Project Overview Document

By: Info Innovators

Team Introduction

Our project is sponsored by Boeing and consists of the following team members:

- **Sirena Akopyan: Product Manager**
- **Mari Woodworth: UX Design/Researcher**
- **Eric Kim: Software Developer**
- **Brandon Mendoza: Data Scientist**
- **Bhavya Garlapati: Cyber Security Analyst**

Problem Context

The problem context surrounding our project is centered on a critical gap in the education of students in software and data engineering, specifically regarding DevSecOps and CI/CD principles. Despite learning to code, students are not sufficiently taught the best practices of software development or the essential industry tools needed for deploying code into production, especially using cloud technologies. This educational shortfall results in significant challenges when these students enter the industry, as they are confronted with a variety of unfamiliar software, programming languages, and concepts. Moreover, the support available to them within their new roles is often inadequate, leaving them to spend considerable time and money reviewing and learning independently rather than applying their skills effectively.

This issue is exacerbated by the fact that interactive courses on cloud technologies, including DevSecOps and CI/CD, tend to be costly, ranging from $60 to over $1000, which places them out of reach for many students. Additionally, the courses that are available in colleges and universities are often too limited, offered only occasionally by guest lecturers, or restricted to specific student groups within computer science or informatics departments. These courses rarely provide the hands-on, project-based learning experiences that are crucial for truly understanding and applying these technologies in a real-world context. Addressing this problem is vital not only for improving the readiness of new graduates to contribute effectively in their future workplaces but also for closing the existing skills gap in the rapidly evolving tech industry.

Problem Statement

How might college students in software and data engineering be better informed about DevSecOps and CI/CD principles so that they can effectively utilize and apply those concepts and technologies into production within industry?

Key Research Insights

We first defined our research questions for initial research:

- How do students currently learn about cloud technologies, if at all, during their coursework at university?
- What current resources are available to learn more about cloud technologies, and how can we improve them?
- How can we incorporate concepts into a pre-existing course curriculum so that professors can easily integrate it into their classwork?

To investigate these questions, we conducted a survey to assess UW students' knowledge of DevSecOps and CI/CD concepts and tools. We received 24 survey responses, 12 from Informatics students and 12 from computer science students at UW. Additionally, we interviewed students to delve deeper on what aspects of these concepts they were familiar with, any classes they have taken that were similar, and their learning styles. Interviewees were seniors in Informatics studying data science or software engineering, an Informatics graduate working in cybersecurity at Accenture, and a 2nd-year computer science student.

Major issues were brought to light from the research including lack of class availability, little access to resources, awareness of its importance, and lack of term familiarity.

Lack of Class Availability:

- **Lack of classes on DevSecOps / security** in general
    - *"We still are in an environment where security is very new and people still don't want to adopt it, so in terms of classes,* ***classes don't really put an***

***emphasis on security or just brush over DevSecOps** when teaching students how to program or when teaching them a new technical skill"*

- Lack of classes explicitly teaching CI/CD, **mostly general**
  - *"Well, I feel like there aren't any classes that are tailored toward using CI/CD, and likewise for DevSecOps exclusively as far as I know."*
- Classes are **far and few and difficult to get**
  - Restricted to **majors only for INFO courses**, making it difficult for nonmajors to take INFO classes
  - Didn't have time to take them as they **were not a requirement for the major / weren't interested**
- *"Wasn't really required in core curriculum - personally I just didn't learn about the fact that those classes existed until I was well into my junior year - then at that point - I **don't really have that much time** to take more classes"*

Access to Resources:

- External resources were used **if interested**
  - YouTube, Reddit, LinkedIn, Google
  - Tryhackme, hackthebox are more cybersecurity focused
  - *"The whole process can be **confusing** but it is mainly because there **aren't enough resources** to teach that stuff"*
  - *"Coursera / Udemy - very do this, but **not do it well**"*
- Learned more about these concepts in a **general sense through internships -** had to **relearn** core principles / best practices

Term Familiarity / Importance:

- *"I kinda understand the practices associated with them but never heard those specific terms in particular"*
- *"I feel like my major (Informatics) **does a poor job of preparing aspiring professionals to go into the tech world**, especially related to development. There are **very few classes** that I can name that teach these core concepts for DevSecOps, cloud computing/software, and CI/CD. We often have to **supplement** it with outside resources, which take up a lot of time and aren't as nearly effective as taking them in a classroom setting."*
- *"[Not teaching them is leading to a new wave of programmers / devs going out] into industry and **creating more vulnerabilities** (not teaching DevSecOps)"*

The comprehensive research highlighted a significant gap in the academic preparation of students in the fields of software engineering, data science, and cybersecurity, specifically regarding DevSecOps and CI/CD principles. Insights reveal that while 70% of students are aware of DevSecOps, a vast majority have never actively used it, and only 35% have some experience with CI/CD, yet lack proficiency. This disconnect is exacerbated by the absence of regular, dedicated coursework on these topics; 83% of students reported that relevant classes are either unavailable or not offered consistently, which is particularly problematic as 92% of respondents recognized the importance of integrating these skills into academic curricula.

Detailed research insights can be found at this link: https://www.notion.so/sirenaakopyan/Research-Insights-a6323871acf8468d89d50d175018c64e?pvs=4.

Personas

1. **Ben:** A junior at UW studying Informatics, looking for software engineering jobs while frustrated by the lack of accessible cybersecurity courses that are either too expensive or not offered often at UW.
2. **Sara:** A former security developer and current professor at UW iSchool, interested in integrating security principles into existing curriculums without creating extensive new content.

Solution Approach / Key Features

Our solution to bridge the knowledge gap in DevSecOps for new graduates, professors, companies, and self-learners involves creating a comprehensive, open-source curriculum hosted on GitHub. This curriculum emphasizes **practical, hands-on learning** and is designed to be **flexible**, **accessible**, and **current** with industry standards.

The curriculum is structured to support self-paced learning, enabling students to engage with the material on their own schedule. This feature is particularly beneficial for balancing the curriculum with regular coursework, part-time jobs, or other personal commitments. By allowing learners to progress at their own pace, we cater to a diverse range of learning styles and speeds, ensuring that each individual can thoroughly understand and absorb the DevSecOps principles without feeling rushed.

Being open-source, the curriculum is freely available to anyone with internet access. This inclusivity fosters a broader learning community, encouraging contributions and feedback from users worldwide. Open-sourcing the curriculum also promotes transparency and continuous improvement, as community members can suggest updates, report issues, and enhance the content based on the latest industry trends and technologies.

Offering the curriculum for free removes financial barriers that often restrict access to quality educational resources, especially in specialized fields like DevSecOps. This ensures that all interested learners, regardless of their economic background, can access state-of-the-art learning materials and gain the skills needed to succeed in the tech industry. The lack of a cost barrier also encourages wider adoption and dissemination across various educational institutions and self-learners globally.

The curriculum includes a range of resources designed to cater to different learning needs. Detailed README files provide step-by-step instructions for setting up environments and performing lab exercises. These are supplemented by interactive wikis that offer deeper dives into complex topics, ensuring learners not only perform tasks but also understand the underlying principles.

User Testing and Validation

Usability testing for the DevSecOps and CI/CD curriculum involved students and professors interacting with the content through designed tasks to evaluate comprehension, application, and navigational ease. We tested with college students going into cybersecurity or software engineering through hour-long interviews doing the labs and reading content. For professors in cybersecurity or software in Informatics, we conducted 30 minute interviews discussing their class approaches and learning how our modules could be best integrated into preexisting courses.

We defined our focus areas as:

- **Usability / Navigation**: Are the modules easy to use and navigate? Are there any functionality issues or bugs?
- **Applicability**: Is the content applicable in the security and engineering Industry? Is it applicable for current UW courses?
- **Comfort / Cohesiveness**: Is the content cohesive for users? Does It all connect together?

Students worked through modules on containerization, providing feedback that **praised the clarity and instructional quality** but suggested improvements such as **more**

visual aids and clearer terminological explanations. One student said, *"If this was part of the class curriculum I'd be **more comfortable** with it going forward"*. Another student mentioned, *"I liked the way that the labs were **very organized** and had commands right there"*.

Professors in cybersecurity and software engineering reviewed the potential integration of these modules into existing courses, valuing the **modular design** for its flexibility and relevance but noting the need for adaptable resources to fit various educational settings. One professor teaching software engineering fundamentals while working as an engineer at Atlassian mentioned how, "*Even the section on pipelines is something I **recently implemented at work**!"* showing the applicability and real-world usage of the content.

Both professors we interviewed mentioned classes such as **INFO 441, INFO 201,** and **INFO 310** as potential integration pathways for the modules. Additionally, both professors were currently working on creating new content for their classes. The software professor mentioned how our modules could be used as an optional activity to solidify concepts better. The cybersecurity professor was excited about our modules and was interested in integrating into his preexisting class, INFO 310, saying how the "Majority of what [we] have, I have limited versions of already" and we would just need to meet and discuss how it would integrate in.

From these interviews, it is clear that students find the modules helpful in their understanding of concepts and that professors are interested in integrating our work into their preexisting classes. In fact, both groups highlighted the **importance of such resources** in enhancing the educational experience without detracting from core curriculum goals. These user testing sessions were **essential** in validating our product not only for its experience but also in its usage for students and professors.

Ethical Considerations

For ethical considerations, it's crucial to recognize the importance of regularly updating the curriculum. The tech industry is dynamic, with principles and technologies evolving rapidly. If the educational content is not kept current, it can mislead students who rely on it to learn relevant and effective practices. Outdated information could not only waste students' time but also hinder their readiness for industry challenges. Therefore, consistent updates are essential to ensure the integrity and utility of the program in preparing students for real-world applications.

Next Steps Beyond Capstone

For our next steps beyond capstone, our project will be transitioned to the Women in Cloud organization within Boeing, who will assume responsibility for its further development and keeping content up to date. They will continue to refine and update the curriculum to ensure it remains current with industry standards and educational needs. Additionally, they will collaborate with Boeing to broaden the program's reach by integrating it into universities across the United States. This partnership aims to enhance the accessibility and impact of the curriculum, empowering a more diverse group of students to master DevSecOps and CI/CD principles and thereby strengthen the talent pipeline into the tech industry.