jite.org/documents/Vol14/JITEv14ResearchP219-241Tanner1840.pdf

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K. M., Nunamaker, Jr., J.F., Sipior, J.C., and de Vreede, G.J. (2010). IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems, ACM and AIS, 2010.

Wong, Wang-chan. (2015). "The Impact of Programming Experience on Successfully Learning Systems Analysis and Design", Information Systems Education Journal (ISEDJ), Volume 13 - Number 3, Pages 15-23, May, 2015

Appendix 1
Survey Results

Status	Has Prog	No Prog	Total	
Junior	12	5	17	36%
Senior	18	12	30	64%
Total	30	17	47	
	64%	36%		

What was the reason for taking the class?	Has Prog	No Prog	Total	
Required course for Major	11	6	17	
Required course for Minor	13	3	16	70%
Elective course for Major	1	6	7	
Interested in exploring SA&D	5	2	7	30%
Total	30	17	47	

Consider self as IT savvy	51%
Simply writing code	57%
Class helps understanding of SA&D	85%

t-test 1 on weighted total of team project scores and individual assignment and test scores		
	Mean	SD
No Programming	79.08	8.60
Has Programming	80.70	12.24
t = -.48, df = 45, p = .63		

t-test 2 on weighted total of individual assignment and test scores only		
	Mean	SD
No Programming	19.80	3.00
Has Programming	20.83	3.16
	t = -1.05, df = 45, p = .30	

Table 2 – Independent Group t-test Results

Prefer hybrid to online classes	68%
Prefer hybrid to face-to-face classes	40%
The class should be split into two classes	28%
Can learn effectively and efficiently online	57%
More face time will be better	30%

The workload of the online materials is:	Has Prog	No Prog	Total		
Too much	4	1	5	11%	
More than other hybrid classes	3	5	8	17%	28%
About right	19	9	28	60%	60%
Need more materials	3	1	4	9%	
Definitely need more materials	1	1	2	4%	13%

Working as a reviewer improved their understanding	83%
Project helped them understand project management	81%
Project was realistic and relevant	79%
Project helped them understand online learning materials	79%
Positive team experience	70%
Would hire team members in the future	70%
Spent more time on project than online learning materials	68%
Reviewer feedback was very helpful	64%
Team members were technically competent	64%
Reviewers were not qualified to review	28%

Table 4: Project and Related Issue Responses

Confusing Topics	Total	
Software Specs	11	23%
UC Descriptions	10	21%
BPM	9	19%
ERD	8	17%
UC Diagram	7	15%
UI	2	4%

Difficult Topics	Total	
Software Specs	12	26%
UC Descriptions	10	21%
BPM	9	19%
UC Diagram	7	15%
ERD	5	11%
UI	4	9%

Want to Know More	Total	
Project Management	17	36%
ERD	10	21%
Software Specs	8	17%
BPM	8	17%
UI	4	9%

Table 5: Topic Issues

Appendix 2

Comparison of the topics proposed by the IS 2010 curriculum guideline with the traditional Systems Analysis and Design exemplified by the textbook of Shelly and Rosenblatt (2010).

Topic Categories Suggested by IS 2010 -- 2010.6	**Table of Contents of the textbook by Shelly et al.**
Systems A & D Philosophies and Approaches, e.g. SDLC, UP, UML RAD, Agile, etc	1. Introduction to Systems Analysis and Design
Identification of Opportunities for IT-enabled Organizational Change	2. Analyzing the Business Case
Business Process Management	3. Managing Systems Projects
Analysis of Business Requirements	4. Requirements Modeling
Different Approaches to Implementing Information Systems	5. Data and Process Modeling
Specifying Implementation Alternatives for Specific Systems	6. Object Modeling
Database Design	7. Development Strategies
User Interface Design	8. Output and User Interface Design
Testing	9. Data Design
Deployment/Implementation	10. Systems Architecture
Configuration & Change Management	11. Managing Systems Implementation
Software Project Management, e.g. feasibility, prioritization, project management	12. Managing Systems Support and Security

The new guideline replaces technical skills of the functional and object-oriented approaches with business process management.

Appendix 3
Project Narrative
Buy For You (B4U)

Synopsis

There are many reasons why shopping for groceries is a real chore for many people. Whenever there is a need, there will be companies started up to meet market demand. Grocery delivery services crashed and burned in the 2000 dotcom bust. Consider the failure and, eventually, the resurrection of WebVan in this article: http://techcrunch.com/2013/09/27/why-webvan-failed-and-how-home-delivery-2-0-is-addressing-the-problems/

Major players have also been jumping on the bandwagon in recent years. Read the following article to get some general ideas:

http://www.laweekly.com/restaurants/12-great-la-grocery-delivery-services-for-when-youre-too-busy-or-lazy-to-shop-4895408

As you can see, there are several "big guns" in this space:

- AmazonFresh

  http://www.cnet.com/news/amazonfresh-vs-supermarket-a-hands-on-shopping-test/
- Google Express

  https://support.google.com/shoppingexpress/answer/6315260?hl=en
- Albertsons/Safeway/Vons

  http://shop.safeway.com/ecom/shop-by-aisle

Online-to-Offline (O2O) Model

Grocery delivery companies such as AmazonFresh and Google Express offer the traditional B2C model: they handle ordering, fulfillment, and sometimes even hold inventory. The so-called O2O model is very different from B2B and B2C business models. O2O models are inspired by C2C (customer-to-customer) models like the early Etsy model (etsy.com) where individual subscribers can be producers/service providers to other subscribers. Another C2C success is elance.com (acquired and changed to Upwork (upwork.com)). Subscribers offer online services such as programming, translation, graphic design, marketing services, etc. However, many services we need are ***offline* services** that cannot be done online. In an O2O model, an online platform provides matching, directory services, validation, verification, guarantees, and other management functions to the subscribers. Offline services are provided by subscriber to subscriber. The notable O2O businesses that are disruptive are Uber and AirBnB. Read this article to learn more on O2O: http://techcrunch.com/2010/08/07/why-online2offline-commerce-is-a-trillion-dollar-opportunity/.

We see there is a niche market for an O2O grocery shopping platform that can compete with the big B2C companies such as AmazonFresh and Google Express. The team project goal is to analyze and design an O2O mobile App for grocery shopping.

Statement of Work (SOW)

1. Background

Most grocery delivery companies are B2C or a combination of B2B2C. For example, when a customer orders groceries from Google Express, the order will be sent to Google Express's fulfillment centers where a Google Express staff member is assigned to go buy the items from grocery stores and deliver them to the customer for a fee. These companies focus on relatively affluent communities in metropolitan cities. However, it is still questionable whether or not this is a sustainable business.

To people like students and those without cars, grocery shopping is indeed a headache, especially in Southern California. If you are a student staying on campus or in an off-campus apartment, how often do you ask friends to buy you groceries when they are shopping for themselves? Imagine a single mom with 2 kids at home without a car, how often will her neighbors and relatives offer to get groceries for her? There are many other potential customers, including senior citizens who live at home on their own, people with disabilities, and so on.

We are going to change their situations and improve their quality of life. We will develop a system that uses a mobile App as frontend and a cloud backend to support an O2O model for grocery shopping.

2. Objectives

The objective of this project is to create a Buy for You (B4U) platform that offers an O2O business for grocery shopping. On one hand, it helps alleviate the hardship and headache of those who need an inexpensive solution to shop for groceries. On the other hand, a person who offers shopping services can monetize his shopping activities. The marginal time spent on buying extra items for others while shopping for their own is minimal. For people who want to earn extra income, the service fee could be a good subsidy to their grocery bills.

B4U is very different from WebVan, Google Express and Amazon Fresh. B4U offers only a platform and does not provide fulfillment services. As such, B4U is very scalable.

3. Scope

The project will focus on collecting and analyzing requirements, defining the process (business logic) behind the scenes, designing the user interface (UI), and the data model that supports B4U. The entire project will take approximately 12 weeks until the end of the semester, with teams of 3-4 students acting as analysts and designers. The project manager role should be rotated.

4. High Level Requirements

The high level requirements in this section, e.g. features of the App, are not exhaustive and meant to give you a head start for the project. You need to discover more requirements with your team.

First of all, there are several major "actors" (we try not to use word "users" because it can mean many things). A B4U Subscriber is the end user of B4U. If a Subscriber makes shopping requests, he becomes a Requester. If a Subscriber offers shopping services, he is a Buyer. The followings are two example scenarios:

Scenario 1: Requester makes Shopping Requests, such as "Need grocery from Trader Joe's @ University." The request is broadcast to the subscribers in the vicinity defined in the Requester's profile. If another subscriber can provide the service, he acts as a Buyer. Then the Buyer will respond to the Requester to find out the details.

Scenario 2: Buyer offers a shopping service and posts a Procurement Request, such as "Will be shopping at Albertsons @ Campus in 10 mins"; "Will go to Whole Foods @ PCH tomorrow 2/5." The posting will be broadcast to the relevant subscribers. If a subscriber does need milk from Albertsons, he becomes a Requester. The Requester will respond to Buyer for the details.

The preliminary list of high level requirements is as follows:

1. A B4U Subscriber needs to download and install the B4U App either from Google Play or Apple's App Store.
2. A B4U Subscriber needs to register with B4U and create a profile.
3. A Subscriber can be a Requester, or a Buyer, or both.
4. A Requester can post a Shopping Request.
5. A Buyer can post a Procurement Request.
6. Buyers and Requesters can update or cancel their requests as long as they are still open.
7. A Subscriber can post their evaluation of other Subscribers based on their experience.
8. Requesters and Buyers will have their rating along with online comments by subscribers.
9. Online comments can be text, image, audio, or video.
10. A Shopping Request can have items from one or more merchants who participate in B4U by offering online product catalogs.
11. A Shopping Request can specify the time when the items are needed.
12. A Shopping Request can be delivered to other addresses (within the service area of the Buyer).
13. A Shopping Request can be a repeated request which will be posted automatically.
14. A Buyer or a Requestor can specify which store to shop at.
15. A Requester can browse product catalogs to generate shopping lists.
16. Items on the shopping list show the pictures, quantity, unit price, aisle/shelf location (if available) and merchant's name and location.
17. A Procurement Request specifies where to shop, the time and date, and the approximate completion time.
18. While the Buyer is shopping, he can text or chat with the Requestor to amend the shopping list, e.g. out of stock, etc.

5. What to Do

1. Do an extensive search on existing grocery shopping services and identify their features. This will be your background research for the document.
2. Discover new actors, define their roles in B4U.
3. As described in the requirements analysis chapter, you are supposed to interview the end users and other stakeholders to identify their needs. Talk to your friends and family and ask them what they would like to see in B4U. Bring your findings to your team meeting and consolidate them.
4. Using the OSAS-StudentSample.pdf in the Project folder as a reference, your team will create a document that contains the following:

   (A) Use the Intra-group evaluation as the first page (the template is in the Project folder).

   (B) The cover page with project title, your team number, member names, date, etc.

   (C) Table of Contents

   (D) Write up sections similar to Sections 1 to 5 of the OSAS-StudentSample.pdf.

   Section 1 will be an Introduction, or an Executive Summary. You need to highlight the opportunity and why B4U works. You can decide to make it either a non-profit or a for-profit business. In either case, you will need to have a sustainable business model.

   In Section 2, you outline and defend the investment of developing B4U with a business case. Very likely you will argue your business case with a framework such as the SWOT analysis. You can have many ways other than SWOT to support your business case. If you can identify financial data such as revenue projection, cost of development, cash flow, etc., you could carry out a cost and benefit analysis to show the breakeven point and ROI.

   In Section 3, list the high level user requirements, which are similar to the ones I gave you above. The list I gave you, however, is incomplete and is not well-organized. Some of them are non-functional. You need to expand and organize the high level requirements, for example, by actors.

   In Section 4, you will expand the high level requirements in Section 3 into more detail. For example, in Section 3 you have a high level requirement such as "A B4U subscriber needs to register with B4U and create a profile." In Section 4, you will need to expand and include details of the registration process requirements and what the profile will entail.

   There will be a lot of redundancy. We do it on purpose (read the lecture notes, videos, etc. in Course Materials.)

   In Section 5, you will identify the constraints and non-functional specs. I have not given you much information on them. You need to brainstorm and "imagine" many of them. The student sample and materials on Blackboard will help you understand them.

   Finally, in the References section, write down the websites, papers, Apps etc. that you have read and a brief description of them.

Note: The report must be coherent and professional. You should set up your own style in Word such as font and font size, margins, etc. Page numbers are a must.