Volume 13, No. 5
September 2015
ISSN: 1545-679X

INFORMATION SYSTEMS EDUCATION JOURNAL

In this issue:

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **EDSIG**, the Education Special Interest Group of AITP, the Association of Information Technology Professionals (Chicago, Illinois). Publishing frequency is six times per year. The first year of publication is 2003.

ISEDJ is published online (http://isedjorg). Our sister publication, the Proceedings of EDSIG (http://www.edsigcon.org) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is under 40%.

Information Systems Education Journal is pleased to be listed in the 1st Edition of Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org.

Information Systems Education Journal

Using a Balance Scorecard Approach to Evaluate the Value of Service Learning Projects in Online Courses

Dana Schwieger
dschwieger@semo.edu
Southeast Missouri State University
Cape Girardeau, MO

Abstract

Service learning projects serve as a valuable tool for applying course concepts in a way to benefit both the students and community. However, they often require a significant amount of additional effort beyond that required of assigning conventional homework problems. When the projects take place in an online course setting, the level of complexity subsequently increases. Valuing the overall contribution of such projects, in light of their additional instructor and course costs, can be difficult. Such valuations are further complicated with the measures being subjective. In 1992, Drs. Robert Kaplan and David Norton published an article in Harvard Business Review detailing a model for measuring the effect of multiple perspectives, both financial and subjective, on business performance (Kaplan & Norton, 1992). The Balanced Scorecard approach provides both an internal (process based) and external (outcomes based) standpoint to evaluating an organization utilizing four perspectives: (1) Learning and Growth, (2) Business Process, (3) Customer and (4) Financial (Kaplan & Norton, 1996). With its thorough approach to evaluating objective and subjective measures of business performance, the model serves as a potential framework for use in the educational setting. In this article, the author developed and applied a modified version of Kaplan and Norton's Balanced Scorecard model to evaluate the value generated by a service learning project in an online course.

Keywords: Service learning projects, Online learning, Balanced Scorecard, and Management Information Systems course

1. INTRODUCTION

Educators have long recognized the potential value to be realized through service learning projects. John Dewey advocated this style of learning in the early 1900's and his work has served as an impetus for numerous educators hoping to enhance the educational experience of their students (Dewey, 1916). Through service learning projects, students are able to contribute to the community by taking concepts, they have learned over the course of their studies, and applying them to real life situations. They are often able to observe the full cycle of a concept's application from issue identification through implementation, results, and finally modification resulting from process feedback.

This hands-on approach to learning provides significant value for both undergraduate and graduate level students. However, service learning provides additional challenges to faculty as they work with local community partners to develop worthwhile projects that will accommodate the students' capabilities while providing value to the community client. Faculty may also serve in a mentoring or overseeing capacity during the course of the assignment requiring greater involvement in the project and with community clients. Providing such assistance becomes a more significant challenge as the educational landscape adjusts to online learning opportunities.

Universities are incorporating more online learning options to provide greater flexibility for how, when and where students participate in their coursework. Thus, faculty are finding that their course enrollments contain students scattered around the world and in multiple time zones. Although the broader perspectives provided by a wider variety of students can enrich the learning environment, these factors produce additional challenges to providing service learning projects in the classroom.

The learning experience provided by online courses is expected to be equivalent to that provided through face-to-face offerings. However, the effort required to incorporate service learning projects into the online classroom may outweigh the value such projects generate.

In this paper, the author examines the application of a service learning project in a graduate level Management Information Systems (MIS) course through the framework of a modified version of Kaplan and Norton's Balanced Scorecard (1992). The literature review examines research on service learning projects as well as Kaplan and Norton's Balanced Scorecard Model and its application in the educational setting. The third section suggests a modified version of the Balanced Scorecard model for use with experiential learning projects. The fourth and fifth sections describe a course and an online service learning project to which the modified model was applied. In the remaining sections, the modified model is then applied to the course, followed by findings and suggestions for further research.

2. LITERATURE REVIEW

The review of literature starts by looking at service learning projects, their value, and then their use in online courses. Because service learning projects require a significant investment in time and course resources, they should be evaluated to determine whether or not their investment generates sufficient educational value to warrant their inclusion in future coursework.

The field of management contains a number of valuation models to appraise the application of resources to operations. One such model is Kaplan and Norton's Balanced Scorecard (1992). The Balanced Scorecard approach to valuation includes variables to which a specific quantitative value cannot easily be applied (Kaplan & Norton, 1992). This model has also been applied in the educational setting to evaluate educational institutions' operations. Thus, the last section of the literature review examines research associated with the application of the Balanced Scorecard model in education.

Service Learning Projects

Robert Sigmon and William Ramsey first used the term "service learning" in their work with the Manpower Development Internship Program in Atlanta, GA in 1969 (ASLCR, 1970). The term included a value added component as learning occurred in the context of a positive, constructive contribution to the community (Stanton, Giles, & Cruz, 1999). Sigmon was instrumental in developing a community based practicum for the University of South Carolina School of Public Health. He was also a strong advocate for community-based public service experiential learning through his work at the local, state, and national levels (Sigmon, 2009)

Sigmon advocated the "reciprocal" nature of experiential learning both for the recipient of the service as well as the provider (Sigmon, 1979) with both direct and indirect benefits possible (Terry & Bohnenberger, 2004). Boyer (1994) advocated a "scholarship of engagement" connecting theory to practice by educating students to be responsible citizens rather than just focusing on educating solely for a career (Boyer, 1994; Coye, 1997).

A number of researchers have realized the value that service learning projects can generate to both community clients and students of all ages with articles describing projects in K-12 programs through community colleges and university settings around the world (Chan, 2012). Service learning projects are increasingly becoming a common teaching tool in educational programs. Chen recently noted the importance of service learning projects worldwide as it has become "a major teaching and learning component in the upcoming curriculum reform in Hong Kong higher education…" (Chen, 2014: 414)

Service Learning Projects for Online Courses

Although there are a number of articles describing service learning projects in education, little has been written about service learning projects offered through online courses or in the online environment (Ball & Schilling, 2006; Cleary & Fammia, 2012; Hagan, 2012; Most, 2011). In an article written by Ball and Schilling (2006), the authors describe an IT service learning project offered at the Indiana University School of Library and Information Science.

Students in the course served as technology consultants providing assistance off-site to Indiana librarians and archivists.

Hagan (2012) noted that although, "… there is widespread use of experiential learning models at the undergraduate level, they are not as popular at the graduate level, especially for hybrid or online courses…" (p 625). The author indicated that further research should be conducted evaluating the effectiveness of service learning projects, student learning from the process, application of course concepts, and satisfaction of clients (Hagan, 2012).

Most (2011) noted that almost half of the accredited graduate programs in library science delivered some or all of their content through the online environment (Most, 2011). Thus, for programs such as theirs, developing valuable experiential learning opportunities for their online students has been difficult, yet important.

Little research has been conducted on service learning projects in distance learning classes. Even less has been conducted on applying evaluative instruments to determine the overall value of service learning projects. The research has primarily focused on one dimension of the service learning project (Helm-Stevens and Griego, 2003, Thomas & Busby, 2003; Toncar et al, 1996). No research was found that provided a multi-dimensional evaluation of the contribution such projects provided to the educational experience.

In the next section, the author examines the Balance Scorecard model. This model is often used in business to evaluate multiple performance metrics when some of the variables being measured are subjective.

Balanced Scorecard

Kaplan and Norton (1992) created the Balanced Scorecard (BSC) approach to business valuation as a performance measurement model to evaluate the value of businesses from multiple perspectives. The framework provides a more balanced view of the organization by including both nonfinancial as well as financial metrics in the valuation process. Since its inception, the model has evolved into a strategic planning tool, rather than simply an evaluative instrument with the vision of the organization guiding the development and measurement of subsequent strategies (Kaplan & Norton, 2005). The 2005 model consists of four perspectives driven by the vision and strategies of the organization which include (Figure 1):

- **Financial Performance:** To succeed financially, how should we appear to our shareholders?
- **Customer/Stakeholder Satisfaction:** To satisfy our customers, how do we create value for them?
- **Internal Business Process Efficiency:** To satisfy our shareholders and customers, at which business processes must we excel?
- **Knowledge and Innovation:** To achieve our vision, how will we sustain our ability to change and improve? (Kaplan & Norton, 2005)

Figure 1 – Kaplan & Norton's Balanced Scorecard

Numerous commercial organizations have applied this model for years to improve their operations. An increasing number of non-for-profits have started applying Kaplan and Norton's model as well. In the next section, the author examines the research that has been conducted on applying the Balanced Scorecard model to higher education.

Balanced Scorecard in Education

Universities are finding the value of applying business models, such as Balanced Scorecard, to the management of the educational setting (Chen, Ching-Chow, & Shiau, 2006; Sudirman, 2014; Taylor & Baines, 2012). Karathanos & Karathanos (2005) cited the similarities between the Baldrige Education Criteria for Performance Excellence and the criteria of the Balanced Scorecard approach. The paper presented the detailed measures of the balanced scorecards of the first three recipients of the Baldrige Education Awards (two school districts and one university). The authors maintained that, "Although the BSCs of these three institutions cover the same perspectives, their individual measures differ considerably, reflecting the differences in their individual missions" (Karathanos & Karathanos, 2005:226)

Chang and Chow (1999) received surveys from the department chairs of 69 college and university accounting programs located in the U.S. and Canada. Overall, the respondents felt that the balanced scorecard could be an effective instrument in helping an accounting department achieve and improve upon their goals. Similar to Karathanos and Karathanos (2005) the authors noted that each program would need to " design its own scorecard consistent with its mission and circumstances…" (Chang & Chow, 1999: 410).

Chang & Chow (1994) took the Balanced Scorecard model even deeper into the educational process. The authors suggested that balanced scorecards be developed for each individual member of a department to exploit their specific skills and capabilities while using the synergistic effect to enhance an overall program. In the next sections, the author considers Chang & Chow's notion of "exploiting specific skills" by applying the Balanced Scorecard model to the course level to examine individual course projects, specifically, online service learning projects.

3. DEVELOPMENT OF A BALANCED SCORECARD FOR SERVICE LEARNING PROJECTS

With service learning projects, students are able to visualize the application of course concepts to real world situations and to experience the benefit of their knowledge and skills applied outside the classroom setting for the greater good. Much like a business setting, service learning projects have internal and external stakeholders as well as processes from which to learn and further improve upon.

In designing a balanced scorecard for an organization, Kaplan noted that the design process starts with determining the mission, then setting strategic objectives to fulfill the mission, and finally defining measures (Kaplan, 1994). The balanced scorecard serves as the framework for organizing and defining strategic objectives. To apply the model to the realm of a service learning project in education requires the perspectives to be slightly modified (Figure 2).

- **Vision and Strategy:** As in business, the vision of the program or course will drive the strategic objectives and, in turn, the measures.
- **Scholarship:** This perspective replaces "Financial" in the original model and focuses upon the value gained by students.
- **Customer Client:** Similar to the business-oriented model, customer client would be the recipient of the project's product or service.
- **Instructional Method Efficiency:** This variable corresponds to "internal business processes efficiency" and determining what the students must excel at in the project for the customer.
- **Innovation and Learning: Similar to business, this variable refers to evaluating the instructional method and determining how it could be improved upon for future classes.**

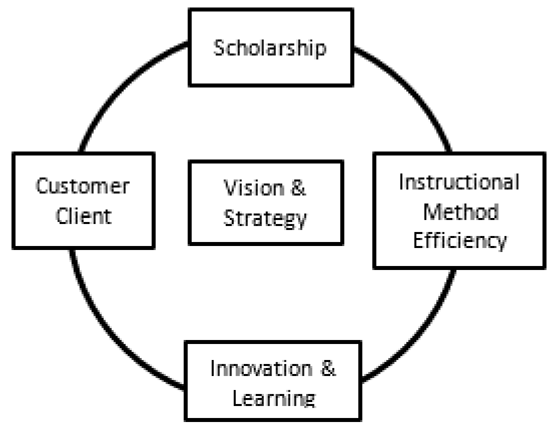


Figure 2 - Educational Balanced Scorecard

In business, the Balanced Scorecard model is used to strategically manage the business to increase the benefits of the business while managing the costs. Course projects have similar goals in the classroom setting. Instructors want to use projects that increase the educational value of the class while managing the instructional costs of the project in relation to the rest of their workload. In the next section, the author describes the course and service learning project on which the modified Balanced Scorecard model is applied.

4. COURSE

Integrated Decision Support Systems (BA630) is a core MIS course required of all MBA students at the author's institution. The course is offered twice a year to graduate students in the College of Business. The three credit hour course is usually offered in a face-to-face format during the fall semester. The online format is offered for six weeks during the summer semester. Although some course work is completed individually, the larger course projects are completed in groups in both formats with consideration made for making the learning experience equal across both offerings.

There are only two group projects assigned in the six week online course, one assigned at the beginning of the semester and the second one assigned halfway through the course. The purpose of the first group project is to familiarize students with online collaboration tools and to help students get to know each other on a small project.

The second group project is a systems analysis and design project with the purpose being to pull together the concepts covered over the course of the semester. Projects have been created for community clients in both the face-to-face and online versions of the course. When community clients have not been found, facsimiles of previous projects are used with written descriptions replacing the interview process.

Service learning projects have a tendency to require extra attention from the instructor to ensure that a quality project is delivered to the client. Likewise, online classes also have a tendency to require additional attention as more individualized assistance is provided to the students. Combining both of those characteristics by requiring online students to participate in service learning projects for class may generate more costs to the instructor, and class in general, than the value they create. With this in mind, the next section examines the value of the project through the framework of a modified version of Kaplan and Norton's Balanced Scorecard (1992).

5. ONLINE SERVICE LEARNING PROJECT

During summer 2013, a systems analysis and database design project was assigned in the six week session of the MBA MIS course. Students were tasked with developing a small database for a local community group to use in hosting their annual 5K race. The purpose of the race was to raise money for their cause as well as generate awareness.

All student groups worked on the same project for the same client. Thus, accommodations had to be made to minimize the time requirements of the community client responding to questions of, and meetings with, seven groups. The author interviewed the client and then wrote the organization's story as a case study. Each student was given the case study as well as contact information for the community client representative. Each group had been assigned a local to campus group member. The groups were asked to utilize their local group member to serve as the liaison with the community client and to filter all of their questions through that person. They were also asked to share their information with the rest of the class through a forum to minimize replication of questions.

When the students were finished with their projects, they submitted their work to the instructor for grading. The instructor then selected the best project to present to the client. Because the 5K race occurs once a year, the database was implemented later in the year.

6. APPLICATION OF THE MODEL

As Kaplan and Norton indicated in their development of the Balanced Scorecard model (1992), there is more than just the financial perspective to consider when valuing an organization. The same can be said when determining the value of classroom projects.

If the instructor examined the projects described from a personal costs perspective only, the projects would probably be abandoned for something less personally time consuming. The instructor spent time interviewing and writing the scenario, fielding some of the questions for the community client, presenting the project to the clients, and finally modifying, installing, and to some extent, maintaining the project once the course was over.

However, by using the modified balanced score card to examine the value generated by the project from multiple perspectives, a more complete view of the project's contributions can be had. In measuring the overall value of the online service learning project, we first need to determine whether or not the vision of the project was accomplished as it guides the rest of the model. (Refer to Figure 2.)

Vision: The vision of the project was to give the students a real life team-oriented experience that would allow them to apply their knowledge and course concepts to providing a product that would help a member of the community. The vision was met as students were given a team-oriented real life project through which to apply course concepts and skills.

Scholarship: The students had the opportunity to work on a project with an actual organization and to know that their work was worthwhile and not solely used to meet a requirement for earning a grade in the course. The project provided a hands-on application of concepts learned in a virtual team environment. In this environment, students had to find ways to work together virtually and capitalize on each other's strengths.

Students also gained experience in project and time management as they had a short time frame in which to accomplish the project. Interest in the project seemed to be maintained through its duration with some students volunteering to continue on with the project once the class was finished.

In evaluating the scholarship aspect of the project, students were presented with the opportunity to gain and apply several skills that could not easily have been attained outside an experiential learning project. Students seemed more engaged in the client based project than past semesters in which a fabricated case was used.

Customer Client: The benefits to the community client were multifold with an increased awareness of the cause among the students and a working system. The client was satisfied with the project and the interaction that they had with the students. They used the database in their most recent 5K race. Because the race was a new activity for the client, the client did not think of all of the reports they would like the database to generate nor the additional fields that would need to be collected to generate those reports. The instructor assisted the client in modifying the database to include the additional fields and reports.

Instructional Method Efficiency: Course concepts were taught through their application in a client setting over a three week time period. In order to provide the three week time block for the project, course concepts had to be rearranged. Preliminary concepts were slightly rushed in order to allocate sufficient time for the project. In addition, the service learning project time frame overlapped with those of other course projects. This overlapping of project due dates added to the stress level of some students.

To accommodate the online environment and shortened time frame, the instructor provided additional groundwork beyond that normally required for service learning projects. Such work included summarizing the client's background and situation, and installing, implementing, and training the client on the final system. In addition, because the project ended so quickly, students were not given enough time to reflect upon the service learning project as a group as well as discus the closing of the project. (Part of the value associated with service learning projects and project management is the collective reflection that takes place at the end of the project.)

Innovation and Learning: The project moved both the course and the instructor forward. In trying to minimize the interruptions on the client, materials were developed that could be used in future semesters in which clients were not available. The instructor also looked for ways to make the project more efficient for all parties involved. By personally working with the client and on the project during and after the course was over, the instructor's skills were sharpened and the project was honed for future semesters.

Overall, although the project was able to accomplish several learning objectives, its application infringed upon the time allotted to other aspects of the course as well as the time resources of the instructor. Had the project been offered during a regular sixteen week time period, the results of the model would have been different. The project would not have been rushed and time could have been allotted for a give-and-take between the client and the students as well as time for a proper project closing. In reflecting upon the project and its evaluation through the model, it seems a fabricated scenario would have best addressed the needs of the project and restrictions of the course without adding undue stress.

7. EVALUATION OF THE MODEL

For the most part, the model applied well to evaluating the service learning project. However, the variables "Scholarship" and "Instructional Method Efficiency" may need to be more carefully defined in future applications to prevent overlap.

8. FUTURE WORK AND CONCLUSION

Little has been done in evaluating service learning projects from multiple perspectives. The modified model could be used to evaluate experiential learning projects that are conducted over a standard semester time frame. The model could also be used to examine online courses and programs as well as universities whose content is delivered wholly online.

In this paper, the author examined literature associated with the use of service learning projects in higher education, service learning projects in the online environment, Kaplan and Norton's Balanced Scorecard model, and use of the model in the area of higher education. The author then proposed a modified version of the model to be applied to managing and evaluating course projects. The modified model was then used to examine a service learning project offered online at the author's institution. Through application of the model, the author found that although the project generated considerable value, the time frame in which the project was taught significantly affected the value the project had individually and what it contributed to the class as a whole.

9. REFERENCES

Atlanta Service Learning Conference Report (1970). Southern Regional Education Board Retrieved from http://files.eric.ed.gov/fulltext/ED072718.pdf .

Balanced Scorecard Institute (2014). *Balanced Scorecard Basics*., retrieved on May 27, 2014 from http://balancedscorecard.org/Resources/AbouttheBalancedScorecard/tabid/55/Default.aspx.

Ball, M.A. & Schilling, K. (2006). Service Learning, Technology and LIS Education. *Journal of Education for Library and Information Science*. *47*(4), p. 277-290.

Boyer, E. (1994). Creating the New American College." *Chronicle of Higher Education*, A48.

Bringle, R.G. & Hatcher, J.A. (1995). A Service-Learning Curriculum for Faculty. Michigan *Journal of Community Service-Learning*. 2, 112-122.

Chan, C. (2012). Exploring a Service learning project through Kolb's Learning Theory using a qualitative research method. *European Journal of Engineering Education*, *37*(4), 405-415.

Chen, S., Ching-Chow, Y., & Jiun-Yan Shiau. (2006). The Application of Balanced Scorecard in the Performance Evaluation of Higher Education. *The TQM Magazine*, *18*(2), 190-205. Retrieved from http://search.proquest.com/docview/227579240?accountid=38003.

Cleary, Y., & Flammia, M. (2012). Preparing Technical Communication Students to Function as User Advocates in a Self-Service Society. *Journal of Technical Writing & Communication*, *42*(3), 305-322.

Coye, D. (1997). Ernest Boyer and the New American College. *Change*, *29*, 20-29. Retrieved from http://search.proquest.com/docview/208054341?accountid=38003.

Dewey, J., (1916). *Democracy and Education: An Introduction to the Philosophy of Education*. New York: The Free Press.

Hagan, L. M. (2012). Fostering experiential learning and service through client projects in graduate business courses offered online. *American Journal of Business Education (Online)*, *5*(5), 623. Retrieved from http://search.proquest.com/docview/1418446153?accountid=38003.

Helm-Stevens, R., & Griego, O. (2009). Evaluating Experiential Learning In Organizational Behavior: Taking Measure Of Student Perception Regarding Group Experience. *American Journal of Economics and Business Administration, 1*(2), 138-140. Retrieved from http://search.proquest.com/docview/1027115297?accountid=38003.

Kaplan, R. S. (2010). Conceptual Foundations of the Balanced Scorecard, *Harvard Business Review Working Paper* 10-074. 1-36.

Kaplan, R. S. (1994). Devising a balanced scorecard matched to business strategy. *Planning Review*, *22*(5), 15. Retrieved from http://search.proquest.com/docview/194368377?accountid=38003.

Kaplan, R. S. & Norton, D. P. (1996). Using the Balanced Scorecard as a Strategic Management System. *Harvard Business Review*, *74*(1) 75-85.

Kaplan, R.S. & Norton, D. P. (1992). The Balanced Scorecard: Measures that Drive

Performance. *Harvard Business Review*, *79* (1) 71-79.

Karathanos, D., & Karathanos, P. (2005). Applying the balanced scorecard to education. *Journal of Education for Business 80*(4), 222-230. Retrieved from http://search.proquest.com/docview/202822712?accountid=38003.

Lutz, K., & Unruh Snyder, L. (2014). Creating a College Level Animal Healthcare Service-Learning Project in International Elementary Schools. *Global Studies Journal*, *6*(1), 1-6.

McClellan, J. L. (2011). Beyond Student Learning Outcomes: Developing Comprehensive, Strategic Assessment Plans for Advising Programmes. *Journal of Higher Education Policy & Management*, *33*(6), 641-652.

Sigmon, R. (1979). Service-learning: Three Principles. Synergist. National Center for Service-Learning, *ACTION*, *8*(1): 9-11.

Sigmon, R. (2009). Robert Sigmon Biography. Retrieved September 16, 2014, from http://org.elon.edu/library/ead/msscoll033robertsigmon.xml.

Stanton, T. K., Giles, D. E., Jr., & Cruz, N. I. (1999). *Service Learning: A Movement's Pioneers Reflect on Its Origins, Practice, and Future*. San Francisco: Josey-Bass.

Sudirman, I. (2012). Implementing balanced scorecard in higher education management. *International Journal of Business and Social Science*, *3*(18) Retrieved from http://search.proquest.com/docview/1127690565?accountid=38003.

Taylor, J., & Baines, C. (2012). Performance Management in UK Universities: Implementing the Balanced Scorecard. *Journal of Higher Education Policy & Management*, *34*(2), 111-124. doi:10.1080/1360080X.2012.662737.

Terry, A. W. & Bohnenberger, J. E. (2004). Blueprint for Incorporating Service Learning: A Basic, Developmental, K-12 Service Learning Typology. *The Journal of Experiential Education*, *27*(1), 15-31. Retrieved from http://search.proquest.com/docview/274926685?accountid=38003.

Thomas, S., & Busby, S. (2003). Do Industry Collaborative Projects Enhance Students' Learning? *Education & Training*, *45*(4), 226-235. Retrieved from http://search.proquest.com/docview/237087973?accountid=38003.

Toncar, M. F., Reid, J. S., Burns, D. J., Anderson, C. E., & Nguyen, H. P. (2006). Uniform Assessment of the Benefits of Service Learning: The Development, Evaluation, and Implementation of the Seleb Scale. *Journal of Marketing Theory and Practice, 14*(3), 223-238. Retrieved from http://search.proquest.com/docview/212168813?accountid=38003.

Introducing Big Data Concepts in an Introductory Technology Course

Mark Frydenberg
mfrydenberg@bentley.edu
Computer Information Systems Department
Bentley University,
Waltham, MA 02452, USA

Abstract

From their presence on social media sites to in-house application data files, the amount of data that companies, governments, individuals, and sensors generate is overwhelming. The growth of Big Data in both consumer and enterprise activities has caused educators to consider options for including Big Data in the Information Systems curriculum. Introducing Big Data concepts and technologies in the classroom often is reserved for advanced students in database or programming courses. This paper explores approaches for integrating Big Data into the Information Systems curriculum, and presents a sample lesson for presenting basic Big Data concepts to first year students in a general education Information Technology course. As the need for IT professionals with Big Data skills will continue to increase, including these topics in a general education technology curriculum is especially pertinent.

Keywords: Big Data, Technology Concepts, Google, Data Visualization

1. INTRODUCTION

The rise of Web 2.0 applications (2004-2009) saw the Internet evolve as a platform upon which users would generate their own content and run applications on many devices. "Data as the next 'Intel inside'," a reference to Tim O'Reilly's analogy that data powers web applications much in the same way that processors power computers, became the mantra for companies moving their products and services online (O'Reilly, 2005). In the years that followed, the growth of commerce and social applications on the web, the proliferation of mobile devices, the rise of cloud computing, the ubiquity of Internet access in consumer and enterprise activities, and the emergence of the Internet of Things has resulted in daily digital activities that generate large quantities of data. Such large databases often are kept on servers distributed across the Internet. New technologies make it possible to gather, store, query, and retrieve all of this data over the Internet.

Social networking sites, open government and healthcare databases, e-commerce transactions and automated sensors are among the many daily digital interactions that generate data to be stored online for analysis of possible trends. With these activities comes information on social interests, consumer trends, and business prospects that if understood, could provide insights into business intelligence, social trends, and organizational opportunities.

In the highly cited report *Big Data: The Next Frontier for Competition*, the McKinsey Foundation (Manyika, Chui, Bughin, Dobbs, Roxburgh, & Byers, 2011) states that the increase in the amount of information produced through these actions and transactions will provide important insights to organization that can understand such large data sets. They define Big Data as "data sets whose size is beyond the ability of typical database software tools to capture, store, manage, and analyze." The value of Big Data "…comes from the patterns that can be derived by making connections between pieces of data, about an individual, about individuals in relation to others, about groups of people, or simply about the structure of information itself" (boyd & Crawford, 2011, p. 1). Business, education, and government sectors routinely produce and manage terabytes and petabytes of data, and rely on distributed tools to store and make sense of it all.

As the enterprise finds value from Big Data and the information it contains, introducing Big Data concepts into the undergraduate Information Systems classroom becomes a necessary curriculum addition in order to prepare tomorrow's IT professionals and knowledge workers.

This literature review describes programs for teaching big data concepts to Information Systems majors and minors, and the narrative shares a lesson for introducing basic Big Data concepts to first-year students in a general education introductory IT course. The author presents a teaching exercise that gives students an appreciation for the issues without requiring significant technology experience.

2. BIG DATA IN THE INFORMATION SYSTEMS CURRICULUM

Where and how to introduce Big Data concepts in the information systems curriculum has received much discussion. "IS programs, faculty members, and text books have so far been, in practice, somewhat lukewarm about their relationship with Big Data" (Topi, 2013, p. 12). Many degree programs in information systems are introducing courses in business analytics and data mining so Information Systems majors can develop skills in managing and interpreting large scale enterprise data sets (King & Satyanarayana, 2013), (Grossniklaus & Maier, 2012). These courses require basic database skills usually taught in the undergraduate business education curriculum.

General education courses also could expose students to Big Data concepts by presenting students with hands-on exercises involving online databases, productivity tools, and data visualization tools. Doing so offers students the opportunity to develop familiarity and skills with these tools from an enterprise perspective.
Today's students need to understand the requirements for storage space, processing power, Internet connectivity, security, and ways to access or update information online in order to recommend solutions to their future employers (Frydenberg & Andone, 2011). Students who learn about relational databases need to understand where that technology falls short. So often, textbooks provide sample datasets, which are nicely structured, well formed, and conducive to a particular problem. In the real world, data sets are not always so clean and well-structured.

As the business education curriculum evolves, educators must consider current technology trends such as Big Data to determine where and how to integrate these topics. "Business analytics is one of the fastest growing areas of focus for Information Systems (IS) programs, particularly at the graduate level. IS departments are establishing new analytics programs at a fast pace, and companies are very interested in graduates with strong capabilities in business analytics. Within this broad topic area, Big Data continues to become increasingly important, even though the concept itself is at times confusing and continues to evolve" (Topi, 2013, p. 12).

The remainder of this paper presents a lesson for introducing big data concepts in a general education introductory technology course at a business university.

3. A LESSON FOR TEACHING BIG DATA CONCEPTS IN A GENERAL EDUCATION CONTEXT

IT 101 (Introduction to Information Technology and Computing Systems) is a first-year required general education course for all students at Bentley University, a business university in Massachusetts. The course introduces students to types of hardware and software, operating systems, cloud storage, the Internet and the World Wide Web, HTML, wireless networking, multimedia, spreadsheet and database applications, and cloud computing concepts.

Introducing Big Data in an introductory technology course allows students to apply their knowledge of several of these topics (especially databases, the Internet and World Wide Web, and cloud computing) in order to understand the technological issues required to store, search, and maintain Big Data sets, and Excel as a data analysis and visualization tool for smaller data sets.

This section describes a series of classroom demonstrations and activities piloted in seven sections of IT 101 over two consecutive semesters. The same instructor presented this material to each section. The presentation was preceded by a survey to ascertain students' familiarity with Big Data concepts. A follow-up survey captured student feedback after students completed a homework assignment on the topic.

Volume, Velocity, Variety
The terms volume, velocity, and variety have become synonymous with features of Big Data sets (Russom, 2011), (Laney, 2001; 2012) (Big Data 3 V's: Volume, Variety, Velocity (Infographic), 2013). Others have added Value and Veracity as additional V's of Big Data to suggest that data is a company's important asset, and that a company's data must be accurate.

Gartner analyst Doug Laney first described the growth of large data sets in three dimensions: volume (the amount of data generated), velocity (the rate at which data is generated), and variety (the different formats in which data can appear.) In 2001, as companies and consumers began to use the Internet for e-commerce, and nearly five years before the dawn of the Web as a social and collaborative platform, Laney (2001) wrote:

> While enterprises struggle to consolidate systems and collapse redundant databases to enable greater operational, analytical, and collaborative consistencies, changing economic conditions have made this job more difficult. E-commerce, in particular, has exploded data management challenges along three dimensions: volumes, velocity and variety.

With this historical context, students are ready to understand these concepts through examples of how their everyday interactions on the web contribute to Big Data.

<u>Volume.</u> Capturing student interest in Big Data is often easy to accomplish by presenting statistics about the amount of information generated by consumers and the enterprise. Companies routinely require storage in petabytes (1000 terabytes) rather than gigabytes and terabytes. For example, Facebook stores and analyzes over 30 petabytes of data generated by its users; Walmart has 2.5 petabytes of customer data. As the volume of data grows, the challenge comes in being able to store, access, and analyze it effectively.
A lesson on data volume begins by asking students to consider that Google archives all search queries from its users. Asking students the amount of storage required based on the number of queries per day, and then what information Google can determine from saved search queries is a relevant way to begin a classroom conversation about Big Data, storage required, and related issues of data privacy.

Collaborative filtering is the process of using data from the activities of many people to make decisions or offer advice. The following two demonstrations, of Google Autocomplete and Amazon.com, show the value obtained from being able to process large volumes of data.

Google Autocomplete allows the search engine to predict search queries that dynamically update with each character entered into the search bar as shown in Appendix I, Figure 1. Google Autocomplete displays search queries that reflect the search activity of the user's search history, other Google users search queries, and the content of web pages indexed by the search engine.

Google Trends "is a real-time daily and weekly index of the volume of queries that users enter into Google" (Choi & Varian, 2012). Appendix I, Figure 2 shows the volume of queries for three technology trends as search terms.

Asking students to analyze this chart develops their critical thinking skills and allows them to make informed predictions about future trends. Examining news stories associated with high and low points on a search term's trend line can give insights into why it increased or decreased in search popularity.

Amazon.com uses data from customer purchases to recommend related items for purchase. For example, Amazon recommends sunglasses, pails, and shovels to complement the purchase of beach balls, as shown in Appendix I, Figure 3.

<u>Velocity.</u> Real-time statistics and visualizations of social media data provide a tangible way for students to understand how quickly data is generated. For example, in 2013, Twitter received 500 million tweets per day, YouTube users uploaded 100 hours of video every minute, and Facebook users liked over 4.5 billion posts per day. National Weather Service sensors at 1800 tracking stations across the country report temperature and climate information every hour of the day. Consumers purchase 426 items per second on Amazon.com during the holiday shopping period. Google processed approximately 5,922,000,000 search queries per day in 2013 (StatisticsBrain, 2013), and saves all of them.

In a business context, it is important for organizations to be able to analyze data that changes quickly in order to make good business decisions. "The Road," an IBM-produced video describes the importance of data analytics, and poses the question:

> "If you were to stand at a road, and the cars are whipping by, and all you can do is take a snapshot of the way the road looked five minutes ago, how would you know when to cross the road? 9 out of 10 organizations still make decisions this way every day, using out of date information. The organizations that are the most competitive are the ones who are going to be able to make sense of what they learn as fast as they learn it." (IBM, 2013)

To visualize data that changes quickly, consider TweetPing.net, shown in Appendix I, Figure 4. TweetPing is a visualization that illustrates the velocity of data generated by Twitter users. The real-time map becomes denser as more Tweets are processed and plotted. The app records the location of origin, hashtags, mentions, and tracks total Tweets, words, and characters in each Tweet.

Variety. Digital information may be represented and stored in a variety of formats that can be easily read and interpreted by both humans and computer programs.

While much data is structured, stored in rows and columns, unstructured data generally is more complex, and may include items such as Tweets, social media "likes" or updates, retina scans, Wikipedia articles, fingerprints, and facial recognition features. This information usually does not originate in a format that is easily searchable or fit into relational database tables. Its growth in recent years has contributed to the development of new technologies for interacting with and processing unstructured Data.

Students can investigate these sources of unstructured data. For example, facial recognition is an up and coming technology that helps in crime fighting, advertising, and social networking. These applications process facial data points and compare them to other faces stored in a database in order to produce appropriate content. For example, Facebook's Deepface technology is accurate in recognizing faces 97% of the time, and is used to make suggestions of people to tag in photographs. In industry, digital advertising kiosks use cameras to scan the person looking at a monitor, and then display content appropriate to the person's age and gender, as determined by facial recognition software.

To show that data is not always represented in neat rows and columns, the instructor searched dbPedia, a semantic database obtained from of Wikipedia, using Relfinder (http://www.visualdataweb.org/relfinder.php). Relfinder uses data from dbPedia, an index of Wikipedia's text content stored in RDF (representational data format) format, to find relationships between seemingly unrelated topics (Exner & Nugues, 2012). RDF uses triples of values (object1, relationship, object2) to store relationships between objects. Students can see that searching for relationships produces different results than searching plain text. In Appendix I, Figure 5, Relfinder shows commonalities between Madonna and Aretha Franklin based on information in dbPedia.

Introducing Big Data Technologies

Several new technologies take advantage of the distributed nature of large databases and parallel processing needed to process them quickly. Today's students should at least be aware of the problems that technologies such as Hadoop or MapReduce try to solve, so they might better succeed as 21st century IT professionals. "Choosing the best collaboration partners requires a good understanding of the technologies themselves, knowledge of the key solutions and solution types available in the marketplace, and general organizational skills and knowledge related to technical resource acquisition." (Topi, 2013, p. 13)

Querying large datasets can be time consuming and costly without an appropriate hardware and infrastructure. The instructor provided a lesson to students using Google BigQuery (Frydenberg, 2013). Google BigQuery is an easy to use technology that allows students to learn about distributed databases, SQL queries, and data mining. Google BigQuery is a query service for running SQL-like queries against multiple terabytes of data in seconds. It is one of Google's core technologies that has been used for data analytics tasks since 2006. Google BigQuery performs rapid SQL-like queries against online database tables, using the processing power of Google's big data infrastructure. Google provides several large sample databases for querying, including Shakespeare, birth and weather station information for use with BigQuery. Advanced users can upload their own databases (Sato, 2012).

After a short demonstration of BigQuery in the classroom, the instructor distributed a tutorial handout for students to follow (Frydenberg, Drinking from the Fire Hose: Tools for Teaching Big Data Concepts in the Introductory IT Classroom, 2013) working in small groups, where students took on the roles of reader, doer, and checker (Frydenberg, Flipping Excel, 2013) to complete the assignment.

Appendix I, Figure 6 shows details about the public natality (birth) database, which has 137,826,763 rows, and takes up 21.9 GB of storage. It provides birth information from 1969 through 2008.

Appendix I, Figure 7 shows a Google BigQuery query to display the average age of birth mothers grouped by year and state. (Programmable Web, 2012) It took Google BigQuery 2 seconds to process 2.5 GB of information when performing this query, which results in 1840 rows retrieved and summarized from the table.

After performing the query in Google BigQuery, students can use a BigQuery Connector tool from https://bigquery-connector.appspot.com/ to import the results into Excel for further analysis. Appendix I, Figure 8 shows a pivot table report in Excel summarizing average age of birth mothers grouped by state during this period. Spark lines provide a handy visualization of this data.

Examining the spark lines clarifies the trend that mother ages has increased during the years for which data is available. Performing the analytics and visualization in Excel (or any other client side data visualization tool) is not nearly as computationally intensive as processing 21.9 GB of data in the original online database.

4. METHODOLOGY

Students in seven sections of an introductory IT course received a presentation demonstrating velocity, variety, and volume as described here, as well as a hands-on activity using Google BigQuery as described in the previous section. They completed questionnaires before and after so the study could ascertain the development of their understanding of Big Data.

166 students (64 female, 102 male) voluntarily completed a survey prior to the lesson on Big Data; 160 students (64 female, 96 male) completed the survey after the lesson.

The first set of questions, shown in Appendix I, Figure 9, asked about awareness of big data and other current technology developments. It is interesting to note that while student awareness of technology developments was at about 60%, relatively few students agreed that they were familiar with "big data" before the exercise. This number nearly doubled after the lesson. Many were aware of social networking sites as a popular source of online data. Responses were provided in a 5-point Likert scale (from strongly disagree to strongly agree).

Students before and after the lesson responded similarly about their familiarity with technology news and developments, and as expected, the number of students "familiar with the phrase 'Big Data'" increased after the lesson.

After the lesson, students were asked two additional questions related to the relevance of data analysis and Big Data, as shown in Appendix I, Figure 10.

The majority of students agreed or strongly agreed that it is important to know how to analyze and interpret data, and that Big Data is a relevant topic that will impact their own lives as digital students. Students commented on aspects of the lesson that they found helpful, as shown in Appendix I, Figure 11.

The demos for velocity, volume and variety, and using Google Big Query, and instructing students about the role of Social Media in generating Big Data were the most useful elements of the lesson.
This is likely because of the hands-on experience with these tools that makes the topic relevant for students.

When asked what students learned about Big Data, in addition to their new awareness of Big Data technology concepts, their responses also touched on social and privacy concerns. Said one student: "Big Data has become a part of our everyday lives. It is tracing our locations through our phones and we constantly leave paths through various social media sites." Another student commented, "Every move we make seems to be recorded and every human being is leaving a data trail that will be there after the human being is long gone."

About learning Big Data concepts in a general education introductory technology course, one student commented, "I found the presentation quite informative, and appreciated the exposure to material, particularly the Google applications of which I was not previously aware. I believe the content was relevant, crucial and necessary in what many would consider the 'technological age,' and I have begun to realize that informing myself of news and materials in this field may prove beneficial, if not necessary." Others wrote that they "had never heard about Big Data specifically and [are] now very interested in the topic."

5. SUMMARY

Both consumers and the enterprise make use of collaborative tools, social networking sites, Internet-connected devices, and online transactions that produce large quantities of data every day. Big Data offers new challenges and technologies for storing, accessing, and processing large sets data as fast as it is being generated.

This paper presents a lesson for introductory students to learn about Big Data concepts and technologies. It offers relevant examples of Volume, Velocity, and Variety of Big Data, and outlines a hands-on experience querying an online database with Google BigQuery, and then using Excel to create a visualization of the selected data. By considering social media, commerce, and public databases, students can gain exposure to Big Data concepts and technologies in their introductory IT course.

6. REFERENCES

Big Data 3 V's: Volume, Variety, Velocity (Infographic). (2013, July 25). Retrieved January 10, 2014, from What's the Big Data?: http://whatsthebigdata.com/2013/07/25/big-data-3-vs-volume-variety-velocity-infographic/

Boyd, d., & Crawford, K. (2011). Six Provocations for Big Data. *A Decade in Time: Symposium on the Dynamics of the Internet and Society* (p. 17). Oxford Internet Institute.

Choi, H., & Varian, H. (2012). Predicting the Pressent with Google Trends. *Economic Record, 88*(1), 2-9.

Exner, P., & Nugues, P. (2012). Entity extraction: From unstructured text to DBpedia RDF triples. *The Web of Linked Entities Workshop (WoLE).*

Frydenberg, M. (2013). Drinking from the Fire Hose: Tools for Teaching Big Data Concepts in the Introductory IT Classroom. *The Proceedings of the Information Systems Education Conference.* San Antonio, TX. Retrieved from http://proc.isecon.org/2013/workshops/2674.html

Frydenberg, M. (2013). Flipping Excel. *The Proceedings of the Information Systems Education Conference.* San Antonio, TX. Retrieved from http://isedj.org/2013-11/N1/ISEDJv11n1p63.html

Frydenberg, M., & Andone, D. (2011). Learning for 21st century skills. *2011 International Conference on Information Society (i-Society).* London: IEEE.

Google. (n.d.). *Google Trends*. Retrieved from http://google.com/trends

Grossniklaus, M., & Maier, D. (2012, March). The curriculum forecast for Portland: cloudy with a chance of data. *ACM SIGMOD Record, 44*(1), pp. 74-77.

IBM. (2013, May 10). *The Road: Intelligent Data Management and Analytics.* Retrieved January 10, 2014, from Daily Motion: http://www.dailymotion.com/video/xdaoae_ibm-commercial-the-road-intelligent_tech

King, B. R., & Satyanarayana, A. (2013). Teaching Data Mining in the Era of Big Data. *2013 ASEE Annual Conference.* King, B. R., & Satyanarayana, A. Teaching Data Mining in the Era of Big Data. In 2013 ASEE Annual Conference. http://www.asee.org/public/conferences/20/papers/7580/view.

Laney, D. (2001). *3D Data Management: Controlling Data Volume, Velocity, and Variety.* Application Delivery Strategies Metagroup.

Laney, D. (2012, January 14). *Deja VVVu: Others Claiming Gartner's Construct for Big Data.* Retrieved January 1, 2014, from Doug Laney: http://blogs.gartner.com/doug-laney/deja-vvvue-others-claiming-gartners-volume-velocity-variety-construct-for-big-data/

Manyika, J., Chui, M., Bughin, J., Dobbs, R., Roxburgh, C., & Byers, A. H. (2011, May). *Big data: The next frontier for innovation, competition, and productivity.* Retrieved

January 1, 2014, from McKinsey and Company: http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation

O'Reilly, T. (2005, September 30). *What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software.* Retrieved from O'Reilly Web 2.0: http://oreilly.com/pub/a/web2/archive/what-is-web-20.html?page=1

Programmable Web. (2012, August 29). *Three Techniques for Visualizing Data from Google BigQuery*. Retrieved from Programmable Web: http://www.programmableweb.com/news/three-techniques-visualizing-data-google-bigquery/how-to/2012/08/29

Russom, P. (2011). *Big Data Analytics.* Renton, WA: The Data Warehousing Institute.

Sato, K. (2012). Retrieved from An Inside Look at Google BigQuery: https://cloud.google.com/files/BigQueryTechnicalWP.pdf

StatisticsBrain. (2013). *Google Annual Search Statistics*. Retrieved from StatisticBrain.com: http://www.statisticbrain.com/google-searches/

Topi, H. (2013, March). Where is big data in your information systems curriculum? *ACM Inroads*, 12-13.

Editor's Note:

This paper was selected for inclusion in the journal as an ISECON 2014 Distinguished Paper. The acceptance rate is typically 7% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2014.

Appendix I

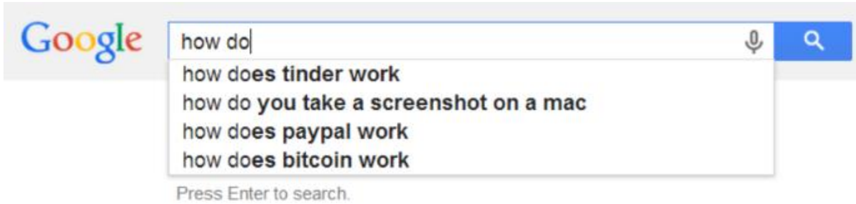


Figure 1. Google Autocomplete.

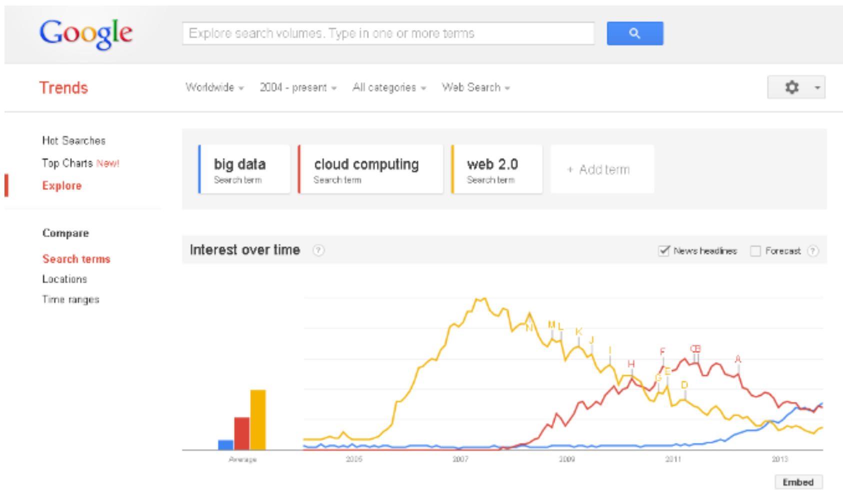


Figure 2. Google Trends.

Figure 3. Big Data allows Amazon to make purchase recommendations.

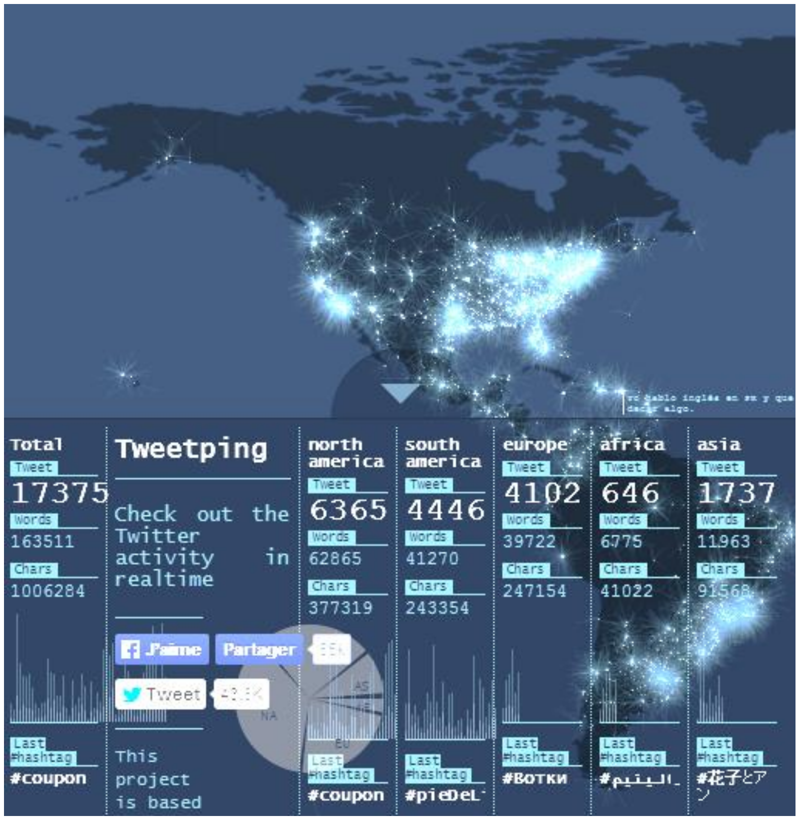


Figure 4. TweetPing is a Twitter visualization in real-time.

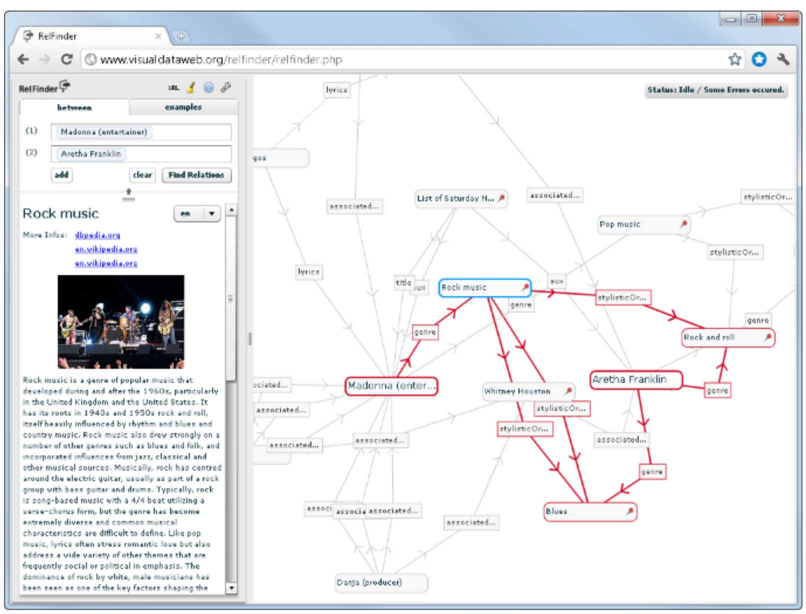


Figure 5. Relfinder determines relationships from text data stored in RDF format.

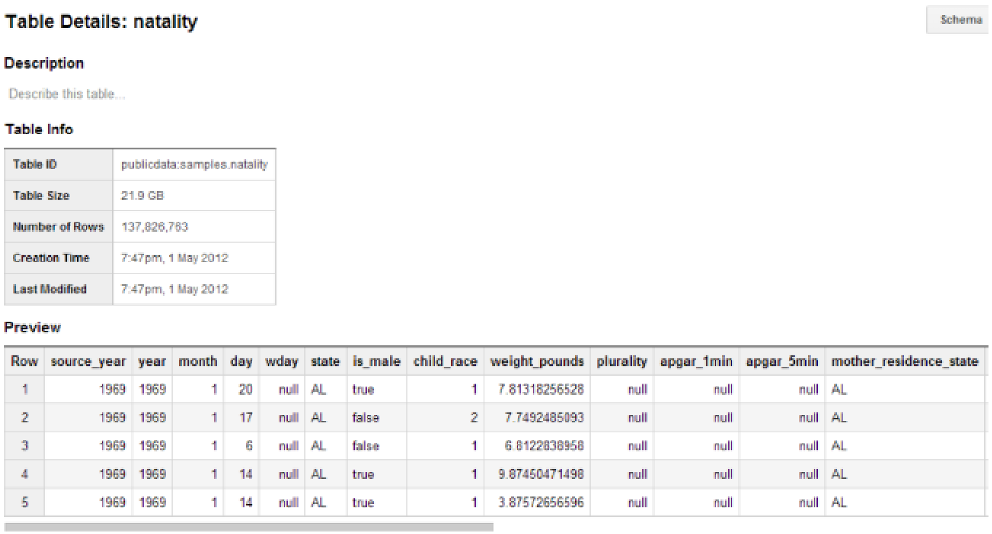


Table Details: natality

Schema

Description

Describe this table...

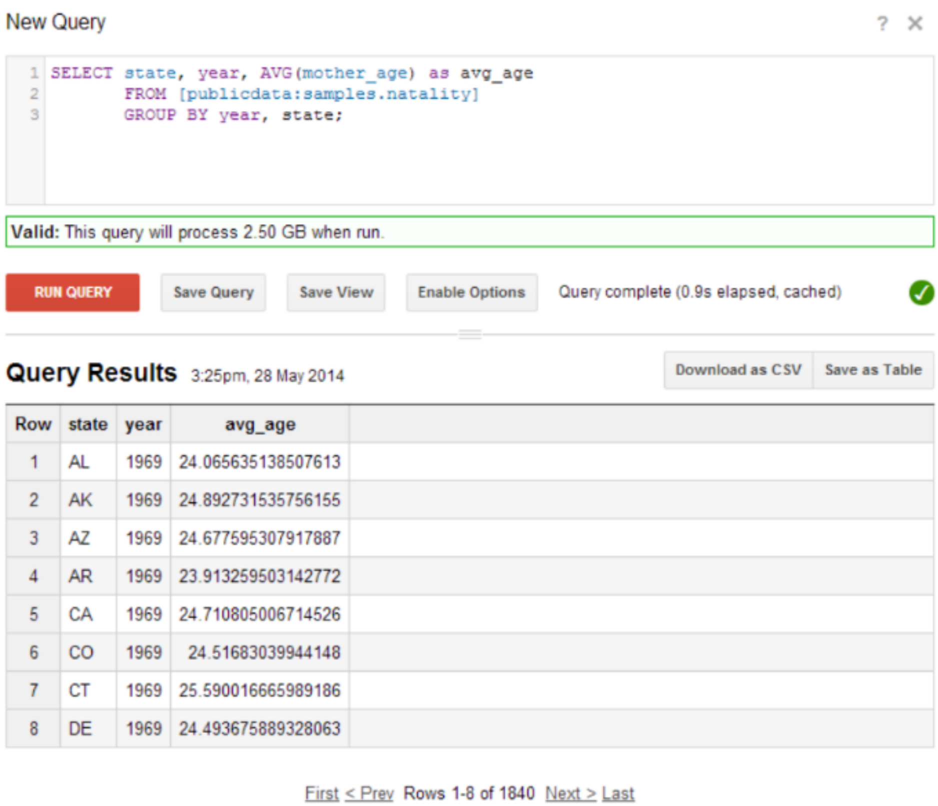
Table Info

Table ID	publicdata:samples.natality
Table Size	21.9 GB
Number of Rows	137,826,763
Creation Time	7:47pm, 1 May 2012
Last Modified	7:47pm, 1 May 2012

Preview

Row	source_year	year	month	day	wday	state	is_male	child_race	weight_pounds	plurality	apgar_1min	apgar_5min	mother_residence_state
1	1969	1969	1	20	null	AL	true	1	7.81318256528	null	null	null	AL
2	1969	1969	1	17	null	AL	false	2	7.7492485093	null	null	null	AL
3	1969	1969	1	6	null	AL	false	1	6.8122838958	null	null	null	AL
4	1969	1969	1	14	null	AL	true	1	9.87450471498	null	null	null	AL
5	1969	1969	1	14	null	AL	true	1	3.87572656596	null	null	null	AL

Figure 6. Details and sample natality information from Google BigQuery.



New Query

```
SELECT state, year, AVG(mother_age) as avg_age
       FROM [publicdata:samples.natality]
       GROUP BY year, state;
```

Valid: This query will process 2.50 GB when run.

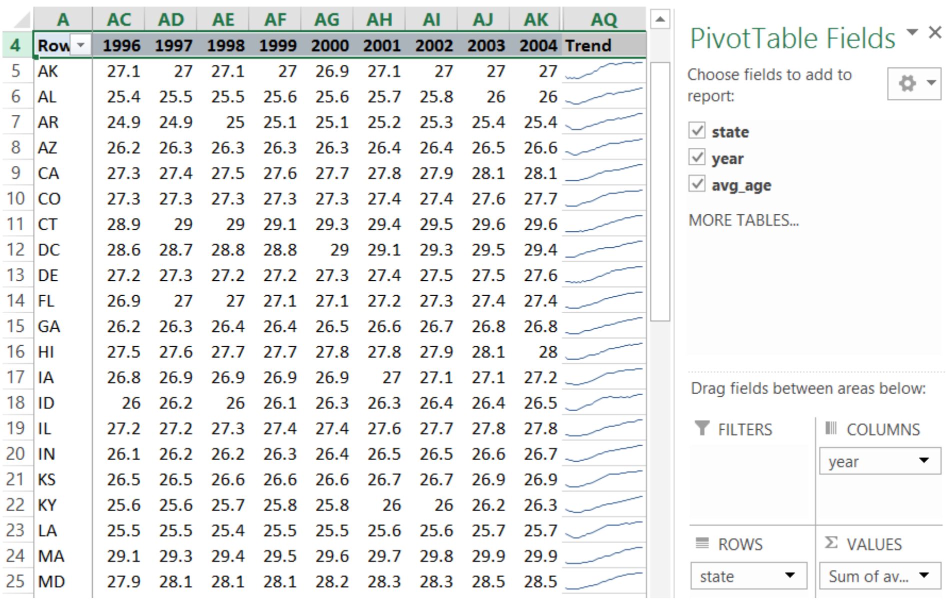
RUN QUERY | Save Query | Save View | Enable Options | Query complete (0.9s elapsed, cached)

Query Results 3:25pm, 28 May 2014 — Download as CSV | Save as Table

Row	state	year	avg_age
1	AL	1969	24.065635138507613
2	AK	1969	24.892731535756155
3	AZ	1969	24.677595307917887
4	AR	1969	23.913259503142772
5	CA	1969	24.710805006714526
6	CO	1969	24.51683039944148
7	CT	1969	25.590016665989186
8	DE	1969	24.493675889328063

First < Prev Rows 1-8 of 1840 Next > Last

Figure 7. Query and Results with Google BigQuery.



	A	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AQ
4	Row	1996	1997	1998	1999	2000	2001	2002	2003	2004	Trend
5	AK	27.1	27	27.1	27	26.9	27.1	27	27	27	
6	AL	25.4	25.5	25.5	25.6	25.6	25.7	25.8	26	26	
7	AR	24.9	24.9	25	25.1	25.1	25.2	25.3	25.4	25.4	
8	AZ	26.2	26.3	26.3	26.3	26.3	26.4	26.4	26.5	26.6	
9	CA	27.3	27.4	27.5	27.6	27.7	27.8	27.9	28.1	28.1	
10	CO	27.3	27.3	27.3	27.3	27.3	27.4	27.4	27.6	27.7	
11	CT	28.9	29	29	29.1	29.3	29.4	29.5	29.6	29.6	
12	DC	28.6	28.7	28.8	28.8	29	29.1	29.3	29.5	29.4	
13	DE	27.2	27.3	27.2	27.2	27.3	27.4	27.5	27.5	27.6	
14	FL	26.9	27	27	27.1	27.1	27.2	27.3	27.4	27.4	
15	GA	26.2	26.3	26.4	26.4	26.5	26.6	26.7	26.8	26.8	
16	HI	27.5	27.6	27.7	27.7	27.8	27.8	27.9	28.1	28	
17	IA	26.8	26.9	26.9	26.9	26.9	27	27.1	27.1	27.2	
18	ID	26	26.2	26	26.1	26.3	26.3	26.4	26.4	26.5	
19	IL	27.2	27.2	27.3	27.4	27.4	27.6	27.7	27.8	27.8	
20	IN	26.1	26.2	26.2	26.3	26.4	26.5	26.5	26.6	26.7	
21	KS	26.5	26.5	26.6	26.6	26.6	26.7	26.7	26.9	26.9	
22	KY	25.6	25.6	25.7	25.8	25.8	26	26	26.2	26.3	
23	LA	25.5	25.5	25.4	25.5	25.5	25.6	25.6	25.7	25.7	
24	MA	29.1	29.3	29.4	29.5	29.6	29.7	29.8	29.9	29.9	
25	MD	27.9	28.1	28.1	28.1	28.2	28.3	28.3	28.5	28.5	

Figure 8. Creating a visualization using a pivot table and sparklines (column AQ) in Excel.

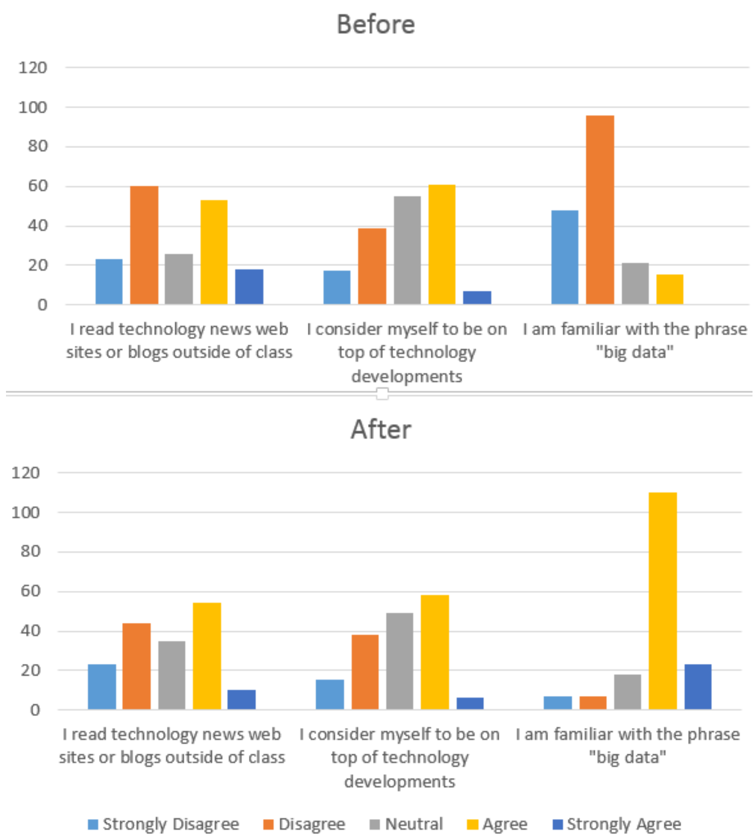


Figure 9. Big Data awareness.

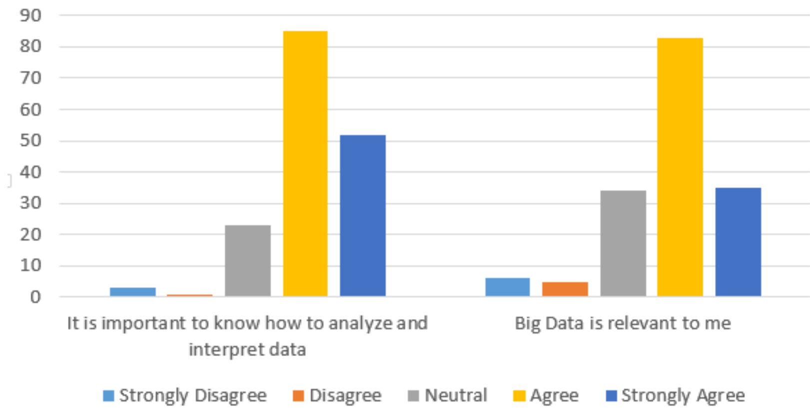


Figure 10. Student Perception of Relevance of Big Data

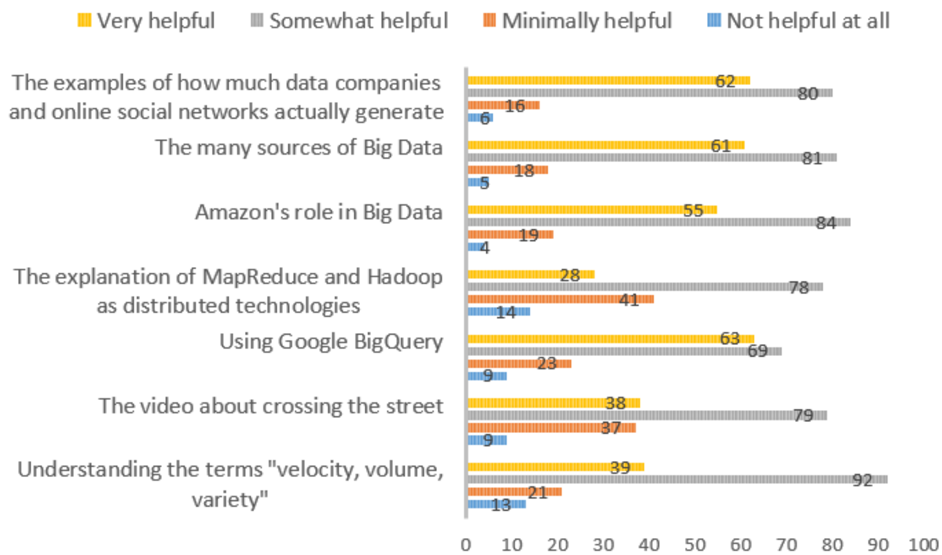


Figure 11. Student reaction to aspects of the Big Data lesson.

Teaching Non-Beginner Programmers with App Inventor: Survey Results and Implications

Andrey Soares
asoares@siu.edu

Nancy L. Martin
nlmartin@siu.edu

Information Systems Technologies
Southern Illinois University
Carbondale, IL 62901, USA

Abstract

This paper reports the results of a survey with 40 students enrolled in an Android Application Development course offered during the spring semester of 2013 and 2014. The course used App Inventor to build the apps and required students to have an introduction to programming course as a prerequisite. The survey asked for demographic information and students' opinions about prerequisites, App Inventor, previous programming skills, new concepts learned, teamwork, and more. The positive responses support the practice of using App Inventor to teach not only beginner programmers, but also more experienced programmers. The paper also shows that App Inventor can be used to support the teaching of more advanced computing concepts.

Keywords: App Inventor, Mobile Applications, Non-Beginner programmers, Survey.

1. INTRODUCTION

App Inventor is a visual programming language developed by Google in 2010 and currently hosted and maintained by the MIT Center for Mobile Learning. App Inventor has been successfully used to teach introductory programming concepts to beginners in both secondary and higher education courses (Abelson, 2012; Haungs, Clark, Clements, & Janzen, 2012; Robertson, 2014).

App Inventor can also be used to teach programming and other computing concepts for students that already have some programming experience (Gestwicki & Ahmad, 2011; Soares, 2014). For example, Gestwicki and Ahmad (2011) suggest that App Inventor and their Studio-Based Learning approach can be used not only to "introduce non-CS majors to concepts of Computer Science-not just programming, but also ideas that tend not to be covered in conventional CS1 courses such as human-computer interaction, incremental and iterative design processes, collaboration, evaluation, and quality assurance" (p. 55). Soares (2014) discusses issues, challenges and opportunities that instructors should be aware of when designing a course in mobile application development with an introductory programming course as prerequisite.

This paper presents the results of a survey with students enrolled in a mobile application development course that used App Inventor as the tool for teaching and building applications, and required an introduction to programming course as prerequisite.

The following section of the paper describes the methods used for data collection. The results and discussion of the data analysis are combined into the next section, and the paper closes with conclusions and recommendations.

2. METHODS

The course, Android Application Development, was offered during spring semester in 2013 and 2014, and had an introductory programming course as a prerequisite.

In order to explore students' perceptions about the course, the instructor developed a brief exploratory survey, and students were asked to complete it during the last week of the semester. The instructor explained that survey completion was voluntary and anonymous and was not in any way related to course grades. A total of 40 students enrolled in the course completed the survey, 16 in spring 2013 and 24 in spring 2014. The survey questions contained 37 unique data points that were designed to gather students' feedback on a variety of topics related to the course. Questions asked for demographic information and perceptions about course prerequisites, the App Inventor tool, the reinforcement of programming fundamentals, concepts learned, teamwork, and an interest in learning more about mobile application development.

All questions, other than demographic related ones, were answered on a 5-point Likert scale ranging from strongly agree to strongly disagree. Since the survey was exploratory in nature with no predicted outcomes, the data was analyzed using descriptive statistics, primarily frequency analysis.

In the next section, the questions and their data analysis are presented, along with a relevant discussion of each topic.

3. RESULTS AND DISCUSSION

Demographics
Forty students completed the survey during the spring semester in 2013 and 2014. Table 1 displays the breakdown by gender and class. Males comprised 88 percent of the sample; females comprised 10 percent, and one person did not report gender. The students were either juniors (22.5 percent) or seniors (77.5 percent). As the course has the Introduction to Programming course as a prerequisite and is currently not required as prerequisite for any other course in the program, students generally take it in their third or fourth year.

	Class		
Gender	Junior	Senior	Total
Female	0	4	4
Male	9	26	35
Missing	0	1	1
Total	9	31	40

Table 1: Sample demographics

Prerequisites
Although an introductory programming course was a prerequisite for the Android course, students were asked whether they agreed or disagreed with the reference text's statement that no programming experienced is required. As Figure 3 reflects, over 46 percent agreed that no programming experience was required, while only 18 percent disagreed and 36 percent were neutral.

Students were also asked whether they believed certain courses should be prerequisites for the Android course. The results of that question are shown in Figure 4. Interestingly, even though about 46 percent of students agreed that no programming experience was required for the course, about 43 percent agreed that Programming II should be required. A database course was also identified by about 43 percent of students as a recommended prerequisite.

Because some assignments were completed in groups requiring close collaboration, some students even considered Project Management (21%) and Software Engineering (15%) as prerequisites for the course. Almost all assignments resulted in a new app created from scratch, with some exceptions where students improved their existing apps. That means, for each app they were supposed to plan, design, implement and test their apps. When working in groups, students will most likely deal with scope definition, scheduling, task management, communication, human resources and other activities needed to complete their apps.

App Inventor Tool for Beginners and for Experienced Programmers
Respondents overwhelmingly agreed that App Inventor is a great tool for teaching both beginners and more experienced programming students. Figure 5 shows that 87.5 percent agreed or strongly agreed the tool was good for beginners, and 85 percent reported the same opinion related to those with some programming experience.
App Inventor provides developers with the ability to quickly design and implement an app

using a variety of features from a mobile device. The use of a visual tool reduces code-related distractions (e.g., missing semi-colon, braces or misspelled code) as students create the application with blocks of code (Soares, 2014). The tool hides some of the complexity of the code by providing predefined blocks for specific functions. Figure 6 shows examples of the event Click for the component Button1 both as a visual code and as a textual code. When the component Button1 is created, the block "When-Button1.Click-do" and several others are automatically created and are available for students to just drag-and-drop it into the blocks editor.

Because of the blocks, students do not have to worry about the syntax of the code, and they can focus "more on the functionalities of the application and what can be done with the phone" (Soares, 2014, p. 59). Nonetheless, they still need to know about the logic of programming to complete the application, especially the event-driven programming approach. Ninety-five percent of the respondents agreed/strongly agreed that App Inventor helped them learn about developing mobile applications, and 85 percent demonstrated interest in learning more about developing mobile applications for smartphones and tablets. When asked if writing the code in Java to create applications would be preferred over the App Inventor, 47.5 percent of students disagreed/strongly disagreed, 27.5 percent were neutral, and 25 percent agreed/strongly agreed.

Similar to others' experience (Robertson, 2014) it appears that overall, both more and less experienced students found value in the use of App Inventor.

Reinforcing Fundamentals of Programming

Most students agreed that it was easy to apply previous programming concepts to the App Inventor environment. Figure 7 shows that 85 percent of the respondents agreed or strongly agreed.

Students also generally agreed that developing mobile applications with App Inventor helped reinforce fundamentals of their programming knowledge. Figure 8 displays the responses related to several programming fundamentals. The first three are the use of variables, conditions, and loops. Eighty percent of students agreed or strongly agreed that their knowledge of variable use improved, and 82.5 felt the same about the use of conditions. Fifty-five percent reported knowledge reinforcement of loops.

Another area of previous programming knowledge reinforced in class is the use of procedures. Figure 8 also displays student opinions about this topic. Regarding the use of procedures with a return value, 60 percent of students agreed/strongly agreed that App Inventor helped them to reinforce the knowledge on how to use procedures. Thirty percent were neutral and 10 percent disagreed/strongly disagreed. Regarding the use of procedures that do not return any value, 77.5 percent of students agreed/strongly, 17.5 were neutral and 5 percent disagreed/strongly disagreed that App Inventor was a useful tool.

In some cases, students would create a global variable and would use a procedure to change its value, rather than making the procedure to return a value. Creating input arguments to allow data to be passed on to the procedures was also challenging, and often students would use global variables or read data from existing components as a workaround. When the data to be passed was related to a component, some students explored the use of the advanced blocks, which allows applications to work with components dynamically.

Working with Lists (Arrays)

More than 80 percent of students reported that mobile application development with App Inventor strengthened their knowledge of both single lists (i.e., arrays) and multidimensional lists (i.e., multidimensional arrays). Specific response percentages are shown in Figure 9.

Arrays are considered a difficult topic for students to learn (Dale, 2006; Lahtinen, Ala-Mutka, & Jarvinen, 2005). Considering the prerequisite of an introduction to programming course, "even students that are already familiar with the concepts of arrays may need a period of adjustment to translate and adapt their prior knowledge with arrays into the new environment" (Soares, 2014, p. 61). Nonetheless, students taking the Android Application Development course may have different levels of programming experience. For example, arrays may not have been covered in the introduction to programming course and thus will be a new topic, or students may have taken other programming courses that cover the topic of arrays and be fairly familiar with the topic.

The concept of lists is introduced early in the course because it is required for several assignments, either as the main focus of an assignment or as part of a larger application. For example, an extended version of the App Inventor's "Map It" tutorial uses lists to store information about the location (e.g., name, latitude and longitude) of points of interest on campus and around town. The app starts with a pre-defined list of 10 locations selected by the students (i.e., static list), but users of the app can also add new items to the list or delete existing ones (i.e., dynamic list).

App Inventor has a variety of functions built-in to work with lists, but these functions generally relate to single lists. To work with multidimensional lists, students must write additional code or create specific functions to work with them.

Web Services

Besides learning how to build applications that work with the features of the Android mobile devices, students are also interested in creating apps that can interact with other web applications (Soares, 2014). Not surprising, "the Web is evolving into a dynamic repository of information on virtually every topic, including people and their connections to one another as well as to content" (Ramakrishnan & Tomkins, 2007, p. 63). And, many people, students included, seek opportunities to be content producers (e.g., share their data) and content consumers (e.g., retrieve data from other sources on the Web).

App Inventor has several components (i.e., TinyWebDB, ActivityStarter, FusionTables, Twitter, and Web) that permit developers to incorporate Web Services and APIs into the applications, and permit instructors to use them to teach a variety of topics. For example, the Google Maps API is explored in combination with the component Location Sensor, which reads global positioning system (GPS) coordinates from the device. Note that a developer can interact with the Google Map app installed on the phone, but can also interact with the Google Map API available on the Web. Lim, Jong, and Mahatanankoon (2005) discuss the potential of integrating Web Services earlier into the curriculum to make the course more interesting and to expose students to Web Services and its potential "to speed up application development and reduce costs to access data on disparate systems" (p.241).

This course included several assignments using Web Services and APIs within the apps such as displaying driving directions on a map or a pie chart with data from an online survey app, returning information about a product after scanning a UPC code or an ISBN code, getting the weather forecast or the list of businesses for a given zip code, displaying Bible passages or displaying products for sale from Craigslist. A great source of Web Services and APIs is the website www.programmableweb.com with a list of over 11,500 APIs (as of July, 2014) that can be used to build applications. However, Soares (2014) cautions about the different formats of responses returning from the APIs (e.g., XML, JSON, etc.) and the need to teach students how to parse the responses in order to use the information needed for their apps.

Despite the issue with parsing Web Services responses, when asked if developing mobile applications with App Inventor helped in learning about Web Services, over 76 percent of students agreed or strongly agreed (see Figure 10).

Animation and Sensors

Designing games is a great approach to learn about mobile application development, especially about what the phone can do in terms of interacting with users. Our results show that more than 85 percent of students agreed that they learned about animation (see Figure 10).

The basic tutorials such as PaintPot, MoleMash and Ladybug Chase (see www.appinventor.org) are great introductions to drawing and animation components as they expose students to several functionalities of the phone such as touching, dragging, and tilting. Because of the relatively small amount of code needed to work on the tutorials, students usually get excited about creating their own games. Of course, some game ideas are too complex and will require students to combine several concepts learned throughout the course. Since this course started being offered in spring 2012, many of the students' final projects utilized some of the drawing and animation components; for example, a chess game played over Bluetooth, a flight combat game, a breakout game, and more.

The component Clock is also used to implement games and other applications that require control of time (e.g., time left to play) or need to take actions repeatedly (e.g., move an object every tick of the clock). In addition, the Accelerometer sensor, Location sensor, and Orientation sensor can be added to the applications to improve the users' experience.

Table 2 presents a list of events and parameters for the sensor components. The location sensor works with the GPS and provides information on latitude, longitude, altitude, and address; the orientation sensor provides information about the phone's orientation (i.e., tilt and direction pointing to); and the accelerometer sensor detects acceleration using the X, Y, and Z dimensions as well as the shaking of the phone.

Accelerometer	AccelerationChanged: xAccel yAccel zAccel Shaking
Location	LocationChanged: latitude longitude altitude StatusChanged: provider status
Orientation	OrientationChanged: azimuth pitch roll

Table 2: Events and parameters for the sensor components

Participants were asked if developing mobile applications with App Inventor helped them to learn about sensors. About 88 percent agreed/strongly agreed they learned about Location sensor, 85 percent agreed/strongly agreed they learned about Orientation sensor, and 87.5 percent agreed/strongly agreed they learned about Accelerometer sensor.

Event-Driven Programming

Even though events are considered an important programming concept, they "are typically taught late in the CS curriculum" (Turbak, Sherman, Martin, Wolber, & Pokress, 2014, p. 81). Because the prerequisite for this course is an introduction to programming course, students may not be familiar with the concept of events or event-driven programming as it may not have been covered in the prerequisite course.

Wolber, Abelson, Spertus, and Looney (2011) explain that "with App Inventor, you design how an app looks and then you design its behavior—the set of event handlers that make an app behave as you want" (p.227). They describe that an app responds to user-initiated events, initialization events, timer events, animation events or external events. Students, especially novice programmers may find it difficult to identify all the events of a behavior (Soares, 2014; Wolber et al., 2011).

In this course, events are introduced in the first week of class and then reinforced throughout the course with examples, in-class discussions, and lab assignments. In addition, for some assignments, students are required to design mockup screens of the applications to be created, which helps them "to think not only about the components but also about the underlying events, functions and blocks that need to be used to achieve the desired results" (Soares, 2014, p. 59). In our survey, 82.5 percent of students agreed/strongly agreed that developing mobile applications with App Inventor helped them to learn about event-driven programming (see Figure 10).

Database

The topic of database is covered in the course in two ways. First, we discuss the phone's internal database and we use the component TinyDB to store and retrieve data from the local database. Second, we discuss the use of web databases, starting with the component TinyWebDB and later exploring the component Fusiontables. Both TinyDB and TinyWebDB are fairly straightforward since the developer simply uses tags to store and retrieve data from the databases. Fusiontables, on the other hand, has more complexities and requires developers to work with Google Drive and Google Fusiontables API in order to create a table and make it available to integrate with an app. During implementation with App Inventor, developers must understand the basics of database design and Structured Query Language (SQL) to query the tables. With Fusiontables, developers can use commands to insert, update, delete and select data from the tables. As Soares (2014) describes, "the query results are in CSV or JSON formats and can be transformed into lists with the appropriate blocks in App Inventor" (p.61).

When asked if they would recommend a database course as a prerequisite for the Android Application Development course, 43 percent of students agreed/strongly agreed, 18 percent were neutral, and 41 percent disagreed/strongly disagreed (see Figure 4). Considering that the majority of students are seniors and have likely taken a database course, these answers are rather surprising, especially since 80 percent of students agreed/strongly agreed that they learned about web databases in the course. On one hand, students may have previous knowledge of database design and SQL, and they considered the learning of how to work with TinyWebDB and Fusiontables during the course. On the other hand, students may be new to the concept of database, and they considered learning about the concept during the course.

User Interface and Input/Output
One of the main reasons students take the Android application development course is the excitement of building their own apps. It is definitely fun, however some students feel overwhelmed by the process of creating an app and focusing on the user interaction with the app and all the necessary validations and tests involved.

As discussed earlier, the visual programming approach of App Inventor helps to hide some of the complexities of programming, providing students with opportunities to concentrate on the design of the application, its features, and how users will interact with it. That means, students should learn and practice the design of user interfaces, user input and output, and input validation as they play an important role in the user's experience with mobile applications.

More often than not, students will detect some problems with their apps that can be a result of poor user interface design. In particular, user inputs are overlooked which will make apps misbehave or crash when users enter unexpected data or do not provide any data. Of course, App Inventor provides several properties to the components to allow developers to set up the application as needed. For example, the component TextBox can be set to number only in order to restrict the type of data entered. However, it is the developer's responsibility to define a range of acceptable numbers and to create the appropriate code to validate it. Other properties such as enable/disable and visible/hidden provide ways for developers to customize their apps. Soares (2014) suggests the use of mockup screens during the planning of applications to help define the apps' user interfaces and behaviors.

Students were asked whether they believed that developing mobile applications with App Inventor helped them to learn about user interface design, user input and output, and input validation. Ninety percent of students answered that they agreed/strongly agreed to have learned about user interfaces, and 77.5 percent of students agreed/strongly agreed to have learned about user input validation (see Figure 10). In addition, 87.5 percent of students agreed/strongly agreed that they reinforced their knowledge about handling user input, and 85 percent of students agreed/strongly agreed their knowledge of user output was strengthened.

Connections and Data Communication
According to a survey by the Pew Research Center's Internet & American Life Project (Duggan & Smith, 2013), "six in ten cell phone owners (63%) now go online using their mobile phones, an eight-point increase from the 55% of cell owners who did so at a similar point in 2012" (p.4). It is not surprising that people are spending more time on their phones and using it mainly for some sort of communications (e.g., with another person or a web/mobile application). In fact, besides making phone calls, the most popular cell phone activities are (Duggan, 2013, p. 2):

- Send or receive text messages (81%)
- Access the internet (60%)
- Send or receive email (52%)
- Download apps (50%)
- Get directions, recommendations, or other location-based information (49%)
- Listen to music (48%)
- Participate in a video call or video chat (21%)
- Check in or share your location (8%)

Students in this course have demonstrated great interest in creating applications that go beyond the capabilities of the mobile device and explore approaches to connect and communicate with other people, devices and applications. During the course, students had the opportunity to use some of the App Inventor components that support connectivity and communication. For example, the Image component can link directly to an image using its URL; the component WebViewer permits the display of a webpage for a specific URL; the component PhoneCall makes a call to the phone number specified; the component Texting permits users to send and receive text messages from other devices; the component FusionTablesControl permits the app to interact with tables stored on Google Drive; the component TinyWebDB connects with a web service that provides database services; the component ActivityStarter permits an app to open an application such as a browser or a map; the components BluetoothClient and BluetoothServer support communication of paired mobile devices; and the Web component supports the use of HTTP methods (e.g., POST and GET) for request-response connections.

Instructors can benefit from these components, when designing course content and assignments, and should encourage students to explore the features of the phone that support connections and data communications. Most students agreed/strongly agreed that developing mobile applications with App Inventor helped them to learn about Bluetooth communication (90%), Web Databases (80%), Web Services (72.5%), and Text Messaging (65%).

Teamwork

Working in groups with App Inventor is not an easy task. Students can work together to plan their apps and share ideas, but when it is time to design and implement their apps, the tool has some limitations that make group work challenging. For example, blocks cannot be copied from one project to another, and two or more people cannot work on the same project at the same time. Even with the limitations of App Inventor to support collaborative work, 68 percent of the respondents agreed/strongly agreed they enjoyed working in teams to develop mobile apps. The course included several group assignments with group sizes of 2 to 5 students.

Besides teamwork, many assignments also included time and scope constraints to challenge and persuade students to manage their tasks and progress. Students had to come up with their own approach to make a group assignment work. For example, groups spent more time planning the app and creating mockup screens to define the components and functionalities needed before starting any code. On some occasions, they even discussed how to name the components so that other group members could easily find them.

Some groups decided to separate their activities and each member would work on their own to complete their respective tasks. After that, they would create a shared account to access App Inventor and then each member would take turns implementing their part of the project. One group tried to get all members logged in on App Inventor at the same time to work on the same project using the same user account. However, they quickly learned that the current version of App Inventor does not support synchronous collaboration. For some groups, each member would work individually on their tasks and send their work to a member that was responsible to combine all parts into one project. Finally, for other groups, the approach was one member working on the computer and the other members around him or her discussing the project and providing support during the implementation.

Learning More about Mobile Application Development

The Android application development course can be considered a success, with great feedback from students, great student evaluations, and students showing interest in learning more about mobile application development. Two students that took the course created their apps, alone or in teams, to enter in the university's App competition.

When asked if, after the course, students would be interested in learning more about developing mobile applications for smartphones and tablets, 85 percent of participants answered that they agreed/strongly agreed and 12.5 percent were neutral. However, some students would prefer to learn how to develop apps using Java (25%).

4. CONCLUSION AND RECOMMENDATIONS

This paper presented the results of a survey with students enrolled in an Android Application Development course with an introduction to programming course as prerequisite. The results show positive feedback from the students about course prerequisites, App Inventor, reinforcing fundamentals of programming, learning new concepts, teamwork, building mobile apps, and more. The paper presented a discussion of the survey results and some recommendations related to the use of App Inventor to teach beginners and more experienced programmers as well as to teach other advanced computing concepts.

App Inventor has been used successfully for teaching beginner programmers from elementary school to higher education and from CS/IS to other majors (Gray, Abelson, Wolber, & Friend, 2012; MacKellar, 2012; Wolber, 2011). In fact, the authors of this paper have experienced firsthand the potential of using App Inventor to introduce programming skills to beginner programmers by offering a summer camp to middle school girls on mobile app development.

The visual programming approach of App Inventor helps students to learn about programming concepts and to apply their existing skills to build mobile apps. As the survey shows, 87.5 percent agreed or strongly agreed the tool was good for beginners and 85 percent agreed or strongly agreed that it was easy to apply previous programming knowledge to the App Inventor environment. Some students even mentioned during the course that they would prefer to have the Android app development course as the first introduction to programming course rather than a Java course. The reward of seeing quick results and not dealing with the code behind the blocks seems engaging to beginners, but a possible disadvantage for more skilled programmers.

Students who took an introductory programming course prior to this course had the opportunity to apply and reinforce their knowledge on the fundamentals of programming, and also to learn new programming approaches and computing concepts. According to Soares (2014), "the time used for teaching logic and the fundamentals of programming could be used to explore more features of the phone and the App Inventor tool" (p.58). It is not surprising to see that students agreed/strongly agreed that App Inventor is a great tool for teaching more experienced programmers (85%). Because of the assortment of features available on mobile devices and the relatively easy way to handle them with App Inventor, instructors can design course assignments that use basic programming skills but also require the application of more advanced skills to build the apps. Each app created with App Inventor provides students with opportunities to practice different phases of the development process, apply different programming skills, and use different features of the mobile device.

With the help of App Inventor, a mobile development course should be fun and packed with several computing concepts besides programming, such as database, data communication, software development, project management, mobile applications development, web services and more. Now that students have built a background on app development, instructors teaching more advanced courses can illustrate the concepts of their specific courses with the support of App Inventor. For example, in a database course, students could create forms to insert data into tables or display data from the tables using both static and dynamic queries. In a software engineering or systems analysis and design course, students could benefit from App Inventor's support for rapid development to plan, design, implement and test mobile apps as part of course assignments or projects. In particular, instructors could explore principles and techniques for user interface design. Also, it should not take long to find units on campus or other organizations that could use apps for their business and would be interested in collaborating with students through class projects. The Bluetooth communication could be used, for example, in a biomedical or heath information technology course, where students could read data from medical equipment to display the status of the machines or the patients connected to the machines. For a network and security course, the Web capability of App Inventor could be used to monitor and communicate the status of network devices.

5. REFERENCES

Abelson, H. (2012). *From computational thinking to computational values*. Paper presented at the Proceedings of the 43rd ACM technical symposium on Computer Science Education, Raleigh, North Carolina, USA.

Dale, N. B. (2006). Most difficult topics in CS1: results of an online survey of educators. *SIGCSE Bulletin, 38*(2), 49-53. doi: 10.1145/1138403.1138432

Duggan, M. (2013). Cell Phone Activities 2013. Washington, D.C.: Pew Research Center's Internet & American Life Project - available from http://www.pewresearch.org.

Duggan, M., & Smith, A. (2013). Cell Internet Use 2013. Washington, D.C.: Pew Research Center's Internet & American Life Project - available from http://www.pewresearch.org.

Gestwicki, P., & Ahmad, K. (2011). App inventor for Android with studio-based learning. *Journal of Computer Science in Colleges, 27*(1), 55-63.

Gray, J., Abelson, H., Wolber, D., & Friend, M. (2012). *Teaching CS principles with app inventor*. Paper presented at the Proceedings of the 50th Annual Southeast Regional Conference, Tuscaloosa, Alabama.

Haungs, M., Clark, C., Clements, J., & Janzen, D. (2012). *Improving first-year success and retention through interest-based CS0 courses*. Paper presented at the Proceedings of the 43rd ACM technical symposium on Computer Science Education, Raleigh, North Carolina, USA.

Lahtinen, E., Ala-Mutka, K., & Jarvinen, H.-M. (2005). *A study of the difficulties of novice programmers*. Paper presented at the Proceedings of the 10th Annual Conference on Innovation and Technology in Computer Science Education (SIGCSE), Caparica, Portugal.

Lim, B. B. L., Jong, C., & Mahatanankoon, P. (2005). *On integrating Web services from the ground up into CS1/CS2*. Paper presented at the Proceedings of the 36th Technical Symposium on Computer Science Education (SIGCSE), St. Louis, Missouri, USA.

MacKellar, B. (2012). *App Inventor for Android in a Healthcare IT course*. Paper presented at the Proceedings of the 13th Annual Conference on Information Technology Education, Calgary, Alberta, Canada.

Ramakrishnan, R., & Tomkins, A. (2007). Toward a PeopleWeb. *Computer, 40*(8), 63-72. doi: 10.1109/mc.2007.294

Robertson, J. (2014). Rethinking how to teach programming to newcomers. *Communication of the ACM, 57*(5), 18-19. doi: 10.1145/2591203

Soares, A. (2014). Reflections on teaching App Inventor for non-beginner programmers: Issues, challenges and opportunities. *Information Systems Education Journal, 12*(4), 56-65. (A preliminary version appears in The Proceedings of ISECON 2013).

Turbak, F., Sherman, M., Martin, F., Wolber, D., & Pokress, S. C. (2014). Events-first programming in App Inventor. *Journal of Computer Science in Colleges, 29*(6), 81-89.

Wolber, D. (2011). *App Inventor and real-world motivation*. Paper presented at the Proceedings of the 42nd ACM Technical Symposium on Computer Science Education, Dallas, TX, USA.

Wolber, D., Abelson, H., Spertus, E., & Looney, L. (2011). *App Inventor: Create your own Android Apps*: O'Reilly Media.

APPENDIX

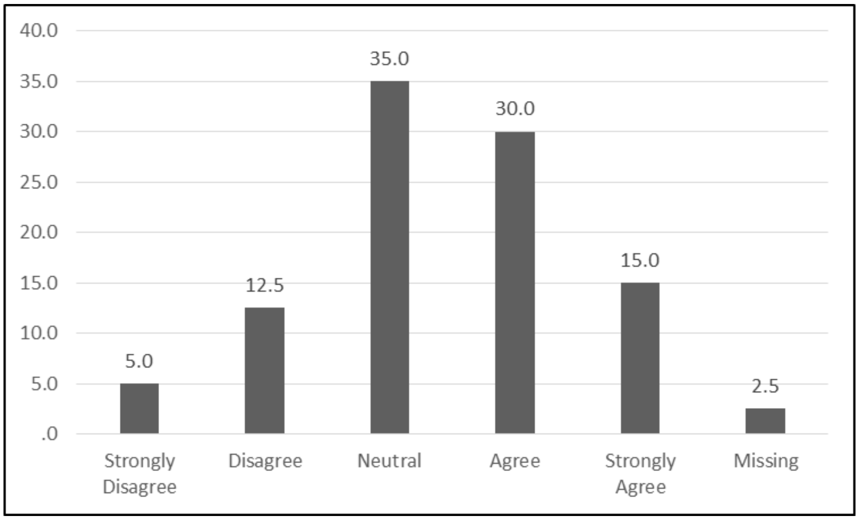


Figure 3: Student response to "no programming experience required".

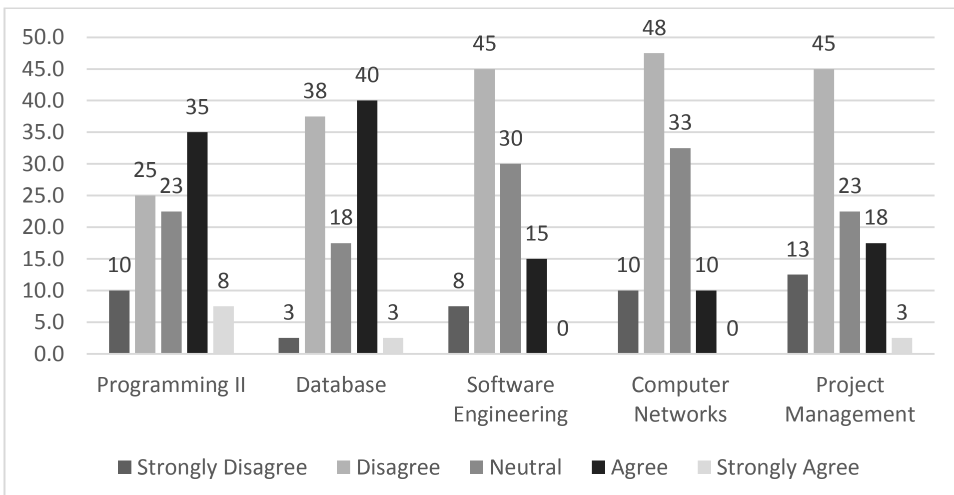


Figure 4: Student responses about Android course prerequisites (in percentages).

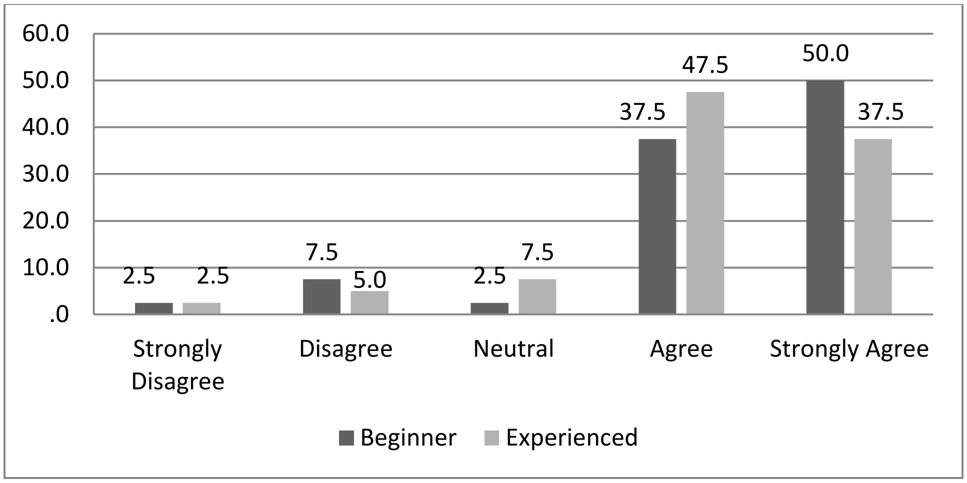
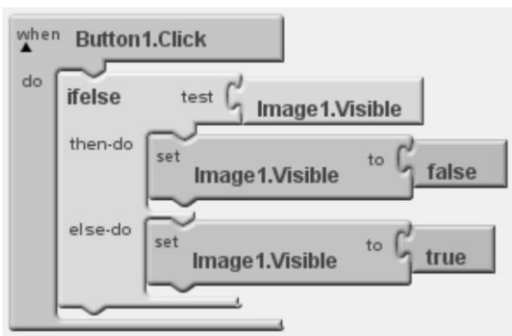


Figure 5: Student responses about App Inventor as a useful tool (in percentages).




```
Button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(Image1.isShown()) {
            Image1.setVisibility(ImageView.INVISIBLE);
        }
        else {
            Image1.setVisibility(ImageView.VISIBLE);
        }
    }
});
```

Figure 6: Handling the event of a button clicked using Visual code (left) and Textual code (right) (Soares, 2014, p. 58).

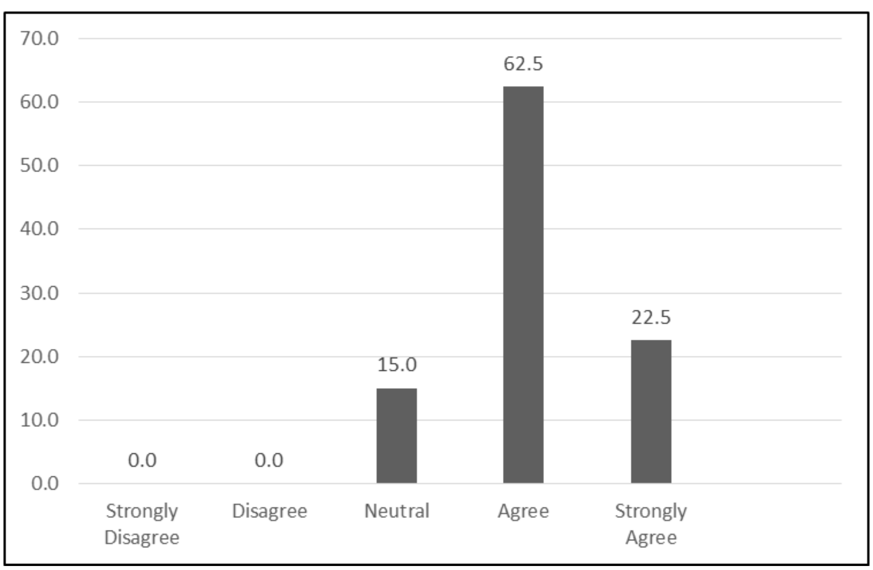


Figure 7: Student response to the ease of applying prior programming knowledge.

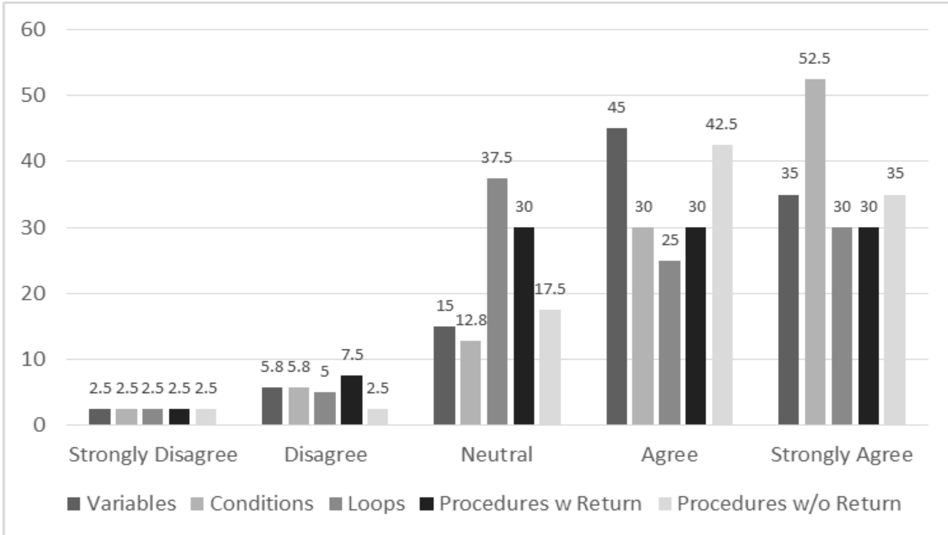


Figure 8: Student responses about the reinforcement of fundamentals (in percentages).

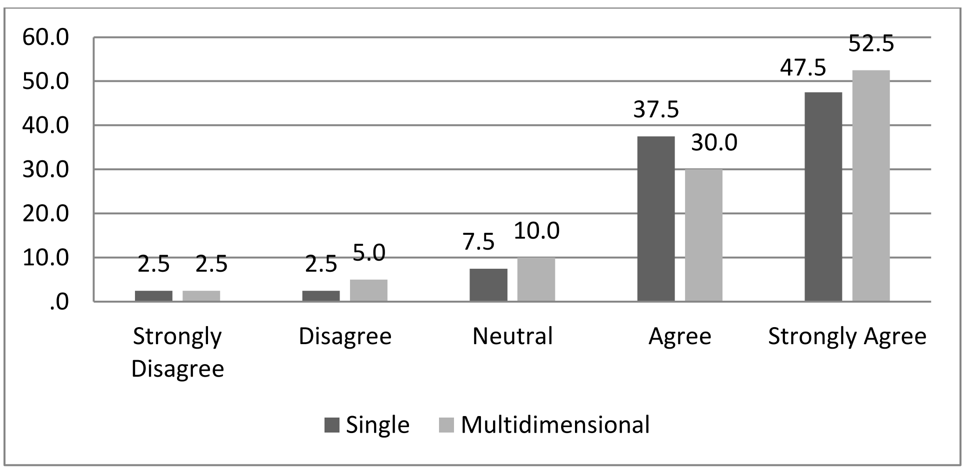


Figure 9: Student responses that App Inventor reinforced knowledge of Lists (in percentages).

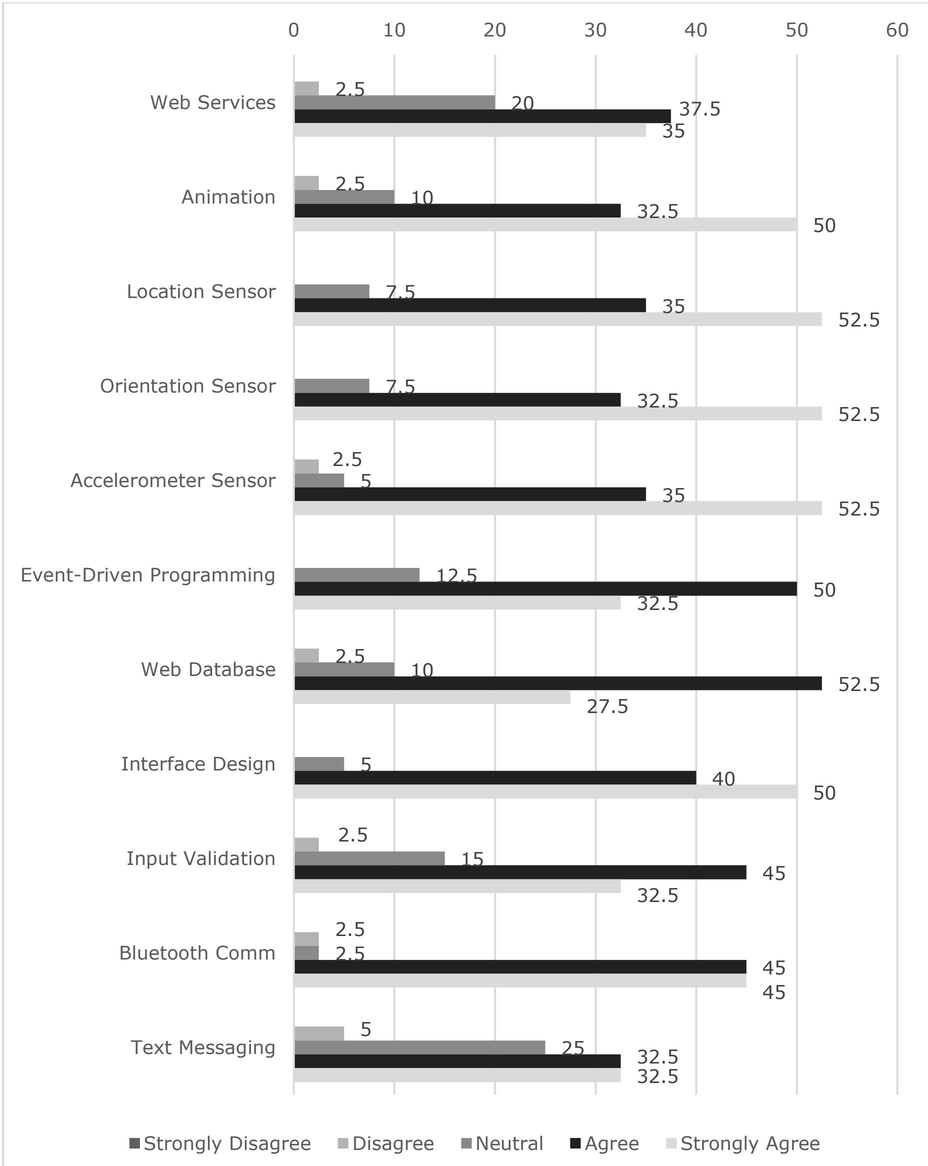


Figure 10: Student response to learning about various topics (in percentages).

Establishing the Basis for a CIS (Computer Information Systems) Undergraduate Program: On Seeking the Body of Knowledge

Herbert E. Longenecker, Jr.
longeneckerb@gmail.com
University of South Alabama
Mobile, AL 36688, USA

Jeffry Babb
jbabb@wtamu.edu
West Texas A&M University
Canyon, TX 79016, USA

Leslie J. Waguespack
lwaguespack@bently.edu
Bentley University
Waltham, Massachusetts 02452, USA

Thomas N. Janicki
janickit@uncw.edu
University of North Carolina Wilmington
Wilmington, NC 28403, USA

David Feinstein
dfeinstein@southalabama.edu
University of South Alabama
Mobile, AL 36688, USA

Abstract

The evolution of computing education spans a spectrum from *computer science (CS)* grounded in the theory of computing, to *information systems (IS)*, grounded in the organizational application of data processing. This paper reports on a project focusing on a particular slice of that spectrum commonly labeled as *computer information systems (CIS)* and reflected in undergraduate academic programs designed to prepare graduates for professions as software developers building systems in government, commercial and not-for-profit enterprises. These programs with varying titles number in the hundreds. This project is an effort to determine if a common knowledge footprint characterizes CIS. If so, an eventual goal would be to describe the proportions of those essential knowledge components and propose guidelines specifically for effective undergraduate CIS curricula. Professional computing societies (ACM, IEEE, AITP (formerly DPMA), etc.) over the past fifty years have sponsored curriculum guidelines for various slices of education that in aggregate offer a compendium of knowledge areas in computing. This paper describes a project to determine the subset of that compendium pertinent to CIS. The project began by surveying experienced academic curriculum designers self-identified with the CIS perspective. The pilot survey results reported herein indicate that many essential knowledge areas of CIS are shared with published IS curricular guidelines but, design and implementation of database systems, software development and project management are distinctive in CIS. The next project phase launches a revised survey suitable for a general audience of computing academics. The intention is to triangulate the perspectives of a widely varied population of computing academics to further crystalize the distinctiveness of CIS as a well-formed closely related discipline of IS with a core of necessary knowledge and skills – then to develop curricular guidelines for undergraduate CIS education.

Keywords: CIS, computer information systems, model curriculum, body of knowledge.

1. INTRODUCTION

This paper explores a proposition that a band within the spectrum of computing disciplines exists which is known as Computer Information Systems. However, this evolving branch of Information Systems (IS) exists largely through self-identification. While Computer Information Systems (CIS) has been somewhat validated in the literature as a "strain" of a discipline known as Information Systems (IS), it is a casual association of self-selection (Shackleford, et al. 2005). IS's interdisciplinary nature explains the numerous and varied attempts to describe its essence and purpose as an academic discipline (Alter, 1998; Checkland & Howell, 1997; Davis & Olson, 1984; Orlikowski & Iacono, 2001; Palwak, 1981). Likewise the breadth of issues and aspects constituent reference disciplines attributable to IS is a challenge to any widespread or consistent adoption of curriculum guidelines for IS programs.

This paper aims to answer these undergraduate curriculum questions:

a) Does CIS reflect a distinct body of knowledge that constitutes an academic discipline?
b) What are the nature, characteristics, aims, goals, expectations, and assurances of CIS?
c) As young as the scientific discipline computing is, what place does CIS hold within it, if any?

In search of answers to the above questions, we first briefly review the origins of computing to highlight its continuing evolution and diversification. We review a brief history of computing curricula and attempts along the way to structure academic curricula to support the evolving education and training of computing professionals: universal connectivity, ubiquitous computing and pervasive organizational dependence on information systems. The remainder of the paper sets the stage for a process our fundamental question might be answered: is there "such a thing" as the Computer Information Systems discipline?

We base the rationale for this exposition upon the extant models of curricula that offer a variety of perspectives to choose from (IS2002, CC2005, IS2010). With CC2005 as a seminal foundation, we attempt to reconcile among the various Knowledge Areas from the Bodies of Knowledge endorsed by computing professional societies over the past fifty years. We propose that compendium of computing knowledge areas and skills (CKS) as the superset, the starting point, from which a subset may be identified to characterize CIS. Towards this end, we enlist the input from a small advisory group experienced in curriculum development. In an extensive survey instrument, each advisor rated each item in the CKS for the desired depth of knowledge, learning outcome orientation (theory, principle, innovation vs. application, deployment, configuration [CC2005]), area of computing practice (organizational, application, software systems, infrastructure, architecture [CC2005]), and finally, the raters' confidence in assessing personal competence in evaluating the KA. The preliminary results are presented with a discussion of limitations and future plans to organize a process that results in a model curriculum for Computer Information Systems.

2. HISTORY OF COMPUTING

Since the mid to late 1950's computers have evolved from museum curiosities into devices that have changed almost all aspects of life and commerce of the world forever. It has not been a single step, rather one of evolution of many technologies. Initial machines such as the IBM 650 were housed in a good-sized room and consumed many kilowatts per hour, many tons of air conditioning, had very small memories, required punched-card input and output and could service at most one person with a small problem. As technology replaced vacuum tubes with transistors; then with simple integrated circuits and eventually with very complex integrated circuits, much changed. Processing speed, memory size, speed of throughput, and machine instruction complexity increased by many orders of magnitude – Moore's Law. Storage media capacity, reliability and speed have had similar rates of growth. Methods of input, output, and inter device communication have advanced exponentially in both speed and diversity. All these technologic changes, yet costs decreased exponentially.

The computer is truly a dazzling piece of electro-mechanical capability. But, it will do nothing until it's told precisely – what to do. Early computers were expensive, few and far between; and rather limited serving only a single user or purpose at a time. By the end of the sixties mini-computers became less expensive, more accessible and increasingly capable. By the mid-seventies multiuser machines allowed "time-shared services" by means of terminals connected by modems to the computer.

In the early nineteen eighties, a major revolution occurred; in 1983 the IBM PC became Time Magazine's "Man of the Year". The expanding role of computing in commerce and government resulted in growing demand for computing professionals and academic programs to educate them. The proliferation of microprocessors ushered in personal computers and another burst of demand for computing education. As communications technologies (telephony, digital signaling, satellite and optical transmission) matured and expanded, connectively took the lead in computing's advancement (e.g. the Internet effectively brought all the points on the globe within reach).

With every advance in computing the need for software and systems developers has grown almost in the reverse relationship to the shrinkage of size, cost, and time to compute described in Moore's Law. But, Moore's Law has not held for productivity or cost/effectiveness of software and systems development practice. This fact motivates this project's concern for the CIS curricular perspective and it potential for addressing the reported shortfall in productivity and cost/effectiveness that appear widespread in the computing industry today.

3. HISTORY OF CURRICULUM

The Association for Computing Machinery (ACM), Data Processing Management Association (DPMA) – now the Association for Information Professionals (AITP) – and Institute of Electrical and Electronics Engineers (IEEE) have consistently supported the advancement of computing professional education. These organizations, along with newer organizations, including the Association for Information Systems (AIS) have sponsored a series of curriculum models that guide and shape the curricula that train and educate computing professionals. Prompted by the introduction and advancement of computers in the late 1950's, and with their availability in the 1960's, model curricula developed to guide programs and faculty. Each model curriculum specifies (to some degree) a focused perspective of professional competency including learning outcomes and the means (courses) for achieving them. Some curricular designs favored flexibility with alternative - but closely related – paths, while others were more prescriptive.

During the late 1960's, as computing and its applications diversified, it became apparent that at least two distinct flavors of computing had emerged. The ACM and IEEE first focused primarily on computer science, the first model curriculum being Computer Science 1968, reflecting its core scientific interests to answer questions related to "what can be computed?" Subsequently, a second group also emerged, focused on how computing could best be utilized for commercial or governmental purposes. The first working product of this "other" flavor of computing - IS model curricula (Ashenhurst, 1972; Couger, 1973, and Nunnamaker, 1982).

In the research on IS curricula that followed DPMA (1981; 1986) and IS'90 (Longenecker and Feinstein, 1991), as many as 126 names for IS programs were identified. These programs were housed in academic divisions, colleges and departments with at least 10 different designations according to Peterson's Guide and the DPMA mailing list. This diversity of labeling and situating IS education persists as a direct consequence of its inter-disciplinary nature.

Over the years, collegiate IS programs often adopted either the DPMA (now known as AITP) or ACM guidelines, or a mixture of both. Also, within the past 15 years, some programs have achieved ABET accreditation, which also has some influence on the curriculum adopted by that program. Regardless when surveying those programs, their faculty indicated much the same technical expectations for their graduates regardless of the academic division's label, or the guidelines they espoused. Aligning with a particular model's guidelines is better explained as a case of program marketing rather than an endorsement of a model curriculum's philosophy.

4. INITIAL COVERAGE OF CURRICULUM

Computing machinery vendors developed and sold systems and application software along with their machines while supplying organizational end-user training as well. Computer vendors also supplied computers so that interested faculty could learn to use the hardware, and perhaps promote the software to students taking classes. Early computing education supported discipline-specific computing applications in the sciences, mathematics or statistics. Programs focused specifically on computing theory evolved in the computer science programs in the mid-sixties along with the establishment of doctoral programs in computing. Computer-related education began to find a way into virtually every academic discipline as computing became an important research tool.

During this period of time, IS programs emphasized operating systems and system software as a platform for sophisticated application systems. (The reader who would like to review the detail of these skills migration is referred to the Appendix material of Longenecker, et al 2012.)

5. POST 1990 CURRICULUM

The post 1990 model curricula began to reflect diversification within the "spectrum" of the computing discipline such that an emergence of several computing disciplines had arisen. For instance, the CC2005 report refers to Information Technology as a new sub-discipline of computing quite distinct from Computer Engineering, Computer Science and Software Engineering programs. Curiously, it was also around this time that society had adopted the label "IT" for any of the endeavors of computing although "IT" is a clear misnomer as an umbrella term in light of description of IT in CC2005. IT programs were distinguished by a focus largely on infrastructure: "off the shelf" hardware and software installation and configuration. Whereas, IS's focus evolved toward creating and extending systems while closely aligning systems with business models and strategies to support the business's end-users, partners and clients including top-management. IS'95, IS97, and IS2002 model curricula all emphasize these core distinguishing aspects of IS as it "sits" among the other computing disciplines.

IT's focus appears from a current vantage point and also as described in CC2005 is planning, installing, configuring, testing and managing infrastructure: networks, operating systems, virtualization servers and server farms; and most recently supporting organizational information processing and security concerns. One could propose that IT should consider IS as a critically-important customer. Figure 1 suggests this relationship.

CC2005 provides a well-reasoned framework for mapping the computing landscape of computing professionals' knowledge, skills and responsibilities. Nearly a decade later, we should consider whether that landscape is evolving and whether the spectrum of computing disciplines should be refined, refocused and/or reconstituted. That is, just as IT emerged as a recognized and independent computing discipline, what other aspects within the computing problem space have changed such that other disciplines have evolved, or new disciplines have emerged? This paper argues that the Information Systems discipline has evolved.

<table>
<tr><td colspan="2" align="center">Organizational End-User</td></tr>
<tr><td colspan="2" align="center">CIO or CTO</td></tr>
<tr><td>Information Systems</td><td>IT Help Desk</td></tr>
<tr><td colspan="2" align="center">Information Technology Management</td></tr>
<tr><td colspan="2" align="center">Virtual and Physical Systems</td></tr>
<tr><td colspan="2" align="center">The Internet and Private Networks</td></tr>
</table>

Figure 1. Relationships of IS, IT, End-Users and the Help Desk to Physical Systems. The CTO/CIO has ultimate authority for hardware and networks through IT.

6. WHY CIS AS A DISCIPLINE IS WORTH THE EFFORT TO DEFINE

In seeking answers to the central question in this paper, has information systems evolved such that Computer Information Systems has arisen as a distinct variant of Information systems, we explore the following propositions:

(P1) *Computer Information Systems is a discernible sub-discipline of computing closely aligned but distinct from IS.*

Although not designating CIS as specifically a sub-discipline, CC2005 does describe a community of programs with a distinctive emphasis on information systems development and software construction. If our first proposition can be explored in a manner that is empirically testable and confirmable, it is reasonable to pursue a curriculum guideline to both describe and promote effective undergraduate education to prepare professionals to pursue CIS as a discipline.

One rationale for exploring the question of CIS as a discipline is the critical centrality of software and application systems in the super-discipline of computing. Although systems building was a core (perhaps the first) goal of IS undergraduate education in its earliest incarnations, the burgeoning catalog of business, organizational, and sociological topics that vie for attention in IS programs has gradually diminished or displaced system building as a core focus in many programs. Indeed the most recent curriculum guidelines for undergraduate IS education, IS2010, does not list the rudimentary knowledge and skills for programming and software development as required learning. This is a clear indication of the challenge in IS program design to allow room for the burgeoning topic catalog within a limited credit hour, four-year undergraduate degree (Waguespack 2012; Babb & Waguespack, 2014).

This leads to a second proposition:

(P2) The undergraduate computing programs that label themselves CIS consistently outline the set of professional knowledge and skills that defines the essential labor competencies to support the age of big data, mobile apps, and ubiquitous computing.

There is no question that innovation relies on availability of systems builders for the information systems that support their evolving products and services. More than ever, governments are turning to information system capabilities to address social and civic challenges in managing resources and public services. Taking nothing away from the value of IS education, there is a distinct and palpable need for undergraduate degree programs to serve the exploding demand for computing professionals who can create, build and rebuild the information processing engines that support the world's economies.

If the project described herein can empirically ground our second proposition, this outcome will support the effort to develop curriculum guidelines for undergraduate degree(s) in CIS. Experiences in our institutions show that programs that can attest to following published guidelines have a greater prospect for establishment, growth and sustainment in colleges and universities. The mantel of guideline compliance supports recruiting of students, faculty, and philanthropic support. And the collegiality that a community of programs and their faculty can develop advances pedagogy and research that advances the discipline.

7. SKILLS TO MEET NEEDS

At this point in our process, we have not attempted to complete a skills analysis, even though a curriculum must be specified by its skills. We suspect that the fundamental skills would be similar to Colvin (2008), Landry (2001) and Haigood (2001), these are the underpinnings of IS 2002. Skills are not a list of topics, rather they are a list of what the graduate of a program must be able to do as an effective practitioner in the discipline. All curricula must identify these skills based on discussions with employers. Once it becomes clear what skills would seem to satisfy the body of knowledge, courses can be proposed as a means of grouping the skills, and course outcomes can be prepared. These concepts are depicted in Figure 2.

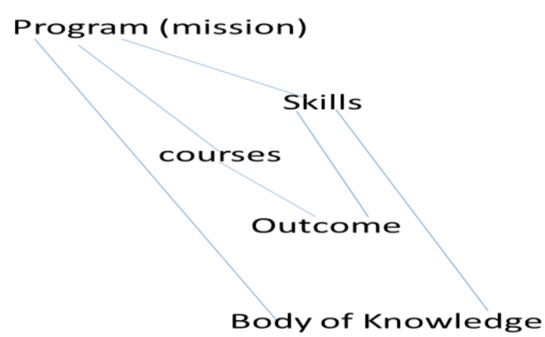


Figure 2 shows the relationships between the entities of the figure: Program has a body of knowledge; program also has courses specified by the course outcomes which outcomes are necessary to provide time on task to achieve the skills which satisfy the coverage of the body of knowledge.

8. THE EFFECT OF ACCREDITATION ON THE DISCIPLINE

From the beginning, Information Systems laid its "anchor" in the port called the "College of Business;" and for many good reasons. Among the facts of life of setting up your interdisciplinary "shop" in a College of Business is that Colleges of Business will typically attain AACSB (Association to Advance Collegiate Schools of Business) accreditation. If we consider that ABET provides a similar program-level accreditation for IS programs in the manner that AACSB provides college-level accreditation, it is important that we consider how these two "cultures" mix (Babb & Abdullat, 2014). While there are many cases where each accreditation co-exist, AACSB will always be the dominant culture and that dominance imposes two serious limitations to computing programs: 1) AACSB would prefer an 8 course limit on the number of courses in a program, and 2) AACSB would prefer no sequences of courses greater than 2 courses. It is clearly difficult to expose the number of topics relevant to IS with this time limitation, and worse yet it is most likely impossible to reach the applications level of knowledge (Bloom, 1956) necessary for practitioners. However, in a "have your cake and eat it too" sense, if Information Systems (and its variants) wish to persist in the college of business, some accommodation or work-around is needed. While ABET-accredited programs have that work-around "built in" by way of its own specific and reliable requirements, very few IS programs are ABET accredited, perhaps largely due to this AACSB-dominance of the college of business culture (Babb & Abdullat, 2014).

9. IS 2002 WORK-AROUND

One work-around is in the nature of the IS 2002 model curriculum, which can be said to offer a more generous approach: 1) a 10-course minimum course count was established; 2) courses were viewed as containers of knowledge specifications—while only a single course was suggested for programming, it was very clear that more class time would be required; and 3) a set of prerequisite recommendations was given either in the curriculum, in general studies, or even in high-school. All ABET- accredited schools have clearly followed this model whether in a business school or not (Feinstein, Longenecker, and Shrestha, 2013). However, the complaint from some quarters is that IS 2002 is too fully and inflexibly specified. However, rather than make additional opportunities available as IS 2010 suggests, the obvious solution is to increase the number of offered courses. We do not as of this writing have an immediate answer. Certainly a coupled master program is one alternative that could be explored.

10. GUIDANCE FROM CC2005

A significant problem of working on a "Computer Information Systems" degree model curriculum is that the underlying discipline does not really exist. The team working on this study made the decision to call the discipline "Information Systems" and then to provide a single model curriculum for our field. To be sure that this new model had an involved professional society, the DPMA believed this as appropriate and funded efforts to promote the model curriculum report at that time. Also, the ACM had published its ACM'72 document with the name Information Systems. The discipline name took hold and as evident with the following publications: IS'95, IS'97, IS2002 sponsored jointly by the AITP (formerly DPMA), the ACM, and the AIS. In pursuit of our research questions, we began with the approach for discipline definition as outlined by the CC2005 task force. First, CC2005 present a sketch of the spectrum and breadth of computing disciplines (albeit in broad strokes), and, specifically, the report is grounded by defining and/or referencing bodies of knowledge appropriate to each discipline. Thus, we take the approach that the body of knowledge of the discipline is its "kernel:" it's central taxonomy, epistemology, and perhaps ontology upon which "knowing" the discipline is founded.

11. HUNT FOR THE CIS BODY OF KNOWLEDGE

Parameters and Picture

With CC2005 as a guide we used a small panel of experts, combined with a very comprehensive survey instrument, to collect and evaluative input regarding a collection of Knowledge Areas (KA) from the various bodies of knowledge that could arguably define the "boundaries" of the CIS discipline. Moreover, these KAs were culled from the more current model curricula guidelines for the most salient disciplines that seemed to relate to our postulated "CIS" discipline. Thus, for each body of knowledge element to be studied, it was decided to collect each of the following parameters:

1. Cognitive depth of knowledge, an integer value with the meaning shown in appendix 2 which ranged from 0 meaning "no knowledge required" to 5 a very high level of professional. Most faculty who work with indicator would be very surprised to find many at level 4—application knowledge. As faculty we seem not to have been able to get our students to do even this well. A

specification below 2 most likely indicates that although considerable time will have been spent, most all of this knowledge will be lost within days of the last exam. Perhaps we should set our expectations higher on that which we really care about: Our students might not make computer jeopardy players, yet they may have become exceptional problem solvers. This parameter will become the z or height off the paper axis. Please note that in CC2005 the parameter is represented by a "dot".

2. Emphasis is a parameter with values between +50 and -50 which describes highly practical (+50) to highly theoretical (-50). This parameter will be the x or horizontal axis.

3. Organizational relevance is a parameter between 1 and 5. Possible values of the parameter are specified as shown in the graph below. Please note, the parameter may take on multiple values.

4. Rater confidence is a value between 0 and 99% and gives the rater's confidence in make the estimates for parameters 1, 2, and 3. The value is contained in tables, but is not plotted.

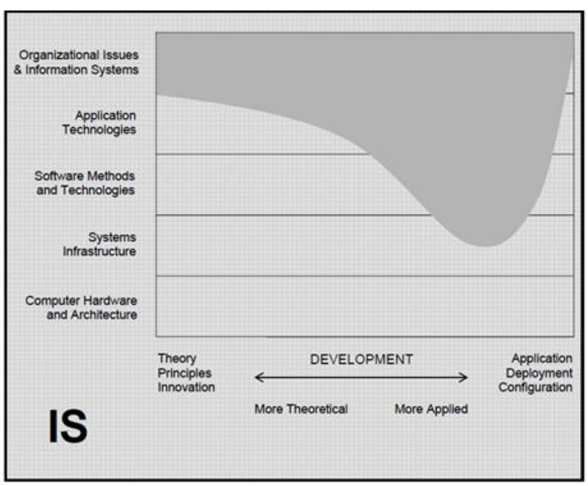


Figure 3. A "definition of IS" according to the three parameters herein used by CC2005. Each program should be different.

Our intention was to begin to empirically define, test, and validate the "problem space" of CIS in a manner that had been notionally compiled in by a respected team of experts in CC2005. Figure 3 (from CC2005) depicts the area that Information Systems "claims" within the problem space of computing and we generally agree that it appears to be correct. However, we feel that Computer Information Systems continues to reflect Figure 3 while recent model curricula (IS 2010) have begun to paint a different picture of Information Systems that seems to "retreat" more towards the "Organizational Issues and Information Systems" end of the y-axis in Figure 3.

Our initial work then was to solicit a team of invited experts to help test the viability of this approach as an "entry point" into exploring what CIS might look like and ultimately answer the questions posed in this paper: does CIS exist? What does it look like?

Body of Knowledge Candidates
All recent ACM model curricula contain a list of knowledge areas (KA) and sub-areas known as knowledge units (KU). These KA's were the basis for the graphics shown above. In other words, all KA's were considered for potential inclusion in the graphic. Therefore, our KA list was comprised of all KA's included in the ACM website showing all model curricula. Our list included KA's from IS'90, IS'95, IS'97, IS2002, and IS2010. Our list is current through CSC 2013. It also contains the graduate SE (Software Engineering) curriculum, but does not include the systems engineering material. It includes a 2014 minimum NICE specification which includes a minimal coverage of IA.

We sorted the list alphabetically, and did remove exact duplicates, but left material that might be different. We did not attempt to resolve hierarchically structured material that could be managed in a second or later pass.

Appointment of Expert Advisory Group
We solicited a group of experts (see Appendix 1) as a way to preliminarily validate a body of knowledge for CIS and provide expert feedback on this new candidate program. We chose individuals with a strong and consistent background of excellence in computing education. The characteristics of this 20-member expert team include:

- Significant ABET experience
- AACSB Schools
- EDSIG (AITP) Fellows
- Chairs / Deans
- National curriculum participants
- Professor / University Professor
- Editors/Associate Editors
- Publishers
- Conference Leaders

All members of the Expert Advisory Group were emailed and then called. All attempted the survey and 16 completed the entire survey instrument. The results of that effort are visible in the appendix material of this document.

12. ANALYSIS OF PRELIMINARY DATA

From an analysis of the depth of knowledge required (e.g. expectations are greater than 3.5), several groups of knowledge clearly are revealed as being the most important to this curriculum (see appendix 3, 4, and 5 material):

- Database
- Information Systems Development (Business Requirements)
- Systems Design
- Software Requirements, Programming (including web)
- Project Management based on Leadership, Team, and Interpersonal skills

The relevance of multiple sources of KA's can be seen by inspection in Appendix 5. Appendix 4 and 5 show the KA's assigned to each category as identified in Appendix 3. Finally, Appendix 6 provides a mechanism for comparison of IS curricula as well as a way to clearly see the differences between the existing IS curricula and the new CIS model

While networking, operating systems, and security are important, it is becoming clear that Information Technology groups will have to take the responsibility to fulfill such requirements.

13. CONCLUSION

The results of the survey indicate that the body of knowledge reported would be a worthwhile adventure. Although we all have experience in curriculum development, we never started without a clear picture of the prize, the definition of CIS. We all have been impressed with the scholarship of that document as well as its practicality.

We were very pleased with the active willingness of our Executive Advisory Team to respond strongly and with a very short lead time. The team responded with remarkable consistence. Characteristic of their leadership ability they were able to focus on key ideas that did not just fall off the turnip truck. Rather, they bring a new focus for our consideration.

Future research will be to adjust the data gathering instrument, to cull the list slightly, and get the material out to as wide a group as possible for response.

Also, we will be identifying some necessary projects and will be asking for help. Those announcements and requests for help will be built into our next data collection effort, and we sincerely hope you will answer the call.

13. ACKNOWLEDGEMENTS

Professor Heikki Topi of Bentley has become a mentor to us. We are deeply appreciative. He has provided warm yet most detailed critique enabling this team to perform much better and at a higher level.

14. REFERENCES

ACM Curriculum Committee on Computer Science 1968. Curriculum 68: Recommendations for the Undergraduate Program in Computer Science. *Communications of the ACM*, 11:3, March 1968, pp. 151-197.

Alter, S. (1998). Information systems. Addison-Wesley Longman Publishing Co., Inc.

Ashenhurst, R. L. (Ed.) (1972). A Report of the ACM Curriculum Committee on Computer Education for Management. *Association for Computing Machinery, Inc., 1972*.

Babb, J. S., & Waguespack, L. (2014). In Search of Design-Focus in IS Curricula. *Information Systems Education Journal*, 12(4), 29.

Babb, J. S., & Abdullat, A. (2014). Communicating the Value of Program-Level Accreditation for Information Systems in the College of Business. Information Systems Education Journal, 12(3), 39.

Bloom, Benjamin S. (Ed.) (1956). The Taxonomy of Educational Objectives: Classification of Educational Goals. Handbook 1: The Cognitive Domain. : McKay Press, New York 1956.

Checkland, P., & Holwell, S. (1997). Information, systems and information systems: making sense of the field.

Davis, G. B., & Olson, M. H. (1984). Management information systems: conceptual foundations, structure, and development. McGraw-Hill, Inc.

Colvin, R. (2008). Information Systems Skills and Career Success, Masters Thesis, University of South Alabama, School of Computer and Information Sciences.

Couger, J. D., Davis, G.B., Feinstein, D.L., Gorgone, J.T. and Longenecker, H.E. (1997). IS'.97: Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems, *Data Base,* Vol. 26 No. 1, pp. I-94.

Couger, J. D., Davis, G. B., Dologite, D. G., Feinstein, D. L., Gorgone, J. T., Jenkins, M., Kasper, G. M. Little, J. C., Longenecker, H. E. Jr., and Valachic, J. S. (1995). IS'95: Guideline for Undergraduate IS Curriculum, *MIS Quarterly* (19:3), 1995, pp. 341-360.

Davis, G., J. T. Gorgone, J. D. Couger, D. L. Feinstein, and H. E. Longenecker. (1997). IS'97: Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *ACM SIGMIS Database*, 28(1).

Davis, G.B., Couger, J. D., Feinstein, D.L., Gorgone, J.T. and Longenecker, H.E. "IS '97 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems," ACM, New York, NY and AITP (formerly DPMA), Park Ridge, IL, 1997.

DPMA 1981. *DPMA Model Curriculum, 1981*. Park Ridge, Illinois: Data Processing Management Association.

DPMA 1986. *DPMA Model Curriculum, 1986.* Park Ridge, Illinois: Data Processing Management Association, 1986.

Gorgone, J.T., Davis, G.B. Valacich, J., Topi, H., Feinstein, D.L. and Longenecker. H.E. (2003). IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *Data Base* 34(1).

Gorgone, John T., J. Daniel Couger, Gordon B. Davis, David L. Feinstein, George Kasper, and Herbert E. Longenecker 1994. "Information Systems '95," *DataBase,* Volume 25, Number 4, November 1994, pp. 5-8.

Haigood, B. (2001). Classification of Performance Level Requirements of Current Jobs Within the Field of Information Systems, Masters Thesis, University of South Alabama, School of Computer and Information Sciences.

Landry, J. P., Longenecker, H.E., Haigood, B. and Feinstein, D.L.. 2000. "Comparing Entry-Level Skill Depths Across Information Systems Job Types: Perceptions of IS Faculty," *Proceedings of Sixth Americas* IT 2008. IEEE/ACM Joint Task Force on Computing Curricula. Information Technology 2008, Curriculum Guidelines for Undergraduate Degree Programs in Information Technology, ACM and IEEE-Computer Society, November 2008. Retrieved at http://www.acm.org/education/education/curricula-recommendations
Conference on Information Systems, Long Beach, CA.

Longenecker, H.E., and Feinstein, D.L. (1991b.) "On Establishing Excellence in Information Systems," *Journal of Information Systems Education*, Volume 3, Number 1, spring 1991, pp. 26-31.

Longenecker, H.E., Feinstein, D. L. (Eds.) (1991c). *IS'90: The DPMA Model Curriculum for Information Systems for 4 Year Undergraduates.* Park Ridge, Illinois: Data Processing Management Association.

Longenecker, H.E., Feinstein, D.L., Couger, J.D., Davis, G.B. and Gorgone, J.T. (1995). "Information Systems '95: A Summary of the Collaborative IS Curriculum Specification of the Joint DPMA, ACM, AIS Task Force," *Journal of Information Systems Education,* Volume 6, Number 4, pp. 174-187.

Longenecker, H. E., Jr., D. L. Feinstein, J. D. Couger, G. B. Davis, and J. T. Gorgone (1995). Information Systems '95: A Summary of the Collaborative IS Curriculum Specification of the Joint DPMA, ACM, AIS Task Force. *Journal of Information Systems Education,* Volume 6, Number 4, pp. 174-187.

Longenecker, H. E. Jr, D. Henderson, E. Smith, P. Cupoli, D. M. Yarbrough, A. M. Smith, M. L. Gillenson, and D. L. Feinstein (2006). A Recommendation for a Professional Focus Area in Data Management for the IS2002 Information Systems Model Curriculum. In *The Proceedings of the Information Systems Education Conference 2006*, v 23 (Dallas): §2115. ISSN: 1542-7382.

Longenecker, H.E., Feinstein, D.L. and Babb, J.S. (2013). Is there a need for a CIS Model Curriculum?, Proceedings of ISECON, San Antonia 2013.

Nunamaker, J.F., Couger, J.D. and Davis, G.B. (1982). "Information Systems Curriculum Recommendations for the 80s: Undergraduate and Graduate Programs," *Communications of the ACM*, Volume 25, Number 11, November 1982, pp. 781-805.

Teichroew, D (1971). "Education related for the use of computers in organizations", CACM 14,9 (September 1971).

Orlikowski, W. J., & Iacono, C. S. (2001). Research commentary: Desperately seeking the "IT" in IT research—A call to theorizing the IT artifact. Information systems research, 12(2), 121-134.

Pawlak, Z. (1981). Information systems theoretical foundations. *Information systems*, 6(3), 205-218.

Shackelford, R., McGettrick, A., Sloan, R., Topi, H., Davies, G., Kamali, R., ... & Lunt, B. (2006). Computing curricula 2005: The overview report. ACM SIGCSE Bulletin, 38(1), 456-457.

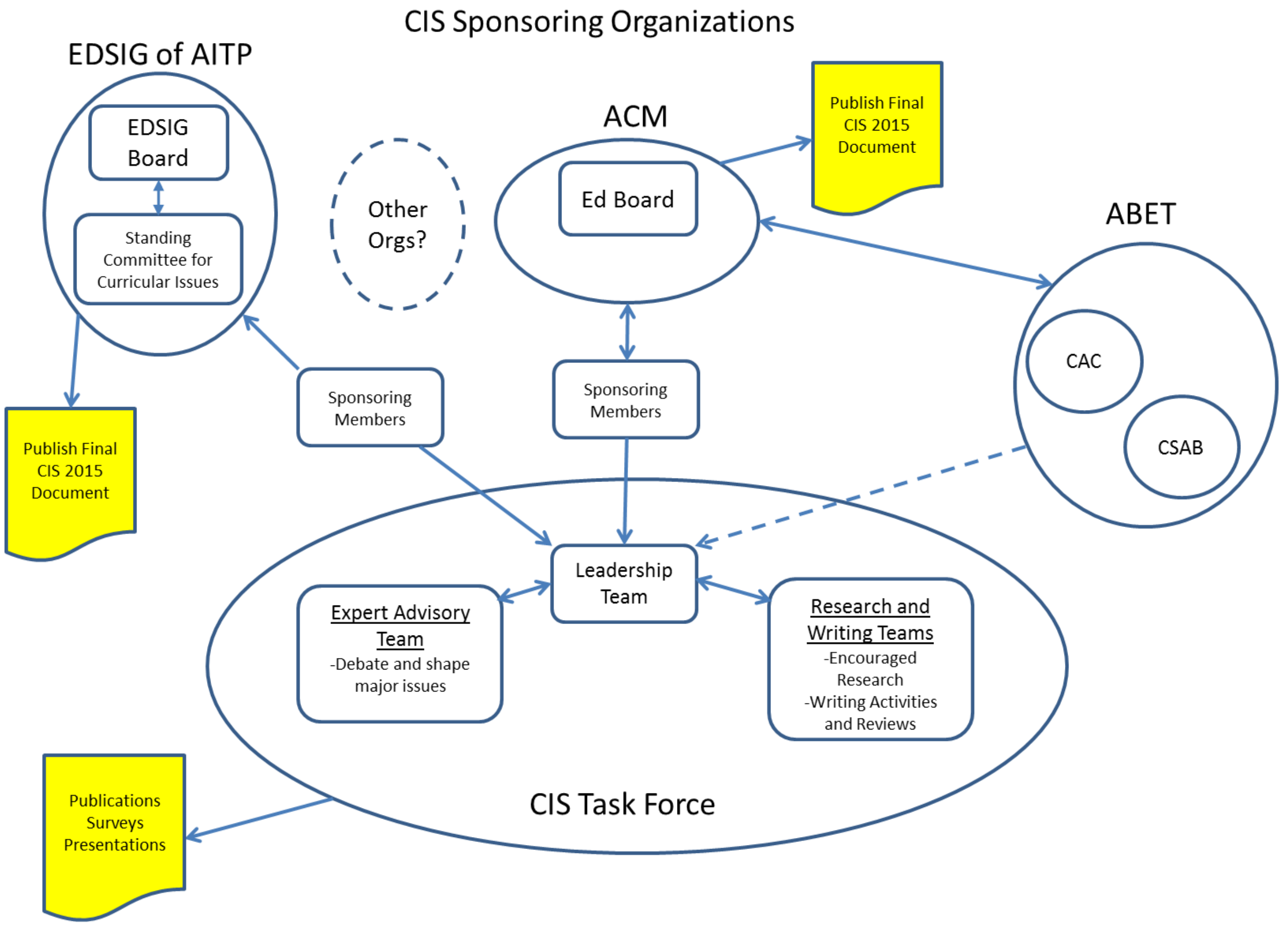
Teichroew, D (1971). "Education related for the use of computers in organizations", CACM 14,9 (September 1971).

Topi, H., Valacich, J., Wright, R.T., Kaiser, K.M., Nunamaker, J.F., Sipior, J.C., and Vreede, G.J. (2010). IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems, Association for Computing Machinery (ACM), Association for Information Systems (AIS)", retrieved July 14, 2012: http://www.acm.org/education/curricula/IS%202010%20ACM%20final.pdf

Waguespack, L. J. (2011). "Design, The "Straw" Missing From the "Bricks" of IS Curricula", ISEDJ 9(2) June 2011.

Appendix 1: Expert Advisory Team

Name	Affiliation
Bart Longenecker	University of South Alabama
Bruce Saulnier	Quinnipiac University
Bruce White	Quinnipiac University
Cameron Lawrence	University of Montana
Cheryl Aasheim	Georgia Southern University
Chuck Woratschek	Robert Morris University
David Feinstein	University of South Alabama
Gayle Yaverbaum	Penn State University
Harold Pardue	University of S. Alabama
Heikki Topi	Bentley University
Jeff Landry	University of South Alabama
Jeffry Babb	West Texas A&M University
Jerry Wagner	California State Polytechnic University
John Turchek	Robert Morris University
Jon Clark	Colorado State University
Karthikeyan Umapathy	University North Florida
Les Waguespack	Bentley University
Paul Leidig	Grand Valley State University
Ronald Kizior	Loyola University
Scott Hunsinger	Appalachian State University
Tom Janicki	University NC Wilmington
William Tastle	Ithaca College



Note: Operation of the CIS Task Force began informally as discussions over the period of a year with David Feinstein and Heikki Topi along with Bart Longenecker. Jeff Babb and Bart invited Les Waguespack to join the discussions because of the closeness of Les and Heikki. Then, Les worked with the AITP-EDSIG Board for formal recognition of the effort. EDSIG formed a Committee for Curricular Affairs appointing Les as Chair, along with Jeff Babb. Internally to the Task Force, Tom Janicki joined the group. The five members operate as Co-chairs of the CIS Force. The task force has in turn invited very well-known members of the community to form the "Expert Advisory Team" The task force plans to this advisory team as a sounding board to verify approaches. The first face-face meeting of the entire task force will occur at ISECON 2014. Members of ISECON will be invited to the meeting.

Note: The relationship with the ACM is proposed. Other groups may be asked to join.

Appendix 2: Introduction for Expert Advisory Team (Included within Survey Instrument)

We are writing to you as the Leadership Team for Curriculum; we are working with both AITP-EDSIG and the ACM. We are writing to you as a computing professional to become part of our Advisory Team. Your first job will be to help vet the attached survey to establish the body of knowledge for the closely related disciplines identified in CC2005.

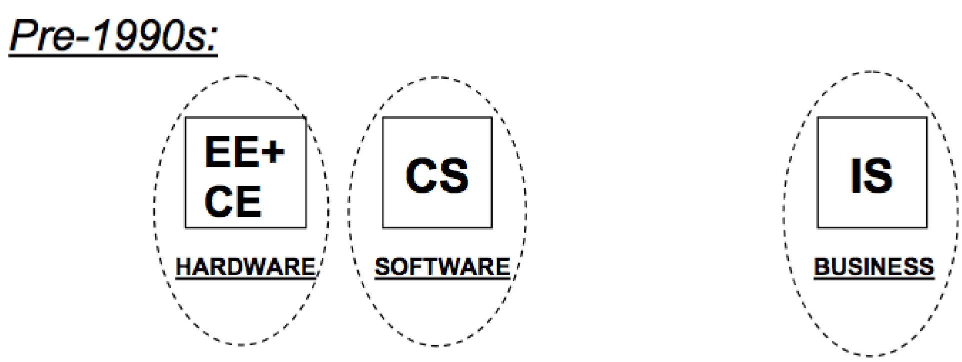
Over the past fifty years computing has evolved, as have computing curricula. Now, there is new hardware and software as well as many new opportunities and risks with systems 1000's of times more powerful and diverse. Most of us readily recognize curriculum in CS, IS, SE. IT, and CE which were identified in a 2005 document entitled: "Computing Curricula 2005--The Overview Report, covering undergraduate degree programs in Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering". There are new initiatives in IA (information assurance), as well as significant change expressed in the observables in the ACM model curricula. However, are there potentially more disciplines (or prominent sub-disciplines)?

We feel that our first major task is to confirm the need for a more technically focused Information Systems (IS) Undergraduate degree. CC2005 identifies CIS as a "more technical form" of IS. We support IS 2010, and view this work as an "extension" of that work. However, it is time to study formally changes that may have occurred to the Body of Knowledge, and to expectations of the computing industry who we would like to hire our graduates.

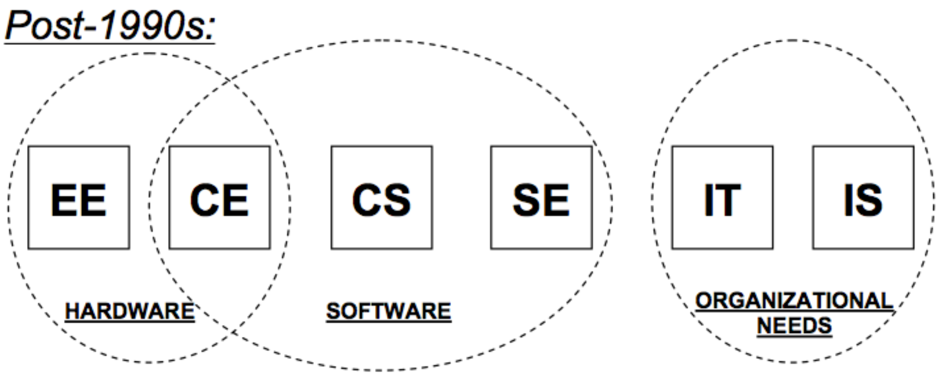
Following our joint effort we plan to release the work product to a wide group of computing academics. We would like to determine if the original five disciplines identified in CC2005 are relevant at the undergraduate level. We would like to know if there are discrete areas in information systems: general information systems, management information systems, and computer information systems. Is there a need for information assurance to be a separate degree program or be included within existing programs?

Therefore, if our focus will most likely be developing of a CIS program, we will be interested in establishing the characteristics of an undergraduate curriculum in computing that best prepares students to design, develop and implement secure information systems.

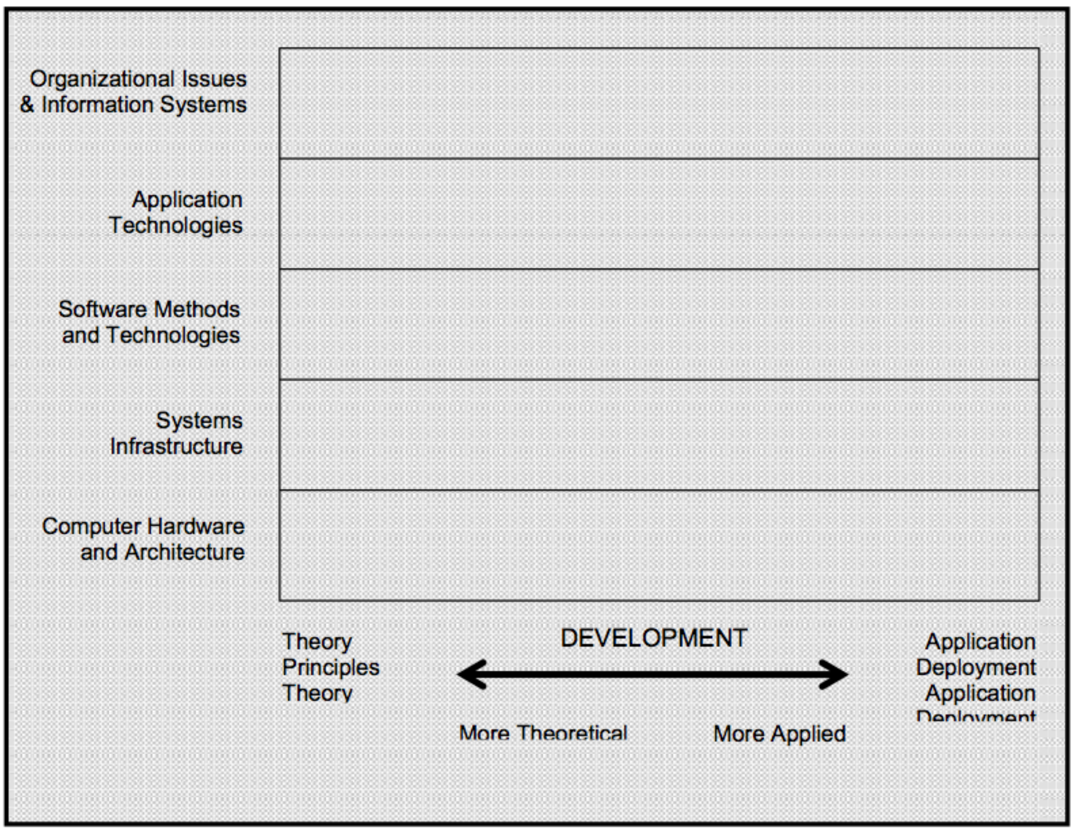
In 2005 a joint computing task force of computing societies (ACM, AIS, and IEEE-CS) portrayed the range of academic computing programs spanning computer hardware, software and organizational needs in the following diagram.



We wish to see how the computing education landscape may have changed since then. This survey duplicates the knowledge areas/skills published in curricular guidelines since 1997 into this survey to study how the clusters of knowledge areas may have evolved into the focus areas of computing education today.



The Joint Task Force for Computing Curricula 2005 developed a conceptual model to illustrate the commonalities and differences among computing disciplines. The model was designed to consider how each computing discipline occupies the problem space of computing. The model was design to reflect the disciplines as they existed at the time (2004/2005). The model was also intended to focus on what students in each of the disciplines typically do after graduation, not on all of the topics a student might study within a curriculum. The model follows:



In the Joint Task Force for Computing Curricula 2005 "Problem Space of Computing" Model, the horizontal range runs from Theory, Principles, and Innovation on the left, to Application, Deployment, and Configuration on the right. The vertical range runs from Computer Hardware and Architecture at the bottom, to Organizational Issues and Information Systems at the top.

The model was designed such that both the horizontal and vertical dimensions are considered together. The structure of this survey instrument is designed to provide a contemporary "picture" of what this problem space looks like today in the estimation and opinion of experts, educators, and practitioners. Whereas the Joint Task Force for Computing Curricula 2005 used this model to create informal illustrations used to communicate the task force's subjective interpretation of the various disciplines, this survey is an initial foray to examine this model from a more quantitative foundation. This information will be used for preliminary purposes to establish a baseline picture in the problem space of computing in order to inform subsequent steps to more clearly define what the computing discipline spectrum may look like today.

We have prepared a list of survey topics derived directly from the body of knowledge areas predominantly from the ACM curricula which we believe to be the Body of Knowledge area for computing. (Please see attachment "Knowledge Areas".)

As a member of our Advisory Team, we are asking you to answer this survey from the perspective of your discipline. We will ask you to identify yourself from a provided list if disciplines. If you are from another discipline, then enter your discipline and describe it. After you have answered the survey we will be looking to see your analysis as an academic professional. How can we improve this work product?

Then (considering the next 3 – 5 years), for each knowledge survey element presented please inform us at what depth of knowledge instruction should be specified with respect to requirements of your discipline. This is a very important first step, and will enable curriculum designers to write behavioral objectives. These objectives will enable a model curriculum and detailed course planning. Please remember that everything cannot be achieved in an undergraduate curriculum (10-12 courses of 36/semester hours -> 360 hours total). The higher the specification level, the more time will be required to learn the material. Most previous committees have found that the highest level specification is very difficult for undergraduates to achieve. The following table gives further insight into the depth of knowledge levels:

Depth of Knowledge	**Meaning of Depth**
0 – No Knowledge Required	No objectives will be planned for this item
1- Awareness	Learners have the ability to recognize this element, but can do nothing with the information. This process is automatic. It can be enhanced through repetition.
2 – Literacy / Strong Knowledge	Learners can differentiate among elements (red pen, black pen, felt tip, quill) and with guided practice can answer simple questions about the elements. Still nothing useful can be done with this level of knowledge.
3 – Concept / Use Skill	It is time for learners to be prompted to "DO" something with knowledge. As learners have increased cognitive burden with multiple elements, it can not be ignored that this is a prompted process of items barely learned at level 2.
4 – Application Ability	Learners at this level will have spent 3-10 times the effort associated with levels 1-3. Initial stages of application ability will enable solution of selection of simple elements to create a desired solution. However, considerable repetition will be necessary to marginally secure this ability. Without the repetition the ability will be rapidly lost. If the learning process involves a level of excitement, the learning will be enhanced.
5 - Advanced	This level consists of analysis, synthesis and evaluation based on Bloom's specifications. Solving a problem is an "ability", not this level of development of new knowledge.

Appendix 3: Data from Expert Advisory Team (Sorted by Body of Knowledge KA's Having a Level of 3.5 or Greater are Highlighting with Color Showing Groups)

KA	Knowledge Areas Sorted by Expert Expected Depth of knowledge	Depth 0-5	Emphasis theory -50 practice +50	Expert Confidence
A22	Database	4.3	28	97%
B03	Data Retrieval and / Manipulation with Database Languages	4.0	26	93%
A23	Analysis and Specification / of System Requirements	4.0	19	90%
A19	Analysis of Business / Requirements	3.9	20	94%
A21	Information Systems Design /	3.9	21	93%
C01	Programming / Fundamentals	3.8	32	94%
A07	Web Systems and / Technologies	3.8	32	92%
A15	Approaches to Systems / Development	3.7	24	96%
G08	Project Plan, Scope, and / Initiation	3.7	21	90%
D14	Systems Analysis & / Design	3.6	23	92%
B05	Data and Information / Modeling at Conceptual and logical Levels	3.5	17	90%
C09	Software / Requirements	3.5	18	93%
A24	Team and Interpersonal / Skills	3.5	20	94%
G07	Leading Project / Teams	3.5	21	91%
B01	Database Systems and / Distributed Databases	3.4	18	86%
C15	Software / Design	3.4	20	90%
A06	Information Technology / Fundamentals	3.4	23	91%
D09	Systems Development / Concepts and Methodologies	3.3	15	91%
C05	Human Computer / Interaction	3.3	17	92%
C16	Software Development / Fundamentals	3.3	25	92%
B12	Data Integrity and / Quality	3.2	19	91%
A20	Information and Business / Analysis	3.2	15	90%
D11	Systems Implementation and / Testing Strategies	3.2	16	89%
C06	Module Design and / Construction	3.2	18	91%
C19	Software / Testing	3.2	21	91%
G10	Project Execution & / Control	3.2	20	87%
A01	Impact of Information / Systems on Organizational Structure and / Processes	3.1	15	91%
D06	System Deployment and / Implementation	3.1	17	89%
B07	Physical Database / Implementation / Data Definition Language	3.0	16	92%
A13	Business Intelligence and / Decision Support	3.0	14	90%
M03	Basic Scripting/ / Programming	3.0	25	92%
B04	Teams and Interpersonal / Skills	2.9	15	89%
M01	Basic Data / Analysis	2.9	21	86%
G12	Project / Quality	2.9	8	82%
C13	Security and Privacy, / Vulnerabilities, Risks, Mitigation	2.9	10	78%
B08	Stored Procedure / Implementation	2.8	20	85%
B10	Data and Database / Administration	2.8	18	88%
A03	Identification of / Opportunities for IT enabled Organizational / Change	2.8	10	92%
A16	Different Approaches to / Implementing Information Systems	2.8	15	91%
C02	Programming / Languages	2.8	14	92%
C17	Software / Construction	2.8	30	89%
G06	IS Project Strategy and / Management	2.8	8	85%
G03	Establishing Project / Communication	2.8	13	88%
G09	Work Break-down / Structure	2.8	18	90%
G13	Project / Closure	2.8	13	84%
E04	Networks and / Communications	2.8	15	81%
H02	Probability and / Statistics--Basic probability theory, random variables and / probability distributions, estimation theo...	2.8	8	82%

B11-	Data Management and / Transaction Processing	2.7	22	90%
A17	Business Process Design and / Management	2.7	15	89%
A26	Computer / Networks	2.7	21	89%
H01-	Math and Statistics for / IT	2.7	18	87%
D15	User / Experience	2.6	16	89%
G11	Project / Standards	2.6	11	83%
A02	Individual and / Organizational Knowledge Work Capabilities	2.6	14	89%
C04-	Integrative Programming and / Technologies	2.6	18	78%
B13-	Security attacks and / mitigations	2.6	15	84%
B06-	Scripting	2.5	25	91%
A09	Enterprise / Architecture	2.5	4	79%
D07	System Verification and / Validation	2.5	12	86%
A25	Configuration and Change / Management	2.5	12	85%
C03-	Programming / Environments	2.5	16	87%
C07-	Software Engineering / Process	2.5	15	89%
C18-	Software / Quality	2.5	11	88%
C20-	Software / Maintenance	2.5	17	89%
D05	System Integration and / Architecture	2.5	13	83%
F02-	Information Assurance and / Security	2.4	3	78%
C11-	Algorithms and Data / Structures	2.3	12	86%
G01	Professional Issues in / Information Systems	2.3	6	85%
E07-	Organizational and / Management Concepts	2.3	-6	83%
M06	Cyber Defense, threats, / attacks, Incidents, incident management	2.3	7	76%
M02	Databases: Database / operations, injection attacks	2.2	19	73%
B02-	Basic File Processing / Concepts	2.2	12	88%
F03-	Managing the Information / Systems Function	2.2	-3	70%
F06-	Information Systems / Sourcing and Acquisition	2.2	9	72%
F05-	Information Systems / Planning	2.2	8	83%
F07-	Information Systems / Strategy	2.2	4	76%
G05	Managing the Process of / Change	2.2	-1	76%
D01	Theory and Development of / Systems	2.2	3	86%
D13	Systems Analysis & / Design Philosophies and Approaches	2.2	-1	88%
G04	IT Risk / Management	2.2	4	79%
E08-	Organizational / Behavior	2.2	-10	80%
F01-	Legal and Ethical Aspects / of IS	2.2	1	79%
M10	Policy, Legal, Ethics, and / Compliance	2.2	-3	80%
M11	System Administration: / installation, authentication, access, backups, virtualizations, / updates/patches, logging audit...	2.2	18	75%
A08	Using IT Governance / Frameworks	2.2	2	79%
M7-	IT Systems Components: / workstations, servers, storage, peripherals	2.2	14	84%
A27	Acquiring Information / Technology Resources and Capabilities	2.2	11	79%
M05	Fundamental Security Design / Principles	2.2	7	81%
D04	System Operation, Administration and / Maintenance	2.2	8	79%
G02	IS Leadership and / Empowerment	2.1	6	76%
B09-	Reporting Services, / ETL	2.0	15	77%

M04	IA Fundamentals: / Vulnerabilities, Attacks, Mitigation	2.0	6	80%
E05-	Operating / Systems	2.0	10	74%
D02	Strategic / Alignment	1.9	0	81%
D08	System Verification and / Validation Enabling	1.9	10	81%
E02-	High level System Design / Issues	1.9	-4	74%
E12-	Policies and / Compliance	1.9	-3	69%
C14	Social Issues and / Professional Practice	1.9	1	89%
A10	Architecture and / Organization	1.9	-2	79%
M08	Network Concepts, / Technology and Protocols, Vulnerabilities, Defense: firewalls, / vpn, dmz, monitoring, tools	1.9	11	76%
C08	Software Engineering / Management	1.8	2	82%
A05	General Organization Theory /	1.8	-10	81%
A12	Decision Theory /	1.8	-7	88%
F09-	Financing and Evaluating / the Performance of Information Technology / Investments	1.8	5	69%
A11	Computer Architecture and / Organization	1.8	9	83%
M09	Operating Systems Concepts, / security issues	1.8	8	81%
E09-	Parallel and Distributed / Computing	1.8	9	80%
C10	Algorithms and Complexity /	1.7	-7	90%
A04	General Systems Theory and / Quality	1.7	-5	87%
D03	Improving Alignment / Maturity	1.2	-4	71%

Appendix 4: Data form Expert Advisory Team (Grouped from the Highest Level Body of Knowledge Categories of Appendix 3)

KA		Depth 0-5	Emphasis Avg	Expert Confidence	
A22-	Database	4.3	28	97%	database
B03-I	Data Retrieval and / Manipulation with Database Languages	4.0	26	93%	
B05-I	Data and Information / Modeling at Conceptual and logical Levels	3.5	17	90%	
B06-I	Scripting	2.5	25	91%	
B01-I	Database Systems and / Distributed Databases	3.4	18	86%	
B07-I	Physical Database / Implementation / Data Definition Language	3.0	16	92%	
B08-I	Stored Procedure / Implementation	2.8	20	85%	
M02-	Databases: Database / operations, injection attacks	2.2	19	73%	
B12-I	Data Integrity and / Quality	3.2	19	91%	
B09-I	Reporting Services, / ETL	2.0	15	77%	
A13-	Business Intelligence and / Decision Support	3.0	14	90%	
B10-I	Data and Database / Administration	2.8	18	88%	
B11-I	Data Management and / Transaction Processing	2.7	22	90%	
B02-I	Basic File Processing / Concepts	2.2	12	88%	
A03-	Identification of / Opportunities for IT enabled Organizational / Change	2.8	10	92%	system development
A16-	Different Approaches to / Implementing Information Systems	2.8	15	91%	
F03-I	Managing the Information / Systems Function	2.2	-3	70%	
F06-I	Information Systems / Sourcing and Acquisition	2.2	9	72%	
A15-	Approaches to Systems / Development	3.7	24	96%	
F05-I	Information Systems / Planning	2.2	8	83%	
G02-	IS Leadership and / Empowerment	2.1	6	76%	
A01-	Impact of Information / Systems on Organizational Structure and / Processes	3.1	15	91%	
D02-	Strategic / Alignment	1.9	0	81%	
D03-	Improving Alignment / Maturity	1.2	-4	71%	
D08-	System Verification and / Validation Enabling	1.9	10	81%	
E02-I	High level System Design / Issues	1.9	-4	74%	
F07-I	Information Systems / Strategy	2.2	4	76%	
G05-	Managing the Process of / Change	2.2	-1	76%	
D01-	Theory and Development of / Systems	2.2	3	86%	
D13-	Systems Analysis & / Design Philosophies and Approaches	2.2	-1	88%	
A23-	Analysis and Specification / of System Requirements	4.0	19	90%	
A19-	Analysis of Business / Requirements	3.9	20	94%	
A20-	Information and Business / Analysis	3.2	15	90%	
D09-	Systems Development / Concepts and Methodologies	3.3	15	91%	
D06-	System Deployment and / Implementation	3.1	17	89%	

A21-	Information Systems Design /	3.9	21	93%	system design
D14-	Systems Analysis & / Design	3.6	23	92%	
B04-I	Teams and Interpersonal / Skills	2.9	15	89%	
D15-	User / Experience	2.6	16	89%	
A09-	Enterprise / Architecture	2.5	4	79%	
A17-	Business Process Design and / Management	2.7	15	89%	
D07-	System Verification and / Validation	2.5	12	86%	
D11-	Systems Implementation and / Testing Strategies	3.2	16	89%	
A25-	Configuration and Change / Management	2.5	12	85%	
E12-I	Policies and / Compliance	1.9	-3	69%	
C01-I	Programming / Fundamentals	3.8	32	94%	progarmming
C02-I	Programming / Languages	2.8	14	92%	
C03-I	Programming / Environments	2.5	16	87%	
M01-	Basic Data / Analysis	2.9	21	86%	
M03-	Basic Scripting/ / Programming	3.0	25	92%	
A07-	Web Systems and / Technologies	3.8	32	92%	
C05-I	Human Computer / Interaction	3.3	17	92%	
C06-I	Module Design and / Construction	3.2	18	91%	
C11-I	Algorithms and Data / Structures	2.3	12	86%	
C10-I	Algorithms and Complexity /	1.7	-7	90%	
C07-I	Software Engineering / Process	2.5	15	89%	
C16-I	Software Development / Fundamentals	3.3	25	92%	
C08-I	Software Engineering / Management	1.8	2	82%	
C09-I	Software / Requirements	3.5	18	93%	
C15-I	Software / Design	3.4	20	90%	
C17-I	Software / Construction	2.8	30	89%	
C18-I	Software / Quality	2.5	11	88%	
C19-I	Software / Testing	3.2	21	91%	
C20-I	Software / Maintenance	2.5	17	89%	
G06-	IS Project Strategy and / Management	2.8	8	85%	project management
A24-	Team and Interpersonal / Skills	3.5	20	94%	
G07-	Leading Project / Teams	3.5	21	91%	
G08-	Project Plan, Scope, and / Initiation	3.7	21	90%	
G03-	Establishing Project / Communication	2.8	13	88%	
G09-	Work Break-down / Structure	2.8	18	90%	
G10-	Project Execution & / Control	3.2	20	87%	
G11-	Project / Standards	2.6	11	83%	
G12-	Project / Quality	2.9	8	82%	
G04-	IT Risk / Management	2.2	4	79%	
G13-	Project / Closure	2.8	13	84%	

A04-	General Systems Theory and / Quality	1.7	-5	87%	Management of CIS
A05-	General Organization Theory /	1.8	-10	81%	
A12-	Decision Theory /	1.8	-7	88%	
E08-I	Organizational / Behavior	2.2	-10	80%	
F01-I	Legal and Ethical Aspects / of IS	2.2	1	79%	
C14-I	Social Issues and / Professional Practice	1.9	1	89%	
A02-	Individual and / Organizational Knowledge Work Capabilities	2.6	14	89%	
F09-I	Financing and Evaluating / the Performance of Information Technology / Investments	1.8	5	69%	
G01-	Professional Issues in / Information Systems	2.3	6	85%	
M10-	Policy, Legal, Ethics, and / Compliance	2.2	-3	80%	
M11-	System Administration: / installation, authentication, access, backups, virtualizations, / updates/patches, logging audit...	2.2	18	75%	
A08-	Using IT Governance / Frameworks	2.2	2	79%	
E07-I	Organizational and / Management Concepts	2.3	-6	83%	
M7-I	IT Systems Components: / workstations, servers, storage, peripherals	2.2	14	84%	
A27-	Acquiring Information / Technology Resources and Capabilities	2.2	11	79%	
D05-	System Integration and / Architecture	2.5	13	83%	
C04-I	Integrative Programming and / Technologies	2.6	18	78%	
M04-	IA Fundamentals: / Vulnerabilities, Attacks, Mitigation	2.0	6	80%	
M06-	Cyber Defense, threats, / attacks, Incidents, incident management	2.3	7	76%	
F02-I	Information Assurance and / Security	2.4	3	78%	
B13-I	Security attacks and / mitigations	2.6	15	84%	
C13-I	Security and Privacy, / Vulnerabilities, Risks, Mitigation	2.9	10	78%	
M05-	Fundamental Security Design / Principles	2.2	7	81%	
A06-	Information Technology / Fundamentals	3.4	23	91%	IT Components
A11-	Computer Architecture and / Organization	1.8	9	83%	
A10-	Architecture and / Organization	1.9	-2	79%	
D04-	System Operation, Administration and / Maintenance	2.2	8	79%	
M08-	Network Concepts, / Technology and Protocols, Vulnerabilities, Defense: firewalls, / vpn, dmz, monitoring, tools	1.9	11	76%	
E04-I	Networks and / Communications	2.8	15	81%	
A26-	Computer / Networks	2.7	21	89%	
E05-I	Operating / Systems	2.0	10	74%	
M09-	Operating Systems Concepts, / security issues	1.8	8	81%	
E09-I	Parallel and Distributed / Computing	1.8	9	80%	
H01-	Math and Statistics for / IT	2.7	18	87%	Other Courses
H02-	Probability and / Statistics--Basic probability theory, random variables and / probability distributions, estimation theo...	2.8	8	82%	

Appendix 5: Knowledge Areas ordered by Classification, by Source (curriculum model)

Source	Classification	Survey	KA Description	Depth	Emphasis	Conf
IS2002	Database	A22	Database	4.3	28	97%
IS2010	Database	B02	Basic File Processing / Concepts	2.2	12	88%
IS2010	Database	A13	Business Intelligence and / Decision Support	3.0	14	90%
IS2010	Database	B10	Data and Database / Administration	2.8	18	88%
IS2010	Database	B05	Data and Information / Modeling at Conceptual and logical Levels	3.5	17	90%
IS2010	Database	B12	Data Integrity and / Quality	3.2	19	91%
IS2010	Database	B11	Data Management and / Transaction Processing	2.7	22	90%
IS2010	Database	B03	Data Retrieval and / Manipulation with Database Languages	4.0	26	93%
IS2010	Database	B01	Database Systems and / Distributed Databases	3.4	18	86%
IS2010	Database	B07	Physical Database / Implementation / Data Definition Language	3.0	16	92%
new	Database	B09	Reporting Services, ETL	2.0	15	77%
new	Database	B08	Stored Procedure Implementation	2.8	20	85%
NSA2014	Database	M02	Databases: Database operations, injection attacks	2.2	19	73%
NSA2014	Database	B06	Scripting	2.5	25	91%

IS2002	Sys Devel	A15	Approaches to Systems / Development	3.7	24	96%
IS2002	Sys Devel	A20	Information and Business / Analysis	3.2	15	90%
IS2002	Sys Devel	G02	IS Leadership and / Empowerment	2.1	6	76%
IS2002	Sys Devel	G05	Managing the Process of / Change	2.2	-1	76%
IS2002	Sys Devel	D09	Systems Development / Concepts and Methodologies	3.3	15	91%
IS2010	Sys Devel	A23	Analysis and Specification / of System Requirements	4.0	19	90%
IS2010	Sys Devel	A19	Analysis of Business / Requirements	3.9	20	94%
IS2010	Sys Devel	A16	Different Approaches to / Implementing Information Systems	2.8	15	91%
IS2010	Sys Devel	E02	High level System Design / Issues	1.9	-4	74%
IS2010	Sys Devel	A03	Identification of / Opportunities for IT enabled Organizational / Change	2.8	10	92%
IS2010	Sys Devel	A01	Impact of Information / Systems on Organizational Structure and / Processes	3.1	15	91%
IS2010	Sys Devel	D03	Improving Alignment / Maturity	1.2	-4	71%
IS2010	Sys Devel	F05	Information Systems / Planning	2.2	8	83%
IS2010	Sys Devel	F06	Information Systems / Sourcing and Acquisition	2.2	9	72%
IS2010	Sys Devel	F07	Information Systems / Strategy	2.2	4	76%
IS2010	Sys Devel	F03	Managing the Information / Systems Function	2.2	-3	70%
IS2010	Sys Devel	D02	Strategic / Alignment	1.9	0	81%
IS2010	Sys Devel	D06	System Deployment and / Implementation	3.1	17	89%
IS2010	Sys Devel	D08	System Verification and / Validation Enabling	1.9	10	81%
IS2010	Sys Devel	D13	Systems Analysis & / Design Philosophies and Approaches	2.2	-1	88%
IS2010	Sys Devel	D01	Theory and Development of / Systems	2.2	3	86%

IS2002	Sys Design	A21	Information Systems Design /	3.9	21	93%
IS2002	Sys Design	D11	Systems Implementation and / Testing Strategies	3.2	16	89%
IS2002	Sys Design	B04	Teams and Interpersonal / Skills	2.9	15	89%
IS2010	Sys Design	A17	Business Process Design and / Management	2.7	15	89%
IS2010	Sys Design	A25	Configuration and Change / Management	2.5	12	85%
IS2010	Sys Design	A09	Enterprise / Architecture	2.5	4	79%
IS2010	Sys Design	E12	Policies and / Compliance	1.9	-3	69%
IS2010	Sys Design	D07	System Verification and / Validation	2.5	12	86%
IS2010	Sys Design	D14	Systems Analysis & / Design	3.6	23	92%
IS2010	Sys Design	D15	User / Experience	2.6	16	89%

CS2013	Programming	C10	Algorithms and Complexity /	1.7	-7	90%
CS2013	Programming	C02	Programming / Languages	2.8	14	92%
CS2013	Programming	C16	Software Development / Fundamentals	3.3	25	92%
IS2002	Programming	C11	Algorithms and Data / Structures	2.3	12	86%
IT2008	Programming	C05	Human Computer / Interaction	3.3	17	92%
IT2008	Programming	A07	Web Systems and / Technologies	3.8	32	92%
NSA2014	Programming	M01	Basic Data / Analysis	2.9	21	86%
NSA2014	Programming	M03	Basic Scripting/ / Programming	3.0	25	92%
NSA2014	Programming	C03	Programming / Environments	2.5	16	87%
SwE2009	Programming	C06	Module Design and / Construction	3.2	18	91%
SwE2009	Programming	C01	Programming / Fundamentals	3.8	32	94%
SwE2009	Programming	C17	Software / Construction	2.8	30	89%
SwE2009	Programming	C15	Software / Design	3.4	20	90%
SwE2009	Programming	C20	Software / Maintenance	2.5	17	89%
SwE2009	Programming	C18	Software / Quality	2.5	11	88%
SwE2009	Programming	C09	Software / Requirements	3.5	18	93%
SwE2009	Programming	C19	Software / Testing	3.2	21	91%
SwE2009	Programming	C08	Software Engineering / Management	1.8	2	82%
SwE2009	Programming	C07	Software Engineering / Process	2.5	15	89%

IS2002	Proj Mgmt	G06	IS Project Strategy and / Management	2.8	8	85%
IS2002	Proj Mgmt	A24	Team and Interpersonal / Skills	3.5	20	94%
IS2002	Proj Mgmt	G09	Work Break-down / Structure	2.8	18	90%
IS2010	Proj Mgmt	G03	Establishing Project / Communication	2.8	13	88%
IS2010	Proj Mgmt	G04	IT Risk / Management	2.2	4	79%
IS2010	Proj Mgmt	G07	Leading Project / Teams	3.5	21	91%
IS2010	Proj Mgmt	G13	Project / Closure	2.8	13	84%
IS2010	Proj Mgmt	G12	Project / Quality	2.9	8	82%
IS2010	Proj Mgmt	G11	Project / Standards	2.6	11	83%
IS2010	Proj Mgmt	G10	Project Execution & / Control	3.2	20	87%
IS2010	Proj Mgmt	G08	Project Plan, Scope, and / Initiation	3.7	21	90%

CE2004	IT Components	A26	Computer Networks	2.7	21	89%
CE2004	IT Components	A11	Computer Architecture and / Organization	1.8	9	83%
CS2013	IT Components	A10	Architecture and / Organization	1.9	-2	79%
CS2013	IT Components	E09	Parallel and Distributed Computing	1.8	9	80%
IS2002	IT Components	D04	System Operation, Administration and / Maintenance	2.2	8	79%
IT2008	IT Components	A06	Information Technology / Fundamentals	3.4	23	91%
NSA2014	IT Components	M08	Network Concepts, Technology and Protocols, Vulnerabilities, Defense: firewalls, vpn, dmz, monitoring, tools	1.9	11	76%
NSA2014	IT Components	M09	Operating Systems Concepts, security issues	1.8	8	81%
SwE2009	IT Components	E04	Networks and Communications	2.8	15	81%
SwE2009	IT Components	E05	Operating / Systems	2.0	10	74%

IS2002	Mgmt of CIS	A12	Decision Theory /	1.8	-7	88%
IS2002	Mgmt of CIS	A05	General Organization Theory /	1.8	-10	81%
IS2002	Mgmt of CIS	A04	General Systems Theory and Quality	1.7	-5	87%
IS2002	Mgmt of CIS	F01	Legal and Ethical Aspects / of IS	2.2	1	79%
IS2002	Mgmt of CIS	E08	Organizational / Behavior	2.2	-10	80%
IS2010	Mgmt of CIS	A27	Acquiring Information / Technology Resources and Capabilities	2.2	11	79%
IS2010	Mgmt of CIS	F09	Financing and Evaluating / the Performance of Information Technology / Investments	1.8	5	69%
IS2010	Mgmt of CIS	G01	Professional Issues in Information Systems	2.3	6	85%
IS2010	Mgmt of CIS	C13	Security and Privacy, Vulnerabilities, Risks, Mitigation	2.9	10	78%
IS2010	Mgmt of CIS	C14	Social Issues and / Professional Practice	1.9	1	89%
IS2010	Mgmt of CIS	D05	System Integration and / Architecture	2.5	13	83%
IS2010	Mgmt of CIS	A08	Using IT Governance / Frameworks	2.2	2	79%
IT2008	Mgmt of CIS	F02	Information Assurance and / Security	2.4	3	78%
IT2008	Mgmt of CIS	C04	Integrative Programming and / Technologies	2.6	18	78%
new	Mgmt of CIS	A02	Individual and Organizational Knowledge Work Capabilities	2.6	14	89%
new	Mgmt of CIS	E07	Organizational and Management Concepts	2.3	-6	83%
NSA2014	Mgmt of CIS	M06	Cyber Defense, threats, / attacks, Incidents, incident management	2.3	7	76%
NSA2014	Mgmt of CIS	M05	Fundamental Security Design Principles	2.2	7	81%
NSA2014	Mgmt of CIS	M04	IA Fundamentals: Vulnerabilities, Attacks, Mitigation	2.0	6	80%
NSA2014	Mgmt of CIS	M07	IT Systems Components: / workstations, servers, storage, peripherals	2.2	14	84%
NSA2014	Mgmt of CIS	M10	Policy, Legal, Ethics, and Compliance	2.2	-3	80%
NSA2014	Mgmt of CIS	B13	Security attacks and mitigations	2.6	15	84%
NSA2014	Mgmt of CIS	M11	System Administration: / installation, authentication, access, backups, virtualizations, / updates/patches, logging audit...	2.2	18	75%

IT2008	Math	H01	Math and Statistics for IT	2.7	18	87%
IT2008	Math	H02	Probability and Statistics--Basic probability theory, random variables and probability distributions, estimation theo...	2.8	8	82%

Appendix 6: Distribution of Survey Items and Impact

								Totals		
# of KA's in Model	27	73	17	13	18	18	4	11	181		
# of KA's used in survey	19	47	12	7	5	2	4	11	107		
	70%	64%	71%	54%	28%	11%	100%	100%	59%		
Models	IS2002	IS2010	SW	IT	CS	CE	new	NSA	Totals		
KA's used from *Model Used on Survey*											
Database	1	9	0	0	0	0	2	2	14	13%	
Systems Dev elopment	5	16	0	0	0	0	0	0	21	20%	
Systems Design	3	7	0	0	0	0	0	0	10	9%	
Programming	1	0	10	1	3	0	0	3	18	17%	
Project Management	3	8	0	0	0	0	0	0	11	10%	
IT Components	1	0	2	1	2	2	0	0	8	7%	
Management of CIS	5	7	0	3	0	0	2	6	23	21%	
Math	0	0	0	2	0	0	0	0	2	2%	
Sum	19	47	12	7	5	2	4	11	107	100%	
Relative impact of Model: (100 * #of Survey KA's) / Total #of Model KA's)							hrs/course =			35	courses to build CIS
Database	4	12								16	0.5
Systems Dev elopment	19	22								40	1.2
Systems Design	11	10								21	0.6
Programming	4		59	8	17					87	2.5
Project Management	11	11								22	0.6
IT Components	4		12	8	11	11				45	1.3
Management of CIS	19	10		23						51	1.5
Math				15						15	0.4
Totals	70	64	71	54	28	11				298	8.5

Note: The columns "new" and "NSA" are partial specifications and are not complete model curricula. For the complete models, the number of KA's in the model are variable. Thus, the numbers of KA's in each column is normalized to 100 for comparison. The totals add up to the number of items used in the survey. To shorten the survey, items were deleted which were duplicated, or which were felt not to be relevant for an IS type curriculum. The culling process was done from an alphabetically ordered list of KA's where the sources of the KA's were not shown. Interestingly, the numbers of KA's from both the IS 2002 model and IS 2010 are very similar except for programming wherein SW (Software Engineering), CS, IT, and CE contributed significantly to the total KA's. Under a lot of assumptions, if the KA's could be taught with the efficiency indicated in 298 hours, only 8.5 courses would be needed. Some caution is indicated: the 59 score for SE represents approximately 60% of the total time for and SW degree…

Enhancing the Classroom Experience: Instructor Use of Tablets

Jeff Cummings
cummingsj@uncw.edu

Stephen Hill
hills@uncw.edu

Information Systems and Operations Management Dept.
University of North Carolina Wilmington
Wilmington, NC 28403

Abstract

Instructors continually search for innovative approaches to interact with and engage students in the classroom. The tablet offers a potential innovation for this purpose. Tablet devices from Apple, Microsoft, and other hardware vendors have overcome many of the challenges of the past (e.g. cost and limited applications) to become useful tools for instructors. Our paper examines instructor use of tablet devices by evaluating current tablet hardware and software solutions. A number of teaching tasks are evaluated to understand how they can be enhanced with tablets. These teaching tasks are utilized in a numbers of classes followed by an examination of student perceptions of tablet technologies in the classroom. Our results suggest that tablets are an effective tool with students rating instructor effectiveness as high compared to traditional classroom approaches. Based on results from student feedback, we recommend continued tablet use in the classroom.

Keywords: Classroom Innovation, Instructor Tools, Tablet, Lecture Tools, Pedagogy, Digital Ink

1. INTRODUCTION

Technological innovation continues to change the educational environment, especially within the classroom. For years, students have embraced technology both inside and outside the classroom. There are even universities shifting to an entirely wireless, tablet environment to increase student engagement in the classroom (McKeown, 2005). With the rapid adoption of technology as a classroom tool, there is an assumption that instructors are doing the same. Unfortunately, many instructors continue to rely on traditional lectures which leverage a static PowerPoint presentation using the projector and computer. However, by incorporating tablet technology both within and outside of the classroom, instructors can utilize a tool to provide improved presentation which allows for flexibility when delivering lectures while also creating a new mechanism to support interaction between instructors and the student.

Tablets are defined as pen-based computing with digital ink that enable instructors the flexibility to take notes, make comments, diagram models or create interactive presentations (Kam et al., 2005). Since Microsoft first introduced the idea of a mainstream "tablet" PC, much has changed in the tablet computing landscape. Many of the hurdles faced with incorporating the use of tablets such as cost and weight have diminished. In particular, two occurrences have reshaped an instructor's ability to integrate tablet technology into his or her teaching. First, was Apple's introduction of the iPad in 2010. Aside from increased mobility through reduced weight, the iPad was a significant change in the tablet market with widespread availability and mass adoption. Second, has been the decrease in the pricing of tablet computing devices as a whole, but particularly devices running Microsoft Windows operating system. Tablets can now be purchased well under $1000 with a robust operating system which allows users to load and operate a variety of applications not previously available for tablets.

In this article, we examine the impact of this innovation both inside and outside the classroom. We first evaluate the current hardware/software available including the pros/cons of various platforms. We then examine instructional tasks accomplished with tablet use. Finally, we evaluate some of these tasks by incorporating them into the classroom in classes across disciplines (both operations and information systems) and then survey students to understand tablet effectiveness.

2. TEACHING WITH TABLETS

Incorporating innovation into the classroom is not a new topic. Instructors continually try to take advantage of the latest technological innovations to forge new opportunities to engage students in the classroom. There continues to be an interest in leveraging the technology mediated classroom and evaluating the impact technology may have on learning (Alavi, Yoo, & Vogel, 1997). Previous research on tablets has focused on the use of tablets in the classroom with much of this research examining systems specifically designed and developed for a single purpose, lecturing in the classroom. These include systems such as ZenPad or Golub designed for some of the original tablets available in the market (Buckalew & Porter, 1994; Golub, 2004). Another example is Anderson et al. (2004) who designed a system that expanded the traditional PowerPoint to include a whitespace for notes. However, most of this research has focused on the early introduction of tablets in which the applications for instructional use was still limited. There has been little research examining the capabilities of newer tablets (e.g. iPad and Surface) and applications specifically designed for tablets.

Tablet Technology

Tablets have changed dramatically over the years. The first tablets introduced were considered Tablet PCs in the form of modified laptops providing the unique feature of the ability to input directly on the screen through the use of a stylus or touch. Many mark Microsoft's Tablet PC introduced in 2002 as the one of the first mass produced tablets available to consumers. The challenge faced during the early introduction of tablets was the cost of the device, weight and lack of applications developed specifically for tablets (Shaw, 2005). Much of the research on incorporating tablets into instructional has been limited to this early form of tablets. Since then, the tablet has evolved exponentially with the introduction of affordable tablets such as the Apple iPad and Microsoft Surface that provide increased mobility and applications designed primarily for tablet use.

With the explosion of tablets available, many instructors face the difficulty in understanding what may be needed for instructional use. Tablets available today share many common components such as built-in cameras and are Wi-Fi enabled (including some with built-in cellular capabilities). However, each has various utility when it comes to the classroom environment. The table below provides a brief overview of some of the current, popular tablets available. Each tablet was chosen to highlight some of the primary differences among tablets across the major platforms.

Tablet Type	Digitizer	OS	Apps Available
iPad (7.9"–9.7")	Passive	iOS	iTunes App Store
Surface (12")	Active	Windows	MS Store + PC Software
Nexus (7" – 10")	Passive	Android	Google Play

Table 1. Comparison of Tablets

Hardware

The screen size of the tablet can be limiting depending on how it will be utilized. Smaller screen sized tablets provide increased portability but one of the primary complaints is writing. Tablets allow users to write or "ink" documents using a stylus. The larger screens provide a larger area to utilize digital ink. The use of digital ink can also be limited depending on the type of digitizer in the tablet.

For classroom use, the type of digitizer built into the tablet (i.e. touch capability) can be significant. Tablets using the traditional, passive capacitive touch stylus (e.g. iPad) reacts to touch on the screen and not pressure. These are considered passive because it has limited capabilities built into the stylus (see figure below). Common complaints of using these devices include the lack of palm rejection (the screen cannot tell the difference from a stylus and palm), no pressure sensitivity, and no additional functions built into the stylus (e.g. eraser).

Alternatively, tablets such as the Microsoft Surface use a Wacom digitizer (or active digitizer) in the touch screen. These tablets use an active stylus which gives users additional features (e.g. right click screen and erase marks) which may be useful when writing extensively on the tablet (e.g. lecture notes). These are preferred because active digitizers are pressure sensitive and have built in palm rejection when the stylus is being used (Hoffman, 2013). Figure 1 shows both the active stylus (top) and passive stylus (bottom). The key differences are the button in the middle of the active stylus (for right clicking) and the "eraser" on top. The passive stylus has neither of these capabilities.

Figure 1. Comparison of Styluses: Active (top) vs. Passive (bottom)

Finally, one last consideration in tablet choice is the manufacturer and operating system. Knowledge of the applications or programs being used in the classroom is needed to ensure the tablet is capable of running the needed software. For example, the Microsoft Surface and other tablets running Windows operating system allows users to load tablet specific applications as well as many of the programs that would normally run on a desktop/laptop. Thus, if a program like Visual Studio will be used in class, the instructor may want to choose a Windows compatible tablet to utilize these programs without switching to another device.

Conversely, the Apple iPad is limited to applications found in the iTunes App Store or apps specifically designed for the iPad. This is similar to other tablets that use the Android operating system. Other hardware considerations include the actual configuration of the device. Devices like the surface come with USB ports on the tablet. This allows users to easily plug in external devices without using Bluetooth and can be used to expand the limited storage that is often a problem with tablets (i.e. external hard drives). Alternatively, with tablets like the iPad, you are limited to the internal storage on the device and manufacturer specific peripherals.

Applications/Software

Across all tablet platforms, there are a variety of applications available for the classroom. For Windows tablets, Microsoft Office products are now being designed for the tablet with most having the capability to "ink" on the document. For example, the instructor could use a skeleton PowerPoint presentation to write notes on and switch to Visio to draw a network diagram related to the material being covered. Excel also allows you to ink spreadsheets to help explain or point out important parts of a problem, for example. Other software applications such as OneNote (free form digital note taking and note organization software), Bluebeam Revu (full-featured PDF annotator), Jing (lightweight recorder for short screencasts and screenshot capture), and Camtasia (full-featured recorder for longer, more customized screencasts) have proven to be particularly useful in tablet environment.

Alternatively, the iPad and Android tablets have access to app stores for applications designed specifically for the device. The iTunes App Store (iPad devices) and the Google Play store (most Android tablets) are the primary stores for these platforms. Various applications exist that allow inking in an iOS or Android environment. For example, the Notability app for the iPad (an iPhone version is also available) is a popular and well-designed application that allows for annotation of PDFs and other digital files. The app can also be used for freeform note taking. LectureNotes is a similar application for Android devices.

3. INSTRUCTIONAL TASKS

While tablets can be used for a variety of instructional tasks, it may not be efficient or effective to try and implement the technology into every aspect of the classroom. For tablets to have a positive impact on the both the students and the overall performance of the instructor, the capabilities of the technology must match the tasks being performed (Bilén et al., 2009). Following the task-technology fit framework (Goodhue & Thompson, 1995), we examine classroom task characteristics and tablet characteristics to understand their effectiveness in increasing instructor performance and utilization. We draw comparisons from teaching tasks done with traditional tools (e.g. PC or Whiteboard) and the tablet. Table 2 below includes the tasks discussed in the subsequent section as well as a comparison between the benchmark tool and tablet.

Activity	**Benchmark Tool**	**Tablet**
Lecture (PowerPoint)	PC	Better
Lecture (Free form)	Whiteboard	Better
Teaching Spreadsheets	PC	Slightly Better
Grading	Hard Copy	Better
Annotation of PDFs	PC	Better
Note taking (in meetings, brainstorming, etc.)	Pen/Paper	Better

Table 2. Activity Comparison

In the following subsections, we break down a number of tasks and discuss how leveraging the tablet can improve many of the traditional approaches to interacting with students. Following this discussion, an example of a basic instructor workflow is provided to understand how these tasks are accomplished during a normal class day. Finally, to understand the effectiveness of tablets from the student's perspective, a survey was conducted across multiple courses that focused on lecturing, computer skills (e.g. Spreadsheets) and discussions (e.g. network designs). The results and implications of the survey are then examined.

Lecturing

There are two general approaches commonly used during lecture: guided, slide-based (e.g. PowerPoint) and free form, "whiteboard" instructor notes. While some argue their usefulness, classroom slides can provide advantages such as the ability to structure material in advance, the capability of providing higher quality illustrations or examples, and the ability to share/reuse material (Bligh, 2000). One of the disadvantages faced with this approach is the lack of interaction and ultimately, student engagement.

Slide-based lectures can benefit from tablet use as the instructor is no longer limited to animations or "reading" bullets on a slide. An approach we found to increase interaction is the use of skeleton slides as a lecture template. By leaving "holes" in the notes, instructors encourage student note taking while raising discussion of topics to fill in the information. This also puts the responsibility of summarizing a topic on the students. Figure 2 below is an example of a lecture on networking in which students provide the pros/cons on a slide template.

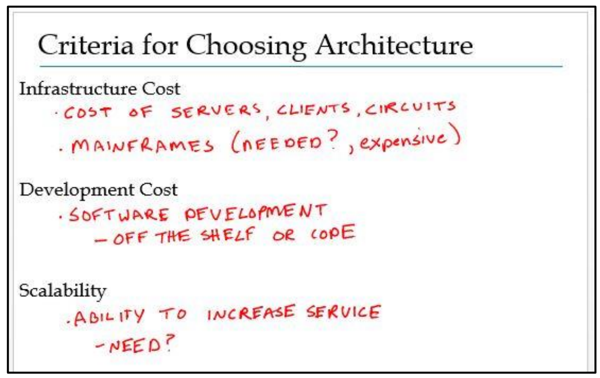


Figure 2. Digital "ink" on traditional slides

Aside from writing notes on slides, digital ink can be used for attention marks to draw student's focus to specific content. Attention marks can have a variety of purposes including grouping, navigation, progress indications and identification of key points (Anderson, Hoyer, Wolfman, & Anderson, 2004). During our lectures, the use of attention marks helped to clarify importance (e.g. key terms) as well as help in understanding formulas by drawing attention to specific parts of that formula. Overall, the benefits of approaching a traditional slide-based lecture using the tablet include (1) digital ink (or annotations) directly on the slide and (2) the ability to save these annotations to send to students.

Alternatively, instructors can use an application such as Word, Windows Journal, OneNote, or other tools to leverage the tablet as a "digital" whiteboard. This is useful for instructors preferring a more discussion based class. Using a digital whiteboard allows the instructor to capture what was discussed in class and pull up the lecture notes to confirm a specific topic was discussed or send classroom notes out to their students after the lecture. Figure 3 below shows an example of lecture notes from a digital whiteboard lecture session in OneNote.

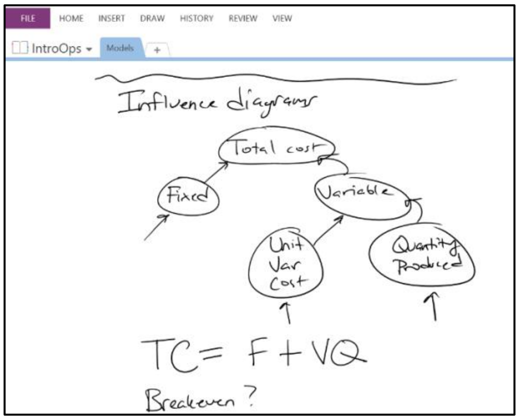


Figure 3. Digital whiteboard example

A potential challenge for lecturing can occur depending on the type of tablet being used. As previously discussed, tablets using active digitizers (e.g. Surface) are often preferred for lecturing to those using a passive digitizer (e.g. iPad). We have used both tablets in the classroom and have found the use of a passive stylus to be challenging because (1) the tablet will often pick up palm movements creating extraneous marks on the notes and (2) the writing can be cumbersome and unnatural. For lecturing extensively, a tablet using an active digitizer was preferred as it had built-in palm rejection with writing similar to that of a pen and pad.

Spreadsheets

Figure 4 is an example of annotation of a spreadsheet during a lecture. While building this model of a basic inventory management problem for use with Excel's Solver Add-in ink annotations were added to the spreadsheet. The ink annotations were saved in the spreadsheet and the spreadsheet could then, if desired, be distributed to the students.

In a non-tablet environment, an instructor wishing to annotate a spreadsheet would have to rely on Excel's Comment feature (in the Review tab of the Excel's ribbon), use a combination of text boxes and shapes, or manually type comments into cells adjacent to the cells being discussed. Each of these options is potentially distracting within the instructor's workflow. Note that Excel and the other Office applications automatically enable inking when an active stylus is near the screen. This allows for a seamless transition into inking.

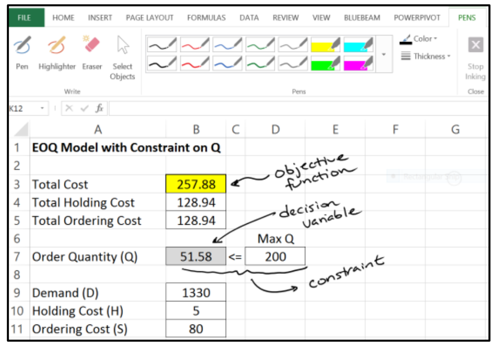


Figure 4. Digital "ink" on an Excel spreadsheet

Grading

Course management systems such as Blackboard, Camtasia, Moodle etc. have become widespread in higher education. These systems allow instructors as well as learners to post content, participate in discussions, post/view grades and engage in learning activities online (Heo, 2009). The disadvantage of these online systems is the difficulty in providing feedback on graded material. While most gradebooks allow instructors to post general comments for grades, students no longer receive detailed feedback to help improve their work moving forward.

One technique utilizing the tablet is using digital ink in grading projects. This turned out to be quite similar to grading in the traditional approach of printing out assignments and using an ink pen. The instructor now has a detailed copy of the graded material that can be "virtually" passed back to the students to help improve future deliverables. This also creates an archived copy for the instructor in the event of a grade dispute.

Note taking and PDF Annotation

In addition to classroom and grading activities, a tablet device can prove useful in other instructional areas. For example, during class preparation, digital versions of articles or case studies can be annotated. The annotations are then available to the instructor electronically or on printed version of the documents. Figure 5 shows the digital annotation of an article in BlueBeam Revu prior to class discussion.

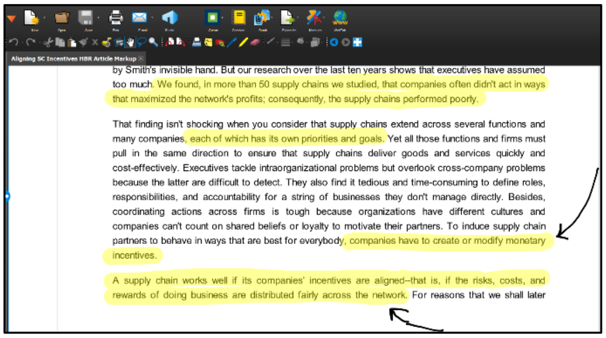


Figure 5. Article annotation during class preparation

Flipping the Classroom

A recent approach gaining popularity is the idea of "flipping the classroom." Flipping involves the reversal of a traditional college lecture by which students gather most of the information outside of class through readings, recorded lectures, podcasting, etc. The idea is to have students actively engaged in the material during class instead of the passive reception from traditional lectures (Berrett, 2012). The use of a tablet can have implications both in and out of the classroom.

Outside of the classroom, lectures can be created in conjunction with additional screen and voice capture software to create a more interactive video lecture. A challenge with flipping the classroom is transforming materials typically taught in class to an interactive form for students to view outside the classroom. Using available software such as Jing in conjunction with the tablet, an instructor can record a portion of their typical lecture or material through screen capture including audio. This enables the instructor to use digital ink to describe the topic going beyond simply providing a deck of PowerPoint slides. With this approach, students get more interactive mini-lectures outside of class reserving in-class time for exercises and application of the material.

In the classroom, the focus can be on capturing the discussion or activities occurring during the day. Whether this be through a digital whiteboard or recording of the discussion, students can reference the instructor's notes from that day. Another approach is capturing the students work on the tablet to display to the class. Many of the new projectors have built in wireless adapters allowing instructors to move freely throughout the classroom which enables students to use the tablet to solve problems or diagram models.

4. INSTRUCTOR'S BASIC WORKFLOW

In this section, we describe a basic workflow of how a tablet can be utilized in class. The workflow includes preparing for a lecture, giving a lecture, and providing lecture notes to the students after class. For a class with content primarily delivered via PowerPoint slides, we prepare two versions of each slide deck. The slide deck can be prepared in a tablet environment or on a traditional desktop or laptop computer. The instructor's version of the slide deck includes all of the content. The student's version is an outline of the PowerPoint slides and is distributed to the students prior to the class.

The outline is a version of the instructor's version of the lecture slides with a significant amount of material removed and with the expectation that students will write (or type in) the missing content as the lecture is given. This outline is distributed electronically (posted to a course website or to Blackboard) and is posted in both PDF (converted from PowerPoint via PrimoPDF or other similar PDF conversion tools) and PowerPoint format. Students are expected to download and/or print the outline prior to class.

In class, the tablet is used to either present the slide deck in PowerPoint or as PDF in a PDF annotator application (e.g., Bluebeam Revu). The tablet is connected via a VGA cable to the in-class projector which projects the image from the tablet's screen onto a screen at the front of the classroom. As the lecture proceeds, the slide deck is annotated by the instructor. The annotations can be seen in real-time via the projector. Spreadsheets (or other documents) can also be annotated during the lecture and a virtual whiteboard can be used for free-form lecturing (in the absence of or as a complement to slide decks).

After the lecture is completed, the annotated slide decks are provided to the students (either immediately or after a pre-determined period of time). Dropbox, a file storage application (available for PC, Mac/iPad/iPhone, and Android devices) is used to seamlessly synchronize files between the various devices used during this workflow.

5. STUDENT IMPRESSION OF TABLET USE

To examine the capabilities afforded in the classroom, a tablet was used as the primary instructional tool by two instructors during almost every class sessions. Multiple classes were included in the study across both information systems and operations management. Class topics were focused on business telecommunications, data analytics, supply chain management, and operations management. All classes were held in-person and incorporated both a lecture portion as well as a hands-on activity requiring use of information technology. The tablet used during the study in all classes included the Microsoft Surface Pro.

Gender	%
Male	60
Female	40
Classification	
Freshman	3
Sophomore	27
Junior	89
Senior	110
Age Range	
18 – 20	45
21 – 23	142
24 – 26	25
27 +	16

Table 2. Demographics

A survey was given at the end of the semester asking students to rate the instructor's use of the tablet. Questions focused on tablet use for presenting lectures, using a digital whiteboard and working through hands-on activities (e.g. Excel spreadsheets). While the tablet was used for some of the grading in the courses, the survey was focused on the classroom interaction and did include questions concerning grading. A total of 229 students responded to the survey. Table 2 includes the general demographics of respondents.

The survey was provided online with the first section focused on three different portions of the class which were all measured using a 5-point Likert-type scale from "Strongly Disagree (1)" to "Strongly Agree (5)". The questions were modified to reflect the current study's environment from Bilen, et al. (2009) and Stickel & Hum (2008). First, a set of questions were asked to get students experience with tablet led lectures. The lectures for the course were slide-based with students being asked to compare the use of the tablet to a traditional classroom experience. The second set of questions focused on in-class, activity based assignments utilizing the digital whiteboard. During class, these assignments included either examples/formulas as well as diagramming examples (e.g. designing a LAN). Finally, questions surrounding the use of the tablet to work through Spreadsheet exercises were also included (Note: these exercises were only in Business Analytics courses so responses were limited to 55 students). Table 3 below provides the means across the different types of courses.

Question	Mean	Std. Dev.
Lecture Slides		
L1: Compared to traditional lectures from slides, the use of the tablet made it easier to understand topics during lecture.	4.36	0.71
L2: The tablet was an effective tool for presenting the lecture material.	4.45	0.69
Whiteboard Examples		
W1: Compared to using the classroom's whiteboard, I was able to follow the instructor better.	4.34	0.85
W2: Compared to using the classroom's whiteboard, the tablet was a more effective tool for working through problems in class.	4.32	0.84
Spreadsheet Exercises		
S1: I found the tablet to be a useful tool in learning spreadsheet material.	4.27	0.83
S2: The tablet was an effective tool for presenting the spreadsheet material.	4.35	0.70

Table 3. Classroom Experience

We did find some variance across the courses. The IS course (i.e. Business Telecommunications) found the tablet to be slightly more useful for Whiteboard Examples (W1 = 4.52 & W2 = 4.44) compared to the Operations classes surveyed (W1 = 4.31 & W2 = 4.29). This suggests digital whiteboards may be more effective for diagrammatic purposes compared to general examples. However, across all classes, students appeared to embrace the use of the tablet and found them to beneficial to their classroom experience.

Finally, students were asked to give the overall impressions of the instructor. One question consisted of assessing the instructor's effectiveness during the course using the same scale previously described. The last questions asked students if they recommend continued use of the tablet and was measured on a 5-point Likert style scale from Highly Recommend (5) to Highly Discourage (1). Table 4 below includes the questions, means and standard deviations.

Question	Mean	Std. Dev.
Effectiveness		
E1: Did the use of tablet technology improve the overall teaching effectiveness of the instructor?	4.25	0.72
Continued Use		
CU1: Would you recommend or discourage that instructors use tablet devices to display lecture notes, work example problems, and demonstrate use of software?	4.34	0.75

Table 4. Impressions of Instructor

Again, we found slight variations between the operations and information systems courses. For the information systems course, student responded higher for both effectiveness (E1 = 4.51) and continued use (CU = 4.56) compared to the operations courses (E1 = 4.23 and CU = 4.31). Regardless of these differences, across all courses, students appeared to feel the tablet did increase the instructor's effectiveness and suggested continued use in future classes.

6. CONCLUSION

Tablets provide an innovative way to incorporate new technologies in the classroom. Our results suggest the tablet is an effective tool to use for both presenting lectures and working through sample problems/exercises. Additionally, students suggested continuing to incorporate the tablet in the classroom as it increases the overall effectiveness of the instructor.

The study was limited to only a few courses that primarily focused on networking and operations. Future studies are needed to examine the effectiveness of the tablet in other IS courses but the initial results suggest these courses could be improved. Instructors must also consider other potential issues that may arise from tablet use. One potential issue is technology disruption. If the entire class is structured around the tablet, instructors should be prepared if the tablet fails. Additionally, previous research has raised concerns such as an instructor learning curve. Milanovic (2006) raised some issues around writing with an electronic pen. The use of a tablet with an active digitizer may alleviate these concerns as this type of stylus is very similar to writing with a normal ink pen.

To keep students engaged in learning, instructors continually look for new, innovative approaches. Tablets provide a relatively easy, cost effective solution to increase interaction during lectures while engaging students. In this paper, we have provided a few approaches to instructional use of tablets. With the evolving nature of the tablet and app market, there continues to be new and effective ways to incorporate the tablet for instructional use. This paper is meant to provide some general guidelines for instructors wanting to implement the tablet in their courses.

7. REFERENCES

Alavi, M., Yoo, Y., & Vogel, D. R. (1997). Using information technology to add value to management education. *Academy of management Journal, 40*(6), 1310-1333.

Anderson, R., Anderson, R., Simon, B., Wolfman, S. A., VanDeGrift, T., & Yasuhara, K. (2004). *Experiences with a tablet PC based lecture presentation system in computer science courses.* Paper presented at the ACM SIGCSE Bulletin.

Anderson, R., Hoyer, C., Wolfman, S., & Anderson, R. (2004). *A study of digital ink in lecture presentation.* Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems.

Berrett, D. (2012). How 'Flipping'the classroom can improve the traditional lecture. *The Chronicle of Higher Education, 19*.

Bilén, S. G., Lee, D., Messner, J. I., Nguyen, H. T., Simpson, T. W., Techatassanasoontorn, A. A., & Devon, R. F. (2009). *Tablet PC Use and Impact on Learning in Technology and Engineering Classrooms: A Preliminary Study.* Paper presented at the Workshop on the Impact of Pen-Based Technology on Education.

Buckalew, C., & Porter, A. (1994). *The Lecturer's Assistant.* Paper presented at the 25th SGCSE.

Golub, E. (2004). *Handwritten Slides on a TabletPC in a Discrete Mathematics Course.* Paper presented at the 35th SIGCSE.

Goodhue, D. L., & Thompson, R. L. (1995). Task-Technology Fit and Individual Performance. *MIS Quarterly, 19*(2), 213-236. doi: 10.2307/249689

Heo, M. (2009). Design considerations for today's online learners: A study of personalized, relationship-based social awareness information. *International journal on E-Learning, 8*(3), 293-311.

Hoffman, C. (Producer). (2013). Not All Tablet Styluses Are Equal: Capacitive, Wacom, and Bluetooth Explained. Retrieved from http://www.howtogeek.com/177376/not-all-tablet-styluses-are-equal-capacitive-wacom-and-bluetooth-explained/

Kam, M., Wang, J., Iles, A., Tse, E., Chiu, J., Glaser, D., . . . Canny, J. (2005). *Livenotes: a system for cooperative and augmented note-taking in lectures.* Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems.

McKeown, J. (2005). Mobile, Agile, Versatile: The Use of Tablet PCs and Wireless Technology in Introductory Programming. *Information Systems Education Journal, 3*(5), 1-7.

Milanovic, Z. (2006). Lecturing with a virtual whiteboard. *The Physics Teacher, 44*(6), 354-357.

Shaw, K. (2005, May 13, 2014). Will the Tablet PC ever become mainstream? *Network World*, from http://www.networkworld.com/newsletters/2005/0613mobile1.html

Stickel, M., & Hum, S. V. (2008). *Lessons learned from the first-time use of tablet PCs in the classroom.* Paper presented at the Frontiers in Education Conference, 2008. FIE 2008. 38th Annual.

Editor's Note:

This paper was selected for inclusion in the journal as an ISECON 2014 Meritorious Paper. The acceptance rate is typically 15% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2014.

Why Phishing Works: Project for an Information Security Capstone Course

Lissa Pollacia
lpollaci@ggc.edu

Yan Zong Ding
yding@ggc.edu

Seung Yang
syang2@ggc.edu

Information Technology
Georgia Gwinnett College
Lawrenceville, GA 30043, USA

Abstract

This paper presents a project which was conducted in a capstone course in Information Security. The project focused on conducting research concerning the various aspects of phishing, such as why phishing works and who is more likely to be deceived by phishing. Students were guided through the process of conducting research: finding background and related work on the topic, determining the hypothesis, development of the survey system, data collection, analysis of the results, and writing of the academic paper. This project was very successful in that students gained in-depth knowledge about phishing, developed an understanding of research and academic writing, and learned to statistically analyze data to support or refute their hypothesis. Educators who are teaching a capstone course in Information Security may be interested in this project because it is an appropriate level for undergraduate seniors, it can be accomplished in one semester, and the participants can be other students at the institution.

Keywords: Security Capstone Course, Security Research Project, Phishing, Student Research

1. INTRODUCTION

Many undergraduate programs in Information Security require a capstone course at the senior-level. This paper describes a project that is appropriate for a capstone course in information security. The authors conducted this project in the capstone course for three semesters. It was successful in achieving the following goals for students in the course: (1) develop a deeper understanding of one area of information security, (2) learn how to conduct research in the computing field, and (3) learn how to write an academic paper.

The project focuses on *phishing*, a type of attack in which attackers use spoofed (phishing) email to deceive users and motivate them to visit and reveal confidential information at fraudulent (phishing) websites. These websites are designed to closely mimic and impersonate real, legitimate sites. Each year phishing attacks succeed in scamming millions of users and stealing billions of dollars from the victims (Hong, 2012). The purpose of the project was to answer questions such as "why phishing works?" and "who is more likely to fall for phishing?"

The project was conducted for three semesters and was successful from both practical and pedagogical viewpoints. The project had sufficient depth and provided challenging material for the students; however it could be completed in one semester. In addition, the study was structured so that there are no consequences to the study participants, thus the project was readily approved by our Human Subjects Board (HSB).

The project also supports the push in STEM education to provide more opportunities for *Scientific Inquiry* (Yager, 2009; Zubrowski, 2009). The National Research Council (NRC) defines Scientific Inquiry to be activities in which learners study a question, formulate hypotheses, collect and evaluate evidence, and then communicate and justify their conclusions (NCR, 2006). Many scientific educators believe that scientific inquiry is critical to helping students develop 21st-century skills and knowledge that are needed to be successful today (Rhoton, 2010).

2. RELATED WORK

The past decade saw a great deal of research activities in the area of phishing. See the excellent survey of Hong (2012) for the state of phishing.

Dhamija et al. (2006) conducted the first published study of phishing. In the study, each participant was shown 20 websites, some real and some fake, and was asked to determine whether each given site was legitimate or fraudulent. For sites that they determined to be fraudulent, the participants were also asked to give their reasons for their decisions. The study found that well designed phishing sites fooled over 90% of the participants. Many participants did not verify the correctness of the sites' URLs or were not able to distinguish between legitimate and fraudulent URLs. Even fewer understood the SSL security indicators, such as "HTTPS" in the URL, the padlock icon, and the certificate. Many participants incorrectly based their decisions on how professional the content of the viewed web pages look, failing to understand that the content of a web page can be easily copied. Moreover, visual deception attacks successfully fooled even the most experienced participants. Examples of visual deception include using visually deceptive text in closely mimicked URLs (e.g. using the number "1" in place of the letter "l", or using two "v"s for a "w"), hiding a hyperlink to a rogue site inside an image of a legitimate hyperlink, and using an image of a real site in the content of a phishing page. Following the work of Dhamija et al. many other researchers led similar studies which show that the findings of Dhamija et al. continue to hold and users remain vulnerable to phishing (Hong, 2012).

Downs et al. (2006) conducted the first study of phishing email messages (as opposed to phishing websites) and how users respond to them. Just as in the case of judging websites (Dhamija et al., 2006), the study of Downs et al. found that users often base their judgments of email messages on incorrect heuristics. Users fall particularly for spear phishing, which involves email messages sent to a specifically targeted group, such as members of a community, employees of an organization, or customers of a business. For example, users who have an account at a company would tend to trust email messages that appear to be sent from the company, and many think that since the company already had their information, it would be safe to give it again. The findings of Downs et al. were confirmed in the work of Jagatic et al. (2007), which showed that people were 4.5 times more likely to fall for social phishing, i.e. phishing email sent from an existing contact, than standard phishing attacks, and it is for this reason that criminals heavily target online social networking sites. Moreover, social phishing was more successful when the phishing email messages appeared to be from a person of the opposite gender.

Dodge et al. (2007) performed a study of the effectiveness of phishing at the United States Military Academy (USMA West Point) over a period of two years. The participants of the study were the entire student body of USMA. Over time the authors developed a system that periodically generates phishing email messages, sends the messages to students, and tracks the students' responses to these messages. The study showed a failure rate of approximately 40%, that is, about 40% of the spoofed messages that appeared to be sent from an administrative office within USMA resulted in a student clicking an embedded link in the message and disclosing confidential information to unauthorized users, or opening attachments that could potentially contain malicious code.

Sheng et al. (2010) conducted the first large-scale study of demographic factors in susceptibility to phishing. They found that women were more susceptible to phishing than men, likely because women appeared to have less exposure to technical knowledge and training than men. They also found that young participants of ages 18 to 25 were more susceptible to phishing than other age groups, possibly because that they had less experience and less exposure to education and training in computer security. In the meantime, the authors found that good educational materials reduced participants' chance of falling for phishing by 40%.

Since lack of knowledge is the primary reason why users fall for phishing, many researchers studied the effects of education and training in helping users prevent phishing (Kumaraguru et al., 2007; Sheng et al., 2007; Kumaraguru et al., 2009; Kumaraguru et al., 2010). Kumaraguru et al. found that simply emailing anti-phishing materials to users is ineffective, as people are used to receiving and ignoring such warning (Kumaraguru et al., 2007). They found that users learn more effectively in embedded training, where users are presented training materials after they fall for an attack. Kumaraguru et al. developed an embedded training system called PhishGuru (Kumaraguru et al., 2009; Kumaraguru et al., 2010). PhishGuru periodically sends simulated phishing email messages to users in training, and when users fall for such a message, they receive an intervention email message that explains to them that they are at risk for phishing attacks and teaches them how to protect themselves against phishing. Study showed that with this approach, participants' chance of falling for phishing reduced by 45%, even one month after the training. Sheng et al. developed an educational game called Anti-Phishing Phil that teaches users basic security concepts related to phishing, and then tests users on what they learned (Sheng et al., 2007; Kumaraguru et al., 2010). Studies showed that this approach improved novices' ability to identify phishing by 61%.

Our information security capstone project was very similar in nature to and draws from the methodologies of the above-mentioned studies on phishing. The main difference is that those studies were conducted by professional researchers, whereas our project was for undergraduate seniors in a capstone course. We are not aware of published scholar articles on capstone courses in information security. However, there is a wealth of literature on capstone courses in IT or IS related disciplines. All those articles show that capstone projects benefit students and add values to a program of study. For instance, Dunlap (2005) shows, based on the analysis of student's outcomes in a software engineering capstone course, that capstone projects promote problem-based learning which enhances students' self-efficacy in learning and problem solving. Such self-efficacy is crucial for remaining competitive in computing related fields that are constantly and rapidly evolving. Gupta & Wachter (1998) and Lesko (2009) show that capstone projects bolster critical thinking and stimulates students' creativity to integrate various concepts and skills, apply the integrated skills to solve problems, and acquire practical knowledge. Our capstone experience confirms all these findings.

There is also literature on methods to deliver capstone courses. Lynch et al. (2004) define four models of delivery. The first is the industry-sponsored model, where students play the role of early career employees within a company. The second is the studio model, where students collaborate with experts and mentors. The deliverables are defined, but their content is flexible. The third is the traditional model, where students collaborate in teams. The deliverables are defined, but there is little interaction with and support from the faculty. The fourth is the directed model, where students form small groups and work closely with the faculty. The groups are provided with a clearly defined set of requirements, milestones and deliverables. The directed model is the model we adopted to deliver our capstone course.

3. THE PROJECT

This section describes the capstone project and how it was organized and implemented. The course was Information Technology (ITEC) 4810, Systems and Security Capstone. The authors taught the course in Spring 2012, Fall 2012, and Spring 2013. The work was self-contained, i.e., the work completed in one semester. The authors were assigned to team-teach the course.

Students were divided into groups of three. We found that three students per group worked better than four per group, because with a smaller group size each student had a sufficient amount of responsibility. The purpose of the project was to research various aspects of *phishing*, using students on our campus as participants. The research attempted to answer questions such as “Do people recognize certain indicators of phishing?” and “Which participants are more likely to fall for phishing?” Students in the class were able to choose their own questions and generate hypotheses, which were then tested using data collected by a web-based survey system (which they also developed).

We delivered the capstone course using the *Directed Method* as defined by Lynch et al. (2004). We not only defined the project, but also organized the project into components and subcomponents, and set a timeline of milestones and deliverables. For each subcomponent we covered the background and tools that the students needed to complete the deliverables on time. We chose to adopt this method of delivery because the students had never been involved in this type of project before.

On the other hand, good planning and organization by the instructors and proper guidance to the students would make the project more accessible and manageable. There are two major components of the project that run concurrently throughout the semester. One was the “research” component of the project, in which students conducted the research, collected the data, analyzed the data, and wrote a paper about the project and the results. The second component was the “development” side, which consisted of developing the web-based survey system. The survey system was used to collect the data and test the hypotheses. The research component is discussed below in this section, and the development component is discussed in Section 4.

The research component of the project requires that the students:

(1) Acquire fundamental knowledge of phishing

(2) Conduct library research into the current phishing literature

(3) Determine one or more hypotheses

(4) Create web pages and email messages, and develop survey questions

(5) Statistically analyze the data

(6) Interpret the results of the analysis and write the academic paper

Acquire fundamental knowledge of phishing
The students in the capstone course were primarily seniors in the Systems and Security concentration, and therefore had some fundamental knowledge of phishing. However, to immerse them into the topic, we required the students to read three in-depth articles about phishing: (1) “Why phishing works” (Dhamija et al., 2006); (2) “You’ve been warned: An empirical study of the effectiveness of Web browser warnings” (Egelman et al., 2008); and (3) “The State of Phishing Attacks” (Hong, 2012). We assigned discussion questions and created discussion forums on these articles in the online learning management system. The students were required to participate in these online discussions, as well as in-class discussions.

Conduct library research into current phishing literature
We contacted the library staff at our institution, who taught a short course on conducting research using our library resources. Although most of the students had been through a similar presentation in the past, they indicated that it was helpful to have a review of these research skills, particularly with respect to the current topic.

Each group was given the assignment to find at least three additional papers related to phishing. After reading these papers, they were required to write the “Background and Related Work” section of their own paper and give an oral presentation in class. The presentations not only gave students experience in public speaking, but also increased their breadth of knowledge concerning phishing.

Determine the hypotheses
Each student in the group was required to develop at least one research question and hypothesis. Therefore, each group would research at least three hypotheses. Some examples of hypotheses are:

- Male participants are able to identify phishing attempts better than female participants.
- Information Technology majors will be more likely to identify phishing attempts than non-IT majors.
- Phishing email is more effective if it contains familiar content or comes from a source that participants recognize.
- Over 50% of participants will be unable

to identify a phishing site when the URL is the only indicator.

- Users with training are less susceptible to phishing than those without training.

Students queried the database containing data collected by the survey system, and then tested all these hypotheses by performing statistical analysis. In order to help students test the last hypothesis on the above list, we did a simple control study. That is, we selected a small control group of participants, and gave a short training session on phishing basics before the control group participated in the study. The data for the control group and that for other participants are stored separately. It may be interesting to note that of the list of sample hypotheses given above, the first one (male vs. female) was refuted, while all the others were confirmed in the study.

Create web pages and email messages, and develop survey questions

Once the hypotheses were developed, then the groups were required to create web pages and email messages, some of which were legitimate and some of which were phishing attempts. These were to be presented to the survey participants as images of the web pages and email messages. Due to restrictions placed by our institution's Human Subjects Board (HSB), the participants would not interact directly with a live phishing site or a live phishing email message, but rather with static images of the phishing site or messages. The survey questions from all of the groups were collected and organized into one cohesive survey.

When participants entered the survey, they first viewed and accepted the Informed Consent information, which was required by the HSB. Next came a demographics form, which collected demographic data such as sex, age, major, class level, etc. that was needed to analyze the hypotheses. This was followed by 10 to 12 screens, which displayed the images of real or phishing web pages and email messages. For every image, the participant was asked to identify whether this was legitimate or fake (phishing). We used a 4-point Likert scale: Strongly Agree, Agree, Disagree, and Strongly Disagree. We did not include the neutral option (Neither Agree nor Disagree), as we wanted the participant to choose one way or the other. Appendix 1 shows an example of an image and survey question presented to participants. This is an example of a phishing site which mimics the site of Fidelity Investments (note the misspelling of the word "fidelity" in the URL.)

Those who identified this screen as a phishing attempt, were asked a follow-up question to further identify which indicator led them to this conclusion. Indicators included bogus URLs, lack of a padlock, strike-through of https, and errors in the content of the page or message. This gives more detail concerning what "gave away" the image as a phishing site or email. Appendix 2 shows the follow-up image and question to the image shown in Appendix 1. Note the boxes surrounding such areas as the URL, the menu, the logo, and the content.

Creating images of phishing sites

Creating images of phishing sites was the component that students enjoyed the most. Making an image of a real site is relatively easy – one simply visits the site and takes a screenshot. A vast majority (about 80%) of the images used in the survey were images of phishing sites, and we encouraged students to be as creative as possible in creating those. However, developing a realistic phishing image is non-trivial. Simple approaches such as using photo-editing software to modify the image of a real site do not work, as they do not produce realistic looking images. The best approach was for the student to create a phishing site first, then take an image of that. We demonstrated a few tools for constructing phishing sites. For instance, we introduced a web crawler known as HTTrack that allows one to copy the content of an entire website to a local computer and based on that come up with a site that mimics the original site.

The students installed and configured a DNS server to establish a phishing site with a desired domain name. Then they set up a web server with HTTPS and created a privately signed certificate. Next they copied the mimicked legitimate web site contents using HTTrack and modified the source code to create the desired content for the phishing site. The students also installed an SMTP (sendmail) server and created phishing email messages.

Ethics

During this part of the course we repeatedly emphasized professional ethics. We reiterated that these tools were introduced in order to complete their project and to better understand phishing attacks. We emphasized that any attempt to use those tools to launch phishing attacks is considered criminal behavior. These students are in the Systems and Security concentration of our ITEC major, therefore it is important that they thoroughly understand the professional ethics involved. All students understood this and took it seriously.

Statistically analyze the data

The students in the Systems and Security concentration are required to take a statistics course, and therefore have some statistical foundation. However, the statistics course is at the sophomore level, and the students in the capstone course are mostly seniors. Therefore, a short review of hypothesis testing was needed (approximately one week of classes).

We chose to model the responses in the binomial form by summing Strongly Agree/Agree and Strongly Disagree/Disagree responses separately. Students analyzed the data based on the following steps for hypothesis testing:

Step 1. Develop the null and alternative hypotheses
Step 2. Specify α
Step 3. Decide if the test is one-tail or two-tail
Step 4. Choose the formula and calculate the test statistic
Step 5. Compute the p-value
Step 6. If p-value $\leq \alpha$, then reject Ho and fail to accept Ha; otherwise accept Ho

The groups were required to turn in a document containing a formal statement of their hypotheses, along with the statistical analysis, which they had conducted. This document would become the next section of their written paper.

For example, one group had a hypothesis which states that most (i.e. over 50%) of users will be unable to identify a phishing site when the URL is the only indicator of phishing. Formally, the alternative hypothesis Ha and null hypothesis Ho are given as follows:

- Ha: More than 50% of attempts will fail to identify phishing websites that have fraudulent URLs as the only indicator of phishing.
- Ho: Less than or equal to 50% of attempts will fail to identify phishing websites that have fraudulent URLs as the only indicator of phishing.

All of the groups together had developed four phishing pages that have fraudulent URLs as the only indicator of phishing. Therefore, participants' answers to the survey questions for these four images allowed this group to test the hypothesis. Answers that chose "strongly agree" or "agree" are considered correct, and answers that chose "disagree" or "strongly disagree" are considered incorrect. This is obviously a one-tail test. The group set α to be 0.05. Applying the correct formulas to the survey data, the group computed a t-value, and then computed a p-value using the Microsoft Excel TDIST function. The resulting p-value is tiny and significantly less than α. Therefore, with a confidence level of 95%, the group rejected the null hypothesis and concluded that most people would fail to identify phishing websites when the URL is the only indicator for phishing. Appendix 3 shows the summary of the group's statistical test for this hypothesis.

Interpret the results of the analysis and write the academic paper

For this component of the project, we followed these characteristics of Scientific Inquiry, as defined by the National Research Council (NRC):

- Learners formulate explanations from evidence to address scientifically oriented questions.
- Learners evaluate their explanations in light of alternative explanations.
- Learners communicate and justify their proposed explanations.

For this section of their academic paper, students were asked to explain what the statistical evidence had shown. This was accomplished by having group meetings, meetings of group members and faculty members, and also general class discussion. We found students in other groups could be quite helpful in discussing and offering explanations.

As previously stated, two purposes of the course are to engage students in scientific inquiry and to learn how to write an academic paper. By the time we reached this point in the semester, most sections of the paper were already written. Essentially, all that remained was the results of the analysis and the conclusions. The advantages of this approach are two-fold: (1) We were able to review sections of the paper and make suggestions/corrections as the project unfolded, and (2) students were not overwhelmed by the task of writing an entire paper at the end of the semester.

Each group gave an oral presentation at the end of the semester, which supports the NRC goal of communicating and justifying their proposed explanations.

4. TECHNICAL DEVELOPMENT OF THE SURVEY SYSTEM

The survey system was developed using LAMP (Linux, Apache 2.2.3 web server, MySQL 5.0 database, and PHP 5.1.6 in CentOS 5.3). This platform was chosen because it is a popular platform for website development and has a large supporting community.

The development process was divided into four tasks: database development, web page development, phishing image development, and creation of a survey system, including OS and software installation. The survey system development tasks were assigned to two development groups: a database development group and a web development group. The database development group designed the database schema and ported the web page and email images into the database. The web development group installed LAMP software and created web pages for the survey. The survey system development time was tight, so we used the Rapid Application Development (RAD) model to minimize planning time and get a working system as soon as possible. Using RAD, revisions of the system occur as it is being developed.

The database development group created a database schema to store the survey questions and the data that would be collected. The major challenge faced by the group was to design a logical data model with many unknown factors and implementing it within a very short period of time. The logical data model had to support various survey types, which were unknown at the time of its design. The survey system needed to be ready by approximately two-thirds into the semester so that the students could test their survey questions with phishing images before the system was released to participants.

As shown in Appendix 4, the group tried to make the database schema as simple as possible but also general enough to support various types of survey questions. The group created the initial conceptual model using E-R diagrams with justifications. They collected feedback from the other students and revised the design as needed. Then the group normalized the database, and developed the physical design of the database structure with MySQL. The group worked together with other students to put the survey questions and images into the database, then to test and debug the survey system.

Designing the reusable survey web pages was one of the key challenges for the web page development group. The web pages are designed to support different types of survey questions with minimal effort. To this end, the web pages were categorized into start and end pages, main question pages, follow-up question pages, and survey result report pages.

The start and end pages consisted of the informed consent form and an end-of-survey thank-you page. To each survey instance, the start page associated a session ID that was used to identify the instance. By utilizing the session IDs, the survey did not need to collect personal information from participants, but was still able to uniquely identify each survey instance.

The main question pages loaded question statements and a related image from the database using an internal question number. Then it recorded the participant's answer in the database. The same web page source code was reused by changing the question number until all of the main questions were given to the participant. The follow-up question pages also used the same source code, but were only displayed when a participant indicated that the image from the main question page was a phishing site or email image. Appendix 5 illustrates how the types of pages were used in the process flows.

The initial version of the survey system was developed in Spring 2012. The system was further enhanced by students in subsequent semesters.

5. CONCLUSIONS

This paper describes the project in an Information Security capstone course that the authors jointly taught in three separate semesters. The course was delivered using the directed method (Lynch et al., 2004), where milestones and deliverables are clearly defined and students are provided with necessary background and tools to complete the deliverables on time. The capstone project was very successful in that we were able to achieve the objectives: (1) students will develop a deeper understanding of one area of information security, (2) learn how to conduct research in the computing field, and (3) learn how to write an academic paper. Under proper organization and guidance, the students were able to complete a research project that was previously conducted by reputable professional researchers. The components of the capstone project reinforced and integrated skills in information security, networking, systems analysis and design, database design and implementation, web development, software development and testing, and statistics. The project was accomplished in one semester with very little additional resources other than the authors' expertise.

Additional experience with "soft skills" also occurred as a result of this project. Students learned to communicate and work with various groups, each having different responsibilities, and to coordinate their efforts. They were also responsible for contacting other instructors at our institution to solicit participants to take the survey. This was excellent experience for the workplace environment in which strong communication and teamwork skills are highly valued.

6. REFERENCES

Dhamija, R., Tygar, J.D., & Hearst, M. (2006). Why phishing works. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 581-590.

Dodge, R.C., Carver C., & Ferguson, A.J. (2007). Phishing for user security awareness. *Computers & Security*, 26(1), 73-80.

Downs, J.S., Holbook, M.B., & Cranor, L.F. (2006). Decision strategies and susceptibility to phishing. *Proceedings of the SOUPS Symposium on Usable Privacy and Security*, 79-90.

Dunlap, J. C. (2005). Problem-based learning and self-efficacy: How a capstone course prepares students for a profession. *Educational Technology Research and Development*, 53(1), 65-83.

Egelman, S., Cranor, L.F., & Hong, J. (2008). You've been warned: An empirical study of the effectiveness of Web browser warnings. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 1065-1074.

Gupta, J. N. D., & Wachter, R. M. (1998). A capstone course in the information systems curriculum. *International Journal of Information Management*, 18(6), 427-441.

Hong, J. (2012). The State of Phishing Attacks. *Communications of the ACM*, 55(1), 74-81.

Jagatic, T.N., Johnson, N.A., Jakobsson, M., & Menczer, F. (2007). Social phishing. *Communications of the ACM*, 50(10), 94-100.

Kumaraguru, P., Cranshaw, J., Acquisti, A., Cranor, L.F., Hong, J., Blair, M.A., & Pham, T. (2009). School of phish: a real-world evaluations of anti-phishing training. *Proceedings of the 5th Symposium on Usable Privacy and Security*, 1-12.

Kumaraguru, P., Rhee, Y., Sheng, S., Hasan, S., & Acquisti, A. (2007). Getting Users to Pay Attention to Anti-Phishing Education: Evaluation of Retention and Transfer. *Proceedings of the Anti-Phishing Working Group's Second Annual eCrime Researchers Summit*, 70-81.

Kumaraguru, P., Sheng, S., Acquisti, A., Cranor, L.F., & Hong, J. (2010). Teach Johnny Not to Fall Phish. *ACM Transaction on Internet Technology*, 10(2), 1-31.

Lesko, C. J. (2009). Building a framework for the senior capstone experience in an information computer technology program. In *Proceedings of the 10th ACM conference on SIG-information Technology Education*, 245-251.

Lynch, K., Goold, A., & Blain, J. (2004). Students' pedagogical preferences in the delivery of IT capstone courses. *InSITE 2004 Informing Science and IT Education Joint Conference*, 431-442.

National Research Council (2000). Inquiry and the national science education standards: A guide for teaching and learning. Washington D.C.: National Academy Press.

National Research Council (2006). How Students Learn: History, mathematics and science in the classroom. Washington D.C.: National Academy Press.

Rhoton, Jack (2010). Science Education Leadership: Best practices for the new century.

Sheng, S., Holbrook, M.B., Kumaraguru, P., Cranor, L.F., & Downs, J.S. (2010). Who falls for phish? A demographic study of phishing susceptibility and effectiveness of interventions. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 373-382.

Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L.F., Hong, J., & Nunge, E. (2007). Anti-Phishing Phil: The design and evaluation of a game that teaches people not to fall for phish. *Proceedings of the Third Symposium on Usable Privacy and Security*, 88-99.

Smithenry, Dennis W. and Gallagher-Bolos Joan A. (2009). Whole-Class Inquiry: Creating student-centered science communities. Arlington, VA: National Science Teachers Association: NSTA Press.

Yager, Robert E. ed. (2009). Inquiry: The Key to Exemplary Science. Arlington, VA: National Science Teachers Association: NSTA Press.

Zubrowski, Bernard (2009). Exploration and the Meaning Making in the learning of science: Innovations in Science Education and Technology Series. Breinigsville, PA: Springer.

Appendices

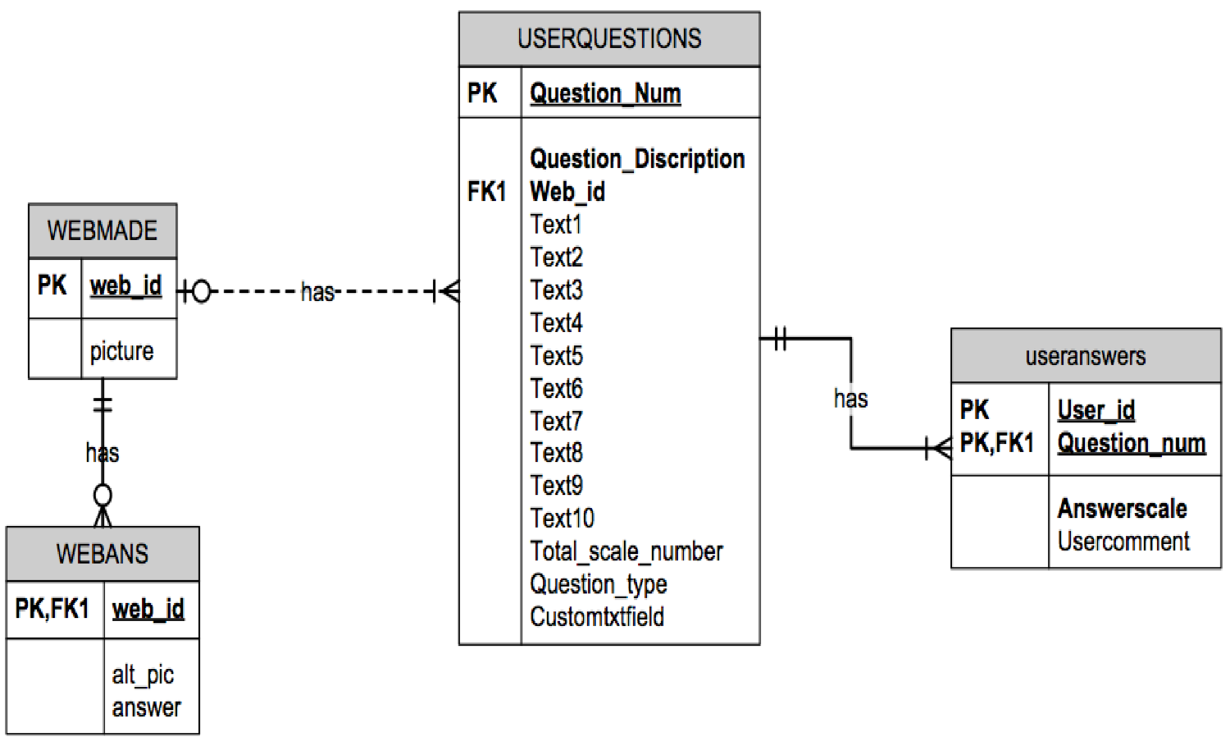
Appendix 1. Sample Survey Image

Appendix 2. Follow-up Question

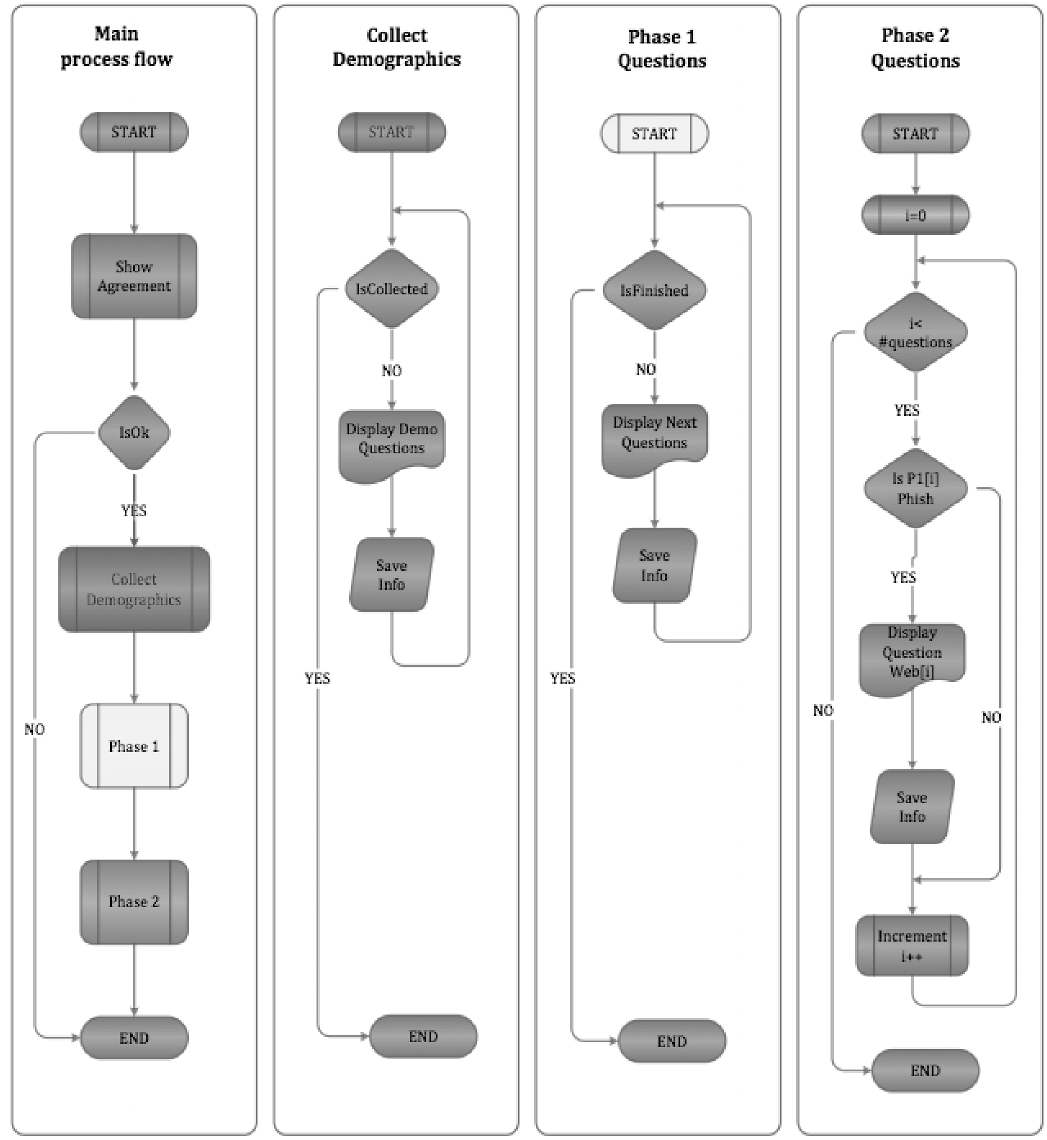
Appendix 3. Sample hypothesis testing by a group

URL identifier accuracy				Ha: µ >	0.5
				Ho: µ ≤	0.5
n	272				
Xbar	37				
σ	10.61445555				
α	0.05				
t	56.71260465				
p-value	1.1512E-152				
Conclusion: Since p is <= alpha, we can reject the null hypothesis					
Therefore it is our conclusion that when the URL is the only identifier					
of a phishing website, people most likely fail to notice they are on a fake website.					

Appendix 4. E-R Diagram for the Survey System Database



Appendix 5


Main process flow
START
Show Agreement
IsOk
YES
NO
Collect Demographics
Phase 1
Phase 2
END
Collect Demographics
START
IsCollected
NO
YES
Display Demo Questions
Save Info
END
Phase 1 Questions
START
IsFinished
NO
YES
Display Next Questions
Save Info
END
Phase 2 Questions
START
i=0
i< #questions
YES
NO
Is P1[i] Phish
YES
NO
Display Question Web[i]
Save Info
Increment i++
END

Teaching Business Intelligence through Case Studies

James J. Pomykalski
pomykalski@susqu.edu
Accounting and Information Systems
Susquehanna University
Selinsgrove, PA, 17870, USA

Abstract

In teaching business students about the application and implementation of technology, especially involving business intelligence, it is important to discover that project success in enterprise systems development efforts often depend on the non-technological problems or issues. The focus of this paper will be on the use of multiple case studies in an information systems strategy course, taught to business majors, which highlight the importance of non-technological factors. Each of the cases reinforces the need for senior management support, effective change management procedures, focus on data acquisition and quality, attention to key business process, and the integration into the existing organizational infrastructure as key drivers in project success. This approach utilizes the work system framework as a basis for case study analysis.

Keywords: Business Intelligence, Project Success, Case Studies, Work System Framework, Business Students.

1. INTRODUCTION

Gartner defines business intelligence (BI) as:

> An umbrella term that includes the applications, infrastructure and tools, and best practices that enable access to and analysis of information to improve and optimize decisions and performance (Gartner, n.d.)

The goal of BI is to provide decision makers access to valuable information and knowledge by leveraging data; the value of business intelligence "is realized in the context of profitable business action" (Loshin, 2003, p. 6).

Being able to access and analyze information is a skill expected of the majority of business professionals. BI is a "new" set tools and techniques, largely borne out of information system developments that must be understood to effectively make critical decisions. Information system (IS) students are well placed to lead in the use of BI due to their technical background and the fact that "the information systems function in an organization has a broad responsibility to plan, develop or acquire, implement, and manage an infrastructure of information technology (computers and communications), data (both internal and external), and enterprise-wide information processing systems" (Topi, et al., 2010, p. 73).

2. COURSE LEARNING OBJECTIVES

As the boundary that divides business and information systems erodes "organizational managers [need to] recognize how integral knowledge and information management are to the bottom line" (Loshin, 2003, p. xiii). Future organizational managers are current business students who generally get exposed to information systems (IS) topics through a single course. A key focus, of this singular IS course, should be on the issues involved in the development of information systems to solve specific business problems.

The basic skill sets for business students in regard to information systems are summarized in Ives, B., Valacich, J. S., et al. (2002). The authors cite that business students should be aware of eight "key information systems concepts". In particular, with regard to business intelligence systems, business students need to know (Ives, et al., 2002):

- How do information systems influence organizational competiveness?
- Why are technology infrastructures so important to modern organizations?
- What are the unique economics of information and information systems?
- How do information systems enable organizational processes?
- How do organizations develop, acquire and implement information systems?

The IS 2010 Model Curriculum focuses its capstone course (IS 2010.07) on these questions. The IS 2010.07 course "explores the issues and approaches in managing the information systems function in organizations and how the IS function integrates/supports/enables various types of organizational capabilities. It takes a senior management perspective in exploring the acquisition, development, and implementation of plans and policies to achieve efficient and effective information systems" (Topi, et al., 2010, p. 402).

The course described in this paper is modeled after the IS 2010.07 course. The five questions, from Ives et al. form the basis for the course's learning objectives. In particular, the course is intended to give both business and IS students an introduction in the development and application of key business intelligence tools and exposes the students to the key issues facing organizations in developing enterprise level information systems. Since the goal of business intelligence systems is to improve decision-making by leveraging data and information to make better decisions. Note the students do not actually develop the BI or analytic solutions as this is beyond the scope of this course.

In the next section, the use of case studies as a pedagogical tool in this course is described. A brief description of the literature on the impact of case studies on student learning is examined.

3. PEDAGOGICAL USE OF THE WORK SYSTEM FRAMEWORK IN CASE STUDY ANALYSIS

Case studies provide students with an "indirect, or vicarious, doing experience" (Fink, 2013, p. 120). "Case studies cut across a range of companies, industries, and situations, providing an exposure far greater than what students are likely to experience otherwise" (Corey, 1996, p. 1). The use of case studies as a pedagogical tool of many information systems (IS) educators is important to help students learn and appreciate the realities of IS-related decision-making situations. The case study allows instructors to *guide* student learning rather than *enforce* learning (Myers & James, 1993). Case studies create opportunities for the instructor "to assist students in gaining critical skills (problem solving, oral and written communication, teamwork, etc.) in a number of different ways through the preparation and presentation of the case study" (Pomykalski, 2013, p. 2). These skills have been shown to be highly valuable to potential employers (Alsop, 2004; Cappel, 2001).

The key aspect in developing these critical skills is in the preparation of the case (both by the student and the instructor). Student preparation can vary widely and guidelines for the preparation of the case by students do exist (Edge, 1982; Ronstandt, 1993; Corey, 1996).

Preparation begins with a close reading of the case to identify key issues, major players, and important facts and scenarios. Pre-case writing assignments, which could be assigned as part of the preparation process, can include a range of activities; from a series of discussion questions focused on the major issues in the case to a formal written analysis (Pomykalski, 2013).

Work System Framework Application in Pre-Case Assignment

In this course, the pre-case assignment is based on the student's understanding and application of the elements in the Work Systems Framework (Alter, 2013).

The work system framework (WSF) "provides a perspective for understanding systems in organization, whether or not those systems use IT extensively" (Alter, 2013, p. 75). The framework's "domain of greatest relevance is IT-reliant work systems" (Alter, 2013, p. 75); business intelligence certainly fits this classification.

Terms	Definition
Work System (WS)	A view of work that is occurring through a purposeful system
Work System Framework (WSF)	A model for organizing an initial understanding of how a particular WS operates and what it accomplishes
Customers	People that receive, use or benefit from products & services that the WS produces
Products & Services	Combination of all the physical things, information, and services that the WS produces for its various customers
Processes & Activities	Includes all of the work practices within the WS, including structured business processes and unstructured, perhaps improved activities
Participants	People who perform the work
Information	Includes the codified and non-codified information used and created as participants perform that work
Technology	Tools that help people work more efficiently
Strategies	Includes the articulated business strategies that the WS is operating under
Environment	Includes the organizational, cultural, competitive, technical, and regulatory environment impacting the WS
Infrastructure	Includes human, informational, and technical resources that support the WS; often shared with other work systems

Table 1: Work System Framework Key Terms

A work system (WS) is defined as "system in which human participants and/or machines perform work using information, technology, and other resources to produce products and/or services for internal or external customers" (Alter, 2006, p. 11); a list of key terms are shown in Table 1 (adapted from Alter, 2006). The framework views IT-reliant systems through nine elements: customers, products & services, processes & activities, participants, information, technology, strategies, environment, and infrastructure.

In order to provide a basis for student understanding and discussion of the cases the WSF of nine elements is introduced and applied to each of the cases in the course. The WSF is used as the basis for student preparation and the pre-case assignment.

The pre-case assignment requires that each student identify and list the "instances" (direct references) of each of the nine elements that are found in the case. The students prepare a listing of the instances, with page numbers, where they find each of the nine elements discussed; the page numbers are for future reference in discussions and post-case analysis.

The benefit of using the WSF of nine elements is that, in the preparation of the case, students can focus on each element separately in identifying these "instances", then in-class discussion can focus on the integration of the elements and their influence in the success (or failure) of the information system.

Discussion

The largest value of teaching with a case is in the discussion. Brookfield (2005) states that "some discussions veer back forth between the analysis of a problem and considerations of how participants might act in response to it" (p. 28-29). Furthermore, two of the four aims of discussion to develop, critical, informed understanding and to help people take informed action align with the case study process. In the context of a case study discussion, this means that the instructor must prepare a set of focused, directed questions that lead to an analysis and meaningful understanding of the issues and complexity in the case. Case study discussion has been shown to lead to enhance both oral and written communication skills of students (Dallimore, Hertenstein, & Platt, 2008) as well as skills in synthesis and integration (Brookfield, 2005).

In general, discussions fail for a variety of reasons. Instructors can minimize the possibility of failure by setting realistic expectations, providing ground rules, modeling good discussion behaviors, and providing well-defined reward systems. Student behavior is also important to successful discussions Barnes, Christensen & Hansen (1986) define positive student behaviors as:

- Participate and listen actively throughout class discussions.
- Contribute ideas, analysis, and personal experiences instead of simply presenting case facts.
- Build on each other's comments and critique and debate different points of view

The discussion, in the current course iteration, begins by examining the key WSF elements that are driving the case. Usually, key elements are identified by the number of instances found in the case so students are asked to identify the key elements in this way.

Discussion then moves to more of an integrative approach in which key elements are linked together, usually by the students through prompting from the instructor. The discussion continues until all integrative components are discussed.

Post-Case Analysis

Finally, a post-case analysis assignment is utilized to finalize the learning experience. From the case study literature, these assignments are in-depth analysis; usually the formal analysis document is used to develop the students' analytical thinking and problem solving skills (Pomykalski, 2013). Rosier (2002) found that through the use of reflective reports as a post-case assignment, "with appropriate guiding questions", improved the value and relevance of the case to students.

Currently, a series of integrative questions are used to elicit the understanding of the students about the case particulars. Starting with the second case, integration, in the form of compare-and-contrast questions are used to show the relationships between the materials in previous cases to the current case. Currently, this is a weakness in the learning process because reflective assignments (Dehler & Welsh, 2014) are seen as a necessary part "learning" of any technical (business) profession.

4. RATIONALE FOR USE OF WORK SYSTEM FRAMEWORK AS ANALYSIS TOOL

The primary reason for using the WST approach is precisely because it incorporates, directly, the "socio-technical" aspects of a system. This contrasts with the view of the "system-as-a-technical-artifact" perspective espoused by many systems analysis textbooks (Whitten & Bentley, 2007; Dennis, Wixom, and Roth, 2009; Hoffer, George & Valacich, 2014; Kendall & Kendall, 2011; Mathiassen, Munk-Madsen, Neilsen & Stage, 2000).

The "socio-technical" view serves two primary purposes: (1) it addresses the final two items business students need to know, as presented in section two, about business intelligence systems and (2) it provides a firmer grounding for business students who often have the "system-as-a-technical-artifact" perspective due to prior coursework and lack of experience with organizational dynamics.

5. FACTORS IN SUCCESSFUL IMPLEMENTATION

The course, taught primarily to junior and senior level business majors, analyzes five cases throughout the semester. The first two cases are focused on the integration of Enterprise Resource Planning (ERP) systems within organizations. These cases highlight unsuccessful implementation efforts where the primary reason for the unsuccessful implementation rests on the "participants" within the system. The last three cases in the course all deal with implementation efforts, largely deemed as successful, of business intelligence tools and techniques.

A predominant number of "instances" in the initial ERP cases discuss the shortcomings of both the participants and the human infrastructure set up for the implementation of the ERP; primarily senior management and front line employees abdicating their responsibility in the development effort (Edwards & Humphreys, 2005; Paper, Tingey, & Hok, 2003). It is easy therefore for the students to understand that this technological solution implementation was derailed by the human elements.

The other ERP case (Zarotsky, Pliskin, & Heart, 2006) contrasts the upgrade for a functional ERP system to the original implementation process. This case illustrates the change in attitude exhibited by upper management; "this upgrade project was perceived by both business and IS management as a pure IS project, requiring minimal involvement of business management" (Zarotsky, Pliskin, & Heart, 2006, p. 18). In addition, this case illustrates the influence of the environment element on the upgrade. The company was forced to undertake the upgrade project, although reluctant to do so due to multiple uncertainties, due to SAP dropping support of the current ERP software version (Zarotsky, Pliskin, & Heart, 2006).

The first of the BI cases describes the benefits derived from the development and use of a data warehouse at Whirlpool Corporation (Haley, Watson, & Goodhue, 2006). This case is a stark contrast to the previous ERP cases. Upper management exhibits a firm commitment to create an integrated infrastructure that allows the participants to be "informated" (Zuboff, 1998); to have their jobs radically changed and expanded by the introduction of the data warehouse. This case is rich in examples of how the project aligns with stated business strategies, positive participant examples, and a well-integrated technical infrastructure focused on problem solving (Haley, Watson, & Goodhue, 2006).

The second BI related case deals with an investigation into the use of data mining (by an Australian insurance company) in order to set automobile policy rates (Yeo & Smith, 2003). This case illustrates the need for capable and knowledgeable participants to perform data mining activities. While the insurance company has a strong technical infrastructure (an existing data warehouse) for analysis purposes, they lack a knowledgeable human infrastructure (participants that understand data mining activities). Outside assistance is used, in the form of a graduate student and her professor, to examine the feasibility of using data mining to rethink pricing strategies. A three step approach (set of processes and activities) is described that led to a new profitable, pricing strategy.

While data mining proves feasible in the creation of the pricing strategy, the primary issue facing the insurance company going forward is the hiring of technical participants to continue these efforts. This case focuses on the processes and activities, participants, and information (in the form of data to create customer clusters and neural networks) necessary to carry out analytics work (Yeo & Smith, 2003).

The final case examined in the course is a description of a mature business intelligence strategy utilized at Norfolk Southern Corporation (Wixom, et al., 2011). Facing a new competitive landscape due to deregulation and the acquisition of Conrail (a service-oriented railroad) Norfolk Southern embarked on a strategy to build data-driven applications to serve customers and minimize previous inefficiencies in operations.

The case discussion focuses on the processes and activities, participants, and technologies that were part of the transition to a customer-facing, data-driven work environment. These technologies, which included a data mart and an operational dashboard, were used to meet multiple corporate objectives designed to transform Norfolk Southern into a competitive, customer responsive railroad. In addition, organizational structures used to support the BI development are also highlighted.

6. CONTRIBUTIONS TO AWARENESS OF INFORMATION SYSTEMS IMPACTS

As Alter (2013) states information systems have been considered to be sociotechnical systems. However, while sociotechnical theory attempts to separate the social systems from the technical system, the WSF views the social and technical as part of a single system. This view is easier for business students to comprehend using the nine elements of the work system framework to guide their initial analysis.

One of the benefits extracted from using this nine element framework is that students see that system implementation issues (both positive and negative) are not largely due to just the technical side but are a blend of the "fit" between the social and technical systems. Students can see that user involvement, knowledge, and training are key elements in the social side that need to be the focus of any new systems development project.

7. FUTURE WORK IN USING THE WORK SYSTEM FRAMEWORK

To date, students have viewed the use of the WSF to analyze cases as both positive and negative.

One of the most significant negatives is the time needed to introduce each of the nine elements. The time utilized to create the base knowledge of the elements has shortened the time available for detailed consideration of the cases; both individually and collectively. One particular option being considered is to create a blended classroom environment where the burden for understanding the nine elements is shifted to the student. However, this still leaves the problem of giving students adequate time to digest and comprehend the nine elements before embarking on the case study analysis.

Another issue that has limited the effectiveness of the WSF elements is the inability of students to see the integration of the nine elements. Without sufficient understanding of the sociotechnical nature of information systems the students do not comprehend the impact of one element on another, for instance, the impact of an insufficient infrastructure for development on how participants perform the relevant processes and activities. One possible method that has been tried once is devoting class time for small group discussion of the elements and their interaction using directed questions.

There are also two changes contemplated for the administration of the case studies. First, a change in the administration of the discussion of the cases to enhance the learning process is under consideration. Dehler (2009) suggests using a discussion focused pedagogic strategy to enhance the learning and critical thinking skills of students. Dehler (2009) emphasizes the need for a mutual student-teacher responsibility for the learning process.

In the future, small groups (3 to 4 students) will be used prior to the whole class discussion. Techniques for facilitating small group discussions, from Barkley (2009), Bean (2011), and Fink (2013) will be considered.

A second change considers a complete rethinking of the post-case assignment. Hibbert (2013), in outlining the work of both Dehler (2009) and Hedberg (2009) suggest the use of learning journals to ask "students to monitor their own learning trajectory in relation to subject and personal and critical goals before, during, and after the execution of a class". Management educators should develop a "pedagogical approach asking students to explicitly identify and articulate their learning" (Dehler & Welsh, 2014, p. 877).

The use of learning logs (Baker, 2003) or reflective reports (Rosier, 2002) have been reported in the IS literature and could lead to deeper learning and help students develop critical thinking skills. A single important caveat to this type of assignment is the requirement of a reduction in the amount of content is a necessary consideration to make time for reflection (Hedberg, 2009).

8. CONCLUSIONS

Once understood, the use of the work system framework has met with generally positive results. Table 2 shows the results of three primary evaluation questions and the average student response. The students ranked the questions on a 1 to 5 scale; with 5 being Excellent and 1 being Poor.

As can be seen from these results the students saw value in using the work system framework. Only two students (out of 68 students) rated the WSF either fair or poor. The students also saw value in the use of the case studies as well. Only five of the 68 had an unfavorable response to the case studies. The students were less impressed with the Alter textbook; however, student comments suggested that they better understood the cases based on the thorough review of the elements and the text was a critical component of that learning.

Question (N=68)	Average Response
Value of the Alter textbook for understanding & learning the course concepts	3.56/5
Value of the work system framework for understanding & learning the course concepts	4.25/5
Value of the case studies for understanding & learning the course concepts	4.00/5

Table 2: Student Course Evaluations

The combination of the use of the WSF and the case studies served to improve the understanding and learning of the students. One student commented that "the course covered a broad range of topics and applied it to real-life situations". Another student believed that a major strength of the course was gaining "a fundamental understanding of information systems in a corporate environment". Finally, one student cited, as a major course strength, "making students realize how important BI implementation is and how involved one must be while the system is being implemented".

From the course evaluation questions and the student comments, the author believes that the students are:

(1) more aware of the influence of information systems on the bottom line of an enterprise

(from all cases), because the students see that implementation failure is costly,
(2) able to see the importance of technical infrastructures (the BI cases), through the application of elements in the WSF,
(3) able to understand the unique economics of information and information systems (the BI cases), through seeing the impact on an organization,
(4) able to see how information enables organizational decision-making (all cases); the decision makers in the cases are "informated", and
(5) able to understand the development, acquisition, and implementation of information systems; the major steps and obstacles in development are shown.

The use of the WSF, in combination with the selected cases, has given students a better understanding of the complexities of information systems, especially BI systems, and why it is crucial for enterprise success to get the implementation correct.

9. ACKNOWLEDGEMENTS

The author would like to thank the three anonymous reviewers for their time and helpful and constructive comments. The revisions suggested have made this paper clearer and more comprehensive.

In addition, the author acknowledges the importance of the questions from the presentation session at ISECON. The most key question asked, and answered in this revision, was, "what would you change?" The question led the author to investigate further the pedagogic literature and has led to a fairly large revision in the final sections of this paper.

10. REFERENCES

Alsop, R. (2004, September 22). How to Get Hired. *The Wall Street Journal*, R8.

Alter, S. (2013). Work System Theory: Overview of Core Concepts, Extensions, and Challenges for the Future. *Jouranl of the Association of Information Systems.* 14(2). 72-121.

Alter, S. (2006). The Work System Method: Connecting People, Processes, and IT for Business Results. Work System Press, Larkspur, CA.

Baker, J. H. (2003). Teaching Tip The Learning Log. *Journal of Information Systems Education*, *14*(1), 11-14.

Barkley, E.F. (2009). Student Engagement Techniques: A Handbook for College Faculty. Jossey-Bass, San Francisco, CA.

Barnes, L. B., Christensen, C. R., & Hansen A. J. (1994). Teaching and the case method: text, cases, and readings. Harvard Business School Publishers, Cambridge, MA.

Bean, J.C. (2011). Engaging Ideas: The Professor's Guide to Integrating Writing, Critical Thinking, and Active Learning in the Classroom. Jossey-Bass, San Francisco, CA.

Brookfield, S.D. (2005). Discussion as a Way of Teaching: Tools and Techniques for Democratic Classrooms (2nd Ed.). Jossey-Bass, San Francisco, CA.

Cappel, J. J. (2001). A Systems Analysis & Design Case: ABC Church. *Journal of Information Systems Education.* 12(4). 233-234.

Corey, E. R. (1996) The use of cases in management education. Harvard Business School Publishers, Cambridge, MA.

Dallimore, E. J., Hertenstein, J. H., & Platt, M. B. (2008). Using discussion pedagogy to enhance oral and written communication skills. *College Teaching*. 56(3). 163-172.

Dehler, G.E. & Welsh, M.A. (2014). Against Spoon-Feeding. For Learning. Reflections on Students' Claims to Knowledge. Journal of Management Education. 38(6). 875-893.

Dehler, G.E. (2009). Prospects and possibilities of critical management education: Critical beings and a pedagogy of action. *Management Learning*. 40. 31-49.

Dennis, A., Wixom, B. H., & Roth, R. M. (2009). Systems analysis & design with UML version 2.0: An object-oriented approach (3rd Ed.). John Wiley & Sons, Inc., New York, NY.

Edge, A. G. (1982) The Guide to Case Analysis and Reporting. Systems Logistics, Inc., Honolulu, HI.

Edwards, H.M. & Humphries, L.P. (2005). Change Management of People & Technology in an ERP Implementation. *Journal of Cases on Information Technology.* 7(4). 143-159.

Fink, L. D. (2013). Creating significant learning expereinces: an integrated approach to designming college courses, revised and updated. Jossey-Bass, San Francisco, CA.

Gartner. (n.d). Business Intelligence, IT Glossary. Retrieved June 5, 2014 from http://www.gartner.com/it-glossary/business-intelligence-bi/.

Haley, B., Watson, H.J., & Goodhue, D.L. (2006). The Benefits of Data Warehousing at Whirlpool. In Khosrow-Pour, M. (Ed.) *Cases on Database Technologies and Applications*. Information Resources Management Association, State College, PA.

Hedberg, P. (2009). Learning through reflective classroom practice: Applications to educate the reflective manager. *Journal of Management Education*. 33. 10-36.

Hoffer, J.A., George, J.F., and Valacich, J.S. (2014). Modern Systems Analysis and Design. Seventh Edition. Pearson Prentice Hall, Upper Saddle River, NJ.

Ives, B., Valacich, J. S., Watson, R. T., Zmud, R. W., Alavi, M., Baskerville, R., Baroudi, J. J., Beath, C., Clark, T., Clemons, E. K., Davis, G., Davis, F., Dennis, A. R., Sawy, O. A., Fedorowicz, J., Galliers, R. D., George, J., Gray, P., Hirschheim, R. Jarvenpaa, S. Jessup, L., Kemerer, C. F., King, J. L., Konsynski, B., Kraemer, K. Luftman, J. N., March, S. T., Markus, M. L., Mason, R. O., McFarlan, F. W., McLean, E. R., Olfman, L., Olson, M. H., Rockart, J., Sanbamurthy, V., Todd, P., Vitale, M., Weber, R., & Whinston, A, B. (2002). What every business student needs to know about information systems, *Communications of the Association for Information Systems*, 9, 30.

Kendall, K. E., & Kendall, J. E. (2011). *Systems analysis and design* (8th Ed.). Pearson Prentice Hall, Upper Saddle River, NJ.

Loshin, D. (2003). Business intelligence: the savvy managers guide, Morgan Kaufmann Publishers, San Francisco.

Myers, C., & James, T. B. (1993). Promoting active learning: strategies for the college classroom*.* Jossey-Bass, San Francisco, CA.

Pomykalski, J. J. (2013). Reinforcing systems analysis and design process learning, *Proceedings of the Information Systems Educators Conference (ISECON 2013)*. November 7-10, San Antonio, TX, USA, EDSIG, 30, 2559.

Ronstandt, R. (1993). The art of case analysis, Lord Publishing, Wayland, MA.

Topi, H., Valacich, J.S., Wright, R.T., Kaiser, K., & Nunamaker, J.F. (2010). IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Infomration Systems, *Communications of the Association for Information Systems*, 26, 18.

Rosier, G. (2002). Using reflective reports to improve the case method. *The Journal of Management Development*. 21(7). 589-597.

Whitten, J.L. & Bentley, L.D. (2007). *Systems Analysis & Design Methods.* (7th Ed.). McGraw-Hill Irwin, New York, NY.

Mathiassen, L., Munk-Madsen, A., Neilsen, P. A., & Stage, J. (2000). *Object oriented analysis & design*. Aalborg, Denmark: Marko Publishing ApS.

Paper, D, Tingey, K.B., & Mok. W. (2003). The Relation Between BPR and ERP Systems: A Failed Project. *Journal of Cases on Informaton Technology.* 5. 45-62.

Yeo, A. C. & Smith, K.A. (2003). Implementing a Data Mining Solution for an Automobile Insurance Company: Reconciling Theoretical Benefits with Practical Considerations. In Khosrow-Pour, M. (Ed.) *Annals of Cases on Information Technology, Volume 5*. Idea group Inc., State College, PA.

Zarotsky, M., Pliskin, N., & Heart, T. (2006). The First ERP Upgrade Project at DSW: Lessons Learned from Disillusion with Simplicity Expectations. *Journal of Cases on Informaton Technology.* 8(4). 13-23.

Zuboff, S. (1998). In the Age of the Smart Machine: The Future of Work and Power. Basic Books, New York, NY.

Wixom, B.H., Watson, H.J., & Warner, T. (2011). Developing an Enerprise Business Intelligence Capability: The Norfolk Southern Journey. *MIS Quarterly Executive.* 10(2). 61-71.

Hibbert, P. (2013). Approaching Reflexivity Through Reflection: Issues for Critical Management Education. *Journal of Management Education.* 37(6). 803-827.

How Students Use Technology to Cheat and What Faculty Can Do About It

Lisa Z. Bain
lbain@ric.edu
Department of Accounting & CIS
Rhode Island College
Providence, RI 02908 US

Abstract

Technology, like most things in life, can provide many benefits to society and improve both the business and academic environments. Technology can also be used in ways that circumvent the educational process and create situations where it is not being used in the appropriate way. College students that use technology to gain access to unauthorized information is a form of academic dishonesty referred to as e-cheating. This can happen both inside and outside of the classroom and takes on many forms. The Internet, in particular, fuels this behavior making it easier, faster, and more convenient then in the past. In addition, electronic devices like smartphones, tablets, and laptops are commonly used across all institutions and will continue to be used for the foreseeable future. Fortunately, faculty can use traditional pedagogical methods for educating, preventing, and reporting academic dishonesty. This paper discusses the common ways technology is being used by students for e-cheating and the actions faculty can take to hopefully reduce academic dishonesty in their courses.

Keywords: e-cheating, academic integrity, academic dishonesty, plagiarism, cheating

1. TECHNOLOGY: THE GOOD AND THE BAD

Technology as a Good Thing

Technology can be a good thing. It has the potential and ability to increase the quality of our lives in many ways, if used correctly and appropriately. From allowing us to video chat with friends and family that are far away to providing lower cost shopping options from the comfort of our homes. There have been countless advances in technology in just the last few years that have made our lives easier and more convenient. Navigating with a Global Positioning System (GPS) is just one of the many examples that, in this case, makes it easier and safer to travel. Cell phones have become almost ubiquitous in the hands of most adults in America with over half of those being smartphones (Smith, 2012). These provide an array of features that include messaging tools, calendars, email, and a variety of apps for both productivity and just plain entertainment. Although some may argue that these particular devices may be a distraction, there are true benefits to most users in terms of faster communication and access to important information. Other more recent innovations include social media, Internet-enabled appliances, and tablet computing. This is just a small sampling of some of the more common and everyday technological devices than can provide many benefits to its users.

Technology can also improve education, providing innovative ways for instructors to teach and creative ways for students to learn. The Internet alone can provide access to simulations and virtual tours that allow educators to expose their students to places and resources otherwise unavailable to many classrooms (Evans, Martin & Poatsy, 2014). Course management software provides a rich online environment that includes tools like assignment submission, test/quiz taking, and discussion board forums. Computer labs enable instructor-led training and hands-on exercises for students. Smart boards can promote collaboration and enhance instruction. Hand-held clickers and in class polling encourage student engagement (Bain & Przybyla, 2009). It appears that technology in the classroom is here to stay for a long time. In fact, many institutions are touting their use of technology as part of their recruitment efforts (Khan & Samuel, 2009). This use of technology is seen as a benefit to both the instructor and the student.

Technology as a Bad Thing

Technology can be a bad thing. It has the potential and ability to increase the ease in which students gain access to information, both inside and outside of the classroom. Many of the same features that allow technology to make our lives better can also be used in ways that are not appropriate in an academic setting. Student course work, assignments, and papers should always represent their own thoughts, ideas, unless properly cited. Most colleges and universities provide students with a code of conduct and/or academic standards that outline academic integrity and examples of academic dishonesty (McCabe, 2005). Unfortunately, students do not always adhere to these policies. Not only are students cheating, but they also report they are cheating a lot more than faculty think (Puccio, 2008). To make matters worse, technology enables cheating in new and creative ways (O'Neil, 2003). The use of technology for academic dishonesty is not a good thing for students or for faculty. Students miss out on the learning process, reducing their education and possible future employment opportunities. Faculty must spend time policing and reporting, lessening the time that could be spent enriching the content of the course or discussing other relevant topics.

Fortunately, there are actions faculty can take to mitigate, and in some cases, eliminate the use of technology for academic dishonesty. Cheating has been a problem on college campuses before the wide spread use of technology. Some of the same techniques that have prevented it before are still usable, even with "high-tech" cheaters (O'Neil, 2003, p. 4). This paper defines e-cheating, identifies the common types of e-cheating by college students in traditional courses, and provides a list of actions faculty can take to address the issue.

2. E-CHEATING DEFINED

The term e-cheating, or electronic cheating, is not defined in Dictionary.com or the online version of the Merriam-Webster dictionary. However, it is commonly used to refer to the type of academic dishonesty that utilizes some type of technology to electronically copy or use material from an unauthorized source or a source that was not cited. This can include a simple copy and paste from the Internet or some other type of electronic media (Jones, Reid & Bartlett, 2008). It is basically a form of plagiarism, where someone else's work or idea is presented as one's own (ICAI, 2014). The terms online or digital plagiarism may more accurately describe this specific type of plagiarism but e-cheating tends to encompass other acts of cheating as well (Sterngold, 2004). E-cheating may also be called digital cheating and can be as broad as meaning any type of cheating using computer technology (Rogers, 2006). Cyber cheating is another term some times used in this context but this may be confused with acts of infidelity through the Internet or social media so it will not be used in this paper. Regardless of the name or terminology used, e-cheating involves some type of academic dishonesty where computer technology is involved.

3. E-CHEATING VERSUS TRADITIONAL CHEATING

Since e-cheating requires the use of technology. Therefore, it eliminates traditional types of cheating. McCabe (2005) surveyed over 80,000 college students asking questions specific to tests and examinations, primarily related to traditional cheating behaviors. These included items like learning what is on a test/exam from someone who has already taken it, using false excuse to delay taking test/exam, copying from another student on a test/exam with and without his/her knowledge, helping someone else cheat on test/exam, and using unauthorized crib/cheat notes helping someone else cheat on a test/exam (McCabe, 2005).

The focus of this paper is only the type of cheating that uses technology to give a student an advantage that would otherwise not be available to them. Technology, especially the Internet, provides many advantages to people and businesses but it also increases the "opportunities" for cheating (Bracey, 2005, p. 413). In addition, these new opportunities are often easier, faster, and more convenient then traditional cheating. Technology has advanced significantly in the past decade and continues to provide end users with new and innovative hardware devices and software applications on a regular basis. The same holds true for businesses and information systems that provide better decision making capabilities and methods for increasing competitive advantage (Rainer, Prince, & Cegielski, 2014). With this in mind, it would logically lead one to think that e-cheating has also advanced to creative and new levels. However, that is not necessarily the case. After researching and gathering information on the common types of e-cheating, it tends to occur inside and outside of the classroom in very common ways. The next section describes the common types of e-cheating, providing a summary at the end.

4. TYPES OF E-CHEATING

E-Cheating Inside the Classroom

The first type of e-cheating happens inside of the classroom where students use electronic devices to access unauthorized information and use unauthorized electronic devices in other ways, primarily during exams. Students use electronic devices, such as laptops or smartphones, to access the Internet (Jones, Reid & Bartlett, 2008). This provides them with an unending source of materials, examples, diagrams, and information. It even allows them to access items on any course management system, like Blackboard. Instructors vary when it comes to the content provided in these systems but many do provide study guides, presentation slides, and class handouts. Along with access to the Internet, comes access to any web-based email system where students can send themselves notes and any other course material. Smartphones, just like other electronic devices, can also access email and be used to view unauthorized information (O'Neil, 2003). With the increased popularity of e-texts, this also opens up the entire textbook to students during exams if they have access to an electronic device. In addition, smartphones have the capability to store and display almost any type of electronic document. Using this small device is no different than using a laptop or tablet, other than its small and somewhat discreet size. Technology enables more high-tech approaches with devices like transmitters, blue tooth, or walkie-talkies but these are not common and require equipment not readily available to most college students (O'Neil, 2003). Therefore, the majority of college students rely on electronic devices they already own and know how to use.

The second main behavior inside the classroom is using electronic devices in other ways, again during exams. Students use the text messaging capabilities of both smartphones and IM-enables calculators to send questions and receive answers from third parties, including people far away (Jones, Reid & Bartlett, 2008). Technology allows this to happen very quickly and easily. Wireless earphones and microphones can also be used to communicate with other students (Jones, Reid & Bartlett, 2008). Of course each of these requires help on the outside but it has happened and fortunately in some cases students have been caught (USA Today, 2003). In addition to smartphones, other electronic devices may be used during exams. Programmable calculators have become quite sophisticated and provide many new features. These can be programmed to store text and pictures, in addition to the standard formulas (Jones, Reid & Bartlett, 2008). Other somewhat creative behaviors include the use of MP3 and smartphone cameras. Forward thinking students can record audio files for MP3 players and use these as well during exams (Jones, Reid & Bartlett, 2008). Again, the pictures from the camera must be sent via text messaging or email to another party making the process a bit more complicated then some of the other methods but certainly doable by students.

Therefore, e-cheating inside the classroom requires the use of an electronic device to access unauthorized information from a variety of sources. The methods vary but the use of a device remains consistent and a necessary factor in order for the academic dishonesty to occur.

Summary of E-Cheating (Inside)

- Use of electronic device to access unauthorized information
  - Internet
  - E-mail
  - E-Texts
  - Electronic documents/files
- Use of text messaging (including IM-enabled calculators)
- Use of programmable calculator
- Use of cameras
- Use of MP3 players
- Use of wireless earphones and microphones

E-Cheating Outside the Classroom
The second type of e-cheating happens outside of the classroom where students tend to copy, purchase and use information from the Internet. The Internet provides a variety of materials that are easily accessible to anyone with access and the basic skills to perform a search. The term google is now included in the dictionary as a verb for searching information on the Internet. Since the results of these searches are electronic, students can quickly copy and paste the information directly into assignments, papers, and other documents (Szabo & Underwood, 2004; Sterngold, 2004). The copy and paste action is not limited to the Internet with students having access to other types of electronic media as well. Online textbooks (e-texts), articles in PDF format from libraries, email, and course management systems are just a few examples. One rather new item is Amazon's "Search Inside the Book" feature, which allows searching and viewing information from books directly online (Sterngold, 2004). Of course, students have the option to cite all of this information if they so chose but, as instructors, we know this does not always happen. In fact, to make it even easier for students there are many web sites dedicated to cheating like www.schoolsucks.com, www.cheathouse.com, and www.123helpme.com (Jones, Reid & Bartlett, 2008). Some of these sites provide information for free while other charge a nominal fee for completed papers. This leads to the discussion of purchasing items from the Internet or via email.

Although this has been done in the past without the use of technology, the Internet and email make it easier, faster, and anonymous for students to purchase materials to use and submit as their own course work. Students can buy all sorts of documents, including papers, online (Sterngold, 2004). Various sites also sell solution manuals, test banks, and instructor manuals. Students have been caught using exact words from instructor manuals in their course work (Puccio, 2008). To make it even easier, YouTube provides videos that direct students to email addresses and websites for such items (Buy Test Banks, 2014). The amount of material and information that is easily and quickly available online will continue to increase and will be a constant source of help to students. Students are comfortable with the online world and the use of electronic resources not only for schoolwork but also for many other aspects of their lives, like social media. They use this environment on a daily basis for social interaction, product reviews, directions, weather reports, and shopping. Students even use Facebook for cheating and it has become another online tool that can be used to share information, like exam questions (Bi, 2013). The overall use of technology in many different ways is just normality today for most students.

E-cheating outside of the classroom requires the use of the Internet and the many options available online for accessing a variety of material. This environment provides the ability to easily copy and paste information, purchase ready-made materials, and use a variety of resources to obtain unauthorized help on assignments. These methods are more difficult for instructors to see since they are happening outside of the classroom and away from the normal face-to-face time of a traditional class setting.

Summary of E-Cheating (Outside)

- Copy information from the Internet
- Copy information from electronic media
- Purchase papers/documents from the Internet
- Purchase solution/instructor manuals from the Internet or by email
- Purchase test banks from the Internet or by email
- Use of Cheating Web Sites
- Use of Amazon's Search Inside the Book
- Use of Social Media to share information

5. FACULTY ACTIONS FOR E-CHEATING

Technology has enabled students with a plethora of options for academic dishonesty and faculty need to be aware of the many actions they can take to help educate students about academic integrity, assist in preventing academic dishonesty and report incidents when they happen.

Educating Students about E-Cheating
One of the first actions faculty can take to reduce e-cheating in their courses is to *educate* their students about academic integrity. This includes maintaining a specific academic integrity policy and discussing this information with the students (Lang, 2013). There are many ways to do this and a variety of options available for faculty to incorporate into their particular teaching styles. First, provide "explicit" descriptions of plagiarism and academic

dishonest to students (Puccio, 2008, p. 20). This can be done as a separate document or as part of the course syllabus but regardless the information should be discussed directly with students in class, preferably at the beginning of the term and again before each high-stake assignment. However, including the information on course syllabi is strongly suggested by many researchers in this area (Baldwin, 2001; Novotney, 2011). Included in this information should also be a description of the "harm" academic integrity does to students themselves (Davis, 2009, p. 345). The information can also be delivered in the form of a quiz or survey, for a grade or not. Many students enjoy this type of format and it can be used as a way to reinforce the topic and engage the students. For a more modern approach, the International Center for Academic Integrity provides a series of short videos depicting examples of what constitutes academic dishonesty and why it is so important (ICAI, 2014). Students make and star in most of the videos and use common examples of both traditional and e-cheating behaviors. These can help clarify expectations of students. At the University of California San Diego, a student group called AIM (Academic Integrity Matters) promotes academic integrity through awards and a petition. This petition focuses on three items, educating students, reducing opportunities and reporting cheating (AIM, 2014). Educating students should also include the reporting process of incidents and the resulting consequences, which adheres to one of the ten principles of academic integrity (McCabe & Pavela, 2004). Even with the advances in technology, faculty can reduce e-cheating by adopting the non-technical practice of educating students early and often.

Preventing Students from E-Cheating

The second action faculty can take is to *prevent* e-cheating by incorporating certain techniques into their course and course work. Inside the classroom, the most obvious action is to prohibit all electronic devices during exams (Jones, Reid & Bartlett, 2008). This addresses each of the most common behaviors students use in the classroom to cheat. Of course, students must be instructed of this at the beginning of the term as well as immediately prior to each exam. Faculty must also actively proctor during the exam to ensure students conform.

The techniques for preventing e-cheating outside of the classroom are very different then the ones for inside. The first item that should be addressed is student paper writing. There appears to be several approaches to prevent problems in this area. First is the use of anti-plagiarism software tools like TurnItIn and SafeAssign (O'Neil, 2003). Both products provide an environment where students submit their papers to the system and then faculty can use a variety of features to grade the papers plus check for the use of similar text from the Internet or other papers. One of these features is an Originality Report in TurnItIn that shows a percentage for the amount of text in the paper that is similar to other works (TurnItIn, 2014). If used properly, this feature alone could provide significant deterrence of copy/paste by students. The downside to these tools is that the institution must provide the tools, student must have access, and faculty must be trained on their use.

The second approach for *preventing* e-cheating on student papers focuses on the assignment itself in terms of its design and implementation. Puccio (2008) recommends that faculty use unique assignments. This reduces the chances that similar papers will be available online, free or otherwise. Faculty can also use this opportunity to create more creativity in their assessments, helping to further engage the student (McCabe, 2005). The more specific the paper is to the course or the discussions in the classroom, the less likely it would be duplicated at another institution. Other researches recommend dividing the paper into smaller parts that are submitted over a designated time period and putting restrictions on the source material, like copies or a designated list (Baldwin, 2001; Puccio, 2008; Sterngold, 2004). These and any other additional parameters help to limit the type of mass online copying so easily available outside of class. Student papers can be very similar to but can also be very different than other types of assignments and assessments.

Faculty can reduce e-cheating by using techniques for assignments as well. Lang (2013) suggests offering more frequent, low-stake assignments that reduce the pressure on students. This technique again makes it more challenging to use someone else's information or copy text from the Internet. Faculty can also apply some of the same principles to assignments that are recommended for papers by making the assignments meaningful and ensure they have clear instructions (Baldwin, 2001). The more direction students are given, the less likely they will turn to other resources for assistance. (AIM, 2014) also suggests that faculty not use the questions in the textbook for homework assignments. This makes sense knowing that students can get access to solution and instructor manuals. These prevention techniques allow instructors the ability to discourage e-cheating before it becomes an issue, regardless of the type of technology that exists today or the future. Unfortunately, students will continue to cheat even after being educated by faculty and in courses that use a variety of prevention tactics. In these cases, it is very important for faculty to take some type of action and report the incident.

Reporting Students for E-Cheating
The third action faculty can take to reduce e-cheating is to *report* incidents and deliver the appropriate consequences. Students know that cheating is happening inside and outside of the classroom. They also know that much of this goes unnoticed and does not get reported (AIM, 2014). To make matters worse, faculty can be reluctant to report due to the lack of evidence or reporting requirements (McCabe, 2005). The side effect for both of these is an environment where students feel that cheating is not taken seriously. This can also encourage students to participate in this behavior knowing the chances of being caught and/or reported are low. Therefore, it is vital that faculty respond to incidents of academic integrity in an appropriate way. Puccio (2008) recommends that faculty should be involved and all offenses should be reported in some way. The penalties for e-cheating should vary by the type and severity of the offense. However, all students that are rightfully caught engaging in this type of behavior should receive some type of penalty (Lange, 2013). Faculty, of course, must follow the policies at their respective institutions. The key here is to ensure that the faculty member takes some type of action, regardless of severity. Students need to know that the instructor and the institution take academic dishonesty seriously. The hope is that students see faculty as a role model and will model their behavior appropriately.

Summary of Faculty Actions
Educating

- Maintain an academic integrity policy
- Describe plagiarism and academic dishonesty
- Set clear expectations
- Explain academic dishonesty consequences

Preventing

- Prohibit use of electronic devices during exam
- Use anti-plagiarism software tools to limit access to electronic information
- Create unique, creative, and course-specific assignments
- Divide papers into smaller components
- Apply requirements to sources for papers
- Assign more frequent, low-stake assignments
- Provide meaningful and clear instructions on assignments
- Use assignments that are not included in the textbook

Reporting

- Respond to all incidents of academic dishonesty
- Penalize students with appropriate consequences

6. CONCLUSIONS

Technology provides many benefits to society and especially to college students, by providing ease of access to a variety of resources. Along with this access comes the temptation to use this information in ways that circumvent the learning process. E-cheating is using this technology to commit academic dishonesty by cheating or plagiarizing, regardless of when or where it is happening. For traditional college courses, technology can be used in the classroom to provide students access to unauthorized resources and information. Technology can also be used outside the classroom to find answers to assignments, text for papers, and completely written papers. The most common types of e-cheating inside the classroom focus primarily on using electronic devices to access and/or receive unauthorized information during exams. There are minor occurrences of other high-tech e-cheating but these are rarer and require sophisticated equipment. Outside the classroom, the Internet rules as the place to find information for all types of course, free or for a minor fee.

In a world with such amazing technology and an Internet as sophisticated as it has become, student methods for e-cheating are not so elusive for faculty to address. Faculty can simply use traditional pedagogical methods for educating, preventing, and reporting academic dishonesty. Students cheat less when they are better educated and consistently reminded about the nuances of academic integrity. Faculty can prevent e-cheating by using low-tech pedagogical techniques like banning electronic devices during exams and designing assignments that promote academic integrity. Lastly, faculty members need to report incidents of academic dishonesty and apply appropriate consequences to students. These actions of educating, preventing and reporting require little, if any, technical skills on the part of the faculty. With the exception of software tools that automatically detect plagiarism, traditional and low-tech methods of pedagogy can address the majority of e-cheating.

7. REFERENCES

AIM (2014). Academic Integrity Matters Petition. Accessed May 29, 2014. Retrieved from http://ucsdaim.org/petition/.

Bi, F. (2013). Better Data Can Help Colleges Fight Cheating. *Chronicle of Higher Education*. September 9, 2013. Accessed May 29, 2014. Retrieved from http://chronicle.com/article/Better-Data-Can-Help-Colleges/141485/.

Bain, L. Z., & Przybyla, J. (2009). The Impact of Student Response Systems on Student Behavior and Performance in a Management Information Systems Course. Issues in Information Systems Journal, (X) 1, 1-12.

Baldwin, A, (May 2001) Practice Plagiarism Prevention, *The Teaching Professor*, 15 (5).

Bracey, G. W. (2005). Research: A Nation of Cheats. Phi Delta Kappa Inc. 86(5), 412-413. Gale Group, Farmington Hills, Michigan.

Buy Test Banks (2014). Buy Test Banks, Solution Manuals & Instructor Manuals. YouTube Video. Accessed May 28, 2014. Retrieved from http://www.youtube.com/watch?v=Brw-UlDi5-I.

Davis, B. G. (2009). Tools for Teaching, 2nd Edition. John Wiley & Sons, Inc. San Francisco, CA.

Evans, A., Martin, K., & Poatsy, M. A. (2014). Technology in Action, Complete. 10th Edition. Pearson. Boston, MA.

ICAI (2014). International Center for Academic Integrity (ICAI) Multimedia Resources. Retrieved from: http://www.academicintegrity.org/icai/resources-3.php.

Jones, K. O., Reid, J., & Bartlett, R. (2008). "Cyber Cheating in an Information Technology Age" .In: R. COMAS, J. SUREDA (coords.). "Academic Cyberplagiarism" [online dossier]. Digithum. No. 10. UOC. [Accessed: 5/9/2014]. Retrieved from http://www.uoc.edu/digithum/10/dt/eng/jones_reid_bartlett.pdf

Khan, Z. & Samuel, S. D. (2009). E-cheating, online sources and technologies: a critical review of existing literature, 9th Global Conference of Business and Cconomics, Cambridge University Press, Cambridge, pp 1-15.

Lang, J. M. (2013). Cheating Lessons, Part 2. *The Chronicle of Higher Education*. Accessed May 29, 2014. Retrieved from http://chronicle.com/article/Cheating-Lessons-Part-2/140113/.

McCabe, D. L. & Pavela, G. (2004). Ten [Updated] Principles of Academic Integrity. *Change*. May/June 36(3).

McCabe, D. L. (2005). Cheating among college and university students: A North American perspective. *International Journal for Educational Integrity*, 1(1), 1-11.

Novotney, A. (2011). Beat the cheat. Monitor on Psychology, 42(6). Retrieved from http://www.apa.org/monitor/2011/06/cheat.aspx.

O'Neil (2003). Technology and Academic Integrity, Cheating Goes Cyber. Information Systems Education Journal, 1(3). Puccio, P. (2008). Encouraging Academic Honesty through Active Plagiarism Instruction and Prevention. Essays from E-xcellence in Teaching (Vol 7). Retrieved from the Society for the Teaching of Psychology Web site: http://teachpsych.org/resources/e-books/eit2007/eit2007.php.

Rainer, R. K., Prince, B., & Cegielski, C. (2014). Introduction to Information Systems, Supporting and Transforming Business 5th

Edition. Hoboken, NJ; John Wiley & Sons, Inc.

Rogers, C. F. (2006). Faculty Perceptions About E-Cheating During Online Testing. Journal of Computing Sciences in Colleges, 22(2), 206-212.

Smith, A. (2012). The Best (and Worst) of Mobile Connectivity. Retrieved from http://www.pewinternet.org/2012/11/30/the-best-and-worst-of-mobile-connectivity/

Sterngold, A. (2012). Confronting Plagiarism: How Conventional Teaching Invites Cyber-Cheating. Change, 36(3), 16-21.

Szabo, A., & Underword, J. (2004). Cybercheats: Is Information and Communication Technology Fuelling Academic Dishonesty? Active Learning in Higher Education, 2004(5), 180-199.

TurnItIn (2014). Reduce Plagiarism. Accessed May 29, 2014. Retrieved from http://turnitin.com/en_us/features/originalitycheck.

USA Today (2003). University students admit to using cell phones to cheat. USA Today. Posted 1/31/2003. Retrieved May 28, 2014 from http://usatoday30.usatoday.com/tech/news/2003-01-31-phone-cheats_x.

Editor's Note:

This paper was selected for inclusion in the journal as an ISECON 2014 Meritorious Paper. The acceptance rate is typically 15% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2014.

Internet Addiction Risk in the Academic Environment

William F. Ellis
fellis@maine.edu

Brenda McAleer
mcaleer@maine.edu

Joseph S. Szakas
szakas@maine.edu

University of Maine at Augusta
Augusta, ME 04330

Abstract

The Internet's effect on society is growing exponentially. One only has to look at the growth of e-commerce, social media, wireless data access, and mobile devices to see how communication is changing. The need and desire for the Internet, especially in such disciplines as Computer Science or Computer Information Systems, pose a unique risk for dependencies to arise. These dependencies are identified under the broad terms of Internet Addiction and/or Internet Abuse.

Keywords: Internet Addiction, Internet Abuse, risk factors, online education

1. INTRODUCTION

Faculty are often on the front line to identify behavior issues with students. To make this task even more difficult, faculty find themselves in a changing educational environment. Online or asynchronous course offerings and student demand for these modalities continue to rise, which in contrast with face to face instruction, provide a myriad of communication challenges. Many faculty comment on the demand of immediacy by online students to respond to e-mails or postings on learning management systems, such as Blackboard™. For all of the positive effects of online education (flexibility, access, availability, etc.) faculty's remote interaction with these online students has a negative effect in that it hinders their ability to identify issues, whether they are social or behavioral. In this environment the Internet is the pervasive medium for communication and services of today's society. With an extreme reliance on any tool there emerges a possibility for an abnormal dependence to occur. The need and desire for the Internet, especially in such disciplines as Computer Science, or Computer Information Systems (regarded from this point as CSIT programs), pose a greater risk for these dependencies to arise.

This paper will discuss an overview of Internet Addiction and Internet Abuse; identify resources and courses within the CSIT discipline and supporting courses where awareness / identification of this type of disorder is possible; identify current opportunities for treatments; and discuss preparation for students into the workforce, where strict Internet use policies are the norm and not the exception. What are the current steps to help to identify students with possible risks to Internet Addiction? What are the opportunities within the CSIT program to address/inform the issues of Internet Addiction? And finally what can we do to reduce Internet Abuse for future CSIT graduates. The authors will provide resources for faculty to help identify those at risk, take steps to support/intervene, and prepare students for Internet Abuse challenges that may face them in the workplace, either as an employee suffering from Internet dependence to being an employer who has to compose policies to restrict and control Internet Abuse in the workplace. The focus is on the CSIT online students, though the authors feel that this can be extended to any online student, or online students who are classified as either traditional or non-traditional.

2. BACKGROUND

In a matter of very few years, the Internet has changed forever the way we do business, the way we learn, and the way we communicate. The Internet has become the conduit of information for millions of people, at home, at school, and at work. Since 1989, the online population worldwide has grown from 500,000 to 2.9 billion now (Morahan-Martin, 2005). In 1995, there were 16 million users worldwide; the Internet reached its 1 billion users mark in 2005; its second billion mark by 2010, and the third billion will be reached by the end of 2014. Eighty six percent of the US population is connected to the Internet (http://www.Internetlivestats.com). The Internet can be a very productive tool. Many people work over the Internet, study on the Internet, and keep in touch with distant friends over the Internet. Whether it be online shopping, education or communication, the Internet (and we can say broadband Internet) is the key component allowing our society to function and interact in a very limited physical manner. It goes without saying that the Internet has become an essential tool for today's students. Because teaching is fundamentally about communication and knowledge transfer, when communication methods are changed, so is education. The Internet's effect on academic programs has already been seen in the rise of online education. Between fall 2002 and fall 2011, institutions of higher education saw a decrease in total enrollment of -0.1 percent; however, enrollment in online courses during that period increased on average by 17.5 percent per year. Online enrollment as a percentage of total enrollment in fall 2011 stood at 32 percent (Allen & Seaman, 2013), meaning nearly 1 out of 3 courses in the U.S is now offered online or at a distance. From an educational viewpoint, the number of students taking at least one online course in the past year increased by over 570,000 to a new total of 6.7 million. Since keeping statistics on online courses, the lowest percent of growth in numbers of online courses in one year is 9.3 percent (http://www.onlinelearningsurvey.com/reports/changingcourse.pdf). Here at the authors' institution, the online rate by fall 2013 is approximately 31 percent, reflecting a 10 percent growth each academic year since fall 2001. Internet has also become an important part of student life (Chou, Condron, & Belland, 2005). Besides the use for education, students appreciate interactivity, ease of use, availability, and breadth of information accessed online (Chou, 2001) In addition, the Internet provides students and all users in general a place to relax, escape pressures, and seek excitement (Morahan-Martin & Schumaker, 2000). More and more courses are being offered asynchronously, and not just viewed on computer, but on smartphones, tablets, and heard through mp3 players which provides an advantage for students, especially non-traditional students (defined as undergraduate students age 25 and higher), who are a growing sector of undergraduate enrollments today. There are enough studies to show the academic effectiveness of online education and the judgment of the worth of online education is not the focus of this paper.

As educators, the faculty's interactions with online students, as well as the students' peer-to-peer support, is fundamentally changing. So the question that becomes apparent is, what effect does this increased movement online in education have specifically for CSIT programs? There are many factors that indicate a high risk for CSIT students. These disciplines already requires high exposure to the Internet and all of its opportunities as part of normal academic use/need/studies. The computer and the Internet are an essential component for study in these disciplines. The Internet is the subject as well as the medium for CSIT education. The typical CSIT student is in the demographic age bracket most likely to be at a greater risk of Internet related addictions. As early as 1997, researcher Brenner suggested that college students' greater accessibility and usage of the Internet may increase their vulnerability to Internet Abuse (Velezmoro, Lacefield, & Roberti, 2010).

It is not uncommon to find students in the CSIT field who are passionate about computers and the world of the Internet; many of these students do not remember a time when broadband Internet service was not a given in a household; rather they see the Internet as a fundamental necessity. Faculty teaching CSIT courses can be seen as part of the problem as we insist on internalizing computer knowledge and require long hours of interaction with the computers and often via the Internet. Time online is an important factor or index for determining Internet Addiction (Chou et al, 2005). Knowing that intensive computer and Internet use has the potential to magnify addiction problems with students, faculty need to be part of the solution. So the fundamental questions become: how do faculty maintain the role of a front line identifier of risks for Internet Addiction? And how can we address/inform students of the issues of Internet Addiction? And, how do we as faculty help students transition into a world of computer monitoring in the workplace where Internet Abuse is a major concern for employers?

3. WHAT IS INTERNET ADDICTION?

The explosive growth of the Internet over the past decade has almost certainly changed the profile of the computer addict (Chou et al, 2005), and therefore the definition of computer Addiction. Defining Internet Addiction is also difficult even for those who have researched extensively in the field. One definition is that Internet Addiction is a "psychological dependence on the Internet" (Chou et al, 2005). Internet Addiction has yet to be formally recognized as a mental health illness (Felix, 2014), Although Internet Addiction appears to be a common disorder that merits inclusion in Diagnostic and Statistical Manual of Mental Disorders (DSM-V), the closest to formal recognition is that of Internet Gaming Disorder in the fifth edition of the DSM-V (Block, 2008). Internet Gaming Disorder was identified in Section III as a condition warranting more clinical research and experience before it might be considered for inclusion in the main book as a formal disorder (www.dsm5.org).

Young concludes that, although the Internet itself may not be addictive, specific applications (e.g. computer games) appear to play a roles in the development of pathological Internet use (Chou et al, 2005). The confusion as to whether or not Internet Addiction should be a unique disorder stems mainly from the fact that it is co-morbid (a condition that exist alongside another condition) with other disorders, such as panic disorder, or obsessive-compulsion disorder (Felix, 2014).

Chang and Man Law (2008) agree that there are four sub-dimensions to the term Internet Addiction. These four are: Compulsive Internet use involving excessive time spent online and failure to control it; withdrawal symptoms when being restricted from Internet use; using the Internet for social comfort; and negative social, academic or work consequences related to Internet use (Gnisci, Perugini, Pedone, & DiConza, 2011). A few studies have compared Internet Addiction and users' social-psychological or personality variables, such as sensation seeking, pleasure experience, use and gratification, loneliness and depression (Chou et al, 2005). The instant access and immediate gratification of the Internet and the services it supports can be addicting or at least a magnifying force to already established addiction disorders, such as online gambling, pornography and substance abuse. (http://whichmbtitype.wordpress.com/2013/06/17/which-mbti-type-is-most-addiction-prone).

In the educational arena, the effects of Internet Addiction can be seen as students get poor grades or are placed on academic probation because they spend too much time on the Internet rather than on their studies (Chou et al, 2005). Problems caused by the Internet: five use related problems such as failure to manage time, missed sleep, and missed meals, suggesting that such patterns are the norm. Some reported more serious problems because of Internet use, trouble with employers or social isolation except for Internet friends, troubles similar to those found with other addictions (Sherer, 1997).

Though most studies that have been conducted refer to students in Asia, it is estimated that Internet Addiction affects at least one in eight Americans (http://netAddiction.com/faqs). Yet, only 1 percent of college-level introductory and abnormal psychology books mentioned Internet Addiction in 2008 (Mossbarger, 2008). CSIT student demographic information reveals that more males are CSIT majors than females, so the fact that males are more likely than females to become Internet Addicts makes the risk even greater for this demographic student body (Sherer, 1997). Chou and Hsiao's study of 910 students in 2000 found that 54 or 6 percent were Internet addicts who spent 20-25 hours per week online, almost triple the number of hours of 856 non addicts spent online (Chou et al, 2005). Online activities or applications (such as chat rooms or online games) are also an important factor in determining Internet Addiction. (Young, 1998).

4. IDENTIFICATION

If faculty are made aware of warning signs and symptoms, they can more readily identify those students who might benefit from intervention through student counseling services at the institution to help identify high risk students/users (Chou, 2001). Young developed an "Internet Addiction diagnosis questionnaire" available free and online. In this questionnaire, respondents are asked to rate themselves in 20 areas such as: How often do your grades suffer because of time spent on the internet? How often has your job performance suffered because of your time on the Internet? How often do you lose sleep because of late night log-ins? The total score, based on answers that range from rarely to sometimes to always, indicate if the respondent has a potential problem. Those who score between 80-100, meaning most of their answers were most often or always, are told: "Your Internet usage is causing significant problems in your life. You should evaluate the impact of the Internet on your life and address the problems directly caused by your Internet usage" (http://netAddiction.com/Internet-Addiction-test).

5. TECHNIQUES

So where do we address this issue of Internet Addiction within the curriculum? In the IS2010 curriculum guidelines (Topi, Valacich, Wright, Kaiser, Nunamaker, Sipior, & de Vreede, 2010), there is no mention of dealing with the issues of Internet Addiction/ Internet Abuse. It should also be noted that, with many degree programs limited by total program credits with major and general education requirements, that adding another specific course to deal with this issue does not make much sense. So, therefore, the quest is to find areas within existing CSIT curriculum that allow for embedding this topic and providing reinforcement throughout the degree program. Fortunately, there are logical places to do that embedding and reinforcement. The first course is the introductory course IS2010.1 (Foundations of Information Systems) which can address the topics of both Internet Addiction and Internet Abuse when discussions deal with the broader issues of computer ethics. Although not in the IS 2010 curriculum, many programs require a computer ethics course, which would serve well for supporting discussion on these topics.

An assignment that could be used in CSIT courses is one that is often used with students taking a Behavior Modification course in the Psychology department. Students are required to identify a specific behavior that they would like to change. Frequently students identify spending too much time on the computer as the excessive behavior that they would like to change. Often in the treatment of addictive behaviors the goal is abstinence (Joosten, DeWeert-Van Oene, Sensky, Van de Staak, & De Jong, 2011). This obviously is not a realistic goal for students taking online courses or future computer programmers. When students are pressed to be more specific about reducing their computer usage, many will identify the amount of time spent on social media websites. The focus of the exercise is to make students aware of the extent of their behavior and learn strategies to effectively monitor and control their actions.

As a first step, students are required to identify antecedents and consequences of the behavior. Then they are asked to monitor the behavior for two weeks. Most will be amazed by the actual frequency or duration of their behavior. Often the mere recording of the behavior will have an impact, typically reducing it. After a two-week monitoring phase, students implement a self-designed intervention plan focusing on altering stimuli that prompt the behavior and changing the contingencies for excessive time on the internet. Students continue to record their behavior allowing for an empirical evaluation of the effectiveness of the intervention. Most students will report some reduction in their time spent online as well as a greater sense of control of their behavior.

Many CSIT programs are seeing value in external internships as a way to give students some real world experience before graduation. Internet Addiction often reveals itself as Internet Abuse. Internet Abuse is defined broadly as inappropriate web surfing during work hours. It is a problem among organizations as it results in low job productivity, security risks, increased turnover rates, and potential for legal liability (Young, 2012). In the workplace, student presentations of internship experiences many times lead to discussion of Internet Abuse affecting the workplace in specific ways. Another common avenue is to find business courses that can facilitate the education on Internet Addiction. First is an Organizational Behavior course. This course has been useful toward CSIT (McAleer & Szakas, 2005) as it not only reminds the CSIT student of the organizational structures that they must navigate in their career ladder, but also may be a point for possible discussion on the topic of Internet Addiction/Abuse. In organizational behavior courses, a typical task is for students to take a Myers/Briggs Personality tests (MBTI). The test is available online at http://www.humanmetrics.com/cgi-win/ jtypes2.asp. This is important for students because it reminds us of the variety of techniques to interact/deal with employees. Additionally, and perhaps more importantly, there are specific personality types that in fact lend themselves to have a higher risk of Internet Abuse. INTP (introvert, intuitive, think, perceiving) students are more likely to be attracted to disciplines which require an ability to focus in depth to solve problems, including CSIT degree programs. They are quiet and are the ones most likely to lose social interaction due to a computer addiction (http://whichmbtitype.wordpress.com/2013/06/ 17/which-mbti-type-is-most-addiction-prone).

The second business course that has the potential to assist in the discussion of internet Addiction/Internet Abuse is the Principles of Management course. Studies have shown that the Internet's impact on the work environment has a potential risk in terms of Internet Abuse. Companies that do not conduct policy training or monitor internal messages can be putting themselves at risk. Some estimates reveal that computer crime may cost over $50 billion per year. Sixty percent of security breaches occur within the company --- behind the firewall. Non-work related Internet surfing results in up to a 40 percent loss in productivity in American businesses. From a management information systems perspective, employees who use the Internet for other than job tasks place a significant drain on network resources and impair the responsiveness of the system for job-related activities (Young, 2010). One of the key pieces of Internet Abuse is in dealing with making sound policies in an organizational setting that are both understandable and enforceable. The topics of Internet Addiction and Internet Abuse should be discussed in a management course in chapters dealing with workforce productivity, managing in today's worlds, organizational design, foundations of control, and work teams (McAleer & Szakas, 2004).

6. CONCLUSION

The impact of online education for CSIT students warrants additional investigation into abnormal dependencies as the understanding of Internet Addiction continues to evolve. As faculty continue to be the key interface between students and the institution, strategies to assist in identifying students at risk of Internet Addiction is essential. With the increasing trends of online education, the importance of preparing now for this issue cannot be understated. This will help in student academic success as well as support a healthy IT workforce as it will assist in dealing with the growing problem of Internet Abuse in the workplace.

7. REFERENCES

Allen, I.E., & Seaman, J. (January, 2013) Changing Course: Ten Years of Tracking Online Education in the United States. Retrieved from http://www.onlinelearningsurvey.com/reports/changingcourse.pdf

Block, J. M.D., Issues for DSM-V: Internet Addiction. American Journal of Psychiatry 2008;165:306-307. doi:10.1176/appi.ajp. 2007.07101556 Retrieved from http://ajp.psychiatryonline.org/article.aspx?articleid=99602

Chang, C.J. & Man Law, S.P. (2008). Factor Structure for Young's Internet Addiction Test: A confirmatory study. Computers in Human Behavior, 24, 2597-2619.

Chou, C; Condron, L; & Belland, J. (December, 2005). "A Review of the Research on Internet Addiction", Educational Psychology Review, Vol 17, No. 4, 363-388.

Chou, C. (2001). Internet Abuse and Addiction among Taiwan college students: An online interview study. Cyber Psychological Behavior. 4 (5) 573-585.

Felix, Marcus, (2014) The Internet Addiction Cure - The Ultimate Guide to Help You Walk Away From the Computer (Obsessive Compulsive Disorder, OCD, Manic Depression, Internet disorder) (2014) Retrieved from http://amzn.com/B00J0JPA20.

Gnisci, A, Perugini, M, Pedone, R, & DiConza, A. (2011) Construct validation of the Use, Abuse, and Dependence on the Internet inventory. Computers in Human Behavior. 27, 240-247.

Internet Gaming Disorder. (n.d.) Retrieved from http://www.dsm5.org/Documents/Internet%20Gaming%20Disorder%20Fact%20Sheet.pdf.

Internet Users in the World. (n.d.) Retrieved from http://www.Internetlivestats.com

Joosten, E. G., De Weert-Van Oene, G. H., Sensky, T. T., Van Der Stask, C. F., & DeJong, C. J. (2011). Treatment Goals in Addiction Healthcare: the Perspectives of Patients and Clinicians. International Journal of Social Psychiatry, 57(3), 263- 276. doi:10.1177/0020764009354835

Jung Typology Test. Human Metrics. (n.d.) Retrieved from http://www.humanmetrics.com/cgi-win/JTypes1.html.

McAleer, B & Szakas, J. (2005) Don't Forget the Manager: Management of IT Professionals by IT Professionals for IT Professionals. In The Proceedings of the Information Systems Education Conference 2005, v 22 (Columbus OH): §3332. ISSN: 542-7382. (A later version appears in 3 4(95). ISSN: 1545-679X.)

___ (2004) Moving Past Gantt and PERT - Reinforcing Metrics as a Management Tool for CIS Students. In The Proceedings of the Information Systems Education Conference 2004, v 21 (Newport): §3445. ISSN: 1542-7382. (A later version appears in Information Systems Education Journal 4(20). ISSN: 1545-679X.)

Morahan-Martin, J. (Spring, 2005). Internet Abuse. Addiction? Disorder? Symptom? Alternative Explanations? Social Science Computer Review. 23, 39-48

Morahan-Martin, J. & Schumacker, P. (2000). Incidence and correlates of pathological Internet use. Computers in Human Behavior. 16, 13-29.

Mossbarger, B. (March, 2008) Is "Internet addiction" addressed in the classroom? A survey of psychology textbooks, Computers in Human Behavior, 24, 2. Pp.468-474. Retrieved from http://dl.acm.org/citation.cfm?id=1332141.1332408.

Scherer, K. (1997). College life online: Healthy and unhealthy Internet use. Journal of College Student Development 38(6), 655-665.

Tatl33. (2013, June 17). Which MBTI Type is most Addiction Prone. Retrieved from http;//whichmbtitype.wordprewss.com/2013/06/17/which-mbti-type-is-most-addiction-prone

Topi, H; Valacich, J..; Wright, R.; Kaiser, K; Nunamaker, Jr., J.; Sipior, J.; and de Vreede, G. (2010) "IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems," Communications of the Association for Information Systems: Vol. 26, Article 18. Retrieved from http://aisel.aisnet.org/cais/vol26/iss1/18.

Velezmoro, R, Lacefield, K & Roberti, J.W. (2010) Perceived stress, sensation seeking, and college students' abuse of the Internet. Computers in Human Behavior. 26. 1526-1530.

Young, Kimberly (2012). Managing Employee Internet Abuse in the Digital Workplace. Retrieved from http://www.igilobal.com/chapter/managing-employee-internet-abuse-digital/64786

____(2010) Policies and procedures to manage employee Internet Abuse. Computers in Human Behavior 26, 1467-1471.

___ (1998). Internet Addiction: The emergence of a new clinical disorder. Cyber Psychological Behavior, 2(5), 381-383.

___(1995). Internet Addiction Test. Retrieved from http://netAddiction.com/Internet-Addiction-test.

Evaluating the Effectiveness of Self-Created Student Screencasts as a Tool to Increase Student Learning Outcomes in a Hands-On Computer Programming Course

Loreen M. Powell
lpowell@bloomu.edu
Department of Business Education &
Information and Technology Management
Bloomsburg University of Pennsylvania
Bloomsburg, PA 18444, USA

Hayden Wimmer
hwimmer@georgiasouthern.edu
Department of Information Technology
Georgia Southern University
Statesboro, GA 30458

Abstract

Computer programming is challenging to teach and difficult for students to learn. Instructors have searched for ways to improve student learning in programming courses. In an attempt to foster hands-on learning and to increase student learning outcomes in a programming course, the authors conducted an exploratory study to examine student created screencasts and their impact on students' performance regarding specific learning outcomes in a hands-on programming course. This study was conducted over four semesters when an instructor taught two sections of the course per semester; one section generated self created student screencasts in-class and the other section did not. The subjects were undergraduate business students enrolled in an upper level applications/programming course at a university in Pennsylvania State System of Higher Education system. The experimental method was used to compare the differences in graded classroom activities, theory assessments, lab assessments, and final exam scores between the classes. Results showed that students who created screencasts while following along with the instructors step by step programming instructions as well as created screencast while independently working significantly ($p<.05$) performed more successful on theory assessments, lab assessments, and the final exam scores verses those students that did not.

Keywords: screencasts, programming, note taking, differentiated learning, active learning

1. INTRODUCTION

Teaching any programming course can be a challenge. However, when students don't buy the book, outline the chapter, take notes in class, review the content, redo the hands-on course material, nor have access to the computer programming application outside of class, it is impossible to successfully teach programming. Furthermore, the computer lab environment, where students learn hands-on computer programming, often makes it difficult for students to stop and take notes. In an effort to find a solution to these challenges, the authors experimented with screencasts. Screencasts are prerecorded videos that are designed to capture the author's computer screen and narration (Udell, 2005; Lang & Ceccucci , 2014). Previous research has identified instructor created screencasts as an good instructional tool in higher education (Ashdown, Doria, & Wozny, 2011; Lang & Ceccucci , 2014; Lee & Dalgarno, 2008; Peterson, 2007; Pinder-Grover, Green, & Millunchick, 2011; Sugar, Brown, & Luterbach, 2010; Winterbottom, 2007). However, no one has examined student created screencasts as a way to enhance learning outcomes in a hands-on learning environment. This exploratory study examines student created screencasts and their impact on students' performance regarding specific learning outcomes in a hands-on programming course. This work has practical implications for computer programming faculty and practitioners alike. The remainder of this paper is structured as follows: a brief review of programming pedagogy, screencasts and video usage, the methodology used in this study, results, conclusions and limitations.

2. LITERATURE REVIEW

Computer programming is one of the longest standing components in information technology/computer science degree programs. Computer programming requires students to logically understand abstract concepts, algorithms and data structure design, along with problem solving, testing, and debugging code (Wang, 2010). This subject matter has presented on-going teaching challenges and student learning difficulties (Sleeman, 1986; Ebrahimi, 1994; Jenkins 2002; Kinnunen et al. 2007; Mow, 2008; Nikula, Gotel, & Kasurinen 2011). Hence, it is no secret that teaching programming is a difficult task. The programming pedagogy literature provides a long list of failed methods known to impede students learning. Among the list of reasons as to why programming is difficult for students to learn is the lack of hands-on experience, student follow-up, and peer-driven learning (Babb et al., 2014).

Typically, the lack of hands-on experience occurs outside the classroom as students do not have access to programming software (Mow, 2008). Many students do not purchase programming software or fail to install the free programming software on their personal computers. Hence, not having access to programming software outside of class prevents students from having the necessary hands-on student follow up/content review, a process which is similar to rewriting lecture notes outside of a course.

Long standing research by Howe (1970) reported that note taking aids in student comprehension and recall. Specifically, there is only a 5% likelihood that content material will be remembered when it is not found in lecture notes (Howe, 1970, in Longman & Atkinson, 1999). However, not all students take notes and males take less notes then females (Cooperative Institutional Research program and the Higher Education Research Institute at University of California, Los Angeles, 2008 in the Chronicle of Higher Education, 2009). In 2009, Cooperative Institutional Research program and the Higher Education Research Institute at University of California, Los Angeles (UCLA) studied 26,758 students from 457 institutions and found that only 51% of males take notes in class. More importantly, their data also showed a decline of 7.5% from the previous years study (Ruiz et al., 2010).

Darmouth College is among the many universities and colleges that have developed websites compiled of note taking resources to helps students because "students frequently do not realize the importance of note taking and listening " (Darmouth College, 2013). Hence, the decline in note taking compounded with necessary hands-on experience has made it difficult for students succeed in programming courses.

Sreencasts and Video Usage

There is a large amount of research conducted on the effectiveness of using videos in the classroom. A recent student by Geri (2011) stated that "videos may improve the achievements of students enrolled in a course" (p.231). Additionally, Shultz and Sharp (2013) studied the effectiveness of using instructor created demonstration videos in a programming course. The instructors used Adobe Captivate to create a series of 20 minute videos for the main concepts of each chapter as well as how to program those concepts in C#. They reported that 89% of students (n=35) preferred videos more then text books.

A screencast is a video capture of the desired section of your computer screen that may or may not include webcam narration, voice narration, and text captions (Udell, 2005; Lang & Ceccucci, 2014). Screencasts are similar to video lectures, E-lectures, and e-notes in that they allow students to reflect back upon content previously learned.

Currently, screencasts have been used as instructional aids via instructor narrated

PowerPoint presentations or lectures, problem-solving demonstrations and application demonstrations (Lang & Ceccucci, 2014). Existing research has shown several positive benefits including, but not limited to, student learning flexibility with asynchronous access, instructor tracking of usage, instructor reusability and increased student performance. Most importantly, statistically significant differences, correlations and percentages have been found with students using instructor created screencasts as a classroom supplement (Falconer et al., 2009; Lloyd & Roberson, 2012; Mullamphy, Higgins, Belward & Ward 2009, Pindar-Grover et al. 2011, Lang & Ceccucci, 2014).

Most of the existing research has focused on instructor created screencasts. A recent literature review by Berardi and Blundell (2014) suggested that student created course materials may have the potential to add value in hands-on experience and peer-driven learning. However, there has been little or no research conducted on student created screencasts in-class and their impact on learning outcomes in a hands-on programming course.

3. METHOD

The purpose of this exploratory research study is to understand the value of student's self-created screencasts as a tool to increase students' success in a hands-on application/programming course. Specifically, this study's research question is:

- In a hands-on programming course, will there be a significant difference in the classroom activities, theory assessments, lab assessments and final exam scores for students that self-create screencasts for instructor and independent hands-on programming versus those who did not?

This study was set up as an experiment for over four semesters. Each semester, one class section created screencasts and the other did not. Before starting the semester, the instructor designated one of the classes as the experimental group and the other as the control group. The experimental group created screencasts and the control group did not create screencasts. Subjects were undergraduate students enrolled in a Pennsylvania State System of Higher Education (PASSHE) University. Students were enrolled in eight different sections of an upper level applications/programming course where students learn to program with Scratch, Alice, Visual Basic, and Stencyl.

The applications/programming course met for fifty minutes three times a week. The class was structure so that students spent fifth teen minutes with theory concepts, fifth teen minutes with hands-on instruction and fifth teen minutes independent hands-on student centered learning with instructor supervision and guidance. The instructor always ensured that the students had five minutes upload their screencasts to screencast-o-matic.com before the class ended.

The same course materials (i.e. lectures, book, theory assessments, lab assessments, and final exam) were used. Each course was fifty minutes in length and followed an introduce, reinforce and apply format. Students in the experimental sections were required to record their own Screencast using www.screencast-o-matic.com. Screencast-o-matic was chosen because it was free, required no software to be downloaded and was accessible anywhere with an internet connection. Screencast-o-matic is also very simple to use and does not require multitasking difficulties or toggling between applications.

Upon the start of the first day of class students in the experimental group were asked to sign up and create a free screencast-o-matic account. By creating a screencast-o-matic account, students had immediate online access to store their self-created screencasts online. This made it easy for students to retrieve their screencasts after class. Only students themselves had access to their screencast-o-matic accounts. The instructor did not have access to student's accounts nor did the instructor ask to view student's accounts.

After the experimental group of students established their own screencast-o-matic accounts, they are given seven simple instructions on how to create a screencast:

1. Go to http://screencast-o-matic.com
2. Login
3. Click on "Start Recording"
4. Resize the recording frame to fit your programming screen.
    - To pause recording, click the universal pause icon located at the bottom left side of the screen
    - To restart recording, click the red circle icon located at the bottom left side of the screen

5. When you are finished, click "Done" at the bottom of the screen.
6. Next click "Publish to Screencast-o-matic"
7. Type in the name your screencast and click "Publish"

Students in the experimental sections were asked to record/create their own screencast while following along with the instructors 15 minute hands-on step by step classroom instructions. Students were also asked to record/create their own screencast while independently working hands-on for 15 minutes. Students did not create screencasts during the theory content or theory lecture. Each student created screencast rerecorded during the Instructor led hand-on programming session directly corresponded to chapter content in the programming text.

It is also important to note that the text book used for both the control and experimental group had narrated videos that were created by the publisher to go along with each chapter. While the instructor did not bring this to the students attention, it was presented in the book as a tool to help students.

Throughout the semester, the instructor took attendance at the beginning of class. If students attended all instructional classes they would have a total of 23 self created screencasts of instructor led hands-on programming and 23 self-created independent student learning hands-on programming screencasts. Each screencast was 15 minutes or less in duration. Each screencast only used the video recording of the students screen. Students did not use the video cam or voice recording features.

During all activities, assessments and final exam review classes, the instructor encouraged the experimental group to reference their own screencast as a helpful way to study. Data was collected via the student's assessment scores on the activities, assessments and the final exam for all eight courses.

4. RESULTS

Statistical analyses were conducted using the Statistical Product and Service Solutions (SPSS) software. Descriptive and inferential statistics, including mean, standard deviation, and two-tailed *t* tests were used to test the research question.

The overall sample size included 225 undergraduate business students enrolled in an upper level undergraduate applications/programming course. Table 1.1 and Table 1.2 provide demographic details about the students.

Table 1.1 Class Status

	Senior	Junior	Sophomore.	Freshman
Experimental	71	27	13	0
Control	58	36	19	1

Table 1.2 Gender

	Male	Female	Total
Experimental	93	18	111
Control	88	26	114

As indicated in Table 1.1, there were 111 students in the experimental group and 114 students in the control group.

According to the instructors records, 68% of students in the experimental group attended all of the classes where students created screencasts. A total of 92% of students only missed less then two classes and 100% of students missed less then five classes.

Results indicate that there is no significant difference between the classroom activities grades for the experimental and control group. However, there was a significant difference (p=.031) between the theory assessment scores for the experimental group and control groups. The experimental group scored slightly higher (M=79.61, SD= 6.31) than the control group (M=72.17, SD=8.44).

A significant difference (p=.048) was also found between the lab assessment scores for the experimental group and control groups. The experimental group scored higher (M=85.33, SD= 5.71) than the control group (M=73.30, SD=7.74).

Another significant difference (p=.026) was also found between the final exam scores for the experimental group and control groups. The experimental group scored higher (M=89.22, SD= 7.42) than the control group (M=82.27, SD=9.55).

Additional analyses were conducted on gender and class status (Freshman, Sophomore, Junior, and Senior). However, there were no significant differences found.

5. CONCLUSION AND LIMITATIONS

The results indicate that by having students create their own screencast while following the hands-on instruction from the instructor as well as independent hands-on student work can be a useful tool to increase student learning outcomes. Test scores in key content areas were enhanced for those students who created screencasts versus those who did not create screencasts but may have taken notes.

This research is important information because currently fewer students are taking notes in classes. This study helps to encourage note taking via student created screencasts. By encouraging students to create their own screencasts during hands-on instruction periods, a useful tool is created for students to review hands-on class content at a later time. Additionally, by having the student create the screencast, the instructor is placing the responsibility for a successful learning experience on the student.

This study is not without limitations. This study made no attempt to control for variables that may impact student performance on activities, theory assessments, lab assessments, and the final exam other then the students in-class screencast creation. Additionally, students were not surveyed or interviewed following the course, so it is uncertain if the experimental students used their created screencasts to review classroom materials for graded activities, theory assessments, lab assessments and the final exam. Furthermore, the authors are uncertain if students in the experimental group collaborated or shared screencasts with students in the control group.

Nevertheless, this study demonstrated how student created screencasts can be used as a tool to increase learning outcomes of a hands-on programming course. Further research should better control variables for construct validity. Finally, further research should be conducted with a larger sample size from various hands-on courses in various computer lab environments.

6. REFERENCES

Ashdown, J., Doria, D. & Wozny, M. (2011). Teaching practical software tools using screencasts while simultaneously reinforcing theoretical course concepts. American Society of Engineering Education St. Lawrence Section Conference. Retrieved August 1, 2014 from http://stl.asee.org/papers_2011/Ashdown.pdf

Babb, J. S., Longenecker, B., Baugh, J. & Feinstein, D. (2014). Confronting the issues of programming in information systems curricula: the goal is success. *Information Systems Education Journal, 12*(1), 42-72.

Berardi, V. & Blundell, G. E. (2014). A learning theory conceptual foundation for using capture technology in teaching. *Information Systems Education Journal, 12*(2), 64-73.

Darmouth College (2013). Classes: note taking, listening, participation. Retrieved on August 10, 2014 from http://www.dartmouth.edu/~acskills/success/notes.html

Ebrahimi, A. (1994). Novice programmer errors: language constructs and plan composition. *International Journal of Human Computer Studies, 41*(4), 457-480.

Falconer, J., DeGrazia, J., Medlin, J. & Holmberg, M. (2009). Using screencast in CHE courses. *Chemical Engineering Education*, *43*(4), 302-305.

Geri, N. (2011). If we build it, will they come? Adoption of online video-based distance learning. *Interdisciplinary Journal of E-Learning and Learning Objects, 7*(1)*, 225-234.*

Howe, M. J. (1970). Notetaking strategy, review and long-term retention of verbal information. *Journal of Educational Research*, *63*(1), 285.

Jenkins, T. (2002). On the difficulty of learning to program. In *Proceedings of the 6th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE'01), 54-56.*

Kinnunen, P., Mccartney, R., Murphy, L. & Thomas, L. (2007). Through the eyes of instructors: a phenomenographic investigation of student success. In *Proceedings of the 3rd International Workshop on Computing Education Research Conference,* 61-72.

Lang, G. & Ceccucci, W. (2014). Clone yourself: using screencasts in the classroom to work with students one-on-one. *Information Systems Education Journal, 12*(6), 4-14.

Lee, M. & Dalgarno, B. (2008). The effectiveness of screencasts and cognitive tools as scaffolding for novice object-oriented programmers. *Journal of Information Technology Education*, *7*(1), 61-80.

Lloyd, S. & Robertson, C. (2012). Screencast tutorials enhance student learning of statistics. *Teaching of Psychology, 39*(1) 67-71.

Longman, D. G. & Atkinson. R. H. (1999). *College Learning and Study Skills*. Belmont, CA: Wadsworth.

Mow, I. T. C. (2008). Issues and difficulties in teaching novice computer programming. *Innovative Techniques in Instruction Technology, E-Learning, E-Assessment, and Education,* M. Iskander Ed., Springer, 199-204.

Mullamphy, D., Higgings, P., Belward, S. & Ward L. (2009) To screencast or not to screencast. *Anziam Journal*, *51*(1), 446-460.

Nikula, U., Gotel, O. & Kasurinen, J. (2011). A motivation guided holistic rehabilitation of the first programming course. *ACM Transactions in Computer Education, 11*(4), 1-38.

Peterson, E. (2007). Incorporating screencasts in online teaching. *International Review of Research in Open and Distance Learning*, *8*(3), 1-4.

Pinder-Grover, T., Green, K. & Millunchick, J. M. (2011). The efficacy of screencasts to address the diverse academic needs of students in a large lecture course. *Advances in Engineering Education*, *2*(3), 1-28.

Ruiz, S., Sharkness, J., Kelly, K., DeAngelo, L. & Pryor, J. H. (2010). Findings from the 2009 administration of the first college year (YFCY): National aggregates. Retrieved August 1, 2014 from http://www.heri.ucla.edu/PDFs/pubs/Reports/YFCY2009Final_January.pdf

Sharp, J. H. & Schultz, L. A. (2013). An exploratory study of the use of video as an instructional tool in an introductory c# programming course. *Information Systems Education Journal, 11*(6), 33-39.

Sleeman, D. (1986). The challenges of teaching computer programming. *Communications of ACM, 9*(29), 840-841.

Sugar, W., Brown, A. & Luterbach, K. (2010). Examining the anatomy of a screencast: uncovering common elements and instructional strategies. *International Review of Research in Open and Distance Learning*, *3*(11), 1–19.

The Chronicle of Higher Education (2009), Men, take note. Retrieved August 1, 2014 form http://chronicle.com/article/Men-Take-Note/48317/

Udell, J. (2005). What is screencasting? Retrieved 5/20/2013 from http://www.oreillynet.com/pub/a/oreilly/digitalmedia/2005/11/16/what-is-screencasting.html

Wang, X. O. P. H. I. E. (2010). Teaching programming skills through learner-centered technical reviews for novice programmers. *Software Quality Professional*, *13*(1), 22-28.

Winterbottom, S. (2007). Virtual lecturing: delivering lectures using screencasting and podcasting technology. *Planet*, *18*(1), 6-8.