Journal of the Midwest Association for Information Systems

Volume 2025 | Issue 2 Article 2

Date: 07-01-2025

Using Scrum for Managing Group Projects in a Face-to-Face Information Systems Project Management Course

Vlad Krotov
Murray State University, vkrotov@murraystate.edu

Pitzel Krotova
Murray State University, mkrotova@murraystate.edu

Abstract

Scrum has become the most widely used Agile project management methodology, and educational institutions around the globe are integrating Scrum into their curricula. Using Scrum in class can potentially help educators not only equip students with hands-on knowledge of the Scrum framework and Agile values, but also teach students valuable soft skills and foster positive group work dynamics. This article provides a detailed account of how Scrum has been adapted to manage group projects in an on-campus graduate Information Systems course devoted to Project Management. The adaptation is deliberately simple and "low-tech" to keep administrative overhead low and to appeal to a broad audience of educators. An online survey was conducted among the students to assess the usefulness of the Scrum adaptation. While certain issues were raised, the feedback was mostly positive. The survey results indicate that implementing the framework may help students acquire practical knowledge of Scrum, Agile, and project management, and enhance positive group work dynamics.

Keywords: Scrum, Agile, Education, Adaptation, Project Management

DOI: 10.17705/3jmwa.000094

1. Introduction

Scrum is arguably the most widely used Agile Project Management methodology in the Information Technology (IT) field. The 17 "State of Agile Report" published in 2023 by VersionOne reveals that 71% of organizations surveyed practice Agile development methodologies; 63% use Scrum for managing their projects (VersionOne, 2023). The popularity of Scrum as alternative to the traditional, "waterfall" approach to managing projects is not surprising: Scrum is "a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value." (Sutherland & Schwaber, 2013, p.3). Scrum works exceptionally well for projects with evolving or unclear requirements, high uncertainty, and the need for frequent stakeholder feedback (Highsmith & Cockburn, 2001). It is most effective when used by small, cross-functional, self-organizing teams in environments that support adaptability, rapid delivery of value, and continuous improvement (Moe et al., 2010; Rigby et al., 2016).

Given the popularity of Scrum as a project management framework, practical knowledge of this Agile methodology is a valuable skill for IT and general business professionals in all industries. Because of that, educational institutions around the globe are integrating Scrum into their curricula (Matkovic et al., 2016; Reynolds et al., 2023). The main goal of this article is to assist educators from various fields in integrating Scrum and Agile into their curricula by providing a detailed account of how Scrum has been adapted for managing group projects involving the creation of detailed project management plans in a graduate, on-campus Information Systems course devoted to Project Management. The implementation of Scrum described in this article is deliberately simple and "low-tech" to appeal to educators from various fields, including non-technical ones. This is also in alignment with the spirit of Scrum, since the methodology does not prescribe any specific tools or technologies (although team members are free to use them as needed) (Sutherland & Schwaber, 2013). Additional Scrum tools, techniques, and artifacts can be added to this customization of Scum—depending on the nature and the goals of the class where it is used.

The rest of this article is structured as follows. First, it provides a brief overview of the Scrum methodology and explains how and why Scrum is increasingly used in education. This section highlights the benefits of using Scrum in the educational process and the need to adapt the framework—originally developed for software projects—to the educational context. Next, the article offers a detailed account of how Scrum was implemented to manage group work in a project management course based on the traditional "waterfall" approach. The unique features and benefits of this adaptation, along with the instructor's insights and recommendations, are discussed as part of this implementation. The article concludes with an empirical evaluation of the effectiveness of the approach, presenting both quantitative and qualitative feedback from students, followed by recommendations for improvement and final reflections on the use of Scrum in education.

2. Using Scrum in Class

2.1 The Scrum Framework

In comparison to traditional, "waterfall" project management methodologies, the Scrum framework is deliberately "light" (see Figure 1). The framework is based on simple, generative rules that empower people and give them an opportunity to show their best performance. For example, the framework does not prescribe any technologies or tools; it is up to the team to decide which technologies and tools should be used to deliver a product. As often mentioned by Jeff Sutherland, one of the creators of Scrum, in his public speeches, the Scrum framework is easy to understand, but often takes a lifetime to master. In most cases, effective use of Scrum is a result of extensive experience and tacit knowledge rather than intellectual understanding of this framework (Sutherland & Schwaber, 2013).

Scrum emphasizes empirical process control, which means that project-related decisions are based on observation and feedback rather than rigid planning. This allows for flexibility and adaptability throughout the project. The framework encourages regular reflection and improvement through retrospective meetings held after each Sprint, where the team discusses what went well, what could be improved, and decides on actions for the next iteration.

Overall, Scrum provides a framework that enables teams to deliver high-quality products in a collaborative and efficient manner, with a focus on continuous learning and adaptation.

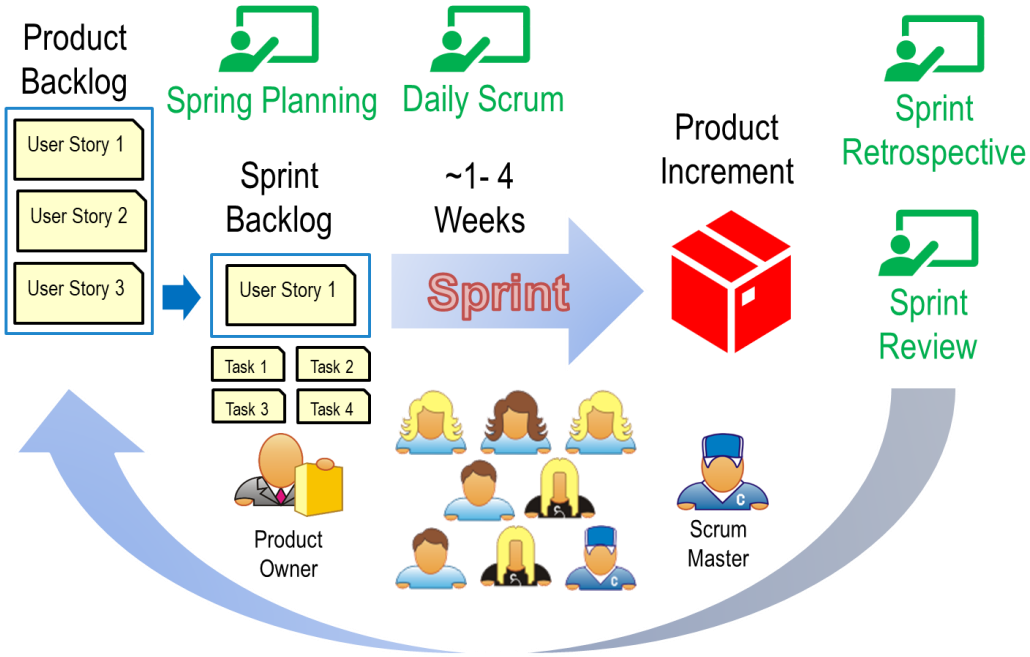

Figure 1. The Scrum Framework (Adapted from Mountain Goat Software, 2005)

Scrum begins by assembling a Scrum Team consisting of six to nine members (ScrumEductor.org, 2023). This team includes a Scrum Master, responsible for facilitating teamwork, removing obstacles, and ensuring adherence to Scrum values and processes; a Product Owner, who represents the customer and has final authority over product requirements; and a Development Team, composed of individuals with the technical and non-technical skills needed to deliver the product.

Once the team is formed, they create a Product Backlog, a list of tasks or User Stories that describe product features from the customer's perspective (e.g., "As a traveler, I want to browse different travel insurance options available on the website"). The Product Backlog captures the full scope of the product's functionality and must be prioritized. The backlog remains flexible, allowing for modifications and reprioritization between Sprints.

After establishing the Product Backlog, the team selects a set of prioritized items to work on during a Sprint—a time-boxed development cycle lasting approximately 2-4 weeks. The Scrum Team commits not only to implementing these selected features but also to delivering a shippable product increment, meaning a functional and demonstrable portion of the product.

Throughout each Sprint, the team holds Daily Scrums, which are short, 15-minute stand-up meetings. Each team member answers three key questions:

1. What did you do yesterday?
2. What will you do today?
3. Are there any obstacles in your way?

These meetings are not meant for problem-solving but rather for team members to stay aligned and commit to their tasks in front of their peers. Problem-solving discussions take place outside the Daily Scrum, helping eliminate unnecessary meetings.

At the end of each Sprint, the Scrum Team conducts a Sprint Review, where they present their completed work and demonstrate new product features. These meetings involve the entire team and should be open to anyone in the organization who wants to attend.

Additionally, the team holds Sprint Retrospective meetings between Sprints. The goal of these sessions is to continuously improve the team's workflow by reflecting on what worked well, what didn't, and what can be improved. While these retrospectives can be conducted after every Sprint, they are especially valuable when the team identifies areas needing refinement.

2.2 The Benefits of Scrum in Higher Education

While Scrum is often associated with computer science education, the framework is useful in a variety of educational contexts, such as nursing and education (Zahorodko, 2023). Research indicates that Scrum can enhance the learning experiences of students in relation to teamwork and diverse subjects by providing structured frameworks for collaboration and communication involving students and their instructors (Reynolds et al., 2023).

One of the key advantages for using Scrum in higher education is to foster student collaboration and engagement. With the help of Scrum, students can engage not only with the instructor and other team members, but also—in the case of client-sponsored projects—with external stakeholders. Students can take on roles that mirror real-world responsibilities, thus enhancing their learning experience through service learning and preparing themselves for the industry (Dong, 2023). Scrum also allows students to get practical insights into project management and teamwork. Similarly, the use of Scrum in project-based courses has been shown to improve student performance and satisfaction, as it encourages team self-regulation and constant reflection on the learning progress and the quality of the deliverable produced as a result of this project (Vogelzang et al., 2021; Fernandes et al., 2021).

The use of Scrum for managing student group projects also addresses challenges related to team dynamics and group project management in higher education. The role of the Product Owner, which can be assumed by the instructor teaching the course, can allow the instructor to guide student projects while also giving students the freedom and responsibility to self-organize their work (Baham, 2020). By establishing clear roles and responsibilities via formal roles, Scrum helps to streamline communication and coordination among team members, which is essential for successful project outcomes (Wolff, 2024). Additionally, the structured nature of Scrum ceremonies, such as Daily Scrums and Sprint Reviews, promotes student accountability and continuous feedback from the instructor, which are important for student learning (Pope-Ruark, 2012; Morales-Trujillo et al., 2021).

Furthermore, the use of Scrum in remote learning environments has gained traction as well (Wolff, 2024). Research indicates that Scrum can enhance communication and collaboration among geographically distributed teams, leading to improved project management and learning outcomes (Wolff, 2024). This adaptability is crucial in today's educational landscape, where remote learning is becoming increasingly prevalent.

2.3 The Main Uses of Scrum

There are many ways in which Scrum is used in the educational context (Sharp et al, 2020). Some instructors use the so-called "Scrum games" as fun, team building activities in their classes that also equip students with hands-on understanding of Scrum and Agile principles (e.g., May et al., 2016). Scrum is also used to deliver educational content in various courses in a student-driven, self-organized, and self-paced manner (Rush & Connolly, 2020). One of the most popular uses of Scrum in class is for managing student group projects (e.g., Wagh, 2012; Manamendra et al., 2013; Zorzo, & Lucredio, 2013; Nejmeh & Weaver, 2014; Mahalakshmi & Sundararajan, 2015).

2.4 The Need for Scrum Adaptation

One of the challenges of using Scum in education is the fact that the framework was originally created for managing software development teams within real organizations. Some of the processes prescribed by Scrum cannot be implemented in class. For example, an instructor cannot have Daily Scrums with students, as most classes meet only 1-2 times per week. Thus, in order to make the framework useful in education, Scrum needs to be adapted to the educational context.

Several adaptations of Agile and Scrum to the educational context have been published in academic and practitioner outlets. Below, we discuss some of these implementations and highlight how the adaptation of Scrum described in this paper differs from these educational adaptations and adds unique value to learning.

First, there are conceptual papers that provide the foundation for "agile teaching." For example, Krehbiel et al. (2017) present an "Agile Manifesto for Teaching and Learning," which reimagines the Agile Manifesto's principles within the educational context. The authors argue for learner-centered, adaptive, and feedback-driven approaches to pedagogy, paralleling the values of Agile software development. While not offering specific Scrum adaptation instructions, this conceptual paper offers a foundational rationale for applying Agile principles in the classroom and includes some guidance on how Scrum can be adopted in class.

Second, there are studies detailing various approaches to using Scrum in the classroom. For example, Rush & Connolly (2020) provide a fairly detailed description of how Scrum was used to deliver educational content related to project management in an Agile fashion. Baham (2020) examines how to improve Business Product Owner (BPO) engagement in student Scrum projects, emphasizing such practices as co-creation workshops and formalized role assignments to enhance stakeholder involvement and project success. Babik (2022) offers a detailed account of a hands-on "Scrum Boot Camp" that builds student confidence in Scrum principles and tools through active learning in both face-to-face and online courses. Thouin and Hefley (2024) explain how they use an experiential simulation based on SimAgile to teach Scrum Product Owner skills, finding that students gain authentic insights into product management and decision-making processes from the simulation exercise. Finally, Mutchler et al. (2024) demonstrate how Scrum can be adapted for concept-heavy courses—specifically Information Security Management—by allowing students to tailor deliverables to their interests, resulting in higher engagement and perceived skill development.

There are also proprietary Scrum-based educational methodologies developed by organizations such as eduScrum (eduScrum, 2020) or ScrumEducator (ScrumEducator.org, 2023), as well as literature reviews of various Scrum implementations by Salza et al. (2019), López-Alcarria et al. (2019), and Reynolds et al. (2023). Collectively, these rarticles communicate general trends in Scrum adaptation and underscore the adaptability of Scrum across diverse course types, student populations, and learning objectives.

We believe that the Scrum adaptation described in this paper provides a unique, valuable, and practical contribution to the literature on the use of Scrum in education. The adaptation includes some unique and valuable features that were not found in previously published adaptations of Scrum. Some of these unique features and value propositions are discussed below.

First, the article provides a detailed and rich description of how Scrum can be adopted in the classroom. These rich descriptions are supplemented with the instructor's own experiences and insights. Many of the reviewed studies lack specific, practical details on implementing Scrum in class.

Second, although various technological tools can be added based on the needs and goals of the course, the Scrum adaptation described here is deliberately simple and "low-tech." It emphasizes tangible tools like physical Scrum boards and burndown charts, making implementation accessible to instructors who do not have access to technical resources or who prefer not to invest time and money into learning new tools.

Third, the emphasis of this adaptation is on the translation and implementation of Scrum in an Information Systems course devoted to the "waterfall" approach to Project Management. The project managed via Scrum involves the creation of traditional project management artifacts, such as a project charter, business case, and quality assurance plan. This integration of Agile practices with "waterfall" material allows the instructor to "kill two birds with one stone": (1) familiarize students with the "waterfall" approach to project management via Team-Based Learning (TBL), and (2) equip students with practical, hands-on skills related to Scrum and Agile.

Fourth, this adaptation focuses on quality face-to-face interactions among students and the instructor. We believe that face-to-face experiences provide students with rich opportunities to learn from their peers and the instructor. The adaptation emphasizes the development of teamwork and other soft skills through these interactions, which is grounded in the principles of Team-Based Learning (TBL).

Fifth, this paper proposes integrating Scrum not as a short workshop, activity, or assignment, but as a full-scale, long-term, semester-long exercise. While the Scrum framework is relatively simple and easy to learn, it often takes a long time to master (Sutherland & Schwaber, 2013). This is largely due to the fact that Scrum is largely about doing (tacit knowledge) rather than

intellectual understanding Scrum roles, processes, and artifacts (explicit knowledge). Thus, providing students with ample hands-on opportunities to gain working knowledge of Scrum and Agile processes is essential to learning these important topics.

Finally, the adaptation includes both quantitative and qualitative evaluation of its effectiveness as perceived by students. This feedback allows readers to gain additional insights into how the implementation of Scrum can be improved further or tailored to their unique educational goals or contexts.

3. Scrum Adaptation

The Scrum framework described in Sutherland & Schwaber (2013) has been adapted to manage student group projects in a required, on campus, graduate IT Project Management course offered at a North American university. The group project involves the creation of a detailed IT project management plan for an organization. The Scrum framework has been implemented in six phases (see Figure 2). The main elements (i.e., artifacts, processes, roles, and rules) of this implementation are outlined in Table 1. The names of these elements are capitalized and italicized throughout the text. Details of the implementation processes (e.g., the nature of the graduate course, group project description, detailed description of the phases, necessary supplies, artifacts, processes, roles, rules, some instructor reflections, etc.) are provided in the sections below.

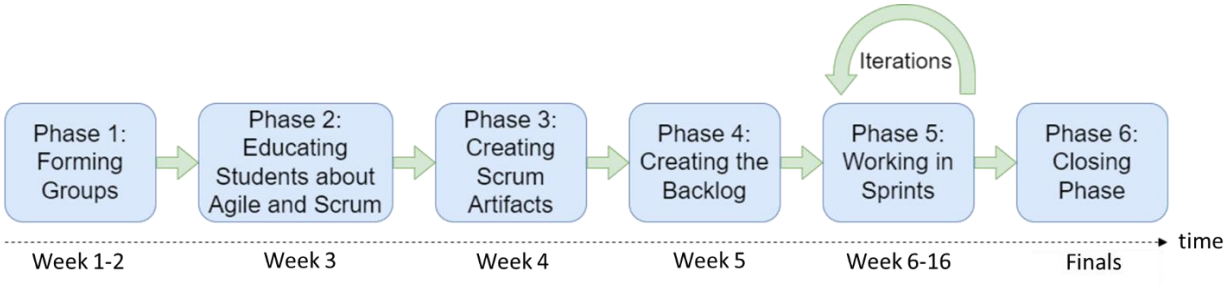

Figure 2. Scrum Implementation Phases (ScrumEductor.org, 2023)

3.1 Information Technology Project Management Course

The Scrum framework has been implemented in a required, graduate, on-campus IT Project Management course. The course is offered by the College of Business at a North American regional university during the spring and fall semesters that span fifteen weeks each. The enrollment in each section is usually close around 20-30 students. This course uses the textbook by Kathy Schwalbe titled "Information Technology Project Management" (6-8th edition). While the textbook touches on the Agile approach to project management and contains brief descriptions of some of the most popular Agile methodologies (e.g., Scrum and Extreme Programming), the textbook adopts a rather traditional, "waterfall" view on project management.

The IT Project Management course is a required course for graduate students majoring in Computer Information Systems (CIS) and Cybersecurity Management (CM). The CIS program is offered by the College of Business. The CM program is offered jointly by the College of Business and the College of Engineering. Other business majors can take this course as an elective, but they rarely do that due to class capacity issues and a false belief that this course may involve programming.

3.2 Group Project Assignment: IT Project Management Plan

The Scrum framework has been used to manage group work involving the creation of a detailed IT project management plan. This group project requires students to create a detailed project management plan involving implementation of an information system in an organization of their choice. Students majoring in CM usually chose to work on a plan involving implementing various networking elements, while CIS majors usually work on project plans involving software development. The project management plan is comprised of 17 Deliverables (see Table 2). The order in which the Deliverables are arranged reflects the dependencies among them.

The Story Points attached to each of the Deliverables are estimated by the instructor. These points reflect the relative workload and grading weight associated with each of the Deliverables. The Final Report involves compiling and summarizing Deliverables 1-15 (see Table 2). The Final Presentation involves presenting the Final Report to the entire class. Many of the Deliverables overlap (i.e., have shared parts). Thus, by the nature of the project, students are forced to work on the plan in an iterative fashion.

Table 1. Main Elements of the Scrum Implementation (ScrumEducator.org, 2023)

Element	Description
Scrum Board	A physical board containing: (1) Group Name; (2) Class Code; (3) *To Do* (also known as the *Backlog*), *Doing*, and *Done* columns. The board is populated with "sticky notes" containing information about *Deliverables*.
Deliverable	A *Deliverable* is a document comprising the group project. A template is provided to students for each *Deliverable*. Students are free to modify these templates. Each *Deliverable* is described using a "sticky note". At the minimum, each note should contain: (1) the name of the *Deliverable*; (2) the value of the *Deliverable* in *Story Points*; and (3) the name of the person responsible for the *Deliverable*. Small "sticky notes" can be used to add additional information about the *Deliverables*.
Story Points	*Story Points* are scale-free estimates of the relative workload required to complete a particular *Deliverable*. The story points are also used to "weigh" grades associated with individual *Deliverables*.
Burndown Chart	A *Burndown Chart* shows team work velocity and progress over time. The line on the chart starts at 60 points - the total value of all *Deliverables* comprising the project. When a *Deliverable* is completed (see the *Definition of Done*), the line "goes down". The slope of the line gives information about the "velocity" of the group work, meaning how productive the team is in completing the work. A "flat line" may suggest that not much work is being done and students may miss the final deadlines. A "steep line" with a negative slope may suggest good progress.
Backlog	This is the *To Do* column on the *Scrum Board*. Contains all the group project *Deliverables* that need to be deliverable and the overall plan for completing the project created by each *Group*.
Scrum Master	A person conducting *Weekly Scrums* and making sure each group member has all the tools and information to be productive. For the first few *Sprints*, the role can be performed by the instructor, but will eventually be transferred to a leader of each of the groups.
Product Owner	This is the instructor teaching the course who explains to students what needs to be done for each of the *Deliverables*, provides feedback on the submitted *Deliverables*, and assigns grades.
Group	A permanent team comprised of 4-6 students working together on a group project for the duration of the entire semester.
Sprint	A period of time (usually 1-2 weeks) during which students work on a particular portion of the group project. By default, each *Sprint* is one week long, although it can be longer if a weekly session is cancelled due to holidays or exams.
Weekly Scrum	An in-class group work session comprised of two meetings. The first meeting is among the *Group* members. During this meeting, the *Group* members discuss what has been done and what needs to be done for the next *Sprint*. Based on this discussion, the *Scrum Board* and the *Burndown Chart* are updated. Other issues can be discussed during the *Group* meeting. The second meeting is a stand-up meeting between the group and the instructor. During these meetings students report to the instructor on what has (or has not) been done, the issues they are facing, and the items they commit to work on for the next Sprint.
Definition of Done	A *Deliverable* is done when it is submitted via the Learning Management System (LMS) for grading; only then the *Scrum Board* and the *Burndown Chart* can be updated.
Peer Evaluation	An online survey asking students to provide quantitative and qualitative evaluation of their own contribution to group work together with the contribution of each of their group members. This feedback is used to assign the peer-evaluation grade component for each of the students.

The instructor does not set any deadlines for Deliverables 1-15. But students are informed that (1) they will make their Final Presentations on the last day of the class; (2) the Final Report will be due on the last day of the semester as per the official Academic Calendar of the university; (3) the last day to submit their anonymous online Peer Evaluations (another major grading component for the group project) is the date of the final exam. Thus, delaying the completion of Deliverables 1-15 may delay the final two Deliverables (16-17).

Table 2. Group Project Deliverables (ScrumEducator.org, 2023)

Deliverable	Story Points
1. Team Contract	2
2. Project Organization Chart	1
3. Project Charter	3
4. Business Case	4
5. Statement of Work	4
6. Scope Statement	2
7. Work Breakdown Structure	3
8. Gantt Chart	3
9. Network Diagram	4
10. Cost Estimate	3
11. Stakeholder Registry	1
12. Stakeholder Management Strategy	2
13. List of Risks	1
14. Communication Plan	3
15. Quality Assurance Plan	4
16. Final Presentation	10
17. Final Report	10
Total:	60

3.3 Scrum Implementation Phases

The framework has been implemented for managing group work on the IT Project Management Plan in several phases: (1) Forming Groups; (2) Educating Students about Agile and Scrum; (3) Creating Scrum Artifacts; (4) Creating the Backlog; (5) Working in Sprints (an iterative phase); and (6) Closing Phase. Each of these phases is described in detail below (ScrumEducator.org, 2023).

Phase 1: Forming Groups

The first week of the course is used by the instructor to introduce himself and the course and to cover Chapter 1 from the textbook. During Week 1, students are also briefly informed about the nature of the group project and asked to form groups of 4-6 people. To speed up the group formation, the instructor requires students to do an in-class group case analysis during the second week of the course. The deadline for forming groups is the end of Week 3 of the course. Once students finalized their group membership, group information is entered by the instructor in Canvas – the Learning Management System (LMS) used in this course. This allows the instructor to create collaboration areas for students in Canvas and to assign group project assignments to groups of students.

Phase 2: Educating Students about Agile and Scrum

By the end of Week 3, the instructor covers the first three chapters of the textbook. These chapters give students a fairly good introduction into the field of Project Management. Moreover, Chapter 3 contains an overview of Agile methodologies used in Project Management, including Scrum. This gives the instructor an opportunity to educate students about Scrum and explain how Scrum will be used for managing their group project work. First, the instructor directs students' attention to the "The Scrum Guide" – a 16-page document outlining the essence of the framework (Sutherland & Schwaber, 2013). The instructor tells the students to read the guide by the end of Week 3 as this material will be needed for the group project and will be covered by the midterm exam. In addition, there is a YouTube video tutorial explaining the essence of Scrum and how it is used in software development in under 10 minutes (Axosoft, 2014). This video is shown in class. The content of this video seems to be sufficient to help students understand the most fundamental aspects of the Scrum framework (even if students did not read "The Scrum

Guide"). After showing the video, the instructor explains how Scrum will be adapted and used in class. Basically, Phases 3-6 described further in this section are presented to students. The instructor deliberately does not use any formal manual containing the rules and processes of Scrum implementation in this particular class to allow for room for discussion and innovation.

Phase 3: Creating Scrum Artifacts

During Week 4, the instructor asks students to create all the necessary Scrum artifacts: Scrum Board, Burndown Chart, and "sticky notes" for the Deliverables. This requires acquiring certain supplies prior to this exercise (see Table 3). While the instructor has experimented with several electronic tools and artifacts for implementing Scrum, such as Trello Boards (see Figure 3), the instructor gives preference to physical artifacts (e.g., physical Scrum Board and physical Burndown Chart made from poster boards) to create opportunities for quality, face-to-face interactions in class.

Table 3. Supplies Needed

Item	Quantity	Additional Notes
Poster Boards	1-2 per team	These poster boards will be used for creating *Scrum Boards* – one per team. If an instructor wishes to have a separate board for a *Burndown Chart*, then additional poster boards (one per team) should be acquired. Preferably, the board used for the *Burndown Chart* should contain a square grid. This will make plotting easier.
Markers	1 pack per team	At the minimum, the instructor should have one marker per team. Ideally, each team should have access to a set of markers of various colors. Students usually like to use several colors for creating their *Scrum Boards* and *Burndown Charts*.
Large "Sticky Notes"	1 stack per team	"Sticky notes" (e.g. Post-it® Notes) are to be used by students to capture the information about the *Deliverables* (or "user stories") comprising their *Backlog*. At the minimum, these notes should contain: (1) the name of the *Deliverable*; (2) the value in terms of *Story Points* (can be indicated by putting thick dots using a marker; and (3) the person in charge.
Small "Sticky Notes"	1 stack per team	Small "sticky notes" can be used by students to add additional information to their "user stories" (captured with the help of large "sticky notes").
Mobile Tripods	1 per team	Mobile tripods are used for mounting *Scrum Boards* and *Burndown Charts* during "stand-up" meetings between team members and the instructor.
Rulers	1 per team	Long rulers (depending on the size of the poster boards used) to assist students with drawing lines on their *Scrum Boards* and *Burndown Charts*.
Pencils and Erasers	1 per team	Pencils are used by students to sketch layouts of their *Scrum Boards* and *Burndown Charts*.
Storage Box	1	A box for storing rulers, "sticky notes", markers, pencils, etc. The instructor can give out these supplies during *Weekly Scrum* sessions and put them away once group work is over.
Folders	1 per team	These folders are used for storing physical copies of templates for each of the *Deliverables*. This allows students to have something in front of them when they discuss the details of each of the Deliverables.

To encourage experimentation and innovation, the instructor deliberately gives students only rough guidelines on how these artifacts should look like (e.g., by showing some examples from previous courses). Over the years, students have created several versions of Scum Boards and Burndown Charts of the course of four semesters. One representative example of a Scrum Board and Burndown Chart is provided in Figure 4.

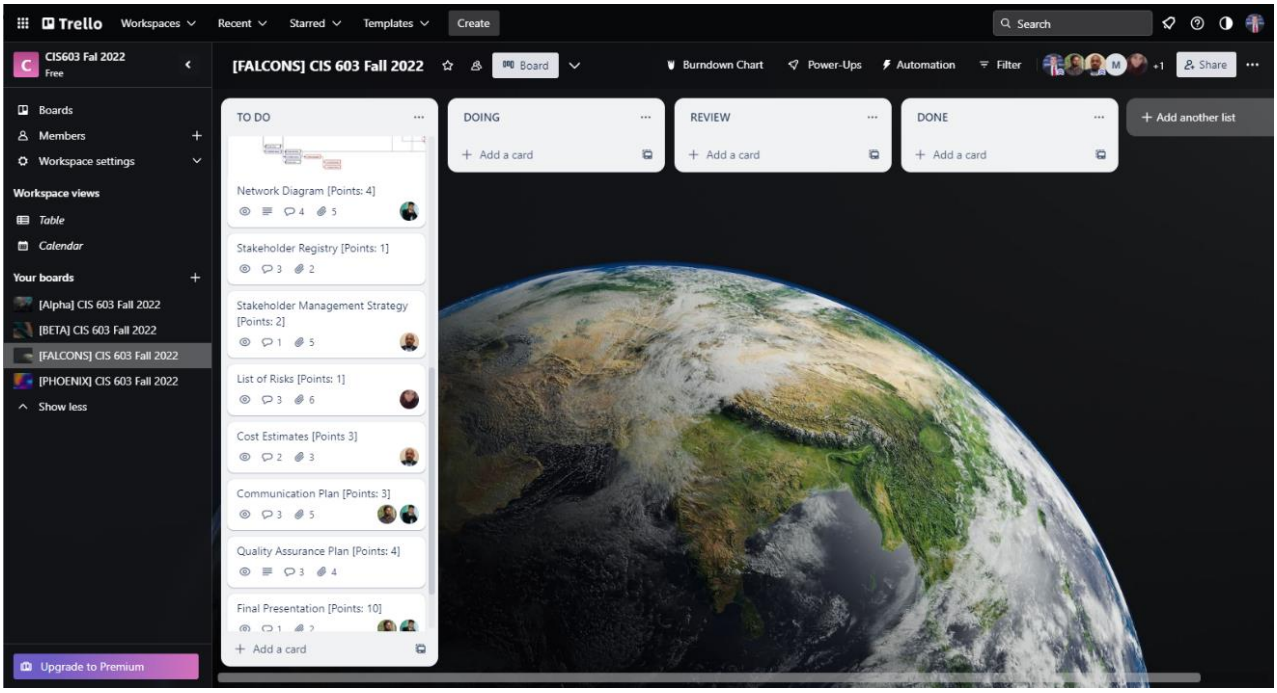

Figure 3. Trello Scrum Board (ScrumEducator.org, 2023)

Eventually, the instructor and students converged on the following version of a combined Scrum Board and Burndown Chart board (see Figure 4).

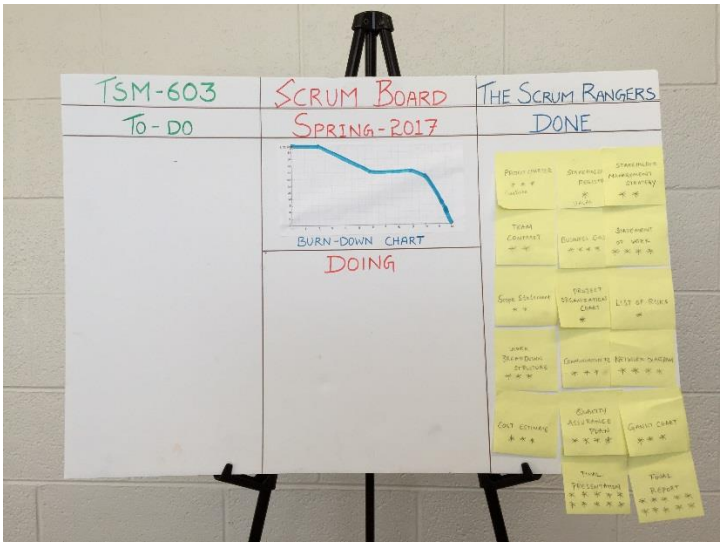

Figure 4. Scrum Board and Burndown Chart Combined (ScrumEducator.org, 2023)

Note that the Scrum Board in Figure 4 is simpler (it has three columns) and also includes a small Burndown Chart in the middle (made from a sheet of graphing paper). This makes it cheaper and simplifies the logistics associated with Weekly Scrums (e.g., only one tripod is needed to mount this board). The Burndown Chart, just like the previous version thereof, contains 60 points reflecting the total weight of all the Deliverables (see Table 2).

Phase 4: Creating the Backlog

To start working on the group project, students need to familiarize themselves with the Deliverables comprising the group project and to create a Backlog. Templates for each of the Deliverables are posted to Canvas at the start of the class. For Week 5 meeting, the instructor should print the templates for each of the team and place them in a folder. This is needed so that each team always has physical copies of the required Deliverables in front of them. Usually, students bring smartphones or laptops to class, so they can also access these templates online via Canvas.

Creating a Backlog requires using a large "sticky note" to capture information (or "user story", as it is called in Scrum) for each of the Deliverables comprising the group project. At the minimum, each of these large notes should contain: (1) the name of the Deliverable; (2) the value in terms of Story Points (can be indicated by putting thick dots using a marker) (3) the person in charge of the Deliverable. The instructor explains to students that several people can collaborate on a particular Deliverable. In fact, this is often necessary as individual Deliverables overlap or rely on several other Deliverables. However, there should be one person in charge of each Deliverable. This person will be responsible for finalizing the Deliverable and submitting it to the instructor via Canvas for grading. Small "sticky notes" can be used to subdivide large Deliverables into sub-parts or to simply add additional notes to each of the Deliverables.

Once a note (or notes) is created for each Deliverable, students need to place them into the "To Do" column of their Scrum Boards in the order in which these Deliverables should be completed. This requires students to develop a shared understanding of what it will take to complete the entire project. The instructor acts as a consultant to make sure students arrange the Deliverables in an order that reflects the actual dependencies among them. For example, students cannot work on their Gantt Chart or Network Diagram unless their Work Breakdown Structure is finalized (see Table 2). Ordering the notes in the To Do column completes the creation of the Backlog.

Once each Group creates a Backlog, the instructor asks students to select a few items to work on for the next week (or Sprint). The "sticky notes" devoted to these items are placed into the "Doing" column of the Scrum Board. Some Groups make very ambitious plans for the first week selecting several labor-intensive items. The instructor can advise them to start with a Project Charter and a Scope Statement. Spending more time on these brief yet important documents can put the entire project on the right track.

Phase 5: Working in Sprints

Starting Week 6, Groups should be ready to start working in Sprints. It is communicated to students that, by default, each Sprint will be one week long. Groups are asked to commit to certain amount of work of each Sprint (week). Sometimes Sprints are longer than 1 week due to class cancelations (e.g., due to holidays or exams). Students seem to prefer longer Sprints, as this gives them more time to work on their Deliverable. Yet the instructor has found that one-week Sprints allow to intervene earlier in case a student is "stuck" or simply fails to do any work.

At the end of each Sprint, the instructor asks each team to have a Weekly Scrum. The Weekly Scrum is comprised of two meetings. First, students are asked to convene with their Group members to discuss what has (or has not) been done the week prior to that and what needs to be done for next week. Before asking students to break into Groups, the instructor gives students overall feedback on the submissions. During the Group meeting, students need to develop a shared understanding of what has been done and what needs to be done. After this, Groups need to update their Scrum Board and Burndown Chart (if some items were actually completed). The second meeting happens closer to the end of the Weekly Scrum. It is a "stand-up" meeting between the Group and the instructor. This requires the entire Group to come forward to the instructor and place their Scrum Board and Burndown Chart in front of the instructor. During the first Sprints, the instructor acts as a Scrum Master and Product Owner. The instructor asks the team the following three questions:

1. What have you done during this Sprint?
2. What issues are you facing in your work?
3. What are you planning to do for the next Sprint?

Going through these questions is usually enough to generate a brief yet useful discussion among the Group members and the instructor. After a few weeks, the instructor can usually identify a "natural leader" within each Group and transfer the responsibilities of a Scrum Master to him or her. This can be done by telling the leader that he or she needs to go through these three questions with the Group in front of the instructor. The instructor can help the student Scrum Master "interrogate" the Group members who do not actively participate in these "strand-up" meetings. It is especially important for the instructor to politely ask those Group members who have not accomplished anything during the Sprint or whether they need any help. This is usually enough to put "peer pressure" on the lagging team members without the Group leader having to do any disciplining in front of everyone. The instructor remains the Product Owner for the rest of the semester, acting as the ultimate authority on what needs to be done for the group project.

In order for a student to say that something has been done during a Sprint, a specific Deliverable needs to be submitted via Canvas. Submission to Canvas constitutes the so-called Definition of Done for this Scrum adaptation. This rule is reiterated and strictly enforced by the instructor throughout the semester. Sometimes students forget the rule and report things as being done without actually submitting them to Canvas. This probably happens due to the fact that these students are embarrassed to admit in front of everyone that they have not done anything during the Sprint.

Sometimes students bring up some grading issues during Weekly Scrums. For example, they may remind the instructor that a particular important Deliverable has not been graded or ask questions about the grade assigned. They may also ask the instructor to clarify some of the feedback that was given to them within the document (the instructor grades each document in Canvas by inserting in-text notes and highlights for each of the documents submitted). The instructor may open these Deliverables on his class computer and discuss those issues with students. The instructor makes sure that all submissions for the week are graded at least one day prior to each class. This gives students some time to look at the feedback and raise meaningful questions during Weekly Scrum meetings.

If students wish to resubmit a particular Deliverable and have it regraded, they need to obtain instructor permission. The instructor grants this permission only if it is important to redo a particular Deliverable to increase the quality of the entire project. Ideally, the instructor wants to give unlimited submission attempts for each of the Deliverables. This will help the students to proceed with their group project in a truly iterative fashion. However, this is simply not possible when a class is large. But students can "resubmit" all of their Deliverables when they submit their Final Report, which is a compilation of all Deliverables submitted during the semester.

At the end of each stand-up meeting with each Group, the instructor gives the team a concluding message to motivate them. For example, if the instructor feels that the team is making good progress, then the instructor will praise the entire Group. If the instructor thinks that the Group is lagging and has noticed certain quality issues with the submitted work (e.g., poor writing style or lack of understanding of some fundamental concepts in project management), the instructor will bring it up and emphasize the need to take some corrective actions from the team. If the class is too big or the instructor simply cannot remember the details of the feedback given to each of the teams, then notes should be taken prior to the class and used as memory aids for formulating this "weekly message" to the team.

Some corrective actions may be required on the instructor side as well. For example, poor writing quality has been an issue with several Groups due to the fact that the majority of students enrolled in the class are international students. The instructor addresses this problem by conducting additional in-class workshops on effective technical writing and proper referencing using APA style.

Phase 6: Closing Phase

After going through around 10 Sprints, teams should be done with the project. As it was mentioned previously, the last week of the semester is devoted to Final Presentations. The Final Report is due at the end of this week as well. The instructor also asks students to complete a simple online survey (created using Google Forms) where they provide quantitative and qualitative Peer Evaluations of their own contribution to the group project and the contributions of their peers. While each student sees his or her overall grade for the peer evaluation component of the group project, the student will not see individual scores assigned by students and the qualitative feedback.

This feedback is used by the instructor to calculate the peer evaluation grade component for each of the students. Typically,

the instructor averages out the quantitative scores given to a particular student by each of the Group members. Qualitative feedback is used by the instructor to validate the numeric scores given. For example, if somebody gives a very low score to another student, then it will be explained that the student being evaluated has not contributed much to the group project at all due to poor attendance. The instructor informs the students that he reserves the right to delete "outliers"—scores that are either too high or too low relative to others and are not backed by convincing justifications. So far, the instructor has not observed too many outliers. Students tend to assign fair and consistent grades to their Group members.

Before the final exam, the instructor asks students to update their Burndown Charts to "zero points left" and place all of the notes into the Done column of their Scrum Boards to officially close the project. As a joke, the instructor recommends the students to go and celebrate the completion of the project and the course as a group after the final exam. Some teams follow this advice.

4. Student Feedback

This section discusses the feedback obtained from students in relation to the Scrum implementation in class. First, this section provides an overview of the survey used to capture student perceptions. The survey is provided in its entirety in Appendix A. Second, the section provides an overview of the demographics of survey participants. Third, the qualitative and quantitative feedback obtained from the survey participants is discussed. Detailed quantitative results of the survey are provided in Appendix B.

4.1 Student Survey

An online survey was designed using Google Forms and administered to students to obtain their feedback on the use of Scrum for managing group project work in the course (see Appendix A). The first part of the survey captures student demographics. The second part of the survey asks students to provide qualitative, open-ended feedback on the things they like and do not like about the Scrum implementation in the course. The third part of the survey also asks students to indicate the degree to which they agree or disagree with statements outlining potential benefits associated with using Scrum in class. A total of 32 questions were asked, with each of the questions falling into one of the following four value dimensions: (1) practical knowledge of Scrum, Agile philosophy, and project management in general; (2) group work dynamics; (3) team-based learning; and (4) overall group performance. Five-point Likert scale (from 1 – "Strongly Disagree" to 5 – "Strongly Agree") is used to capture student responses to each of the questions.

4.2 Student Demographics

While the survey was sent to all students who took this course over the period of two years, most of the students who responded were enrolled in the latest course. Responses were obtained from a total of 28 students. Majority of the respondents are males in their 20s majoring in CIS and CS and having some IT experience (see Table 4).

4.3 Qualitative Feedback

Some examples of positive and negative qualitative feedback obtained from students with the help of the survey are provided in Table 5. All meaningful quotes were included. Some quotes were edited to improve grammar, punctuation, and clarity.

4.4 Quantitative Feedback

An aggregated summary statistics for the four value dimensions of using Scrum in the classroom is provided in Table 6. As shown in the table, an overwhelming majority of students agree that using Scrum for managing the class group project helps students to (1) acquire practical knowledge of Scrum, Agile philosophy, and project management in general; (2) enhance positive group work dynamics; (3) improve team-based learning; and (4) improve the overall performance of groups on the group project assigned to them in class. Detailed survey results are provided in Appendix B.

Table 4. Student Demographics

Demographic Variable	Demographic Group	(n)	(%)
Gender	Female	7	25%
	Male	21	75%
Age	21-25	18	64%
	26-30	10	36%
Major	Computer Information Systems (CIS)	14	50%
	General Business	1	4%
	Telecommunications Systems Management (TSM)	13	46%
Years of Professional Experience	None	12	43%
	6 months to less than 1 year	1	4%
	1 year to less than 3 years	7	25%
	3 years to less than 5 years	4	14%
	5 years or more	4	14%
Years of Professional Experience in IT Sector	None	13	46%
	6 months to less than 1 year	2	7%
	1 year to less than 3 years	10	36%
	3 years to less than 5 years	3	11%
	5 years or more	0	0%

Table 5. Qualitative Feedback from Students: Likes and Dislikes

Likes	Dislikes
• Knowing the progress of the project in real time and right from the beginning of the course • Helps in enhancing the quality of the group project • Helped complete the project on time • Shows the flow of work done in specific time • Helps organize the project activities easily • Helps to see the bigger project through smaller tasks • Weekly meetings and incremental approach to group project work was quite effective • It makes accountability and control within the team easier • Good way to visualize work • Makes it easier to make changes to how the project is done • Good way to organize teamwork • Stand-up meetings were effective because everyone in the team knew what to do in the following week	• Sprints should be 2 weeks or more; one-week sprints make things hectic sometimes • Sometimes we forget to update the Scrum Board • The Scrum framework forces you to rely on the entire team. If a team member is sick, then his work has to be reassigned to another team member to keep up with Sprint commitments • It would be more helpful if Scrum Board was online • Giving updates on task might distract from actual work • Too much paper work to do • Project quality is hard to manage unless the team members do some quality assurance at the end of each sprint • Becoming an effective Scrum Master takes time while the project is due fairly soon

4.5 Possible Improvements

While student response to the implementation of Scrum in the course is mostly positive, some improvements may be necessary based on the student feedback and the instructor's own reflections. First, additional online tools (e.g., Trello for creating and managing online Scrum Boards) can be used to improve convenience and accessibility of some of the main Scrum artifacts while also preserving the benefits associated with face-to-face interaction. Perhaps an online board can be broadcasted to the projector

screen during Weekly Scrums. Second, grading should be somehow incorporated into the methodology. For example, the Burndown Chart can be modified to include not only "points completed" but also "points earned" towards the grade. Perhaps a modified Earned Value Management (EVM) approach can be used (Schwalbe, 2007). Third, ways of formalizing and automating assessment of individual member's contribution to the group project and "soft skills" necessary to be an effective team member should be researched. All these improvements should be done in a way so that the framework remains "light", consistent with the vision of the creators of Scrum (Sutherland, 2014), and does not increase the "overhead" associated with the implementation of the framework substantially. "Keeping it simple" has been the most important goal of the instructor since the very first time the framework was introduced in his course. This makes the implementation simple and general enough to be used in any Information Systems or general business course involving a group project. Still, additional Scrum tools, techniques, and artifacts can be added to this adaptation of Scrum based on the nature and the goals of the course where Scrum is used.

Table 6. Quantitative Feedback from Students: Value Dimensions of Scrum

Dimension Using Scrum for managing the group project allowed me…	**Mean**	**Standard Deviation**	**"Agree" or "Strongly Agree" (%)**
…acquire practical knowledge of Scrum, Agile philosophy, and project management in general	4.54	0.57	96%
… enhance positive group work dynamics	4.49	0.72	91%
… improve team-based learning	4.53	0.62	93%
… improve the overall performance of groups on the group project	4.45	0.63	93%

5. Conclusion

This paper has presented a practical and accessible model for integrating Scrum into a face-to-face Information Systems Project Management course, offering a low-tech yet effective approach to managing student group work. By applying Scrum to the development of traditional project management artifacts—rather than software deliverables—this adaptation bridges the gap between Agile principles and "waterfall-based" learning objectives in a typical Project Management class. The Scrum adaptation model consists of six clearly defined phases, with each phase containing detailed implementation instructions and the instructor's recommendations based on his experience implementing Scrum in the course. The implementation is supported by simple tools, such as physical or electronic Scrum boards, and processes, such as peer-evaluations, making it applicable across a wide range of educational settings. Empirical feedback from students suggests that this approach not only enhances engagement and accountability but also supports the development of both Agile competencies and teamwork skills. For instructors, the framework offers a straightforward method for monitoring progress, promoting team collaboration, and aligning classroom practices with real-world project dynamics. Future research may explore the adaptation of this model to online or hybrid settings and investigate long-term impacts on student performance and soft skill development. As the demand for Agile-ready graduates continues to grow, this classroom-tested implementation of Scrum provides a meaningful and effective pathway for educators to prepare students for today's collaborative, fast-paced work environments across diverse organizations and industries.

6. References

Axosoft. (2014). *Scrum in under 10 minutes* [Video]. YouTube. https://www.youtube.com/watch?v=XU0llRltyFM

Baham, C. (2020). Improving business product owner commitment in student scrum projects. *Journal of Information Technology Education: Research, 19*, 243–258. https://doi.org/10.28945/4549

Babik, D. (2022). Teaching Tip: Scrum boot camp: Introducing students to Agile system development. *Journal of Information Systems Education*, *33*(3), 195-208.

eduScrum. (2020). *The eduScrum guide: The rules of the game*. https://www.scribd.com/document/551896600/The-eduScrum-guide-English-2-0-update-21-12-2019

Dong, L. (2023). Strengthening sustainability in Agile Education: Using client-sponsored projects to cultivate Agile talents. *Sustainability*, *15*(11), 8598.

Fernandes, S., Dinis-Carvalho, J., & Ferreira-Oliveira, A. T. (2021). Improving the performance of student teams in project-based learning with scrum. *Education Sciences, 11*(8), 444. https://doi.org/10.3390/educsci11080444

Highsmith, J., & Cockburn, A. (2001). Agile software development: The business of innovation. *Computer, 34*(9), 120–127. https://doi.org/10.1109/2.947100

Krehbiel, T. C., Salzarulo, P. A., Cosmah, M. L., Forren, J., Gannod, G., Havelka, D., & Merhout, J. (2017). Agile manifesto for teaching and learning. *Journal of Effective Teaching, 17*(2), 90–111.

López-Alcarria, A., Olivares-Vicente, A., & Poza-Vilches, F. (2019). A systematic review of the use of agile methodologies in education to foster sustainability competencies. *Sustainability, 11*(10), 2915. https://doi.org/10.3390/su11102915

Mahalakshmi, M., & Sundararajan, M. (2015, February). Tracking the student's performance in web-based education using Scrum methodology. In *Computing and Communications Technologies (ICCCT), 2015 International Conference on* (pp. 379–382). IEEE. https://doi.org/10.1109/ICCCT2.2015.7292761

Manamendra, M. A. S. C., Manathunga, K. N., Perera, K. H. D., & Kodagoda, N. (2013, April). Improvements for agile manifesto and make agile applicable for undergraduate research projects. In *Computer Science & Education (ICCSE), 2013 8th International Conference on* (pp. 539–544). IEEE. https://doi.org/10.1109/ICCSE.2013.6554019

Matkovic, P., Tumbas, P., & Vukovic, V. (2016). Agile development processes and software architecture in study programmes of higher education institutions. In *Proceedings of ICERI2016 Conference* (pp. 3092–3100).

May, J., York, J., & Lending, D. (2016). Play ball: Bringing Scrum into the classroom. *Journal of Information Systems Education, 27*(2), 85–98.

Moe, N. B., Dingsøyr, T., & Dybå, T. (2010). A teamwork model for understanding an agile team: A case study of a Scrum project. *Information and Software Technology, 52*(5), 480–491. https://doi.org/10.1016/j.infsof.2009.11.004

Morales-Trujillo, M. E., Galster, M., Gilson, F., & Mathews, M. (2021). A three-year study on peer evaluation in a software engineering project course. *IEEE Transactions on Education, 65*(3), 409–418. https://doi.org/10.1109/TE.2021.3117032

Mutchler, L. A., Connolly, A. J., & Rush, D. E. (2024). Teaching tip: Leveraging scrum to increase student engagement and skill building in information security management. *Journal of Information Systems Education*, *35*(4), 422-437.

Nejmeh, B., & Weaver, D. S. (2014, October). Leveraging Scrum principles in collaborative, interdisciplinary service-learning project courses. In *2014 IEEE Frontiers in Education Conference* (pp. 1–6). IEEE. https://doi.org/10.1109/FIE.2014.7044087

Pope-Ruark, R. (2012). We Scrum every day: Using Scrum project management framework for group projects. *College Teaching, 60*(4), 164–169. https://doi.org/10.1080/87567555.2012.660761

Reynolds, S., Caldwell, A., Procko, T., & Ochoa, O. (2023, October). Scrum in the classroom: An implementation guide. In *2023 IEEE Frontiers in Education Conference* (pp. 01–08). IEEE. https://doi.org/10.1109/FIE58773.2023.10339309

Rigby, D. K., Sutherland, J., & Takeuchi, H. (2016). Embracing Agile. *Harvard Business Review, 94*(5), 40–50.

Rush, D. E., & Connolly, A. J. (2020). An agile framework for teaching with Scrum in the IT project management classroom. *Journal of Information Systems Education, 31*(3), 176–187.

Salza, P., Musmarra, P., & Ferrucci, F. (2019). 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* (1st ed., pp. 25-45). Springer.

ScrumEducator.org. (2023). *Using Scrum in Education: A Definitive Guide*. https://scrumeducator.org/wp-content/uploads/2023/07/Scrum_in_Education_V01_07122023-.pdf

Schwalbe, K. (2007). *Information technology project management* (6th ed.). Boston, MA: Cengage Learning.

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.

Sutherland, J. (2014). *Scrum: The art of doing twice the work in half the time*. New York, NY: Random House Business Books.

Sutherland, J., & Schwaber, K. (2013). *The Scrum Guide*. Scrum.org. https://scrumguides.org

Thouin, M. F., & Hefley, W. E. (2024). Teaching tip: Teaching scrum product owner competencies using an experiential learning simulation. *Journal of Information Systems Education*, *35*(1), 37-47.

VersionOne. (2023). *17th annual state of agile report*. https://stateofagile.com/

Vogelzang, J., Admiraal, W. F., & van Driel, J. H. (2021). Scrum methodology in context-based secondary chemistry classes: Effects on students' achievement and perceptions. *Instructional Science*, *49*(5), 719–746. https://doi.org/10.1007/s11251-021-09553-1

Wagh, R. (2012). Using Scrum for software engineering class projects. In *AGILE India 2012* (pp. 68–71). IEEE. https://doi.org/10.1109/AGILEIndia.2012.13

Wolff, E., & van Ryneveld, L. (2024). The effective management of remote higher education curriculum design teams using Scrum. *Revista de Gestão Social e Ambiental*, *18*(9), Article e03291. https://doi.org/10.24857/rgsa.v18n9.2024.3291

Zahorodko, P. V. (2023). Overview of Agile frameworks in computer science education. *Educational Dimension, 9*, 206–214. https://doi.org/10.31812/educdim.v9i3.937

Zorzo, S. D., de Ponte, L., & Lucredio, D. (2013, October). Using Scrum to teach software engineering: A case study. In *2013 IEEE Frontiers in Education Conference* (pp. 455–461). IEEE. https://doi.org/10.1109/FIE.2013.6684899

Appendix A: Online Student Survey

Scrum Framework Survey

Dear Participant,

This study explores the effectiveness of using Scum Framework for managing class group work. This research can potentially benefit various educational institutions by equipping them with a theoretical and practical understanding on how to use Scrum for improving group work and team-based learning in the classroom.

Completing this survey should take approximately 15 minutes of your time.

Your participation in this survey is absolutely voluntary. You are free to choose not to participate in the survey or discontinue the survey at any time. Your responses will be anonymous. The information that you provide through this survey will not be linked to your identity. Because this survey is anonymous, the researcher reserve the right for a variety of ethical uses of this data at their own discretion. The use may involve sharing this data with other researchers or the public.

Please feel free to contact [deleted] if you have any questions or comments about this survey. Thank you!

Personal Information

What is your gender?

How old are you?

Which of the following describes you best?

Are you an online student?

What is your major?

How many years of professional experience do you have?

How many years of professional experience in the IT sector do you have?

General Comments

In general, what did you like about using the Scrum Framework for managing your group project? How did the framework help you with your group project?

In general, what are the things that you DID NOT like about using the Scrum Framework for managing your group project work? Are there any specific elements of this framework that you found to be of little value or distracting?

What suggestions do you have for the instructor on how the Scrum Framework should be used in class? What are some of the specific ways in which the instructor can improve the effectiveness of the framework for managing class group projects?

Project Management Knowledge Dimension (D1)

Using the Scrum Framework for managing our group project work helped me(us)...

D1Q1 [...Learn the fundamental principles of Agile project management philosophy]

D1Q2 [...Learn about the Scrum methodology]

D1Q3 [...Gain a practical understanding of the Scrum methodology]

D1Q4 [...Learn the main roles under the Scrum framework]

D1Q5 [...Learn about the Scrum artifacts and how to use them]

D1Q6 [...Learn Scrum processes and practices]

D1Q7 [...Learn about the concepts, ideas, practices, documents, and deliverables used in project management]

D1Q8 [...Gain a practical understanding on how to manage projects]

D1Q9 [...Gain practical experience working on projects]

Group Work Dimension (D2)

Using the Scrum Framework for managing our group project work helped me(us)...

D2Q1 [...Learn how to lead teams]

D2Q2 [...Learn how to work as a part of a group]

D2Q3 [...Learn how to facilitate group work]

D2Q4 [...Improve communication among the group members]

D2Q5 [...Improve communication between the instructor and the team]

D2Q6 [...Reduce conflicts among the team members]

D2Q7 [...Reach consensus among the team members on various issues related to the group project]

D2Q8 [...Improve collaboration among the group members]

D2Q9 [...Increase involvement of team members into the group work]

D2Q10 [...Solve problems related to the group project]

D2Q11 [...Become more organized with our group work]

Team-Based Learning Dimension (D3)

Using the Scrum Framework for managing our group project work helped me(us)...

D3Q1 [...Learn from other team members]

D3Q2 [...Get timely feedback on my understanding of various project management concepts, documents, and practices from my team members]

D3Q3 [...Get timely feedback on my understanding of various project management concepts, documents, and practices from the instructor]

D3Q4 [...To understand various project management concepts, documents, deliverable, and practices]

D3Q5 [...Reach a mutually shared understanding of various project management concepts, documents, deliverables, and practices]

Group Project Performance Dimension (D4)

Using the Scrum Framework for managing our group project work helped me(us)...

D4Q1 [...Complete group project tasks in shorter time]

D4Q2 [...Avoid mistakes in relation to the group project]

D4Q3 [...Improve the quality of group project deliverables]

D4Q4 [...Increase our grade for the group project]

D4Q5 [...Enjoy working in the group]

D4Q6 [...Enjoy the class]

D4Q7 [...Benefit from the class]

Appendix B: Detailed Survey Results

Dimension 1 (D1)	**Mean**	**Median**	**SD**	**% of 5**	**% of 4**
D1Q1	5	5	0.638285	57%	36%
D1Q2	5	5	0.57275	61%	36%
D1Q3	5	5	0.57275	61%	36%
D1Q4	5	5	0.507875	54%	46%
D1Q5	4	5	0.685257	50%	39%
D1Q6	4	5	0.576204	50%	46%
D1Q7	5	5	0.417855	79%	21%
D1Q8	5	5	0.503953	57%	43%
D1Q9	4	4	0.57275	46%	50%
Dimension 1 Overall	**5**	**5**	**0.566947**	**57%**	**39%**
Dimension 2 (D2)	**Mean**	**Median**	**SD**	**% of 5**	**% of 4**
D2Q1	4	5	0.772374	50%	32%
D2Q2	5	5	0.693889	61%	29%
D2Q3	5	5	0.692935	64%	25%
D2Q4	5	5	0.566947	64%	32%
D2Q5	5	5	0.475595	68%	32%
D2Q6	4	5	0.854493	50%	32%
D2Q7	4	5	0.780042	50%	39%
D2Q8	5	5	0.57275	61%	36%
D2Q9	4	5	0.999338	68%	21%
D2Q10	4	5	0.685257	50%	39%
D2Q11	5	5	0.611832	75%	18%
Dimension 2 Overall	**4**	**5**	**0.715056**	**60%**	**31%**
Dimension 3 (D3)	**Mean**	**Median**	**SD**	**% of 5**	**% of 4**
D3Q1	4	5	0.780042	54%	29%
D3Q2	4	5	0.576204	50%	46%
D3Q3	5	5	0.48795	64%	36%
D3Q4	5	5	0.637248	61%	32%
D3Q5	5	5	0.558721	68%	29%
Dimension 3 Overall	**5**	**5**	**0.616825**	**59%**	**34%**
Dimension 4 (D4)	**Mean**	**Median**	**SD**	**% of 5**	**% of 4**
D4Q1	4	4	0.645497	**36%**	**54%**
D4Q2	4	4	0.662687	25%	57%
D4Q3	4	4	0.558721	39%	57%
D4Q4	5	5	0.566947	64%	32%
D4Q5	5	5	0.693889	61%	29%
D4Q6	5	5	0.547964	71%	25%
D4Q7	5	5	0.460044	71%	29%
Dimension 4 Overall	**4**	**5**	**0.6267**	**53%**	**40%**

Author Biographies

Dr. Vlad Krotov received his PhD in Management Information Systems from the Department of Decision and Information Sciences, University of Houston (USA). Currently, he is a Professor of Information Systems at Murray State University and a consultant at Accreditation.Biz - an international accreditation consulting company for business schools. He is also a founder of ScrumEducator.org – a free resource for training students, educators, and professionals in essential soft skills. His research and teaching interests are at the intersection of business and technology and fall under such topics as business analytics, strategic IT management, project management, innovation, and ethical aspects of AI.

Dr. Pitzel Krotova is the Director of Institutional Assessment at Murray State University (Kentucky, USA). She earned her EdD in P-20 and Community Leadership with a specialization in Postsecondary Leadership from Murray State University. She has fifteen years of international experience in higher education, specifically in academic administration and institutional effectiveness. Prior to joining Murray State University, she was the College of Business's Academic Quality Assurance Coordinator at Abu Dhabi University in the UAE. Her research interests are related to assurance of learning, assessment, educational leadership, program review, accreditation, and curriculum alignment.

This page intentionally left blank.