GLOSSARY OF DEVOPS

CI/CD PIPELINE TERMS

A Quick Reference Guide to DevOps CI/CD PipelineTerminologies

Get any Video course - https://zarantech.teachable.com/courses
LinkedIn Learner Community page - https://www.linkedin.com/showcase/devops-learner-community

Glossary of CI/CD Pipeline

Term	Definition
DevOps	A software development approach that emphasizes collaboration and communication between development and operations teams to improve the efficiency and quality of software delivery
CI/CD	Continuous Integration/Continuous Delivery or Continuous Deployment, a set of practices and tools that enable teams to deliver software faster and more reliably
Pipeline	A series of automated steps that code changes go through from development to production deployment
Source Control	A system that tracks changes to source code over time, enabling collaboration between developers and maintaining a history of changes
Build	The process of compiling code into an executable format, often including tasks such as compiling, linking, and testing
Test	A process of verifying that code changes function as expected, often including unit tests, integration tests, and other automated tests
Artifact	A versioned package or binary that is the output of a build, containing the compiled code and dependencies needed to run the application

Term	Definition
Release	The process of making a new version of software available for deployment, often including steps such as tagging the code and creating release notes
Deployment	The process of deploying code changes to a production environment, often including steps such as deploying artifacts, running database migrations, and configuring environment variables
Environment	A specific configuration of resources, such as servers and databases, used for development, testing, staging, and production
Infrastructure as Code (IaC)	A practice of managing infrastructure using code, allowing teams to version, test, and automate the creation and management of infrastructure resources
Orchestration	The process of managing the flow of a CI/CD pipeline, often including steps such as triggering builds and deployments, and managing parallel and dependent tasks
Pipeline as Code	A practice of managing the CI/CD pipeline using code, allowing teams to version, test, and automate the creation and management of the pipeline itself
Continuous Monitoring	The practice of continuously monitoring application performance, availability, and other metrics in production, often using tools such as log aggregation and monitoring systems

Term	Definition
Rollback	The process of reverting a deployment to a previous version, often used in the event of a production issue or failure
Canary Release	A deployment strategy where a new version of an application is released to a subset of users or servers, allowing for monitoring and testing before a full release
Blue/Green Deployment	A deployment strategy where two identical environments, one active (blue) and one inactive (green), are used to deploy and test new versions of software before switching traffic to the new environment
Feature Flags	A technique for toggling features on or off in production, allowing for more granular control over deployments and the ability to gradually release new features to users
Immutable Infrastructure	A practice of treating infrastructure as disposable and using automation to create and manage identical infrastructure instances, reducing the risk of configuration drift and increasing reliability
Microservices	A software architecture style where an application is broken down into small, independent, and loosely coupled services that can be developed and deployed independently

Term	Definition
Containerization	A technique for packaging software code and dependencies into lightweight, portable containers that can be run consistently across different environments
Docker	A popular tool for creating, deploying, and running applications in containers
Kubernetes	An open-source container orchestration system that automates the deployment, scaling, and management of containerized applications
Infrastructure Automation	The practice of using code to automate the creation, configuration, and management of infrastructure resources, such as servers, storage, and networking
Infrastructure Provisioning	The process of creating and configuring infrastructure resources, often using tools such as Terraform, CloudFormation, or Ansible
Infrastructure Configuration Management	The process of managing the configuration of infrastructure resources, often using tools such as Chef, Puppet, or SaltStack
Infrastructure Monitoring	The practice of monitoring the health, performance, and security of infrastructure resources, often using tools such as Nagios, Zabbix, or Prometheus
Infrastructure as a Service (IaaS)	A cloud computing service model where customers rent virtual machines, storage, and networking resources from a provider, such as AWS or Azure

Term	Definition
Platform as a Service (PaaS)	A cloud computing service model where customers rent a platform for developing, deploying, and managing applications, such as Heroku or Google App Engine
Serverless	A cloud computing model where customers rent computing resources on-demand, without needing to manage servers or infrastructure, often using a function-as-a-service (FaaS) offering, such as AWS Lambda or Google Cloud Functions
Git	A popular version control system used for source control, allowing teams to collaborate on code changes and maintain a history of changes over time
Jenkins	An open-source tool for automating the software development process, including continuous integration and delivery
CircleCI	A cloud-based tool for continuous integration and delivery
Travis CI	A cloud-based tool for continuous integration and delivery
Code Coverage	A metric that measures the proportion of source code that is covered by automated tests, often used as an indicator of the quality of test coverage

Term	Definition
Code Review	A process of reviewing code changes by peers, often including static analysis and manual inspection, to improve code quality and maintainability
Infrastructure Security	The practice of securing infrastructure resources against unauthorized access, data breaches, and other security threats, often including practices such as network security, access management, and encryption
DevSecOps	An extension of DevOps that emphasizes integrating security practices into the software development process, ensuring that security is a shared responsibility between developers, operations, and security teams
ChatOps	A practice of using chat tools, such as Slack or Microsoft Teams, to facilitate collaboration and communication between development, operations, and other teams involved in the software delivery process
Continuous Integration (CI)	The practice of regularly merging code changes from multiple developers into a shared code repository and running automated tests to detect integration issues early
Continuous Integration (CI)	The practice of regularly merging code changes from multiple developers into a shared code repository and running automated tests to detect integration issues early

Term	Definition
Continuous Delivery (CD)	The practice of automatically building, testing, and deploying software changes to production environments, often using automated deployment pipelines
Continuous Deployment	The practice of automatically deploying software changes to production environments without manual intervention, often used in conjunction with continuous delivery
Deployment Pipeline	A series of automated steps that deploy software changes from development to production environments, often including stages such as building, testing, and deploying
Blue-Green Deployment	A deployment strategy where two identical environments (blue and green) are maintained, with only one active at a time, allowing for seamless rollbacks and minimizing downtime during deployments
Canary Release	A deployment strategy where a small subset of users receive new software changes first, allowing for testing and validation before a wider release
A/B Testing	A technique for comparing two versions (A and B) of a software application or feature to determine which version performs better with users, often used for making data-driven decisions about software changes

Term	Definition
Load Balancer	A network device or software that distributes incoming traffic across multiple servers to improve availability and performance
High Availability	The practice of designing systems that minimize downtime and ensure that services remain available even in the event of hardware or software failures
Disaster Recovery	The process of restoring system functionality and data after a catastrophic event, such as a natural disaster or cyber attack
SLA (Service Level Agreement)	A contract between a service provider and a customer that defines the level of service expected, often including metrics such as availability, uptime, and response time
SLO (Service Level Objective)	A target for a specific service metric, such as availability or response time, used to measure and improve the quality of service
SLI (Service Level Indicator)	A metric used to measure a specific aspect of service performance, often used to calculate SLOs and track service performance over time
Metrics	Quantitative measures of software and infrastructure performance, often used to monitor and improve system health and reliability

Enjoyed this?

One favor to ask...

Sharing = caring

"Be a good friend.

Support free content with a repost."

@devops-expert