Transforming the Client Relationship to Support Large Capstone Classes

Bowen Hui, Samantha Hodge, Dilpreet Samra
Department of Computer Science, Mathematics, Physics, and Statistics
University of British Columbia
Kelowna, Canada
bowen.hui@ubc.ca, {zinnia37,dsamra}@student.ubc.ca

***Abstract*—This innovative practice full paper describes a case study from a software engineering capstone project course. Undergraduate programs often have a final-year capstone course designed to integrate and apply the knowledge and skills students have previously acquired while adapting to industry-standard practices. Capstones play a critical role in bridging the gap between academia and the industry as students transition to the workforce. Over the past twelve years, our institution has adopted a client-based model where industry clients work closely with a single team to solve a real-world problem. However, rising enrollment has put a strain on running this model effectively because of difficulties in recruiting clients, managing numerous client relationships simultaneously, and keeping client-student interactions sustainable. To tackle these challenges, we propose a new client model where clients pitch their ideas as themes in a competition and act as panel judges in evaluating student team submissions. We call this the *hackathon client model* and evaluate it in a class with 22 teams and 104 students. Through a thematic analysis of the qualitative responses gathered from this study, our findings suggest this new model provides a scalable alternative to operating a large capstone class while preserving many of the benefits of the traditional client model. However, both students and clients indicate having more means of communication would improve the project requirements phase and strengthen their relationship. We discuss ideas on improving the hackathon client model and plans for future experimentation in large capstones.**

***Index Terms*—Computer science capstone, client interaction, client model, large class sizes**

I. Introduction

Capstone courses, often occurring near the end of a program of study, are designed to integrate and apply the knowledge and skills students have acquired throughout their academic journey. Common approaches to teaching and running capstone courses include project-based learning, interdisciplinary collaboration, and real-world problem-solving [8]. Over the past twelve years, our institution has adopted a client-based model for running capstone projects in computer science to give students a real-world problem context. Students work in teams to tackle challenges faced by external clients, such as developing a minimum viable product for local entrepreneurs, adding new functionality to an existing product for small business owners, and building low-maintenance software for non-profit organizations. This experience provides students with a relevant application context while pooling the skills they gained from their degree and applying them to accomplish a year-long project. Key challenges include managing expectations and communication with clients, adapting to changes in the project scope, and ensuring a balance between meeting client needs and academic objectives. The growth in our student population has further introduced additional obstacles such as an inability to recruit enough external clients and not providing the necessary support to simultaneously manage the many client-team relationships throughout the capstone project. It is questionable whether a client-based model is sustainable in large capstone classes.

To tackle these challenges, we present a new client-based capstone course modeled after a hackathon competition. Rather than maintaining a close client relationship throughout the project where teams and clients meet weekly, we ask clients to propose their project ideas as themes in the competition and act as panel judges in evaluating team projects in two milestones. We implemented this hackathon client model in a software engineering capstone course with 104 students grouped into 22 teams. We explain the pros and cons of this approach from a theoretical lens to support students in teamwork and a practical perspective in administering and managing the projects. In particular, we evaluate this approach with respect to the criteria necessary for running successful capstone courses, such as forming effective student teams, monitoring student team dynamics, managing project scope, ensuring assessment consistency, managing logistical constraints, encouraging high engagement levels, managing client expectations, and scaling to large class sizes. Our specific research questions are:

RQ1: What are the administrative advantages and disadvantages of the hackathon client model?
RQ2: What are the students' perceptions of the hackathon client model?
RQ3: What are the clients' perceptions of the hackathon client model?

The overarching goal of this work is to present an alternative curriculum design to successfully operate software engineering capstone courses in the face of increasing student enrollment and university budget constraints. The benefits gained from this approach include minimal client oversight, increased student motivation and project ownership, more stable requirements, and increased focus on teamwork. However, students and clients mutually wanted more opportunities to

communicate with each other, as they both believed increased interaction would foster better mentoring and higher ability to meet client expectations. In light of these findings, we discuss teaching and learning implications for future consideration. We plan to continue experimenting with this approach to better understand its ability to scale to even larger class sizes.

II. Related Work

We conducted a literature review using combinations of the following search terms: "computer science", "capstone", "client", and "assessment". The resulting papers described capstones in Computer Science or Computer Engineering save for one from Engineering Technology [5] and one from Data Science and Machine Learning [3]. These papers reported variations in how the capstones were run. A survey from 2011 reviewed 200 papers that described capstones varied from 1 to 8 semesters [8]. From 2014 onwards, some of the papers described running for one semester [1] [14] [4] and others for two [17] [16] [5] [12]. The majority focused on undergraduate capstones [7] [1] [11] [17] [10] [13] [8], though some included graduate courses [15] [9], and one had both undergraduate and graduate students [3]. A few involved individual projects [7] [11] [14], though most were team projects [7] [1] [2] [3] [15] [17] [4] [16] [9] [5] [12] [10]. Class size varied from less than 10 [8] to 94 students [9] with the majority not specifying the number of students in the class. The instructive and evaluative resources from each capstone largely manifested as a set of instructors usually accompanied by and composing of a committee [7] [1] [11] with individual professors mentoring and observing one or more teams. Some roles akin to student teaching assistants were mentioned briefly in only two papers but their responsibilities were unclear [4] [12].

Many papers focused on grading and rubrics for capstones [7] [1] [15] [17] [9] [5] but none honed in on the interactions between the students and clients or the instructor and clients. Some capstones did not specify a client model [17] [9] or appeared to not have a client at all [1]. Only three papers commented on the potential disadvantages of a client model [3] [13] [8]. For example, some students found it difficult to handle scope creep especially if the client feedback was used in the student evaluation [8]. There were also cases reporting teams that struggled to meet client expectations [12]. The rest of this section reviews the papers that specify a client model.

A. Industry Clients

The 2011 literature survey [8] defined industry projects as ones with a business problem from "a real client that intended to use the project when it was completed." This type of project was noted to have benefits on student motivation and practical experience especially if the students worked within a company environment, with the potential for additional technical support [8]. Occasionally, company-sponsored projects required a legal agreement to govern how shared data and proprietary information was managed. This could include students signing away intellectual property rights as in [3] and added an extra layer of complexity to navigate with a corporate client. The client interaction reported involved weekly meetings where students presented their work and the clients provided feedback and requested revisions as needed [5] [14]. Our experience with industry clients also aligns with these findings. One additional complication we observed was that some clients had the false expectation that our students were working on their projects full-time when in reality they were simultaneously taking four other senior undergraduate courses.

B. Community Service Client Models

A community service project works with non-profit and government organizations. The majority of papers that fall under this category involved university clients [16] [8]. In particular, the authors of the 2011 survey mentioned they vet projects to deliberately avoid research projects and projects where the course instructor act as the client. The vetting process also ensured the project was not critical to the organization because it would put too much pressure on students. Additionally, our experience with non-profit organizations revealed their inability to sustain technical projects after the capstone, which starkly contrasts the additional support and resources of industry projects. Furthermore, our experience with university professors as clients showed a lower level of professionalism exhibited by the students, possibly because they were already comfortable with these professors from previous classes.

C. Internal Clients

This category includes project topics that are generated by a student or the course instructor. Frequently, these take the form of research projects akin to an undergraduate Honors thesis. One study reported that students brainstorm topics with the instructor [2]. Once a topic is chosen, students continue to develop the project requirements without any interaction with an external party. Although very few papers reported using internal clients, student-led projects are likely simpler in terms of scope and time management. We also suspect having the instructor act as the client introduces a conflict of interest due to their two roles.

D. Mixed Client Models

The last category involves projects that use any combination of the aforementioned client models within the same class. Although not all the papers provided enough details (e.g., [11] [13]), we summarize the unique operational experiences pertaining to the client relationship from these papers.

This approach solicits projects from local industry and community organizations as well as university professors. Rather than matching teams to projects, one paper mentioned using a sponsor fair where clients presented their ideas for student teams to choose from [12]. Two papers also reported that students typically preferred industry client projects over other types of clients [7] [3].

The frequency of meetings between student teams and clients varied. At a minimum, we found teams were asked to meet with clients 3 times for requirements elicitation,

prototype presentation, and final project presentation, though some clients may have chosen not to attend the prototype presentation [15]. Others reported that client meetings took place biweekly [3] or weekly [12]. Communication outside of these meetings could also be structured through a dedicated team member assigned to a client liaison role [15].

One of the papers reported difficulty managing the project scope [3]. These authors noted that clients often needed guidance in framing their projects in a manner appropriate for the capstone so that the scope was neither too restrictive nor too open-ended. As a result, they recommended the instructor regularly check in with clients throughout the project to manage unexpected problems. In contrast, our capstone solicits projects from clients where the instructor and the clients co-develop a general scope feasible for the capstone at the start, and then the instructor works with students to help manage the potentially changing requirements throughout the course without the instructor contacting the client.

Two papers mentioned the use of client feedback forms to help improve future course offerings [7] [3]. One case reported that the feedback responses were "overly generous" and follow-up conversations revealed "a more nuanced, and possibly more realistic assessment of outcomes" [3].

One unique setup was reported where students worked in the client's office and had direct contact with the clients and related project resources [4]. This arrangement resembled internship experiences where students became accustomed to office environments and had access to the on-site client and a mentor at weekly meetings. In this case, emphasis was placed on understanding client motivation with instructors acting as mediators between students and clients when critical issues crop up. This format is also the only capstone from our review that explicitly included clients in the student evaluation rubric.

III. Course Context and Redesign

In our four-year undergraduate computer science program, we run a final-year software engineering capstone course that spans two semesters. During the winter session, this course runs between September and April, giving students 26 calendar weeks outside of the midterm breaks and final exams to work on the course. In the first few weeks of the course, students typically focus on course logistics, teamwork, establishing a common collaboration process, and writing a project charter that defines the requirements. By the end of October, the teams will have developed design mocks, set up their technical stack, designed some database relationships, and started implementing their software solution. Generally, the teams are expected to have completed a minimal working prototype by the end of the fall semester so they can focus on adding features in the winter semester. During the winter semester, students work on extending the prototype, gathering peer feedback on their systems, and fixing bugs. By the end of the project, each team has a fully developed project to showcase. In some cases, some projects have known bugs that are documented in the handover document. Many projects also get deployed publicly, such as websites and mobile apps.

The course has three overarching components for evaluating student work: a team component, an individual component, and a client component. The details of each component have been refined over the years. Students are required to pass all components in order to pass the course.

A. Previous Client Models

When class sizes were small (i.e., less than 40 students), students formed teams with 3 to 5 students and worked closely with a dedicated industry client for 8 months. Students were matched with a team and a project based on the best collection of student skills to meet all the project requirements while also considering students' project preferences. Most of our industry clients were not well-versed in technology, so students practiced communicating project details in an understandable way. This close client relationship often resulted in requirements changing throughout the development lifecycle. In rare cases, clients placed unreasonable demands on the students or did not provide them with the key resources needed for timely completion. These cases usually required heavy instructor intervention to resolve. On the other hand, positive outcomes include cases where students worked alongside industry programmers in a real-world work environment, received reference letters for future job applications, and accepted job offers from client companies directly after graduation. Under this model, clients also had to provide a midterm and final review of the team based on an instructor-facilitated questionnaire. The client feedback served as an external validation of the projects and an assessment of the professional conduct of the students.

Since 2020, our class size increased to 85+ students. Without additional instructors, the course could not sustain the one-to-one relationship that student teams had with clients. At this time, we experimented with matching three student teams to one client in hopes of maintaining the same learning outcomes as before. Unfortunately, this setup introduced additional complications where some clients tripled the time they spent with students, many clients mixed up the teams and, thus, could not fairly evaluate them, not all the team members could attend weekly client meetings, and only the least technical member attended client meetings but subsequently miscommunicated and misrepresented the technical details of the project.

To ensure the success of external client projects, student teams were formed primarily by ensuring each team had a strong student. This indicator was approximated by a combination of the technical skills of the students demonstrated in previous courses and self-reported responses on specific skills needed to meet project requirements. The remaining members were matched to form a balanced team. We also tried to include a friend on each team to provide a supportive working environment for all the students.

B. The Hackathon Client Model

In 2023, we experimented with the new hackathon client model with over a hundred students in the class. Students formed teams based on friendships, elicited technical skill levels, and mutual goal orientation. Each team chose a

project from a list of client-provided project themes. These themes outlined the overall objectives and constraints of the projects, providing teams with a high-level framework while allowing significant autonomy in implementation details. The hackathon-style format required minimal interaction between clients and students, shifting the focus toward student-led exploration and problem-solving. Teams relied on the provided themes to develop their project plans, which gave students considerable freedom in determining the scope and direction of their work.

The clients served primarily as evaluators rather than supervisors or collaborators who worked with the teams every week. Clients interacted directly with student teams on only two formal occasions during the semester through judging panels. The first panel assessed the teams' initial project progress, providing feedback and rankings that helped guide students toward improvements and goal alignment. The second panel, held at the end of the semester, was a competition that evaluated the final projects and offered insights into project strengths and areas for improvement. This competitive structure fostered an environment where students could gauge their progress relative to others while adapting their projects based on the panel's feedback.

IV. Method

This study is part of an end-of-course survey from the capstone course. The purpose of this survey is to identify what worked well with the new hackathon client model and which aspects need improvement in future course offerings.

A. Participants

The class had 104 students enrolled in the capstone course. Among them, there were 85 males, 15 females, 1 non-binary, and 3 either preferred not to say or data not provided. Also, 63 out of 86 are racial minorities. Of the 104 students in the class, 86 students responded to the survey.

A total of 4 clients participated in this capstone, but only 3 of them answered the survey. The respondents were 1 female and 2 males, and 2 out of 3 were racial minorities.

B. Procedure

At the end of the course, students completed an experience survey designed to capture their perspectives on the hackathon-style format and overall capstone experience. The survey allowed students to express their satisfaction or concerns regarding project selection, client interaction, and perceived educational value. Similarly, clients provided feedback through a structured survey, commenting on their interactions with the student teams and suggesting potential improvements for future iterations of the capstone course.

To analyze the open-text responses from these surveys, we conducted a thematic analysis and developed codes discovered in the responses [6]. The responses were segmented by sentences and further by phrases if a conjunction was used to separate out multiple ideas. Specifically, two raters independently analyzed all of the qualitative responses from these surveys following the codebook established from the initial analysis. In total, the raters completed three passes to reach an acceptable intercoder reliability level of $\alpha = 0.87$ for the student data and two passes to obtain $\alpha = 0.81$ for the client data. The final codebook is provided in Table I.

C. Materials

1) Student Survey Questions: The student survey had three main questions. The first question asked students whether they preferred the hackathon format used in the capstone over a more direct relationship with the client where they meet weekly throughout the year. The second question sought to understand how the students would have preferred to interact with their clients differently, if at all, during the course. The third question probed students' opinions on the scalability of the hackathon format, specifically if they believed it would work for a larger class size, as enrollment is expected to double. Students were asked to elaborate on their responses.

2) Client Survey Questions: Feedback from the clients was gathered through twelve questions. First, we asked clients to reflect on whether their goals in participating in the capstone were achieved and, if not, provide suggestions on how those goals could be better supported. Next, we asked clients to indicate the number of teams that submitted work to their project theme and to express whether the time dedicated to evaluating their projects was sufficient or too much. Although interaction between the students and clients was not expected, some clients received questions from students. Thus, we also asked about the amount of time clients spent interacting with students. The fifth question evaluated whether clients felt more interaction with students would be desirable. The sixth question asked clients their preferred approach to student interaction in future capstone courses. The next questions asked whether clients would consider hiring students who submitted projects in their theme or writing a positive reference letter for them. Lastly, we asked clients to express what they liked, what they wanted to see improve, and any other comments they may have about the overall capstone experience.

D. Results: Student Feedback

Figure 1 shows the counts from our coding analysis from both the student and client data. A list of unexpected benefits is also provided in Table I.

1) Communication, Freedom, Requirements, Relevance: Overall, students enjoyed the opportunity to work on meaningful, real-world problems. They also expressed a desire for the projects to align with industry standards, although they find it challenging to balance industry expectations with academic demands. Some students even expressed the desire to work as part of a company project similar to what was reported in the literature [4]. Many students seemed to value the creative freedom the hackathon model affords, allowing them to explore innovative solutions. The freedom of choice is also closely tied to motivation where students could choose to work on projects that were personally interesting to them. The literature states a sense of project ownership can provide significant motivation

TABLE I
CODEBOOK USED IN THE ANALYSIS OF STUDENT AND CLIENT DATA, SHOWN WITH CODING FREQUENCIES.

Code	Student Counts	Client Counts	Explanation and Examples
Communication	96	21	Anything related to communication between the client and students while excluding information about project requirements **Student:** "Yes, I would have preferred to have regular meetings with the client." **Client:** "Might be nice to have more checkpoints that are casual, perhaps every month."
Freedom	83	7	Options for the available projects (too much, too little, just enough) **Student:** "I enjoyed the format and found that there was more freedom when it came to what features you wanted to implement." **Student:** "I would have enjoyed the concept of coming up with our own project idea." **Client:** "The students could also be polled on which types of projects they'd prefer and in which areas (although maybe gathering clients for this might be harder)."
Requirements Challenges	71	3	Understanding of the project requirements and client expectations **Student:** "It often felt like the vision and requirements of the professor and TA's were misaligned with what the client was asking." **Client:** "It's also a place where they can ask any new questions about client requirements that might arise during development."
Generic	70	8	Anything unrelated to the hackathon format or not making any particular statement. **Student:** "I would probably get a straight answer if I have done both things at least once." **Client:** "Develop an MVP of a product."
Scalability	60	1	Scalability of the hackathon model **Student:** "Despite my opinions on this format, it may be difficult to have several groups being able to meet with clients regularly, especially if there aren't many clients." **Client:** "An even split for the projects to students."
Industry Relevance	49	11	Capstone as representative of the industry and its real-world experience and benefits **Student:** "If I had to choose one, I would have chosen creating a real product for a real client with real stakes and real consequences." **Client:** "Nurturing students' potentials with projects they enjoy."
Feedback	46	5	Anything relating to feedback, opinions, or trajectory on project work already done **Student:** "Moreover, I think a big part of Software Engineering, in general, is to able to meet with clients regularly for updated requirements and validation that the project is being made to their standards." **Client:** "Future meetings could be demo of progress and Q/A to make sure the product is following based on the client's expectations."
Stress	46	3	Anything related to stress or lack thereof and therefore how busy the students were **Student:** "I heard from peers from prior years that working directly with individual clients can be very invasive and challenging, especially with other coursework to do." **Client:** "Not the case currently, but in previous years, a few clients were quite dependent on student teams to deliver a working product that they would use, and I feel that adds undue pressure on some students."
Support	37	3	Desire for more support in ways unrelated to client communication and project requirements. **Student:** "Since they are in the industry, they would know things we as students don't (or sometimes can't even think of), and working with them would have been a great learning experience, beyond just computer science." **Client:** "I know this is a constraint from the dept but it's difficult to manage all the teams as 1 instructor."
Motivation	32	5	Reasons to strive for a good end product **Student:** "The Hackathon format created a competitive environment which inspired the team to put in work and do their best." **Client:** "Saying that the hackathon idea and creating it as a competition noticeably pushed the students to get more done than I have ever seen in other capstone projects."
Panel Format	29	6	Comments on the judging format **Student:** "I was in option 4 (web-based game), so I found the inclusion of a mock client for the panel was great and found the feedback provided to be very good." **Client:** "Serving as a judge twice is fine, but it felt like I was a bit more disconnected with the students and perhaps some issues could have been avoided with more regular check-ins."
Grading	19	1	Anything related to how marks are given **Student:** "Having such encouraging marks from the prof/TAs at the beginning of the semester just to be thrashed by client feedback later is not fun, very disillusioning, and makes you feel like somehow you're at fault despite following your approved project plan." **Client:** "I feel that teams that don't get selected as the 'best' project in their option might feel quite discouraged after the fact and I don't believe that is the point of the course."
Unexpected Benefits	14	0	Mention of an unexpected benefit **Student:** "Was fun and competitive and I like having the prof as a buffer between us and the client"; "I got a chance to see other people['s] perspective[s] on the same project"; "This way even if the teams did the same option, projects can be very different"; "This gave us a chance to know our own limitations and to see how much can we do as a team for our project"; "Hosting more meetings within our group will enhance our communication and help each other in the project work process"; "I like competing with other teams and see their perspectives"; "I like the mode that other group members come to our session and try our website, which lets us find our problems easier because we work in the same theme"; "Students would also be able to provide support to each other if they are doing the same project which may be helpful with a larger class size"

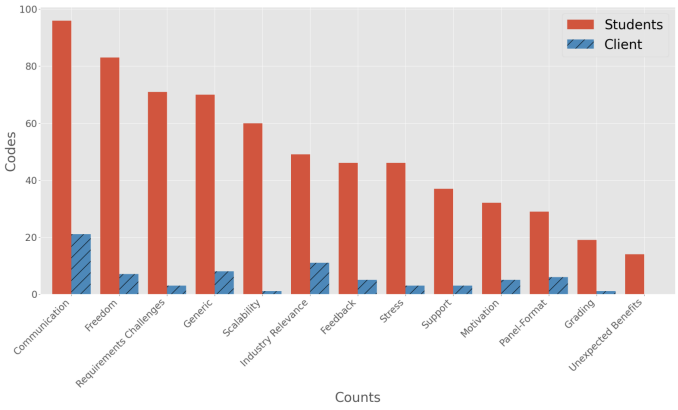

Fig. 1. Resulting codes discovered from our thematic analysis.

for students [2]. For example, one student stated, "working on a project I wanted to work on meant that I cared about the project far more, which caused me to take the course more seriously." This personal connection to the project was seen as a crucial factor in maintaining high levels of motivation because "it felt like the project meant something to someone."

A key concern was client communication because students felt they did not understand client expectations and, therefore, had poor chances of succeeding in the hackathon competition. Although most students prefer the hackathon format over weekly client meetings, a good portion of them wanted more client feedback. The students also attributed their frustrations with tackling an open-ended project to the lack of client communication and feedback on their work. Furthermore, some students felt that not having closer client interactions was undermining its importance, and that resulted in a lower level of engagement and challenge than expected in a more client-focused format. A student questioned the purpose of the client role and whether it helps meet the course learning outcomes. Despite these criticisms, the overall sentiment was that working on client projects provided invaluable learning opportunities and better preparedness for the workplace.

2) General Course Format: Mixed opinions were observed, with some students really liking how the way the course was run and others falling at the extreme opposite of the spectrum. Although details were not provided, we suspect these differences emerged due to the varying student skills. We anticipate that students who had previous project experience found the capstone course more comfortable than those who only programmed in traditional, lecture-based courses. Despite these concerns, the students generally felt that the course format allowed them to focus more on learning and teamwork, without the overwhelming pressure of client expectations.

3) Panel Judges: Students shared various perspectives on the panel format where clients act as judges to evaluate their projects. On the positive side, some students appreciated the structure and external perspective it provided. For instance, one student expressed appreciation for having their project work externally validated, stating, "It was interesting to work in a team and create a project and then the clients would rate us based on that project." Some students agree that the panel format was effective for the class size.

However, several students expressed concerns about how the panel format was implemented and its effectiveness in fairly evaluating their work. One student questioned, "we wonder whether the client really had the chance to look at the proposal at all," which suggests a distrust in the client's opinions and the thoroughness of the review. Another student expressed a disconnect in how the clients evaluated their work because they felt that the criteria should focus on the project's features rather than the design choices made in arriving at the prototype. Some suggestions were made in adjusting the process of the panel evaluation by including "a platform to justify [their] reasoning" in how their proposed solutions.

4) Stress and Motivation: A considerable number of comments mentioned the hackathon format significantly boosted their drive and personal engagement. Students commented that they "push[ed] each other to perform better in the class", competition "boost[ed] the team's energy to push the project to the final [deadline]", and "it [felt] more rewarding if [they] did well." At the same time, a few students expressed they were stressed and felt burnt out by the workload. Since teams competed in a project theme, it would be natural to think that clients would not choose to adopt the projects from a non-winning team. For this reason, some students felt demotivated "if [their] project isn't even going to be used".

An aspect that students enjoyed is the ability to "fully build an app that [they] want" and being able to dedicate eight months to it. In contrast, other project courses in our program typically only start in the last month of the class.

Many students highlighted positive aspects of reduced stress and workload management, such as completing work efficiently and the weekly structure created a rhythm that enabled steady progress. Others noted that the current format "managed to take away unnecessary stress while still providing a valuable experience" and "allowed us to focus on learning concepts instead of working in a rush to meet [the client's] criteria." Some comments compared their client interaction to stories from previous years "that working directly with individual clients can be very invasive and challenging."

5) Feedback and Support: Students wanted more support that would enable them to better align with client expectations and industry standards, however, they felt the feedback received was too generic or not constructive enough to make significant improvements. Many comments reflected the student's lack of preparedness and comfort in tackling such open-ended projects where they had to troubleshoot their own technical problems instead of being placed into a structured assignment context like a traditional course. Overall, the comments highlighted a desire for more comprehensive and personalized support to help students navigate their projects more effectively.

6) Grading: Although the grading components of the course emphasized team collaboration and individual coding contributions, when a team received an unsatisfactory ranking from the client feedback, students felt the client feedback

contradicted the positive progress they had made in the project. Some students also commented they felt a disconnect between the grades they received on the course deliverables and the rankings clients gave them. For example, as long as students fulfill course requirements, their course grades will be high, but if their prototype is relatively worse than those of the other teams, the client ranking will still be low.

Other grading comments did not pertain to the client model, but rather to issues with TA training, requiring students to produce a coding feature every two weeks, and misunderstandings about what constitutes a sizable feature.

7) Scalability: The feedback on scalability, particularly with the expected increase in class size, was predominantly positive, with many students recognizing the benefits of a hackathon format in larger class sizes. Students also suggested that adding more TAs and having an additional instructor could alleviate potential logistical difficulties with larger classes.

E. Results: Client Feedback

1) Communication, Freedom, Requirements, Relevance: Several clients expressed they wanted "more interaction with student groups," and potentially introducing less formal checkpoints "perhaps every month". Some clients seemed to want to be more involved, stating that "students did not reach out to [them]" and suggested an increase in communication could resolve potential challenges in flushing out the project requirements. However, the clients recognized the complexity of maintaining close contact with their teams because "with so many teams, it's hard to build and maintain a good relationship with all the students."

One client pointed out that "the 4 options given seem to cover most of the areas of interest for people, but some of them were skewed with way more teams applying for them." This highlights a potential imbalance, where certain projects attract more interest than others, potentially leading to an uneven distribution of student engagement and learning opportunities. Some suggestions were made about including more projects and including students in the project generation process.

Overall, the clients were impressed by how the projects allowed them to observe students in a professional context without needing to intervene, highlighting the autonomy and initiative demonstrated by the students. Several clients indicated their willingness to hire and provide references for students based on this experience.

2) Panel Format: Clients felt their goals were fully met through this hackathon model. There was a slight concern regarding the student teams feeling discouraged if they did not win the competition.

3) Stress and Motivation: When compared to their previous experience, one client commented that the hackathon format reduced stress for students, stating that "in previous years, a few clients were quite dependent on student teams to deliver a working product that they would use, and I feel that adds undue pressure on some students."

Clients observed that the sense of competition fostered by the hackathon structure was a significant motivator for the students. One specifically noted, "the sense of competition definitely felt like it spurred students to strive to be better, and that was great to see." Another remarked that the hackathon competition "noticeably pushed the students to get more done than I have ever seen in other capstone projects." This feedback suggests that the dynamic and intense nature of the hackathon inspired students to be more productive, leading to impressive outcomes that surpassed the previous years. Clients believed these aspects significantly contributed to a more energetic, engaged, and ambitious student body.

4) Feedback and Support: Clients were asked to provide the top three strengths and areas of improvement during the panels, but there were no comments on that. Instead, the clients emphasized they wanted more opportunities to provide project feedback. The type of feedback the clients envisioned giving would be based on "a demo of progress and Q/A" in "low-pressure meeting[s]". This comment underscores the importance that clients clearly valued the opportunity for regular, structured interactions with the teams, recognizing these sessions would be key in driving project success and student learning. The clients also felt more interaction would provide better support for students. Moreover, one client expressed their ideal version of a client, indicating, "I think having clients who are interested in the development of the students more than the development of their products is a necessity."

Clients also empathized with the increased enrolment constraint and mentioned that "the course could benefit from having multiple instructors" because they recognize that it is "difficult to manage all the teams" alone.

5) Grading: As part of the panel evaluation, clients completed a feedback form about the progress of all the projects in their theme. The instructor converted the responses from these forms to a small client grade which was not visible to the clients. The only comment related to student evaluation was toward teams that did not win the competition and how that result may indirectly negatively impact those students.

6) Scalability: Among the four project themes, one had 2 teams competing in it and another had 12 teams. Clients suggested "an even split for the project [themes]" would help balance the load of the clients. No comments on increasing class sizes were made.

V. Discussion

A. Limitations

One of the limitations of our study is the number of students and clients involved in this single offering of the capstone course. The reported experience could be influenced by the nature of the projects or the specific members of the teaching staff. We should ideally gather data from multiple sections of this course, across TAs and instructors, to gain broader insights on the client and student perceptions of this hackathon model.

B. Benefits of the Hackathon Client Model

This model is less intensive on the administrative side with the instructor not needing to manage the clients and monitor communications between the clients and every student team.

It also provides students with creative freedom, not only allowing them to explore innovative ideas and take ownership of their work, but also showcasing their independence without client interference. Compared to the traditional one-to-one relationship between clients and teams, this hackathon model allows for a greater degree of ownership which contributes to an increase in motivation as evidenced in the literature [2]. The findings show that both students and clients recognize the motivation generated by the dual impetus of a real-world problem and a competitive environment. The new capstone also retains its emphasis on industry alignment, with both students and clients highlighting the model's capacity to bridge the gap between education and real-world industry needs.

C. Drawbacks of the Hackathon Client Model

While the competitive element may have been motivating for most students, it also increased stress levels with its added demands on some students. Students also struggled with other aspects of the competition – they did not understand the differences between how course grades and client rankings are assigned because some students received a low ranking despite having completed all the course requirements satisfactorily. Some students did not understand the role of the client because they expected the clients to provide technical support and ignored some client recommendations made at the panels. A lack of communication with the clients often manifested as a misalignment in expectations and project requirements. Students perceived a lack of project guidance from the client which also manifested as frustration in the first judging panel when they received lower client rankings than expected.

Negative sentiments toward intensive workload may be misattributed to the hackathon client model since the instructor's expectations and project grading approach would not change irrespective of the shape of the client's involvement. The nature of a long-term, open-ended capstone project also caused discomfort for students who felt unprepared and unsupported in the course. Rather than associating this issue with the hackathon client model or the capstone course, the underlying cause likely points to a necessary program-level change to help students be better prepared for such workplace settings where students can independently troubleshoot technical problems.

D. Implications in Course Design

The literature emphasizes that continuous course revision is necessary to enhance learning [5]. The following are points that should be taken into consideration in future capstone offerings. First and foremost, the course design should address the communication challenges by creating structured channels for clear dialogue between students and clients. Considering the context of large classes, a possible solution is to introduce information sessions where teams have the option to ask clients questions at dedicated intervals of the course.

Although the clients want a more even distribution of teams working on different project themes, we note that the literature reports industry projects are generally more popular for students [3]. We also saw that when a famous company is involved as a client, many students express interest in working on their projects over other projects offered by smaller companies. Therefore, careful consideration is needed in recruiting and selecting clients for the same course.

We observed discrepancies in the student's understanding of the client's role and how clients might evaluate projects. For students with little work experience and a limited understanding of the competitiveness of the industry environments, more explanation and context are needed. For large cohorts, it would be infeasible to replicate situated learning and experiential learning opportunities for students to gain this industry context. One possible solution is to have industry speakers come to class to pitch their projects and talk about their expectations for the year. This approach may create more congruence between the course evaluation criteria, student workload expectations, and clients' desired project outcomes.

VI. Conclusion

Capstone courses are critical in bridging the gap between academic learning and industry practice. While our institution has adopted a client-based model for our capstone course, class sizes have become unsustainable in recent years and we must look for new pedagogical approaches to provide similar learning outcomes for our students. In this work, we experimented with a novel hackathon client model as a potential solution to running client-based capstone courses in large classes. While it was generally perceived as a successful operation by most students and clients alike, the logistics of implementing the model can benefit from several improvements. Future work can further explore how hackathons may promote certain student learning outcomes, such as innovative design, collaborative problem-solving, and professional programming conduct. Beyond the confines of the capstone, this case study unveiled the continued disjoint between the expectation of independent thinking and troubleshooting common to industry settings and students' desire for scaffolded learning and carefully guided support. This transition cannot be placed solely inside the capstone but must be extended into earlier courses that better prepare students throughout the curriculum.

References

[1] F. Abu Salem, I. Damaj, L. Hamandi, and R. Zantout. Effective assessment of computer science capstone projects and student outcomes. *International Journal of Engineering Pedagogy (iJEP)*, 10:pp. 72–93, 2020.

[2] R. Adams and C. Kleiner. Collaboration support in an international computer science capstone course. In G. Meiselwitz, editor, *Proceedings of the Social Computing and Social Media Conference*, pages 313–323, 2016.

[3] G. I. Allen. Experiential learning in data science: Developing an interdisciplinary, client-sponsored capstone program. In *Proceedings of the ACM Technical Symposium on Computer Science Education (SIGCSE)*, page 516–522, 2021.

[4] M. C. Bastarrica Piñeyro, D. Perovich Gerosa, and M. Marques Samary. What can students get from a software engineering capstone course? In *Proceedings of the International Conference on Software Engineering: Software Engineering Education and Training Track*, 2017.

[5] F. C. Berry, W. Huang, and M. Exter. Improving accuracy of self-and-peer assessment in engineering technology capstone. *IEEE Transactions on Education*, 66:174–184, 2023.

[6] V. Braun and V. Clarke. Thematic analysis. In H. Cooper, P. Camic, D. Long, A. Panter, D. Rindskopf, and K. Sher, editors, *APA handbook of research methods in psychology: Research designs: Quantitative, qualitative, neuropsychological, and biological*, page 57–71. American Psychological Association, 2012.

[7] N. Deepamala and G. Shobha. Effective approach in making capstone project a holistic learning experience to students of undergraduate computer science engineering program. *JOTSE: Journal of Technology and Science Education*, 8:420–438, 2018.

[8] R. F. Dugan. A survey of computer science capstone course literature. *Computer Science Education*, 21(3):201–267, 2011.

[9] V. Farrell, G. Ravalli, G. Farrell, P. Kindler, and D. Hall. Capstone project: Fair, just and accountable assessment. In *Proceedings of the Innovation and Technology in Computer Science Education (ITiCSE)*, pages 168–173, 2012.

[10] R. Mandale, A. Patil, and A. Adamuthe. Improved undergraduate software capstone project development with adoption of industry practices. *Journal of Engineering Education Transformations*, 34:74–89, 2021.

[11] J. J. Olarte, C. Domı´nguez, A. Jaime, and F. J. Garcı´a-Izquierdo. Student and staff perceptions of key aspects of computer science engineering capstone projects. *IEEE Transactions on Education*, 59(1):45–51, 2016.

[12] R. Parker. How do you feel: Affective expressions from computer science senior capstone projects. In *Proceedings of the International Conference on Learning and Teaching in Computing and Engineering (LaTICE)*, pages 35–42, 2017.

[13] A. Radermacher, G. Walia, and D. Knudson. Investigating the skill gap between graduating students and industry expectations. In *Proceedings of the International Conference on Software Engineering*, page 291–300, 2014.

[14] K. Sha. Lessons and experiences from teaching computing science capstone project courses. In *Proceedings of the International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 990–995, 2021.

[15] C. C. Tappert, A. M. Leider, and S. Li. Student assessment in a capstone computing course. In *Proceedings of the Southeast Decision Sciences Institute (SEDSI) Conference*, pages 1–7, 2019.

[16] I. Weissberger, A. Qureshi, A. Chowhan, E. Collins, and D. Gallimore. Incorporating software maintenance in a senior capstone project. *International Journal of Cyber Society and Education*, 8:31–38, 2015.

[17] J. Yousafzai, I. Damaj, and M. El Abd. A unified approach for assessing capstone design projects and student outcomes in computer engineering programs. In *Proceedings of the IEEE Global Engineering Education Conference (EDUCON)*, pages 333–339, 2015.