Infusing Computing: Moving a Service Oriented Internship Program Online

Amy Isvik
Dept. of Computer Science
North Carolina State University
Raleigh, North Carolina, USA
aaisvik@ncsu.edu

Veronica Cateté
Dept. of Computer Science
North Carolina State University
Raleigh, North Carolina, USA
vmcatete@ncsu.edu

Dave Bell
Dept. of Computer Science
North Carolina State University
Raleigh, North Carolina, USA

Isabella Gransbury
Dept. of Computer Science
North Carolina State University
Raleigh, North Carolina, USA

Tiffany Barnes
Dept. of Computer Science
North Carolina State University
Raleigh, North Carolina, USA

***Abstract*—As virtual conferencing technology becomes more common and situations make in-person experiences difficult or unsafe to host, the need for online internships to support sustained participation in computing increases. We investigate the problem of how to provide a meaningful experiential education program in a virtual environment and serve geographically dispersed participants through our experience with moving a service oriented internship program online. Our computer science internship program leverages high school interns' programming skills and classroom experience to assist teachers in developing computing-infused lessons for their classrooms. Using a combination of synchronous and asynchronous activities, we trained our interns in how to make these lessons and helped interns build community amongst themselves. Our interns created over 90 lessons during the summer and helped over 50 teachers create their own lessons at an infusing computing professional development.**

***Keywords*—computing education, virtual internship, service-learning**

I. Introduction & Background

Many internships were cancelled in summer 2020 due to COVID-19 [1]. To avoid this, our university-based computer science (CS) research team decided to run a virtual program, leaving us with a daunting task and question: *how do you provide a meaningful experiential education program in a virtual environment?* Our CS internship program provides high school students a chance to develop technical and professional skills while solving community challenges by helping teachers develop computing-infused lessons for their classrooms. Our original iteration of the program occurred over 8 weeks in Summer 2019 where 19 interns served as programming support for an infusing computing professional development (IC PD), created computing-infused projects, discussed ethics, and did research rotations in socially relevant computing labs focusing on health, education, and critical infrastructures [2].

To run our online Summer 2020 internship, we adjusted our content, delivery methods, and recruitment; advertising it as a completely virtual online program. Using an online Zoom-based model, we expanded our applicant pool to non-local students and recruited 29 participants from around the country. We replaced research lab rotations with multiple development sprints, developing professional software engineering skills while collaborating on projects for social good. This article relays the intern experiences and establishes a lightweight model for broadening participation in computing at multiple ends in the computing education pipeline.

In a similar domain, researchers found positive evidence of science internships helping students see science beyond familiar stereotypes and reflect upon their relationship with science [3]. Internships in research labs were also shown to humanize scientists and provide insight on working in that discipline [3]. Roth suggests internships should reflect activities of professionals in the field with open-ended tasks, "communities of inquiry in which knowledge, practices, resources, and discourses are shared," and assistance from peers and advisors [4].

One successful high school CS internship was the 6-week Girls Embrace Technology (GET) program, where 36 girls collaboratively developed hands-on educational software to teach elementary school children [5]. Researchers reported that students' misconceptions about CS (it wasn't collaborative) were clarified and their interest in computing grew. This program combined learning real-world technical skills in an industry-modelled experience with a service-oriented project that showed interns how CS could be used to help others.

Buckley et al. believe that focusing on teaching using "computing for a cause, or computing as a means to solve problems, rather than as an end in and of itself" is an important tool to recruit students to the discipline [6]. This can show how CS can be used to impact communities, promoting sustained engagement in CS and STEM in general [7]–[11].

Our program immersed interns in an active research lab, completing the same tasks as our lab researchers: creating educational tools and software. For this task, interns decide

978-1-6654-4905-2/21/$31.00 ©2021 IEEE

what and how they will teach in their computing-infused lessons. They receive a structured community to collaborate and ask for help from peers, graduate mentors, and researchers.

A. Collaboration

Research has shown that explicit training in online collaboration tools is needed to design an entirely virtual training experience, especially when developing a sense of community [12]. Virtual programs also require intentional designs for active participation, collaboration, and engagement [13]. Regarding our virtual internship experience, collaboration and community are very important, as research by Denner et al. showed that girls benefit from learning environments that "involve collaboration with peers, activities that challenge stereotypes, and exploration of social identities" [14]. Additionally, NCWIT identifies collaborative learning methods as a 'best practice' for growing inclusive communities and broadening participation in computing [15].

We encouraged interns to work using pair-programming as studies have demonstrated the pedagogical benefits of pair-programming, including more efficient learning of tasks [16]. Originally used in industry, pair-programming tasks one person to be the 'Driver' constructing the code and another to be the 'Navigator' articulating the plan [17]. In a "driver-driver" model, both students program elements of the project at the same time [18]. Research found that collaborating while working on separate tasks allows greater autonomy over one's own works and creates positive interdependence on each partner to complete the task [19]. We trained interns on both methods and let them choose their preference as the research suggests there are negligible differences and reasonable trade-offs between them [18], [20].

II. Context

Since interns and researchers were satisfied with the original program's outcomes, skills gained, and contributions made [2], we aimed to preserve the overall participant outcomes when transitioning online. We detail the critical elements for transition below.

A. Minors Regulations

Expectations for employee and staff interactions with minors in virtual settings included agreements to only communicate through official program platforms and not bully or harass people. Similar expectations were emailed to participants. Staff were expected to have at least two responsible parties on video calls with minors. Participants could work in breakout rooms while staff members randomly checked in.

B. Tool Selection

We chose Zoom [21] for synchronous activities due to its video, audio, screen-sharing, breakout room capabilities, and chat features; participants not using video or audio could still participate in discussion. Breakout room capabilities were particularly important to facilitate collaboration. For asynchronous communications, we used Slack [22], a messaging platform which allows you to send messages to individuals, a large group regarding a specific topic, and create small group messages. Interns also used Hopin [23], a web-conferencing service selected by organizers, during the IC PD.

C. Participant Information

We selected 29 interns (23 female, 7 male) from 9 states by contacting high schools across our state and advertising to two national AP CS Principles teacher networks. Intern programming experience ranged from self-taught beginner to having taken multiple programming courses in school. Intern demographics are shown in table I; interns were predominantly Asian (24), with 1 Hispanic (any race), three 3 Black/Multiracial, and 4 White/Caucasian. Facilitators were a Hispanic female research scientist and a White female graduate student.

TABLE I
DEMOGRAPHIC INFORMATION OF HIGH SCHOOL INTERNS

Total	Gender			2019-2020 Grade Level				Multiracial, Black or Hispanic
	Female	Male	Other	9th	10th	11th	12th	
N=29	79%	21%	0%	10%	21%	59%	14%	14%

III. Program Implementation

Two events bound our summer internship: piloting a distributed computing curriculum and the IC PD [24]. Between these, interns learned about CS education research, pedagogy, and software engineering practices before developing their own educational tools and programs during three one-week sprints.

A. Week One: Onboarding to New Frontiers

The first week operated synchronously from 9 AM to 4 PM with an hour lunch break. Each morning, interns completed training with program staff. They were introduced to the tools they would use and encouraged to set Zoom profile pictures if their webcams are off to help connect with peers. Interns split into groups of four to read *Computational Thinking* by Jeanette Wing and another CS education research paper unique to their group [24]–[29] which they reported on in group discussion. Additionally, interns familiarized themselves with block-based coding activities and terminology used in the IC PD [26], [28], [30]. After activities, debriefing sessions helped interns synthesize the information and consider ways to build better computing-infused lessons.

Each afternoon, to pilot the CS Frontiers project and standardize intern programming skills, 7 high school CS teachers with NetsBlox programming training [31] worked in teams to teach classes of 8-9 interns. Interns worked in synchronous pairs using NetsBlox to learn advanced computing topics including remote procedure calls, graphing charts with health data, and distributed computing. Afterwards, interns extended an activity with their own ideas, often plotting Covid-19 data on charts and maps. Interns presented their projects to NetsBlox and Snap*!* creators in a final demo showcase, marking a "very cool" event for interns.

B. Week Two: Planning to Create

Week two focused on software engineering skills and pedagogical foundations for creating computing-infused lessons. The first three days focused on software engineering practices such as project planning and debugging strategies. Interns used planning tools to complete one-day coding activities and practiced debugging by reading their code aloud, writing/using code comments, and tinkering with snippets of code. They concluded by creating mini computing-infused lessons.

Each morning, we presented CS education research and related theories. Topics included, but not limited to, culturally relevant education, socially relevant computing, scaffolding, universal design, and using the Teacher Accessibility, Equity, and Content (TEC) rubric for evaluating computing curricula [32]. Interns were encouraged to discuss previous educational experiences and consider applications for their lessons.

C. Weeks Three - Five: Sprint to the Finish

Interns self-arranged into teams or worked individually to develop computing-infused lessons for non-computing classes. Interns aligned activities with state educational standards and chose content areas interesting to them or requested by teachers at the IC PD.

To structure their asynchronous development, interns had daily scrum meetings with facilitators and 3-5 peers to share their progress, challenges, plans, and request help, following industry conventions [33]. Interns were prompted to consider how their lessons would be used and create accompanying materials, including teacher and student guides. Interns also presented their activities and materials in weekly project showcases to receive peer feedback and share their work.

D. Week Six: Infusing Computing PD

During the final week of the internship, interns were code helpers for a 120+ person, 4-day IC PD. This PD was the catalyst for our program and interns were expected to assist teachers ranging in attitude from 'I'm afraid of technology' to 'why are we using a toy language?'. Interns assisted with technology difficulties, coding assignments, and create tasks [24]. Intern pairs were assigned to a facilitator and groups of 12-14 teachers. Interns moved between teachers' Hopin [23] rooms, offering help and guidance. Teachers without partners would pair-program with interns when needed.

During the Create sessions, interns gathered into help channels encompassing 5 participant cohorts. While IC PD participants worked, interns waited, discussing with peers and facilitators project ideas and future plans. When teachers arrived, interns would provide assistance for a range of tasks including brainstorming, altering lessons interns made, and development of specific functionality such as word banks.

The final day of the internship featured end-of-summer presentations to department faculty followed by an afternoon celebration with virtual games and superlative awards such as Science Super Star.

IV. Intern Experiences & Discussion

Data on the interns' experience was collected from 28 end-of-internship presentations, as one intern was absent. Interns answered several questions about the internship: 1) What they liked, 2) What they would change, and 3) Their favorite thing that happened. We followed the strategies in Braun and Clarke's six-phrase framework for our thematic analysis [34]–[36], using open coding to identify interesting themes discussed below.

A. Flexibility

Similar to the in-person program [2], over half of the interns spoke positively about the flexibility of the program. Interns liked the freedom to choose lesson content, partners (or work individually), and asynchronous work schedule. Providing a flexible framing to the program could help deter perceptions of this being a rigid academic class.

B. Collaboration

Interns enjoyed working and connecting with others their age around the country. Several interns mentioned that they wished for more collaboration and support to help find partners to work with. Interns that collaborated with partners in weeks 3-5 employed both driver-navigator and driver-driver methods. Several interns spoke positively about the atmosphere; saying it was helpful and collaborative. They also appreciated the quick responses from facilitators when requesting help. We were pleased to replicate this feature from the prior implementation despite the virtual change [2].

Unfortunately, we overlooked synchronous recreational time for interns to socialize. Due to the asynchronous development cycles and shortness of the daily scrum meetings, social opportunities diminished after the first week and interns mainly interacted with their partner. Our in-person internship provided numerous organic chances for socializing and had a near daily games group that convened during lunch [2]. In future virtual internships, we will intentionally include a weekly 'social hour' as building community and increasing interns' sense of belonging can positively impact their persistence in CS [37].

C. Working with Teacher Clients

Two-thirds of the interns listed working with teachers as (one of) their favorite thing(s), mentioning they enjoyed helping teachers bring their ideas "to life." Interns viewed the IC PD as the culmination of their summer work and enjoyed "seeing everything coming to fruition." Being situated as the experts, one intern mentioned that her favorite thing was "the satisfaction of helping teachers who were overwhelmed and seeing them understand what they were doing." We have seen the importance of helping teachers and "the ability to have an impact" in student reflections from this iteration of the internship and the previous internship [2], and will continue to make this a central part of our program.

Teachers also praised the interns, stating that they were "a wonderful support" and "great to have" on their team. One teacher stated that they would have skipped the rest of the

PD without the help they received from an intern. In general, teacher feelings about the interns can be summarized by this chat message: "I LOVE THE HS INTERNS."

Intern suggested connecting with teachers earlier to get more accurate teacher requests for lessons and to better see the impacts of their work directly in the classroom.

D. Program Structure

Most interns would have preferred an in-person experience, if possible. However, some mentioned they preferred working from home. While most interns considered scheduling and communication as timely, two thought they could be improved. Interns were given a written schedule at the start of the internship, but it was not referred to later in the program.

Generally, interns liked using Slack and Zoom to communicate. Many, however, expressed frustration with Hopin. They found navigation difficult and wanted more time to become familiar with the interface. We allowed them to explore Hopin before the PD, but the organizers had not finalized Hopin's usage yet, limiting the tutorial. It would be best if all parties were comfortable with new software tools before any critical use.

Interns found our tools and resources very useful, "If I needed help with anything I could find the answer to my problem through the presentations." However, they wished we created a computing-infused lesson as a group for an example of the process. One intern suggested providing a rubric for self-evaluation of the lessons. We found these suggestions helpful and although we introduced the TEC rubric during training, we did not highlight its use for self-evaluation nor provide it in an easily applicable format. Research has shown that using a rubric for self-evaluation during lesson plan creation can result in higher quality lesson plans [38].

E. Internship Activities

Four interns stated they liked helping visiting researchers in general and sixteen interns mentioned beta-testing the NetsBlox distributed computing curriculum as a favorite activity. They enjoyed learning about block-based programming languages and felt special having faculty and Netsblox creators attend their project showcase.

Interns also enjoyed training in computing education, computational thinking, and pedagogical practices. Eight interns directly mentioned enjoying the pedagogical training. One intern appreciated seeing "strong examples of CS lessons" by exploring the Epidemics and Food Web curriculums [26], [28]. Another shared that "the presentations and research papers we read at the beginning of the internship [were] super informative and gave me an idea of how to improve my project."

Nearly half of the interns mentioned liking the creation of computing-infused lessons, counted separately from helping teachers. Interns appreciated the project showcase to see their peers' work and gain inspiration for their own projects, valuing peer feedback. Several interns shared thoughts on improving project showcasing such as making them shorter (they were typically 2 hours) and dedicating time to investigate the materials more closely.

F. Student Engagement

Participants remained highly engaged in internship activities despite conflicts. For example, one intern was traveling with family, and excused, during the NetsBlox introduction. However, they were so fascinated that they tethered their laptop to cell phone data to work while traveling. Despite the unreliable network and repeat disconnects, they persisted with the activities. Other interns with conflicts during the first week completed the lesson materials outside of standard intern hours and made their own extension activities. Many interns expressed interest in continuing to work with our lab to meet teacher requests for computing-infused lessons or on other projects after the internship.

V. Outcomes & Limitations

Throughout six weeks, interns created over 90 computing-infused lessons, available first to 146 teachers at the IC PD and then to the public in an online repository. They also assisted over 50 teachers with creating computing infused lessons. Interns gained insight into CS as a profession, experiencing a wide range of tasks and responsibilities including project management, meeting client requirements, and teleworking. Interns' understanding of CS as a discipline and career developed as their awareness of the possibilities of working as a computer scientist and exposure to different role models increased.

This report is limited to qualitative data gathered from intern presentations, teacher comments, and other activities witnessed by program facilitators. Therefore, our interview data and pre-post test data measuring intern outcomes in CS attitudes, changes in understanding computational thinking, self-efficacy, and desire to pursue computing as a major are not included in this report. Furthermore, we are not fully aware of the interns' physical working conditions.

VI. Conclusions & Future Work

We have shown that our program can be successfully adapted to a virtual format while retaining important outcomes of the in-person internship [2]. We provided a successful and engaging experience where interns gained computing and professional skills and acted as experts by assisting teachers. Through this opportunity, we believe we have helped build their confidence and efficacy in the field, which is important for retention [7]. This structure also allows teachers at the IC PD to focus on connecting their subject to computational thinking while providing meaningful active learning experiences for their students.

Overall, we feel that our internship is an example of a successful service-based experiential education program for CS that can be implemented virtually or in-person with equivalent levels of success. We hope that others can use this knowledge to create similar programs. We plan to continue running this internship in future years, extending it to another university. We additionally plan to analyze changes in the intern's understanding of computational thinking as a result of participation.

REFERENCES

[1] N. N. A. of Colleges and Employers, "Coronavirus quick poll - april 2020 quick poll results," apr 2020.

[2] A. Isvik, V. Cateté, and T. Barnes, "Flames: A socially relevant computing summer internship for high school students," in *2020 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*. IEEE, 2020.

[3] P.-L. Hsu and W.-M. Roth, "From a sense of stereotypically foreign to belonging in a science community: Ways of experiential descriptions about high school students' science internship," *Research in Science Education*, vol. 40, no. 3, pp. 291–311, 2010.

[4] W.-M. Roth, *Authentic school science: Knowing and learning in open-inquiry science laboratories*. Netherlands: Springer Science & Business Media, 2012, vol. 1.

[5] J. F. Sullivan, D. Reamon, and B. Louie, "Girls embrace technology: A summer internship for high school girls," in *33rd Annual Frontiers in Education, 2003. FIE 2003.*, vol. 1. Westminster, CO, USA: IEEE, 2003, pp. T4D–6.

[6] M. Buckley, J. Nordlinger, and D. Subramanian, "Socially relevant computing," *ACM SIGCSE Bulletin*, vol. 40, no. 1, pp. 347–351, 2008.

[7] L. J. Sax, K. J. Lehman, J. A. Jacobs, M. A. Kanny, G. Lim, L. Monje-Paulson, and H. B. Zimmerman, "Anatomy of an enduring gender gap: The evolution of women's participation in computer science," *The Journal of Higher Education*, vol. 88, no. 2, pp. 258–293, 2017.

[8] S. V. Rosser, *Female-friendly science: Applying women's studies methods and theories to attract students*. New York, NY, USA: Pergamon, 1990.

[9] N. Z. Khan and A. Luxton-Reilly, "Is computing for social good the solution to closing the gender gap in computer science?" in *Proceedings of the Australasian Computer Science Week Multiconference*. New York, NY, USA: ACM, 2016, pp. 1–5.

[10] S. Chapman and R. Vivian, "Engaging the future of stem: A study of international best practice for promoting the participation of young people, particularly girls, in science, technology, engineering and maths (stem)," Chief Executive Women (CEW) Ltd, Tech. Rep., 2017.

[11] J. Hoxmeier and M. M. Lenk, "Service-learning in information systems courses: Community projects that make a difference," *Journal of Information Systems Education*, vol. 14, no. 1, p. 10, 2020.

[12] T. J. McConnell, J. M. Parker, J. Eberhardt, M. J. Koehler, and M. A. Lundeberg, "Virtual professional learning communities: Teachers' perceptions of virtual versus face-to-face professional development," *Journal of science education and technology*, vol. 22, no. 3, pp. 267–277, 2013.

[13] J. M. Mumford, L. Fiala, and M. Daulton, "An agile k-12 approach: Teacher pd for new learning ecosystems," in *Handbook of Research on Teacher Education and Professional Development*. Hershey PA, USA: IGI Global, 2017, pp. 367–384.

[14] J. Denner, L. Werner, S. Bean, and S. Campe, "The girls creating games program: Strategies for engaging middle-school girls in information technology," *Frontiers: A Journal of Women Studies*, vol. 26, no. 1, pp. 90–98, 2005.

[15] N. C. for Women and I. Technology. (2019, mar) Engagecsedu - engagement practices. [Online]. Available: https://www.engagecsedu.org/engagement/make-it-matter

[16] N. Salleh, E. Mendes, and J. Grundy, "Empirical studies of pair programming for cs/se teaching in higher education: A systematic literature review," *IEEE Transactions on Software Engineering*, vol. 37, no. 4, pp. 509–525, 2010.

[17] L. Williams, R. R. Kessler, W. Cunningham, and R. Jeffries, "Strengthening the case for pair programming," *IEEE software*, vol. 17, no. 4, pp. 19–25, 2000.

[18] Z. Zacharia, D. Boulden, J. Vandenberg, J. Tsan, C. Lynch, E. Wiebe, and K. Boyer, "Collaborative talk across two pair-programming configurations," in *A Wide Lens: Combining Embodied, Enactive, Extended, and Embedded Learning in Collaborative Settings, 13th International Conference on Computer Supported Collaborative Learning (CSCL) 2019*, vol. 1. Lyon, France: École Normale Supérieure, 2019, pp. 224–231.

[19] A. Lie, "Cooperative learning," 2002.

[20] J. Tsan, J. Vandenberg, Z. Zakaria, J. B. Wiggins, A. R. Webber, A. Bradbury, C. Lynch, E. Wiebe, and K. E. Boyer, "A comparison of two pair programming configurations for upper elementary students," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. New York, NY, USA: ACM, 2020, pp. 346–352.

[21] I. Zoom Video Communications. (2020, aug) Video conferencing, web conferencing, webinars, screen sharing - zoom. Zoom. [Online]. Available: https://zoom.us/

[22] I. Slack Technologies. (2020, aug) Where work happens — slack. Slack. [Online]. Available: https://slack.com/

[23] Hopin. (2020, aug) Hopin: Online venue for virtual events. Hopin. [Online]. Available: https://hopin.to/

[24] R. Jocius, D. Joshi, Y. Dong, R. Robinson, V. Cateté, T. Barnes, J. Albert, A. Andrews, and N. Lytle, "Code, connect, create: The 3c professional development model to support computational thinking infusion," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. New York, NY, USA: ACM, 2020, pp. 971–977.

[25] J. M. Wing, "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33–35, 2006.

[26] N. Lytle, V. Cateté, D. Boulden, Y. Dong, J. Houchins, A. Milliken, A. Isvik, D. Bounajim, E. Wiebe, and T. Barnes, "Use, modify, create: Comparing computational thinking lesson progressions for stem classes," in *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, 2019, pp. 395–401.

[27] N. Lytle, Y. Dong, V. Cateté, A. Milliken, A. Isvik, and T. Barnes, "Position: Scaffolded coding activities afforded by block-based environments," in *2019 IEEE Blocks and Beyond Workshop (B&B)*. Memphis, TN, USA: IEEE Computer Society, 2019, pp. 5–7.

[28] V. Cateté, N. Lytle, Y. Dong, D. Boulden, B. Akram, J. Houchins, T. Barnes, E. Wiebe, J. Lester, B. Mott *et al.*, "Infusing computational thinking into middle grade science classrooms: lessons learned," in *Proceedings of the 13th Workshop in Primary and Secondary Computing Education*. New York, NY, USA: ACM, 2018, pp. 1–6.

[29] T. Barnes, H. Richter, E. Powell, A. Chaffin, and A. Godwin, "Game2learn: building cs1 learning games for retention," in *Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education*. New York, NY, USA: ACM, 2007, pp. 121–125.

[30] Y. Dong, V. Catete, R. Jocius, N. Lytle, T. Barnes, J. Albert, D. Joshi, R. Robinson, and A. Andrews, "Prada: A practical model for integrating computational thinking in k-12 education," in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. New York, NY, USA: ACM, 2019, pp. 906–912.

[31] B. Broll, A. Lédeczi, P. Volgyesi, J. Sallai, M. Maroti, A. Carrillo, S. L. Weeden-Wright, C. Vanags, J. D. Swartz, and M. Lu, "A visual programming environment for learning distributed programming," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. New York, NY, USA: ACM, 2017, pp. 81–86.

[32] D. Weintrop, M. Coenraad, J. Palmer, and D. Franklin, "The teacher accessibility, equity, and content (tec) rubric for evaluating computing curricula," *ACM Transactions on Computing Education (TOCE)*, vol. 20, no. 1, pp. 1–30, 2019.

[33] L. Rising and N. S. Janoff, "The scrum software development process for small teams," *IEEE software*, vol. 17, no. 4, pp. 26–32, 2000.

[34] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.

[35] M. I. Alhojailan, "Thematic analysis: A critical review of its process and evaluation," *West East Journal of Social Sciences*, vol. 1, no. 1, pp. 39–47, 2012.

[36] M. Maguire and B. Delahunt, "Doing a thematic analysis: A practical, step-by-step guide for learning and teaching scholars." *All Ireland Journal of Higher Education*, vol. 9, no. 3, 2017.

[37] R. M. Powell, "Improving the persistence of first-year undergraduate women in computer science," *ACM SIGCSE Bulletin*, vol. 40, no. 1, pp. 518–522, 2008.

[38] G. Ozogul, Z. Olina, and H. Sullivan, "Teacher, self and peer evaluation of lesson plans written by preservice teachers," *Educational Technology Research and Development*, vol. 56, no. 2, p. 181, 2008.