Aditya. A. Parwlekar
T.E. AI&DS T2-1
Roll.No → 2201072

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

21/01/2025

* Experiment No: 02 *

- **AIM**: To understand DevOps: Principles, Practices, and DevOps Engineer Role and Responsibility.

- **THEORY:**

→ What is DevOps ?

DevOps is a Collaborative approach where teams work together to build and deliver Secure Software efficiently. It combines Software Development (dev) and Operations (Ops) to decide how to accelerate delivery through automation, collaboration, fast feedback, and iterative improvement. Built on Agile Methodology, DevOps creates a culture of accountability, collaboration, and shared responsibilities for business outcomes.

→ Key Principles of DevOps :-

1. Collaboration & Communication → Breaking down silos between teams to work together seamlessly.
2. Automation → Automating repetitive tasks such as code deployment, testing and monitoring.
3. Infrastructure as Code (IaC) → Managing and provisioning infrastructure through code rather than manual processes.
4. Monitoring & Feedback → Continuous monitoring of applications and infrastructure to improve performance and reliability.
5. Agility & Iterative Improvements → Making small, incremental changes and quickly addressing issues.

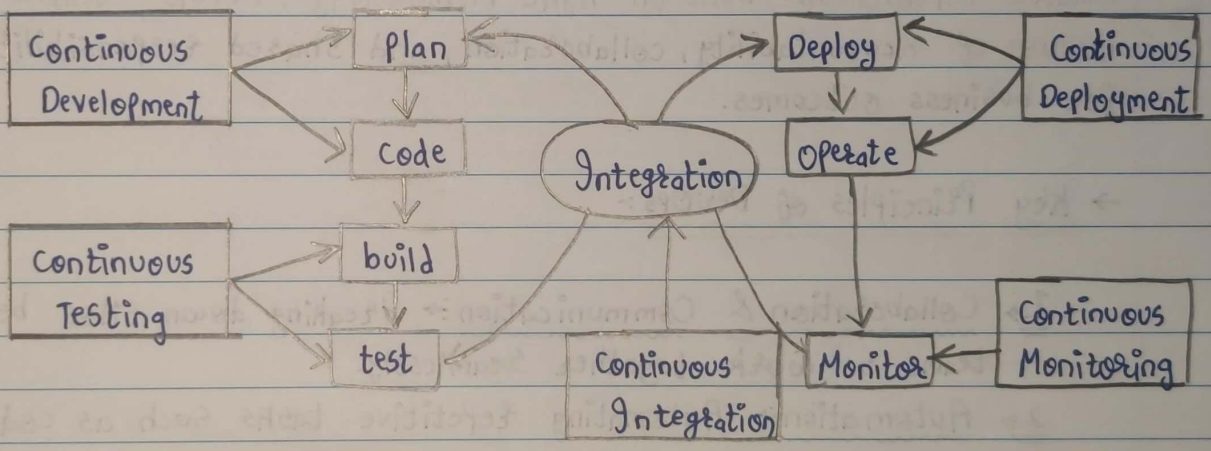
P.T.O.

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

→ Benefits of DevOps :-

1) Faster Time to Market :- Speeds up Software development and delivery.
2) Higher Quality Software :- Automated testing ensures fewer bugs and better performance.
3) Reliability :- Continuous monitoring and feedback loops improve system reliability.
4) Scalability :- Automation and IaC make it easier to scale systems.

→ DevOps Practices :-



[A] Continuous Development :- This is the phase that involves planning and coding, versioning and managing ~~build~~ builds of the Software application's functionality. Examples :- GitHub, maven, git, <APACHE ANT>, Subversion.

P.T.O.

THADOMAL SHAHANI
TSEC
ENGINEERING COLLEGE

[B] Continuous Testing :- Continuous testing is, executing automated tests, continuously and repeatedly against the code base and the various deployment environments. It is a Software testing methodology which focuses on achieving continuous quality & improvement. Example:- appium, Bamboo.

[C] Continuous Integration :- It refers to the build and unit testing stages of the Software release process. Every revision that is committed triggers an automated build and test. It Improves Developer Productivity ; Find & Address Bugs Quicker; and Deliver Updates Faster. Examples:- Jenkins, Travis CI, circleci.

[D] Continuous Delivery & Deployment :- Continuous delivery and deployment originate from Continuous Integration, a method to develop, build and test new code rapidly with automation so that only code that is known to be good becomes part of a Software Product.

[E] Infrastructure Management :- Without automation, building and maintaining Large-Scale modern IT Systems can be a resource-intensive undertaking and can lead to increased risk due to manual error. Configuration and resource management is an automated method for maintaining Computer Systems and Software in a known, consistent State. Some Infrastructure Services are like Storage Management ; Database Management; Network Management; Server Management ; Server Provisioning ; Security Management; Application Management; and Monitoring Alert and Notification.

P.T.O.

THADOMAL SHAHANI
TSEC
ENGINEERING COLLEGE

[F] Infrastructure as Code :- It is the practice of describing all software runtime environment and networking settings and parameters in simple textual format, that can be stored in your Version Control System (VCS) and versioned on request. These text files are called manifests and are used by DevOps tools to automatically provision and configure build servers, testing, staging and production environments. Examples:- CHEF, Terraform, etc.

[G] Microservice Architecture :- Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it need, such as libraries and other dependencies, and deploy it as one package. By doing so, thanks to the container, the developer can rest assured that the application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.
Examples:- Nagios ; SplunK.

[H] Cloud Based DeVOps :- DevOps automation is becoming cloud-centric. Most public and private cloud computing providers support DevOps systematically on their platform, including continuous integration and continuous development tools.
Examples:- AWS; Amazon Lambda, etc.

P.T.O.

→ DevOps Engineer :-

A DevOps engineer is responsible for the smooth operation of a company's IT infrastructure. They work with developers to deploy and manage code changes, and with operations staff to ensure that systems are up and running smoothly. To be successful in this role, a DevOps engineer must have a deep understanding of both development and operations processes, as well as a strong technical background.

As the world of business becomes increasingly reliant on technology, the role of a DevOps engineer is becoming more and more important. Companies are looking for individuals who can help them to streamline their operations and make the most of their IT infrastructure. If you have a strong technical background and have interest in working with both development and operations staff, than a carrer as a DevOps engineer could be the perfect fit for you.

→ Core Responsibilities of DevOps Engineer :-

1> Understanding customer requirements and project KPIs.
2> Implementing various development, testing, automation tools, and IT infrastructure.
3> Planning the team structure, activities, and involvement in project management activities.
4> Managing stakeholders and external interfaces.
5> Defining and setting development, test, release, update, and support processes for DevOps operation.

P.T.O.

6> Have the technical skill to review, verify and validate the software code developed in the project.
7> Troubleshooting techniques and fixing the code bugs.
8> Identifying and deploying cybersecurity measures by continuously performing vulnerability assessment and risk management.
9> Incidence management and root cause analysis.
10> Selecting and deploying appropriate CI/CD tools.

- CONCLUSION: Thus we Learnt about what is DevOps, DevOps Principles, Practices, DevOps Engineer and it's Responsibilities.