PAIRING CONSTRUCTIVISM WITH THE MULTIDISCIPLINARITY OF IS EDUCATION

Samuel B. Fee, Washington & Jefferson College, sfee@washjeff.edu
Amanda Holland-Minkley, Washington & Jefferson College, ahollandminkley@washjeff.edu
Thomas Lombardi, University of the Virgin Islands, thomas.lombardi@uvi.edu

ABSTRACT

This paper explores two common threads in IS education, constructivist approaches like active learning and the multidisciplinary educational objectives of Information Systems as a discipline, and demonstrates how these themes are well paired. The paper builds on prior research into effective applications of interdisciplinary and multidisciplinary computing education (Fee, Holland-Minkley, & Lombardi, 2017) and presents a deep survey of the literature in IS education to identify the primary forms that multidisciplinary education takes within the classroom or curriculum. This survey demonstrates how the multidisciplinarity of IS is consistently enhanced by the application of constructivist techniques. The discussion addresses how the predicted benefits of both using constructivist approaches and including multidisciplinary content within a course are well aligned with the objectives of IS education. This will include considering how these two elements of IS education can be paired to support greater diversity and inclusion within the discipline.

Keywords: multidisciplinary computing, constructivist instruction, diversity, pedagogy, project-based learning, active learning

INTRODUCTION

There is extensive discussion in the IS education literature of the merits of constructivist theory (Connolly & Begg, 2006; Tobias & Duffy, 2009; Rob & Rob, 2018) and approaches such as active learning (Astani, 2006), project-based learning (Wang, 2015), problem-based learning (Fee & Holland-Minkley, 2012), studio-based learning (Vest, Champan, & Denton, 2011), and other learner-centered techniques (Ramos-Torres, 2014; Guzdial, 2016). These approaches commonly present students with real-world problems that lend authenticity to the problem solving process and engage student interest. These approaches are also commonly valued for their ability to help students develop both technical and "soft" skills.

The IS education literature contains an underlying thread focusing on the inherently multidisciplinary nature of Information Systems (Cohen, 2000; Welch, Madison, & Welch, 2008; Kohun, Rodi, & Lorenzo, 2012;Barr, 2016; Holland-Minkley & Fee, 2017). Integrating skills from both the computing and business disciplines, and often applied within industries with their own disciplinary concerns such as healthcare or manufacturing, IS practitioners must be educated to be multidisciplinary thinkers.

This paper will explore these two common threads in IS education and demonstrate how constructivist approaches and multidisciplinary educational objectives can be well paired. Building on prior research into effective applications of interdisciplinary and multidisciplinary computing education, (Fee, Holland-Minkley, & Lombardi, 2017) the paper carries out a deep survey of the literature in IS education to identify the primary forms that multidisciplinary education takes within the classroom or curriculum. This survey demonstrates how the multidisciplinarity of IS is consistently enhanced by the application of constructivist techniques. The review discusses the potential benefits of using constructivist approaches and multidisciplinary content within a course. This will include considering how these two elements of IS education can be paired to support greater diversity and inclusion within the discipline.

REVIEW OF CONSTRUCTIVISM

In a broad sense, a constructivist approach is one which includes learners as partners in obtaining and structuring knowledge and problem solving abilities rather than serving as recipients of instructional content. A constructivist epistemology suggests that learners achieve more through correlating educational content to real-life experiences, and that such an approach informs their ability to conceptualize content (Duffy & Jonassen, 1992; Rob & Rob, 2018). Constructivism suggests that educators seek circumstances for learning that are experiential, active, collaborative, and that also develop problem-solving skills (Jonassen, 2000). These constructivist pedagogies include approaches such as cognitive apprenticeship (Collins, Brown & Holum, 1991), situated learning (Lave & Wenger, 1990), active learning (Bonwell, & Eison, 1991), project-based learning (Blumenfeld, et al., 1991) and problem-based learning (Savery, 2015).

However it is important to note that pedagogical approaches do not ensure constructions of knowledge – if the mind is not engaged the activity alone will not result in knowledge acquisition (Resnick, 1997). Constructivist approaches are driven by an awareness that students learn better when they see the need for what they are learning. This appeal to student motivation (Savery & Duffy, 1995) is also one of the major drivers of increased interdisciplinary and multidisciplinary content within programming courses. There is a natural alignment between the underlying reasons an educator might adopt a constructivist approach and choose to design a course to engage students with multidisciplinary content and skills. In constructivist approaches, student motivation is often accomplished through a focus on robust and real problems (Savery, 2015). The multidisciplinary context of Information Systems can be a rich source for these problems.

One challenge to implementing constructivist instruction is that educators do not always know the prior knowledge and experiences their students are bringing to the classroom and this can complicate crafting appropriate activities and projects or creating collaborative groups (Tobias & Duffy, 2009). However, we suggest that instructors who are able to rise to this challenge can find that constructivist approaches provide a tool for managing the increasingly varied background of students, particularly in introductory and programming courses. Giving students ownership over the learning activity can let them leverage the knowledge they already possess and illuminates the students' background to the instructor. This increases the instructor's ability to adjust to the level of prior knowledge; however it requires more effort to evaluate and respond to those varying levels of preexisting knowledge as the instruction is underway. In this way, constructivist approaches align well with goals of attracting a more diverse student body, both with respect to the demographics of students and also with respect to attracting students from other business disciplines to coursework in IS. Pedagogical approaches that are designed to work with rather than attempt to alleviate or eliminate differences between students are not only effective but avoid pedagogies that may implicitly communicate to some students that they are not appropriately equipped for a technical classroom. As discussed below, the design of technical courses around the multidisciplinary context of IS can enhance this approach.

CONSTRUCTIVISM IN MULTIDISCIPLINARY IS CONTEXTS

IS education has embraced constructivist instruction in many forms, ranging from culminating capstone experiences to introductory coursework. When examining these applications of constructivism, it is striking to note that such approaches are consistently deployed to enhance the multidisciplinary skill development of IS students.

Many have written about the inherent multidisciplinarity of Information Systems. Cohen (2000) writes about the placement of Information Systems within the transdiscipline of Informing Science, with IS specifically drawing on topics from technology, business, and issues in how people use information. Kohun, Rodi and DeLorenzo (2012) characterize their Computer Information Systems program as having a "blended technical and business orientation" and describe how their program produces "a multi-skilled, multi-dimensional graduate". The ACM 2010 model IS curriculum states that one of its major revisions is to broaden the model beyond a typical business school context in recognition of the fact that Information Systems is central to an increasing range of disciplines (Topi et al., 2010). In fact, the model curriculum was developed following the principle that "the model curriculum is not restricted to a specific domain; all Information Systems programs are, however, linked to some domain".

This section looks at four common formats where constructivist approaches are aligned with the underlying multidisciplinary goals of IS education.

Capstone Courses
The most prominent examples of hands-on constructivist and integrative learning in the IS curriculum are likely through capstone courses or similar culminating senior experiences. These courses include project-based and collaborative learning approaches to engage students with these multidisciplinary problems.

It is common for IS capstone courses to incorporate real-world projects, often for a client, that require technical expertise, communication skills, and project management abilities. One study found real-world projects employed in 69% of the courses surveyed that offered an IS capstone course; these real-world projects are lauded for the opportunity they provide "for students to learn multi-disciplinary skills that will make them more competitive in the marketplace" (Kovalchick, Boff, & Kovaks, 2013). In some cases, these capstone experiences can be spread across multiple courses, enhancing student learning while also increasing the amount of engagement possible with industry partners. The Innovations for Industry culminating experience at Juniata College involves students in this work at multiple points in the curriculum by engaging mixed teams of third and fourth year students on applied projects for local industry (Thomas, Rhodes, & Kruse, 2017). Even when client-facing projects are not employed, IS capstones frequently integrate multidisciplinary skills using an active learning approach (Mitri, 2006; Astani, 2006). This is a productive pairing – robust, authentic problems are important in ensuring that students exercise a range of skills in solving a problem and are a key element of active learning and constructivist approaches.

Cameron University has extended this model by combining the capstone courses from its Management Information Systems, Computer Science, and Multimedia majors (Smith, Estep, Zhao, Moinian, & Johari, 2014). Within their model, MIS and CS students work in integrated teams, with a team of multimedia students supporting these integrated teams. They have also included students from disciplines such as English and Foreign Language to offer technical writing support. The course uses experiential service learning where students undertake an authentic database project for a business client. In addition to providing a domain-embedded problem, students learn to collaborate across disciplines; evaluation of the program showed that students found the course effective at helping them collaborate and work in teams and many were aware that this type of multidisciplinary setting reflected the environment they would be working in after graduation. Here too, there is a shared emphasis on collaboration within both the content and the pedagogy of the course.

Multidisciplinary IS Projects
Prior to the capstone, IS students are likely to have encountered work in their IS courses that incorporate constructivist approaches such as project-based learning and that emphasize the multidisciplinarity of the IS discipline.

In introductory courses, this pairing is used to strong effect to generate interest in the IS major. While enrollments have seen improvements, within the last decade there was a significant decline in the number of students interested in the IS field. Many have suggested that student perceptions of IS, such as lack of relevance, only being for "geeky" students, fears of outsourcing, or an overall misunderstanding of the IS field contributed to this decline (Ramos-Torres, 2013; Kohun, Rodi, & DeLorenzo, 2012). Introductory courses take advantage of the increased engagement produced by constructivist techniques to generate student interest. A project-based approach to introductory IS has been found to have some positive effect on increasing student interest (Marshall, Cardon, & Godin, 2014). In their application of a project-based learning approach to an introductory course, Frost, Pike and Kenyo (2008) stress that presenting projects that allow students to see business-oriented content will illuminate the value of the more technical content within the course. They also observe that aligning the content and the pedagogy of the course to be interesting and creative can help faculty have a more positive experience teaching the introductory course. Focusing on both constructivist instruction and multidisciplinary work to create early student engagement in IS can be linked to the body of work on revising introductory programming courses in CS majors to be project-based and multidisciplinary in order to engage students and attract a more diverse student body (Barr, 2016; Baldwin et al., 2017).

Systems Analysis and Design courses also frequently merge project-based learning with multidisciplinary projects to good effect. The use of project-based learning is motivated by its effectiveness at engaging students with the knowledge and skills from both business and technical domains, with multidisciplinary projects of small groups or pairs of students found to increase student completion rate and overall performance (Wang, 2015). Rob (2006) discusses the ways in which real-word projects can improve student development of "hard skills" as well as "soft skills", particularly when multidisciplinary projects are selected that are authentic to the business environment students will be working in and capture many cross-functional areas. Creating a tangible final product reinforces the constructivist mindset underlying project-based learning but also ensures that students must engage with the entire range of skills underlying effective analysis and design (Rob & Rob, 2016). In more recent work, Rob and Rob (2018) have implemented a blended framework based on the software development life cycle, project-based learning and presentation in a graduate data warehousing class and an undergraduate systems analysis and design class.

Other content-area courses also use this pairing of active- or project-based learning and multidisciplinary content. Cybersecurity and Information Security courses often pursue an active-learning approach centered around simulations or "serious games" to provide the benefits of hands-on learning while mitigating the risks involved in permitting students to engage with live information security problems (Yerby, Hollifield, Kwak, & Floyd, 2014). For example, North (2016) compared the efficacy of simulation versus virtual machines in cybersecurity education. These simulations can engage students in a broad range of skills necessary for information security professionals, from interviewing or reporting to forensic analysis. Games and simulations can also be deployed in data analytics and decision making courses; Wang (2018) merges learning about business and management processes with learning about data analytics through engaging students in an ERP simulation that provides students with access to real-time data analytics about the simulation as they proceed through it. This multidisciplinary approach is enabled by substituting active-learning through ERPsim for more traditional case studies.

Multidisciplinarity in Programming Instruction

Effective programming education is a challenge across computing disciplines, with many papers dedicated to the topic from the perspective of IS, IT, and CS education. Within IS in particular, there is ongoing debate around the proper role of programming within the IS curriculum (Baugh & Kovacs, 2014). Whether required or elective, teaching students programming has common challenges related to the underlying difficulties of instructing in problem solving, regardless of the discipline carrying out the instruction (Schuyler & Skovira, 2007).

With programming generally being a first course within the CS curriculum, the CS education community has done significant work on effective programming instruction. Much has been written about the effectiveness of incorporating interdisciplinary and multidisciplinary content in order to improve student engagement and learning in programming courses. Interdisciplinary approaches to programming instruction have been found effective at increasing student engagement overall and is particularly recognized to be effective in broadening participation in computing (Lewis, 2017). Guzdial's work on Media Computation and making introductory computing and programming education accessible for all students is well known (Guzdial, 2016). The EngageCSEdu project (EngageCSEdu, n.d.) makes explicit the alignment between content and pedagogy by highlighting instructional activities that use relevant content, highlight interdisciplinary connections, let students choose course topics or problems, and including student interaction or collaborative learning. Faculty who have developed these materials and used them in their classrooms report that incorporating interdisciplinary content into their computing courses particularly helps engage students from other majors and promote computational work across disciplines (Boese, LeBlanc, & Quinn, 2017).

We posit that the inherent multidisciplinary nature of Information Systems positions the field well to similarly take advantage of the benefits of multidisciplinary content in an introductory programming course. As the research above indicates, student engagement and learning can be enhanced by this approach. Additionally, contextualizing programming can help IS students better see the relevance of programming to their field of study as compared to programming being taught as an isolated "tech only" course.

The pairing of multidisciplinary content with constructivist pedagogies is currently seen within many introductory programming courses taught within IS. Problem-based learning has been effectively used in a games-programming course to improve student success (Soares, Fonseca, & Martin, 2015). The multidisciplinary domain of game design attracts students while the paired problem-based learning pedagogy helps motivate and engage students in the

difficult task of learning to program. Baugh and Kovacs (2012) have used a project-based pedagogy in an introductory programming course that involves students in large, real-world projects such as drawn from a hospital surgery department or a college's Registrar office. Institutions have also used studio-based learning within their introductory programming courses because of its ability to motivate students and provide support for student learning through an improved sense of community (Vest, Chapman, & Denton, 2011). There is here again an argument for the merits of these approaches not just for student success but also for inclusivity, as the approach encourages not just perseverance but also a sense of affinity.

We also observe that multidisciplinary approaches to programming instruction do not have to be restricted to just business content. The CIS faculty at Elon University have pursued this philosophy by developing an introductory bioinformatics course in partnership with the Biology department that draws students from backgrounds in human services, history, communications, and biology and introduces them to bioinformatics content including programming (Heinrichs & Niedziela, 2009). The course also introduces core IS content such as the discussion of social, ethical and legal issues in the development and use of bioinformatics systems.

Curricular-Level Innovations
Most ambitiously, some programs are taking a curriculum-wide approach to integrating constructivism with their multidisciplinary educational goals. These programs embed both multidisciplinary content and constructivist pedagogies in the mission and structure of the program.

The Computer Information Systems program of the University of Puerto Rico has formally embedded project-based learning in their curriculum. This work was done in conjunction with reviewing the skills they wanted students to learn, recognizing that employers are looking for graduates with multidisciplinary skills crossing technical, business, and soft skills (Ramos-Torres, 2013; Ramos-Torres, 2014). Extensive work has been done to align the important skills within the Computer Information Systems major with multiple projects across all levels of the curriculum (Ramos-Torres & Rodriguez-Orellana, 2015). Notably, projects are also reviewed to ensure that they emphasize multiple skills of different types. In this way, the pedagogical approach is intertwined with the multidisciplinary nature of the content and skills being learned.

At Washington & Jefferson College, the Computing and Information Studies department serves an interdisciplinary mission and has also constructed an academic major organized around constructivist principles (Holland-Minkley & Fee, 2010). Courses from the introductory level through the service-learning project management capstone draw on contextualized problems. Additionally, a recent curriculum revision enhanced the support for this approach by embedding the scaffolding for a shared problem-based learning pedagogy across the curriculum as a whole (Holland-Minkley & Fee, 2017). With the interdisciplinary work being naturally constructivist, students benefit from a curriculum that is designed to orient students to the sorts of exploratory problem-solving from their first course and support them not only in gaining computing proficiency but gaining proficiency as effective learners in a constructivist setting (Fee & Holland-Minkley, 2012).

It is interesting to contextualize these programs with the trend in computing education of developing integrative and cross-discipline "CS+X" majors or "joint majors", such as at Stanford University, University of Illinois, and Northwestern. These programs generally differ from a double major by embedding opportunities for students to integrate their learning across the two majors. An effective case can be made, though, that students consider pursuing IS as a second major in support of a primary disciplinary major. For example Welch, Madison, and Welch, (2008) found that employers rated Information Systems to be particularly valuable as a supporting major to an undergraduate accounting major. In all of the cases discussed in this section, educators are experimenting with new possibilities for coordinating and integrating constructivist approaches across the curriculum.

DISCUSSION

The review of the literature demonstrates that IS educators are actively engaging with constructivist approaches such as active learning, project-based learning, and collaborative learning for a variety of reasons, including the goals of attracting students to study IS, increasing student engagement, improving student outcomes, and preparing students for careers.

What is less explicitly discussed is the positive way that these pedagogies interact with the multidisciplinary nature of IS. Developing multidisciplinary skills, particularly with an eye to how they will be used in the workplace, requires that students learn how to execute these skills in concert with each other rather than in isolation. This requires coursework where students work in multidisciplinary ways. Engaging students in these types of robust problems is challenging, and constructivist approaches are effective in helping students succeed in bringing multiple skills together in complex ways.

While some educators are clear about this, it would be valuable for IS educators to continue to communicate the message that IS work is inherently multidisciplinary, particularly to their students. One productive way to make this case is to discuss the ways in which IS careers are increasingly multidisciplinary. The ACM 2010 IS model curriculum lists as its first guiding assumption about the information systems profession that "IS professionals exist in a broad variety of domains, including, for example, business, health care, government, and non-profit organizations" (Topi et al, 2010). This is supported by reviews of job postings that show that most technology job postings are within industries such as health care, banking and finance, manufacturing, and retail instead of within traditional tech companies (Chamberlain, 2017; Stolzoff, 2018).

Beyond helping students appreciate the breadth of skills they must master, focusing on the multidisciplinary nature of IS can help diversify the discipline and thus the IS and technology workforce. Between 2009 and 2015, women were consistently awarded about 22% of all IS bachelor's degrees each year, black, non-Hispanic students were awarded about 15% of all IS bachelor's degrees, and Hispanic or Latino/a students were awarded about 7-8% of all IS bachelor's degrees (National Academies of Sciences, Engineering, and Medicine, 2018). This report from the National Academies of Sciences, Engineering, and Medicine on the problems of representation in computing disciplines specifically recommends best practices for addressing the issue that include constructivist instructional choices such as including service learning, real-world projects, or collaborative problem solving as well as specifically recommending interdisciplinary projects and considering the multiple pathways that can be used to draw students into computing majors. That IS in many cases is already adopting these approaches may explain why representation is somewhat better in IS than in CS, particularly for women. But with more progress to be made, these efforts can be expanded upon.

CONCLUSION

This paper illustrated the productive ways that IS educators are deploying constructivist approaches to instill students with the inherently multidisciplinary skills required for a career in IS. This work is taking place at multiple points in the IS curriculum and, in some cases, is being integrated into the curriculum as a whole. Evidence shows that students are benefiting in terms of improved learning outcomes and greater engagement with their studies as well as, in many cases, greater enjoyment of their studies. Faculty and academic programs are benefiting by attracting more and, it seems, more diverse students and being able to engage with those students in more interesting ways. In some cases, though, the connections between pedagogy and multidisciplinary goals are more implicit than explicit and therefore educators may wish to consider the benefits of more intentionally and explicitly connecting these elements of their instruction.

REFERENCES

Astani, M. (2006). The MIS capstone course: An active learning approach.
Issues in Information Systems, 7(1), 119-123.

Baldwin, D., Barr, V., Briggs, A., Havill, J., Maxwell, B., & Walker, H. M. (2017).
CS 1: Beyond programming. *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science
    Education* - SIGCSE '17 (pp. 677-678). New York, NY: ACM.

Barr, V. (2016). Disciplinary thinking, computational doing: Promoting interdisciplinary computing while
transforming computer science enrollments. *ACM Inroads, 7*(2), 48–57.

Baugh, J., & Kovacs, P. (2012). Large programming projects for the beginning programmer.
Issues in Information Systems, *13*(1), 85-93.

Baugh, J., & Kovacs, P. (2014). Programming not required? Did the IS-model curriculum get it right?
Issues in Information Systems, *15*(1), 399-408.

Blumenfeld, P., Soloway, E., Marx, R., Krajcik, J., Guzdial, M., & Palinscar, A. (1991). Motivating project-based
learning: Sustaining the doing, supporting the learning. *Educational Psychologist, 26*(3&4), 369–398.

Boese, E., LeBlanc, M., & Quinn. B. (2017). EngageCSEdu: Making interdisciplinary connections to engage
students. *ACM Inroads 8*(2), 33-36.

Bonwell, C., & Eison, J. (1991). *Active learning: Creating excitement in the classroom.*
AEHE-ERIC Higher Education Report No. 1. Washington, D.C.: Jossey-Bass.

Chamberlain, A. (2017). Beyond silicon valley: Tech jobs spreading out of tech hubs. *Glassdoor*. Retrieved from
    https://www.glassdoor.com/research/beyond-silicon-valley-tech-jobs-spreading-out-of-tech-hubs/.

Cohen, E. (2000). A guide to the transdiscipline informing science for the perplexed and multiplexed.
Issues in Information Systems. 1(1), 73-78.

Collins, A., Brown, J. S., & Holum, A. (1991). Cognitive apprenticeship: Making thinking visible.
American Educator, 15(3), 1–18.

Connolly, T., & Begg, C. (2006). A Constructivist-based Approach to Teaching Database Analysis and Design.
Journal of Information Systems Education, 17(1), 43-53.

Duffy, T., & Jonassen, D. (1992). *Constructivism and the technology of instruction: A conversation.*
Philadelphia, PA: Lawrence Erlbaum.

EngageCSEdu. (n.d.). *Engagement Practices*.
Retrieved from https://www.engage-csedu.org/engagement/make-it-matter

Fee, S. B., & Holland-Minkley, A. M. (2012). Correlating problems throughout an interdisciplinary curriculum.
In Fee, S.B. & Belland, B. R. (Eds.). *The Role of Criticism in Understanding Problem Solving*
 (pp. 215-230). New York, NY: Springer.

Fee, S. B., Holland-Minkley, A. M., & Lombardi, T. E. (Eds). (2017). *New Directions for Computing Education:
Embedding Computing Across Disciplines*. New York, NY: Springer.

Frost, R., Pike, J., & Kenyo, L. (2008). Generating student interest in the information systems major:
A strategic framework for the introductory course. *Issues in Information Systems*, *9*(1), 188-195.

Guzdial, M. (2016). *Learner-Centered Design of Computing Education: Research on Computing for Everyone*.
Williston, VT: Morgan & Claypool Publishers.

Heinrichs, L., & Niedziela, L. (2009). Building a bioinformatics bridge one course at a time.
Issues in Information Systems, 10(1), 97-102.

Holland-Minkley, A.M., & Fee, S. B. (2017). An interdisciplinary model for liberal arts computing curricula.
In S.B. Fee, A. M. Holland-Minkley, & T. E. Lombardi (Eds.), *New Directions for Computing
Education: Embedding Computing Across Disciplines* (pp.161-185). New York, NY: Springer.

Jonassen, D. (2000). Toward a design theory of problem solving. *Educational Technology Research
and Development 48*(4), 63–85.

Kohun, F., Rodi, A., & DeLorenzo, G. (2012). Does information systems suffer an identity crisis?
A case study of confusion and misinformation. *Issues in Information Systems*. *13*(2), 328-335.

Kovalchick, L., Boff, G., & Kovacs, P. (2013). Using live projects in an information systems capstone course.
Issues in Information Systems, *14*(2), 149-155.

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.*
New York: Cambridge University Press.

Lewis, C.M. (2017). Good (and bad) reasons to teach all students computer science.
In S.B. Fee, A. M. Holland-Minkley, & T. E. Lombardi (Eds.), *New Directions for Computing Education:
Embedding Computing Across Disciplines* (pp.15-34). New York, NY: Springer.

Marshall, B., Cardon, P., & Godin, J. (2014). A study of project-based learning in an introductory MIS course.
Issues in Information Systems, *15*(2), 24-30.

Mitri, M. (2006). A capstone project in software development for CIS majors.
Issues in Information Systems, *7*(1), 51-55.

National Academies of Sciences, Engineering, and Medicine (2018). *Assessing and Responding to the Growth of
Computer Science Undergraduate Enrollments*. Washington, DC: The National Academies Press.

North, M. (2016). War Games: Simulation vs. Virtual Machines in Cybersecurity Education.
Issues in Information Systems, *17*(4), 120-126.

Ramos-Torres, A. (2013). Current IS program challenges and proposed strategies to address them.
Issues in Information Systems, 14(2), 101-108.

Ramos-Torres, A. (2014). Experiential learning to make information systems programs more responsive to student
and industry needs. *Issues in Information Systems*, *15*(1), 88-97.

Ramos-Torres, A., & Rodriguez-Orellana, M. (2015). Analyzing an information systems program skill-set:
A bottom-up validation of a top-down design. *Issues in Information Systems*, *16*(3), 80-89.

Resnick, M. (1997). *Turtles, termites, and traffic jams: Explorations in massively parallel microworlds.*
Cambridge: MIT Press.

Rob, M. (2006). Development of project documentation: Key ingredient in teaching systems analysis and design.
Issues in Information Systems, *7*(1), 83-87.

Rob, M., & Rob, F. (2016). A constructionist approach of blending SDLC with project-based learning. *Issues in Information Systems*, *17*(2), 115-122.

Rob, M., & Rob, F. (2018). Dilemma between Constructivism and Constructionism: Leading to the Development of a Teaching-Learning Framework for Student Engagement and Learning. *Journal of International Education in Business, 11*(2), 273-290.

Savery, J. R., & Duffy, T. M. (1995). Problem-based learning: An instructional model and its constructivist framework. In B. Wilson (Ed.), *Constructivist learning environments: Case studies in instructional design* (pp. 135–148). Englewood Cliffs, NJ: Educational Technology Publications.

Savery, J. (2015). Overview of problem-based learning: Definitions and distinctions. In A. Walker, H. Leary, C. E. Hmelo-Silver, & P. A. Ertmer (Eds.), *Essential readings in problem-based learning.* West Lafayette, Indiana: Purdue University Press.

Schuyler, S., & Skovira, R. (2007). Is the problematic in CS1 a student's problem solving ability? *Issues in Information Systems*, *8*(2), 112-119.

Smith, K., Estep, M., Zhao, C., Moinian, F., & Johari, A. (2014). Teaching case combined discipline capstone teams: Using service learning to provide a business solution. *Issues in Information Systems*, *15*(2), 8-13.

Soares, A., Fonseca, F., & Martin, N. (2015). Teaching introductory programming with game design and problem-based learning. *Issues in Information Systems*, *16*(3), 128-137.

Stolzoff, S. (2018). Some of the biggest recruiters of tech talent are from outside the tech sector. *Quartz at Work.* Retrieved from https://qz.com/work/1465979/jobs-in-tech-data-from-indeed-shows-whos-hiring-the-most/

Thomas, W. H., Rhodes, L. K., & Kruse, G. W. (2017). A capstone experience in information technology. In S.B. Fee, A. M. Holland-Minkley, & T. E. Lombardi (Eds.), *New Directions for Computing Education: Embedding Computing Across Disciplines* (pp.217-236). New York, NY: Springer.

Tobias, S., & Duffy, T. (Eds.). (2009). *Constructivist instruction: Success or failure?* New York, NY: Routledge.

Topi, H., Valachich, J., Wright, R.T., Kaiser, K.M., Nunamaker, J.F., & Vreede, G.J. (2010). *IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems*. Technical Report. ACM, New York, NY, USA.

Vest, S., Champan, D., & Denton, L. (2011). Studio-based learning in an introductory programming class: A collaborative study. *Issues in Information Systems*, *12*(2), 129-134.

Wang, W. (2015). Lessons learned from teaching project-based systems analysis and design. *Issues in Information Systems*, *16*(4), 132-142.

Wang, M. (2018). Real time data analytics and decision making in ERP game simulation: An across disciplinary approach. *Issues in Information Systems, 19*(4), 106-113.

Welch, O., Madison, T., & Welch, S. (2008). Is there working professionals' support for recommending students double major in accounting and information systems? *Issues in Information Systems*, *9*(1), 160-164.

Yerby, J., Hollifield, S., Kwak, M., & Floyd, K. (2014). Development of serious games for teaching digital forensics. *Issues in Information Systems*, *15*(2), 335-343.