Journal of Information Systems Education

Volume 36
Issue 1
Winter 2025

Teaching Tip

Experiential and Peer Learning in an IT Project Management Course: Flipped Classroom, Concept Maps, and Project Dojos

Elahe Javadi, Judith Gebauer, and Season Tanner

Recommended Citation: Javadi, E., Gebauer, J., & Tanner, S. (2025). Teaching Tip: Experiential and Peer Learning in an IT Project Management Course: Flipped Classroom, Concept Maps, and Project Dojos. *Journal of Information Systems Education*, 36(1), 13-24. https://doi.org/10.62273/WMDE5362

Article Link: https://jise.org/Volume36/n1/JISE2025v36n1pp13-24.html

Received: March 18, 2024
First Decision: July 30, 2024
Accepted: October 26, 2024
Published: March 15, 2025

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

Journal of Information Systems Education, 36(1), 13-24, Winter 2025
https://doi.org/10.62273/WMDE5362

Teaching Tip
Experiential and Peer Learning in an IT Project Management Course: Flipped Classroom, Concept Maps, and Project Dojos

Elahe Javadi
School of Information Technology
Illinois State University
Normal, IL 61790, USA
ejavadi@ilstu.edu

Judith Gebauer
Congdon School of Supply Chain, Business Analytics, and Information Systems
University of North Carolina Wilmington
Wilmington, NC 28403, USA
gebauerj@uncw.edu

Season Tanner
State Farm
Bloomington, IL 61710, USA
season.tanner.lem8@statefarm.com

ABSTRACT

In this teaching tip, we describe our approach to elevating the quality of group work in an information technology (IT) project management course by implementing three practices of experiential and peer learning that work more effectively when combined. The first practice addresses slacking in group work by applying a flipped classroom style that allowed students to acquire individual skills and then reflect on those skills in pairs with the instructor's support and supervision. The second practice uses concept maps to deepen the understanding of relevant tools and concepts and to foster the synthesis of topics and big-picture composition. The third practice addresses the prevalent divide-and-conquer culture in group work, in which we structured intense work sessions in the classroom with instructor guidance and supervision to complete group projects. In this proposed framework, the work sessions mimic the concept of a dojo that is widely used in martial arts. Even though it is still rare in information systems (IS) education, this concept has been applied in the context of software engineering and IT projects—both in higher education and in industry for employee training. The three-pronged approach described in this paper is intended to provide students with an experiential learning experience that transforms group work from an inefficient process to an intense and focused short-term period of work. Based on our observations and student feedback, the approach is effective and can positively impact group experience by countering common pitfalls, such as the divide-and-conquer mentality and social loafing, and by enhancing peer learning efficacy. We also provide suggestions for its application in the context of an IT project management course.

Keywords: Group projects, IT project management, Dojo, Concept maps, Peer learning, Experiential learning & education

1. INTRODUCTION

The importance of experiential learning and group work in Information Systems education is widely recognized by professionals (Middleton, 2024; Qurban & Austria, 2009) and academic accrediting bodies, such as ABET (ABET Computing Accreditation Commission, 2023) and the Association to Advance Collegiate Schools of Business (AACSB) (AACSB, 2023). Both have long been included in curriculum guidelines, such as IS 2010 (Topi et al., 2010) and IS 2020 (Leidig & Salmela, 2021).

ABET expects the graduates of its accredited computing programs to have experience leading and effectively participating in discipline-specific teams (ABET Computing Accreditation Commission, 2023), while AACSB-accredited business schools typically include applied experience and teamwork in their stated school-level learning goals (AACSB, 2023; Leidig & Salmela, 2021).

IS 2010 emphasizes the importance of team skills for graduates who will need to effectively interact and collaborate with clients and other professionals, as well as perform successfully at the individual level (Topi et al., 2010). It includes leadership and collaboration, communication, and negotiation in its list of foundational knowledge and skills. The IS 2010 curriculum guidelines further state the need for laboratories to provide students with opportunities to collaborate on group projects, and emphasize the value of group skills for the success of IS professionals. The need to provide an applied learning experience and to cover team-related skills is highlighted specifically for the IS core course on project management (Topi et al., 2010).

In the most recently revised guidelines for IS curricula, Leidig and Salmela (2021) emphasize the need for applied experiences and teamwork. One example would be the areas of web and mobile development, which rely on interpersonal group skills. They further point out the critical importance for students to "experience unanticipated and unscripted problems as a team" as part of the IS integration competency realm (p. 60), for which two required competency areas are suggested, namely project management and IS practicum. While project management focuses more on organizational concerns, an IS practicum is typically geared more towards technical and implementation concerns. Both areas may be combined, both aim to provide practical experience, and both rely substantially on teamwork. Applied learning and teamwork are also emphasized in several competency areas, including the application of emerging technology, systems analysis and design, web and mobile development, IS management and strategy, IS project management, and the IS practicum (Leidig & Salmela, 2021).

There is a rich repository of theory, research, and practice on cooperative learning, which is described as an approach where "students can work together cooperatively to accomplish shared learning goals. Each student achieves his or her learning goal if and only if the other group members achieve theirs" (Johnson et al., 1998, p. 28). In small groups, the students collaborate, and each group member must "achieve up to a preset criterion. When all group members reach criteria, each member may receive bonus points" (Johnson et al., 1998, p. 28). A closely related concept is peer learning, defined as the "acquisition of knowledge and skill through active helping and supporting among status equals or matched companions. It involves people from similar social groupings who are not professional teachers helping each other to learn and learning themselves by doing" (Topping, 2005, p. 631). In mutual peer learning in particular, "students are encouraged to work together in small teams on academic tasks in order to achieve a common goal and to develop mutual knowledge and skills" (Topping et al., 2017, p. 2).

However, group work does not automatically result in cooperative learning, but instead presents limitations and challenges and requires careful design and adaptation on the part of the faculty (Boud et al., 2001; Johnson et al., 1998; King, 1990, 2002; Piaget, 1970; Topi et al., 2010; Topping, 2005; Vygotsky, 1978). There is also a learning curve for students who enter higher education more used to competitive and individualistic learning than cooperative approaches (Johnson et al., 1998). When group work is presented to students before they are prepared to tackle the tasks of the assignment in conjunction with the intricate aspects of human interaction in groups, the results can be inefficiencies such as slacking and social loafing (Kapp, 2009), dissatisfaction, and a divide-and-conquer approach (O'Donnell & King, 1999; Topping et al., 2017) that produce lower quality work than when individuals work alone. According to Brooks (1974), a perfectly partitionable project that is particularly amenable to the divide-and-conquer technique implies a group work design that only superficially engages students in the group work mechanics. In such scenarios, the group outcome as a whole will not necessarily be better than the sum of its parts (Frazee, 2021). A group project aimed at preparing students for the professional journey after college will require continuous collaboration, communication, and evaluation and integration of ideas. These elements of collaborative work cannot be practiced in partitionable tasks that group members can work on separately with limited communication or integration required for the group outcome (i.e., divide-and-conquer). Instructor supervision can occur during class, but class-time is often used inefficiently and ineffectively (Boud et al., 2001; Cooper, 2002).

The approach described in this teaching tip aims to be comprehensive by equipping students with in-depth skills related to the course content and big-picture-view knowledge of the topics and their connections, as well as providing a work environment (dojo) that is conducive to collaboration and co-creation of group outcomes. More specifically, we build our classroom structure on three interrelated pillars as follows. (1) Students learn and practice course-related concepts and skills in a flipped classroom manner. To counter slacking, we focus this first pillar on students' individual acquisition of skills that they then reflect on in pairs with the instructor's support and supervision. (2) Students subsequently engage in group-based reflection and meaning-making via collaborative concept mapping to deepen the understanding of relevant tools and concepts, and to foster the synthesis of topics and big-picture composition. (3) To counter the divide-and-conquer culture that is so prevalent in group projects and to help students develop effective group skills, we facilitate dojo-type work sessions in class, in which student interactions and group work are observed and guided by the instructor as needed.

The rest of this paper is organized as follows. Section two presents key components of our suggested approach within the literature. The third section provides more details about our course. Sections four and five present suggestions for teaching and evidence of teaching effectiveness. Section 6 concludes the paper.

2. LITERATURE REVIEW

Scholars of experiential learning regard learning as a process that includes dialectic tensions, conflicts and interactions between the learning person and their environment (Kolb, 1984). Kolb's (1984) model is especially instrumental to inform the design and analysis of learning environments where experiential and peer learning are interwoven and are both important intended outcomes. According to Kolb (1984), "knowledge results from the combination of grasping experience and transforming it" (p. 41), whereby the key building blocks of the learning process include abstract conceptualization, active experimentation, concrete experience, and reflective observation that together form a four-stage cycle (Figure 1).

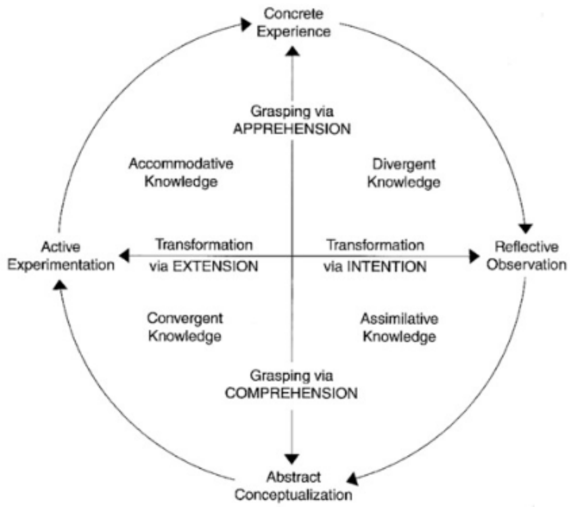


Figure 1. Structural Dimensions Underlying the Process of Experiential Learning and the Resulting Knowledge Forms (Kolb, 1984, p. 42)

Kolb (1984) emphasizes that for learning to occur and knowledge to be developed, both "[a] figurative representation of experience and some transformation of that representation" must happen (p. 42). In Kolb's (1984) own words, "Either the figurative grasp or operative transformation alone is not sufficient. The simple perception of experience is not sufficient for learning rather something must be done with it. Similarly, transformation alone cannot represent learning, for there must be something to be transformed, some state or experience that is being acted upon" (p. 42). Following Kolb's (1984) model, we expose students to abstract concepts as well as concrete experimentation and cover all four stages of the learning cycle throughout the course.

Kolb's (1984) structural foundations of the learning process are informed by and align with the constructivist view of learning, according to which learning is socially constructed during interactions and activities with others (Piaget, 1970). Students each make meaning, discover problems, and resolve problems within their individual minds. Higher levels of cognitive processing and meaning-making can happen through interactions with more competent peers (King, 2002; Topping, 2005; Vygotsky, 1978). If each student has competencies that complement those of others in a group within the context of a course, the interactions that follow individual meaning-making become a more meaningful practice. Cognitive growth results from the coordination of perspectives as individuals work to attain common goals (Palincsar & Herrenkohl, 2002).

Our approach to teaching an IT project management course builds on and applies the pedagogical concepts of experiential learning (Kolb, 1984) as well as cooperative and peer learning (Boud et al., 2001; Cooper, 2002; Topping et al., 2017). Following Kolb's (1984) model (Figure 1), we begin a learning module with a concrete experience where we expose the students to relevant tools, skills, or technologies in a flipped classroom manner, followed by reflective observation in the form of a journaling assignment (Hiemstra, 2001). Both the flipped classroom and journaling assignment are intended to provide the basic content knowledge that the course is focused on (IT project management), motivate the learners, and offer free class time for peer learning. We next employ abstract conceptualization in the form of concept mapping, which is intended to encourage meaning making and learning at a higher level. Lastly, we facilitate active experimentation with project dojos where students practice group work as they collaborate and interact with the tool or technology in a goal-oriented and evaluative manner. These three pillars are intended to contribute to a holistic learning experience. In the following, we position our three pillars (flipped classroom with journaling, concept mapping, and project dojos) within the literature.

2.1 Flipped Classroom With Journaling to Free Class Time for Peer Learning

A flipped classroom is an educational approach where students acquire content knowledge by working through lecture material in the form of textbook readings, foundational learning activities, or lecture videos outside of the classroom. Class time is then spent with active learning and diagnosis activities, which typically include experiential and peer learning. Bergmann and Sams (2012) are credited with first using the term (Tucker, 2012), but the concept has been around longer. For example, King (1990) describes a "guided reciprocal peer-questioning procedure" that is used to promote peer interaction and learning in cooperative groups in the classroom that relies on students preparing course material outside of class. Similarly, Palincsar and Herrenkohl (2002) propose reciprocal teaching and the use of cognitive tools and intellectual roles (CTIR) to promote collaborative learning among peers, whereby the procedural nature of the approach is emphasized as the teacher retreats from active teaching to coaching and peer learning becomes prevalent.

The flipped-classroom approach has been shown to help save time for structured peer learning during class time (Bergman & Sams, 2012). Roehl et al. (2013) report that "using class time for active learning versus lecture provides opportunities for greater teacher-to-student mentoring, peer-to-peer collaboration and cross-disciplinary engagement" (p. 44). A flipped classroom can be particularly effective to engage millennial (and younger) students who have become accustomed to using technology and "demonstrate decreased tolerance" for traditional classroom-based lectures (Roehl et al., 2013, p. 44).

In the context of IS/IT and computing courses, Yildiz Durak (2022) applied different models of flipped-classroom teaching (peer-assisted groups, small group collaborative groups, and individual learning) and found that peer-assisted learning was associated most with improved performance and reduced technology anxiety, while collaborative group work was associated most with increased motivation and self-efficacy. De Miras et al. (2022) used a combination of flipped classroom and peer instruction methodologies in introductory programming courses. The results show improved student performance, reduced drop-out rates, and increased attendance in laboratory sessions. Students also reported an increase of time and quality spent on autonomous work, as reflected in the time spent on class preparation at home.

The flipped-classroom approach has some reported limitations, particularly for content such as statistics. Other reported challenges include reduced flexibility to make changes to recorded lectures, access to technology and the willingness and ability of both the instructors and students to use it, and the need for students to assume more responsibility for their own learning, which needs to be communicated clearly by the instructor, such as in the syllabus (Roehl et al., 2013). Zheng et al. (2020) present a meta-analysis of the effectiveness of the flipped classroom on students' learning achievement and motivation. The results are mixed, showing an overall positive yet moderate impact of the flipped classroom for learning achievement and motivation. Other factors that are believed to also play a role include sample size, intervention duration, and geographic regions, which can reflect students' cultural backgrounds. Zheng et al.'s (2020) results also point to the importance of deliberate course design and structure to make the flipped-classroom model successful.

2.2 Concept Maps to Support Meaning Making

A "concept map … is a graphical tool for organizing and structuring knowledge by depicting concepts as nodes, and relationships between concepts as edges" (Wei & Yue, 2017, p. 13; see also Van Boxtel et al., 2002, p. 41). Concept maps are powerful tools for knowledge management, brainstorming, and communication with stakeholders (Hellström & Husted, 2004). In the pedagogy realm, these visual models of concepts learned during the course have been used as an analytical research tool to assess and/or reinforce student learning (Croasdell et al., 2003; Gregoriades et al., 2009; Novak & Musonda, 1991) as well as a learning tool to help students make meaning of and develop a big-picture view of the concepts covered during a course (Van Boxtel et al., 2002). During the development of a concept map, students articulate relationships between the concepts, terms, techniques, and themes that were previously covered. In the process, a big-picture view of the seemingly disparate topics that students have been working on is expected to emerge. The literature on concept maps suggests that they can enable discovery processes as well as the development of cognitive models (Novak & Cañas, 2008; Van Boxtel et al., 2002; Wei & Yue, 2017).

Concept mapping can be assigned to individual students (Freeman & Urbaczewski, 2001) but has been shown to be particularly effective if students develop the maps in small groups and engage in discussion about the concepts and relations between them during the assignment (Van Boxtel et al., 2002). Students may be primed with a list of categories and asked to identify concepts from each category, then illustrate the connections among the concepts with logical qualifiers (labels) that are consistent with the body of knowledge in the course and discipline. Alternatively, students may be primed with a list of concepts, then asked to connect the concepts and place them in different categories as part of the discovery process (Van Boxtel et al., 2002; Wei & Yue, 2017).

Wei and Yue (2017) report on the use of concept maps in computer science education and emphasize their importance to help students assimilate new concepts into their "existing cognitive structure in substantive ways so they can be applied" (p. 13). Freeman and Urbaczewski (2001) discuss the application of concept maps in an IS capstone course, with the intent to assess the knowledge that students gained over their entire IS curriculum. As part of the assignment, students were instructed "to include at least 150 distinct concepts or items and the appropriate relationships between concepts" (p. 4). While students found the assignment to be "fun," effective, and meaningful, Freeman and Urbaczewski (2001) also report limited overlap between the concept maps developed by the students and those produced by the expert instructor. More specifically, the concepts included in the maps by the students described the knowledge domain at a lower level of understanding than the "model map" by the instructor, pointing to the need to carefully apply structure during the design of the assignment (Van Boxtel et al., 2002).

2.3 Project Dojos to Facilitate Peer Learning

In software engineering education and industry practice, coding dojos have become an established tool to enable experiential learning in small groups, specifically related to agile principles that are notoriously difficult to teach and practice with traditional, lecture-based methods (De Oliveira et al., 2018; Heinonen et al., 2013; Pollard, 2012; Sato et al., 2008).

The meaning of the Japanese word dojo is "place of the way." With a long history in the context of martial arts practice, dojos refer to intensive, deliverable-oriented work sessions that are guided by one or several teachers (masters) and focused on team goals (Heinonen et al., 2013). Participants find the dojo to provide a safe and non-competitive way to engage in the gradual understanding of concepts and practicing of skills in close interaction with the teacher and fellow learners. The teacher explains and demonstrates concepts and provides temporary help (scaffolding) until the learners have developed a grasp on the task and can start performing independently. The intensive and purposeful nature of the collaborative dojo sessions is thought to accelerate learning (development) and improve the quality of the acquired skills (deliverables) (De Oliveira et al., 2018; Scrum Alliance, 2021).

Similar in concept to hackathons (De Oliveira et al., 2018), coding dojos play a role, not only in higher education (Heinonen et al., 2013; Sato et al., 2008), but also in industry (De Oliveira et al., 2018; Scrum Alliance, 2021). Heinonen et al. (2013) report on the use of coding dojos as a "safe learning environment for practicing various programming-related skills" in a software engineering course (p. 97), while Sato et al. (2008) employ coding dojos as an extracurricular initiative with the objective of fostering and sharing agile coding practices, such as test-driven development, refactoring, and pair programming. As an example from industry practice, U.S. retailer Target launched a surprisingly popular coding dojo initiative in 2015 (Scrum Alliance, 2021, see also https://www.youtube.com/watch?v=_H9aLH9W420). De Oliveira et al. (2018) combined participants from industry and higher education when they applied coding dojos with the goal of modernizing two enterprise systems for the Brazilian Army. As one defining aspect of a coding dojo, teams bring actual (as opposed to toy) projects to the training, which helps to ensure the relevance of the effort and of the newly learned methods and technologies (De Oliveira et al., 2018; Sato et al., 2008).

3. THE COURSE

In the following, we describe how we apply the learning concepts that we introduced in the previous section in an undergraduate course on IT project management. The course is required for IS majors and is an elective for several other majors within the school (e.g., computer science and cybersecurity). According to its catalog description, the course covers the "processes, methods, techniques, and tools in managing information technology projects including scope, time, cost, quality, and risk management." Student learning goals relate to the foundations of project management, as well as the practice and application of a broad range of project management principles, skills, and tools. In addition to course-specific topics and concepts, the syllabus emphasizes soft skills related to working in teams, engaging with stakeholders, and building community. Assessments include exams (45%), lab exercises (10%), dojo sessions (20%), quizzes (10%), service learning/peer learning work (5%), and other assignments and activities (10%). In the format described here, two sections of the course have been taught in each of four semester semesters, most recently in Spring 2023. A typical course section comprises 25 upper-level undergraduate students majoring in IS, cybersecurity, and computer science. Table 1 summarizes the major learning phases and course activities that are described in more detail in the remainder of this section.

Learning Phase (Kolb, 1984	Course Activity	Description
Concrete experimentation and reflective observation	Flipped classroom and journaling	Computer-based problem-solving labs: students read and watch video lectures, then practice solving a problem using tools, techniques, and programming. Individual students use journaling to reflect on the activity, its application, and its place in the big picture. In class, students discuss in pairs the use of the concepts etc. in project management. The class concludes with a review. The instructor asks questions and facilitates the discussion.
Abstract conceptualization	Concept mapping	Review and prelude for the dojo sessions: students engage in collaborative concept mapping. The instructor provides a concept repository, models the activity, and provides feedback.
Active experimentation	Project dojos	Project sessions start with introductions and icebreakers. Students complete a project assignment in an 8-day period, including three class sessions. The instructor provides guidelines for brainstorming and work processes and actively observes to counter the "divide-and-conquer" culture.

Table 1. Learning Phases and Course Activities

3.1 Concrete Experimentation and Reflective Observation With Flipped Classroom and Journaling

A typical skill-focused class meeting plan in a flipped-classroom setting starts with required pre-meeting readings and videos that cover the foundational knowledge and content in the form of concepts, definitions, and use cases primarily drawn from freely available books (via O'Reilly Safari), homegrown lecture videos, and videos recommended by our industry collaborators. Class then starts with a brief summary-lecture and is followed by a relatively raw exposure of students to the techniques with hands-on practice exercises that may include Python coding labs (Figure 2), journaling (Figure 3), and in-class discussions (Figure 4).

For instance, to create risk heat maps (Figure 4), we might facilitate a class-level discussion on risks, probability, and impact before moving to a brainstorming activity to identify risks in the context of a specific, assigned scenario. Students then create their own modified version of a risk map, followed by reflections between pairs of students and at the class-level. The session concludes with a class-level review.

To track individual sense-making, students complete reflective journal entries related to the skill-focused activities and receive formative feedback on those entries (Figure 3). In other words, after reviewing the concepts and techniques at the theoretical and descriptive levels and practicing typical use cases, students summarize what they learned in the form of a low-stakes one-page journaling assignment. Each student keeps a single electronic journal that is stored in the cloud (e.g., Google Drive or OneDrive), which can be continually updated by the student and is also accessible and editable by the instructor.

3.2 Abstract Conceptualizations: Concept Mapping to Synthesize Knowledge and Skills

To support higher-level meaning making and as a prelude for the active experimentation in the form of project dojos, our students practice abstract conceptualization. More specifically, they reflect on the concepts and skills covered in the current module with a concept mapping activity, which is a quite flexible learning practice, as discussed in Section 2.2. We have used concept mapping at the individual level (paper and pencil), group level (using software tools such as vue, https://vue.tufts.edu/), and class level (using magnetic boards) (Figure 5).

As discussed earlier (Section 2.2), the concept inventory that is required for the mapping activity may be developed by the students or be provided by the instructor. In our course design, students are given a starting list of concepts and are then asked to grow the concept inventory as needed to identify links and qualify those links with appropriate labels.

Our observations suggest that identifying the labels and grouping the concepts into different clusters are most critical to creative discussion and reflection and require continuous formative assessment provided by the instructor.

3.3 Active Experimentation: Dojo Sessions Focused on Group Deliverables

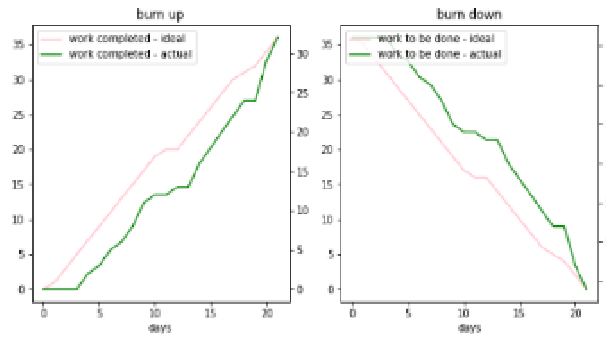
Our course activities culminate in active experimentation (Kolb, 1984) in the form of dojo sessions, during which students complete in-class projects in small groups while the instructor observes and guides the student interactions. In fact,

Burndown/up Chart

Remember last week we talked about velocity as the total work done during a given timebox, the total work can be measured in the number of story points completed.
Burndown charts follow progress of the work in each day, keeping track of the work that has been completed in a given day.
In this data example, we have the

- ideal work progress
- actual progress
- cumulative work done (under ideal/actual scenarios)
- work left to be completed (under ideal and actual scenarios).

Note that we used planned vs. actual cost in our earned value analysis, there is a similar concept here.



read the data

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```python
from google.colab import drive
drive.mount('/content/drive/')

%cd /content/drive/My Drive/
```

```python
myData=pd.read_csv('/content/drive/My Drive/BDU.csv')
```

```python
myData.shape
myData.head(2)
```

plot them

```python
fig, (ax1,ax2)= plt.subplots(1,2, figsize=(40,10))

ax1.set_xlabel('days')
ax2.set_xlabel('days')

#they both show up on the same chart
# work completed under idea scenario
ax1.plot(myData.Cideal, color='pink',label='work completed - ideal')
ax3=ax1.twinx()
# work completed under actual scenario
ax3.plot(myData.Cactual,color='green', label='work completed - actual')

ax2.plot(myData.Lideal, color='pink',label='work to be done - ideal')
ax4=ax2.twinx()
ax4.plot(myData.Lactual, color='green', label='work to be done - actual')

ax1.set_title('burn up')
ax2.set_title('burn down')

h1, l1 = ax1.get_legend_handles_labels()
h3, l3 = ax3.get_legend_handles_labels()
ax3.legend(h1+h3,l1+l3,loc=2, prop={'size':20})
h2, l2 = ax2.get_legend_handles_labels()
h4, l4 = ax4.get_legend_handles_labels()
ax4.legend(h2+h4,l2+l4,loc=2, prop={'size':20})
```

Figure 2. Example of a Python-Based Coding Assignment

we reserve a sizeable portion of the overall class time for the dojo sessions and set up the other activities (flipped classroom, journaling, concept mapping) with the intent to prepare the students for the intense interactions that occur during the dojo sessions while countering typical limitations of small groups, such as a divide-and-conquer culture.

The dojos are intended to be learning spaces for collaboration, where students engage in intensive work on the group project. During the sessions, students must practice communication, problem solving, and evaluation and integration of ideas to construct a coherent unit of work. Because they are timed and involve intensive work, the dojos have some resemblance of agile sprints. However, across the time dimension, dojos are shorter, and a given sprint may contain multiple dojos. The purpose of dojos is a focus on long-term skill development and immersion through intensive collaboration that is intended to go beyond a single sprint or project. To our knowledge, dojos have rarely (if at all) been used in IS education or IT project management courses.

After initially allocating six weeks for the group projects, we now give each group eight working days, which includes three 75-minute class sessions, to complete an assignment with the expectation that most of the work is done during class time.

Lab 3 : Alternative Analysis (AA) - Related: Decision or Peugh Matrix	
1. Copy & paste this table and place it at the top of your journal - Do NOT create or submit a new journal, same file, append new information 2. Make a copy of this file for Lab 3, name it as JaneD_AA 3. Submit completed Lab3 file as a separate link in DropBox	
Concept	Your responses/notes
Alternative analysis schema Or decision matrix	1. Consists of a list of criteria to compare and 2.
When do we create a decision matrix?	
Who should come up with the list of criteria?	
Relative weighting: what's the major feature of relative weighting	Weights must add up to …
Who sets the scoring scale?	The team must agree on a scoring system. Determine the scoring range (e.g., 1 to 5 or 1 to 10) and ensure that all team members have a common understanding of what high, medium, and low scores represent.
How are partial scores calculated?	
How is the total score calculated for each option?	

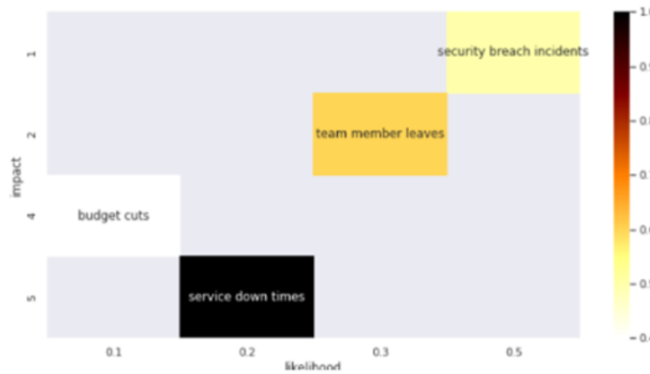
Figure 3. Examples of a Journaling Assignment (Partially Filled In)

https://doi.org/10.62273/WMDE5362

Risk Heat Map

Individual work: After you created the risk pivot table and heatmap using Seaborn package and the code guide do the following:

1. Reflect on a project you have done in the past, and the top risks involved with the work, add those to the map after qualifying the likelihood and impact on the project's processes and outcomes.
2. Reflect on a given risk (e.g., security breach), could the same risk be assessed and qualified differently across two organizations? Or in two different projects?



Partner work: with a partner, examine your map and discuss the following questions. Be ready to share with the class at the end.

1. Explain why a heatmap could be useful during the project's lifecycle?
2. What do you think would happen if risks are not qualified for their likelihood and impact?
3. How are the concepts of risk qualification and heatmap similar to the steps we applied in the decision matrix to compare alternative project solutions?
4. Some organizations add a 3rd dimension to their risk qualification process and that is 'Maturity' level, i.e., maturity of organizational processes for dealing with risk of that kind, how would you use maturity to advance your risk management process?
5. And when during the project life cycle would you think the map is created or revised?

Figure 4. Example of an Assignment for Individual and Pairwise Reflection

A typical project presents a scenario, such as housing for the first Mars colony. Students pick a focus area (e.g., living conditions) for which they create a business case, receive approval, and compose a project charter as their main deliverables (Figure 6). Students also perform the three agile ceremonies of setting daily goals, doing daily scrums, and performing a retrospective (Figure 7), and they are given a series of discussion ideas to focus on and questions to answer for each in-class project dojo session (Figure 8). Student interactions happen mainly on MS Teams to allow the instructor to observe the level of engagement of each group member and to assess group processes.

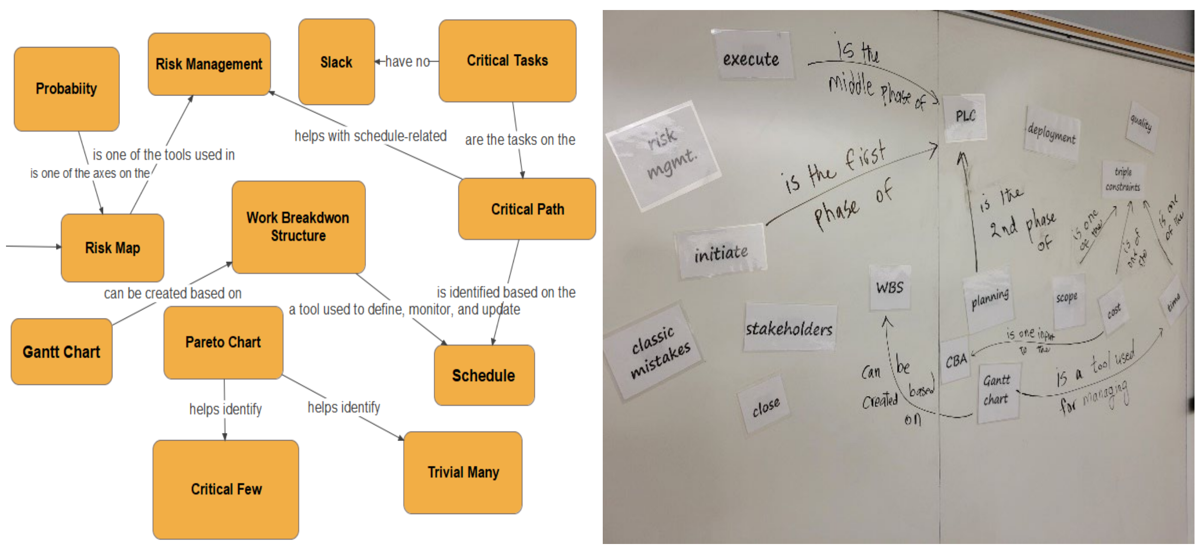


Figure 5. Left: Partial Deliverable of a Collaborative Concept Mapping Activity; Right: Whiteboard Modeling and Practicing Connecting Concepts and Labeling Links

Following the skill-based labs, concept maps, group work, and readings/journaling, we assessed student learning with two exams which accounted for 45% of the grade. The exams included readings (concepts) (40%) and skills (60%) (e.g., risk map completion/interpretation, cash-flow analysis). The class activities and journaling practices were closely aligned with the course's assessment tools, i.e., exams.

Deliverable 1 – Business Case

1- **Write a business case for the project**
   a. List 2-3 MOV
   b. List two alternative solutions
   c. Discuss advantages and disadvantages of the two alternatives
   d. Perform cost-benefit analysis on the two alternatives (labs 2)
   e. Create an alternative analysis schema to compare the alternatives (lab 3)
   f. Perform alternative analysis and make a recommendation

Deliverable 2 – Project Charter

2- **Write a project charter for your proposed/approved solution**
   a. Use sample project charter that you created before as a reference (simplify when needed)
   b. Include in- and out- of scope items
   c. Include phases and major deliverables (lab 1)
   d. Discuss stakeholders and their role in your project
   e. Include a risk map (and if there's space risk table, table not required) (lab 4)
   f. Attach a WBS & Gantt chart (high level - only phases and major deliverables) (lab 1)
   g. Include brief statements about any other project management knowledge area not listed above
   h. *** include your quality management metrics *** (lab 5)
   i. Discuss project metrics (lab 6)

Figure 6. Instructions for Project Deliverables Business Case and Project Charter

Agile Ceremony	Description	Number of Occurrences	Time-line
All your group communication must happen on MS Teams chat I will observe your communication for group processes			
Daily scrums	5-7 minutes quick check-ins, sharing progress & obstacles	6: once every weekday	
Retrospective	Review & reflect, identify areas of improvement	Once	
Setting weekly goals	Have small goals for each day during this 1 week; example: write a 50-minute description or create a table with initial numbers	6: once every weekday	

Figure 7. Instructions for Agile Ceremonies

Suggestions for questions to ask when brainstorming with the teams:	
Topic	Questions/Discussion Ideas
Components of the Project	Start with training as game design + content development. Are there any other components?
Measurable Organizational Values	Think about the VR training, with your information systems project in place, what would you be able to improve? Would you make anything faster, more reliable, easier, more efficient, reduce cycle time and latency, or make more of something or offer more of a service? See the impact areas from the book (at the bottom of this page) and brainstorm what your proposed system will be able to achieve. Make sure each MOV has a metric (how you measure progress), quantify goals (x% increase/decrease), then associate a time frame to the goal (when would you anticipate achieving it?).
Alternative Options	You are in charge of the information systems project, so if you're comparing alternative ways of doing the information systems project, would you think your organization has the skills to develop the system in-house? What are the advantages and disadvantages? The case asks you to consider outsourcing the whole system, what are the benefits and disadvantages of outsourcing?

Figure 8. Suggested Questions and Discussion Items Provided by the Instructor

4. TEACHING SUGGESTIONS

Based on our experience with the three-pronged approach described, we offer some general suggestions related to its application, as well as suggestions that are specific to teaching an IT project management course. The components of agile ceremonies, guided discussions, and observation of group interactions that we employed throughout the course, particularly in dojo sessions, are consistent with the premise that experiential learning and peer interactions must be well-structured, guided, and observed (Kolb, 1984; Topping et al., 2017). Our design accounts for the four dimensions emphasized by Cooper (2002) in that (1) the classroom environment was participatory with a balanced focus on individual skills and group work; (2) we employed technology to make learning trackable and providing formative feedback possible; (3) knowledge, skills, and their method of retention and practice were well chosen, and the instructor assumed an active role in the individual learning (formative feedback) and group processes.

Based on several semesters replicating the same setup in face-to-face, hybrid, and online (post-COVID) courses, we have the following teaching suggestions:

(1) **Flipped classroom and journaling for foundational knowledge building**: Course material, plans, and video lectures should be made available to students before the topics are covered in class. Journaling is structured by the instructor to ensure that students develop and reflect on the foundational knowledge before the low-stakes lab exercises. See Figure 2 for an example of a journaling assignment with some information already filled in.

(2) **Low-stake lab exercises and pair discussion**: Students should be allowed to focus on a specific skill (e.g., creating project schedules), reflect on what they have learned, and document their learning through continued journaling (e.g., revisiting and editing previous entries). The instructor provides formative feedback on the skills and the documentation.

(3) **Concept mapping**: At the small group- and class-levels, students should be engaged in meaning-making at a higher-level by creating and qualifying relationships among concepts, techniques, and methods that they have learned.

(4) **Dojo designs**: The dojo sessions allow for group projects with specific goals. The group process is guided, structured, and supervised by the instructor. The sessions should be set up such that all major creative work for the group project can be completed during class with group collaboration.

(5) **Technology choice**: For group coordination, progress documents, and individual work that was not creativity-intensive (e.g., writing descriptions for pictures), we used MS Teams. However, any platform that allows for electronic communication, file sharing, and messaging should be applicable.

The following are specific suggestions for teaching an IT project management course. The offerings of the course described here included work on both the Project Management Institute (PMI) Project Management Body of Knowledge (PMBOK) ten knowledge areas, and PMI Agile Certified Practitioner (ACP)'s seven domains.

(1) Set reading expectations for before the class and track progress using students' journaling activities. Utilize cloud-based services to avoid excessive submission and deadline handling.

(2) Design and deliver low-stakes, skill-based activities related to each knowledge area and/or domain. Examples include PMBOK's risk map and risk mitigation table, as well as stakeholder management table & discussion and PMI-ACP's story sizing and Weighted Shortest Job First. These activities are expected to take 15-30 minutes and are done in-class in a pass/fail style with the instructor providing guidance and feedback.
(3) Help students "avoid spending too much time on the fuzzy front end" (Nelson, 2007, p. 71). Support project initiation and the planning phase by scaffolding and providing sample questions and discussion items for that phase in the same day. This approach counters the common practice of archiving projects for outside-classroom meetups.
(4) Support synthesizing and connecting by using concept maps; the ten PMI PMBOK knowledge areas, associated processes and tools, and the seven PMI-ACP domains must all come together to equip students with a big-picture view of choices, techniques, and their relationship to the project elements and needs. Because a diverse set of ideas is introduced in PM courses, a concept map is especially helpful.
(5) Document-sharing and progress tracking tools will help. MS Teams offers free Kanban boards and is consistent with policies of many universities, so we used that.

5. EVIDENCE

In Fall 2021 and Spring 2022, two cohorts of 30 students answered questions about the flipped classroom-style Python skill sessions, collaborative concept mapping, and project dojo sessions. Flipped classrooms are especially useful for courses such as project management in which many new terms and concepts are introduced and a basic understanding of them allows more effective skill-focused activities during the class. As summarized in Table 2, among the three practices, the project dojos were perceived as the most helpful by the students, followed by collaborative concept mapping, and then coding labs. The students were also asked to suggest areas of improvement and elements that needed to be added or removed from each of the three components of the course design.

The approach of intensive work in class during the dojo sessions was received well by the students, as noted in several comments made in the open-ended questions of the survey. Students echoed the sentiment that there was no wasted effort in the work done on the project during dojo sessions. They shared that while the hardest part of a group project is always the scheduling and getting everyone coordinated, dojo sessions' requirement of having most of the work done in class helped remove that obstacle and significantly reduced slacking behaviors. There were notes from students explaining slacking still happened, and they suggested even more guidance was needed to counter persistent slackers.

The sentiment was generally positive for the collaborative concept mapping activity, as shared by the students: "…I thought it was a great way to wrap up a lot of the concepts that we've been talking about and I thoroughly enjoyed working with my group;" the general sentiment was that concept mapping helped students "...take a wide look at everything..." in order to gauge their progress and identify gaps. However, there was also the desire to spend more time on the activity, to allow students to continue working on it as homework, and to do the activity longer because "... it helped me understand how all of the concepts fit together."

Activity	1-5 Likert scale (1: very negative, 5: very positive)	Average (StdDev)*
Computer-based problem-solving labs (flipped classroom-style)	Impact of coding labs on the understanding of project management techniques	3.80(.81) 4.20 (.81)
	Impact of coding labs on the overall value of the course for you	4.03 (.81) 4.20 (.76)
	Impact of coding labs on your learning process	3.77 (.94) 4.23 (.73)
	Impact of coding labs on your view of the project management course as whole	3.97 (.72) 4.17 (.75)
Concept mapping activity	Impact of concept mapping on understanding the relationships among project management concepts, terms, and techniques	3.87 (.68) 3.94 (.81)
	Impact of concept mapping on creating a big-picture view of the course material	3.97 (.61) 4.01 (.75)
	Impact of concept mapping on your learning process	3.87 (.68) 4.13 (.74)
	Impact of concept mapping on your ability to put together or synthesize what you've learned	3.87 (.86) 3.70 (.83)
Project dojo sessions	Impact of dojo sessions' structure/guidelines on group planning	4.33 (.66) 4.13 (.73)
	Impact of dojo sessions' structure/guidelines on group coordination	4.30 (.70) 4.03 (.85)
	Impact of dojo sessions' structure/guidelines on group productivity	4.23 (.73) 4.03 (.85)
	Impact of dojo sessions' design/structure on members' collaboration level	4.07 (1.01) 4.23 (.82)
	Impact of dojo sessions' design/structure on members' contribution level	4.37 (.76) 4.17 (.83)

*N1=30 (Fall 2021); N2=30 (Spring 2022)

Table 2. Student Perceptions of Course Components

Students' perceptions of the flipped classroom-style Python skills labs were mixed but skewed toward positive as noted in these comments because they were very fun, easy to do, and "…felt very beginner friendly." There were concerns related to the speed of the in-class exercises, lack of prior coding experience, not fully understanding areas of code, and the need for deeper exploration of programming constructs. The comments suggest the need for a more structured approach to the labs. This would allow students to engage in short activities before and after the skills sessions, which would help them feel more comfortable during the sessions and guide them on how to advance the fast-paced work they complete in class. The course is taken by different majors within the IS field as an elective, and the preferences for a coding-heavy or coding-light experience are shown clearly in the mixed comments.

In the most recent delivery of the course, we further evolved the course components to accommodate the mixed preferences of students for coding, adjust the design of the concept-mapping sessions, and improve the dojo sessions according to frequent themes that emerged from students' perceptions. More challenge activities (optional follow-up work) were added to the computer-based problem-solving labs. We also increased the frequency of the concept-mapping sessions and added group ground rules to counter the presence of bystanders (students who attend the sessions and sit with the team but do not contribute). Sample rules are: "… (2) to earn your participation points you must actively participate. Just being in class and staying on phones or not participating does not count. (3) Your peer interactions and work during the sessions will be closely observed; do all the heavy lifting in groups, small details can be polished or individual pieces can be worked on between sessions. (4) Practice democracy and hearing other ideas: the dojo sessions should not be focused on one person's idea, you must strive to be a team member, share your ideas, and also actively listen to others' ideas…"

6. CONCLUSIONS

In this teaching tip, we outlined a comprehensive and effective structure for an IT project management course that seeks to provide students with support for acquiring content knowledge as well as group working skills. More specifically, our concept facilitates experiential and peer learning in the form of individual and pair-wise skill-building activities in a flipped-classroom setting that is augmented by journaling and hands-on activities, followed by higher-level meaning making in the form of concept mapping, and culminating in project dojo sessions that are intended to (1) allow the instructor to observe, guide, and improve peer learning practices in the groups, (2) limit the amount of time wasted on group coordination, and (3) counter the divide-and-conquer culture in group work. Although individuals make meanings in their own minds, peer interactions impact the cognitive processes within individual minds (O'Donnell & King, 1999; Piaget, 1970). The classroom activities described were found to be effective as they enabled experiential and peer learning in a systematic, guided, and IT-enabled classroom structure.

We believe that our approach is beneficial for IT courses such as IT project management and IS practicums that emphasize experiential and peer learning. We also believe that enhancing experiential and peer learning is a fruitful effort with impacts that can last beyond the scope of a given course.

7. REFERENCES

AACSB (2023). *2020 Guiding Principles and Standards for Business Accreditation* (updated July 1). AACSB. https://www.aacsb.edu/-/media/documents/accreditation/%202020-aacsb-business-accreditation-standards-june-2023.pdf?rev=d31cfbe864e54792816ff426fe913e65&hash=33A159779F107443A64BDACBBB7000C5

ABET Computing Accreditation Commission (2023). *Criteria for Accrediting Computing Programs*. ABET. https://www.abet.org/wp-content/uploads/2023/05/2024-2025_CAC_Criteria.pdf

Bergmann, J., & Sams, A. (2012). *Flip Your Classroom*. International Society for Technology in Education. https://www.rcboe.org/cms/lib/ga01903614/centricity/domain/15451/flip_your_classroom.pdf

Boud, D., Cohen, R., & Sampson, J. (Eds.) (2001). *Peer Learning in Higher Education: Learning from and with Each Other*. Kogan Page. https://doi.org/10.4324/9781315042565

Brooks, F. P. (1974). The Mythical Man-Month. *Datamation*, 20(12), 44-52. https://doi.org/10.1109/MS.1995.10042

Cooper, S. M. A. (2002). Classroom Choices for Enabling Peer Learning. *Theory Into Practice*, 41(1), 53-57. https://doi.org/10.1207/s15430421tip4101_9

Croasdell, D. T., Freeman, L. A., & Urbaczewski, A. (2003). Concept Maps for Teaching and Assessment. *Communications of the Association for Information Systems*, 12(1), 24. https://doi.org/10.17705/1CAIS.01224

De Miras, J. R., Balsas-Almagro, J. R., & Garcia-Fernandez, A. L. (2022). Using Flipped Classroom and Peer Instruction Methodologies to Improve Introductory Computer Programming Courses. *Computer Applications in Engineering Education*, 30, 133-145. https://doi.org/10.1002/cae22447

De Oliveira, C. M. C., Canedo, E. D., Faria, H., Amaral, L. H. V., & Bonifácio, R. (2018). Improving Student's Learning and Cooperation Skills Using Coding Dojos (In the Wild!). *2018 IEEE Frontiers in Education Conference (FIE)* (pp. 1-8). https://doi.org/10.1109/FIE.2018.8659056

Frazee, A. (2021). Teaching Collaboration: From "Divide and Conquer" to "More Than the Sum of Its Parts." In D. P. Domizi (Ed.), *Engaged Student Learning: Essays on Best Practices in the University System of Georgia* (vol. 3, pp. 35-36). https://www.usg.edu/teaching-and-learning-excellence/assets/facultydevelopment/documents/ebook/FullVolume3.pdf

Freeman, L. A., & Urbaczewski, A. (2001). Using Concept Maps to Assess Students' Understanding of Information Systems. *Journal of Information Systems Education*, 12(1), 3-8. https://jise.org/Volume12/n1/JISEv12n1p3.pdf

Gregoriades, A., Pampaka, M., & Michail, H. (2009). Assessing Students' Learning in MIS Using Concept Mapping. *Journal of Information Systems Education*, 20(4), 419-430. https://jise.org/volume20/n4/JISEv20n4p419.pdf

Hiemstra, R. (2001). Uses and Benefits of Journal Writing. In L. M. English & M. A. Gillen (Eds), *Promoting Journal Writing in Adult Education. New Directions for Adult and Continuing Education*, 90, 19-26. Jossey-Bass. https://doi.org/10.1002/ace.17

Hellström, T., & Husted, K. (2004). Mapping Knowledge and Intellectual Capital in Academic Environments: A Focus Group Study. *Journal of Intellectual Capital*, 5(1), 165-180. https://doi.org/10.1108/4691930410512987

Heinonen, K., Hirvikoski, K., Luukkainen, M., & Vihavainen, A. (2013). Learning Agile Software Engineering Practices Using Coding Dojo. *Proceedings of the 14th Annual ACM SIGITE Conference on Information Technology Education*.

Association for Computing Machinery, New York, NY, USA, pp. 97-102. 9781450322393. https://doi.org/10.1145/2512276.2512306

Johnson, D. W., Johnson, R. T., & Smith, K. A. (1998). Cooperative Learning Returns to College: What Evidence Is There That It Works? *Change: The Magazine of Higher Learning*, 30(4), 26-35, https://doi.org/10.1080/00091389809602629

Kapp, E. (2009). Improving Student Teamwork in a Collaborative Project-Based Course. *College Teaching*, 57(3), 139-143. https://doi.org/10.3200/CTCH.57.3.139-143

King, A. (1990). Enhancing Peer Interaction and Learning in the Classroom Through Reciprocal Questioning. *American Educational Research Journal*, 27(4), 664-687. https://doi.org/10.3102/00028312027004664

King, A. (2002). Structuring Peer Interaction to Promote High-Level Cognitive Processing. *Theory Into Practice*, 41(1), 33-39. https://doi.org/10.1207/s15430421tip4101_6

Kolb, D. A. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice-Hall. https://www.syndetics.com/index.php?isbn=0132952610/toc.html&client=uncwh&type=rn12

Leidig, P., & Salmela, H. (2021). *IS2020 – A Competency Model for Undergraduate Programs in Information Systems*. The Joint ACM/AIS IS 2020 Task Force. https://doi.org/10.1145/3460863

Middleton, T. (2024). *The Importance of Teamwork (As Proven by Science)*. Worklife by Atlassian. https://www.atlassian.com/blog/teamwork/the-importance-of-teamwork

Nelson, R. R. (2007). IT Project Management: Infamous Failures, Classic Mistakes, and Best Practices. *MIS Quarterly Executive*, 6(2), article 4. https://aisel.aisnet.org/misqe/vol6/iss2/4/

Novak, J. D., & Musonda, D. (1991). A Twelve-Year Longitudinal Study of Science Concept Learning. *American Educational Research Journal*, 28(1), 117-153. https://doi.org/10.3102/00028312028001117

Novak, J. D., & Cañas, A. J. (2008). *The Theory Underlying Concept Maps and How to Construct and Use Them*, Technical Report IHMC CmapTools 2006-01 Rev 01-2008, Florida Institute for Human and Machine Cognition. https://cmap.ihmc.us/publications/researchpapers/theoryunderlyingconceptmaps.pdf

O'Donnell, A. M., & King, A. (Eds.). (1999). *Cognitive Perspectives on Peer Learning*. Routledge. https://doi.org/10.4324/9781410603715

Palincsar, A. S., & Herrenkohl, L. R. (2002). Designing Collaborative Learning Contexts. *Theory Into Practice*, 41(1), 26-32. https://doi.org/10.1207/s15430421tip4101_5

Piaget, J. (1970). *Science of Education and the Psychology of the Child*. New York: Viking.

Pollard, C. E. (2012). Lessons Learned from Client Projects in an Undergraduate Project Management Course. *Journal of Information Systems Education*, 23(3), 271-282. https://jise.org/Volume23/n3/JISEv23n3p271.pdf

Qurban, M. H., & Austria, R. D. (2009). Improving the Communication Skills of IS Developers During Requirements Elicitation Using Experiential Learning. *Journal of Information Systems Education*, 20(3), 301-311. https://jise.org/Volume20/n3/JISEv20n3p301.pdf

Roehl, A., Reddy, S. L., & Shannon, G. J. (2013). The Flipped Classroom: An Opportunity to Engage Millennial Students Through Active Learning Strategies. *Journal of Family and Consumer Sciences*, 105(2), 44-49. https://doi.org/10.14307/JFCS105.2.12

Sato, D. T., Corbucci, H., & Bravo, M. V. (2008). Coding Dojo: An Environment for Learning and Sharing Agile Practices, *AGILE Conference* (pp. 459-464). Los Alamitos, CA, US: IEEE Computer Society. https://doi.org/10.1109/Agile.2008.11

Scrum Alliance (2021). *Enter the Dojo: Target's Innovative Agile Training Ground*. Scrum Alliance. https://resources.scrumalliance.org/Article/enter-the-dojo

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, Jr., J. F., Sipior, J. C., & de Vreede, G. J. (2010). IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. *Communications of the Association for Information Systems*, 26(1), 359-428. https://doi.org/10.17705/1CAIS.02618

Topping, K. J. (2005). Trends in Peer Learning, *Educational Psychology*, 25(6), 631-645, https://doi.org/10.1080/01443410500345172

Topping, K., Buchs, C., Duran, D., & Van Keer, H. (2017). *Effective Peer Learning: From Principles to Practical Implementation*. Taylor and Francis. https://doi.org/10.4324/9781315695471

Tucker, B. (2012). The Flipped Classroom. *Education Next*, 12(1), 82-83. https://www.educationnext.org/wp-content/uploads/2023/04/ednext_XII_1_what_next.pdf

Van Boxtel, C., van der Linden, J., Roelofs, E., & Erkens, G. (2002). Collaborative Concept Mapping: Provoking and Supporting Meaningful Discourse. *Theory Into Practice*, 41(1), 40-46. https://doi.org/10.1207/s15430421tip4101_7

Vygotsky, L. S. (1978). *Mind in Society: Development of Higher Psychological Processes*. Harvard University Press.

Wei, W., & Yue, K. (2017). Concept Mapping in Computer Science Education. *Journal of Computing Sciences in Colleges*, 32(4): 13-20. https://doi.org/10.5555/3055338.3055341

Yildiz Durak, H. (2022). Flipped Classroom Model Applications in Computing Courses: Peer-Assisted Groups, Collaborative Group and Individual Learning. *Computer Applications in Engineering Education*, 30(3), 803-820. https://doi.org/10.1002/cae.22487

Zheng, L., Bhagat, K. K., Zhen, Y., & Zhang, X. (2020). The Effectiveness of the Flipped Classroom on Students' Learning Achievement and Learning Motivation: A Meta-Analysis. *Educational Technology & Society*, 23(1), 1-15. https://www.jstor.org/stable/26915403

https://doi.org/10.62273/WMDE5362

AUTHOR BIOGRAPHIES

Elahe Javadi is an associate professor of information systems

at the School of Information Technology at Illinois State University. Her current research focuses on applied AI techniques, collaborative work in virtual reality, and information systems pedagogy. She teaches courses in systems development, project management, IT/AI strategy and policy, Python programming, applied machine learning, and computer architecture. She is also passionate about enhancing IS students' competencies in development life cycles and machine learning pipelines involving both hardware and software.

Judith Gebauer is a professor of information systems in the

Congdon School of Supply Chain, Business Analytics, and Information Systems, which is part of the Cameron School of Business at the University of North Carolina Wilmington. Her current research focuses on the management of information systems and new technologies, including artificial intelligence and virtual reality, stakeholder analysis, and IS pedagogy. Areas of teaching include strategy, management, and governance of IS resources in the organization, systems analysis, project management, and introductory IS courses. She is also interested in curriculum development at the undergraduate and graduate levels.

Season Tanner is a Group Product Manager for State Farm

Insurance. She has worked in IT for the last 20 years, providing leadership for the State Farm Agile & Product Management Dojo, many large scale, high visibility software development initiatives, and coaching many different teams to high performance. She has been a featured speaker at AgileDev conferences, and the Scrum Gathering and is known for using creative games to help others improve their performance and results with Agile and product management best practices.

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