Software Development Preparation: Evaluating the CS 3240 (Advanced Software Development) Curriculum to Industry

CS4991 Capstone Report, 2024

Benny Bigler-Wang
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
brb9ths@virginia.edu

ABSTRACT

As currently designed, UVA’s CS 3240 Advanced Software Development course uses the Django framework instead of more popular frameworks in industry. By the end of the course, students have learned how to use many different third-party technologies and frameworks but have not learned ones that may be more relevant to today's standards. My proposal is to consider adding a new requirement to the semester-long project that incorporates a relatively new AWS tool to better prepare students and adjust to current industry standards. In the future, continuous assessment will be needed to reevaluate the efficacy of the revised CS 3240 curriculum to ensure that it is preparing students to use the ever evolving technology being introduced into the industry.

1. INTRODUCTION

The technology realm constantly evolves as new products/technologies are introduced to the market. With this rise in new technology comes a rise in the need for skilled software developers. The preparation of students for successful careers in the field of software development requires an adaptive curriculum that allows them to adapt to industry trends and requirements. One cannot solely focus on software development in the curriculum as there are other crucial subfields such as discrete mathematics, computer architecture, and operating systems among others that introduce new fields in computer science even though they may not be the most useful to know in the context of software development.

An assessment of the CS3240 Advanced Software Development course when I took it in 2022 and the two years I have been a teaching assistant for the course have allowed me to understand the type of knowledge and skills learned from the course and how the course has changed over the years. Evaluating the computer science learning experience within CS 3240, my proposal addresses the challenges encountered in preparing students for real-world software development careers.

2. RELATED WORKS

The field of computer science education has seen significant advancements in recent years, driven by the need to bridge the gap between academic learning and industry demands. Lawrence-Fowler (2015) highlights the importance of integrating practical, hands-on experiences into computer science curricula to enhance students' readiness for professional roles. This research emphasized the significance of incorporating industry-relevant technology into course content to ensure its relevance and effectiveness. They also discussed strategies implemented in a computer science program to address the issue of student preparation for the workplace of the 21st century

Webb, et. al. (2017) showcases an analysis of various computer science curricula, showing a disparity in the approaches and teaching methodologies but also highlighting the specific issues and challenges of these curriculums. These studies underscore the importance of periodically reassessing and updating the curriculum to ensure alignment with the evolving needs of the software development field.

3. PROPOSED DESIGN

As with any computer science course, understanding the curriculum and its underlying structure is paramount for students to navigate through the class effectively. In my experience with CS 3240, I was tasked with delving deep into the intricacies of software engineering principles and practices. This was done through both theoretical and application-based means.

3.1 Review of CS 3240 Curriculum

CS 3240's curriculum is designed around modern software engineering methodologies and tools. In the course, students work on a semester-long web application project in Python using the Django Framework. Students also attend lectures and have other assignments (i.e., Guided Practices) to learn theoretical concepts that will later be applied to the semester-long project. The core concepts that are emphasized throughout the course are:

- Software quality, including testing and inspections
- Project management, scheduling, planning, with an emphasis on the Scrum agile method
- Requirements elicitation, analysis and specification
- Architecture and design principles
- Security in the development of software applications
- Programming and team-based development practices
- Professional ethics

Each of these concepts plays a vital role in shaping the skill set of aspiring software engineers. Each concept is applied in different steps to the semester-long project, roughly divided up on a bi-weekly basis. The team is organized through the use of SCRUM, an agile methodology that a lot of technology companies use to manage smaller teams, which meet on a weekly basis to discuss and plan the development of their project for the upcoming week. Through this process, students are expected to apply the more nuanced and theoretical topics they have learned in class to their actual development.

3.2 Key Technological Components

Just as in the development of software products, understanding the key technological components of a course curriculum is essential for students to grasp the underlying concepts and their practical implications. In CS 3240, various tools and technologies form the backbone of the learning experience, contributing to the development of different parts of the web application. The current technologies used during the semester-long project as of Spring 2024 are: language used for development (Python 3), database system (PostgreSQL), framework (Django 5), version control system (Git), build environment (GitHub Actions CI), source control management (GitHub), authentication system (Google

Login API), cloud hosting (Heroku), and cloud storage (Amazon S3).

This set of technologies mostly remains the same but are updated slightly from semester-to-semester. Recent changes to the curriculum, such as using Amazon S3 for cloud storage, show that the curriculum is being updated to adapt to new industry technologies. Large changes in the curriculum may be too difficult or require too long to institute between semesters, which is why small incremental changes to the project guidelines have been made each semester. However, the curriculum is slowly falling behind the current software development landscape, even when adapting from semester-to-semester, so additional effort to remain up to date is needed.

3.3 Challenges and Solutions

The transition to a new curriculum that aligns with current computer science industry standards presents a multitude of challenges, each requiring careful consideration and strategic planning. The key issue with the CS3240 course is that it uses the Django framework, which is not commonly used in industry to develop enterprise software. The course could switch to using another more popular framework such as React or Spring but they use a different programming language: JavaScript and Java respectively. Javascript is widely used in real-world development and is not taught in any classes in the CS curriculum, therefore, switching to React may be too difficult to implement realistically in a semester-long class. Spring is based on Java which has been taught in a previous class, Software Development Methods, but faces similar problems to Django. Django is consequently a good framework to stick with because it uses Python, a very popular programming language, and it has an extensive documentation and resources online to help students in their development. It has also been taught in a prerequisite course CS 1110, so students are more familiar writing code in the language.

Another change that could be considered is incorporating new third-party technologies such as new AWS tools to accomplish specific tasks. This would call for new feature requirements that would need, for instance, certain AWS products such as load balancers, simple queue services, lambda functions, databases. Although there are already a lot of features required for the project, I believe the addition of implementing one such AWS product into the web application would give students better experience and more preparation for industry work.

4. ANTICIPATED RESULTS

Adopting the proposed adjustment into the CS 3240 curriculum will greatly benefit CS students, the software development industry, and UVA as whole. Students will be better prepared for future work in internships and jobs. They will also have a more relevant project to talk about in interviews which will increase their likelihood of getting roles since they have previous experience using technology that is used in the real world. Companies will have, as a result, a better-trained employee pool that will allow them to develop better products. Last, UVA will boost its reputation as an institution that provides a quality, cutting-edge education for its CS students which will attract more attention, more applicants to the program, and potentially more funding in the future.

5. CONCLUSION

The evaluation of the computer science learning experience with CS 3240 Advanced Software Development shows that even though the course does a good job of introducing key software development principles it is slowly lagging behind in preparing students for real-world software development careers. The use of the Django framework for the development of the semester-long project is currently the best option, but it may not be a good choice for giving students industry-level software development experience. Its substitution for a more popular framework such as React, should be evaluated in order to ensure an education that best prepares students for their future careers.

As of now, incorporating new technologies such as an advanced AWS tool could give more relevant experience and enhance students' readiness for the evolving industry landscape. This adjustment in the project requirements could lead to benefits for the students, CS employers, and UVA. Ultimately, a flexible and adaptive curriculum that incorporates emerging technologies and industry practices is crucial in order to equip students with the most relevant skills to succeed in the software development field.

6. FUTURE WORK

Future updates to the CS 3240 curriculum should be considered in order to enhance student learning and proficiency in modern software engineering principles. Considering integrating more relevant frameworks such as React, Spring or other frameworks should be reevaluated annually in order to best prepare students. As of now, the best reason to remain with using the Django framework is that it provides a nice balance between using previously-learned technologies and introducing students to full-stack development.

Incorporating an additional feature that utilizes an AWS tool such as a load balancer to help distribute incoming traffic may help prepare students to interact with external APIs and third-party technologies. Third-party tools are already incorporated into other aspects of the project such as the Google login API and an AWS S3 bucket, but at a very surface level.

REFERENCES

Lawrence-Fowler, W. A., Grabowski, L. M., & Reilly, C. F. (2015, October). Bridging the divide: Strategies for college to career readiness in computer science. In *2015 IEEE Frontiers in Education Conference (FIE)* (pp. 1-8). IEEE.

Webb, M., Bell, T., Davis, N., Katz, Y. J., Reynolds, N., Chambers, D. P. & Mori, N. (2017). Computer science in the school curriculum: Issues and challenges. In *Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing: 11th IFIP TC 3 World Conference on Computers in Education, WCCE 2017, Dublin, Ireland, July 3-6, 2017, Revised Selected Papers 11* (pp. 421-431). Springer International Publishing.