THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

SEPM

Name- Viraj Dawoodi
Roll no - 22
Batch - T-11

Q. To understand DevOps : Principles, Practices and DevOps Engineer role & responsibilities.

Q. What is DevOps?

⇒ DevOps is a Collaborative approach where teams work together to build & deliver secure Software efficiently. It Combines Software development (dev) & operations (ops) to accelerate delivery through automation, Collaboration, fast feedback & iterative improvement. Built on ~~Agile~~ agile methodology, DevOps creates a Culture of accountability, Collaboration & shared responsibility for business outcomes.

Core Principles of DevOps:
- Develop & test in production - like environments.
- Develop build frequently.
- Continuously validate operational quality.

Key Practices of DevOps:

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

2. Continuous Development - This is the phase that involves Planning

& Coding, Versioning and Managing builds of the Software applications functionality. Eg. Git, Github, Maven.

3. Continuous Testing:- Continuous testing is executing automated tests, Continuously & repeated against the code base & the various deployment environments. It is a software testing methodology which focuses on achieving Continuous quality & improvement. eg. Appium, Bamboo.

4. Continuous Integration:- Continuous Integration refers to the build & unit testing stages of the Software release process. Every revision that is committed triggers an automated build & test. eg. Jenkins, Travis CI.

5. Infrastructure Management:- Without automation, building & maintaining large-scale modern Without automation IT systems can be a resource intensive undertaking & can lead to increased risk due to manual errors. Configuration & resource Management is an automated method for maintaining Computer Systems & Software in a known, consistent state.

6. Configuration Management:- Infrastructure as Code is the practice of describing all Software runtime environment & networking settings & parameters in simple textual format, that can be stored in yours Version Control System (VCS) & Versioned on request. These text files are called Manifests & are used by Devops tools to automatically provision & configure build servers, testing, staging & production environments. eg. Chef, fairStack.

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

Q) Devops Engineer Role:
A Devops engineer manages a Company i.e IT Infrastructures, bridging development & operation, the primary goal is to improve the process and efficiency throughout the software development lifecycle.

Key role -

1. Facilitator of Collaboration -
Bridging the gap between development, operations & QA teams to Streamline Communication.
2. Automation Specialist -
Automate repetitive tasks like testing, deployment & Monitoring.
3. Continuous Integration & Continuous Delivery (CI/CD) -
Design, Implement & Maintain CI/CD pipelines to enable faster, reliable & repeatable software releases.
4. Infrastructure as Code -
Use tools like Terraform, Ansible or Cloud formation to define & provision infrastructure through code.
5. Monitoring & Incident Management -
Set up Monitoring System to track application performance & troubleshoot issue in real time. It also ensures that systems are resilient & downtime is minimized.
6. Cloud & Infrastructure Management -
Deployment, Manage & optimize applications on Cloud Platforms like AWS, Azure or Google Cloud, also handles Container orchestration.

Key responsibilities -

1. Collaboration & Planning -
   Work with development & operations teams to plan & design scalable solutions.
2. Configuration Management -
   Uses tools like Puppet, Chef or to manage server configuration & ensure consistency.
3. Pipeline Management -
   Maintain CI/CD pipelines to ensure seamless build, test & deployment workflows.
4. Monitoring & logging -
   Implement monitoring tools like Prometheus, Grafana or Splunk to track system health & measurement performance.
5. Support & Trouble Shooting -
   Respond to incidents & resolve production issues promptly & identify root causes of failure & implement fixes.
6. Documentation & reporting -
   Document system configurations, deployment processes & trouble shooting guides.