Chapter 9
Using Active Learning to Teach Software Engineering in Game Design Courses

Bruce R. Maxim and Jeffrey J. Yackley

Abstract Game developers are beginning to understand it is important to approach computer game design like how all software engineers approach projects involving large numbers of people and significant investment of time. Engineering instructors often rely on the traditional lecture model when they teach topics to a classroom of students. Students often fail to engage with the material presented by lecturers. Many engineering educators regard experiential learning as an effective way to train future generations of engineers and game developers. The authors have created two courses that focus on software engineering and game development. These courses were initially offered as traditional lecture classes to both in-person and online groups of students. This chapter describes the authors' approaches to revising these game design classes to make use of flipped classroom models that rely on active learning, role-play, and gamification to cover software engineering topics in these courses. Students learn to use Agile software engineering practices to design, implement, and test game prototypes. In-person students were surveyed to measure their perceived levels of engagement with course activities. Our assessment data suggests that students attending flipped class meetings were slightly more engaged with the course materials than those taking the class offered using lectures only. Students interacting with the active learning course materials felt better able to apply their knowledge than students in a traditional lecture course.

Keywords Active learning · Student engagement · Role-play · Game design · Agile development

B. R. Maxim (✉)
University of Michigan-Dearborn, Dearborn, MI, USA
e-mail: bmaxim@umich.edu

J. J. Yackley
University of Michigan-Flint, Flint, MI, USA
e-mail: jyackley@umich.edu

K. M. L. Cooper, A. Bucchiarone (eds.), *Software Engineering for Games in Serious Contexts*, https://doi.org/10.1007/978-3-031-33338-5_9

9.1 Introduction

Engineering instructors often rely on the traditional lecture model where they cover a topic, with or without a slideshow, to a classroom of students. Students often fail to engage with the material presented by lectures until an assessment activity is near. Many engineering educators regard experiential learning as the most effective way to train future generations of engineers and game developers. The authors have noticed higher levels of engagement when students participate in class activities rather than passively listening to lectures. These activities may include games, discussions, role-play, peer reviews, and group problem-solving or design exercises. This chapter describes the authors' approach to revising two lecture heavy game design courses to make use of a flipped classroom model that relies on active learning, role-play, and gamification to cover software engineering topics in game design courses.

Covid-19 restrictions forced a shift to the online delivery of all courses at our university in 2020. In Fall 2021, face-to-face class meetings were allowed if vaccination, masking, and social distancing were enforced. Often, activities developed for face-to-face delivery of software engineering topics cannot be used without modification in the online delivery of course materials. Following Covid protocols in face-to-face classes also required modification of active learning course materials.

Students learning software engineering principles and practices may find it difficult to apply them in the development of complex software projects. Software engineering involves acquiring application domain knowledge to understand the client's needs. It is therefore important to do more than simply use a game as the term project in a software engineering course as some authors have suggested [1–3]. Additionally, adding game topics to already crowded software engineering courses, as some authors have advised [3, 4], requires sacrificing important software engineering topics. Focusing on one application area in the first software engineering class is not fair to all students as not every software engineering student wants to become a game developer.

9.2 Background

Game developers are beginning to understand that it is important to treat computer game design in the same way that other software engineers approach projects involving large numbers of people and a significant investment of time [5]. Game developers can benefit from using evolutionary software process models to manage their development risks and reduce their project completion times. The process of determining the technical requirements for a game software product is like that used to specify any other type of software product. However, unlike most software products, games have an entertainment dimension. People play computer games because games are fun [6].

The authors believe that the capstone design course should not be the only opportunity for students to manage complex software development projects. This suggests the use of other courses in the curriculum such as a game design course as a means of providing additional software engineering experiences. This paper describes the authors' experiences revising and employing active learning materials to teach software engineering content in a sequence of two face-to-face game design courses with or without social distancing and online either synchronously or asynchronously spanning a 6-year period.

9.2.1 Active Learning

Engineering educators regard experiential learning as the best way to train the next generation of engineers [7]. Toward this end, it is reasonable to believe that the interaction practiced in active learning classrooms can improve software engineering education at the undergraduate level and better prepare students for the experiential learning that comes with their capstone projects [8].

Active learning is "embodied in a learning environment where the teachers and students are actively engaged with the content through discussions, problem-solving, critical thinking, debate and a host of other activities that promote interaction among learners, instructors and the material" [9]. Prince defines active learning as any classroom activity that requires students to do something other than listen and take notes [10]. Active learning opportunities can complement or replace lectures to make class delivery more interesting to the students. Active learning using flipped classes can also foster developing an attitude of lifelong learning among students [11].

Specifically, active learning helps students develop problem-solving, critical reasoning [12], and analytical skills, all of which are valuable tools that prepare students to make better decisions, become better students, and better employees [10]. Raju and Sankar undertook a study to develop teaching methodologies that could bring real-world issues into engineering classrooms [13]. The results of their research led to recommendations to engineering educators on the importance of developing interdisciplinary technical case studies that facilitate the communication of engineering innovations to students in the classroom.

Active learning helps students learn by increasing their engagement in the process [14, 15]. Active learning techniques help students better understand the topics covered in the curriculum [16]. Active learning also helps students be more excited about the study of engineering than traditional instruction [17]. The group work that often accompanies active learning instruction helps students develop their soft skills [18] and makes students more willing to meet with instructors outside of class [19]. Krause writes that engagement does not guarantee learning is taking place, but learning can be enhanced if it provides students with opportunities to reflect on their learning activities [20].

There is consensus among members of our department's professional advisory board that professional practice invariably requires strong verbal and written communication skills. To develop their oral communications skills, students need opportunities to present their work as well as observe their peers doing the same. Some instructors believe that the project activities inherent in real-world software development encourage students to improve their written and oral communication skills [21].

Day and Foley used class time exclusively for exercises by having their students prepare themselves through the study of materials provided online [22]. Bishop and Verleger presented a comprehensive survey of flipped classroom exercise implementations [23]. Wu et al. effectively implemented class exercises as active learning tools in their flipped classroom approach [24]. Research suggests that the success of flipped classroom approaches depends on the nature of the course being taught. Learning content after engaging in course activities can be easier for some students [25]. The investment in time required for instructors to develop quality out-of-class materials and in-class active learning experiences can be substantial [26].

The active learning approach of problem-based learning (PBL) has consistently been demonstrated to lead to positive learning outcomes such as self-directed learning habits, problem-solving skills, and deep disciplinary knowledge while engaging students in collaborative, authentic, and learning situations [27]. While PBL was first incorporated into medical school curricula in 1969, it is currently used in a wide variety of courses [28]. For instance, within the field of engineering, Warnock and Mohammadi-Aragh investigated the impact of PBL on student learning in a biomedical materials course and found that students made significant improvements in their problem-solving, communication, and teamwork skills [29].

PBL has also been used in senior-level engineering courses with the same positive results [30–32]. Although students in a PBL software engineering course reported that the projects were more time intensive than a typical course project, they were receptive to the approach since they thought it was related to the professional environment and provided them with opportunities to relate theory and practice. This contrasted with students taught using a traditional lecture and project approach to the course who viewed completing a traditional course project more negatively [33].

9.2.2 Student Engagement

Active learning techniques such as think-pair-share exercises [34], pair programming [35], peer instruction [36], and flipped classrooms [37] have been demonstrated to increase student engagement [11]. Many of these interventions are used in introductory-level instruction, primarily to address broadening participation in large classes [38]. Admittedly, lack of access to technology to create and access the videos needed to flip a classroom can pose challenges to both students and teachers [26].

Ham and Myers introduced process-oriented guided inquiry learning (POGIL) into a computer organization course [39]. In software engineering courses, the use of real-world, community-based projects may be an effective way to engage students with a meaningful problem while teaching them software engineering concepts [40]. Students often become more invested in their projects when they see that their products are more than simply a paper design. In our course redesign, we used the class activities to motivate students to design game software products and use software engineering techniques to solve real-world programming problems.

An important aspect of software engineering education is the development of soft skills such as communication and project management. There are several examples of courses that make use of project work to help students enhance their soft skills simultaneously with their software development skills [41]. Decker and Simkins [42] introduced the use of an extended role-play approach in a game development process class where the students were not assessed solely on the artifacts they produced but the processes by which they created their artifacts. Their role-play activities emphasize industry best practices for both technical and soft skills (project management, communication, marketing, and interdisciplinary design).

9.2.3 Role-Play

Simkins [43] defines role-play as simulating the real world in environments where consequences can be mitigated safely. Role-play allows students to get hands-on practice with engineering concepts and practice the soft skills that make for successful professional engineers: communication, problem-solving, and analytical skills. We believe this makes role-play a critical tool in the active learning engineering classroom. Numerous researchers have investigated the use of role-play in the software engineering classroom with success.

Moroz-Lapin [44] and Seland [45] used role-play in human-computer interaction courses to engage students with the requirement engineering process to better understand system behavior from the users' point of view. Similarly, Zowghi and Parvani [46] also investigated requirements engineering using role-play to have their students understand the process of requirements gathering from both the client and developer perspective. Role-play was used by Börstler [47] to teach students object-oriented programming concepts with class-responsibility-collaborator cards. Vold and Yayilgan [48] achieved greater student engagement with role-play in an information technology course. Further, we draw inspiration from a study that used the Second Life online virtual world as a platform for students to role-play a fictional company for enterprise resource planning [49]. Other online role-play simulations focus on students taking the role of project managers with students receiving immediate feedback on their decisions [50–52].

The redesign described in this paper builds upon the work of Maxim, Brunvand, and Decker [53], which used role-play in a re-designed game design course, CIS 488, at the University of Michigan–Dearborn. We re-use this work with some slight modifications as the second course in our two-course game design sequence [54]. The course from 2017 had the students role-play as developers of a failing game company with the goal of simulating concept ideation to creation and release of 3D computer games using Unreal Engine 4. The failing game company backstory used to motivate the role-play in our course is discussed further in Decker and Simkins [42]. Decker and Simkins provide the framework we used to build and adapt our role-play modules. These modules emphasize industry best practices for the technical game development work and soft skills development as well as the introduction of secondary learning objectives based in business and legal concerns that naturally arise during the role-play [54].

9.2.4 Gamification

Gamified learning or the gamification of learning has been defined as the use of game design elements in non-game settings to increase motivation and attention on task [55, 56]. Using active learning in the authors' experience may lead to issues with group participation and motivation if students do not feel the need to work outside of class. Adding gamification elements to active learning can help mitigate this problem.

James Gee [57] has identified 36 learning principles that are present in good games. These learning principles provide the backbone for good game design and, in turn, can be used as guiding principles when designing a gamified learning environment. For instance, good games provide players with information when they need it and within the context in which the information will be used [58]. Effective game design includes challenging players, so they are routinely working at the edge of their abilities and knowledge, also known as their zone of proximal development [59]. Having students, or players, operate within this optimal learning zone helps keep them engaged and encourages them to learn more to meet the demands of the next challenge.

According to Gee [58], games can promote collaboration and skill building, if players are required to share knowledge and skills to be successful. Games that reward teamwork can have a positive impact on the development of prosocial skills [60]. Gee contends that well-designed games are motivational specifically because of the different learning principles outlined previously [58]. Working at the limits of their abilities keeps players engaged as they continue to take on new challenges [61]. Gee refers to this process as a cycle of expertise, which requires players to constantly learn, act, revise, and learn again to demonstrate proficiency and be successful in a game [57].

In addition to the motivational aspect of the cognitive element of games, Lee and Hammer [62] suggest that the social and emotional aspects of rewards and consequences earned in gaming environments contribute to motivation as well. However, there needs to be a balance between positive and negative outcomes to prevent discouraging or overwhelming the students [56]. A well-designed game can also motivate players to stay engaged by enhancing the value of the task or tasks being completed [63]. This is particularly beneficial with educational games focused on school-related subjects that students might not otherwise choose to immerse themselves within. Toth and Kayler [64] created a role-playing game that made use of quests to motivate students' assignment completion.

Gamification can be used as a means of promoting rewards for completing tasks. Students can be rewarded for compliance to software process steps and for taking the initiative to improve their "soft skills."

It is important to acknowledge the debate that centers around gamification. There are critics such as Ian Bogost who colorfully proclaim "Gamification is bullshit" and that it is little more than a marketing term for exploitative practices [65]. A more nuanced criticism from Casey O'Donnell argues that gamification at its heart is a form of algorithmic surveillance that provides data of dubious merit and use [66].

9.3 Proposed Solution

The University of Michigan–Dearborn offers a two-course undergraduate sequence, CIS 487 and CIS 488, in game design. These courses are offered in-person on campus and paired with an online section that allows enrolled students to complete the course requirements asynchronously. Prior to 2017, these involved students attending or observing a 3-h lecture with slides. Little in-class interaction between students was observed with in-person course delivery. In our experience following students throughout the two-semester sequence, most students spent their class time with their laptops more than with the course lecture material [17]. We wanted to change the structure of these courses to better engage the students with the software engineering content covered in these courses. We describe our experiences in altering these courses to include active-learning role-play.

Given all the positive evidence discussed previously, it was determined that a PBL pedagogical approach was well suited for software engineering project courses. In our classes, students are encouraged to reflect on the lessons learned from the activities either in writing or orally during class postmortem discussions.

We included role-play activities in our course redesign to allow students to practice skills such as project management, communication, marketing, and interdisciplinary design. To encourage the development of soft skills, the investigators made use of small group activities with the expectation that students would provide written or oral summaries (either live online or using video) of the strategies used to complete their tasks and their lessons learned. The decision was made to continue to use the term-long role-play activities in CIS 488 since those students had a good grasp of software engineering and game design from the prerequisite courses CIS 487 and CIS 375 Software Engineering 1.

Gamification can be used as a means of promoting rewards for completing tasks. Students can be rewarded for compliance to software process steps and for taking the initiative to improve their "soft skills." In this way, the authors hope to resolve the discrepancies in personal efforts that are often present in student project work. We believe gamification can be accomplished in a non-manipulative and non-exploitative manner where the goal of the gamification is to provide different opportunities for involvement in the courses thereby allowing students to work on what interests them the most.

We designed tasks covering the gamut of game design and engineering process tasks and assigned them point values for successful completion. Students were allowed to negotiate their own tasks within their team structures while also being encouraged to work on a variety of different tasks to earn points toward their final course grade. These tasks encouraged development of soft skills through team communication, planning, and problem-solving. Allowing students to negotiate the nature of their activities and rewards upfront often goes a long way to ensuring that all students are engaged for the entire semester.

When Covid-19 forced us to eliminate or modify the way we offered our in-person game design courses, we developed strategies to improve online student engagement. In 2020 and Winter 2021, our game design classes were offered entirely online. Some students participated in these classes by attending synchronous class meetings using Zoom and completed small-group assignments in breakout rooms. Asynchronous online students watched online videos of class lectures and activities. Starting in Fall 2021, in-person instruction was allowed for students attending classes on campus, if they wore masks, were vaccinated, and followed social distancing rules while in the classroom or lab. Asynchronous online students continued observing class by viewing video recordings 1 day after the class meetings.

9.3.1 Course Overview: CIS 487 Computer Game Design I

The purpose of CIS 487 is to introduce students to the technology, science, and art involved in the creation of computer games. The course meets once a week for 3 h over a 15-week semester. Before the Fall 2017 semester, this course split time between lectures on game design principles and Unity 2D and 3D game engine video tutorials. The revisions to this course focused primarily on introducing active-learning activities on game design as an alternative to a lecture heavy focus for presenting course content. Table 9.1 shows a week-by-week listing of the topics for the course.

The weekly class was taught using a flipped classroom approach and was split into three principal components. The first component was a short interactive presentation on the game design material for the week. These presentations were reduced to 30–45 min on average and were then followed by the second component, an activity designed to engage the students more deeply with the material. Finally, the third component was a 30-min, tutorial video on a particular Unity engine tutorial on a particular topic usually related to the game design content for the week.

Table 9.1 A listing of the weekly topics and activities for CIS 487

Week	Software engineering topic	Activities
1	Game Design Evaluation Intellectual Property	Bartok Rule Changes Exercise Copyright Card Game
2	Game Storylines in Design Puzzle Design Process	Storyline Exercise Shocking Puzzle Design
3	Game Quality Review	Peer Review of Game Review
4	Game and Balance Storyboarding Feasibility Prototypes	Analysis of 3 Dot Game Paper Prototype—Test Feasibility of New First Person Shooter Game Design
5	Design Documents Brainstorming and Pitches Trade-off Analysis	Ideation and One Page Creation Create Game Pitch for One Page Game Analyze Impact of Adding or Removing Features Using Paper Prototypes
6	Formal Technical Reviews Playtesting	Peer Review 2D Pitch Document Playtest 2D Game Feasibility Prototype
7	User Experience Design Agile Development	Revise User Interface Design Process Improvement Game (PIG) Contest
8	UX Sound Design UX Level Design	Create Skit Using 2D Games Sounds Only Create Outline for New 2D Game level
9	2D Game Testing	Peer Review 2D Game Beta Prototype
10	Game AI design Game AI testing	Design New Finite State Game AI for 2D game Test Game AI Using Paper Prototype and Roleplay
11	Game Design Documents Formal Technical Reviews	Peer Review 3D Game Concept Presentations
12	Playtesting and Testing	Create Testing Script for 2D game External Testers use Script to Test 2D game
13	Playtesting	Playtesting of 3D Alpha Prototypes
14	Marketing	Marketing Exercise for 3D Game
15	Quality Assessment	Peer Assessment of 3D Beta Prototypes

The students were evaluated on the completion of five projects, four of which were team-based assignments and one which was an individual assignment. The group assignments involved the use of gamification to reward differential student project contributions that were broken down into elective components each with its own point value. Students could select any number of electives from the assignment to complete to earn a maximum number of points on the assignment. Students also submitted write-ups of the small-group activities completed in class. These write-ups were started in class, completed individually, and submitted for grading.

The first project was an individual review of a professionally produced computer game. Students prepared their reviews of the game and their critiques in a PowerPoint. They were then required to present them to the class. The reviews were to cover the basic information of the game (i.e., title, type, price, authors); a summary of the game, which was to include items such as the story, gameplay, user interface, etc.; and their thoughts on a number of questions such as the quality, fun, comparison to similar games, design mistakes, strengths, and weaknesses.

Projects two and three were completed by a group of three or four with the same students completing both projects together. Students selected their own partners for the projects. The two projects were the creation of a 2D Unity game pitch and the production of the game itself (delivered as two prototypes). The game pitches involved creation of a pitch document that outlined the game story, game play look and feel, and the development specifications. The 2D game required a playable game with at least one playable character, one level transition, and rudimentary physics and AI.

The fourth and fifth projects were also team-based, but the students were required to form teams of four or five individuals. The students again could choose their own partners but were not required to collaborate with the same partner from their 2D game. The fourth and fifth projects were to design and implement a 3D game alpha and beta prototype. The game requirements were like those for the 2D game with the expectation of a more polished and complete game.

9.3.2 Course Overview: CIS 488 Computer Game Design II

The CIS 488 course contains a semester-long role-play in which the students function as the employees of a struggling game company. Also, the course makes use of gamification and active-learning elements as did its predecessor, CIS 487. CIS 488 meets 1 day a week for 3 h over a 15-week semester. Table 9.2 shows the weekly topics and activities. During the first class period, students were introduced to the backstory of the role-play and how it would affect the conduct of the course. In previous offerings of this course, much of the class time was spent observing instructor lectures on Unreal4 programming techniques. In the current course offering, most class time was spent in game design studio role-play activities. Classes often began with an all-hands meeting to introduce the day’s role-playing activities. Students were expected to use video tutorials outside of class to learn to use the Unreal4 Blueprint system and level editor.

The fictitious company created for the role-play had a tradition of using a green light system for continuing or stopping development of game products. The first task was for each company developer to do a quick market research review and create a pitch for an innovative game product. The top five pitches were selected by class vote. The winning pitch authors were allowed to recruit four or five team members during the third class period. Each team was asked to provide a representative for a committee to write a company-wide software process standards document based on the scrum framework. A contest was held within the company to create a new name and logo. The developers selected their favorite, and Imagination Studio was launched.

Table 9.2 A listing of the weekly topics and activities for CIS 488

Week	Software engineering topic	Activities
1	Role-play Introduction	
2	3D Game Pitch Presentation	Peer Green Light Vote Team Formation
3	Software Process Definition	Teams Refine Game Concepts as One Pages Develop Agile Company Process Model
4	Business Plan Creation	Process Model Presentation and Approval One Page Review
5	Formal Technical Reviews	Peer Review of Draft Design Document
6	Elevator Pitches IP Ownership	Creation and Review of Game Elevator Pitch Game Theme Ownership Dispute Activity
7	Contracts and Scope Creep	Two Pitch Swaps Contract Dispute Activity Lens Presentations
8	Playtesting	Peer Review of Alpha Game Prototypes
9	Retrospective Game AI Design	Greenlight Vote on Alpha Prototypes Alpha Retrospective and Beta Planning Lens Presentations
10	Security	Game Espionage Activity Lens Presentations
11	Formal Technical Review Playtesting	Peer Review of Final Game Design Document Playtesting of Beta Game Prototype
12	Software Evolution	Create an Outline for a Game Sequel with Taking Game Asset Reuse into Consideration Lens Presentations
13	Game Packaging Marketing	Create the Script for the Team Game Project Lens Presentations
14	Marketing Presentations	Peer Review of Game Marketing Video
15	Quality Assessment	Peer Assessment of Gold Release Candidates

Each team's first task was to create a game design document and a business plan for their game. To assist them in this task, two local game company owners were recruited to act in the role of business consultants who shared their experiences with creating a company and bringing their first games to market. The second team deliverable was a game alpha prototype, which included one complete logic path and a draft user manual. This delivery signaled the end of the first sprint in the scrum framework. These games were evaluated for quality of game play. The company looked at the productivity of each team. The team leads were asked to make an oral presentation to confirm that they had sufficient resources to complete their game products on time (the end of the semester was designated as the end of the fiscal year). All developers discussed the future of the game products and decided (without the instructor's influence) to cancel one of the projects. The developers from the canceled project were reassigned to existing development teams.

The third team deliverable was a beta prototype, which needed to accommodate a requirement change. The change resulted in the addition of a significant game artificial intelligence (AI) element to their evolving design. This deliverable also included the creation of the final game design document and test plan. The final team deliverable was the gold release prototype and a marketing presentation that included a video piece to promote their game product. Company developers scored each game (other than their own) using a rubric provided by the instructor. The average of these scores was used as the grade for the prototype.

The students participated in several role-play scenarios through the semester, in addition to greenlighting the games. One element of this class that was hard to fit into the role-play framework was the assignment where each developer uses their own game to illustrate game design features from Schell's book of game design lenses [67]. In this assignment, each student selects a group of three related lenses and creates a 20-min presentation discussing how these lenses illustrate qualities from their game or not. This is sold as continuing education or inspiration for undertaking perfective maintenance activities to the company developers.

9.4 Results and Discussion

Each of the course assignments was evaluated using Canvas rubrics designed by the instructor for each type of submission. Currently, these rubrics contain 2–10 criteria, each scored from 1 to 5. Table 9.3 shows the rubric used to evaluate the active learning assignments that called for students to conduct experiments or create design artifacts. Specialized rubrics were created for the team project assignments.

No statistical comparisons of performance on the assignment write-ups were made between students in the in-person section and the asynchronous online sections of CIS 487 during Fall 2021 or for the in-person and online sections of CIS 488 in Winter 2022. However, informal comparisons of student data from the two modes of CIS 487 delivered by the instructor in Fall 2021 suggest that students attending the in-person class meetings produced work, which seemed to receive

Table 9.3 CIS 487/488 activity question rubric

Topic	Rating and feedback (0=missing, 4=satisfactory, 5=exceeds specification)
Quality of answers	
Completeness of write-up	

higher scores using similar grading rubrics. Similar observations were made for CIS 488 students in Winter 2022.

The authors created four research questions to compare the levels of engagement by students taking CIS 487 and 488 under flipped classroom in-person (FC) active learning as compared to the engagement of students taking previous offerings of CIS 487 and 488 with fewer active learning opportunities.

RQ1: Is the flipped classroom student performance worse than student performance in other course delivery modes?

To answer this question, the authors looked at data analytics (number of late and missing assignments) collected by the Canvas management system for three iterations of each course sequence shown in Tables 9.4 and 9.5. To briefly describe the difference between the semesters, Fall 2016 and Winter 2017 represented a Lecture-Heavy (LH) version of the courses before active learning activities were fully introduced in the curriculum, while Fall 2017 and Winter 2018 represent an intermediate (IM) step between the previous LH version of the courses and the flipped classroom (FC) version that fully embraced active learning techniques in Fall 2021 and Winter 2022, both of which involved heavy social distancing.

In Table 9.4 for CIS 487, it initially appears that there was a decrease in student grades and performance as the class transitioned into using active learning. However, this is due to an outlier from an underperforming student. There is no statistical difference at the 95% confidence level from the student t-test for the overall grade between F2016-LH and F2017-IM for overall grade or average number of late assignments per student. Yet, there was a statistical difference in populations for the average number of missing assignments per student. We attribute this to the increased workload caused by having students report on their activities in the

Table 9.4 CIS 487 Canvas course analytics for the Fall 2016, Fall 2017, and Fall 2021 semesters

	F2016-LH N = 24	F2017-IM N = 22	F2021-FC N = 23
Average overall course grade	91.2%	84.6%	91.8%
Average number of late assignments per student	0.4	0.7	0.2
Average number of missing assignments per student	0.1	1.8	0.5

Table 9.5 CIS 488 Canvas course analytics for winter 2017, winter 2018, and winter 2022 semesters

	W2017-LH N = 17	W2018-IM N = 18	W2022-FC N = 23
Average overall course grade	87.7%	95.3%	95.1%
Average number of late assignments per student	0.6	0.2	0.6
Average number of missing assignments per student	2.9	3.1	2.2

class. There was no statistical difference at 95% confidence between F2016-LH and F2021-FC.

Although at first glance in Table 9.5 it would appear the introduction of active learning techniques in W2018-IM and W2022-FC had a positive effect on student performance, there was no statistical difference between the W2017-LH version of the course and either W2018-IM and W2022-FC at 95% confidence for overall course grade, average number of late assignments, and average number of missing assignments. Therefore, we conclude from this that flipped class student performance is at least not hindered in active learning course modalities, but it is important to keep in mind the added burden of daily assignment write-ups as students transition to new course delivery methods.

While it is not reflected in data shown in Tables 9.4 or 9.5, students seem to be exhibiting better communication skills in the flipped classroom delivery because of the increased writing and oral presentation requirements as compared to the lecture versions of the courses. While not measured explicitly in our work, most students seem to write better and more meaningful peer reviews as they progress through the courses. Team participation is better in the active learning classes than in the lecture heavy versions of the classes.

9.4.1 Course Surveys

We surveyed the students during the final weeks of each semester, to gather the students' own perceptions of their levels of engagement with the class, active learning, and gamification. The CIS 487 survey emphasized active learning and engagement. The CIS 488 survey emphasized gamification and engagement.

RQ2: Do flipped classroom students have a different perception of their level of engagement as reported on the CIS 487 final survey than students in other course delivery modes?

Students rated each statement on their perceptions of active learning and their engagement in the survey from 1 (strongly disagree) to 5 (strongly agree). The distribution of responses to each question for CIS 487 is seen in Table 9.6. We performed a statistical analysis of the responses using the Mann-Whitney U Test. We found no statistical difference between the responses for the F2016-LH and F2017-IM groups at the 95% confidence level. This indicated that in F2017-IM, we had begun to implement some activities that students in both groups did not seem to feel differently about their active learning and engagement in the course. However, students in the F2021-FC course were significantly different at the 95% confidence level than the F2016-LH group for survey questions 2–5. Students agreed more that the course activities were useful (65% vs. 26% strongly agree) and allowed them to apply what they learned (70% vs. 44% strongly agree), and when asked if they did not understand the connection between the class activities and other aspects of the course reported, they strongly disagreed 65% to 41%.

Table 9.6 End-of-course student perception survey results focusing on agreement with the statements evaluating their engagement for three CIS 487 courses

Survey statement	Strongly disagree	Disagree	Neutral	Agree	Strongly agree	Course
1. There were opportunities for me to actively engage in learning	1 (4%)	0	1 (4%)	3 (11%)	22 (81%)	F2016-LH
	0	0	1 (5%)	7 (33%)	13 (62%)	F2017-IM
	0	0	1 (5%)	4 (20%)	15 (75%)	F2021-FC
2. Course activities were a useful way to learn	3 (11%)	0	3 (11%)	14 (52%)	7 (26%)	F2016-LH
	0	2 (10%)	3 (14%)	11 (52%)	5 (24%)	F2017-IM
	0	0	1 (5%)	6 (30%)	13 (65%)	F2021-FC
3. Course activities let me apply what I learned	2 (7%)	2 (7%)	2 (7%)	9 (33%)	12 (44%)	F2016-LH
	1 (5%)	2 (10%)	3 (14%)	10 (48%)	5 (24%)	F2017-IM
	0	0	0	6 (30%)	14 (70%)	F2021-FC
4. Course is an example of active learning	2 (7%)	0	2 (7%)	12 (44%)	11 (41%)	F2016-LH
	0	1 (5%)	3 (14%)	8 (38%)	9 (43%)	F2017-IM
	0	0	1 (5%)	2 (10%)	17 (85%)	F2021-FC
5. I didn't understand connection between class activities and other aspects of course	11 (41%)	8 (30%)	6 (22%)	0	2 (7%)	F2016-LH
	10 (48%)	8 (38%)	2 (10%)	1 (5%)	0	F2017-IM
	13 (65%)	6 (30%)	1 (5%)	0	0	F2021-FC
6. Working in groups was an effective way for me to learn	2 (7%)	2 (7%)	3 (11%)	9 (33%)	11 (41%)	F2016-LH
	0	3 (14%)	3 (14%)	8 (38%)	7 (33%)	F2017-IM
7. I prefer to learn primarily through lecture	7 (26%)	6 (22%)	6 (22%)	6 (22%)	2 (7%)	F2016-LH
	6 (29%)	9 (43%)	2 (10%)	3 (14%)	1 (5%)	F2017-IM
8. I had more opportunities to actively engage in learning in this class compared to other classes I've taken	2 (7%)	0	0	9 (33%)	16 (59%)	F2016-LH
	0	1 (5%)	1 (5%)	11 (52%)	8 (38%)	F2017-IM

In addition, we also looked at comparing F2017-IM to F2021-FC students' perceptions of how active learning was different between the intermediate implementation of the course and a fully flipped classroom. Again, we used the Mann-Whitney U Test finding that the student populations for survey questions 2–4 were significantly different at the 95% confidence level. Students more strongly felt that course activities were a useful way to learn (65% vs. 24% strongly agree) and that the course let them apply what they learned (70% vs. 24%).

We additionally asked students to rate their engagement for six survey questions on a scale from "very little of their time" to "most of the time" for specific behaviors. Unfortunately, as we redesigned the course, we modified the survey for F2021-FC to be less time intensive and do not have student response data for questions 9–14 as seen in Table 9.7. Therefore, we only compare F2016-LH to F2017-IM.

We used a Mann-Whitney U Test to statistically compare populations. At the 95% confidence interval only, question 13 had a statistical difference. Students stated stronger disagreement in F2017-IM for being expected to memorize facts

Table 9.7 End-of-course student survey focusing on rating active learning elements of their experience in three courses of CIS 487

Survey statement	Very little of the time	Less than half the time	At least half the time	Most of the time	Course
9. I was actively engaged in my learning	0 0	1 (4%) 0	9 (33%) 7 (33%)	17 (63%) 14 (67%)	F2016-LH F2017-IM
10. The professor created opportunities for me to actively engage in my learning	0 0	3 (11%) 0	3 (11%) 8 (38%)	21 (78%) 13 (62%)	F2016-LH F2017-IM
11. I applied the course material to real-world situations	3 (11%) 2 (10%)	2 (7%) 4 (19%)	8 (30%) 8 (38%)	14 (52%) 7 (33%)	F2016-LH F2017-IM
12. My small group worked effectively and collaboratively	1 (4%) 0	2 (7%) 3 (14%)	6 (22%) 8 (38%)	18 (67%) 10 (48%)	F2016-LH F2017-IM
13. I was expected to memorize facts and information	4 (15%) 10 (48%)	16 (59%) 9 (43%)	1 (4%) 1 (5%)	6 (22%) 1 (5%)	F2016-LH F2017-IM
14. I spent time working on activities that were too simplistic or irrelevant	15 (56%) 10 (48%)	8 (30%) 10 (48%)	2 (7%) 1 (5%)	2 (7%) 0	F2016-LH F2017-IM

and information than students in F2016-LH with 48% to 15% strong disagreement. We attribute this to the insertion of small activities in some lectures, which moved students from listening to lectures to investigating and discovering how the information they have learned works in practice.

Overall, student agreement seemed stronger when compared to the intermediate course than the lecture heavy. It may be possible that an incomplete or partial implementation of active learning techniques in a class prevents or diminishes students' perceptions of active learning. We suggest that those wishing to implement similar changes in their pedagogical approach may be better served by fully embracing an active learning course redesign rather than a slow or partial implementation spread out over several semesters.

Without prompting students in the active learning courses showed strong preference for working on the activities and projects, as opposed to taking exams. They felt that the activities and project-based learning approach not only prepared them better for their senior design class but also prepared them better for their careers.

Overwhelmingly, the projects are the biggest strength cited by students in the course surveys. Their comments reinforce the positive effect of projects on practical learning as well as the development of collaborative, problem-solving skills. Several students also indicated that replacing exams with projects provided a more meaningful learning experience and knowledge that would be otherwise difficult to assess with a traditional assessment approach.

RQ3: Does gamification affect the choices of flipped classroom students differently than students in other course delivery modes as reported on the CIS 488 final survey?

Gamification was examined in the CIS 488 final survey (see Table 9.8). We only administered this survey to students taking this course in Winter 2017 and Winter 2022. Students again submitted their responses as a 1 (strongly disagree) to 5 (strongly agree). We also performed a statistical analysis with the Mann-Whitney U test on the W2017-LH and W2022-FC student populations. At 95% confidence level, there was no statistical difference between the two groups' responses. Both student groups seem evenly split on statement 3, "I did what I had to, but didn't feel I had a choice," while also agreeing 70% vs. 80% with statement 2, "I felt like I

Table 9.8 CIS 488 end of course survey on student perceptions on gamification

Survey statement	Strongly disagree	Disagree	Neutral	Agree	Strongly agree	Course
1. I put more effort into assignments for than I normally do for the courses I take	0 0	1 (10%) 0	3 (30%) 2 (11%)	3 (30%) 9 (50%)	3 (30%) 7 (39%)	W2017-LH W2022-FC
2. I felt like I had more control and choice over the assignments I completed than I normally do	1 (10%) 0	1 (10%) 0	1 (10%) 2 (11%)	2 (20%) 8 (44%)	5 (50%) 8 (44%)	W2017-LH W2022-FC
3. In this course, I did what I had to, but I didn't feel like it was really my choice	1 (10%) 2 (11%)	3 (30%) 7 (39%)	2 (20%) 3 (17%)	2 (20%) 4 (22%)	2 (20%) 2 (11%)	W2017-LH W2022-FC
4. In this course, I picked assignments based on what interested me	1 (10%) 1 (6%)	0 1 (6%)	0 4 (22%)	5 (50%) 5 (28%)	4 (40%) 7 (39%)	W2017-LH W2022-FC
5. In this course, I feel I had control over how I demonstrated my understanding of the course material	1 (10%) 0	1 (10%) 1 (6%)	0 1 (6%)	4 (40%) 7 (39%)	4 (40%) 9 (50%)	W2017-LH W2022-FC
6. When picking the assignments you submitted for this course, how important to you when deciding was how many points I could earn by doing the assignment?	0 2 (11%)	1 (10%) 0	3 (30%) 11(61%)	2 (20%) 2 (11%)	4 (40%) 3 (17%)	W2017-LH W2022-FC
7. When picking the assignments you submitted for this course, how important to you when deciding was how much the assignment allowed me to collaborate with my classmates?	1 (10%) 0	3 (30%) 4 (22%)	2 (20%) 7 (39%)	2 (20%) 4 (22%)	2 (20%) 3 (17%)	W2017-LH W2022-FC

had more control and choice than I normally do." We conclude that in our limited study, it did not appear that students in the W2022-FC class were influenced by gamification than students in the lecture delivery mode.

However, student comments clearly indicated they liked being given the ability to make choices that impacted their learning. It allowed them to tailor their experiences directly to their interests and skills. We believe this contributed to the high quality of the games produced by the students during the semester.

We suggest this was due to an increase in motivation caused by being permitted to pursue their individual interests. As one student wrote reflective of multiple other comments, "I'm more driven to do a good job, since I choose to do it." Meanwhile, another student commented, "This inspires creativity and forces students to solve real world problems, along with delivering a full product." Interestingly, the point valuation seemed less important to the students when picking an assignment even if it meant fewer points were awarded.

9.4.2 Course Evaluations

Students on our campus are requested to complete a standard set of course evaluations at the end of the semester. The evaluation form is completed online and anonymously prior to receiving their final course grades. We wanted to compare the course evaluations of socially distanced students in other active learning conditions.

RQ4: Do flipped classroom students have different course experiences than students in other course delivery modes?

Questions are rated from 1 (strongly disagree) to 5 (strongly agree). Our college redesigned the course evaluations during the period between W2018 and F2021 to solicit different information, so we have included the most pertinent survey questions for CIS 487 in Table 9.9 and for CIS 488 in Table 9.10.

The student comments on the course evaluations indicated that they enjoyed the design activities and felt these activities helped them when creating their project deliverables. They also felt that sharing ideas and insights with other students during class discussions helped them learn. They enjoyed being able to apply the material covered in the lectures and tutorials to solve actual problems.

Students appreciated the class activities for a variety of reasons. They felt these activities were more engaging than just listening to a lecture accompanied by slides. The students liked the redundancy that was built in the activities that often had them look at different facets of similar design concerns. Some students wrote that they felt the group work and writing activity summaries helped them become more at ease when speaking in class.

Students felt that the strengths of this course were the dynamic learning activities, the lack of exams, and game project development. They also felt that completing the class activities collaboratively provides better opportunities for students to master the material.

Table 9.9 CIS 487 end-of-term collegiate course evaluations

1 = Strongly disagree 5 = Strongly agree	F2016-LH N = 21	F2017-IM N = 22	F2021-FC N = 24
Course met my expectations	4.33	4.55	4.56
Course objectives were clear	4.24	4.36	4.67
Typical workload compared to other courses	~	~	4.21
Course advanced my understanding of subject	~	~	4.75
Lab activities increased my understanding of lecture topics	~	~	2.92
I knew what was expected of me	~	~	4.52
Overall course rating	4.52	4.73	4.63

Table 9.10 CIS 488 end-of-term collegiate course evaluations

1 = Strongly disagree 5 = Strongly agree	W2017-LH N = 13	W2018-IM N = 13	W2022-FC N = 11
Course met my expectations	4.85	4.38	4.50
Course objectives were clear	4.85	4.31	4.64
Typical workload compared to other courses	~	~	4.00
Course advanced my understanding of subject	~	~	4.50
Lab activities increased my understanding of lecture topics	~	~	2.09
I knew what was expected of me	~	~	4.64
Overall course rating	4.85	4.46	4.45

9.4.3 Lessons Learned

We believe that some of our findings can be applied to other engineering project courses. Looking at the course analytics, course evaluations, and engagement survey data, we found two common themes. The first is that there are few statistical differences in the academic performance between students in the lecture heavy versions of the courses and flipped classroom versions. We interpret this to mean that these two courses successfully transitioned from lecture heavy to active learning. Harder to measure is the growth in the students' soft skills (written and oral communication, collaboration, and project management). The second is that students feel more engaged in the active learning versions of these courses and like the flexibility gamification brings them.

Students enjoyed the role-play (CIS 488) and felt is added to the realism of the development process. It is interesting to note that seven student teams from CIS

488 have gone on to form LLCs to continue their game development activities professionally. This did not happen prior to the introduction of the failing game company role-play in CIS 488.

Students enjoyed the class activities but sometimes needed more guidance (scaffolding) and more time to complete some of the activities. Students love working in small groups, but they do not like talking to the whole class without a script. In courses involving both in-person and online students, both types of students were included on the same project teams as this tends to increase online student engagement.

9.4.4 Threats to Validity

We recognize that one of the limitations of this study was that we did not have a control group. We also acknowledge that the instructor teaching all CIS 487 and 488 course offerings may also account for the lack of significant differences on some of the evaluation measures.

The asynchronous course delivery, pre- and post-Covid shutdown, was significantly different than that occurring using zoom during 2020 or 2021. Prior to 2019 and university implemented a policy which required the pairing of an asynchronous, distance learning section with a face-to-face section of the same course. The live class sessions were captured, verbatim, for later viewing by the asynchronous students. This course feature was implemented in most CIS courses prior to Covid. This provided an advantage to the 2019 and 2021 asynchronous online sections in that they could witness the live lecture and some class activities as a virtual classroom observer. During 2021–2022, the asynchronous online students were only able to view the class activities plus any recorded videos posted for pre-class viewing by the flipped class in-person students.

One area of uncertainty when measuring the student responses is the unknown amount of interaction between students in the synchronous and asynchronous sections of CIS 487 and 488. Students in the CIS department know each other from other classes that they have taken together. Even though a student registered in the asynchronous online section was not allowed to attend any in-person class meetings, it is quite possible that a friend from an in-person course section may have shared their course experiences with them giving them additional insight into group activities completed in the classroom. In other words, the asynchronous student may not be totally isolated from knowledge learned in the group activities. We did not attempt direct comparisons between in-person and online students in this chapter. Student engagement can only be measured indirectly in online courses using surveys and course analytics. In 2016–2017, direct observation of student behavior was used to provide insight into their levels of engagement among in-person class instruction. We did not include direct observation of students in the socially distanced in-person sections of either CIS 487 or CIS 488. Trying to measure student engagement using chat comments or interaction with shared Google documents is a viable alternative but also lacks the immediate visual feedback an instructor experiences with a real-time view of a student's face.

There were no surveys taken between the Winter 2017 and Winter 2022 offerings of CIS 488. These surveys provide the most direct and candid feedback on active learning from the student's perspective. Although 2017 CIS 488 survey data provides some good baseline data, it would have been more beneficial to have data from Winter 2018.

The 2019–2020 and 2020–2021 academic years presented extraordinary challenges for students. All students, not just those from this university, were asked to learn under circumstances never-before experienced. While it would be expected that many students were excited to return to face-to-face instruction, it may also be expected that many felt anxious or even distracted with the fresh look of face-to-face instruction. It is difficult to assess what effects, both positive and negative, this might have had on the return to an active learning classroom in Fall 2021.

9.5 Conclusions and Future Direction

During the past 3 years, most institutions across the world were required to switch to online formats. This switch to using videoconferencing often required major adjustments to course design and left many students simply watching online lecture videos and taking exams. We demonstrated in previously reported studies that it is possible to move an in-person active learning software engineering course online [68, 69]. We also believe that engineering project courses can be run using social distancing Covid protocols without observing significant reductions in student levels of engagement as compared to other course formats. We take this as evidence that it is possible to design a socially distanced active learning course that can be more engaging than its online counterpart. We credit the active learning components of the class and the levels of student interaction that accompany them for making this possible. We encourage other instructors to adopt active learning practices and modify them as needed to satisfy Covid protocol requirements in their course deliveries to achieve higher levels of student satisfaction and engagement.

We were encouraged by the enthusiasm that students exhibited while working with the active learning modules during the in-person class meetings and look forward to continuing to develop this course content. It may be important to develop ways in which asynchronous students are encouraged to be a part of some sort of face-to-face experience, even if it is not during formal online class meetings. Informal study or discussion groups that would meet online, with flexible meeting times, might be a way to increase engagement with activities. The demand for online game design offerings is strong among the students on our campus. Experiences from the Fall 2021 course delivery of CIS 487 and Winter 2022 course delivery of CIS 488 will be used to revise the next offering of these courses and their corresponding active learning materials. The challenge will be to seek ways to ensure that online students feel engaged with the class materials.

Acknowledgments This project was partially supported by a grant from the University of Michigan-Dearborn Advancement of Teaching and Learning HUB Creative Teaching Fund.

References

1. Becker, K.: Teaching with games: the minesweeper and asteroids experience. J. Comput. Sci. Coll. **17**(2), 23–33 (2001)
2. Jones, R.: Design and implementation of a computer games: a capstone course for undergraduate computer science education. In: Proceedings of 31st SIGCSE Technical Symposium (Austin, TX, March 2000), pp. 260–264. ACM Press, New York, NY (2000)
3. Pleva, G.: Game programming and the myth of child's play. J. Comput. Sci. Coll. **20**(2), 125–136 (2004)
4. Claypool, K., Claypool, M.: Software engineering design: teaching software engineering through game design. In: Proceedings of 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, Caparica, Portugal, June 2005, pp. 123–127. ACM Press, New York, NY (2005)
5. Rollings, A., Morris, D.: Game Architecture and Design. New Riders, Indianapolis, IN (2004)
6. Maxim, B.R.: Software Requirements Analysis and Design. NIIT, Atlanta, GA (2004)
7. Samavedham, L., Ragupathi, K.: Facilitating 21st century skills in engineering students. J. Eng. Educ. **XXVI**(1), 38–49 (2012)
8. Maxim, B.R., Acharya, S., Brunvand, S., Kessentini, M.: WIP: introducing active learning in a software engineering course. In: Proceedings of the 2017 Annual Meeting of the American Society for Engineering Education, Columbus, OH, June 2017, pp. 1–12
9. Promoting Active Learning. https://utah.instructure.com/courses/148446/pages/active-learning. Accessed 25 Feb 2016
10. Prince, M.: Does active learning work? A review of the research. J. Eng. Educ. **93**, 223–231 (2004)
11. Luster-Teasley, S., Hargrove-Leak, S.C., Waters, C.: NSF TUES: Transforming undergraduate environmental engineering laboratories for sustainable engineering using the case studies in the sciences instructional method. In: Proceedings of the 2014 Annual Meeting of the American Society for Engineering Education, Indianapolis, IN, June 2014
12. Jungic, V., Kaur, H., Mulholland, J., Xin, C.: On flipping the classroom in large first-year calculus courses. Int. J. Math. Educ. Sci. Technol. **46**(4), 1–8 (2015)
13. Raju, P.K., Sanker, C.C.: Teaching real-world issues through case studies. J. Eng. Educ. **88**(4), 501–508 (1999)
14. Nickels, K.M.: Do's and don'ts of introducing active learning techniques. In: Proceedings of the 2000 Annual Meeting of the American Society for Engineering Education, St. Louis, Missouri, June 2000
15. Lavelle, J.P., Stimpson, M.T., Brill, E.D.: Flipped out engineering economy: converting a traditional class to an inverted model. In: Krishnamurthy, A., Chan, W.K.V. (eds.) Proceedings of the 2013 Industrial Systems Engineering Research Conference, pp. 397–407 (2013)
16. Wood, K., Jensen, D., Dutson, A., Green, M.: Active learning approaches in engineering design courses. In: Proceedings of the 2003 Annual Meeting of the American Society for Engineering Education, Nashville, Tennessee, June 2003
17. Maxim, B.R., Decker, A., Yackley, J.J.: Student engagement in active learning software engineering courses. In: Proceedings of 49th IEEE Annual Frontiers in Education Conference, Cincinnati, OH, October 2019, pp. F3G1–F3G5
18. Yelamarthi, K., Member, S., Drake, E.: A flipped first-year digital circuits course for engineering and technology students. IEEE Trans. Educ. **58**(3), 179–186
19. Meier, R.D.: Active learning in large lectures. In: Proceedings of the 1999 Annual Meeting of the American Society for Engineering Education, Charlotte, North Carolina, June 1999

20. Krause, R., Hayton, A.C., Wonoprabowo, J., Loo, L.: Is engagement alone sufficient to ensure "active learning?". Loma Linda Univ. Stud. J. **2**(1) (2017)
21. Ardis, M., Chenoweth, S., Young, F.: The 'Soft' topics in software engineering Education. In: Proceedings of 38th Annual Frontiers in Education Conference (Vol. 1, Oct 2008), pp. F3H1–F3H6. IEEE Press, Saratoga Springs, NY (2008)
22. Day, J.A., Foley, J.D.: Evaluating a web lecture intervention in a human-computer interaction course. IEEE Trans. Educ. **49**(4), 420–431 (2006)
23. Bishop, J.L., Verleger, M.A.: The flipped classroom: a survey of the research. In: Proceedings of the 2017 Annual Meeting of the American Society for Engineering Education, Atlanta, GA. (2013)
24. Wu, P., Manohar, P., Acharya, S.: The design and evaluation of class exercises as active learning tools in software verification and validation. Inf. Syst. Educ. J. (2016)
25. Cheng, L., Ritzhaupt, A.D., Antonenko, P.: Effects of the flipped classroom instructional strategy on students' learning outcomes: a meta-analysis. Educ. Technol. Res. Dev. **67**(4), 793–824 (2018)
26. Morrison, G.R., Ross, S.M., Kemp, J.E., Kalman, H.: Designing Effective Instruction. Wiley (2010)
27. Savery, J., Duffy, T.: Problem-based learning: an instructional model and its constructivist framework. Educ. Technol. **35**(5), 31–38 (1995)
28. Silva, A., Bispo, A., Rodriguez, D., Vasquez, F.: Problem-based learning: a proposal for structuring PBL and its implications for learning among students in an undergraduate management degree program. Revista de Gestão. **25**(2), 160–177 (2018)
29. Warnock, J.N., Mohammadi-Aragh, M.J.: Case study: Use of problem-based learning to develop students' technical and professional skills. Eur. J. Eng. Educ. **41**(2), 142–153 (2016)
30. Dunlap, J.: Problem-based learning and self-efficacy: how a capstone course prepares students for a profession. Educ. Technol. Res. Dev. **53**(1), 65–83 (2005)
31. Urbanic, R.: Developing design and management skills for senior industrial engineering students. J. Learn. Des. **4**(3), 35–49 (2011)
32. Gavin, K.: Case study of a project-based learning course in civil engineering design. Eur. J. Eng. Educ. **36**(6), 547–558 (2011)
33. Souza, M., et al.: Students perception on the use of project-based learning in software engineering education. In: SBES 2019: Proceedings of the XXXIII Brazilian Symposium on Software Engineering, pp. 537–546 (2019)
34. Kothiyal, R., et al.: Effect of think-pair-share in a large CS1 class: 83% sustained engagement. In: Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research (ICER '13), pp. 137–144. ACM, New York, NY (2013)
35. Nagappan, M., Williams, L., Ferzli, M., Wiebe, E., Yang, K., Miller, C., Balik, S.: Improving the CS1 experience with pair programming. In: Proceedings of the 34th SIGCSE technical symposium on Computer science education (SIGCSE '03), pp. 359–362. ACM, New York, NY (2003)
36. Porter, L., Bouvier, D., Cutts, Q., Grissom, S., Lee, C., McCartney, R., Zingaro, D., Simon, B.: A multi-institutional study of peer instruction in introductory computing. In: Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16), pp. 358–363. ACM, New York, NY (2016)
37. Greer, T., Hao, Q., Jing, M., Barnes, B.: On the effects of active learning environments in computing education. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), February 27–March 2, 2019, Minneapolis, MN, 6 pages. ACM, New York, NY
38. Hoffman, B., Morelli, R., Rosato, J.: Student engagement is key to broadening participation in CS. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), February 27–March 2, 2019, Minneapolis, MN, 7 pages. ACM, New York, NY
39. Ham, Y., Myers, B.: Supporting guided onquiry with cooperative learning in computer organization. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), pp. 273–279. ACM, New York, NY (2019)

40. Stone, J.A., Madigan, E.: Experiences with community-based projects for computing majors. J. Comput. Sci. Coll. **26**(6), 64–70 (2011)
41. Kharitonova, Y., Luo, Y., Park, J.: Redesigning a software development course as a preparation for a capstone. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), February 27–March 2, 2019, Minneapolis, MN, 7 pages. ACM, New York, NY
42. Decker, A., Simkins, D.: Leveraging role play to explore the software and game development process. In: Proceedings of 46th IEEE Annual Frontiers in Education Conference, Erie, PA, October 2016, pp. S3F6–S3F10
43. Simkins, D.: The arts of larp: design, literacy, learning, and community in live-action role play. McFarland, Jefferson, NC (2015)
44. Moroz-Lapin, K.: Role play in HCI studies. In: Proceedings of the 2009 international conference on HCI Educators: playing with our Education (HCIEd'09), pp. 12–12. British Computer Society, Swinton (2009)
45. Seland, G.: Empowering end users in design of mobile technology using role play as a method: reflections on the role-play conduction. In: Kurosu, M. (ed.) Proceedings of the 1st International Conference on Human Centered Design: Held as Part of HCI International 2009 (HCD 09), pp. 912–921. Springer, Berlin (2009)
46. Zowghi, D., Paryan, S.: Teaching requirements engineering through role playing: lessons learnt. In: Proceedings of the 11th IEEE International Conference on Requirements Engineering (RE '03), pp. 233–241. IEEE Computer Society, Washington, DC (2003)
47. Börstler, J.: Improving CRC-card role-play with role-play diagrams. In: Companion to the 20th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA '05), pp. 356–364. ACM, New York, NY (2005)
48. Vold, T., Yayilgan, S.Y.: Playful participation for learning in higher education — The introduction of participatory role play simulation in a course at Hedmark University College. In: Proceedings of 2013 International Conference on Information Technology Based Higher Education and Training (ITHET), Antalya, 2013, pp. 1–4
49. Rudra, A., Jaeger, B., Aitken, A., Chang, V., Helgheim, B.: Virtual team role play using second life for teaching business process concepts. In: Proceedings of 44th Hawaii International Conference on System Sciences (HICSS), Kauai, HI, 2011, pp. 1–8
50. Maxim, B.R., Kaur, R., Apzynski, C., Edwards, D., Evans, E.: An agile software engineering process improvement game. In: Proceedings of 46th IEEE Annual Frontiers in Education Conference, Erie, PA, October 2016, pp. S3F1–S3F5
51. Nakamura, T., Maruyama, H., Takashima, A., Sambe, Y.: Role-play exercises for project management education that incorporate a software agent. In: Proceedings 2012 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE), Hong Kong, 2012, pp. W2A-8–W2A-14
52. Navarro, E., Hoek, A.: SimSE: an interactive simulation game for software engineering education. In: Proceeding of the Seventh IASTED International Conference on Computers and Advanced Technology in Education, pp. 12–17 (2004)
53. Maxim, B.R., Decker, A., Brunvand, S.: Use of role-play and gamification in a software project course. In: Proceedings of 47th IEEE Annual Frontiers in Education Conference, Indianapolis, IN, October 2017, pp. T3D1–T3D5
54. Yackley, J.J., Maxim, B.R., Brunvand, S., Decker, A.: Active learning and gamification in game design courses. In: Proceedings of Meaningful Play 2018 Conference, East Lansing, MI, October 2018, pp. 165–178
55. Domínguez, A., Saenz-de-Navarrete, J., de-Marcos, L., Fernández-Sanz, L., Pagés, C.A., Martínez-Herráiz, J.J.: Gamifying learning experiences: practical implications and outcomes. Comput. Educ. 380–392
56. Simões, J., Redondo, R.D., Vilas, A.F.: A social gamification framework for a K-6 learning platform. Comput. Hum. Behav. **29**, 345–353 (2012)
57. Gee, J.P.: What Video Games Have to Teach Us About Learning and Literacy, 2nd edn. St. Martin's Press (2014)

58. Gee, J.P.: What video games have to teach us about learning and literacy. Comput. Entertain. **1**(1), 1–4 (2003)
59. Vygotsky, L.S.: Mind and Society: The Development of Higher Mental Processes. Harvard University Press (1978)
60. Granic, I., Lobel, A., Engels, R.: The benefits of playing video games. Am. Psychol. **69**(1), 66–78 (2014)
61. Ott, M., Tavella, M.: A contribution to the understanding of what makes young students genuinely engaged in computer-based learning tasks. Procedia Soc. Behav. Sci. **1**(1), 184–188 (2009)
62. Lee, J.J., Hammer, J.: Gamification in education: what, how, why bother? Definitions and uses. Exchange Organ. Behav. Teach. J. **15**(2), 1–5 (2011)
63. Yang, Y.T.C.: Building virtual cities, inspiring intelligent citizens: digital games for developing students' problem solving and learning motivation. Comp. Educ. **59**(2), 365–377 (2012)
64. Toth, D., Kayler, M.: Integrating role-playing games into computer science courses as a pedagogical tool. In: Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15), pp. 386–391. ACM, New York, NY (2015)
65. Bogost, I.: Gamification is bullshit. http://bogost.com/writing/blog/gamification_is_bullshit/. Accessed 30 Aug 2018
66. O'Donnell, C.: Getting played: gamification, bullshit, and the rise of algorithmic surveillance. Surveill. Soc. **12**(3), 349–359 (2014)
67. Schell, J.: The Art of Game Design: A Book of Lenses. CRC Press (2015)
68. Maxim, B.R., Limbaugh, T., Yackley, J.J.: Student engagement in an online software engineering course. In: Proceedings of 51st IEEE Annual Frontiers in Education Conference, Lincoln, NE, October 2021, pp. T3D1–T3D9
69. Maxim, B.R., Limbaugh, T.: WIP: Engaging software engineering students in synchronous asynchronous on-line course. In: Proceedings of the 2021 Annual Meeting of the American Society for Engineering Education, Long Beach, CA, July 2021, pp. 1–17