Experiment 01 - Case Study

Roll No.	24
Name	Sreekesh Iyer
Class	D15-A
Subject	DevOps Lab
LO Mapped	LO1: To understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet your business requirements

Aim: To understand DevOps: Principles, Practices, and DevOps Engineer Role and Responsibilities.

Introduction:

The word DevOps is a combination of the terms - development and operations, meant to represent a collaborative or shared approach to the tasks performed by a company's application development and IT operations teams. DevOps describes the adoption of iterative software development, automation and programmable infrastructure deployment and maintenance. The term also covers culture changes, such as building trust and cohesion between developers and systems administrators and aligning technological projects to business requirements.

Benefits of DevOps:

1. Speed:

   Move at high velocity so you can innovate for customers faster, adapt to changing markets better, and grow more efficiently at driving business results. The DevOps model enables your developers and operations teams to achieve these results.

2. Rapid Delivery:

   Increase the frequency and pace of releases so you can innovate and improve your product faster. Continuous integration and continuous delivery are practices that automate the software release process, from build to deploy.

3. Reliability:

   Ensure the quality of application updates and infrastructure changes so you can reliably deliver at a more rapid pace while maintaining a positive experience for end users. Use practices like continuous integration and continuous delivery to test that each change is functional and safe.

4. Scale:

   Operate and manage your infrastructure and development processes at scale. Automation and consistency help you manage complex or changing systems efficiently and with reduced risk.

5. Improved Collaboration:

   Build more effective teams under a DevOps cultural model, which emphasizes values such as ownership and accountability. Developers and operations teams collaborate closely, share many responsibilities, and combine their workflows.

6. Security:

   Move quickly while retaining control and preserving compliance. You can adopt a DevOps model without sacrificing security by using automated compliance policies, fine-grained controls, and configuration management techniques.

Software and the Internet have transformed the world and its industries, from shopping to entertainment to banking. Companies interact with their customers through software delivered as online services or applications and on all sorts of devices. They also use software to increase operational efficiencies by transforming every part of the value chain, such as logistics, communications, and operations.

Case Study:

How Siemens created an open-source DevOps culture with GitLab

THE CUSTOMER

Siemens was founded in 1847 as a “backyard machine shop” in Berlin, Germany. Siemens AG (Berlin and Munich) is a global technology powerhouse that has stood for engineering excellence, innovation, quality, reliability, and internationality for more than 170 years. Active around the world, the company focuses on intelligent infrastructure for buildings and distributed energy systems and on automation and digitalization in the process and manufacturing industries.

Siemens brings together the digital and physical worlds to benefit customers and society. Through Mobility, a leading supplier of intelligent mobility solutions for rail and road transport, Siemens is helping to shape the world market for passenger and freight services. Via its majority stake in the publicly listed company Siemens Healthineers, Siemens is also a world-leading supplier of medical technology and digital health services.

In addition, Siemens holds a minority stake in Siemens Energy, a global leader in the transmission and generation of electrical power that has been listed on the stock exchange since

September 28, 2020. In fiscal 2019, which ended on September 30, 2019, the Siemens Group generated revenue of €58.5 billion and net income of €5.6 billion. As of September 30, 2019, the company had around 295,000 employees worldwide on the basis of continuing operations.

THE CHALLENGE

Large-scale company with large-scale needs

With over 20,000 developers, Siemens is divided into multiple organizations acting within different domains, mainly focused on business-to-business initiatives. According to Fabio Huser, Software Architect, the challenge was, “How do we build a DevOps culture around this really fractured federalistic company structure?”

Siemens needed a DevOps platform that offered collaboration, transparency, and proper code management to achieve their goal: A community for employees around the world, and a single source of truth for code. In order for a tool to be successful, Siemens required developers to have a collaborative mindset, full stack engineering knowledge, experience as an open source contributor, and a scalable platform that can be used to build upon itself. The vision for an improved workflow included the ability to collaborate on code and share it within minutes, speed up time to market, empower people to own their own code, and set the technological foundation for future business models.

THE SOLUTION

Adopting open source first

A small team within Siemens adopted GitLab in 2013 for collaboration and version control to develop Linux based embedded devices. In a typical grassroots approach, the team opened the platform for the whole company and scaled it up to over 40,000 users. The DevOps platform provides a place for different teams to work on the same project with the ability to share code within minutes and to collaborate easily across the world.

“The open source world comes up with new tools every week. But at the end of the day, we really try to solve it like a human issue. We want to collaborate, and the tool is just a secondary thing after all,” Huser said. “Thanks to GitLab, we found a tool which facilitates this ideology. It’s all about the people behind it and to maintain this idea and also have this community spirit within Siemens, you really need to establish such a community.”

In 2015, the code.siemens.com team shifted its focus to DevOps CI/CD, calling its specific workflow style “junkyard computing” in the early days to enable integration builds for open source components. “Thanks to the ease of use of the GitLab runner, you can set up new machines in a matter of minutes,” according to Huser. “If you have old machines laying around and you have a good enough set up in terms of network, you can literally set up new runners, new capabilities in a minute. It’s quite cost effective.”

THE RESULTS

Code, collaboration, and community

Today, code.siemens.com has its IT infrastructure on AWS. There is no longer a need for “junkyard computing” because code.siemens.com is a fully established service with a large in-house developer community provided by the Siemens IT organization.

The infrastructure evolved to a highly tuned and sophisticated setup, with a large number of EC2 instances all managed as Infrastructure as Code. SaaS solutions such as S3, RDS, ElastiCache, EFS, and ELB are used as well, since those can be replaced by standard open source solutions to minimize vendor lock-in. GitLab is hosted on AWS, also supporting services such as GitLab CI runners, monitoring, logging, crash reporting and more. Siemens has exceeded over 38 million CI builds since adopting GitLab. “If you’re part of Siemens you have different repositories you can collaborate with. We really try to bring the open source culture in and so far, we really succeeded. With CI/CD we have one and a half million builds every month. The whole culture has completely changed,” Huser said.

With GitLab, Siemens saves both time and money because there is no need to maintain local patches or manually update fixes. The code.siemens.com team follows an ‘upstream first’ workflow. “We go without patches. We only deploy upstream versions, nothing else. If we want to have new features, we contribute them to GitLab. We do not patch our instance,” said Roger Meier, Principal Key Expert and Service Owner of code.siemens.com from Siemens IT. “As soon as they are merged upstream, we will deploy the next version. So we ship every month. We do about four production deployments per month.”

The code.siemens.com platform is managed by a team of just eight people distributed across four countries in a highly agile fashion. All team members are committed to the open source way of working. They are coaching, supporting, and guiding the internal developer community, on top of managing the whole infrastructure and application. They use GitLab day by day to manage all their activities. All team members contribute and/or maintain several open source projects, while providing a reliable service for the wider Siemens developer community to increase developer happiness.

Collaboration happens throughout the entire organization with over 40,000 GitLab users and the potential to expand. GitLab helps Siemens ensure scalability internally and with customer development opportunities. “Our customers and our developers just want to have a reliable service that is running all the time,” Meier added.

Siemens teams heavily contribute to GitLab with over 150 merged MRs in GitLab. In addition, Huser and Meier are GitLab Heroes and were selected as GitLab MVPs. The teams not only use the DevOps platform, but they pride themselves with being so knowledgeable that they don’t use a support team from GitLab. “Since the beginning, we were talking about all our ideas and to have our roadmap visible for all the people within the company. You have to walk the talk, that’s key. Of course, focus on your customers: for developers, from developers,” Meier said.

DevOps Engineer Roles and Responsibilities:

A DevOps engineer works with diverse teams and departments to create and implement software systems. People who work in DevOps are experienced IT professionals who collaborate with software developers, quality assurance professionals, and IT staff to manage code releases.

DevOps engineers need to be able to multitask, demonstrate flexibility, and deal with many different situations at a time. Specifically, a DevOps engineer’s responsibilities include:

1. **Documentation**: Writes specifications and documentation for the server-side features.
2. **Systems analysis**: Analyzes the technology currently being used and develops plans and processes for improvement and expansion. The DevOps engineer provides support for urgent analytic needs.
3. **Development**: Develops, codes, builds, installs, configures, and maintains IT solutions.
4. **Project planning**: Participates in project planning meetings to share their knowledge of system options, risk, impact, and costs vs. benefits. In addition, DevOps engineers communicate operational requirements and development forecasts.

5. **Testing**: Tests code, processes, and deployments to identify ways to streamline and minimize errors.

6. **Deployment**: Uses configuration management software to automatically deploy updates and fixes into the production environment.

7. **Maintenance and troubleshooting**: Performs routine application maintenance to ensure the production environment runs smoothly. Develops maintenance requirements and procedures.

8. **Performance management**: Recommends performance enhancements by performing gap analysis, identifying alternative solutions, and assisting with modifications.

9. **Management**: Depending on the size of the organization, the DevOps engineer may also be responsible for managing a team of DevOps engineers.

The role of a DevOps engineer is not defined by a set career path. But there are a few skills that are required if you want to make DevOps your career.

1. **Communication**: A DevOps engineer needs to work with a lot of different people across different teams. Verbal and written communication among the teams is very important. If you do not enjoy working with a lot of different people or if you don’t communicate very well, this may not be the job for you.

2. **Problem-solving and troubleshooting**: DevOps engineers must be able to quickly identify problems or bottlenecks in everyday processes and procedures. They suggest improvements and implement fixes as needed to ensure processes meet expectations.

3. **Software development**: DevOps engineers need to have experience developing software applications and a good knowledge of scripting languages such as Java, JavaScript, Perl, Ruby, Python, PHP, Groovy, Bash, and so on.

4. **Process-oriented**: DevOps engineers must have the ability to focus on processes and procedures rather than results and underlying causes. They try to understand a design problem before trying to fix it. They also don’t try to force solutions that worked on old problems onto new problems.

5. **Documentation**: Processes must be clearly documented so that knowledge is passed on.

6. **Agile experience**: An understanding of, or experience in, an Agile software development environment helps a DevOps engineer.

7. **Product/project management**: A DevOps engineer needs to have had experience with leading a development project and managing schedules.

8. **Education**: This job usually requires a bachelor's or master's degree in Computer Science, Engineering, Software Engineering, or a relevant field.

9. **IT best practices**: This position requires experience with, or a good understanding of, IT best practices in an always-up, always-available service.

10. **Deployment/configuration management**: DevOps engineers should have experience with managing deployments and integrations using configuration management tools such as Git, Jira, GitLab, or Jenkins.

11. **Operating system platforms**: This role benefits from experience with Windows and Linux operating system environments.

Conclusion

In this experiment, we learned about the fundamentals of DevOps engineering and were fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet the business requirements.