

Educating New Software Engineers

Software engineering is an ever-evolving field that demands a significant amount of self-study and a genuine passion for the subject matter. Consequently, students often find themselves inadequately prepared to transition into the workforce. Students are expected to learn emerging technologies such as MongoDB and Angular JS. Universities primarily emphasize the academic aspects and theoretical foundations of software engineering, often overlooking the practical applications of this field. To adequately equip students, universities should consider implementing interview preparation courses and placing greater importance on the development of production code. This structure enables students to understand the fundamentals of programming while allowing them a smoother transition to the job market. Additionally, a greater emphasis on the interview process needs to be established by institutions to ensure students' success. Internships are critical to the success of Computer Science students, allowing for simulation of the job process, because of this it is critical to promote a balanced curriculum that mixes theoretical knowledge with practical abilities to prepare students for the evolving terrain of software engineering.

Many proponents of the current structure of the computer science degree argue that it aims to provide a fundamental basis for its applications, such as Software engineering. Arguing that understanding the theory, as well as the lowest level application, is critical to understanding the practical applications of programming. At Florida State University, languages such as Python are taught following a concrete foundation within C, C++, as well as assembly (FSU 2023). Python consists of high levels of abstraction, seemingly working like magic compared to predecessors such as C (Frampton 2009). Developing an understanding of lower-level languages allows for the layers of abstraction to be removed, providing a better understanding of the inner workings of the languages. By learning the fundamentals of programming other frameworks become easier to understand. Learning new technologies is difficult, even for experienced programmers. After utilizing one language consistently other languages tend to become muddled, especially when syntax is vastly different. One study in 2022, found that experienced programmers tend to maintain their old habits within a new language. When observing a programmer transitioning from R to Python they found that “They are unable to successfully relate their previous knowledge about subsetting, in R, to Python” (Shrestha 2022). To properly prepare for the job process, universities should be instructing students on relevant frameworks and materials to avoid this confusion between languages. Courses need to be constructed centered around emerging technologies so that students may establish a concrete foundation within their self-study. Learning emerging languages may be difficult with the variety of different methods of programming, by teaching these technologies within institutions a concrete standard will be set for students. This allows students to further their concrete foundation while maintaining a common standard for the technologies.

To properly prepare students, universities should prioritize learning emerging technologies such as MongoDB. Learning the fundamentals of computer science is important, however, they leave a steep learning curve for students while entering the job market. Consequently, computer science internships are rarely granted to students, with one study reporting less than 60 percent of Computer Science students pursue internships before

graduation (Kapoor 2020). Internships are often pivotal to learning emerging technologies. Internships prepare students for the creation of production code through rigorous training. However, due to the shortage of students achieving internships, students are left without any formal experience when entering the job market. Internships provide a training course for students who are not otherwise satisfied with their degree (Jaradat 2017). Internships allow students to shadow, as well as work on production code environments which are often not properly simulated within the computer science degree. Computer Science programs should therefore place a higher emphasis on preparing students for their internships rather than only approaching the theory behind the subject matter. Many courses such as Code Camp, as offered by Florida State University, provide training for internship interviews. Courses such as code camp should be emphasized within a computer science degree. Practicing for interviews has been shown to significantly increase performance, by approximately 16 percentile points (Williams 2008). The interview process within the field of computer science, with 98% of surveyed employers reporting that the technical interview process was the most important factor in considering a new hire. However technical interviews are often not reflective of the skills needed to succeed in the job (Tabrez 2019). Many universities fail to offer courses about any relevant emerging technology that may assist students within their workload. At Florida State University, there are a few classes centered around emerging technology from the last 5 years. CAP4601 is taught in LISP, released in 1959, a declining technology rather than the industry standards such as Python. The failure of many universities to update their programs has further developed the need for self-study among computer science students. In a survey of 100 successful software engineers, 95% reported that they self-studied for 10 to 15 hours per week. The study also concluded that in highly specialized fields, such as computer science, self-study is extremely important (Weimer 2005). To succeed in software engineering students are required to self-study because of the diverse required skill sets.

While much of the path to becoming a software engineer is filled with self-study, many universities have resources and clubs that may assist students. One such program at Florida State University is the ACM Programming Club. The club offers numerous lessons every week, diving further into the intricacies of programming. Clubs offer valuable insight into the software engineering process because of the exposure to new challenges and opportunities. Hackathons hosted by clubs, such as the ACM programming club, offer various methods of project-based learning. Within a Hackathon, contestants are expected to provide a solution to various challenges from a diverse topic. This allows aspiring software engineers to learn from their peers and build cohesive software within the allotted time. A Hackathon effectively simulates the skills needed for developing production code. Additionally, Project-based learning is one of the most effective ways to learn a new technology or to improve students' knowledge of their chosen technology (Rahman 2023). By emphasizing project-based learning students are granted a more concrete understanding of the material. Similarly, problem-based learning places a greater emphasis on changing the behavior of students rather than mastering the concepts (Richardson 2011). Both problem and project-based learning offer valuable insight into the challenges of software engineering to students. Programming clubs additionally offer numerous resources to prepare for technical interviews. This enables students to better learn and understand many of the problems they may face within the job market. One study found that. The ACM programming club also offers programming challenges that prepare students for technical interviews. Within these programming challenges, students are encouraged to complete problems faster than their peers, in a similar manner to learn code-style problems. To properly prepare students Florida State University should consider implementing courses centered around the development of production code. Allowing students to develop a large cohesive project, and properly understand their contributions can help ensure a student's success within the job market. Florida State University requires CEN4090L, a course centered

around the development of one project for the entire semester. However, the course fails to offer students an insight into the development process apart from self-study. Students are expected to learn with their group through their prior knowledge to develop software. The current direction of computer science departments inadequately prepares students for working on large projects since their coursework is largely based on small assignments. A greater emphasis needs to be placed on creating large multi-stage projects rather so that students may better understand the vision of their smaller assignments. Florida State University should require classes that reflect group work as well as large projects to properly reflect their future working environments, these changes will assist students in their employment endeavors. These courses would additionally provide students with opportunities to improve their resumes, through large projects. These courses would help ensure students' success, allowing for increased opportunities to secure internships as well as future job offers.

Recognizing the value of internships and project-based learning, colleges should aggressively encourage students to engage in real-world situations. The gap between academic understanding and industry expectations can be addressed by the provision of tools such as programming clubs, hackathons, and updated coursework that is aligned with current industry standards. To properly prepare students, universities should place a higher emphasis on projects to simulate future working environments. In this way, institutions may empower students to not only understand the theory but also successfully implement their knowledge, ensuring a smoother transfer from academics to the professional arena. A well-rounded and forward-thinking curriculum will better educate the next generation of software engineers for success in a rapidly changing technological context. In conclusion, software engineering is an evolving discipline, thus curricula must be well-balanced to include both theoretical and hands-on abilities. Teaching emerging technologies, encouraging internships, and improving project-based learning should be university priorities. This method guarantees that students are ready for the constantly shifting requirements of the labor market.

Work Cited

Amanpreet Kapoor Computer & Info. Science & Engineering, et al. "Barriers to Securing Industry Internships in Computing: Proceedings of the Twenty-Second Australasian Computing Education Conference." *ACM Other Conferences*, 1 Feb. 2020, dl.acm.org/doi/abs/10.1145/3373165.3373181.

Author links open overlay panelGhaith M. Jaradat, et al. "Internship Training in Computer Science: Exploring Student Satisfaction Levels." *Evaluation and Program Planning*, Pergamon, 19 Apr. 2017, www.sciencedirect.com/science/article/abs/pii/S0149718916303275.

Computer Science Grad Student Job Application & Interview Guide, www.cs.cmu.edu/~clegoues/grad-job-guide.html. Accessed 20 Nov. 2023.

Effects of Practice and Feedback on Interview Performance, tigerprints.clemson.edu/cgi/viewcontent.cgi?article=1477&context=all_theses. Accessed 20 Nov. 2023.

Demystifying Magic: High-Level Low-Level Programming - Researchgate, www.researchgate.net/publication/221137901_Demystifying_magic_High-level_low-level_programming. Accessed 11 Nov. 2023.

Educating Software Engineering Managers - Semantic Scholar, www.semanticscholar.org/paper/Educating-Software-Engineering-Managers-Revisited-Peters-Moreno/ba6fa3763fbec2066ecc4d2f65de4f9b77b1de88. Accessed 11 Nov. 2023.

Guide, Academic Program. "Academic Program Guide." *FSU*, undergrad1.its.fsu.edu/academic_guide/map-display.php?program=computer-science-bs. Accessed 10 Nov. 2023.

Nischal Shrestha, Colton Botta. "Here We Go Again: Why Is It Difficult for Developers to Learn Another Programming Language?" *ACM*, 1 Mar. 2022, cacm.acm.org/magazines/2022/3/258915-here-we-go-again/fulltext.

Tabrez, Miraan. "Why Coding Interviews Still Suck." *Medium*, HackerNoon.com, 24 Jan. 2021, medium.com/hackernoon/why-coding-interviews-still-suck-aa189e3b6c93.

University, Tajmilur Rahman Gannon, et al. "Project Based Learning: Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2." *ACM Conferences*, 1 Mar. 2023, dl.acm.org/doi/10.1145/3545947.3576331.