Curricular Hackathons for Engineering Design Learning: The Case of Engineering Design Days

Christopher Rennick, *Member, IEEE*, Greg Litster, C. C. W. Hulls, *Member, IEEE*, and Ada Hurst

Abstract*—*Contribution: **This article defines "curricular hackathons" and describes the features, challenges, and opportunities of adopting the hackathon format into engineering curricula. It is based on experience implementing a series of curricular hackathons which provide students with formative design experiences as they address ill-structured and integrative design problems.**

Background: **Design is an essential part of engineering practice. However, undergraduate engineering students are provided few opportunities to develop the skills necessary for solving ill-structured and complex design problems prior to their final year capstone projects. Cornerstone projects and other summative activities are common attempts to introduce students to design, but often carry significant academic weight in the host course. Adapting the hackathon format into curricular elements could provide a useful pedagogy for formative design practice.**

Research Question: **How can the hackathon format be effectively adapted into curricular activities?**

Methodology: **The study uses a multiple case study research design to extract insight from 12 implementations of curricular hackathons in engineering at the University of Waterloo. The twelve cases were analyzed through two methods: 1) a synthesis of eight previous publications and 2) an interview study of 12 instructors involved in their design and implementation.**

Findings: **Adapting the popular hackathon format to a curricular setting requires several adaptations to maximize the impact on students. Curricular hackathons: are a short, high intensity social experience; that guide students through the design-build-test cycle of a design problem, include opportunities for reflection; and achieve some level of integration and/or embedding in a program's curriculum.**

***Index Terms*—Co-curricular, design process, design projects, experiential learning.**

Manuscript received 15 August 2022; revised 27 April 2023 and 17 June 2023; accepted 12 July 2023. This work was supported in part by the NSERC Chair in IDEAs and in part by the University of Waterloo LITE Grant. *(Christopher Rennick and Greg Litster are co-first authors.)* *(Corresponding author: Christopher Rennick.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the University of Waterloo's Office of Research Ethics under Application No. ORE 41348.

Christopher Rennick is with the Engineering IDEAs Clinic, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: crennick@uwaterloo.ca).

Greg Litster was with the Department of Management Sciences, University of Waterloo, Waterloo, ON N2L 3G1, Canada. He is now with the Institute for Transdisciplinary Engineering Education and Practice, University of Toronto, Toronto, ON M5S 1A1, Canada (e-mail: greg.litster@mail.utoronto.ca).

C. C. W. Hulls is with the Mechanical and Mechatronics Engineering Department, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: chulls@uwaterloo.ca).

Ada Hurst is with the Department of Management Science and Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: ada.hurst@uwaterloo.ca).

Digital Object Identifier 10.1109/TE.2023.3295754

I. Introduction

FAR TRANSFER and integration of knowledge are difficult and distinct cognitive skills for students to acquire [1]. Knowledge integration is frequently a requirement in solving the types of design problems that are common in engineering; these problems have vaguely defined goals, multiple criteria for evaluating solutions, many unstated constraints [2], and are often multidisciplinary in nature. The nature of design problems makes them ideal situations to practice application and integration of curriculum knowledge.

Traditionally, engineering students have been given few opportunities to develop the cognitive skills necessary for solving these problems prior to their final year capstone projects. First-year engineering students can struggle to navigate the ambiguities present in ill-structured problem solving, especially when collaborating with their peers [3]. In addition to the cognitive skills required to solve these types of problems, novice students also need to develop their confidence in their ability to succeed at this task. According to Bandura [4], self-efficacy—a person's belief in their ability to succeed at a task—develops through exposure to mastery experiences (overcoming obstacles and persevering), social persuasion, social modeling (seeing peers persevere and succeed in a task), and through developing their physical and emotional states.

Initiatives that support undergraduate students' early development in design practice—to build both cognitive skills and confidence—have grown in popularity. Several large curricular initiatives [5], [6], [7], [8] have brought in holistic experiences where students can practice applying and integrating their curricular knowledge. While varied in their implementations, these experiences are long in duration; and require substantial curricular changes to implement, potentially limiting the ability of other institutions to emulate their successes. Cornerstone projects have also grown in popularity in first year engineering curricula to provide students with a mastery design experience in a supportive environment. However, cornerstone projects are typically implemented as a summative project with significant academic repercussions should a student fail at the task, and require substantial course redesign(s) to implement. To counter this risk for students, they are then highly scaffolded to ensure students move through the design process smoothly and with limited frustration. How then, can engineering schools provide additional exposure to high-impact [9], formative, team-based design experiences?

A promising new format for achieving these goals is that of the hackathon. This article highlights the promise of curricular hackathons—educational activities in which the hackathon format is adopted to support design learning in a curricular setting. This article provides insight and reflection from the years designing and implementing the engineering design days (EDDs) at the University of Waterloo to demonstrate the value of these hackathon-like events as a novel pedagogical tool for engineering educators. Building on this experience and synthesis, this article also presents a guiding definition of a "curricular hackathon" for others who wish to build on this pedagogy.

The remainder of this article is organized as follows. The following sections present an overview of design processes and design learning at hackathons and an introduction to the hackathon format that has been adopted in engineering curriculum at the University of Waterloo in the form of EDD. Then, the aims of this study and the methodology employed are described, followed by a detailed description and discussion of how features of the curricular hackathon pedagogy are embodied in EDD activities. This article concludes with a final reflection and implications for engineering design educators.

A. Design at Hackathons

Hackathons are time bounded events where participants gather and work together in teams to design a solution to a problem [10], [11], [12]. In their original form, hackathons were events where computer programmers, developers, and designers would collaborate in teams with the aim of solving complex software-related problems or producing innovative technologies. Since the initial inception, hackathons have grown significantly in popularity, with thousands held globally each year. Single events typically have a theme (e.g., public engagement, environmental issues, and spreading knowledge about new technologies) [12] and attract a particular audience (e.g., *StarterHacks* [13] attracts first time hackathon participants).

In a review of design activity at hackathons, Flus and Hurst [11] outlined the common elements of hackathons and use the Double Diamond design process [14] to model the steps that participants follow during a hackathon. Hackathon participants follow a similar design process to that exhibited in other more common design tasks, with the main design activities at hackathons roughly mapping onto the Discover-Define-Develop-Deliver stages [11], [15]. Hackathons typically begin with an introduction to the structure and theme (if there is one) of the event. This is a phase of Discovery, where participants engage in the critical activity of team formation (if not formed prior to the event) and search for an idea that takes advantage of the team's skillset, is feasible to design and build within the hackathon duration, and aligns favorably with the event theme and sponsors to improve chances of winning prizes. In the subsequent phase—Define—hackathon teams work to better understand the problem they are tackling, outline the requirements for their design, and create a project development plan to structure the work to be done and team members' roles and responsibilities. Teams will then begin working on their solutions (the Develop phase) through an iterative process of brainstorming, prototype building, and testing. In the final (Deliver) phase of the hackathon, participants present or pitch their final solution, usually to a panel of judges that award prizes for the best projects. All this work is typically completed in a condensed time frame of 24–48 h, though some events can last longer. Hackathons have become extremely diverse, with variations in each of the above characteristics; however, some common features are present.

1) Hackathons are *short-duration, immersive events*.
2) *Hackathons Include Hands-On Application of Design:* Participants tend to implicitly follow a Discover-Define-Develop-Deliver design process.
3) *Hackathons Are Very Social:* Participants work in teams to solve open-ended problems, surrounded by more teams of their peers and/or near peers.
4) *Low-Stakes:* Extra-curricular hackathons do not have any academic stakes.

B. Hackathons for Design Teaching and Learning

Though these examples demonstrate the draw and effectiveness of hackathon-like events, the format also poses challenges. The un-interrupted immersion into a problem over a short period of time creates challenges to the overall design process for students: there may be parts of the design process that are overlooked or rushed through, increased levels of fatigue are a common (often unhelpful) occurrence, and competition may limit overall peer learning [16]. Nonetheless, with intentional and careful planning the hackathon format may also bring significant advantages for teaching design to engineering students in curricular settings.

As a result of hackathons' rise in popularity and their ability to elicit the design process among participants, the hackathon format has also been adopted by universities for curricular purposes, both in and outside of STEM fields [17]. Studies have found that after participating in hackathons, students have increased reference to design thinking processes and an increased awareness of where it might be implemented [18]. For example, Uys [19] used a 24-h period toward the end of a capstone design course for students to implement a project they had developed throughout the semester. Students reported that they felt more prepared for work in industry, having developed both hard and soft skills (e.g., teamwork, project management, time management, and creativity) during the event. Gama et al. [20] used a similar method in an undergraduate *Web of Things* course. Students developed their solutions to the final course project during the hackathon. Lappeenranta University of Technology (LUT) organizes hackathon activities, called code camps, in which software engineering students learn new technologies and implement solutions to a predefined challenge. In the various implementations at LUT, these events have emphasized different aspects of engineering education in addition to the technical outcomes of the event, including teamwork and leadership [21]. Finally, Harz University of Applied Science offers similar 48-h code camps, with learning objectives like gaining knowledge of multiple technologies, understanding of societal problems and helping students achieve a clearer picture of their interests [21].

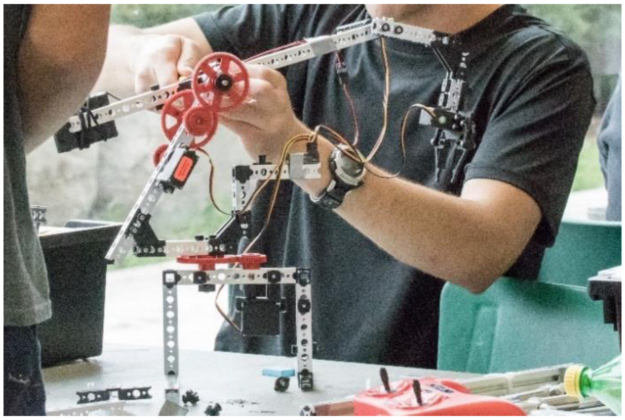

Fig. 1. Student working on robotic arm prototype from Mechatronics Engineering.

C. Engineering Design Days

When adopted into a curriculum, the hackathon structure allows teams of students to engage with an entire design cycle in a short period of time, in a high-intensity but low-stakes environment, easing the introduction of these new experiences to a very full curriculum. At the University of Waterloo, the hackathon format has been successfully adapted into short duration curricular design experiences called EDDs, which have now been successfully implemented in all 14 engineering programs at the institution. Since the first offering of an EDD in 2016, almost 10 000 students have participated in one of these academic events. Having delivered each EDD several times (totaling more than 45 individual offerings), iterated on their structure and gathered feedback on this process, significant insight has been garnered into the promise of curriculum embedded, hackathon-like design experiences, hereon referred to as curricular hackathons.

The development of EDD began in 2015 by the Pearl Sullivan Engineering IDEAs Clinic in the Faculty of Engineering with the intention of introducing design-centered activities to engineering curricula to mitigate the problems of heavily siloed engineering programs. EDD provide students an opportunity to engage in open-ended problem solving and develop hands-on design and professional skills, like teamwork and communication [22], which are strengths of the traditional hackathon format. EDD activities require few structural changes to the existing curriculum to implement—typically requiring only coordination with the term's instructors to find a suitable time to hold the event—and are fundamentally formative in nature.

While EDD events superficially resemble hackathons in format, they differ in some key ways as well. EDD events include a prescribed problem, a structured process, and are designed to achieve specific course-level learning outcomes. The design problem is chosen to maximize the potential for application and integration of curricular knowledge, for example, one of the EDD offered in Mechatronics Engineering integrates mathematical modeling, programming, and mechanical design using a robotics problem (see Fig. 1).

Most EDD activities use a similarly structured design process, typically with the following stages.

Stage 1 (Warmup): Student teams collaborate to solve simple warm-up problems designed to: a) elicit connections to/between course content and b) to reactivate their domain knowledge.

Stage 2 (Design): Teams are presented the same authentic, open-ended problem along with basic building materials. Students then design solutions to the problem.

Stage 3 (Build): Teams collaborate to build their proposed solution to the assigned problem. This stage requires teams to iterate on their design.

Stage 4 (Test and Reflect): Teams test, and demonstrate their design, and reflect on the outcome.

These stages differ somewhat from the Discover-Define-Develop-Deliver format commonly seen at extra-curricular hackathons, but have been tweaked to emphasize curricular connections in the design problem and to provide structure to the design process for novice design students. Most student participants conclude an EDD event with a functional prototype (though not all groups are able to achieve this). A total of 5333 students have participated in these specific EDD activities since 2015.

II. Objectives

It is believed that EDD effectively embody the three features of extra-curricular hackathons previously mentioned; EDD:

1) are short-duration, immersive events;
2) include hands-on application of design;
3) are highly social; and
4) are low stakes.

Their present form is a result of a large-scale collaborative effort of many faculty and support staff across multiple departments who have fine-tuned their delivery year-after-year. It is unclear, however, if there are other commonalities in EDD implementations which can be generalized for others' benefit. The research described in this article seeks to collect and organize the collective knowledge of the many faculty and staff who have developed and delivered EDD events. For any other educators seeking to adopt curricular hackathons at their institutions it may be useful to learn from this collective experience about how curricular hackathons are implemented in practice. Guiding this investigation is the following research question:

> *How can the hackathon format be effectively adapted into curricular activities?*

The objective of this article is thus to provide a detailed description and discussion of the EDD as a prototypical case study of curricular hackathons, with the aim of developing a consensus definition of curricular hackathon which reflects best known practices.

III. Method

This investigation follows a multiple case research design, an approach that allows researchers to compare findings across a number of similar cases in order to increase generalizability [22]. The cases that make up this study are the 12 individual EDD "events"— C1 to C12–listed in Table I. The selected cases all share the structured design process described in Section I-C that is prototypical of EDD. Each

TABLE I
OVERVIEW OF EDD ACTIVITIES AND DATA SOURCES USED

Case label	Program & level (offerings)		Design challenge description	Duration (timing during term)	Associated course grades	Prior Publication			Interviewee		
						Ref	Term(s) described	Student Pop'n	Identifier (role)	Term discussed	Student Pop'n
C1	Architectural Engineering (once annually, 2018-2022)	1A	Piece of furniture for an assigned location on campus	2 consecutive days (week 1)	No grades	[28]	Fall 2018	90	I9 (academic staff)	Fall 2019	78
C2	Civil Engineering (2017-2022)	2A	Highway layout and river crossing (bridge)	2 consecutive days (week 10)	Grades in multiple courses	[26]	Fall 2017	121	I1 (faculty)	Fall 2019	120
C3	Electrical, Computer Engineering (twice annually, 2017-2018)	1B	Mobile ping-pong ball launcher	2 consecutive days (week 5)	Bonus grades in one course	[22]	Winter 2017 Spring 2017	140 258	-	-	-
C4	Electrical, Computer Engineering (once in 2019)	1B	Sun-tracking solar panel	2 consecutive days (week 5)	Bonus grades in one course	-	-	-	I12 (faculty)	Winter 2019	134
C5	Environmental, Geological Engineering (once annually, 2018-2020)	2A	Build an earthen dam	2 consecutive days (week 8)	Grades in one course	[27]	Winter 2018	60	I10 (faculty)	Winter 2019	63
C6	Management Engineering (once annually, 2015-2017)	1A	Software decision support tool for student travelling for work	3 days (1 day in weeks 4, 9, and 11)	Grades in one course	[23]	Fall 2015	75	I2 (faculty)	Fall 2015	75
C7	Mechanical Engineering (twice annually, 2017-2018)	1B	"Tipping Bucket" water park feature	2 consecutive days (week 2)	Grades in one course	[22]	Spring 2016	104	-	-	-
C8	Mechanical Engineering (twice annually, 2019-2020)	1B	Water-powered mechanical clock	2 consecutive days (week 2)	Grades in one course	-	-	-	I11 (faculty)	Winter 2020	120
C9	MechatroniC Engineering (once annually, 2017-2019, 2022)	1A	Remote controlled robotic arm	2 days (1 day in weeks 4 and 5)	Grades in multiple courses	[22] [29]	Fall 2017 Fall 2019	219 254	I3 (academic staff) I11 (faculty)	Fall 2019 Fall 2018	260 220
C10	MechatroniC Engineering (once in 2020)	1A	Video game and custom controller (conducted remotely)	2 days (1 day in weeks 4 and 5)	Grades in multiple courses	[29]	Fall 2020	220	I4 (faculty)	Fall 2020	230
C11	Nanotechnology Engineering (once annually 2019-2020)	1B	Scanning tunneling microscope frame, tip, and amplifier	2 days (1 day in weeks 10 and 11)	Grades in multiple courses	[25]	Winter 2019 Spring 2019 Winter 2020	110 99 115	I8 (academic staff)	Winter 2020	115
C12	Software Engineering (once annually, 2018-2022)	1A	Autonomous systems for a digital spaceship simulation	4 half days (weeks 4 and 5)	Grades in one course	[24]	Fall 2018	120	I5 (post-doc) I6 (faculty) I7 (faculty)	Spring 2020 Fall 2019	48 130

case represents a unique EDD design—as described under the "Design challenge description" heading in Table I. The study aims to identify similarities and differences in how the various EDD cases have adapted the hackathon format into curricular activities.

Two data sources were used in the case studies—previous publications describing and evaluating various individual EDD implementations, and an interview study with academic staff responsible for the design and delivery of each EDD—as detailed below. The combination of these two methods reinforces the identified themes and triangulates the qualitative interview data with other sources.

A. Synthesis of Prior Publications

Various prior publications have described and evaluated individual EDD offerings. In this article, those prior publications are reviewed with an eye on extracting patterns that justify their positioning as curricular hackathons. Publications were included in this review if they described an EDD activity (case) that possessed the hackathon features highlighted in Section II (note: there are other implementations of EDD that do not exactly match the criteria outlined above, e.g., [31]).

In all, eight publications were included in this analysis, included as citations [23], [24], [25], [26], [27], [28], [29], and [30], that provided insight on all but two of the 12 cases. The mapping between each publication and the EDD case that it evaluates is provided in Table I. The table also lists the specific EDD offerings and participating student populations reported on in each respective publication. The EDD offerings described in the eight included publications had a total population size of 2080 students.

The evaluation of EDD offerings has most often been accomplished by surveying student participants; most frequently through an anonymous post-event survey [25], [27], [28], [29], or with a pair of matching pre- and post-event surveys [23], [26], [30]. Some other methods have been employed, typically in combination with surveys, including focus groups [25], or by examining student performance in a course or courses in the same term [27].

B. Interviews

Computer assisted interviews were conducted with instructors during the spring and fall of 2020 to learn more about the EDD activities they had developed and run. The results presented in the following sections were gathered as part of a larger research project related to EDD (a description of the methodology and objectives of that project can be found in [32] and [33]). The study had received approval from the

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

institutional research ethics board at the University of Waterloo (ORE #41348). A full discussion on the validation of this methodology is presented in [32].

The interviews were conducted virtually over Microsoft Teams and were between 60 and 90 min in length. During the interviews, facilitators (1–2 members of the research team) guided study participants through a series of questions that elicited precise details about their EDD activity. The goal was to generate insight into how each activity was designed, what its intended learning outcomes were and to what extent the instructor perceived the activity to be successful. Throughout the interview, participants were asked follow-up questions and encouraged to provide additional details and reflections that were otherwise not captured by the prepared questions.

In total, 12 interviews were conducted. Interview participants (summarized in Table I as I1 through I12) were faculty members, academic staff (technologists, laboratory managers, or instructional support staff), or post-doctoral researchers at the university. Participant involvement in the EDD of interest varied, however, they were all a part of either the planning and/or facilitation of their respective EDD. Each interview focused on one implementation of a single EDD (as detailed in Table I), however, many of the interviewees have observed and/or developed multiple EDD activities, occasionally resulting in comments on other EDD activities they were involved in as well. The EDD implementations discussed in these interviews had a total reach of 1730 students.

Transcripts of the recorded interviews were automatically generated using Otter. The text underwent an inductive content analysis [34]. The analysis process began with open coding of the data, followed by an iterative grouping of codes by similarity to create categories, which were further collapsed into larger categories representing broader common themes.

IV. Results

The content analysis of the interview data identified three main themes: 1) EDD events are short, high-intensity social experiences; 2) provide students an opportunity to learn the design process through a design-build-test experience; and 3) feature curricular integration. These themes are presented in the following sections, illustrated with direct quotes from interviewees. These data are also triangulated with direct evidence extracted from the synthesis of prior studies on EDD implementations (as described in Section III-A).

A. Short, High-Intensity Social Experiences

Hackathons typically run continuously over a 24–36 h period (usually over a weekend) with participants working overnight with few breaks (if any) [11]. In EDD, the hackathon format is adapted so that the students' time and other commitments are balanced, while still maintaining some elements of the high-intensity environment that is characteristic of hackathons. For example, the Mechatronics Engineering EDD (C10) occurs over two full days held one week apart [30]. Even when an EDD occurs over two consecutive days, as was the case of C3 (an Electrical and Computer Engineering EDD) [23], the activity deliverables are scaffolded such that at the beginning of each day, students are provided with a separate list of deliverables that are due at the end of that day. This ensures that overnight work between the two days is not possible.

Extra-curricular hackathons are very social events, with teamwork and professional networking being common features [35]. EDD seek to capture that social experience by providing a common experience for a cohort of students to participate in. This typically manifests in not only intended learning outcomes for EDD events that directly relate to professional skills like communication and teamwork but also broader goals that relate to building of class community. In various surveys, the majority of students participating agreed that their EDD activity provided a good opportunity to meet their classmates [23], [25], and/or to get to know their instructor(s) better [29].

Activities which take place early in the 1A term, like for example the Architectural Engineering EDD (C1), which takes place in week 1, and the Mechatronics Engineering EDD (C9 and C10), which take place in weeks 4 and 5, have an emphasis on building class community and on introducing students to the discipline. The interview data provide rich evidence of the community-building potential of EDD. One interviewee commented on the potential for the activities to bring the cohorts together, pointing out the social modeling that can happen during EDD activities:

> '[The Mechatronics Engineering EDD activity]... gels the class together. [The students] came to know at least 20 people in the class... They also started to realize that ...other students are equally capable because they saw their work...They were forced to form communication channels amongst each other...They were forced to talk to each other. So that broke the ice, right. Basically, it transformed the class, from a set of individuals to a cohort of connected individuals.' - (I4, on C10)

Some other EDD activities, like the Civil and Mechanical Engineering implementations (C2, C7, and C8), provide an opportunity for students to work together in same teams that they will be working with on course projects later in the term. Though the topic of the EDD activity may differ from their course project, the students still have the chance to familiarize themselves with their team and have an initial design experience together. One interviewee discussed their choice in holding the Software Engineering EDD activity (C12) early in the term:

> 'We do it that early because we wanted them to get to know each other that early, to get engaged... So we thought that would be a good start for them to... encourage them talk to work together ... So at least they will get to know another 15 students, ...and work with them. And, you know, start at least with a small community' – (I7, on C12)

EDD activities also provide a means for the teaching team and the students to connect. One interviewee who offers an EDD activity early in the 1A term in the Software Engineering program commented on the usefulness of embedding teaching team members, like teaching assistants (TAs), with the team to help with breaking the ice:

> 'Well, yeah, I think we learned this summer, [...] is just a huge volume of TAs··· It really helps with those first two challenges that we identified of the students like getting started and building that team. Having a TA, or a volunteer who breaks the ice, gets them talking to each other, helps' - (I6, on C12)

The activities also offer an opportunity for students to connect with industry partners. One interviewee commented:

> 'Having industry involved... plays a big role, because then the students see the most more benefits in it. So what we did with the Enviro/Geo design days for the 2A term, where we had some industry come and do some presentations was very helpful. And it did add a lot of value to the students.' – (I1, on C2)

Industry partnerships with the activities are still in the early stages of development with only a couple of EDD implementations having implemented some component of industry involvement into their activities. The instructor quote above suggests that students found value in the insight from industry partners in guiding or informing their design choices. Though the industry representatives will not directly benefit from the artefacts produced by the students during these events, they still have the opportunity to observe student capabilities during the activity. As such, industry involvement in EDD activities often aligns with their own organizational priorities around talent acquisition.

Finally, EDD activities can provide an early setting in which student leaders emerge from the class:

> 'One of the things was, they got to meet other colleagues. The second thing is, it led to the rise of nascent leadership candidates. So in other words, the students who eventually became the class leaders, they kind of emerged during this activity. And for the right reasons, these students had the right empathy with the other students, they were very good at communicating the things and they understood the role of the instructors. So it was very amazing to see that.' – (I4, on C10)

B. Learning the Design Process Through the Entire Design-Build-Test Cycle

Design is obviously a core feature of extra-curricular hackathons, and this emphasis was maintained throughout the EDD curricular hackathons implemented at Waterloo. Lab or other tutorial activities often have a "minimum path" structure where all students are expected to leave the lab having gained some skill or piece of knowledge. EDD activities are different in that there are limited instructions given to students beyond the design goal for the activity—though there is still some structure in place to assist students in navigating the engineering design process. This open-ended opportunity is more closely aligned with the hackathon structure and better represents the ill-structured problems students can expect to find in real engineering practice.

Student surveys have shown that they perceive EDD activities to be a good setting in which to practice design [29] as well as related skills like teamwork [24], communication, and problem solving [23], [26], [30]. Similarly, by adapting the hackathon format, EDD activities create an environment where students have freedom to pursue creative solutions to the problem, but where they are supported—intellectually and emotionally—to increase the odds of success and lower the instances of frustration, with an aim to improve design self-efficacy. In this way, EDD events are most typically formative in nature, focusing on design process quality and not product quality—a stark contrast from cornerstone or capstone projects which are summative in nature:

> 'I think in the [Software Engineering] capstone projects... I push them more on product... At that end of education I try to shift the needle toward product... whereas in the third week of 1A this is about process and not about product.' – (I6, on C12)

Prior student surveys suggest that students perceive EDD activities as an opportunity to be creative while solving the problem [26], [27], [28]. Pre- and post-event survey evaluations have shown that students' self-efficacy in design, teamwork, and program-specific skills (viz. programming) improved in semesters where EDD events take place [23], [30], though it is not possible to conclude that EDD caused these gains as there were other learning activities present in the term. Some stronger conclusions can be drawn from [26], which analyzed C11, where the Engineering Design Self-Efficacy instrument [36] was used both pre- and post-event. The EDD activity was the only meaningful design instruction for students in that academic term, and analysis of the surveys showed that across the three cohorts that participated in the activity, there was an increase in design self-efficacy, motivation to conduct design, and in students' expectations of success in design after participating in their EDD activity.

C. Curricular Integration

Designing an activity with strong ties to the students' curriculum is the most significant factor that differentiates curricular hackathons from extra-curricular ones, and EDD activities are well embedded with the curricula. Although each of these curricular hackathon events follow the four stages outlined in Section I-C, there are differences in implementations that warrant emphasis. This integration has taken many forms, but has included one or more of the following elements.

1) Scheduling the activity during regular class hours.
2) Intentionally connecting the activity theme/problem or process to material students are expected to learn in their courses.
3) Including a high degree of involvement from teaching staff.
4) Providing structural opportunities for learning and reflection for both students (on the material and content related to the activity) and instructors (on the structure of the activity itself).

In the following sections, we discuss various aspects of each of the four elements of integration.

1) Scheduling: One significant variation between individual EDD implementations relates to the time when the activity is offered and the time elapsed between activity days. It is preferred that students are not expected to participate in mandatory curricular activities outside of regular class times; therefore, integrating a hackathon-like format into the curriculum necessitates finding the time within existing course activities to offer the event. This poses an important constraint for offering curricular hackathons, as existing teaching activities need to be canceled or rescheduled to create the necessary space in the students' timetables.

One way in which EDD events have coped with this constraint is by varying the amount of time between activity days. Although the students are working on the same problem, significant space between EDD activity days contributes to the perceived intensity of curricular hackathons. As seen in Table I, activities can run on consecutive days; have short durations between them or be spread out over the entire semester. There are both practical and pedagogical reasons for structuring the activities in this way. For some, running the activity in consecutive days limits the amount of time the materials need to be present in the IDEAs Clinic space. From a pedagogical perspective, the consecutive EDD structure contributes to the intensity of the experience for all parties involved, with reports of students, instructors, and TAs being very tired by the end of day two. For example, one instructor commented on the intense nature of contiguous EDD activities (and those that occurred during the COVID-19 pandemic):

> '... The versions of the [Software Engineering] activity that I've been involved with, have all been contiguous. That makes it a very intense experience, which might be off putting, particularly this summer with the pandemic and the timing, and the proximity to final exams.' - (I5, on C12)

There are logistical reasons that explain why some activities space their parts by few days or a week between them. The first is a practical consideration due to custom manufacturing of design elements. For example, in the Nanotechnology Engineering activity (C11), students can 3-D print or laser cut elements of their final design. As such, pieces are manufactured outside the scheduled time of the activity and are provided to them when the activity begins on the second day. Another reason that EDD are separated by a short duration is course scheduling; it can be less disruptive to student timetables, and to the courses in the term to have the event take place on the same day of the week in subsequent weeks. There are also pedagogical reasons to have time in between the days of an EDD activity. For example, in C9 (a Mechatronics Engineering EDD), students are required to submit sketches and simple CAD models of their robotic arm design at the end of day 1, on which they receive feedback from the teaching team before they start on day 2. This break between days gives students a chance to slow down their thinking and process what happened on the first day before they enter the build and test phases in day 2, as well as gives the teaching team the time needed to provide early feedback to the students.

2) Course Embedding: There are two important differences between an extra-curricular hackathon and a curricular hackathon. The first is an obvious difference related to the commitment toward the project. In an extra-curricular hackathon, there is usually no penalty for a team withdrawing their design project from the competition. These events are, after all, extracurricular. In the case of curricular hackathons like EDD, students are required to participate to gain the (typically small) amount of credit toward their final grades in the term, which may range from bonus grades in one course (e.g., the Electrical and Computer Engineering activity—C3 [23]), to grades in multiple courses (e.g., the Nanotechnology Engineering activity – C11 [26]). Having grades tied to completing EDD events encourages students to see the project through to the end. Instructors emphasized the importance of a grade incentive to increase student participation:

> 'I do not know how the other IDEAs Clinic activities go but a major challenge that I see in offering any of these activities is if you do not have curricular integration, they [the students] do not do it. If it is voluntary, then you get a very low uptake.' - (I6, on C12)

Presently, EDD activities are most commonly offered in the first or second semester of first year (1A and 1B terms, respectively), or the first semester of second year (2A term). This decision is often made for practical reasons: these terms often have pre-existing design courses which are natural hosts of the activity (e.g., the Mechanical Engineering EDD—C7 and C8 - in the 1B term design course, ME 101); and/or a term may have courses which are more easily integrated (e.g., the programming and introduction to mechatronics courses in the 1A Mechatronics Engineering term—C9 and C10). The timing of the curricular hackathon (in the overall curriculum, the semester, and even day of the week) has significant implications for its effectiveness. EDD events that took place close to major assessments in the term (e.g., midterm or final exams), often had greatly reduced student attendance/attention, even when grades were attached to participation. Identifying an ideal time to hold an EDD often requires extensive cooperation with all the instructors in the chosen academic semester.

In EDD activities, an entire student cohort will be typically presented with a common problem statement, which can then present opportunities for horizontal integration of concepts. In contrast to many hackathons where students select their own problem to solve, by providing the problem statement, EDD activity planners have an enhanced ability to achieve curricular integration of their activity by directly anchoring the problem statement to course learning objectives. For example, in the robotic arm problem for Mechatronics Engineering (C9), students are first asked to create a mathematical model of the arm by solving a simplified inverse kinematics problem. In this way, mathematics concepts and their usefulness in engineering design can be shown to students. In the Nanotechnology Engineering EDD (C11) students design a scanning tunneling

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

microscope that draws on course concepts from past and concurrent courses (e.g., physics, chemical principles, linear circuits, and materials science). Vertical integration is also possible, as in the Software Engineering activity (C12), where concepts such as Djikstra's algorithm are introduced to students in their first semester as part of their EDD activity, even though these concepts are otherwise not taught until second year.

3) Instructional Supports: Further distinguishing curricular hackathons from extra-curricular ones, are the supports in place to assist students as they complete their designs. Extra-curricular hackathons typically have few supports in place to explicitly teach, or assist, participants. Some will include workshops for participants to learn about relevant technology or may have some periods where feedback is given to participants by volunteer mentors, but largely teams are expected to independently work on their projects beginning to end [35]. The EDD curricular hackathons on the other hand, have significant supports to assist students as they progress throughout the activity. These supports can take many forms.

1) *Detailed activity instructions:* For example, the Software Engineering EDD (C12) provides students with an extensive manual to explain the software tools they are expected to use in the activity.
2) *Informational presentations:* For example, there are several short presentations spread throughout the Nanotechnology Engineering EDD (C11) to give students important information as they need it.
3) *A structured design process:* With checkpoints with the teaching team (like the 5-stage process outlined in Section I for EDD events).
4) *Formal checkpoints with members of the teaching team:* For example, partway through the first day, students in the Electrical and Computer Engineering EDD (C3 and C4) have their circuit diagrams checked by members of the teaching team before they proceed to construct their prototype.
5) *Informal peer review sessions:* For example, in the Mechatronics Engineering EDD (C10), five groups of four students each met at multiple times during the design window to provide constructive feedback to each other.
6) *Constant presence of members of the teaching team:* Which increases the potential for student learning by reducing frustration. Students can seek information and problem-solving advice from the teaching team, rather than spending hours searching for it on their own.

These sorts of intellectual supports are crucial for avoiding student frustration when the complexity of the task is high. Having a strong teaching team presence in the space where the EDD event is taking place means that the bar to ask for help is low and provides opportunity for the teaching team to interact informally with students, further strengthening the community building potential of the event.

4) Learning and Reflection: One final difference between the EDD curricular hackathons and a typical extra-curricular hackathon is related to the *Test and Reflect* phase that is integrated in each EDD activity. In an extra-curricular hackathon, final designs are presented or pitched to a panel of judges in the form of a product pitch, assessed on a variety of criteria, and awarded prizes where appropriate. However, the final presented designs may not always be functional, but are described in enough detail that the intended functionality is conveyed to the judges [35], and these designs are still eligible for prizes. In a similar way, the final products of the EDD activities are presented and usually tested in front of the teaching team and fellow students. However, there is limited emphasis on "pitching" the idea; rather, students must explain how they proceeded through the design process. In the case where designs do not work (which happens frequently in EDD events) students can reflect on what did not work and explain what they would do differently in the future. The "closed loop" of design-build-test-*reflect* is a significant difference between the curricular hackathon approach that is presented here, and the approach used in traditional extra-curricular hackathons.

Curricular hackathons can also be a meaningful learning experience for the teaching team offering the activity. EDD events at various times have been both curricular (with grades in one or more courses), and optional for students (but still maintaining ties to the knowledge students are learning in their courses); and while the curricular events tend to have greater student turnout, the same activity can be used both inside and outside the classroom. For the activity designers, running the activity as an extra-curricular pilot can be a very useful feature in refining the activity before it becomes a formal part of a course. It can also be a useful opportunity for the teaching team to gain comfort with supporting students in an open-ended design activity. There is no doubt that the environment of a curricular hackathon is a very different setting than a traditional lab, lecture, or tutorial—it is much more akin to supporting a studio experience than more traditional engineering teaching methods—and it can be challenging for instructors the first time through the activity as they adjust to the role of coach or mentor.

V. Discussion

A. Contributing Definition of Curricular Hackathons

The examples explored above contribute to a growing body of literature and understanding of how hackathons and the hackathon format can be used in curricular settings for the purposes of design teaching and learning. As this pedagogical approach is further explored, it is useful to define the concept of a curricular hackathon. Uys [19] is one of the rare cases in the literature that uses this term, however, they do so in the context of their very specific implementation, in which the hackathon is only used as a structured sprint to finish the last steps in a semester-long project. In addition, the discussion related to how students are supported throughout the design process provided in Uys [19] is limited, which is expected to be a key component of the curricular hackathon.

As such, this article contributes its own definition of the curricular hackathon format that highlights what are believed to be three core aspects.

1) It is a short, high intensity social experience.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

2) It guides students through the design-build-test cycle of an engineering design problem, and includes opportunities for reflection.
3) It achieves some level of integration and/or embedding in a program's curriculum.

This definition captures the defining features of extra-curricular hackathons from the literature, while emphasizing design *learning* and explicit connections to the curriculum.

B. Implications for Design Teaching and Learning

Curricular hackathons are an emerging pedagogy which provides unique opportunities, many of which are not afforded by traditional classroom experiences. There are numerous advantages to the curricular hackathon format for improving student design skills. Ericsson [37] described the need for repetition and practice when developing cognitive skills. For a novice designer who is new to the domain, which fits the profile of a first-year engineering student, curricular hackathons can be a place where they are able to practice designing in their respective domain, surrounded by their peers solving the same problem. For students with slightly more experience, the open-endedness of the curricular hackathon format can also provide elements of deliberate practice [38]—a necessary form of training to move toward expert designing. The challenge of the event combined with the support provided by the teaching team (in the form of coaching, or cognitive supports like activity scaffolding) provide undergraduate students with a wide range of prior design skills a meaningful learning experience.

The short duration, high intensity, and significant social experience of a curricular hackathon presents its own advantages. This environment is fertile ground for developing student self-efficacy in design. To provide a mastery experience as described by Bandura [4], the design problem needs to be sufficiently challenging that students feel an accomplishment from completing it, with supports in place to assist any students who grow frustrated during the event. The social experience of the curricular hackathon provides a natural environment for meaningful social modeling to take place as an entire cohort of students are all solving the same problem at the same time in the same space. For students, seeing an entire classroom of their peers overcome a challenging problem is a very powerful experience both for self-efficacy development, but also for broadening perspectives and observing the different paths that their peers took to reach their solutions. The broad diversity of solutions is a powerful demonstration that there are multiple solution paths to a design problem—an important realization for developing their personal epistemology [39], [40]. Curricular hackathons are much better suited for social modeling than extra-curricular ones as there is no guarantee that other participants in traditional hackathons are your peers (they could be different ages, from different programs, or even different institutions) and participants do not see the process that other groups went through, nor do they necessarily appreciate the challenges those other participants had to overcome during the event.

An important feature of curricular hackathons is the requirement that students participate in a full design-build-test sequence in a formative setting. Clearly course projects present a similar opportunity for students, however, they are generally summative in nature. Summative experiences like capstone design projects can be terrific learning environments, and are often required for program accreditation, however, they will be treated differently by students as the consequences for weak performance are much higher. The formative experience of curricular hackathons gives students a chance to fail, with limited academic consequences, and to learn from that failure. Curricular hackathons then, are not replacements for course projects (or for other learning environments like labs), they are complementary to those pedagogies.

Curricular hackathons are excellent vehicles for integrating curricular concepts, for providing students an opportunity to practice applying theory to real-world problems, and for reinforcing other design instruction. Traditional teaching and learning methods like course projects and labs generally only focus on applications of content from a single course in the curriculum. Far transfer and integration of concepts is a distinct and challenging cognitive task [1], and so students should be given opportunities to practice these tasks. Integrating curricular hackathons with the broader curriculum enables instructors to prepare students for the activity in advance, include formal student reflection on learning, and hold follow-up discussions afterwards; all features not present in extra-curricular hackathons. These integrations can improve learning and help students understand their experience better. In addition, the social connections forged between the instructor and the class can reinforce self-efficacy gains by engaging in social persuasion and assisting students with regulating their emotional states.

C. Limitations on Generalizability

This study used a multiple case research design, using a synthesis of prior publications and an interview study to analyze features of curricular hackathons across 12 unique EDD cases. The 12 cases encompassed a large number and variety of EDD implementations: 11 different engineering programs, with activities spanning over six years, offered to thousands of students. While this research design can increase the robustness of results [22], there are still limitations to the generalizability of the findings presented here. Even though there was a large breadth of EDD activities included, ultimately, they came from a single institution, and only included offerings in the first three academic terms. It is difficult to say what modifications would be needed to maximize the impact of curricular hackathons in the later parts of the undergraduate curriculum. These students would have much stronger engineering foundations, and may have experienced design in settings outside of the classroom (like during co-operative work terms, as is the requirement at Waterloo), and so adaptations to the format should be expected. For example, the warmup and design sections of the EDD process would likely include more rigorous analysis of the problem and/or proposed solution, as the students would have the engineering science background to conduct this work. The tools used to build the designs may also need to change to better match the greater experience in the student participants. Further efforts are needed to implement this concept in these later terms, and careful evaluation is required to ensure maximum impact.

VI. Conclusion

The hackathon structure is being recognized by engineering educators as a novel approach to teach engineering design [16]. This article has presented an advancement in the understanding of curricular hackathons as a way to engage students and provide meaningful practice of design skills. Using a multiple case study research approach that combined interview data with staff and faculty who developed and implemented EDDs, and a synthesis of past published research on the format, a new definition of curricular hackathons is provided, which includes social, technical, and reflective elements for both students and instructors. Based on this synthesis, curricular hackathons are described as short, high intensity social experiences that guide students through the design process and facilitate student reflection on their design learning, in a curriculum-embedded context.

References

[1] S. Ambrose, M. W. Bridges, M. DiPietro, M. C. Lovett, M. K. Norman, and R. E. Mayer, *How Learning Works: Seven Research-Based Principles for Smart Teaching*. San Francisco, CA, USA: Wiley, 2010, pp. 103–120.

[2] D. Jonassen, *Learning to Solve Problems: A Handbook for Designing Problem-Solving Learning Environments*. New York, NY, USA: Routledge, 2011.

[3] E. Dringenberg and Ş. Purzer, "Experiences of first-year engineering students working on ill-structured problems in teams," *J. Eng. Educ.*, vol. 107, no. 3, pp. 442–467, 2018.

[4] A. Bandura, "On the functional properties of perceived self-efficacy revisited," *J. Manag.*, vol. 38, no. 1, pp. 9–44, 2012.

[5] P. Ostafichuk, E. Croft, S. Green, G. Schajer, and S. Rogak, "Analysis of mech 2: An award-winning second-year mechanical engineering curriculum," in *Proc. Int. Conf. Innov. Good Pract. Res. Eng. Educ.*, 2008, pp. 1–12.

[6] A. A. Maciejewski et al., "A holistic approach to transforming undergraduate electrical engineering education," *IEEE Access*, vol. 5, pp. 8148–8161, 2017.

[7] J. Membrillo-Hernández, M. J. Ramírez-Cadena, M. Martínez-Acosta, E. Cruz-Gómez, E. Muñoz-Díaz, and H. Elizalde, "Challenge based learning: The importance of world-leading companies as training partners," *Int. J. Interact. Des. Manuf.*, vol. 13, pp. 1103–1113, May 2019.

[8] M. Somerville et al., "The Olin curriculum: Thinking toward the future," *IEEE Trans. Educ.*, vol. 48, no. 1, pp. 198–205, Feb. 2005.

[9] G. Kuh, *High-Impact Educational Practices: What They Are, Who Has Access to Them, and Why They Matter*, Washington, DC, USA: AAC&U, 2008.

[10] G. Briscoe and C. Mulligan, *Digital Innovation: The Hackathon Phenomenon*, Creative Works London, London, U.K., 2014.

[11] M. Flus and A. Hurst, "Design at hackathons: New opportunities for design research," *Des. Sci.*, vol. 7, pp. 1–24, Feb. 2021.

[12] M. A. M. Angarita and A. Nolte, "What do we know about hackathon outcomes and how to support them?—A systematic literature review," in *Collaboration Technologies and Social Computing*. Cham, Switzerland: Springer, 2020, pp. 50–64. [Online]. Available: https://doi.org/10.1007/978-3-030-58157-2_4

[13] "StarterHacks." Accessed: Aug. 5, 2022. [Online]. Available: https://starterhacks-2020.devpost.com

[14] *The 'Double Diamond' Design Process*, Des. Council, London, U.K., 2005.

[15] M. Flus and A. Hurst, "Experiences of design at hackathons: Initial findings from an interview study," in *Proc. Des. Soc. Int. Conf. Des. (ICED)*, Gothenburg, Sweden, 2021, pp. 1461–1470.

[16] M. Flus and A. Hurst, "Hackathons as a novel design pedagogy in engineering education," *Int. J. Eng. Educ.*, vol. 38, no. 1, pp. 36–44, 2022.

[17] M. Calco and A. Veeck, "The Markathon: Adapting the hackathon model for an introductory marketing class project," *Marketing Educ. Rev.*, vol. 25, no. 1, pp. 33–38, 2015.

[18] J. A. Artiles and K. E. LeVine, "Ta-Da! you're a design thinker! validating the DesignShop as a model for teaching design thinking to non-designers and achieving systemic re-design in the education system," presented at the ASEE Annu. Conf. Expo., Seattle, WA, USA, 2015.

[19] W. F. Uys, "Hackathons as a formal teaching approach in information systems capstone courses," in *Proc. Annu. Conf. Southern African Comput. Lecturers' Assoc.*, 2020, pp. 79–95.

[20] K. Gama, B. Alencar, F. Calegario, A. Neves, and P. Alessio, "A hackathon methodology for undergraduate course projects," in *Proc. Front. Educ. Conf. (FIE)*, 2018, pp. 1–9.

[21] J. Porras et al., "Hackathons in software engineering education—Lessons learned from a decade of events," in *Proc. 2nd Int. Workshop Softw. Eng. Educ. Millennials*, New York, NY, USA, 2018, pp. 40–47.

[22] S. Hunziker and M. Blankenagel, "Multiple case research design," in *Research Design in Business and Management*. Wiesbaden, Germany: Springer, 2021, pp. 171–186. [Online]. Available: https://doi.org/10.1007/978-3-658-34357-6_9

[23] C. Rennick, C. Hulls, D. Wright, A. J. B. Milne, and S. Bedi, "Engineering design days: Engaging students with authentic problem-solving in an academic hackathon," presented at the ASEE Annu. Conf. Expo. Salt Lake City, UT, USA, 2018.

[24] K. N. McKay, S. Mohamed, and L. Stacey, "Concepts Only Please! Innovating a First Year Engineering Course," in *Proc. CEEA-ACEG Annu. Conf.*, Halifax, NS, Canada, 2016, pp. 1–7.

[25] C. Rennick, D. Rayside, J. Harris, and P. Lam, "Software engineering days: Using a video game platform to teach collaborative software development," in *Proc. CEEA-ACEG Annu. Conf.*, Ottawa, ON, USA, 2019, pp. 1–8.

[26] J. A. Coggan and C. Rennick, "Development and implementation of an integrative and experiential design project: Design, build and test a scanning tunneling microscope," *Int. J. Eng. Educ.*, vol. 38, no. 1, pp. 25–35, 2021.

[27] S. J. Phillips, K. Giesinger, R. Al-Hammoud, S. Walbridge, and C. Carroll, "Enhancing Student learning by providing a failure risk-free environment and experiential learning opportunities," in *Proc. ASEE Annu. Conf. Expo.*, Salt Lake City, UT, USA, 2018, pp. 1–13.

[28] B. MacVicar, C. Clow, C. Muirhead, R. Al-Hammoud, and J. Craig, "Design, construction, and destruction in the classroom: Experiential learning with Earthen dams," *J. Hydraulic Eng.*, vol. 146, no. 6, pp. 1–10, 2020. [Online]. Available: https://doi.org/10.1061/(ASCE)HY.1943-7900.0001745

[29] R. Mui, S. J. Woo, S. Arbuckle, R. Al-Hammoud, and S. Walbridge, "Architectural engineering starts with design from day 1," in *Proc. ASEE Annu. Conf. Expo.*, Tampa, FL, USA, 2019, pp. 1–12.

[30] C. Rennick, C. C. Hulls, and A. Gryguc, "Assessing the impact of transitioning introductory design instruction to an online environment," in *Proc. ASEE Annu. Conf.*, 2021, pp. 1–21.

[31] I. Ivkovic, T. L. Willett, M. J. Borland, and M. Gorbet, "Design days boot camp: Enhancing student motivation to start thinking in engineering design terms in the first year," in *Proc. CEEA Annu. Conf.*, Toronto, ON, Canada, 2017, pp. 1–7.

[32] A. Hurst, G. Litster, and C. Rennick, "Operationalizing Jonassen's design theory of problem solving: An instrument to characterize educational design activities," in *Proc. ASEE Annu. Conf. Expo.*, 2020, pp. 1–10.

[33] C. Rennick, G. Litster, A. Hurst, C. Hulls, and S. Bedi, "Characterizing engineering design activities using Jonassen's design theory of problem solving," *Int. J. Eng. Educ.*, to be published.

[34] S. Elo and H. Kyngas, "The qualitative content analysis process," *J. Adv. Nurs.*, vol 62, no. 1, pp. 107–115, 2008.

[35] M. Flus, *An Exploratory Study of Experiences of Design at Hackathons*, Univ. Waterloo, Waterloo, ON, USA, 2021.

[36] A. R. Carberry, H.-S. Lee, and M. W. Ohland, "Measuring engineering design self-efficacy," *J. Eng. Educ.*, vol. 99, no. 1, pp. 71–79, 2010.

[37] K. A. Ericcson, "The acquisition of expert performance as problem solving," in *The Psychology of Problem Solving*, J. E. Davidson and R. J. Sternberg, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2003, pp. 31–86.

[38] K. A. Ericsson, "Experience, practice, and deliberate practice," in *The Cambridge Handbook of Expertise and Expert Performance*, 2nd ed., K. A. Ericsson, R. R. Hoffman, A. Kozbelt, and A. M. Williams, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2018, pp. 745–769.

[39] W. G. Perry, *Forms of Intellectual and Ethical Development in the College Years: A Scheme*, San Francisco, CA, USA: Jossey-Bass, 1970.

[40] R. M. Felder and R. Brent, "The intellectual development of science and engineering students. Part 1: Models and challenges," *J. Eng. Educ.*, vol. 93, no. 4, pp. 269–277, 2004.

Christopher Rennick (Member, IEEE) received the B.A.Sc. and M.A.Sc. degrees in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 2007 and 2009, respectively, where he is currently pursuing the Ph.D. degree in students' design skill development.

He is the Engineering Educational Developer with the Engineering IDEAs Clinic, where he designs and implements real world, hands-on design activities for undergraduate engineering students.

Greg Litster received the Bachelor of Knowledge Integration and the M.A.Sc. degree in management sciences (Joint Honour Mathematics) from the University of Waterloo, Waterloo, ON, Canada. He is currently pursuing the Ph.D. degree in engineering education with the University of Toronto, Toronto, ON, Canada.

His research interest is in engineering education, specifically in the teaching and learning of design and problem solving, collaboration, and teamwork.

C. C. W. Hulls (Member, IEEE) received the B.A.Sc., M.A.Sc., and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada.

She is the Associate Chair, Teaching with the Mechanical and Mechatronics Engineering Department, and has been teaching courses in programming and digital logic since 1999.

Dr. Hulls has developed a variety of innovative techniques and in 2016, received the Brightspace Innovation Award in Teaching and Learning. She is a P.Engineer with the University of Waterloo. In 2022, she became a Fellow of the Canadian Engineering Education Association.

Ada Hurst received the bachelor's degree in electrical engineering, and the master's and Doctoral degrees in management sciences from the University of Waterloo, Waterloo, ON, Canada.

She is a Lecturer with the Department of Management Science and Engineering, University of Waterloo. Her research falls in the areas of design cognition, and design teaching and learning. She is interested in innovations in engineering design pedagogy and regularly teaches engineering capstone design courses.