WIP:Enhancing Career Preparedness Through a Software Engineering Capstone Course Design

Isabel Hyo Jung Song
Department of Computer Science
San Francisco State University
San Francisco, CA
hjsong@sfsu.edu

Jingyi Wang
Department of Computer Science
San Francisco State University
San Francisco, CA
jingyiwang@sfsu.edu

Rafael Sunico
Department of Computer Science
San Francisco State University
San Francisco, CA
rsunico@mail.sfsu.edu

Andrew Dahlstrom
Department of Computer Science
San Francisco State University
San Francisco, CA
adahlstrom@sfsu.edu

***Abstract*—This research to practice WIP paper describes capstone projects in software engineering which effectively combine theoretical education with practical skills, fostering career development in alignment with Career Construction Theory (CCT). This paper introduces a course designed to enhance students' career readiness by incorporating Agile methodologies for soft skills development, proficiency in modern technologies like Large Language Models (LLMs), and targeted career preparation such as resume building. The course's effectiveness, evaluated through CCT adaptability for 42 students, shows a positive impact on career preparedness in three of the four dimensions. This is the first attempt to measure the impact of a capstone course on career development using CCT adaptability. While the initial results are promising, further research is crucial to fully enhance all dimensions of CCT adaptability and to explore the scalability of this study across other engineering fields, potentially transforming engineering education and career preparation on a broader scale.**

Keywords—career development, software engineering education, capstone course, Career Construction Theory, Agile methodology, Large Language Model (LLM).

I. Introduction

Capstone projects in software engineering serve as an essential link between theoretical learning and practical application, potentially enhancing career development. Career Construction Theory (CCT) [1], a prominent psychological framework, combines vocational and developmental perspectives to explain how individuals shape their careers throughout their lives. While extensive research in engineering curriculum design supports the integration of real-world challenges and the cultivation of vital soft skills such as teamwork and leadership, the direct influence of such experiences on career readiness has not been extensively examined. Additionally, many public higher educational institutions face a scarcity of career counseling resources, often having only a few full-time career counselors available [2]. This shortage often limits discipline-specific career guidance for students. In such academic settings, engineering capstone courses could play a crucial role in advancing career development, although their effectiveness has not been fully explored within CCT.

In this paper, we hypothesize that a thoughtfully designed software engineering capstone course can significantly improve CS (Computer Science) students' career readiness. We develop a course around three core elements: enhancing soft skills through Agile methodology [3], building proficiency in modern, advanced technologies—including commercial software development tools and large language models (LLMs) [4]—and developing career resources, with a particular focus on aiding students in crafting resumes that effectively highlight their capstone projects. We implement this course in the CS department of a public state university, targeting undergraduate senior students. We assess students' career readiness, as defined by CCT, at the start and end of the semester. This assessment marks the first attempt to measure the course's impact on career development within real academic settings using CCT.

The rest of the paper is organized as follows. Section II examines previous studies in academic settings for CS and engineering students focusing on career development, as well as exploring the principles of CCT. Section III describes the proposed course in detail to test our hypothesis. Section IV evaluates the course, with a questionnaire-based evaluation tool designed around CCT principles. Section V concludes the paper and discusses potential future research directions to expand this work into other engineering fields.

II. Related Work

A. *Career Development for Computer Science and Engineering Students in Academic Settings*

The integration of essential career skills into various academic settings is becoming increasingly prevalent, particularly in the fields of CS and Engineering. These disciplines are directly linked to specific career opportunities that rely on the skills learned during academic training. In particular, many initiatives have been introduced to incorporate capstone courses, serving as a vital mechanism to bridge the gap between theoretical knowledge and practical experience, which is crucial for career readiness. An example of this is the Applied Software Engineering (ASE) Program [5] introduced by the CS department at Northern Kentucky University. This program is designed to prepare students for their professional careers by engaging faculty with industry expertise, aligning the curriculum with real-world industry demands, and helping students develop practical portfolios. While the program established criteria to measure its effectiveness for career development, including student-perceived career preparedness and employment rates, the data demonstrating its impact remains unpublished.

This project was partially funded by NSF (award number: 2330984).

Schibelius [6] proposed a capstone design course at the engineering school that emphasizes soft skills like communication and project management, critical to the workforce. The study has revealed that students find these skills challenging, highlighting the need for educators to enhance the design of capstone courses to better develop such skills. Quek [7] and Song [8] integrated real-world, team-based projects into the software engineering curriculum for CS students, utilizing the Agile process. Quek demonstrated the effectiveness of Agile process in enhancing teamwork. Similarly, Song implemented the Agile process in software engineering courses, which significantly increased students' satisfaction with the course. Additionally, Salmani [9] investigated how undergraduate research experiences in Electrical Engineering and CS impact student retention as well as their career and academic goals. Garcia [10] formulated mentoring strategies to aid in the academic development of Hispanic female freshmen, focusing on cultivating their career identity. The study revealed that through mentoring, 95% of the students recognized the significance of STEM fields as professional careers. Despite the implementation of Agile methodologies, real-world projects, research programs, and mentoring strategies—all linked to skills critical to the industry— there is still no concrete data demonstrating the specific impacts of these initiatives on various aspects of career development using benchmarks such as CCT.

In a non-academic context, Vicente [11] carried out a quasi-experimental study to evaluate the impact of internships on the career development of minoritized engineering students. Grounded in CCT, the study examined outcomes such as changes in vocational identity, company engagement, occupational interest, and confidence levels. Vicente conducted pre- and post-internship surveys with 33 students, and the findings indicated positive developments in all evaluated aspects following the internships.

While numerous initiatives aimed at enhancing students' career development have been introduced in academic settings, they lack quantitative evidence addressing various aspects of career progression among CS and Engineering students.

B. Career Constcution Theory (CCT)

The framework used to demonstrate the effectiveness of the course design is based on CCT. It explains the dynamic process wherein individuals build their career development through a series of meaningful vocational behaviors and work experiences. Considered as a self-regulating psychosocial ability, career adaptability [1] is a core element of this process and reflects the psychosocial capital of how people adapt to the interaction between their career and environment. Adaptability in CCT involves four key dimensions—concern, control, curiosity, and confidence. Each dimension plays a critical role:

- Concern reflects an individual's proactive planning and preparation for future career changes.
- Control represents the self-discipline, effort and persistence over one's vocational future, encouraging persistent effort towards career goals.
- Curiosity encourages exploration of various career possibilities and potential professional identities.
- Confidence is about self-efficacy and the belief in one's ability to achieve career aspirations.

Individuals who possess high career adaptability are able to swiftly alter their perspectives, recognize their strengths and interests, and actively seek beneficial advice, all of which are crucial for attaining career success. Consequently, career adaptability significantly enhances the career development process. However, in the realms of CS and engineering academia, the impact of courses has not yet been evaluated from the standpoint of CCT adaptability.

III. Proposed Course Design

To assess how the software engineering capstone course influences CCT adaptability, we have developed an in-depth course centered around three core areas: improving soft skills via Agile methodologies, gaining expertise in state-of-the-art technologies and tools such as LLMs, and integrating career resource development by crafting resumes that reflect the capstone project.

A. Agile Methodologies

The software engineering capstone curricula we developed is specifically structured to cultivate essential soft skills like communication, teamwork, critical thinking and time management — skills that are vital for success in the software industry but often inadequately addressed in traditional CS programs. This course integrated Agile methodologies, the industry-standard project management approach, which is known for enhancing interactive communication, peer feedback, and operational efficiency.

The course is structured around a semester-long group project, with teams of six students each adopting specific Agile roles such as product owner and scrum master. The product owner is tasked with constantly reviewing project deliverables to ensure they meet quality standards and deadlines. The scrum master facilitates communication within the team, organizes bi-weekly team meetings, and aims to keep everyone engaged towards common goals and milestones. In case of conflicts, the scrum master acts as the mediator. During scrum meetings, all members discuss their challenges and progress, exchange help, and collaboratively set future objectives. This Agile-based teamwork is pivotal for fostering communication and teamwork.

To cultivate critical thinking, each team is tasked with proposing their own project themes, gathering requirements and developing a web application that meets these needs. This challenges students to tackle not only technical issues but also identify meaningful problems and work with a strict deadline. The use of task management tools such as Notion [12] and Jira [13] is strongly encouraged to help students maintain transparency and efficiency, thereby enhancing time management skills.

The instructor plays a key role in promoting the exchange of best practices and technical insights among teams, thus enhancing peer-to-peer learning and better preparing students for professional settings. This collaborative educational approach equips students to thrive in the dynamic, team-oriented world of software development. Throughout the course, the instructor provides crucial support by teaching software engineering principles, offering mentorship, and sharing conflict resolution strategies for issues that teams cannot resolve on their own.

B. *State-of-the-art Technologies and Tools*

The course encompasses a wide range of software development topics, including requirement specification, UX design, software architecture, implementation, and testing. Each stage incorporates modern tools tailored to specific tasks: Figma [14] for UX design, UML [15] for architectural design, GitHub [16] for implementation, and both Jest [17] and GitHub Actions [16] for testing and continuous integration (CI) [18]. As the course progresses, students are instructed to use these tools to produce deliverables at various project milestones.

Recognizing the increasing significance of ML and LLMs in the industry, the curriculum includes specialized training on how these technologies can enhance project value. For example, during the requirement gathering phase, students investigate features like personalization, recommendations, or generative content—where LLMs are strong. During the implementation phase, they gain hands-on experience by implementing these features into their applications, utilizing ChatGPT APIs [19] to embed advanced functionalities. Incorporating ML components into the curriculum is designed to familiarize students with essential ML tools, equipping them to keep up with rapid industry developments and align their skills with career prospects.

C. *Career Resouce Development*

In the final session, the course features a component dedicated to resume building. It guides students on how to professionally present themselves, emphasizing both their technical skills and soft skills. This component stresses the importance of showcasing practical skills developed through academic projects, particularly the web application created during the course.

To support students who have limited experience in resume writing, we developed ML-based resume builder tool and introduced it to students. This tool offers suggestions for resume content based on students' free-form inputs, allowing students to concentrate on the content of resumes rather than the format or layout. This tool reduces the stress associated with learning resume formatting and supports the creation of effective resumes. Fig. 1 provides an overview of the resume builder tool, and Fig. 2 shows how it suggests resume bullet points (in grey) given the students' free-from inputs (in green).

Additionally, the final session of the course includes an analysis of the career trajectories of alumni, particularly focusing on those who secured jobs within six months after graduation, using publicly available alumni profiles. We examined their academic achievements, professional experiences, and extracurricular activities during their time in school. This information provides current students with valuable insights and inspiration for shaping their own career paths. This crucial component of the course aids students in navigating and preparing for the job market, equipping them with essential tools and knowledge for effective job searching.

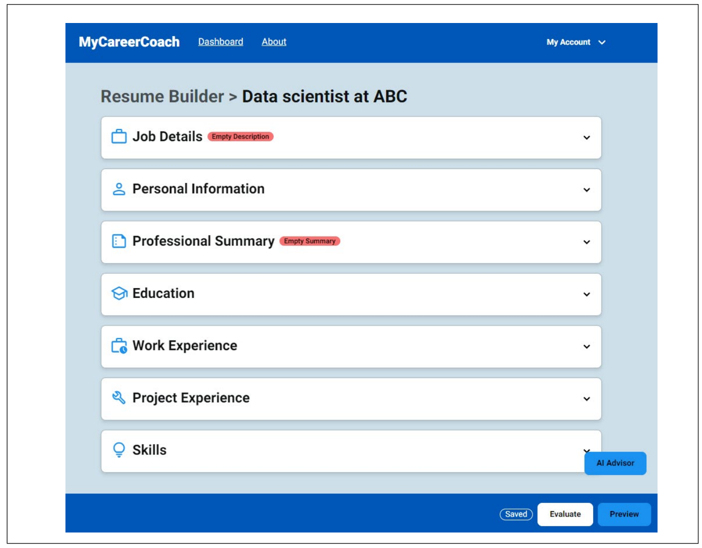

Fig. 1. Overview of Resume Builder.

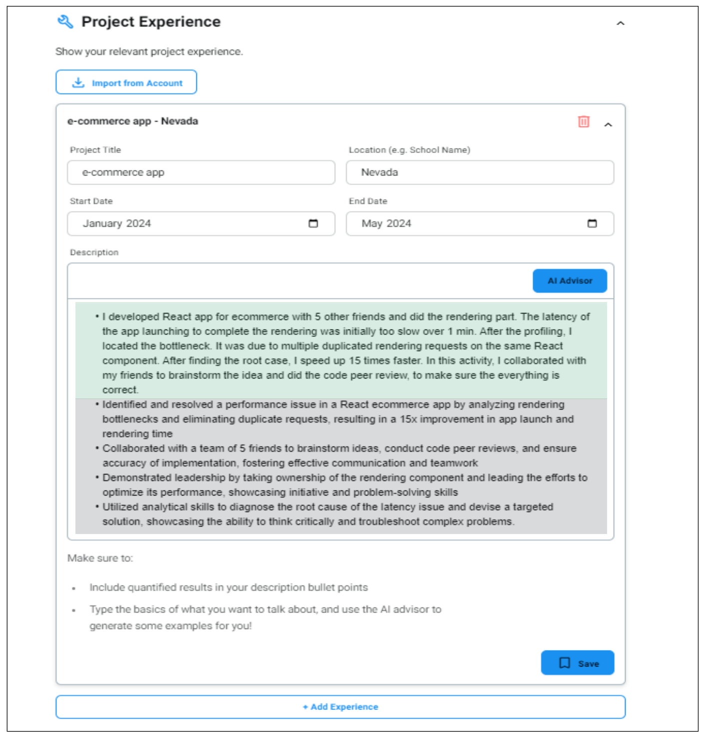

Fig. 2. Overview of Resume Builder.

IV. Evaluation

We implemented the software engineering capstone course as the culminating mandatory course in the undergraduate CS curriculum. This course predominantly attracts senior students who are nearing graduation and actively seeking employment.

To assess the effectiveness of the curriculum based on CCT adaptability—which help evaluate an individual's capability to handle career-related challenges—we implemented a survey with 25 questions. The evaluation framework focused on four key CCT adaptability dimensions: concern, control, curiosity, and confidence. The survey, 7 questions addressing concern, 6 on each for control, curiosity and confidence, was conducted at

TABLE I. REPRESENTATVE QUESTIONNAIRES FOR CAREER ADAPTABILITY DIMENSIONS

Dimension	*Questions*
concern	Q) I know what occupational path I want to pursue when I get out of school. Q) I could easily describe my ideal job to a recruiter.
control	Q) How comfortable are you making independent decisions about your career path, even if they differ from expectations of others? Q) How important is it for you to be able to take ownership of your career development and actively pursue opportunities for growth?
curiosity	Q) How interested are you in learning about different career paths and industries? Q) How often do you actively seek out new information and experiences in your professional life?
confidence	Q) How confident were you in your ability to understand the software development life cycle (SDLC)? Q) How confident are you in your ability to consistently deliver high-quality work, even under pressure?

the start and last week of the semester for 42 students, utilizing a 5- point scale from 1 (strongly disagree) to 5 (strongly agree). Only representative questions for each category are displayed in TABLE 1, due to space constraints.

Statistical analysis was performed using a paired-sample t-test [20] to identify significant changes in mean scores from the beginning to the end of the semester. The results, detailed in Table II, revealed improvements in the dimensions of concern, curiosity, and confidence, with mean increases of 0.33 (from 3.87 to 4.20), 0.30 (from 4.06 to 4.36), and 0.38 (from 3.86 to 4.24), respectively. The corresponding t-values were 1.49, 1.35, and 1.94, with p-values of 0.0, 0.008 (<0.01), and 0.0001 (<0.0005), respectively, indicating statistically significant improvements. These results suggest that the capstone software engineering course substantially enhanced students' career planning, exploration of career possibilities, and self-confidence. In contrast, the dimension of control did not show a statistically significant change, potentially due to relatively high initial mean scores (4.14) and possibly the impact of recent layoffs in the tech industry [21]. The most notable enhancement was seen in responses to the question "I know what occupational path I want to pursue when I get out of school," which increased from a mean of 3.71 to 4.46. This significant improvement likely stems from the curriculum’s emphasis on aligning with industry practices such as Agile methodologies and the state-of-the-art technologies.

V. CONCLUSION

This study introduces a distinctive capstone course in software engineering that combines industry-relevant project management techniques, cutting-edge technology, and a direct focus on resume building. Although many academic programs aim to enhance students' career development, they often lack quantitative evidence and impact analysis. This study addresses this gap by evaluating the course's effects on each of the four dimensions of CCT adaptability, providing an assessment that distinguishes it from prior work.

Feedback from evaluations indicated notable enhancements in students' career readiness, particularly in the dimensions of

TABLE II. ANALASIS FOR CAREER ADAPTABILITY DIMENSIONS

Dimension	*When*	*mean*	*Std. dev*	*t*	*P-value*
concern	Start	3.87	0.93	1.49	0.0
	Last	4.20	0.84		
control	Start	4.14	0.94	0.54	0.35
	Last	4.26	0.82		
curiosity	Start	4.06	1.04	1.35	0.008
	Last	4.36	0.78		
confidence	Start	3.86	0.89	1.94	0.0001
	Last	4.24	0.69		

concern, curiosity, and confidence. However, the control dimension did not show marked improvement, prompting further investigation into potential influences such as external conditions or gaps in skill training related to this dimension.

Future work will incorporate qualitative methods such as focus groups and interviews to gain deeper insights into these areas, aiming to further refine the curriculum. Once the findings from this study are clear, they can be used to refine other engineering curricula and establish a practical basis for various educational environments. Institutions can replicate and modify this course structure across different engineering and technology fields, tailoring it to local industry needs and student profiles to enhance career preparedness. Investigating the deployment of similar courses across diverse educational institutions is essential to confirm the scalability and adaptability of this study.

REFERENCES

[1] P. Chang, Y. Guo, and Q. Cai, “Proactive Career Orientation and Subjective Career Success: A Perspective of Career Construction Theory,” Behavioral Sciences, vol. 13, issue 6, 2023.

[2] Hiration, “Behind the Scenes: Top 7 Challenges & Insights for US Career Services in 2023,” https://www.hiration.com/blog/career-services-challenges/ (accessed May, 2024).

[3] Wikipedia, “Agile Software Development,” https://en.wikipedia.org/wiki/Agile_software_development (accessed 2024).

[4] Wikipedia, “Large Language Model,” https://en.wikipedia.org/wiki/Large_language_model (accessed 2024).

[5] Samuel Sungmin Cho, Nicholas Caporusso, Maureen Doyle, “NKU Applied Software Engineering Program: A Novel Approach to Software Engineering Education,” IEEE Frontiers in Education, 2023.

[6] Lisa Schibelius, Olivia Ryan, Susan Sajadi, “Student perceptions of teamwork, conflict, and industry preparedness in engineering interdisciplinary capstone design,” IEEE Frontiers in Education, 2023.

[7] S. Quek *et al.*, “Improving Teamwork in Software Engineering Projects in Higher Education,” IEEE Frontiers in Education, 2023

[8] Isable Hyo Jung Song and Cameron Paczek, “Work in progress: Agile Methodologies for Online Software Engineering Education under the Pandemic,” IEEE World Engineering Education Conference (EDUNINE), 2023

[9] Hassan Salmani, Mohsen Mosleh, Gloria Washington, “Undergraduate Research Experience Impact onRetention in an Electrical Engineering and Computer Science (EECS) program,” IEEE Frontiers in Education, 2023

[10] Victor M. Garcia *et al.*, “Understanding Career Identity Development and Preparedness of Freshmen Students to Leverage Convergence in Engineering Education,” IEEE Frontiers in Education, 2023

[11] S. Vicente *et al.,* “Exploring the Significance of Internship Experiences for the Career Development of Racially Minoritized Undergraduate Engineering Students,” IEEE Frontiers in Education, 2023

[12] Notion, “Notion,” https://www.notion.so/ (accessed May, 2024).

[13] Jira, “Jira official site,” https://www.atlassian.com/ (accessed May, 2024).

[14] [14] Figma, “Figma”, https://www.figma.com/ (accessed May, 2024).

[15] UML, “UML,” https://www.uml.org/ (accessed May, 2024).

[16] Github, “ Code With GitHub,” https://github.com/ (accessed May, 2024).

[17] Jest, “Jest,” https://jestjs.io/ (accessed May, 2024).

[18] Wikipedia,“Continuous integration,” https://en.wikipedia.org/wiki/Continuous_integration (accessed, 2024).

[19] OpenAI, “API reference – OpenAI API”, https://platform.openai.com/docs/api-reference (accessed May, 2024).

[20] IBM, “Paried-Sample T Test”, https://www.ibm.com/docs/en/spss-statistics/saas?topic=tests-paired-samples-t-test (accessed May, 2024).

[21] Techcrunch, “A comprehensive list of 2024 tech layoffs”, https://techcrunch.com/2024/05/07/tech-layoffs-2023-list/, (accessed May, 2024).