CS 4501:
Entrepreneurial Applications of Computer Science

CS4991 Capstone Report, 2023

Brandon Park
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
bjp4un@virginia.edu

ABSTRACT

While the University of Virginia's computer science department is ranked among the most recognized in the nation, one aspect that the university needs to refine is the real-world entrepreneurial potential of CS applications. The proposed course, CS 4501: Entrepreneurial Applications of Computer Science, aims to provide support on this end by introducing students with a different set of skills. The course would consist of three major components: 1) a semester-long project; 2) lectures regarding other tools businesses use to help software applications succeed; and 3) guest speakers invited to talk about their experiences with entrepreneurial endeavors. The proposed course would aim to provide students with a different perspective on how CS application-based businesses are run and enable them to gain insights to the reasoning behind decisions companies make. Further investigations targeting students will be needed to understand the effectiveness and interest levels of different aspects of the proposed course.

Keywords: Business, Computer Science, Entrepreneurship, Software Development

1. INTRODUCTION

The University of Virginia (UVA) has a highly ranked engineering program, supported by a well experienced faculty, staff and opportunities that allow students to thrive. Due to the transition to online classes during the Covid-19 pandemic and recent rollbacks to normal face-to-face courses, universities throughout the nation are currently adjusting to a new norm as expectations within the classrooms have changed. However, while the media in recent years have touted the university having one of the best programs, both in online and offline experiences, the reality is that the program is facing mixed reviews.

The department of computer science (CS) is currently going through a massive curriculum overhaul. Issues such as the inability to scale up to offer necessary courses to the ever-increasing number of computer science majors and limitations in course diversity means that many students are faced with limited options. They face problems in gaining knowledge that is applicable in the real world, rather than theories that most typical CS programs like to focus on. Real-world applications that students can potentially partake in, creating their own solutions and taking an entrepreneurial route often gets glanced over. UVA is currently sitting on a wealth of entrepreneurial and business knowledge in the form of the McIntire School of Commerce and the Darden School of Business. With knowledge from courses within the school of business and commerce, the potential opportunities that students can make for themselves can be practically endless. It is possible that given the right resources, students have ideas that can succeed. However, many CS students currently lack crucial business knowledge skills needed to support these projects. Currently, the CS department fails to provide adequate education in company management and business decision-making. Restricting these introductory business courses to business and commerce students limit options for CS students to learn this information. This inadequacy needs to be addressed.

2. RELATED WORKS

Efforts to bridge the gap between knowledge learned within the CS department and the real world has been done before. Koehler (2023) suggests integrating computer science electives into the current curriculum to improve the knowledge pool that CS students would have once they graduate [2]. While the proposal is solid, taking away student choice in electives in an already strict degree will result in a lack of diversity in the type of students produced. Having concentrations that focus on different topics and providing options for course requirements may be an alternative.

Cheng, et. al. (2021) focused on issues CS students face, particularly during interviews, by offering a course that prepares students for technical and behavioral interviews that many companies conduct during the recruitment process [1]. Cheng focuses on the recruitment process and improving a student's chances of succeeding in the interview process, However, the proposed course aims to help students understand how companies are run in the macro-scale and how this information could be applied in a self-driven, entrepreneurial environment.

3. PROPOSED CURRICULUM

The current CS curriculum lacks any offerings in business or commerce knowledge. The course that I propose seeks to fill that gap, offering a chance for CS students to learn more about both fields. There are some course offerings like Internet Scale Applications and Advanced Software Development Techniques, that offer some means to learn how ideas are built into pieces of software and the different factors that go into the decision making process of what technology is being used.

The proposed course will focus more on the creative nature of both courses, allowing students to decide upon what idea they would like to develop from the ground up. Rather than replace the two courses mentioned above, the following course will serve as an addition to the knowledge base of CS students and allow students to apply their CS knowledge on a more real world application that they seek to build upon. I propose a course comprising three major components: An organized set of business-related lectures, a semester-long project, and guest speakers. The following design is based on three major components crucial to course design: learning outcomes, learning activities and learning assessments [4]. The course, with backings from research done in regards to effective course design, should serve to add to the knowledge base of CS students and allow them to utilize this knowledge in real-world applications and situations.

3.1 SEMESTER-LONG PROJECT

The bulk of the course will be placed within this semester-long project, as students go through a process similar to what many pre-startup and startup companies go through. Students would explain their idea to faculty, presenting what the problem they aim to solve is and how their solution differentiates them from the competition. It's possible that certain ideas will be too grand to achieve any significant research within the semester. As such, one goal that this step aims to achieve is to ensure that what students plan to do can be accomplished within the semester. Special exceptions for larger projects will be made for students and teams that have already made prior significant progress.

Once project teams and ideas have been approved, student teams would start researching their market, either through collecting relevant information or through interviews from potential clients, and collect their findings to be turned in as an assignment. A mock business plan of the company will also be created alongside, specifying ideas like how the company may scale, how they intend to turn a profit and the general structure of the organization in the early stages of its creation. Thereafter, students would now begin creating their software solutions, going through the agile development cycle and utilizing crucial technologies, such as git for version control. In the end, students would present their final product to faculty and fellow classmates, framing this as a pitch to prospective investors for their created software solution. Students will be graded on some criteria such as: How convincing students are during their presentation, the quality of their final product and how detailed their plans for the organizational structure are.

3.2 BUSINESS RELATED LECTURES

The lectures will serve to supplement the previously described semester-long project , focusing on crucial knowledge that is important in understanding key business functions within software development companies, such as how companies make decisions, the ethics behind certain processes and how to market a product. Each week would introduce a different topic, specifically a different department that would exist within companies, that directly or indirectly influence the software development cycle.

The first few weeks would start with information that would aid in helping students understand the market for a given idea and what key players exist within the field they are interested in. Methods of conducting market research on a simple scale, and ways to find specific target markets will be introduced, with students having a chance to exemplify their understanding through in-class activities.

Once they have built an understanding of market research, topics related to the full software development process, such as agile development methodology, minimum viable product and testing will be introduced in the following weeks. Starting at this point, students will also receive knowledge on different languages and libraries that they can use on their projects. Common tools, such as bootstrap and HTML/CSS, are some examples with others to follow.

The final lecture topics would focus on providing students with supplementary knowledge for their semester-long project pitches. Product marketing and pitching to potential investors are some topics that would be covered, with the main focus of these last lectures being how to gain attention from the public in order to be successful when their supposed product would launch. Chances during class will be provided to practice with others before they proceed during their final pitches for their projects.

3.3 GUEST SPEAKERS

The final element of the proposed course would be to invite guest speakers to share their own experiences with entrepreneurial endeavors and what they have learned. The main goal is not to teach students new technologies or frameworks that they can use. Rather, speakers would be there to provide some context as to how startups actually come to be and allow students an opportunity to interact with the speakers. The following talks will be formatted like a keynote speech, allowing guest speakers to first share their general accomplishments in their careers and answer a list of important questions that faculty will prepare ahead of time. It will be concluded with a Q&A session, allowing for students to ask questions for themselves.

4. ANTICIPATED RESULTS

The aim of this course is to provide students with general knowledge of how someone can take an idea for a software solution and take an entrepreneurial route with it by introducing the beginning steps of what many software startups go through. Each portion of the curriculum seeks to provide a basis that students can use to start their own companies, or better understand the factors that influence the decision making process that companies have to go through. Understanding this process will allow students to better cater their interviews to potential workplaces, as they better understand the needs and wants of these companies. Other students may be able to leverage this course to put significant effort into a software idea that they previously had, allowing them to use this course to develop and sharpen their solution.

Another aspect that this course seeks to improve upon in the current CS curriculum is providing means to be creative in an otherwise theory heavy department. Many courses are relegated to teaching conceptual ideas. While it may prove to serve useful in understanding the inner workings of software, it fails to provide students with a creative outlet to try out new things and build upon their own ideas. Providing means for students to try and implement their own ideas into working software provides the opportunity to not only have a creative outlet in an otherwise theory-heavy major, but it also allows students to view software from a different point of view. Software is often seen as a product, but seeing software as the main driving point for a company opens up a new way to think about things, especially when it comes to software development.

5. CONCLUSION

The proposed course seeks to fill in the gap that the current CS curriculum at UVA fails to address by providing students with means to learn business knowledge specifically targeting software development companies. Understanding the key factors that influence the development cycle and going through the beginning stages of how a software implementation becomes a company can be crucial, especially for those that intend to further improve upon their code base.

6. FUTURE WORK

Interviews with students that take this course must be conducted to further understand its benefits and shortcomings. While our intention is for students to take this course and better understand the factors that influence the software development cycle, it would be beneficial to look into how different students have used this information and whether or not this course is viable to add to the current CS curriculum.

Another point to consider is what kind of overlaps exist within the CS curriculum itself and ensure that taking this course will not result in redundancy. Current ongoing revisions of the CS curriculum suggests that there are indeed redundancies amongst courses[3]. Courses, such as CS 3240: Advanced Software Development and CS 4260: Internet Scale Applications, already cover similar topics. Thus, understanding the similarities and differences between such courses and the following proposed course is necessary in order to determine which topics may need more coverage and ensure the course is effective in its purpose.

The last point to look into is the method of inviting guest speakers for the proposed course, which the current course design relies heavily on the connections of the current faculty of the CS department. It is totally plausible that many individuals within the department have gone down similar paths, resulting in similar connections and individuals being invited for the guest speaker sessions. The entire goal of the guest speakers is to gain differing perspectives, which makes it innately difficult to find a variety of different people willing to present. Later attempts at this course may look to keep in contact with past students from the university that have had and would be willing to share interesting experiences they have gone through.

7. REFERENCES

[1] Cheng, E. (2021). A New CS Curriculum: Bridging the Gap Between In-Class Assignments and Real World Problems; Exploration of Online Learning Integration. Charlottesville, VA: University of Virginia, School of Engineering and Applied Science, BS (Bachelor of Science), 2021. Retrieved from https://doi.org/10.18130/jmjx-tq54

[2] Koehler, M. (2023). Improving CS Education at the University of Virginia by Integrating Electives; Efforts to Successfully Add Ethics to CS Education. Charlottesville, VA: University of Virginia, School of Engineering and Applied Science, BS (Bachelor of Science), 2023. Retrieved from https://doi.org/10.18130/80q0-ab07

[3]Sherriff, M., & Horton, T. (2023, March 9). CS Undergraduate curriculum transition plan. University of Virginia School of Engineering and Applied Science. https://engineering.virginia.edu/cs-undergraduate-curriculum-transition-plan

[4] Whetten, D. A. (2007). Principles of Effective Course Design: What I Wish I Had Known About Learning-Centered Teaching 30 Years Ago. Journal of Management Education, 31(3), 339-357. https://doi.org/10.1177/1052562906298445