Learning Opportunities and Challenges for Software Engineers

by Devmountain Tutorials

☰ WHAT'S COVERED

In this lesson, you will learn how to list some of the challenges for software engineers. Specifically, this lesson will cover:

1. Big Data

The learning opportunities and challenges of being a Software Engineer are vast. Our development team has been increasing our use of automation to speed up the release process for our product. This takes an investment from the company but will allow us to release changes with more reliability and speed while reducing costs.

CI/CD stands for 'continuous integration' and 'continuous delivery.' This includes a set of practices where code between developers is merged quickly or continuously integrated. Continuous delivery includes pipelines or automation to make the release process quick and easy.

As connection speeds increase and the cost to store data goes down, it is becoming easier to capture and store large amounts of data. Big data and machine learning have been buzzwords for some time, but creating insights and value from that data is still a challenge for many organizations.

This challenge comes with an ethical dilemma; how do we give users the ability to know what data is collected and how it is used? In 2016, the European Parliament and Council created the General Data Protection Regulation (GDPR.)

This regulation was groundbreaking for the tech sector because it requires all companies that sell to European Union residents to comply regardless of the company's location. Among other things, the regulation requires companies to gain consent from users to process their data, allow users to anonymize their data, and provide notifications to users in the event of a data breach.

GDPR has been the largest data regulation, but it isn't the only one. The state of California adopted the California Consumer Privacy Act (CCPA) in 2018, which is similar to GDPR. With new and different regulations expected for other locations, the tech industry is coming together to establish best practices and tools to make compliance with privacy regulations easier.

Protecting Big Data

---

2. Tech Debt

One of the most notorious challenges for Software Engineers is managing tech debt. Technical debt refers to an old or outdated code that needs to be reworked to improve performance, maintainability, or security.

Maintaining code can quickly become a problem even at a small startup, where a simple website could have 10,000 lines of code. This grows quickly when you talk about large applications and operating systems that grow easily into millions of lines of code.

> ⚙ **THINK ABOUT IT**
>
> In 2017 Jeff Desjardins published an **infographic** showing how many million lines of code it takes to power different technologies. This estimates the Android operating system contains 12–15 million lines of code, but all of Google's services are estimated at 2 billion lines of code. Which helps explain why they need so many engineers.

The amount of technical debt is increasing the need to focus on cybersecurity. Older versions of programming languages have known security vulnerabilities that can be exploited.

Keeping your programming languages up to date can require significant rework for engineers. It can feel like a constant battle between maintaining good hygiene on the code you already have and balancing the new development and needs of the organization.

Trying to stay up to speed with the changes can seem overwhelming, but our team has a culture of continuous learning. We set aside time every week to stay up on the new developments, and we share what we learn with the entire IT department during lunch 'n' learn meetings.

☑	SUMMARY

This lesson explored two major challenges faced by software engineers: **big data** and **tech debt**. With the rise of automation and continuous integration and delivery, big data, and machine learning have become buzzwords in the industry. However, while capturing and storing large amounts of data has become easier, creating insights and value from it remains a challenge, and compliance with data privacy regulations such as GDPR and CCPA adds an ethical dilemma. On the other hand, managing tech debt (i.e., outdated code that needs to be reworked) can also pose a significant challenge, especially for large applications with millions of lines of code. To address these challenges, continuous learning and sharing of knowledge are essential.

Source: This tutorial was authored by DEVMOUNTAIN and Sophia Learning. Please see our **Terms of Use**.