

## **Board 63: Work in Progress: Adapting Scrum project management to ECE courses**

### **Prof. Branimir Pejcinovic, Portland State University**

Branimir Pejcinovic received his Ph.D. degree from University of Massachusetts, Amherst. He is a Professor and former Associate Chair for Undergraduate Education at Portland State University, Electrical and Computer Engineering department. In this role, he has led department-wide changes in the curriculum with emphasis on the project- and lab-based instruction and learning. His research interests are in the areas of engineering education, semiconductor device characterization, design and simulation, signal integrity and THz sensors. He is a member of IEEE and ASEE.

### **Mr. Phillip Wong, Portland State University**

Phillip Wong received an M.S. degree in electrical engineering from Carnegie Mellon University in 1990. Since then, he has been with Portland State University, Oregon, USA, where he is currently the ECE Lab Coordinator and an instructor.

### **Dr. Robert Bass, Portland State University**

Robert Bass, Ph.D. is an associate professor in the Department of Electrical & Computer Engineering at Portland State University. His research focuses on electrical power systems, particularly distributed utility assets and the overlaying control and communications architectures that link them together. Dr. Bass specializes in teaching undergraduate and graduate courses on electric power, electromechanical energy conversion, distributed energy resources, control theory and power systems analysis.

# Work in Progress: Adapting Scrum project management to ECE courses

## Motivation

There are many reasons that undergraduate engineering programs rely on hands-on type of student activities, such as labs and projects. Collaborative work not only helps student learning, but it also mimics actual engineering work, which is often done in teams. Various projects, at both the course and program levels, are also meant to be authentic and are known to improve student motivation to study engineering. ABET accreditation explicitly requires some level of teamwork through Student Outcomes, in particular criterion 5, “*an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives*” [page 42, 1]. Clearly, both ABET and engineering faculty expect students to engage in substantive projects that include teamwork and project management.

Typically, the senior-level capstone design project is when students engage in more authentic projects. However, there are engineering programs that have also developed so-called cornerstone courses at lower divisions (e.g. [2], [3]). Projects may differ in complexity and sophistication, but common to all of them, students reduce a given problem into subtasks, divide responsibilities among team members, and develop a feasible schedule to solve those tasks. In other words, a key component is some form of project management (PM).

Frequently though, students use an ad hoc project management approach, as they lack experience in formal techniques. This lack of PM training is sometimes addressed in the senior year, either as part of a capstone project or as a separate course. However, best educational practices suggest that we should start teaching such skills early and scaffold student development of mastery, instead of waiting until senior year [4].

To that end, we have developed a gradual approach to teaching PM during the freshman, sophomore, and senior years. The choice of a particular PM technique is important, as it should be adaptable to student development and simple enough not to overwhelm the learners. Scrum provides a framework that facilitates teamwork and project management through an adaptable, incremental process. We have tailored our variant of Scrum for students working on engineering projects in a higher-education environment. In this manuscript, we describe some general features of Scrum, and how it fits with project-based courses and curriculum. Following this, we describe our implementation. We then provide a summary of our work, along with suggestions for future development.

## Introduction to Scrum

Scrum is a popular form of Agile project management. Its applications now include diverse areas such as software development, engineering, urban planning, and law [5]. Scrum employs a cyclical feedback process in which the current progress of a project is used to update project planning incrementally and produce products on an iterative basis. Development cycles, called Sprints, are only a few weeks long, which enables quick adjustments to changing requirements or to new knowledge gained during the sprint. The Scrum framework consists of [5]:

- **Members:** Product Owner, Scrum master, and Development Team
- **Events:** Sprint Planning, Stand-up, Sprint Review, and Sprint Retrospective
- **Artifacts:** Product and Sprint Backlogs
- **Definitions and rules** governing Scrum implementation

Figure 1 illustrates how Scrum works in practice. Its main steps include:

- In consultation with customers (stakeholders), the Product Owner defines requirements, features, and tasks that are needed for a successful product. These are formalized as User Stories, and are stored in a Product Backlog.
- Development Team Members also contribute User Stories to the Product Backlog. The Development Team includes the Scrum Master, who is responsible for smooth managerial functioning of the team.
- Each revolution of the Scrum process is called a Sprint. At the beginning of each Sprint, the Product owner and the Development Team select and prioritize User Stories from the Product Backlog. The Development Team selects items from the Product Backlog that they are committed to completing during the upcoming Sprint.
- The Development Team meets daily for brief, 10 to 15 minute, Standup meetings, during which everyone provides updates on the progress for each User Story. If roadblocks are identified, the Development Team takes additional actions, which may result in additions to the Sprint backlog.
- At the end of the Sprint, the Development Team and the Product Owner participate in a Sprint Review meeting. Together, they review all of the previously-selected User Stories, ensuring they are complete or identifying why they are not. At this point, the Development Team may demonstrate a prototype to Product Owner for their feedback.
- Following the Sprint Review meeting, the Development Team, without the Product Owner, conducts a Retrospective meeting to review the procedural performance of the team during the Sprint. The Retrospective is concerned with managerial performance, not technical issues, i.e., how effective was the team at managing their assigned tasks, how well did the team function, and how may managerial processes be improved?
- This cyclical processes then begins again with a new Sprint Planning meeting.

Frequent, but short, meetings lead to quick discovery of problems and makes each team member's progress visible to the entire team. The iterative nature of the process allows changes based on feedback received from both stakeholders and team members. The summary given above covers only the basic features of Scrum, but this is a sufficient framework for our discussion. More details can be found in, e.g., [5],[6].

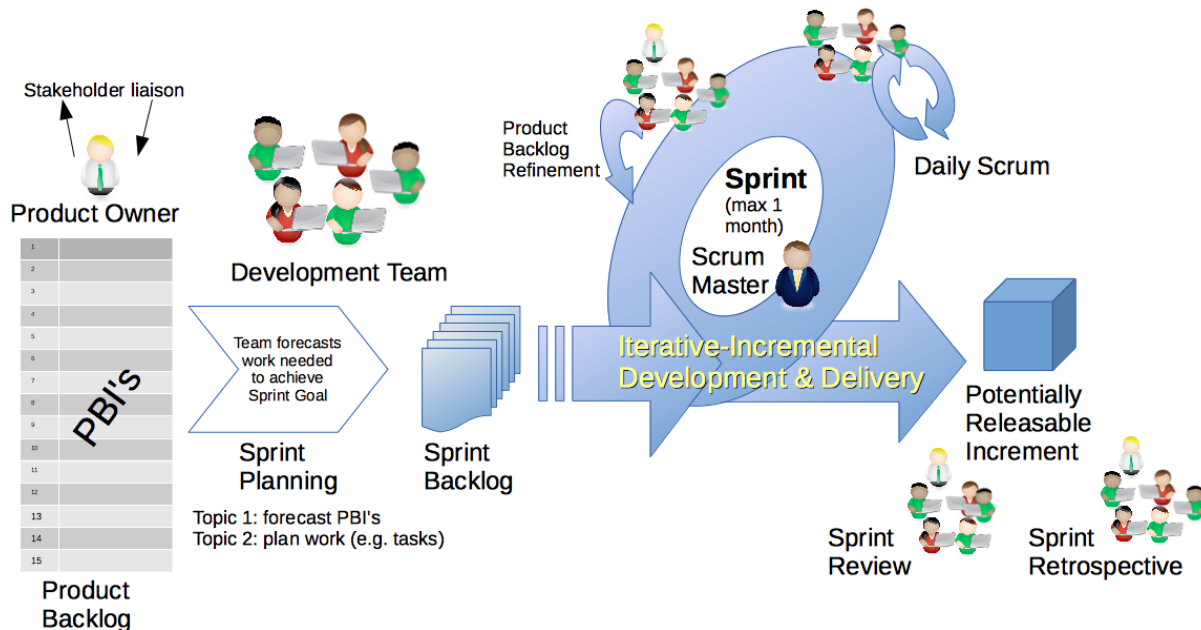


Figure 1: Elements of the full Scrum process (figure obtained from [22])

While Scrum and other Agile methods have been widely used in the software industry, their use in software engineering programs is still limited [7]. Even less has been reported on their use and effectiveness in other engineering disciplines [8]-[13]. Our team's initial findings on using Scrum in electrical and computer engineering program were reported in [14]. Next we present the case for using Scrum in engineering education.

### Projects and Scrum in Engineering Education

Projects are essential components of all engineering programs and are also part of ABET's requirement for "curriculum culminating in a major design experience," [1] which most often means a senior capstone activity that involves a substantial project-based component. In ECE programs at Portland State University (PSU), as in many others, projects are interspersed among many courses, with high concentration in the freshman year and culminating with the senior capstone project. To better connect the freshman and senior project experiences, we have

introduced two sophomore-level courses that serve as mini-capstone experiences, as discussed in the section on Sophomore Courses. In order to provide students with a unified learning experience, we share many project components among these and other courses, chief among them being Scrum project management. This allows students to gradually develop their proficiency in various aspects of project development and implementation. Scrum is a good fit for course projects since the short sprint cycles allow for multiple feedback points during the term, while the frequent stand-up meetings keep students focused on the project and in sync with their teammates between class periods.

When used within engineering courses and programs, Scrum provides several advantages over traditional project management techniques [8,14]:

- Rapid prototyping and incremental development of both software and hardware, which gives students a sense of progress as project deliverables incrementally improve throughout the project period.
- Quick feedback from stakeholders provides students with direction through constructive conversation.
- Discovery of core values, which are important to the customer but not obvious at the start of the project, become clear as the project develops.
- Decentralized project management allows all students to have opportunities to contribute in meaningful ways.
- Transparency in teamwork and project progression through frequent, but brief, meetings expose individual shortcomings or slow development pace. Team members can see the full picture instead of just their task, and then make adjustments accordingly.

Existing literature on using Scrum in engineering education deals almost exclusively with upper-division (capstone) or graduate engineering courses [8]. While this may be the most obvious place to utilize Scrum, for pedagogical reasons we should teach it across the curriculum and in the context of engineering courses and projects, and not as a separate course. For example, one of the important professional skills is technical writing. Literature has shown that technical writing ought to be taught and reinforced across multiple engineering courses and years [15,16]. For any project management technique to be really useful, we should adopt a similar approach, by teaching it early and reintroducing it often. For these reasons, we have been implementing Scrum project management within three years of our ECE undergraduate program.

We cannot expect freshman or sophomore engineering students to have the sophistication necessary for full implementation of Scrum. Furthermore, Scrum in educational environments is a teaching and learning tool. As such, it needs to be modified from its original design. Therefore, we have adjusted Scrum in the following ways:

1. Roles of Product Owner and Scrum Master are flexible and adjusted to specific level and course contexts.

- a. In freshman courses, neither is assigned and the whole team effectively serves in both roles.
  - b. For sophomore courses, Scrum Masters are junior or senior students assigned from outside the class, while the team serves as Product Owner.
  - c. For the senior pre-capstone course, the course instructor and teaching assistant serve as Product Owners. Students select one of the Team Members to serve as Scrum Master, a position some teams choose to rotate from sprint to sprint.
  - d. For capstone projects proper, a faculty member serves as the Product Owner, serving as a liaison between the Team Members and the client (an industry sponsor). The client attends some Sprint Review meetings. Team Members select one member of the team to serve as Scrum Master.
2. We require students use a kanban board to track and document User Stories.
    - a. In freshman courses, it is not mandatory yet for students to write a User Story in the classic Scrum style. Instead, the simpler approach of defining a goal and writing a list of required actions or activities can suffice at this stage.
  3. Teams are formed by instructors.
  4. Sprint Review and Retrospective meetings are introduced in the sophomore year.
  5. Daily Stand-up meetings are held 3 or 4 times per week at regularly scheduled times.
  6. Gantt charts are used for overall project planning and to visualize Sprints.
  7. Until their capstone year, we do not emphasize the final product; instead, more attention is paid to process: design, project management, teamwork.

Details of these changes are illustrated in sections below. As discussed, Scrum involves teams, so we have to support other educational goals, such as teaching students how to engage in effective teamwork.

### **Teamwork**

Effective teamwork is essential for project success. Typically, much of the “training” in teamwork is experiential - students may be given some theoretical framework(s) for understanding what is going on, but learning seems to happen during actual project work. Assessment of teamwork is non-trivial and tends to rely heavily on peer assessment, such as use of CATME [17]. It is also difficult to provide timely feedback to teams unless an instructor is overseeing all of the student teams on a daily basis. Frequent Scrum meetings under the supervision of the Scrum Master provide an opportunity to gain real-time insights. However, what is to be assessed and how to evaluate it is not obvious. In professional environments, teams may use burn-down charts or similar tools to gauge their effectiveness, but that is beyond what we can expect from undergraduate engineering students with limited experience. We are in the process of examining this problem in more detail.

## **Project Management Tools**

Many of our ECE courses use software applications on web-hosted platforms to assist students with their project management needs. CATME and Trello are the current tools we use, and their purpose is discussed here.

### *CATME*

The CATME SMARTER Teamwork website ([catme.org](http://catme.org)) is a resource that teaching staff uses to form project teams and for students to provide feedback about team performance. At the start of a course, students are asked to answer an online survey sent from CATME, which has questions such as gender identity, GPA, available time schedule, among others. The instructor assigns a weighting factor to each question in order to create teams of people having either similar or dissimilar attributes. Once all of the surveys are collected, we run the CATME Team-Maker module, which applies an algorithm to assign compatible students to teams automatically, though manual adjustments can still be made if needed.

Early in a project, students utilize the CATME Peer Evaluation service to rate their teammates' productivity and efficacy, as well as their own. Comments left by students can be viewed by other team members and the instructor. CATME compiles this information into a summary report, which can be used to detect if a team is becoming dysfunctional, when there is still the possibility of making corrective structural or behavioral changes. After the project is over, students are required to do a final peer evaluation, which the instructor may use to make grading decisions based on each person's participation and contribution. Self-reflection and the ability to write thoughtful evaluations are skills that we expect our students to learn and practice.

### *Trello*

One project management scheme that our ECE curriculum has adopted is the kanban board. It originates from the Kanban system, which is a scheduling method developed by a Japanese automotive company in the 1950s to improve manufacturing efficiency. In its contemporary form, a kanban board consists of "sticky notes" placed on a wall in a grid-like pattern. There are columns labeled Backlog, Ready, In-progress, Review, and Done, for example. The project is divided into a set of User Stories, which are descriptions of objectives. Each User Story is assigned its own sticky note along with acceptance criteria that must be met. A User Story starts in the Backlog column and moves to the Ready column once the needed resources for the story are available. When a User Story moves to the In-Progress column, it is actively worked on until the criteria are satisfied, after which it goes to Review for verification by the team. Once cleared, the User Story finally makes it to the Done column. At any given time, multiple User Stories may be at different stages in the pipeline. The main project goals are completed once all of the User Stories arrive at the Done column. Overall, the kanban board provides an efficient visual method for tracking the project's workflow and current status.

Using a physical wall and sticky notes provides a visceral connection to the team’s progress, and the shifting of notes from one column to the next is highly motivating. However, finding enough free wall space for multiple teams that will not be disturbed by others is often unfeasible. Hence, we have migrated to an online version of a kanban board, from a company named Trello (trello.com). Each team is responsible for managing its own Trello kanban board. Students create “cards,” which are the electronic version of sticky notes, and edit the cards to define the User Stories, criteria checklists, deadlines, and team member assignments. A card can be dragged and dropped from column to column, which mimics the movement of sticky notes on a physical kanban board. Example of a well-designed and utilized Trello board is given in Figure 2.

To ensure students use Trello consistently and in the manner we intend, staff conduct periodic Trello reviews. Weekly screen captures of boards are taken to document the team’s progress. This includes verifying that cards are indeed moving, checklists are being completed on time, and students are actively participating. As formative motivation, the staff provide feedback to each team regarding the quality of their board and how they could improve it. At the end of the project, a rubric is used as a grading tool to judge the team’s overall management skills, such as determining how much each student’s activity contributed to the team’s success. Trello is an important part of both the first and second year curriculum, since it introduces basic project management ideas and prepares students for Scrum.

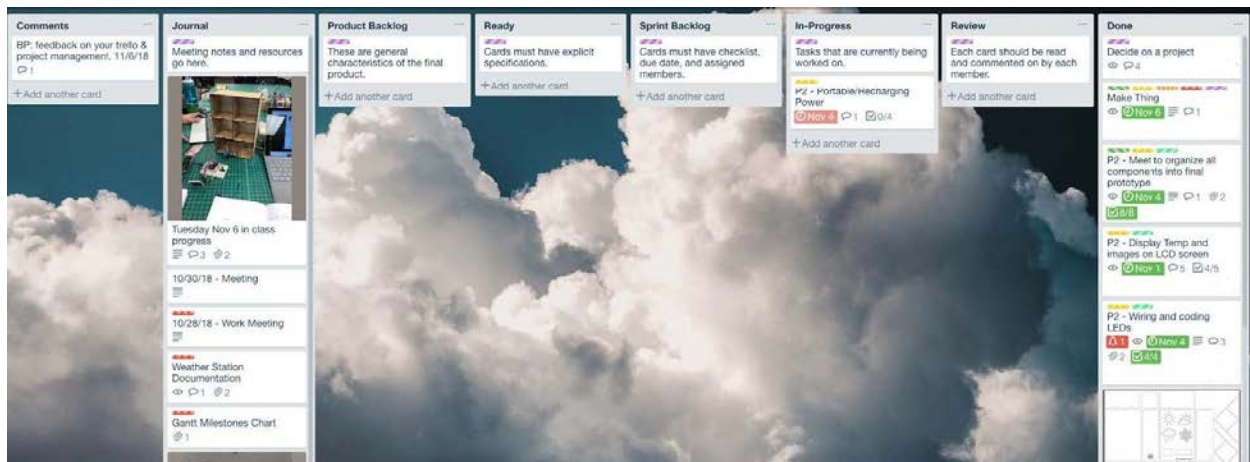


Figure 2: Example of a kanban board layout on Trello.

### Example of Scrum in ECE

We use Scrum to help students improve their teamwork efficacy in projects and courses. In previous publications, we presented our initial experiences and observations when implementing Scrum in ECE courses [14]. In this paper we elaborate on how Scrum is applied across different years and how we scaffold student learning.



## **Curriculum Overview**

Prior to 2010, first-year ECE students were enrolled in general “introduction to engineering” courses that were open to anyone, regardless of department. By and large, these courses did not have projects that required any formal PM methods. Eventually, we decided to offer our own introductory courses that were tailored to ECE students. From the start, the new ECE 101, 102, and 103 sequence mandated that student teams work on a substantial final project. The original projects were simplistic at first and often led to students creating their own ad hoc project management style. With time, the projects have evolved in complexity, and structured management techniques are now formally taught and practiced. In addition to this sequence, students are also expected to complete two courses in digital circuits and system design. At the sophomore level, students take a year-long electric circuits course sequence along with a newly-introduced ECE 211 and 212 sequence on design processes and project development.

Students are formally admitted into the EE or CMPE programs in their junior year, which is dominated by ECE courses in circuits, electromagnetics, signals and systems, power systems, and microprocessors, to name a few. Senior year is reserved for electives and a year-long capstone design sequence, which we described in [18]. Projects are featured in many courses, but are more concentrated in the freshman year and in the new sophomore courses. Obviously, the capstone sequence is a critical component in this string of projects across the curriculum.

One of the links connecting all of these project-intensive courses is utilization of Scrum project management and similar assessment tools. Our learning goals related to Scrum are grouped into four areas:

- A. developing projects and their components
- B. applying Scrum
- C. utilizing project management tools
- D. running effective teams

In the following three sections, we will discuss how these goals are accomplished at different years (freshman, sophomore, and senior), describe our recently-introduced project courses and how they use Scrum, present initial assessment data and observations, and mention what issues or problems still remain.

## **Freshman Courses**

Freshman students should

- A. Show basic skills in breaking down given assignments into team tasks
  - a. as demonstrated by Trello boards and in-class exercises
- B. Be able to show concrete evidence of planning their projects
  - a. as demonstrated by their Trello board, lists and cards
- C. Learn the basics of project planning tools

- a. by observing in-class Trello demonstrations and watching tutorial videos
  - b. by doing the rater practice exercise on CATME
- D. Learn how to run team-based projects with minimal team conflict
- a. as evaluated through the peer-evaluation platform CATME

Most students taking our first-year sequence are still learning the basics of project management and are not ready to adopt the full Scrum process. Taking this into account, in the freshman courses we only familiarize students with kanban boards, which lays a foundation for using Scrum in later design courses.

ECE 101 Introduction to Electrical Engineering is the first course, which gives new engineering students a chance to experience what the fields of electrical and computer engineering have to offer. Along with introducing core engineering topics such as problem solving and ethics, the course also features very simple circuits-based labs, which culminate in a multi-week long final project. CATME is used to generate teams of four to five students each, who design and construct a “Rube Goldberg”-like apparatus that incorporates electrical and mechanical elements to perform an otherwise simple function in an overly-complicated and amusing manner. Since the purpose of ECE 101 is to provide a low-key and fun entry into the ECE program for students from a variety of backgrounds, Scrum is not emphasized yet, though kanban principles are introduced via Trello.

The follow-on course is ECE 102 Engineering Computation, which introduces the MATLAB technical computing package as a computation aid and as a “first” programming language, as described in [19]. In addition, students are taught the fundamentals of engineering analysis in an electrical engineering context. Using CATME, students are assigned to compatible three-person project teams. Every team is given the same well-defined major project, such as constructing a scale model of a street intersection with working LED traffic lights and crosswalk switches, all under real-time MATLAB program control. The project is divided into a research stage, a construction/programming stage, and a final demonstration. At the end of the research period, students write a report describing their work and evaluate their own performance and that of their teammates using CATME. This information gives both the team and the instructor critical feedback on the team’s working dynamics and interaction. Another peer evaluation is done after the final demonstration. For long-term planning, each team is expected to define the major goals and timeline needed to complete their project. From this, they are asked to generate a traditional Gantt chart to visualize the overall project schedule. This can be done manually or by using specialized software, such as that available for free online. For daily management, the instructor provides a standardized kanban-style board on Trello to each team. The students decompose the project requirements into a set of goals and tasks, which they use to make the cards for their Trello boards. Teaching staff periodically review each team’s Trello work to assess its progress and to provide feedback.

The final course in the sequence is ECE 103 Engineering Programming, in which students learn to write programs in the C language to solve engineering problems. In addition, they apply their experience from the prior course to develop C code that interfaces with and controls hardware. This time, students may pick from a variety of predefined projects, which are now more complex and open-ended, instead of being highly specific and directed as in ECE 102. CATME is used to form three- or four-person teams and for processing peer evaluations. Trello is once again the primary tool for project management, though now the boards are reviewed with a more discerning eye toward the details of project decomposition.

To assess how students reacted to their first exposure to a kanban board, in 2015 and 2017 (27 and 58 students enrolled, respectively) we utilized an exit survey to ask ECE 101 students how helpful Trello was in managing their project. While the majority of students did find Trello to be useful or at least not detrimental, a combined 20% claimed that Trello was not worthwhile. A possible explanation is that project management is a new concept to many students, so the learning curve is an additional burden on top of the actual project itself. Some students can be skeptical of the usefulness of formal methods, especially if it takes more effort than improvising their own approach. They may exhibit resistance to using PM tools on a consistent basis, which hinders their ability to take full advantage of the benefits of proper management techniques. Getting freshman teams to embrace project planning is challenging, but introducing the concepts at this early stage and enforcing its use is the scaffolding needed for successful Scrum adoption.

### **Sophomore Courses**

For the learning goals A-D given in section Curriculum Overview we expect that sophomore<sup>1</sup> students should be able to:

- A. Develop projects from a starting idea and functionally decompose them
  - As demonstrated by developing a “product” from idea to functioning prototype
- B. Begin to apply the Scrum process more fully
  - Evaluated through regular sprint reports and direct observation by Scrum Masters
- C. Effectively use project planning tools
  - Evaluated through rubrics for Trello use
- D. Develop deeper understanding of team dynamics
  - Evaluated by Scrum Masters and through team contracts

The biggest development relative to freshman expectations is in the area “A. developing projects and their components.” Students now have much more freedom, but also more responsibility, to develop their project ideas. In the other three areas, we expect more incremental improvement. Students are expected to move up from barely “adequate” level of performance. For example,

---

<sup>1</sup> Many of our students transfer into ECE programs as Juniors and may be taking these classes.

whereas freshman may be asked to only show evidence of planning, at the sophomore level, we expect effective use of the same tools. In other words, we are starting to evaluate the quality of their use of planning tools, such as Gantt charts, for overall project planning and Trello boards for day-to-day operation.

We recently introduced ECE 211, Introduction to Design Processes, and ECE 212, Introduction to Project Development, with two additional goals in mind:

1. Teach students design and project development well before they encounter them in their Capstone projects.
2. Provide an environment for experiential learning where integration of various strands of electrical and computer engineering disciplines can happen.

Similarly to the first goal, some programs offer so-called “cornerstone” courses [3], but the majority of these seem to be freshman courses aiming to provide motivation for potential engineering students while providing somewhat authentic experiences. These freshman students, however, will typically not have enough technical background to accomplish the second goal. In our curriculum, students enrolled in ECE 211/212 will be very familiar with topics such as programming, problem solving, DC circuits, and digital logic design. They should also be taking a sophomore-level circuits classes concurrently. This background should enable them to undertake more challenging technical projects.

ECE 211 is a one credit course during which students:

1. Further develop their non-technical, professional skills, which were originally introduced at the freshman level, including teamwork, project management, product design, and ethics.
2. Implement one trial run of what we call a “practicum,” wherein teams of students are asked to accomplish a very specific engineering task by following principles discussed in item 1. above
3. Prepare a proposal of a project that a team will work on during ECE 212.

Obviously, this does not give us much time, but it has to be remembered that this course is only a stepping stone towards the senior capstone project and is, therefore, formative in nature. The Practicum component mentioned in 2. above is introduced due to our positive experience in using it as part of the capstone design course sequence [18]. There are relatively few restrictions on the type of project that students can undertake and which are to be described in item 3. Note that “design” is only one of several components in this short class. Among many topics, such as those found in, e.g., [20], we focus only on:

- identifying a specific need,
- refining initial idea through design cycle,

- identifying product requirements, and
- performing functional decomposition at L0 and L1 levels.

While the list is incomplete, we believe that if students indeed master these skills and practice them in ECE 212, they will be much better prepared for their capstone projects. We impose only a few technical constraints on the projects, including: using an advanced IoT-ready microcontroller (e.g., ESP32), developing a PCB for electronics, and using at least one sensor and one actuator. Students have a great deal of resources at their disposal through our student-run Makerspace lab [21].

ECE 212 is a two credit project-based course during which students work on bringing their design idea to fruition, while following the best practices established in ECE 211. The timeline for the project is based on Figure 1 and includes:

- idea development and project proposal during last 3 weeks of fall quarter of ECE 211
- final version of project developed during 1st week of winter quarter
- project implemented during four 2-week-long sprints
- project demos during the 10th week
- final project documentation submitted in the 11th week

All of the usual Scrum events are implemented: daily Stand-ups, Sprint Planning, Sprint Review and Sprint Retrospective. Project backlog, sprint backlog and all in-progress tasks are maintained within Trello. After finishing each sprint, teams submit simple reports for which a template is provided. Each team is assigned a Scrum Master, who is an undergraduate student with prior experience with projects and Scrum. Scrum Masters also submit a bi-weekly evaluation of their teams and provide feedback to each team about overall project progression. They use rubrics that we previously reported in [14]. Scrum Masters also participate in daily standups, which are held at least three times a week during the regularly scheduled class hour. Many teams schedule additional meeting times.

One person that is missing from this Scrum setup is the Product Owner. Given that the project is developed by team members, they are expected to perform some of Product Owner's tasks. In practice this means they are responsible for backlog pruning and prioritizing backlog tasks. They are helped in this by their Scrum Master and the course instructor, who both participate in Sprint Planning, Sprint Review, and Retrospective meetings.

To address the learning goal "D. Develop deeper understanding of team dynamics," we started using team contracts. These contracts are updated after students have gone through the practicum so that they can make meaningful modifications. We are currently developing rubrics to evaluate if students are indeed making progress in their understanding and practice of teamwork, and if that is reflected in their team contracts. Relatively few teams, around 2 out of 16 teams,

experience any serious problems with teamwork. CATME has been a very good tool to identify such teams. Furthermore, having Scrum Masters come from outside the class gives a detailed and first-hand insight into functioning of each team. However, we still need to work on training Scrum Masters more fully in all of their tasks, which are complex and include not only keeping teams on track but also providing them with technical and organizational feedback, as well as providing input to the instructor about the quality of work being done and student participation.

At this time our assessment is limited, but we can report on the project proposal reports that are submitted at the end of ECE 211. We use a rubric that is very similar to the one used in freshman courses [19]. Our initial observation is that most teams have done a good job, but we need to stress the critical importance of some items, such as a sketch of the proposed product. Note that at this level, we do not insist on more formal project documentation, such as test plans, which would be expected in capstone-level projects. In our initial student survey, we found that students are confident in their project management and teamwork skills, but they are also concerned about the amount of time and effort devoted to ECE 211, which is assigned only one credit. We are in the process of independently evaluating the former and will eliminate some topics to address the latter. This will happen through better integration of course topics between freshman and sophomore courses.

Overall, our initial implementation of Scrum within the context of sophomore-level project-based courses is progressing well. The remaining issues for both freshman and sophomore courses include:

- developing a more complete rubric for project management assessment in particular, and professional skills in general,
- training a cohort of undergraduate students to serve as Scrum Masters,
- coordinating content with freshman courses, and
- scheduling activities to accommodate a non-traditional student population that has many competing demands on their time outside of the classroom.

### **Senior Capstone Course Sequence**

Senior (capstone) students should

- A. Fully develop projects with clear functionality, specifications, and deliverables; adjust project goals to changing customer requirements
- B. Implement the full Scrum process
- C. Be fluent in using project planning tools
- D. Have very clear expectations and rules that lead to effective teamwork

During their senior year, students undertake a three-course sequence, which consists of a one-term pre-capstone course followed by two-term capstone project.

### *The Pre-capstone Course*

The objective of the pre-capstone course is to ensure students are prepared for their industry-sponsored capstone project. Until recently, ECE students were not exposed to formal project management techniques at the freshman and sophomore levels, so this pre-capstone course has been critical for preparing them for their capstone project. Industry partners sponsor all of our capstone projects, thereby ensuring all senior students engage in a practical, multi-term engineering project under the guidance of an industry professional. As such, this course has been critical for ensuring students are prepared to perform within a professional environment.

To date, the students who have gone through the Scrum-based projects at the freshman and sophomore levels described previously have not yet matriculated to the senior level. Whether the curriculum within the pre-capstone course will change once these students begin to matriculate to the 400-level has not yet been decided. The department serves a large number of transfer students who are not exposed to our lower-division project management curricula, so the course will still need to focus on developing project management skills in preparation for the capstone project.

Within the pre-capstone course, students undertake a term-long engineering project for which students must realize a set of engineering specifications. The project serves as the context within which students practice teamwork and project management. In line with the aforementioned lower-division projects, students use Scrum as the framework for project management. Students iterate through four or five Sprints during the term, which provide them with multiple opportunities to develop project management experience. Prior to each sprint, the course instructor assigns a new set of engineering specifications, which build upon those previously assigned. Students articulate the engineering of these specifications as User Stories for the upcoming Sprint. At the end of each Sprint, students conduct a CATME survey so that they may reflect on their teamwork experience and provide quantified feedback to the course instructor.

### *The Capstone Project*

Following the pre-capstone course, students are assigned an industry-sponsored capstone project for the final two terms of their senior year. The ECE department solicits capstone project ideas from industry partners early in the academic year. Students then select projects that they are interested in pursuing. Student teams are formed based on student interest as well as technical ability, course experience, and teamwork capacity. Capstone projects are real-world design projects relevant to the sponsor's company. As such, the capstone experience provides students an opportunity to apply their engineering experience in a professional scenario under the guidance of an experienced industry practitioner and an academic advisor. Capstone projects are substantial, multi-faceted experiences that require student to engage in engineering design, product prototyping, specification validation, project planning, and time management.

An ECE Faculty member assumes the role of Product Owner, who represents the industry sponsor during Sprint Planning and Review meetings. One of the capstone team members or a graduate student serves in the role of Scrum Master. The remaining seniors serve as team members. Together, the team participates in Scrum meetings and defines User Stories.

As is done for the pre-capstone project, students conduct a CATME survey at multiple points throughout their capstone experience. The survey provides them with an opportunity to reflect on their teamwork and project management, critique their teammates and provide feedback to the course instructor.

### *Capstone Sequence Assessment Examples*

The CATME peer-evaluation survey provides ready-made questions for assessing a variety of team-related performance metrics, including Task Commitment, Relationship Conflict, Process Conflict, Task Conflict, Team Satisfaction, and others. By tracking the CATME data, we can measure the students' perceptions of their teamwork, commitment to tasks, and enjoyment of the project work, as well as conflict-related metrics like task conflict, relationship conflict, and process conflict.

As an example, results from Task Conflict (T), Team Satisfaction (Q) are shown in Table 1 for each of the three courses in the capstone sequence. We consider 1-5 scores greater than 4 and 5-1 scores less than 2 to be exceeding expectations. The results in Table 1 show students meet or exceed expectations in these metrics. Tracking these results through the academic year provides insights in how perceptions of team satisfaction and task conflict change over time. For example, we observe that Team Satisfaction scores degrade by about 12 to 15% and Task Conflict scores increase by 14 to 18% during the capstone period.

**Table 1.** Results of the CATME peer-assessment tool showing results from six survey questions for each of the three terms of the capstone sequence. Scale range is 1-5.

For (Q), best performance is 5, while for (T), best performance is 1.

		EE 411, 2017 (28)		EE 412, 2018 (32)		EE 413, 2018 (32)	
		<i>Midterm</i>	<i>End Term</i>	<i>Midterm</i>	<i>End Term</i>	<i>Midterm</i>	<i>End Term</i>
<b>Q1</b>	I am satisfied with my present teammates	4.63 (0.48)	4.68 (0.55)	4.42 (1.04)	4.19 (0.89)	3.94 (1.20)	3.97 (1.22)
<b>Q2</b>	I am pleased with the way my teammates and I work together	4.63 (0.55)	4.64 (0.57)	4.35 (1.12)	4.29 (0.77)	3.88 (1.13)	4.21 (1.0)
<b>Q3</b>	I am very satisfied with working in this team	4.67 (0.54)	4.68 (0.55)	4.35 (1.09)	4.19 (0.86)	3.91 (1.18)	4.03 (1.22)
<b>T1</b>	How much conflict of ideas is	1.41 (0.62)	1.65 (0.91)	1.39 (0.49)	1.61 (0.75)	1.75 (0.87)	1.76



	there in your work group?						(0.93)
<b>T2</b>	How frequently do you have disagreements within your work group about the task of the project you are working on?	1.37 (0.48)	1.43 (0.92)	1.23 (0.48)	1.48 (0.76)	1.56 (0.79)	1.76 (0.82)
<b>T3</b>	How often do people in your work group have conflicting opinions about the project you are working on?	1.48 (0.63)	1.52 (0.88)	1.45 (0.61)	1.54 (0.71)	1.69 (0.88)	1.76 (0.93)

Data from CATME peer evaluation surveys provides feedback that can help improve our project management curricula. We would like to address a number of research questions in order to illuminate the efficacy of our efforts. For instance, do we see improvement in CATME scores between the junior and senior levels? Currently, students do not receive project management instruction in their junior-level courses, yet we administer the CATME survey in some of them. Also, can we observe differences in student experience between senior capstone groups that use Scrum and those that do not? Around two-thirds of our seniors do not take the Scrum-based pre-capstone course. Rather, they receive instruction in using the waterfall project management process in a parallel capstone course. So, can we observe differences between the cohort of students who use waterfall and those who use Scrum? And, do students use scrum when assigned term-long projects in courses that do not impose a project management structure? We may be able to make this observation as the freshman and sophomores who have been exposed to Scrum matriculate into upper division courses.

### Summary

The Electrical & Computer Engineering Department at PSU is in the process of developing a project management curriculum program that spans from the freshman to the senior levels. The program objectives include improving teamwork skills and understanding the value of project management processes. In addition to developing strong technical skills in our ECE students, we also aim to help them develop their professional skills, hence our focus on the development of project management and teamwork skills. To help with the latter, we have implemented the CATME Peer Evaluation survey tool, which probes our students' opinions about team conflict, task commitment, interpersonal cohesion, and team satisfaction.

In this paper, we argue that Scrum provides an appropriate framework for such development. Its main features, rapid prototyping, incremental development, quick feedback, decentralized project management, and transparency, lead to relatively easy adoption within an educational setting. We describe in detail the organization and scaffolding that we provide across three years: freshman, sophomore and senior. Though this multi-year program is still under development, we find the current results to be encouraging, and we intend to continue refining our

implementation. We also hope that we have provided sufficient details to provide an initial guidance for other faculty or programs interested in adopting Scrum in their courses or curricula.

## References

- [1] ABET Criteria for Accrediting Engineering Programs [http://www.abet.org/wp-content/uploads/2017/12/E001-18-19-EAC-Criteria-11-29-17-FINAL\\_updated1218.pdf](http://www.abet.org/wp-content/uploads/2017/12/E001-18-19-EAC-Criteria-11-29-17-FINAL_updated1218.pdf), Last accessed 2/4/2018.
- [2] D. Elzey, "Teaching Intro to Engineering in Context – UVA Engineering's New Cornerstone," in 2006 ASEE Annual Conference & Exposition, 2006, pp. 11.1212.1-11.1212.8.
- [3] K. G. Gipson, J. J. Henriques, and S. Sequeira, "Development and Implementation of a Cornerstone Course: Engineering Opportunities," in 2015 ASEE Annual Conference & Exposition, 2015, pp. 26.509.1-26.509.19.
- [4] S. A. Ambrose, M. W. Bridges, M. DiPietro, M. Lovett, and M. K. Norman, *How learning works: seven research-based principles for smart teaching*, 1st ed. San Francisco, CA: Jossey-Bass, 2010.
- [5] The Scrum Guide, <http://www.scrumguides.org/scrum-guide.html> . Last accessed 2/3/2019.
- [6] K. S. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, Upper Saddle River, NJ: Addison-Wesley Professional, 2012.
- [7] M. R. Marques, A. Quispe, and S. F. Ochoa, "A systematic mapping study on practical approaches to teaching software engineering," in 2014 IEEE Frontiers in Education Conference (FIE) Proceedings, 2014, DOI: 10.1109/FIE.2014.7044277.
- [8] M. Edin Grimheden, "Can agile methods enhance mechatronics design education?," *Mechatronics*, vol. 23, no. 8, pp. 967–73, Dec. 2013.
- [9] Reith, C. Bachmeier, and I. Gross, "Be an Engineer right from the start: An introductory project with Rube Goldberg machine and SCRUM," in 2015 IEEE Global Engineering Education Conference (EDUCON), 18-20 March 2015, pp. 249–54.
- [10] Valentin, J. Carvalho, R. Barreto, "Rapid Improvement of Students' Soft-skills Based on an Agile-process Approach," in 2015 IEEE Frontiers in Education Conference (FIE), 2015, DOI: 10.1109/FIE.2015.7344408.
- [11] R. de Souza, S. Zorzo, D. da Silva, "Evaluating capstone project through flexible and collaborative use of Scrum framework," in 2015 IEEE Frontiers in Education Conference (FIE), 2015, DOI: 10.1109/FIE.2015.7344249.
- [12] L. Collingbourne, W. Seah, "Teaching Project Management using a Real-World Group Project," in 2015 IEEE Frontiers in Education Conference (FIE), 2015, DOI: 10.1109/FIE.2015.7344301.
- [13] Rover, C. Ullerich, R. Scheel, J. Wegter, and C. Whipple, "Advantages of agile methodologies for software and product development in a capstone design project," in 2014 IEEE Frontiers in Education Conference (FIE) Proceedings, 2014, DOI: 10.1109/FIE.2014.7044380.
- [14] B. Pejcinovic, R. B. Bass, and P. Wong, "Assessing Scrum Project Management and Teamwork in Electrical and Computer Engineering Courses," in 2018 ASEE Annual Conference & Exposition, Salt Lake City, Utah, 2018.
- [15] S. Conrad, W. A. Kitch, T. J. Pfeiffer, T. R. Smith, and J. V. Tocco, "Students Writing for Professional Practice: A Model for Collaboration Among Faculty, Practitioners, and Writing Specialists," in 2015 ASEE Annual Conference and Exhibition, Seattle, WA, 2015, DOI: 10.18260/p.24769.

- [16] S. Conrad, "A Comparison of Practitioner and Student Writing in Civil Engineering," *J. Eng. Educ.*, vol. 106, no. 2, pp. 191–217, Apr. 2017.
- [17] M.W. Ohland, M.L. Loughry, D.J. Woehr, C.J. Finelli, L.G. Bullard, R.M. Felder, R.A. Layton, H.R. Pomeranz, and D.G. Schmucker, "The comprehensive assessment of team member effectiveness: Development of a behaviorally anchored rating scale for self and peer evaluation," *Academy of Management Learning & Education*, vo. 11, no. 4, pp. 609-630, 2012.
- [18] M. Faust, A. Greenberg, and B. Pejcinovic, "Redesign of senior capstone program in electrical and computer engineering and its assessment," in *2012 Frontiers in Education Conference Proceedings*, 2012, DOI: 10.1109/FIE.2012.6462390.
- [19] B. Pejcinovic and P. Wong, "Evolution of an Introductory Electrical Engineering and Programming Course," in *2017 ASEE Annual Conference & Exposition*, Columbus, OH, 2017.
- [20] R. Ford and C. Coulston, *Design for Electrical and Computer Engineers*, Boston: McGraw-Hill Education, 2007.
- [21] Electronics Prototyping Lab at Portland State University, <http://psu-epl.github.io/> , Last accessed Feb. 25, 2019.
- [22] Obtained from [https://commons.wikimedia.org/wiki/File:Scrum\\_Framework.png#](https://commons.wikimedia.org/wiki/File:Scrum_Framework.png#), last accessed 2/4/2019. Licensed under Creative Commons license [https://en.wikipedia.org/wiki/Creative\\_Commons](https://en.wikipedia.org/wiki/Creative_Commons)