Journal of Information Systems Education

Volume 31
Issue 4
Fall 2020

Agile Teaching and Learning in Information Systems Education: An Analysis and Categorization of Literature

Jason H. Sharp, Alanah Mitchell, and Guido Lang

Recommended Citation: Sharp, J. H., Mitchell, A., & Lang, G. (2020). Agile Teaching and Learning in Information Systems Education: An Analysis and Categorization of Literature. *Journal of Information Systems Education,* 31(4), 269-281.

Article Link: http://jise.org/Volume31/n4/JISEv31n4p269.html

Initial Submission:	30 April 2019
Accepted:	11 February 2020
Abstract Posted Online:	8 September 2020
Published:	10 December 2020

ISSN: 2574-3872 (Online) 1055-3096 (Print)

Agile Teaching and Learning in Information Systems Education: An Analysis and Categorization of Literature

Jason H. Sharp
Marketing and Computer Information Systems
Tarleton State University
Stephenville, TX 76402, USA
jsharp@tarleton.edu

Alanah Mitchell
Information Management and Business Analytics
Drake University
Des Moines, IA 50311, USA
alanah.mitchell@drake.edu

Guido Lang
Computer Information Systems
Quinnipiac University
Hamden, CT 06518, USA
guido.lang@quinnipiac.edu

ABSTRACT

In this paper, we analyze and categorize research related to Agile teaching and learning in Information Systems education using an existing conceptual framework. To this end, a systematic literature review beginning with 642 papers led to the identification of 30 relevant papers written in English and published through 2018 in academic IS outlets. Our analysis reveals three ways in which Information Systems educators incorporate Agile into their courses: 1) using Agile as a pedagogical approach to teach non-Agile content, 2) using Agile as a pedagogical approach to teach Agile content, and 3) using non-Agile pedagogical approaches to teach Agile content. The majority of relevant papers were published between 2016 and 2018. We present an analysis of the three instructional approaches to serve as a resource for interested individuals and recommend directions for future studies related to Agile teaching and learning in IS education.

Keywords: Agile, IS education, Literature review, Curriculum design & development, Pedagogy

1. INTRODUCTION

With roots in manufacturing, Agile software development, or Agile for short, continues to be a mainstay in software and systems development and is expanding into other business-related areas, such as project management and marketing. As such, it is imperative that Information Systems (IS) education keeps pace with the growth and implementation of Agile. With the historical influence of eXtreme Programming and the growing popularity and implementation of Scrum and Kanban, IS educators must ensure that the current and future generations of IS graduates possess the knowledge and skills necessary to be marketable, productive, and successful IS professionals in an increasingly Agile workplace. Since the publication of *The Manifesto for Agile Software Development* in 2001, a great deal of academic research has investigated a myriad of topics related to Agile in the professional software and systems development environment. Consequently, a significant body of literature exists in the areas of Computer Science (CS) and Software Engineering (SE) education regarding Agile in the classroom and continues to be a popular topic of research (Parsons and MacCallum, 2018). As Agile is also an area of great interest in IS education, there is a need on the part of IS educators and researchers to examine the way it has been implemented into the curriculum and classroom and glean insights into how to improve both.

Agile teaching and Agile learning embody the values, principles, and practices of Agile Software Development

methodologies as guided by the *Agile Manifesto*. Agile teaching is a "student-centered approach where learners work in teams and respond to rapid feedback" (Krehbiel et al., 2017, p. 93). Agile teaching emphasizes, "the continuity of the learning process, goal orientation, seeking feedback from students, flexibility in responding to student needs, a short feedback cycle, and demand-based personalization of what is being taught" (Razmov and Anderson, 2006, p. 2). Similarly, Agile learning "applies the processes and principles of Agile software development to the context of learning" (Lang, 2017, p. 14). A defining characteristic of Agile learning is the implementation of sprints which represent short iteration cycles of learning. Taken together, Agile teaching and Agile learning focus on three key characteristics: agility, extreme, and independence (Chun, 2004). According to Chun (2004), agility in teaching is necessary to handle "changing and diverse learning needs," while agility in learning is required to address "changing research, business, and technology environments" (p. 11). As such, the purpose of this study is to conduct a systematic review of Agile teaching and learning literature in IS education. Findings from the present study provide a succinct and timely resource for IS educators who teach Agile or use Agile as a teaching strategy and/or a course design approach.

This study operationalizes Agile teaching and learning from existing definitions in Agile literature (i.e., Chun, 2004; Razmov and Anderson, 2006; Krehbiel et al., 2017; Lang, 2017) to adhere to the values, principles, and practices of Agile software development within an educational context. Agile teaching and learning includes, but is not limited to, the teaching of Agile (i.e., a course on Scrum), the use of Agile as a teaching methodology (i.e., teaching Java using pair programming), and/or designing courses which may or may not teach Agile content using Agile pedagogies. IS educators who use Agile teaching and learning focus on the learning process among students rather than performance outcomes. Essentially, Agile teaching and learning offers IS educators a flexible, student-centered framework that incorporates continuous iterations and feedback cycles.

We organize the remainder of this paper into four sections. First, we provide thorough background information that traces the history of Agile as well as Agile in teaching and learning. Next, we describe the conceptual framework used to guide this study, followed by the specific methods used to collect and analyze data. Finally, we present the findings of this study and a general discussion of their significance to IS education.

2. BACKGROUND

2.1 History of Agile

The Agile movement gained widespread recognition in 2001 with the publication of *The Manifesto for Agile Software Development*, often referred to simply as the *Agile Manifesto*. Seventeen well-respected members of the software development community gathered in Snowbird, Utah, where they developed values and principles for guiding Agile (Ashmore and Runyan, 2015). The Agile Manifesto extols the following four values:

1. Individuals and interactions over processes and tools,
2. Working software over comprehensive documentation,
3. Customer collaboration over contract negotiation, and
4. Responding to change over following a plan (Beck et al., 2001).

The Agile Manifesto also includes 12 principles as presented in Table 1.

	Agile Manifesto Principles
1	Our highest priority is to *satisfy the customer* through early and continuous delivery of valuable software.
2	Business people and developers must *work together* daily throughout the project.
3	The most efficient and effective method of conveying information to and within a development team is *face-to-face conversation*.
4	Build projects around *motivated individuals*. Give them the environment and support they need, and trust them to get the job done.
5	The best architectures, requirements, and designs emerge from *self-organizing teams*.
6	At regular intervals, the *team reflects* on how to become more effective, then *tunes and adjusts* its behavior accordingly.
7	*Deliver working software frequently*, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
8	*Simplicity* – the art of maximizing the amount of work not done – is essential.
9	*Welcome changing requirements*, even late in development. Agile processes harness change for the customer's competitive advantage.
10	*Working software* is the primary measure of progress.
11	Continuous attention to *technical excellence* and *good design* enhances agility.
12	Agile processes promote *sustainable development*. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Table 1. Principles of The Agile Manifesto (Beck et al., 2001)

The creation of a number of well-known Agile approaches preceded the *Agile Manifesto*, including Dynamic Systems Development Method, Scrum, Feature-Driven Development, and Extreme Programming (XP). However, the publication of the *Agile Manifesto* marked the beginning of the current Agile movement in software development which has continued to grow within industry and education. Following the publication of the *Agile Manifesto*, additional Agile approaches, including Lean Software Development, Crystal, and Kanban, rose in popularity. Initially, XP experienced broad popularity, while more recently Scrum and Kanban are among the most often implemented Agile approaches. Even with the inclusion of Agile approaches in extant literature for more than 20 years, it is of particular interest that the publication of the first traditional textbook on Agile did not appear until 2015 (Ashmore and Runyan, 2015).

2.2 Agile in Teaching and Learning

The application of Agile into teaching and learning is not a new concept (e.g., Chun, 2004; Andersson and Bendix, 2005, 2006; Razmov and Anderson, 2006; Vuokko and Berg, 2007). In fact, there is early research related to CS and SE education shortly after the publication of *Extreme Programming Explained:*

Embrace Change (Beck, 1999) and the creation of the *Agile Manifesto* (Beck et al., 2001). A themed issue of *Computer Science Education* is a case in point. In Volume 12, Issue 3, 2002, six articles addressed the following topics: instructional issues related to Agile in academia (Hislop et al., 2002), teaching Agile alongside existing plan-driven approaches (Boehm, Port, and Brown, 2002), incorporation of pair programming (Tomayko, 2002; Williams et al., 2002) and eXtreme Programing practices in the classroom (Johnson and Caristi, 2002; Sanders, 2002).

IS education was not too far behind the other educational computing disciplines in addressing Agile within its literature. For example, Conn (2004) expounded on the implications of Agile for IS education, while other early IS education articles related to Agile were published shortly afterward (McAvoy and Sammon, 2005; McBride, 2005). The continued interest and application of Agile in IS education literature is further evidenced by a themed issue of *Journal of Information Systems Education* published in 2018 (Volume 29, Issue 2) that focused on Agile teaching and learning as well as other recent IS education-related journal article publications (Lang, 2017; Adkins and Tu, 2018; Frydenberg, Yates, and Kukesh, 2018).

At least as early as the mid-2000s, efforts to teach Agile beyond a course subject in and of itself emerged. These efforts employed Agile as a pedagogy apart from the actual course content, which may or may not be Agile-related. For example, the development of the Agile Teaching/Learning Methodology (ATLM) as a means to design courses by borrowing from the best practices of Agile appeared (Chun, 2004). Although applied primarily to technology-related courses, the creator argues, "the methodology itself is general enough to be applied to other disciplines as well" (Chun, 2004, p. 11). Specifically, ATLM suggests that both teaching and learning must be Agile in order to respond to the changes present in students, research, business, and technology.

The concept of Agile teaching emphasizes the use of Agile principles to design course content that can adapt to student needs and dynamic educational situations, ultimately providing a personalized and flexible instructional approach (Razmov and Anderson, 2006). Student engagement and frequent feedback are key factors to academic success, which are promoted by Agile teaching. The philosophy of Agile teaching "rests on actively soliciting student feedback and promptly reacting to it in order to increase the relevance of topic and advice directed toward the particular student (or group of students), with the ultimate goal of positively affecting student learning" (Razmov and Anderson, 2006, p. 2).

eXtreme Teaching (XT) is another concept developed based upon the values, principles, and practices of Agile, specifically in relation to XP. As such, Andersson and Bendix (2005) attempted to translate XP practices into teaching practices. Through this process, they discovered that while some practices translated well, others were a bit more difficult, and some were impossible. In a subsequent work, Andersson and Bendix (2006) extended the idea of XT to include six additional teaching practices. Andersson and Bendix (2005) suggest that the most extreme part of XT was the "extreme focus on students," and that "more student involvement and more dialogue between students and teachers will work wonders too in producing higher quality learning with less resources" (p. 39).

Agile SAD Teaching and Learning is another concept, which suggests systems analysis and design (SAD) courses should both teach Agile and use it as a method to teach the course itself (Pieters, 2013). The Agile principles of self-organization and self-direction may encourage and support students in their efforts to learn prerequisite material needed from other fields to inform the SAD process. The practice of short iterations lends itself nicely to comparing and contrasting different SAD methodologies such as structured, object-oriented, and Agile. Agile teaching and learning also lends itself to the changes or adaptations needed when selecting a SAD methodology. Another positive factor is the emphasis on instructor-student as well as student-student communication. The basic idea of this approach is to iterate through the teaching and learning elements as necessary to achieve desired results.

Borrowing from the *Agile Manifesto* (Beck et al., 2001), as well as other Agile education-related manifestos (e.g., Peha, 2011; Kamat, 2012; Royle and Nikolic, 2016), Krehbiel et al. (2017, p. 96) developed an *Agile Manifesto for Teaching and Learning* with the following values:

1. Adaptability over prescriptive teaching methods
2. Collaboration over individual accomplishment
3. Achievement of learning outcomes over student testing and assessment
4. Student-driven inquiry over classroom lecturing
5. Demonstration and application over accumulation of information
6. Continuous improvement over the maintenance of current practices

Krehbiel et al. (2017) report on the implementation of Agile-based instructional methods and assignments across a broad range of disciplines including computer science and software engineering, information systems, supply chain management, English, teacher education, civic studies, and political science. Krehbiel et al. make a significant contribution by expanding the idea of Agile teaching and learning outside of the traditional computer science, software engineering, and information systems disciplines.

Finally, Agile learning is a pedagogical approach focused on learning rather than instruction (Lang, 2017). Lang defines Agile learning as "the application of the processes and principles of agile software development to the context of learning" (p. 5). Using sprints, a common Agile practice, students produced a functioning deliverable by participating in the cycle of planning, designing, building, testing, reviewing, and launching. Based on findings from this study, Lang reports a strong preference for Agile learning among students while indicating that learning style did not influence preference nor performance. Challenges of designing and implementing Agile learning include: 1) a significant amount of time is needed on the part of the instructor for planning, 2) striking a balance between instructing students on how to do something with providing explanations on why to do it, and 3) a significant amount of time is needed for one-on-one student support from the instructor.

This background information illustrates the various methodologies and findings in previous research on teaching and learning in relation to Agile. Some of the articles offer support for IS educators teaching Agile, while others offer

support for IS educators using Agile as a teaching method and/or course redesign approach.

3. CONCEPTUAL FRAMEWORK

This study uses the conceptual framework for integrating Agile in teaching and learning proposed by Sharp and Lang (2018). According to the framework, previous research on Agile in teaching and learning falls into two broad categories: 1) studies related to teaching about Agile software development concepts and/or how to use them and 2) studies focused on the integration of pedagogical approaches or interventions based upon the principles of Agile without necessarily teaching about and/or how to use Agile itself. These categories may be differentiated in terms of the content (i.e., what is being taught) and the pedagogy (i.e., how it is being taught). The framework can be depicted as a matrix: The horizontal axis creates two categories designated for content, and the vertical axis creates two categories designated for pedagogy (see Figure 1). Within each category, the degree of Agile that an instructor incorporates into course content or pedagogy may be determined.

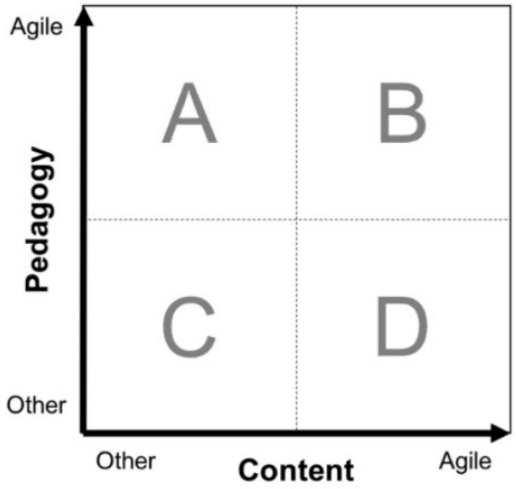


Figure 1. Conceptual Framework for Integrating Agile in Teaching and Learning (Sharp and Lang, 2018)

In knowing that the degree of Agile may vary in terms of content and pedagogy for a given curriculum, course, lesson, or exercise, Figure 1 can be understood with the following examples: 1) Teaching an Agile software development course in a traditional lecture format falls into Quadrant D (Other Pedagogy, Agile Content); 2) Teaching the same Agile software development course incorporating an Agile method, such as Scrum, as an instructional method falls within Quadrant B (Agile Pedagogy, Agile Content); 3) Teaching a cybersecurity course based on cases falls within Quadrant C (Other Pedagogy, Other Content); and 4) Teaching the same cybersecurity course incorporating an Agile practice, such as pair programming as an instructional strategy, falls within Quadrant A (Agile Pedagogy, Other Content).

In this conceptual framework, the borders between Agile and other content, as well as between Agile and other pedagogy, are fluid. An instructor should consider individual student learning needs and their own preferences to determine the degree of Agile content and pedagogical features to implement in a course. For example, an instructor may choose to cover Agile in only parts of a course or lesson. Similarly, an instructor may choose to incorporate "iterative development approaches or reflection journals without fully committing to a complete Agile pedagogy" (Sharp and Lang, 2018, p. 46). Ultimately, the conceptual framework provides a tool for researchers and educators to determine the extent of incorporation of Agile content and pedagogy into a single course. By conducting such an analysis, researchers and educators are equipped to make productive decisions concerning the use of Agile in teaching and learning (Sharp and Lang, 2018). Although serving as the source of the conceptual framework for the study, the Sharp and Lang (2018) paper itself was excluded from review because it did not meet the established inclusion criteria as it was not a research study on the use of Agile as a pedagogical approach nor of a class teaching Agile either using Agile as a pedagogical approach or another pedagogical approach.

4. METHODOLOGY

In this study, we used an existing structured approach for assessing prior research (Mitchell and Zigurs, 2009), consisting of five steps: 1) identifying the concepts of interest, 2) identifying prior research to be included, 3) coding prior research related to the concepts of interest, 4) analyzing the coding results, and 5) identifying the areas for future research. The following sections provide further information about the actions taken in each step.

First, we identified Agile teaching and learning as the concept of interest. Agile teaching and learning derives its definition from prior literature as characterized by flexibility, student-centered, feedback-orientated, communication-based, and iteration-driven (Chun, 2004; Razmov and Anderson, 2006; Krehbiel et al., 2017; Lang, 2017). It may include the teaching of Agile and the use of Agile as a teaching methodology and/or course-design approach, and it may focus on the learning process rather than the instructional method.

Next, we identified prior research to be included. In order to achieve the purpose of this study, the search strategy focused on traditionally recognized IS education publication outlets, rather than traditional CS or SE publication outlets (e.g., ACM Digital Library, IEEE Xplore). Furthermore, we excluded any theory-based papers, abstracts, research-in-progress manuscripts, and/or workshop presentations. Using EBSCOhost, we consulted the following databases: Academic Search Complete, Business Source Complete, Computer Source, Computers and Applied Sciences Complete, Education Source, E-Journals, and ERIC. During database searches, we used available search delimiters to narrow queries. In the Academic Search Complete database, we limited our search to include only peer-reviewed publications written in English and published before December 31, 2018. For the other databases, we limited our search to include only publications written in English. We conducted multiple searches in each database using the following search term combinations: 1) "agile" and "information systems" and "education," 2) "agile" and "information systems" and "teaching," and 3) "agile" and "information systems" and "learning." This search of databases yielded 201 total papers. After the exclusion of exact duplicates, 106 resulted (see Appendix 1 for a full explanation of the search results). While an argument exist that limiting search terms causes the omission of potentially relevant literature (Boell and

Cecez-Kecmanovic, 2016), it is a common practice within information systems research to establish specific search parameters when investigating a focused phenomenon (e.g., Gregory et al., 2015; Dressen and Diegmann, 2017; Diegmann et al., 2018).

In addition to the EBSCOhost database searches, we conducted a search of the AIS eLibrary (AISeL) using the same search term combinations outlined previously. When possible, we conducted searches of the Senior Scholars Basket of Eight within AISeL. If not possible, we used built-in search features available on the official websites of the publications. The AISeL and website searches returned 392 total papers.

Finally, we conducted keyword and/or article title searches with the search term "agile" using the built-in search features available on official websites of relevant IS education-related publications. The sites are as follows: *Computers & Education*, *EDSIG Conference on Information Systems and Computing Education (EDSIGCON)*, *Information Systems Education Conference (ISECON)*, Information Systems Education Journal (ISEDJ), *Informing Science Institute (ISI)* library, *International Association for Computer Information Systems (IACIS) Conference*, *Journal of Computer Information Systems (JCIS)*, and *Journal of Information Systems Education (JISE)*. The search efforts produced 144 unique papers after the removal of duplicates (see Table 2).

Source	Total	Total w/ Duplicates Removed
Computers & Education	30	30
EDSIGCON	14	4
ISECON	8	4
ISEDJ	14	11
ISI library	30	23
IACIS Conference	10	10
JCIS	61	61
JISE	15	1
Total	**182**	**144**

Table 2. IS Education Publication Search Result Counts

Taken together, all searches produced 642 papers. Among these papers, we conducted cursory reviews of article titles to extract papers that were relevant to Agile teaching and learning in IS courses at the college level. This first review of papers yielded 95 papers. Of these papers, we conducted a second review of article titles and a detailed review of article abstracts. This second review of papers yielded 67 papers.

Among these 67 papers, we established a systematic, two-phase review process to determine which papers were relevant papers for inclusion in the study. As mentioned above, only research papers exploring our concept of interest (i.e., Agile teaching and learning) were identified as relevant and any off-topic or theory-based papers, abstracts, research-in-progress manuscripts and/or workshop presentations, papers presenting curriculum ideas (but not testing them), and papers simply using the word "Agile" were excluded. In the first phase, we performed independent reviews of each paper and issued one of the following votes related to its relevance: yes, no, or unsure. A relevant paper marked for inclusion received two "yes" votes and zero "no" votes. A paper receiving two "no" votes and zero "yes" votes received no further consideration. At the end of the first phase, we identified 30 relevant papers (see Appendix 2) and 23 excluded papers based on a review of title and abstract. In the second phase, we performed another review of the 14 remaining papers. This detailed review of each paper resulted in the exclusion of 12 papers. For the two remaining papers, we held a virtual discussion to discuss the merits of the papers and ultimately decided that both papers were not suited for inclusion in our analysis. Table 3 provides a summary of publication name and counts.

Publication Name	Count
Communications of the Association for Information Systems	1
Information Systems Education Journal	7
International Journal of Information & Communication Technology Education	1
Issues in Informing Science and Information Technology	1
Journal of Information Systems Education	11
Journal of the Midwest Association for Information Systems	1
Proceedings of the AIS SIGED IAIM Conference	1
Proceedings of the Americas Conference on Information Systems	3
Proceedings of the Australasian Conference on Information Systems	1
Proceedings of the EDSIG Conference on Information Systems & Computing Education	2
Proceedings of the Informing Science and IT Education Conference	1
Total	**30**

Table 3. Count by Publication Name

Once we identified the 30 relevant papers for inclusion in this study (see Appendix 2), each of us performed an independent evaluation of each paper to determine its placement in the conceptual framework. Each of us coded the papers as follows: 1) Quadrant A: Agile Pedagogy AND Other Content, 2) Quadrant B: Agile Pedagogy AND Agile Content, 3) Quadrant C: Other Pedagogy AND Other Content, or 4) Quadrant D: Other Pedagogy AND Agile Content. We met after the first round of coding to compare our results, which demonstrated complete consensus for the categorization of five papers: four papers in Quadrant A and one paper in Quadrant B. Of the remaining 25 papers, at least two of us categorized 22 papers to the same quadrant, while 3 papers had no consensus at all. Instead of using majority rule, we performed a second independent review of the 25 papers and engaged in virtual discussions regarding the merits of each paper until we reached a complete consensus.

The remaining sections present the results of the structured approach for assessing prior research used in this research. Specifically, the following sections present the results and analysis of our coding as well as a discussion of our findings and the identification of areas for further research.

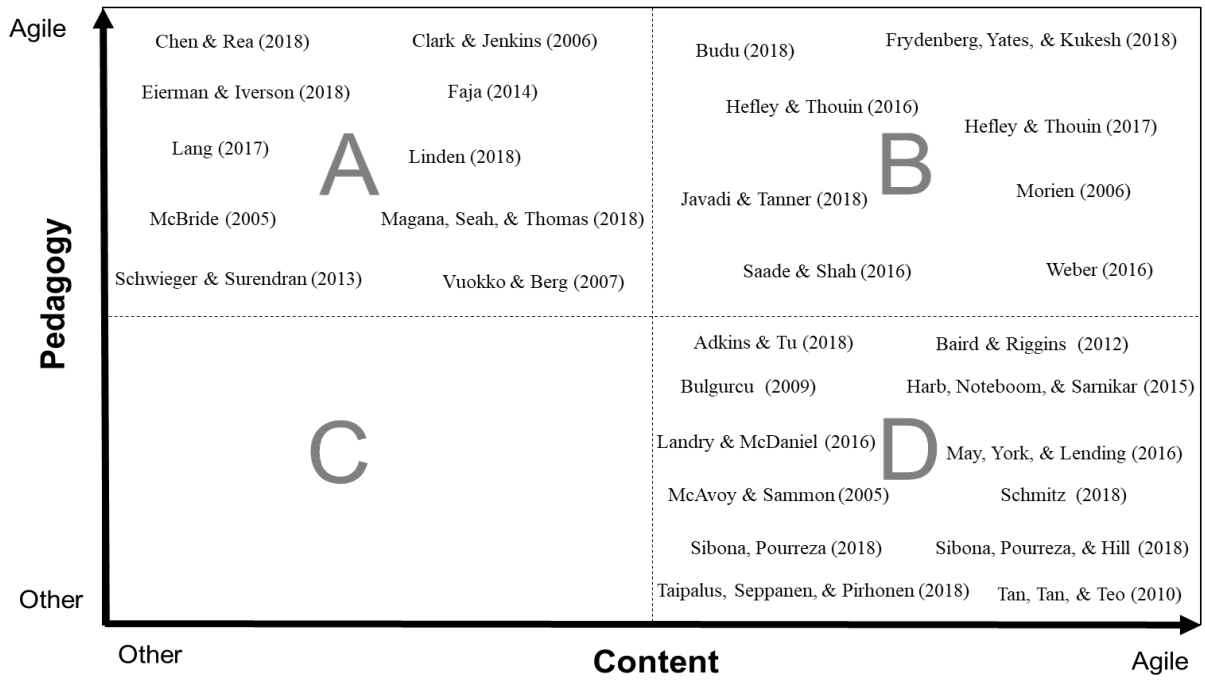


Figure 2. Categorized Papers within the Conceptual Framework

5. RESULTS

At the conclusion of the coding process, we categorized 10 relevant papers in Quadrant A, 8 relevant papers in Quadrant B, and 12 relevant papers in Quadrant D. We did not categorize any of the papers in Quadrant C, as these papers would not have shown up in our search. Figure 2 provides a summary of the categorization of each paper into the three quadrants. The placement of the papers in each quadrant reflects alphabetical order of name of author(s) rather than placement based upon degree of Agile or Other in relation to Pedagogy and Content.

It should be noted that we categorized three of the papers (Schmitz, 2018; Sibona, Pourreza, and Hill, 2018; Taipalus, Seppänen and Pirhonen, 2018) into a different quadrant (Quadrant D) when compared to the Sharp and Lang (2018) paper (Quadrant B). While this appears to create a discrepancy in the validity of our approach, we would argue just the opposite. The fact that papers might be re-categorized after further, careful review actually strengthens the validity of the approach. In regard to these three papers, the first round of review by the authors resulted in two authors categorizing these papers into Quadrant D, while one author categorized them into Quadrant B. A second round of review of these three papers was conducted individually by each author, followed by discussion among the authors. Through this discussion, consensus was reached among all the authors. This iterative process, in our estimation, helped to ensure that each paper was thoughtfully and appropriately categorized.

5.1 Quadrant A: Agile Pedagogy, Other Content

The relevant papers categorized in this quadrant used Agile as a pedagogical approach in IS courses during the teaching of non-Agile content. More and more, Agile is influencing areas outside software development, namely teaching and learning (Parsons and MacCallum, 2018; Salza, Musmarra, and Ferrucci, 2018). Given the popularity of XP, it is not surprising that six of the relevant papers in this quadrant utilized elements of its principles and/or practices. Among these relevant papers, McBride (2005) applied the values of XP in an e-Commerce course as an "analogy for the structuring of the content and process through which learning takes place" (p. 75) while Vuokko and Berg (2007) implemented a methodology based upon XP (i.e., eXtreme Teaching) into a course entitled Implementation of Information Systems in Organizational Context. Additionally, four relevant papers incorporated pair programming, a popular XP practice, as a pedagogical approach in IS courses (Clark and Jenkins, 2006; Faja, 2014; Chen and Rea, 2018; Eierman and Iversen, 2018).

Within this quadrant, two relevant papers incorporated Scrum, a popular Agile methodology, as an instructional approach in IS courses. Linden (2018) investigated the use of Scrum in an introductory programming course, while Magana, Seah, and Thomas (2018) investigated its use in a course for systems analysis and design. In addition, Lang (2017) examined the use of Agile learning, which incorporates short development cycles commonly referred to as sprints, in a web development course. The final relevant paper in this quadrant applied the test-first principle to redesign a course in information technology management (Schwieger and Surendran, 2013). Table 4 summarizes the relevant papers in Quadrant A.

Author(s)	Year of Publication	Agile Pedagogical Approaches	Other Content
Chen and Rea	2018	Pair programming	Introduction to business computing; Business application programming; Business analytics I; Business data mining
Clark and Jenkins	2006	Pair programming	Systems design
Eierman and Iversen	2018	Pair programming	C# programming
Faja	2014	Pair programming	Introductory programming
Lang	2017	Sprints	Web development
Linden	2018	Scrum	Introductory programming
Magana, Seah, and Thomas	2018	Scrum	Systems analysis and design
McBride	2005	XP	E-Commerce
Schwieger and Surendran	2013	Test-first principle	Information technology management
Vuokko and Berg	2007	eXtreme Teaching	Implementation of information systems in organizational context

Table 4. Relevant Papers Categorized in Quadrant A: Agile Pedagogy, Other Content

5.2 Quadrant B: Agile Pedagogy, Agile Content

It is becoming increasingly common in IS education to not only teach Agile content, but also to use Agile as the method for teaching the content. As such, the eight relevant papers categorized in Quadrant B represent the use of Agile pedagogy to teach Agile content. Several of the relevant papers revealed a natural fit between Scrum and the actual teaching of Agile project management. For example, Saade and Shah (2016) sought to use Agile as "an activity-based approach" to teach Agile project management (p. 96). Consequently, the authors not only designed and structured the course around Agile in terms of the activities, but also taught students about Agile and how to implement it within the context of a peer group task. Other studies implement this combination of Agile Pedagogy-Agile Content at both the individual and group levels using a simulation-based Agile project management exercise (i.e., Fissure SimProject Agile Simulation) in a graduate-level Agile project management course (Hefley and Thouin, 2016, 2017).

In another relevant paper, Javadi and Tanner (2018) developed an Agile teaching framework incorporating elements of Scrum and Kanban to teach these actual project management methodologies within the context of IT project management courses. Budu (2018) used the values and principles delineated in the Agile Manifesto to design Agile-guided activities. As a pedagogical approach, Agile is not limited to IT project management courses but is suitable for introductory and advanced IS courses as well. For example, Frydenberg, Yates, and Kukesh (2018) applied Scrum processes and roles to teach students enrolled in an introductory computing concepts course the differences between traditional and Agile methodologies through techniques as simple as creating paper airplanes. On the other end of the curriculum, Morien (2006) and Weber (2016) used Agile pedagogy to teach students enrolled in advanced IS courses Agile content that may be applied to capstone projects and other more advanced IS topics, such as systems analysis and design and web/mobile programming. Table 5 summarizes the relevant papers in Quadrant B.

5.3 Quadrant D: Other Pedagogy, Agile Content

With the current emphasis in higher education on student engagement with real-world experiences, instructors are employing more active-learning approaches into the design of their courses. Instructors who use active-learning course designs promote student engagement with course material and provide students opportunities to practice using the essential knowledge and skills needed for their future work environments. With this in mind, the relevant papers categorized in Quadrant D describe a variety of non-Agile ways in which instructors teach Agile content. Due to the applied nature of IT project management, systems analysis and design,

Author(s)	Year of Publication	Agile Pedagogical Approaches	Agile Content
Budu	2018	Agile Manifesto values and principles	IT project management
Frydenberg, Yates, and Kukesh	2018	Agile principles	Computing concepts
Hefley and Thouin	2016	Scrum	Agile project management
Hefley and Thouin	2017	Scrum	Agile project management
Javadi and Tanner	2018	Scrum and Kanban	IT project management
Morien	2006	Agile Adaptive Development Approach	System development
Saade and Shah	2016	Agile learning activity	Agile project management
Weber	2016	Agile methodology	Systems analysis and design Web and mobile programming

Table 5. Relevant Papers Categorized in Quadrant B: Agile Pedagogy, Agile Content

IS development, capstone experiences, and project-based courses, it seems logical that instructors would rely upon active-learning approaches. By doing so, instructors of these types of courses have the ability to teach Agile approaches alongside other traditional IS approaches and make comparisons among approaches (Hoskey and Hoskey, 2016; Luce, 2016). Considering the natural alignment of Scrum within these topic areas, it is not surprising to see its inclusion in this quadrant.

As noted by Prince (2004), problem-based learning (PBL) is a well-established, active learning approach and is quite suitable within IS education. Consequently, two of the relevant papers in Quadrant D studied the principles and practices of PBL alongside Scrum within the context of a capstone course (Adkins and Tu, 2018) and an Agile systems development course (Taipalus, Seppänen, and Pirhonen, 2018). Within the realm of IT project management, two of the relevant papers investigated a hybrid, project-based approach to teach both traditional and Agile project management using Scrum (Baird and Riggins, 2012; Landry and McDaniel, 2016). Additionally, two of the relevant papers examined the use of Origami as an active-learning strategy for teaching the principles of Scrum (Sibona and Pourrezza, 2018; Sibona, Pourrezza, and Hill, 2018). Another studied the use of the "Ball Game" to supplement the teaching of Scrum within a systems analysis and design course (May, York, and Lending, 2016), while another explored a role-play simulation for presenting Agile project management (Schmitz, 2018).

Among all of these relevant papers, the overarching goal of each instructor was to expose students to Agile through engaging and interactive approaches. Within this quadrant, three relevant papers described other non-Agile approaches to assist students with selecting the appropriate Agile software development methodology to implement, which were the facilitation of workshops (Harb, Noteboom, and Sarnikar, 2015) and use of a teaching case and case study (McAvoy and Sammon, 2005; Bulgurcu, 2009). Lastly, one relevant paper described how to design an IS development project course using the system approach model, which was based upon the Dick, Carey, and Gagne model of instructional design (Tan, Tan, and Teo, 2010). Table 6 summarizes the papers in Quadrant D.

5.4 Publication Trends

As demonstrated by the increasing number of relevant papers published, particularly from the year 2016 forward (see Figure 3), there is undoubtedly a growing interest of Agile teaching and learning in IS education. It is notable that the first relevant papers related to Agile teaching and learning in IS education appeared soon after publication of the *Agile Manifesto* (i.e., McAvoy and Sammon, 2005; Clark and Jenkins, 2006; Morien, 2006). However, this initial interest experienced a lull of nine years during which publications dropped to zero or one relevant papers annually. Beginning in 2016, Agile teaching and learning in IS education began to gain traction again among IS education researchers, as demonstrated by the 19 relevant papers published over the next two years. We found this research drought puzzling, as the adoption of Agile methods and practices in the workplace experienced continuous, rapid growth since 2005 (VersionOne, 2007, 2018). Moreover, Agile persisted as a robust topic in CS and SE education and research (e.g., Parsons and MacCallum, 2018). With these trends in mind, it is not obvious why there was a lack of research interest in Agile teaching and learning in IS education between 2007 and 2016. This result warrants further investigation.

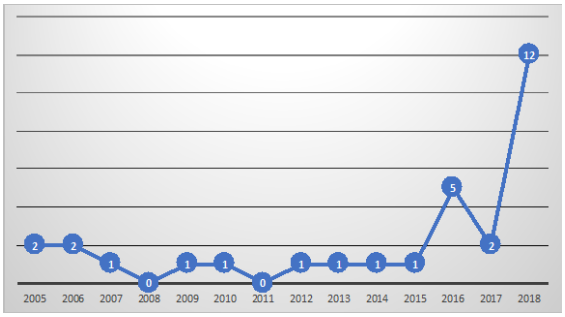


Figure 3. Number of Agile Teaching and Learning Publications by Year

6. DISCUSSION AND CONCLUSION

The overwhelming predominance of Agile methods and practices in the workplace (VersionOne, 2018) has substantiated a need for the renewed focus on Agile teaching and learning in IS education to maintain momentum. With this in mind, the goal of this study was to provide a concise, relevant, and timely resource for IS educators who strive to prepare their students in accordance to current industry

Author (s)	Year of Publication	Other Pedagogical Approaches	Agile Content
Adkins and Tu	2018	Problem-based Learning	Scrum
Baird and Riggins	2012	Project-based	Agile Project Management
Bulgurcu	2009	Case-study	IS development using XP
Harb, Noteboom, and Sarnikar	2015	Teaching case	Selection of Agile Software Development Methodology
Landry and McDaniel	2016	Project-based	Agile project management
May, York, and Lending	2016	Ball game	Scrum
McAvoy and Sammon	2005	Critical adoption workshops	Agile methodologies
Schmitz	2018	Role-play simulation	Agile Project Management
Sibona and Pourreza	2018	Active learning and lecturing	Scrum project management
Sibona, Pourreza, and Hill	2018	Origami active learning activity	Scrum project management
Taipalus, Seppänen, and Pirhonen	2018	Problem-Based Learning	Agile Systems Development
Tan, Tan, and Teo	2010	System approach model	IS development project

Table 6. Relevant Papers Categorized in Quadrant D: Other Pedagogy, Agile Content

standards. We realized this goal by conducting a systematic literature review to narrow down available literature and uncover relevant papers related to Agile teaching and learning in IS education. We further enhanced the helpfulness of this resource by using a conceptual framework to categorize relevant papers according to the degree in which Agile content and pedagogy were incorporated in specific IS courses. As a result, IS educators are empowered to make evidence-based decisions about the instructional design of their own courses.

In addition to achieving the goal of this study, we also noted a secondary finding. At the onset of this study, a conceptual framework defined four types of instructional designs that IS educators may use to promote student understanding of Agile. Since one of these instructional designs did not address Agile for content or pedagogical purposes, we only considered the following three instructional designs for the categorization of relevant papers: 1) use of Agile as a pedagogical approach to teach non-Agile content, 2) use of Agile as a pedagogical approach to teach Agile content, and 3) use of non-Agile pedagogical approaches to teach Agile content. While slightly more papers fell into the latter category, the distribution of papers across instructional designs is relatively even. Closer inspection of the relevant papers revealed that more than half were published between the years 2016 and 2018. It is not clear why this aspect of Agile teaching and learning in IS education has been studied more extensively in recent years or what the implications of this secondary finding may be.

Finally, in an effort to address the final step of the structured method for assessing prior research followed in this study, we pose the following questions as opportunities for expanded research related to Agile teaching and learning in IS education: What factors influence an IS educator's selection of an instructional design in an IS course? What are the levels of efficacy IS educators have with each instructional design? Furthermore, we feel it is essential to explore Agile teaching and learning practices in IS education beyond a single course. Future studies should also conduct programmatic analyses to better understand how a sequence of courses, within a specific program of study, prepares students for their future work contexts. Future work might also apply the matrix from this study to the broader concept of systems development pedagogy.

While findings from this study have provided new insights for an under-researched area, there are limitations that require acknowledgment. A common limitation of systematic literature studies is the methods used to locate relevant publications (Boell and Cecez-Kecmanovic, 2016; Gough, Oliver, and Thomas, 2017). A number of factors may affect the conduction of an exhaustive search for relevant publications. These may be within the control of the researcher (e.g., choice of search terms, selection of databases) or beyond (e.g., database indexing of journals). Systematic literature reviews require use of "explicit, accountable rigorous research methods" (Gough, Oliver, and Thomas, 2017, p. 4), so we employed an existing structured approach as the methodology in this study (Mitchell and Zigurs, 2009). While this approach offered a standardized protocol with which to search IS education literature, it may have not captured all relevant papers. Future studies should address this limitation by: (a) including non-English studies and leveraging translation resources; (b) carrying out a catch-up search to identify research published during intervening periods; and (c) searching sources that are not indexed in electronic databases, such as books, book chapters, conference proceedings, dissertations, theses, unpublished research reports, and other non-journal sources.

7. REFERENCES

Adkins, J. & Tu, C. (2018). Applying an Agile Approach in an Information Systems Capstone Course. *Proceedings of the Fourth EDSIG Conference on Information Systems and Computing Education*, 4(4622), 1-9.

Andersson, R. & Bendix, L. (2005). eXtreme Teaching. *Proceedings of 3:e Pedagogiska Inspirationskonferensen*.

Andersson, R. & Bendix, L. (2006). Towards a Set of eXtreme Teaching Practices. *Proceedings of the Fifth Koli Calling Conference on Computer Science Education*, 33-40.

Ashmore, S. & Runyan, K. (2015). *Introduction to Agile Methods*. Upper Saddle River, New Jersey: Addison-Wesley.

Baird, A. & Riggins, F. J. (2012). Planning and Sprinting: Use of a Hybrid Project Management Methodology within a CIS Capstone Course. *Journal of Information Systems Education*, 23(3), 243-257.

Beck, K. (1999). *Extreme Programming Explained: Embrace Change*. Boston, Massachusetts: Addison-Wesley.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *The Manifesto for Agile Software Development*. Retrieved February 14, 2020, from http://agilemanifesto.org/.

Boehm, B., Port, D., & Brown, A. W. (2002). Balancing Plan-Driven and Agile Methods in Software Engineering Project Courses. *Computer Science Education*, 12(3), 187-195.

Boell, S. K. & Cecez-Kecmanovic, D. (2016). On Being 'Systematic' in Literature Reviews. In L. P. Willcocks, C. Sauer, & M. C. Lacity (Eds.), *Formulating Research Methods for Information Systems* (pp. 48-78). New York, New York: Palgrave Macmillian.

Budu, J. (2018). Applying Agile Principles in Teaching Undergraduate Information Technology Project Management. *International Journal of Information and Communication Technology Education*, 14(3), 29-40.

Bulgurcu, B. (2009). E-Learning for Software Engineering: A Case Study on Teaching Information Systems Online Group Project with Extreme Programming. *Proceedings of the AIS SIGED IAIM Conference*, 29, 1-10.

Chen, K. & Rea, A. (2018). Do Pair Programming Approaches Transcend Coding? Measuring Agile Attitudes in Diverse Information Systems Courses. *Journal of Information Systems Education*, 29(2), 53-64.

Clark, S. R. & Jenkins, A. (2006). Paired Programming in a Clayton's Capstone Project Course. *Proceedings of the Australasian Conference on Information Systems*, Article 62.

Conn, S. S. (2004). A New Teaching Paradigm in Information Systems Education: An Investigation and Report on the Origins, Significance, and Efficacy of the Agile Development Movement. *Information Systems Education Journal*, 2(15), 1-19.

Chun, A. H. W. (2004). The Agile Teaching/Learning Methodology and its e-Learning Platform. In W. Liu, Y. Shi, Q. Li (Eds.), *Lecture Notes in Computer Science – Advances in Web-Based Learning* (pp. 11-18). Heidelberg, Germany: Springer-Verlag.

Diegmann, P., Dreesen, T., Binzer, B., & Rosenkranz, C. (2018). Journey Towards Agility: Three Decades of Research on Agile Information Systems Development. *Proceedings of the Thirty Ninth International Conference on Information Systems*, 1-17.

Dressen, T. & Diegmann, P. (2017). Journey Towards Agility: Where are We Now and Where are We Heading? *Proceedings of the Thirty Eighth International Conference on Information Systems*, 1-11.

Eierman, M. A. & Iversen, J. (2018). Comparing Test-Driven Development and Pair Programming to Improve the Learning of Programming Languages. *Journal of the Midwest Association for Information Systems*, 4(1), 23-36.

Faja, S. (2014). Evaluating Effectiveness of Pair Programming as a Teaching Tool in Programming Courses. *Information Systems Education Journal*, 12(6), 36-45.

Frydenberg, M., Yates, D., & Kukesh, J. S. (2018). Sprint, then Fly: Teaching Agile Methodologies with Paper Airplanes. *Information Systems Education Journal*, 16(5), 22-36.

Gough, D., Oliver, S., & Thomas, J. (2017). *An Introduction to Systematic Reviews, 2nd ed.*. Thousand Oaks, California: SAGE Publications.

Gregory P., Barroca L., Taylor K., Salah D., & Sharp, H. (2015). Agile Challenges in Practice: A Thematic Analysis. In C. Lassenius, T. Dingsøyr, & M. Paasivaara (Eds.), *Agile Processes in Software Engineering and Extreme Programming*, *XP 2015, Lecture Notes in Business Information Processing* (pp. 64-80). Cham, Switzerland: Springer.

Harb, Y. A., Noteboom, C., & Sarnikar, S. (2015). Evaluating Project Characteristics for Selecting the Best-fit Agile Software Development Methodology: A Teaching Case. *Journal of the Midwest Association for Information Systems*, 1(1), 33-52.

Hefley, W. E. & Thouin, M. F. (2016). Evaluation of Experiential Learning in Scrum and Agile Project Management. *Proceedings of the Twenty-second Americas Conference on Information Systems*, 1-5.

Hefley, W. E. & Thouin, M. F. (2017). Evaluating Individual and Group Experiential Learning in Scrum and Agile Project Management. *Proceedings of the Twenty-third Americas Conference on Information Systems*, 1-5.

Hislop, G. W., Lutz, M. J., Naveda, F., McCraken, W. M., Mead, N. R., & Williams, L. A. (2002). Integrating Agile Practices into Software Engineering Courses. *Computer Science Education*, 12(3), 169-185.

Hoskey, C. & Hoskey, A. (2016). Cultivating Sprightly Students: Using Agile Development in an Information Systems Capstone Course. *Proceedings of the Information Systems Education Conference*, 33, 88-99.

Javadi, E. & Tanner, S. (2018). Design and Implementation of an Agile Teaching Framework. *Proceedings of the Twenty-fourth Americas Conference on Information Systems*, 1-10.

Johnson, D. H. & Caristi, J. (2002). Using Extreme Programming in the Software Design Course. *Computer Science Education*, 12(3), 223-234.

Kamat, V. (2012). Agile Manifesto in Higher Education. *Proceedings of the Fourth IEEE International Conference on Technology for Education*, 231-232.

Krehbiel, T. C., Salzarulo, P. A., Cosmah, M. L., Forren, J., Gannod, G., Havelka, D., Hulshult, A. R., & Merhout, J. (2017). Agile Manifesto for Teaching and Learning. *The Journal of Effective Teaching*, 18(2), 90-111.

Landry, J. P. & McDaniel, R. (2016). Agile Preparation within a Traditional Project Management Course. *Information Systems Education Journal*, 14(6), 27-33.

Lang, G. (2017). Agile Learning: Sprinting Through the Semester. *Information Systems Education Journal*, 15(3), 14-21.

Linden, T. (2018). Scrum-Based Learning Environment: Fostering Self-Regulated Learning. *Journal of Information Systems Education*, 29(2), 65-74.

Luce, T. (2016). Adding Agility to the MIS Capstone. *Issues in Information Systems*, 17(1), 119-127.

Magana, A. J., Seah, Y. Y., & Thomas, P. (2018). Fostering Cooperative Learning with Scrum in a Semi-Capstone Systems Analysis and Design Course. *Journal of Information Systems Education*, 29(2), 75-92.

May, J., York, J., & Lending, D. (2016). Play Ball: Bringing Scrum into the Classroom. *Journal of Information Systems Education*, 27(2), 87-92.

McAvoy, J. & Sammon, D. (2005). Agile Methodology Adoption Decisions: An Innovative Approach to Teaching and Learning. *Journal of Information Systems Education*, 16(4), 409-420.

McBride, N. K. (2005). A Student-Driven Approach to Teaching E-commerce. *Journal of Information Systems Education*, 16(1), 75-83.

Mitchell, A. & Zigurs, I. (2009). Trust in Virtual Teams: Solved or Still a Mystery? *Data Base for Advances in Information Systems,* 40(3), 61-83.

Morien, R. I. (2006). Student Experience of Using Agile Development Methods in Industrial Experience Projects. *Information Systems Education Journal*, 4(103), 3-12.

Parsons, D. & MacCallum, K. (Eds.). (2018). *Agile and Lean Concepts for Teaching and Learning: Bringing Methodologies from Industry to the Classroom*. Singapore: Springer.

Peha, S. (2011). Agile Schools: How Technology Saves Education (Just Not the Way We Thought it Would). *InfoQ*. Retrieved February 14, 2020, from http://www.infoq.com/articles/agile-schools-education.

Pieters, L. (2013). An Agile Approach to Systems Analysis and Design Teaching and Learning. *Proceedings of the Nineteenth Americas Conference on Information Systems*, 1-8.

Prince, M. (2004). Does Active Learning Work? A Review of the Research. *Journal of Engineering Education*, 93(3), 223-231.

Razmov, V. & Anderson, R. (2006). Experiences with Agile Teaching in Project-Based Courses. *Proceedings of the ASEE Annual Conference and Exposition.*

Royle, K. & Nikolic, J. (2016). Agile Work Practices for Learning and Teaching: What We Can Learn from Agile Work Practices about Learning and Teaching in Schools. Retrieved February 14, 2020, from https://www.researchgate.net/publication/303961839_What_we_can_learn_from_Agile_Work_Practices_about_Learning_and_Teaching_in_Schools.

Saade, R. G. & Shah, S (2016). Exploring an Agile Learning Activity to Teach Agile Project Management. *Proceedings of the Informing Science and IT Education Conference (InSITE)*, 95-101.

Salza, P., Musmarra, P., & Ferrucci, F. (2018). Agile Methodologies in Education: A Review. In D. Parsons & K. MacCallum (Eds.), *Agile and Lean Concepts for Teaching and Learning: Bringing Methodologies from Industry to the Classroom* (pp. 25-45). Singapore: Springer.

Sanders, D. (2002). Extreme Programming: The Student View. *Computer Science Education*, 12(3), 235-250.

Schmitz, K. (2018). A Three Cohort Study of Role-Play Instruction for Agile Project Management. *Journal of Information Systems Education*, 29(2), 93-104.

Schwieger, D. & Surendran, K. (2013). Information Technology Management: Course Re-Design Using an Assessment Driven Approach. *Information Systems Education Journal*, 11(2), 23-35.

Sharp, J. H. & Lang, G. (2018). Agile in Teaching and Learning: Conceptual Framework and Research Agenda. *Journal of Information Systems Education*, 29(2), 45-52.

Sibona, C. & Pourreza, S. (2018). The Impact of Teaching Approaches and Ordering on IT Project Management: Active Learning vs. Lecturing. *Information Systems Education Journal*, 16(5), 66-77.

Sibona, C., Pourreza, S., & Hill, S. (2018). Origami: An Active Learning Exercise for Scrum Project Management. *Journal of Information Systems Education*, 29(2), 105-116.

Taipalus, T., Seppänen, V., & Pirhonen, M. (2018). Coping with Uncertainty in an Agile Systems Development Course. *Journal of Information Systems Education*, 29(2), 117-126.

Tan, C., Tan, W., & Teo, H. (2010). Designing an Information Systems Development Course to Incorporate Agility, Flexibility, and Adaptability. *Communications of the Association for Information Systems*, 26(10), 171-194.

Tomayko, J. E. (2002). A Comparison of Pair Programming to Inspections for Software Defect Reduction. *Computer Science Education*, 12(3), 213-222.

VersionOne. (2007). *Survey: "The State of Agile Development."* Retrieved February 14, 2020, from https://explore.versionone.com/state-of-agile/1st-annual-state-of-agile-report-2.

VersionOne. (2018). *The 12thAannual State of Agile Report*. Retrieved February 14, 2020, from https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report.

Vuokko, R. & Berg, P. (2007). Experimenting with eXtreme Teaching Method – Assessing Students' and Teachers' Experiences. *Issues in Informing Science and Information Technology*, 4, 523-534.

Weber, E. V. (2016). Performance Learning of Agile Methodology Using Paired Courses of Systems Analysis and Design and Web/Mobile Programming. *Proceedings of the EDSIG Conference on Information Systems and Computing Education*, 1-9.

Williams, L., Wiebe, E., Yang, K., Ferzli, M., & Miller, C. (2002). In Support of Pair Programming in the Introductory Computer Science Course. *Computer Science Education*, 12(3), 197-212.

AUTHOR BIOGRAPHIES

Jason H. Sharp is a professor of computer information systems at Tarleton State University. His Ph.D. is from the University of North Texas. Prior to entering academia, he worked as a systems support specialist and developed custom database solutions for small business. His research focuses on systems and software development methods, instructional design and technology, and computing education. His work appears in *Journal of Information Systems Education*, *Information Systems Education Journal*, *Journal of Information Technology Education: Innovations in Practice*, and *The DATA BASE for Advances in Information Systems*, as well as in the proceedings of numerous international and national conferences.

Alanah Mitchell is an associate professor of information systems in the department of information management and business analytics in the College of Business and Public Administration at Drake University. Her Ph.D. is in information technology from the University of Nebraska at Omaha. Mitchell's research focuses on global virtual projects, human-computer interaction, and IS pedagogy. She has published in such journals as *Journal of the Association for Information Systems*, *Communications of the Association for Information Systems*, *DATA BASE, International Journal of e-Collaboration*, and *Journal of Information Systems Education*, as well as others.

Guido Lang is an associate professor of computer information systems at Quinnipiac University. He is an academic entrepreneur and has founded successful companies in the information technology, publishing, and healthcare sectors. He has published dozens of articles in the areas of computing education and educational technology which have won awards from the Education Special Interest Group of the Association for Information Technology Professionals and appeared in journals such as the *Journal of Computer Information Systems*, *Journal of Information Systems Education*, and *Journal of Research on Technology in Education*. He holds a Ph.D. in information systems from City University of New York.

Appendix 1: Detailed Results of Search Strategy

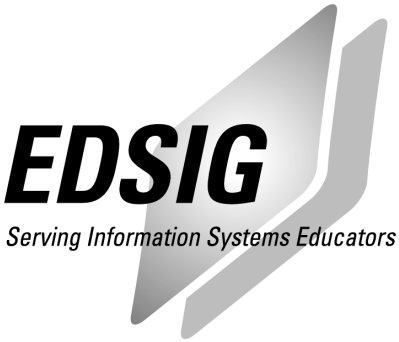
Source	Search term(s)	Limiters	Total #/Duplicates removed/	# Selected
EBSCOhost			202/106	36
Academic Search Complete	agile AND information systems AND education OR agile AND information systems AND teaching OR agile AND information systems AND learning	Scholarly (Peer Reviewed) Journals through 12/31/2018 English		
Business Source Complete	Same as above	English		
Computer Source	Same as above	English		
Computers & Applied Sciences Complete	Same as above	English		
Education Source	Same as above	English		
E-Journals	Same as above	English		
ERIC	Same as above	English		
AIS eLibrary	agile AND information systems AND education OR agile AND information systems AND teaching OR agile AND information systems AND learning		392/392	24
Informing Science Institute	Agile		30/23	4
Journal of Computer Information Systems	agile AND information systems AND education OR agile AND information systems AND teaching OR agile AND information systems AND learning		61/61	7
Computers & Education	Agile		30/30	1
IACIS Proceedings	Agile		10/10	6
*JISE	Agile	Keyword (15) Title (10)	15/10/1	1
ISEDJ	Agile	Keyword Title	15/11/8	8
EDSIGCON	Agile	Keyword Title	14/8/4	4
ISECON	Agile	Keyword Title	8/6/4	4
Totals			**777/657/639**	**95**

Of the **639** non-duplicated papers returned from the searches, on review, **95** were selected.
Of the **95** papers, a second round of review resulted in the inclusion of **67** papers.
Of the **67** papers, a third round of review resulted in **30** papers included in the study.

Appendix 2: Relevant Papers

Quadrant A: Agile Pedagogy, Other Content
Chen & Rea (2018)
Clark & Jenkins (2006)
Eierman & Iversen (2018)
Faja (2014)
Lang (2017)
Linden (2018)
Magana, Seah, & Thomas (2018)
McBride (2005)
Schwieger & Surendran (2013)
Vuokko & Berg (2007)
Quadrant B: Agile Pedagogy, Agile Content
Budu (2018)
Frydenberg, Yates, & Kukesh (2018)
Hefley & Thouin (2016)
Hefley & Thouin (2017)
Javadi & Tanner (2018)
Morien (2006)
Saade & Shah (2016)
Weber (2016)
Quadrant D: Agile Content, Other Pedagogy
Adkins & Tu (2018)
Baird & Riggins (2012)
Bulgurcu (2009)
Harb, Noteboom, & Sarnikar (2015)
Landry & McDaniel (2016)
May, York, & Lending (2016)
McAvoy & Sammon (2005)
Schmitz (2018)
Sibona & Pourreza (2018)
Sibona, Pourreza, & Hill (2018)
Taipalus, Seppänen, & Pirhonen (2018)
Tan, Tan, & Teo (2010)

Information Systems & Computing
Academic Professionals



STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 2574-3872