**Case Study**

Assignment 4: CSDO 1020 – DevOps CICD Pipeline Modernization  
  
Submitted July 09, 2023

Members:

1. Yael Karmani (220760047)
2. Gary (Lyndon) Reid (220546388)

3. Vasudeo Rane (220778213)

4. Sopuruchi Owen (220143962)

5 Sotola Olusegun (220275715 )

# 

[**Introduction - Gary Lyndon Reid 3**](#_rrs1ygant965)

[**Next-Gen DevOps can improve our innovation, speed, agility and accelerate digital transformation - Yael Karmani 4**](#_6xirqj4tby02)

[**How SRE will help maintain application reliability - Gary Lyndon Reid 6**](#_py2tter1yo3k)

[**DevOps impact on business Productivity - Gary Lyndon Reid 7**](#_vcn5znnozx3a)

[**Explain and apply DevOps best practices in relation to cloud-native, serverless and Kubernetes. - Vasudeo 8**](#_9ltkiy3wxq9d)

[**DevOps Impact On Financial Results ( Sotola Olusegun) 11**](#_m3l9fh266su9)

[**Devcops Impact on data management–Vasudeo 12**](#_dnf31wj6ayk1)

[**DevOps impact on Machine learning :-Vasudeo 14**](#_ypdpdoj6q69j)

[**DevOps impact on artificial intelligence (AI) :- Vasudeo 15**](#_ypdpdoj6q69j)

[**Pipeline Design Diagram – Sopuruchi Owen 17**](#_5jdibrrmb89n)

# 

# 

# Introduction - Gary Lyndon Reid

This document outline many critical aspects DevOps. It aims to achieve three goals:

1. **Educate**

Our hope is that you and your teams will better understand the benefits of embracing DevOps, Cloud, FinOps, and SRE practices. We’ve outlined what these concepts mean and how they can accelerate software delivery, minimize cost, and increase our revenue.

1. **Support**

We are seeking your support to continue our journey of building and fostering a world-class multi-cloud DevOps organization. We are looking for developers to embrace the cloud and DevOps culture, we need product owners to understand the benefits, we need finance to approve investment, and we need leadership support.

1. **Design**

Finally, we will share our CICD pipeline design. The design should help you understand the patterns that will be embraced to increase the speed to ship, decrease the MTTR, and ultimately delight our customers.

Please reach out to our team with any questions at all. And if you’d like to get really involved, we invite you to join our DevOps Jam Sessions.

We host them every Friday from 11:00am to 1:00pm in Room 16C in Building A. We broadcast them via the DevOps Rocks Microsoft Teams channel for remote participation, but we ask that you attend in person if you are located in the GTA.

# 

# Next-Gen DevOps can improve our innovation, speed, agility and accelerate digital transformation - Yael Karmani

Next-Generation DevOps can greatly enhance innovation, speed, agility, and accelerate digital transformation within organizations. By adopting these advanced technologies, tool with efficient DevOps practices, businesses can revolutionize their software development and deployment processes, resulting in faster delivery of updates and new features while promoting collaboration and efficiency.

key aspects and example tools that leverage machine learning and AI to achieve these benefits:

**Better Collaboration and Communication:** Next-Gen DevOps encourages teams to work together and leverage AI-powered collaboration tools such as Slack, Microsoft Teams, and Jira. These tools facilitate real-time communication, knowledge sharing, and efficient decision-making, leading to improved teamwork, faster progress, and innovative problem-solving.

**Agile Development:** Next-Gen DevOps embraces AI and machine learning algorithms, including tools like TensorFlow and PyTorch, to enhance Agile development practices. These tools enable organizations to analyze data, user feedback, and market trends, providing valuable insights for continuous improvement and enabling them to respond quickly to changing requirements.

**Automated Infrastructure**: Next-Gen DevOps utilizes AI-driven tools like Terraform and Ansible for infrastructure management and optimization. These tools leverage AI algorithms to automate provisioning, configuration, and management of infrastructure resources, resulting in more efficient infrastructure utilization and reduced downtime.

**Scalability and Reliability**: Next-Gen DevOps leverages machine learning techniques and tools like Kubernetes and Docker to ensure scalable and reliable applications. These tools enable organizations to automate containerization, orchestration, and scaling processes, allowing systems to handle increased workloads, maintain availability, and adapt to changing resource requirements.

**Continuous Learning and Improvement**: Next-Gen DevOps promotes a culture of continuous learning and improvement using AI-driven analytics tools such as Splunk and ELK Stack. These tools can analyze vast amounts of data collected from various sources, identify patterns, detect anomalies, and provide actionable insights for optimizing processes and enhancing performance.

**Monitoring and Visibility:** Next-Gen DevOps relies on AI-powered monitoring and observability tools like Prometheus and Grafana. These tools use machine learning algorithms to analyze system metrics, log data, and user behavior, enabling proactive detection of issues, rapid troubleshooting, and ensuring optimal application performance.

**AI-Enabled Testing and Deployment:** Next-Gen DevOps incorporates AI in testing and deployment processes with tools like Jenkins X and GitLab CI/CD. These tools leverage AI algorithms to automate testing, identify potential defects, and optimize release strategies, ensuring high-quality software and faster time-to-market.

With Next-Generation DevOps organizations can work more efficiently as teams, deliver products faster, and continuously improve their processes. This not only helps businesses stay competitive in the digital world but also enables them to leverage the power of AI and machine learning through the use of mentioned tools.

# 

# 

# How SRE will help maintain application reliability - Gary Lyndon Reid

Our SRE can help us maintain the reliability of our application by leveraging many best practices. A core responsibility of our Site Reliability Engineers is to work with stakeholders to define Service Level Objectives, Service Level Agreements, and Sevice Level Indicators. Once these metrics are defined, we will have quantifiable targets. Our SRE will then instrument our application so that they can monitor the health of the application and notify the operations team in the event that performance is not as expected.

In addition to the above, our SREs will assist with all aspects of site reliability, and that includes items such as:

* Load, performance, and other nonfunctional testing
* Assisting with the design of highly available\fault tolerant applications and infrastructure
* Disaster recovery design and testing
* Embracing automation to reduce manual tasks (DevOps)

Ultimately, our investment in Site Reliability Engineering will position us to achieve our reliability targets. Our customers will be thrilled, and we will reduce the time spent troubleshooting issues and increase our time shipping features.

# 

# 

# DevOps impact on business Productivity - Gary Lyndon Reid

Establishing DevOps practices will have a positive impact on business productivity. As outlined in detail in the document, we will achieve several key benefits. You may be familiar with the project management that says you can choose 2 of 3 - fast, cheap, quality. With DevOps, we can have all three - we can ship code faster, ensure that we are shipping high-quality code, and do it cheaply. Best of all, we will spend less time firefighting, allowing us to invest more time and money into shipping features for our customers.

So how will we achieve all of this? We will build a fully automated CICD pipeline to provision infrastructure and data and ship our code for testing. We will write fully automated tests to quickly confirm that our features are high quality. We will deploy our code via automation once the tests are passed. We will instrument our application and work with SREs to ensure the reliability of our sites. Our pipeline will include functional tests, regression tests, static security scanning, dynamic security testing, static code analysis, and more. Our infrastructure, or rather our containers that operate in AWS, will allow for easy scaling.

# 

# Explain and apply DevOps best practices in relation to cloud-native, serverless and Kubernetes. - Vasudeo

DevOps best practices are :-

**Cultural Change** :- DevOps requires collaboration, transparency, trust, and empathy. Teams has to overcome working in silos and encourage teams to have cross-team communication or collaboration.

Cloud Native: Cloud-native environments promote collaboration by providing shared platforms and tools that facilitate communication and knowledge sharing among development, operations, and other teams. The use of shared services, APIs, and infrastructure-as-code enables teams to work together more effectively.

Serverless: Serverless architectures encourage small, focused teams that work closely together to develop and deploy individual functions or microservices. This collaborative approach fosters cross-functional communication and a sense of shared responsibility for delivering working software.

Kubernetes: Kubernetes enables the orchestration and management of containerized applications, which promotes collaboration between developers and operations teams. Kubernetes provides a common platform that encourages joint ownership of the deployment and monitoring of applications, bridging the gap between development and operations.

**Delivering fast using agile project management -** Agile methodology is a project management approach that prioritizes cross-functional collaboration and continuous improvement. It divides projects into smaller phases and guides teams through cycles of planning, execution, and evaluation.

Cloud Native, Serverless, and Kubernetes can significantly support Agile project management by providing the necessary infrastructure, scalability, and flexibility to deliver software products iteratively and respond quickly to changing requirements. These technologies enable teams to respond to customer feedback and changing requirements swiftly, fostering an iterative and customer-centric approach to software development.

**Shift left with CI/CD -** Shift left is an approach that moves testing to earlier in the software development lifecycle. Continuous integration (CI) is a process that improves code quality through deployment pipelines and support shift left strategy.

Cloud-native environments allow teams to replicate production-like environments easily, thus enables to identify and address issues related to scalability, performance, and integration early in the development process, reducing the chances of encountering such problems during later stages.

With serverless architectures, individual functions or microservices can be tested independently, enabling more targeted and granular testing. This approach facilitates early detection of issues and enables developers to fix them before they impact the overall system.

Kubernetes enables the containerization of applications, which provides a consistent and isolated environment for testing. Containers can be created and destroyed quickly, allowing developers to perform various types of testing, such as unit testing, integration testing, and end-to-end testing, in parallel

**Tool Strategy**:- A DevOps toolchain requires the right tools for each phase of the DevOps lifecycle, with key capabilities to improve software quality and speed of delivery. Cloud Native, Serverless, and Kubernetes provide the platform/tools which help to achieve the goals.

**Automation:-** Using automation the delivery of the software was done early. There is less manual intervention and hence there are fewer chances of any human error involved. Automation in build, implementation, testing, and deployment is the key aspect in DevOps implementation.

Cloud-native platforms provide tools and services for automated testing, such as integration testing, load testing, and security testing. By integrating these tests into the CI/CD pipeline, teams can identify and resolve issues early, ensuring higher quality and reliability of the application.

Serverless platforms provide fast execution and deployment of functions or microservices. This allows developers to quickly receive feedback on code changes, ensuring that issues are caught early in the development process and reducing the time required for debugging and resolution.

Kubernetes helps in automating the deployment, scaling, and management of containerized applications.

**Observability:-** For the successful implementation of DevOps monitoring play a critical role. The early an issue is identified helps it to fix early respecting in less effort and time as compared to identifying the issue later.

Cloud Monitoring Services provide very useful tools for monitoring services, and gathering logs which helps to gauge the health of cloud-native applications.

Serverless platforms like AWS Lambda and Azure Functions provide built-in monitoring capabilities that capture metrics at the function level. You can monitor execution duration, invocation counts, error rates, and more. This helps identify performance issues and optimize the function's performance.

Kubernetes offers built-in monitoring components to collect various metrics, tools like Prometheus and Grafana can scrape metrics exposed by the pods and provide insights into resource consumption, response times, and other performance indicators.

# 

# DevOps Impact On Financial Results ( Sotola Olusegun)

Integrating DevOps with Financial Services can bring to your company multiple benefits. And the best part is that you can improve your processes within your business and while having happy clients. In the end,By implementing DevOps, financial companies can automate key business processes, accelerate the deployment of new features, and enhance the reliability of their systems.DevOps in financial services has brought in the much-needed revolution in the Financial Services industry and helped shed its age-old, traditional, and siloed practices and replace them with modern, agile, and integrated systems that helped the finance industry immensely.But maybe you are wondering What is the relation between Financial Services and DevOps?

Well, since Financial Services companies are all about working with money, they need to be sure to count on quick delivery of their services in order to have continuous money flow, and as a result, a higher customer experience, which translates to higher engagement. And the only way to achieve these is through DevOps practices.

Highlighted below are various ways Devops have impacted Positively in the Financial services sector..

1. Reliability

Reliability has two main fields. On one side, applying DevOps for Financial Services ensures that your applications are up and running all the time without any interruptions, or at least, just a few of them. Certainly, if this happens, your customers are going to be happy with your services because they are permanently available without any kind of downtime.

On the other hand, reliability also means that you have a team you can rely on. A nearshore team would be an excellent idea. Why? Because you can be sure that you are working at the right time with experts in the field. As a curious fact, there is a 62% productivity increase in people who work from home.

2. Security Compliance

Financial Services are always dealing with investments and transactions, in consequence, they need to have all their data protected. That’s why security is a key aspect of your company and a well-protected app needs to be regulated by the best security standards and regulations in the Cloud. One of the most important aspects of security compliance is DevSecOps which is a practice to implement security during all the Software Development Lifecycle (SDLC) of DevOps practices.

3. Automation

DevOps can help Financial Services to automate their processes, a really valuable benefit because sometimes it can take a lot of time. For example, if you deal constantly with repetitive tasks, you can automate them and simplify the processes that are too complicated. This automation can help to create even from reports to big data.

In addition, automation minimize human errors and as a result, increases accuracy and safety measures. With automation, Fintech companies can benefit from reducing costs and increasing the speed of their services.

4. Cost savings

Migrating to the cloud can help your company reduce your costs by 20% or 30%. And if after that, you adopt the DevOps practices you can reduce your costs another 20% or 30%. Outsourcing a DevOps company in the USA will cost you around 90 – 120 USD; with a nearshore company, it’ll be around 45 – 60 USD. In the end, while outsourcing Remote DevOps you’ll have a cost reduction on DevOps for a considerable 50%..

5. Compliance

With new laws being enforced and customers concerned about the optimization and security, compliance is such a great benefit. With Soc2, customers will be confident of not getting their data compromised due to hacking or attacks. What is Soc2 and what is it required to get compliant? Soc2 is an international standard that establishes the basic security criteria to fully protect your application interface and your customers’ data. Getting compliant will increase security protection and faster releases to the market while decreasing

6. Time-to-market

Speeding up the time to market is an excellent benefit because, in a matter of days or even seconds, you can release new issues instead of waiting a long time. In order to fulfill these objectives, companies need to adopt the best DevOps practices to reduce their time to market.

Nowadays, most of the business work with applications with the objective of complete sales and operate in an easier way. In consequence, they need to bring those applications to the market very quickly so they can’t miss available opportunities.

# 

# Devcops Impact on data management–Vasudeo

**1) Accelerate analytics velocity :**- Using DevOps more data is gathered around the application using various DevOps tool which helps rapid improvement of development so they can continually deliver a valuable user experience. Data gathered can be transformed and made more actionable to make iterative improvements to their applications.

**2) Data governance:** Key objective of data governance is to ensure data remains secure and high quality. DevOps practices can have a significant impact on data governance within an organization. Data governance involves establishing policies, procedures, and controls to ensure the availability, integrity, and security of data throughout its lifecycle.

**3) Data mapping:** DevOps practices can help teams to plot out the path data takes from one point to another, confirming the data gets to the correct target fields without error.

**4) Data monitoring:** Observability is one of the key components of DevOps. Team can gain real-time visibility into data flows, identify potential issues or bottlenecks, and ensure data integrity and availability, this also helps the teams to visualize the entire database ecosystem, and can pinpoint the root cause of a potential problem and address it quickly.

# 

# DevOps impact on Machine learning :-Vasudeo

Machine learning operation implementation in an organization requires specific skills, process, and technology. The objective is to deliver machine learning solutions that are robust, scalable, reliable and automated. Exploration and experimentation are recurring activities and needs throughput the machine learning process, Devops practices can helps to implement all these aspects to achieve the goals of delivering fast.

A) **Version control for code, data and experimenting outputs:**- Data has a direct influence on the quality of machine learning models. Versioning of the experimentation code base and datasets helps to reproduce experiments or inference results. Versioning experimentation outputs like models can save effort and the computational cost of re-creating them.

B) **Multiple Environments :-** Separate development, testing and production environment helps to replicate the infrastructure which is needed to enable continuous experimentation and comparison against a baseline model. Access control for users might be different for each environment.

c) **Manage infrastructure and configuration as code -** Use infrastructure as code to create and update infrastructure components in work environments, so inconsistencies won’t get develop in environments. Manage machine learning experiment job specifications as code so that it can be easily rerun and reuse a version of experiments in multiple environments’ performance

D) **Track and manage machine learning experiments -** Machine learning experiments need a track of key performance indicators and artifacts. DevOps tools can be leveraged to keep track of the job performance through which a quantitative analysis of experimenting success can be gauged and can be used to enhance team collaboration and agility.

E) **Test code , validate data integrity and ensure model quality -** Left shift testing movement should be used to ensure the testing was done early and often this reduces the number of bugs and increase the quality of the code. Experimentation code based for correct data preparation and feature extraction functions, data integrity, and model performance.

F) **Machine learning continuous integration and delivery:**- Continuous Integration ( CI) is leverage to automated testing for the team. Model training should be included as part of continuous training pipelines. A/B testing should be included as part of the release to ensure that only a quantitative model is used in production.

G) **Monitor services, models and data :-** It is critical to monitor the services for their infrastructure uptime, compliance, and model quality when using serve models in a machine learning operations environment, setting up monitoring to identify data, model drift and to understand whether retraining is required. Automated triggers should be configured for automatic retraining.

# DevOps impact on artificial intelligence (AI) :- Vasudeo

DevOps enables AI at scale through the operationalization of machine learning models from design to production. DevOps for AI ensures that the right AI delivery process is in place and can bring the flexibility and "fail fast" approach needed during these times of constant change and technological transformation. DevOps will facilitate continuous delivery, deployment, and monitoring of models through

**Speed:** Improve time-to-market by reducing non-value-added activities in AI delivery

**Quality**: Accelerate cleaning datasets and promote continuous learning to improve AI model quality

**Scalability**: Preempt scalability considerations and ensure AI models can scale on-demand

**Stability:** Monitor deployed AI models to keep them reliable, stable and accurate

DevOps principles address the challenges around AI delivery since they are in a state of constant flux and present a repeatable, yet adaptable, approach to increase the maturity of your AI delivery through ongoing change.

**Data Preparation:** Preparing the right datasets for developing AI models is a crucial starting step as model accuracy depends on the quality and the size of the training dataset. Traditionally, data preparation—data extraction, data cleansing, data labeling and data validation—is a manual and cumbersome task where data scientists typically spend around 70% of their time. DevOps for AI automates such steps and enables data pipelines to handle big data. This improves the quality and the size of datasets and frees up data scientists to focus on feature engineering and AI model development.

**AI Model Development:** AI model development comprises three main activities—feature engineering, algorithm selection and dataset training. Model development is an iterative process that requires multiple rounds of model training to arrive at an optimal solution. Usually this happens in the local machines of data scientists and without much collaboration between various AI teams.

DevOps practices speed up AI model development by providing a cultural change of collaboration, providing infrastructure and the processes for parallel development, parallel testing and model versioning. This reduces the time and effort required to arrive at the optimal model.

**AI Model Deployment**: Deploying an AI model in production is an extremely challenging area for many organizations. Individual data scientists deploy a model in silos in their local workstations. For AI models to perform well in production, they should be able to handle incoming streams of data in real-time on highly scalable and distributed platforms. DevOps methods make AI models portable and modular. Such architecture allows AI at scale in operationalizing AI.

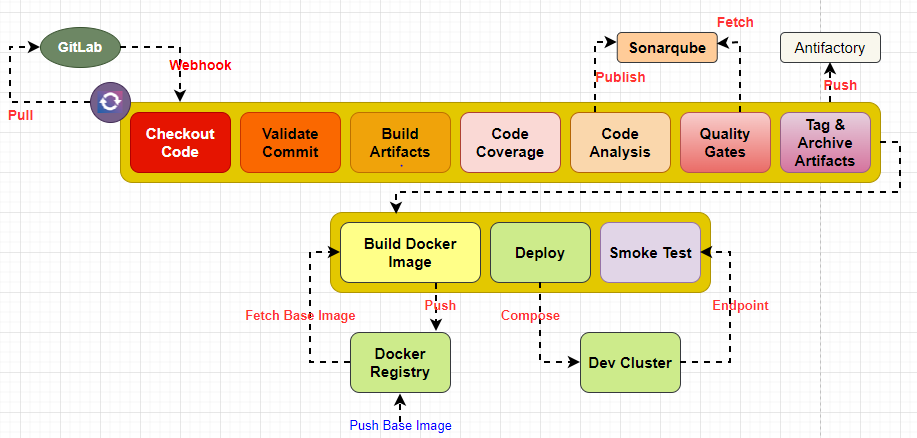
**AI Model Monitoring & Continuous Learning:** Once deployed, models face the threat of ‘model drift’. This is where the models deployed in production were developed based on historical datasets initially. Then as time passes, the data and the model become outdated and the model accuracy decreases resulting in ‘the drift’. DevOps for AI brings in the concept of continuous learning, where data and model performance indicators such as drift and accuracy, are monitored to ensure they stay relevant for a longer time. This results in better and more responsible AI solutions in the market.

# Pipeline Design Diagram – Sopuruchi Owen

**NEXT-GEN DEVOPS CI-CD PIPELINE**

**Stages of Build Pipeline:**

- Trigger[Webhook(On-commit/merge), Scheduled, On-demand], Checkout, Build, Unit Test, Code Coverage, Store Artifacts, Product Build Image, Smoke Test, Push Image to Docker Registry.

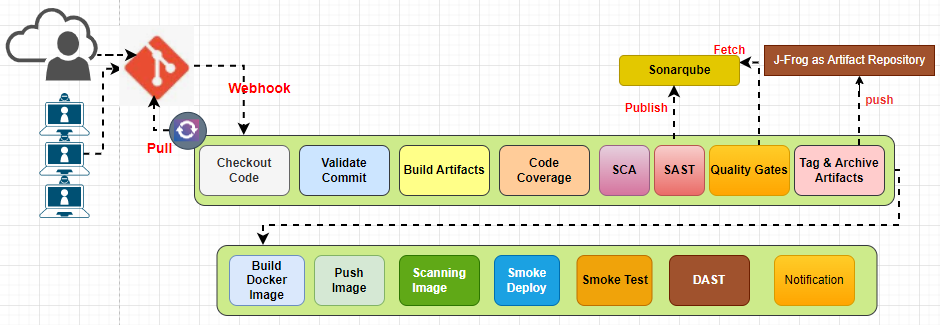


**NEXT-GEN DEVSECOPS CI-CD PIPELINE**

The Diagram below represents the DevSecops based CICD pipeline for SAST, DAST, Vulnerabilities, GitOps for Next-Gen DevOps

**Stages of DevSecops Build Pipeline**:

- Trigger [ Webhook(On-commit/merge), Scheduled, On-demand ], Checkout, Validate code, Build, Unit Test, Code Coverage, SCA, SAST, Quality Gates, Store Artifacts, Product Build Image, Scan Image, DAST, Smoke test, Push Image to Docker registry



**Trigger Stage:** Determines when the build pipeline should start running, with options such as webhook (on commit/merge) , scheduled intervals, or on-demand triggers.

**Checkout:** Retrieves the source code from VCS to create a working copy for subsequent stages.

**Validate code:** Performs static code analysis and checks for syntax errors, code formatting, and adherence to coding standards.

**Build:** Compiles the source code into executable artifacts or packages, preparing them for deployment.

**Unit Test:** Executes a set of automated tests to validate the behaviour and functionality of individual units or components within the codebase.

**Code Coverage:** Measures the percentage of code covered by the unit tests, providing insights into the effectiveness of the testing efforts.

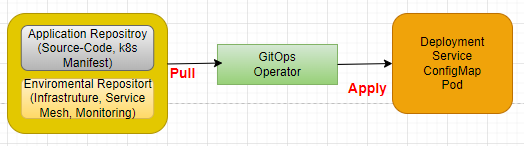
**Store Artifacts:** Archives the built artifacts and associated metadata in a secure location for future reference and deployment using **Jfrog** as **Repository**

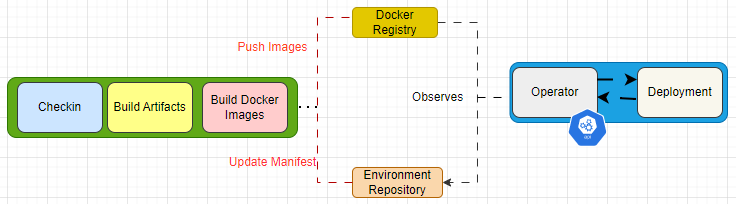
**Product Build Image**: Generates a production-ready image or package containing the compiled code, dependencies, and configurations.

**Smoke Test:** Executes a basic set of tests to verify that the built image or package is functional and can perform essential tasks.

**Push Image to Docker registry:** Transfers the product build image to a Docker registry, making it available for deployment in various environments or to other team members

**NEXT-GEN DEVOPS CONTINUOUS DELIVERY (GITOPS)-ARGOCD**





The Diagram above represents Next-Gen DevOps/DevSecOps implementation of CICD practice using Jenkins and Argo CD, a popular GitOps tool, using Kubernetes.