**Developing a Practical-Based Capstone Course Focused on Real-World Coding Challenges.**

**Analyzing the Disconnect between Academic Preparation and Industry Expectations in Computer Science.**

A Thesis Prospectus
In STS 4500
Presented to
The Faculty of the
School of Engineering and Applied Science
University of Virginia
In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Computer Science

By
Adam Khan

October 27, 2023

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

ADVISORS

Benjamin Laugelli, Department of Engineering and Society

Briana Morrison, Department of Computer Science

## Introduction

The prevalent disconnect between the skills imparted through computer science curricula in universities and those expected in the software engineering industry presents a significant sociotechnical challenge. Despite university programs specializing in theoretical knowledge, they often inadequately equip students with practical technical skills that are valuable in real-world software engineering scenarios like collaborative problem-solving, project management, maintainable code writing, and managing large codebases, leading to a strenuous transition for graduates into their careers. To address this issue, my technical project proposes creating a new capstone course designed to effectively mirror industry-like software engineering challenges. This proposed curriculum addition aims to provide students with a hands-on experience that closely aligns with the demands of the software engineering industry workplaces.

Given that the issues at hand span both technical and social factors, understanding the root of this disconnect is vital to overcoming it. My Science and Technology Studies (STS) research will employ the Social Construction of Technology (SCOT) framework to understand the perspectives of various stakeholders, including students, educators, and employers, revealing how the prevailing approach to education might be reoriented to better suit all the stakeholders. By examining the incentives and priorities that currently inform the theory-focused curriculum dominant in academia, we can explore possibilities for a curriculum that better serves real-world needs.

The problem of the industry-academia divide in computer science education is inherently sociotechnical, requiring both improvements in the hands-on training provided and a critical analysis of the organizational dynamics upholding the current status quo. By examining the institutional context and drawing significant insights from the STS research, we can inform and guide the design of the technical project to better adhere to the needs of the industry. In the following sections, I outline two mutually informing research proposals: a technical project proposing the design of an immersive capstone course and an STS project investigating the causes of the prevalent industry-academia gap in computer science.

**Technical Project Proposal**

The primary challenge I aim to address pertains to the gap between what is taught in academic computer science curricula and the technical skills expected from graduates in the software engineering industry. While theory forms an important foundation for understanding concepts, the industry mandates a functional, application-focused approach. This divergence often leaves both aspiring engineers and the industry at large at a disadvantage. As a result, many CS graduates struggle to transition into software engineering careers where they must work on large, complex codebases and deal with real-world programming challenges (Parsons, 2022).

Several academic institutions and practitioners attempt to bridge this gap via internships, industry-academia collaborations, and mock projects (Salzabilla, 2023). Some CS programs offer a small number of elective courses focused on software engineering, but practical training is not a priority or requirement in many cases. At the University of Virginia, CS 3240: Advanced Software Development Techniques is the only elective regularly offered that covers software engineering fundamentals (*UVA CS Advising Guide* 2023).

Without sufficient hands-on experience, graduates lack critical abilities expected by employers, like using version control, collaborating on teams, writing readable and maintainable code, documenting systems, and debugging in production environments (Hanna et al., 2014).

By refining this aspect of education, my proposed solution delivers several practical benefits. Graduates of the new curriculum would be 'industry-ready,' capable of tackling real-world problems right out the gate, reducing the learning curve when they start their careers. This proactive arrangement would also serve employers well, reducing onboarding times and training cost overheads.

To address this gap, I propose developing a new practical capstone course for senior CS majors modeled after real-world software development. The course will involve working in groups on an existing large codebase to implement complex features, fix bugs, refactor code, write documentation, and practice key skills like release management, testing, and code review. Through realistic projects, students will gain vital experience and become better prepared for the practical challenges of software careers after graduation.

To construct this course, I will apply various software engineering principles and pedagogical models, underpinned by Computer Science guidelines and industry best practices. A blend of traditional teaching methodologies and a Problem-Based Learning Model will serve as a foundational strategy in its design (Duch et al., 2001). The course will utilize proven software engineering methods like agile development, continuous integration, unit testing, and refactoring to guide student work.

To validate the effectiveness and viability of this new approach, I will collect a variety of data. This will encompass pre- and post-course surveys to measure student confidence and skill levels, analysis of student projects for improvements over their course tenure, and industry feedback for project relevance and quality. Surveys of industry professionals will identify the most important practical skills to incorporate. Student project work and feedback will be used to evaluate and refine the curriculum.

## STS Project Proposal

In the endeavor to frame the STS research question, it is necessary to consider the stakeholders involved and their perspectives on the issue. The primary case under examination is the disconnect between what is taught in universities and the expectations of the software industry expectations, particularly pertaining to practical skills. I propose to employ the Social Construction of Technology (SCOT) framework as the analytical model for understanding different perspectives, including those of students, academia, and the industry.

A historically similar case, that provided a transition from theoretical-based learning to that of real-world market demand, was witnessed in the field of business studies. The business curricula that were too theoretical and non-responsive to business needs came under criticism in the 1950s, leading to the formulation of the 'Ford and Carnegie' reports which advocated for increased relevance of teaching methodologies to business practices (Jarzabkowski & Whittington, 2008). This prompted the integration of case studies, internships, and capstone courses within business curricula thus leading to more 'business-ready' graduates. The lessons learned from this transition can be useful in understanding factors that make the realignment of academic curricula with industry expectations a challenge.

Previous researchers have mainly focused on intervention methods like capstone projects, internships, MOOCs (Massive Open Online Courses), and their impacts (Achee et al., 2022; Patil et al., 2023; Minichiello et al., n.d.; Lent & Brown, 2013). They've however paid considerably less attention to the root causes of the disconnect. By focusing on this knowledge gap, this project aims to provide an in-depth understanding of the social, cultural, and institutional factors that perpetuate the status quo in computer science education. SCOT positions that societal segments shape and define technology, underlining their indispensable role in constructing its development (Johnson, 2005). Specifically, SCOT emphasizes that technological design is inherently a social process, influenced by the interests, values, and concerns of various 'relevant social groups' – those with a stake in the technology's development, use, and non-use. It introduces the concept of interpretive flexibility, suggesting that technologies can have different meanings for different groups. By engaging in customer discovery and stakeholder analysis, SCOT enables a deeper exploration of how these social groups perceive and value technology, ensuring that its development aligns with their concerns and priorities.

The social groups include computer science students, university educators, and software industry employers. Examining these distinct viewpoints aids in identifying specific shortcomings in the educational curriculum from an end-user perspective. The feedback from educators helps consider factors like historical dependencies, academic culture, or a philosophical preference for theoretical learning that may contribute to the persistence of the current system. Likewise, the software industry's stance on the readiness of graduates for practical challenges provides a crucial interpretation of the problem from an industrial vantage.

Applying SCOT, we can discern conflicts and commonalities within the narratives of these groups. As hypothetical examples, educational institutions could be promoting theoretical teaching considering it foundational for practical work, while students might perceive a lack of readiness for industry challenges. Conversely, employing industries might be seeking specific practical skills, leading them to pour substantial resources into training new hires. Analyzing such interactions and negotiations across groups facilitates a better understanding of dynamics that maintain the theory-centric status quo. This extended understanding, courtesy of SCOT, can then inform the design and integration of the proposed capstone course to synchronize it with the collective expectations of these stakeholders, thus transforming it from merely a confrontational problem to an opportunity for enhancing computer science education. Consequently, this research's broader implications transcend the theoretical exploration, offering practical solutions ensuring a more industry-aligned academia in computer science, and thereby bridging the prevalent disconnect observed today.

Academic sources, including books and articles published in educational, technological, and sociological journals, will serve as the foundation of my research. Primary data will be obtained through interviews with university lecturers and software professionals to garner a diverse set of data regarding expectations and experiences pertaining to computer science graduates. Instructors who have taught capstone courses will be questioned to understand their methodologies and gauge effectiveness. Likewise, software professionals will be interviewed to gain an understanding of which skills they deem necessary, which academia fails to provide.

This research hopes to understand the factors contributing to the disconnect in teaching and real-world expectations, thus facilitating the broader goal of a software industry-aligned computer science education policy. The insights obtained from the STS analysis will guide the design and development of the solution proposed in the technical project, allowing for a curriculum that effectively serves the needs of the CS graduate, academia, and the industry.

**Conclusion**

The interactive relationship between the technical and STS projects proposed here offers a comprehensive plan to address the sociotechnical challenge of bridging the gap between the academic computer science curricula and software engineering industry expectations.

The technical approach will manifest in the innovative design of an effective capstone course, providing in-depth practical skills to CS majors. This project aims to develop an immersive real-world software development experience, allowing computer science graduates to transition smoothly into the professional industry. The redesigned learning platform offers benefits to a plethora of stakeholders. It prepares students to instantly contribute when entering a professional environment and can reduce time and training investments from employers. Hence, the technical project contributes towards tackling the academia-industry gap from a pedagogical perspective, equipping students with the necessary practical abilities in tune with market demands.

The STS research, on the other hand, provides a holistic understanding of the underpinnings of the academia-industry disconnect in computer science education, leveraging the SCOT framework. This research does not end at an academic level but extends to practical application. Findings from the research could be insightful in shaping the practical capstone course and making it more aligned with industry standards. The insights from the investigation would inform the direction of course design and maintenance, ensuring it serves the needs of its key stakeholders.

Together, the two-fold approach, technical and STS, synergizes to comprehensively address the identified sociotechnical challenge. They cater to both the technical deficiency in current pedagogy and offer an extensive understanding of the sociocultural factors contributing to the stasis. The holistic approach ensures a nuanced solution addressing multiple dimensions of the problem, potent in making higher education in computer science more industry-relevant, thereby effectively narrowing the academia-industry gap.

**References**


Achee, B., Alkadi, G., & McNulty, M. S. (2022). Industry connect initiative experience report: undergraduate computer science courses. Journal of Systemics, Cybernetics and Informatics, 20(2), 1-7. https://doi.org/10.54808/jsci.20.02.1

Duch, B. J., Groh, S. E., & Allen, D. E. (2001). The power of problem-based learning. Routledge.

Hanna, S., Jaber, H., Almasalmeh, A., & Jaber, F. A. (2014). Reducing the gap between software engineering curricula and software industry in jordan. Journal of Software Engineering and Applications, 07(07), 602-616. https://doi.org/10.4236/jsea.2014.77056

Indriani Putri Salzabilla, Putri Astri Rangkuti, Tigor Halomoan Tambunan, & Feri Ferdian (2023). From industry to career: assessment of internship experience advantage for future career readiness in the hotel industry. Jurnal Pendidikan Dan Perhotelan (JPP), 3(1), 31-39. https://doi.org/10.21009/jppv3i1.04

Jarzabkowski, P. and Whittington, R. (2008). A strategy-as-practice approach to strategy research and education. Journal of Management Inquiry, 17(4), 282-286. https://doi.org/10.1177/1056492608318150

Johnson, D. (2005). Social Construction Of Technology., *Encyclopedia of Science, Technology, and Ethics.* (pp. 1791-1794). Macmillan Reference USA. https://books.google.com/books?id=MqMRAQAAMAAJ

Lent, R. W. and Brown, S. D. (2013). Social cognitive model of career self-management: toward a unifying view of adaptive career behavior across the life span.. Journal of Counseling Psychology, 60(4), 557-568. https://doi.org/10.1037/a0033446

Minichiello, A., Asghar, M. Z., Ewumi, E. R., Claiborn, C. S., & Adesope, O. A characterization of engineering and computer science undergraduate participation in high-impact educational practices at two western land-grant institutions. 2021 ASEE Virtual Annual Conference Content Access Proceedings. https://doi.org/10.18260/1-2--36557

Parsons, D. (2022). Agile and lean software engineering and the swebok - position paper for panel: educational and professional implications of swebok. https://doi.org/10.18293/seke2022-060

Patil, A., Chavan, P., Chatanya, S., & Patil, S. (2023). Enhancement in student employability by providing internship and project track. Journal of Engineering Education Transformations, 36(S2), 52-56. https://doi.org/10.16920/jeet/2023/v36is2/23008

University of Virginia. (2023, October 26). UVA CS Advising Guide. https://uvacsadvising.org/