07.05Apr. DevOps Process

5 April 2023 07:09 PM

Automation + Infra(DevOps+Cloud)

1. Project Management = Agile(Scrum Master = Tasks)
2. Developer = DevOps (DevOps Engineer Git, Maven, Jenkins= CI/CD)
3. Operations = SysOps (System Engineer --> Ansible, etc. = Images/VMs)
4. Infrastructure = Cloud (IaaC ---> Terraform = GitOps)

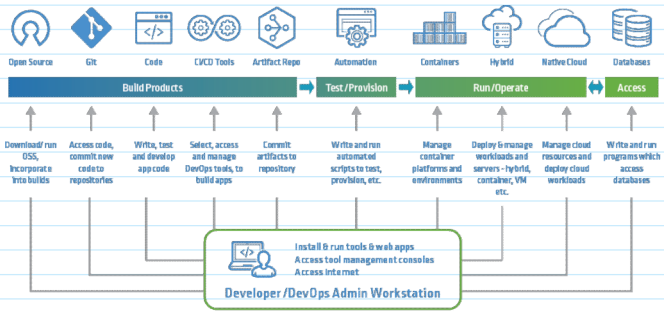
DevOps:

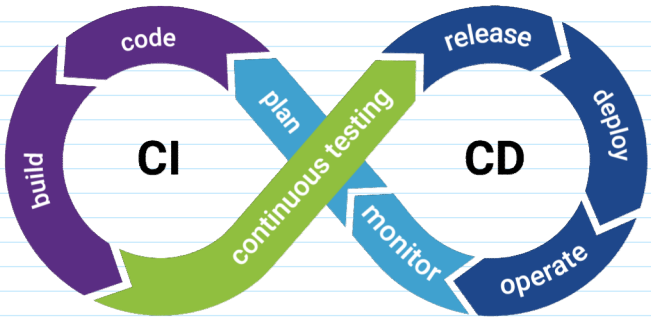
- Practice for Pipeline (CI/CD/CM/CT/CL)

CI/CD Pipeline

Agile --> DevOps(CI/CD) --> [Infra --> Operations] --> Client(End User)

CI - Integration



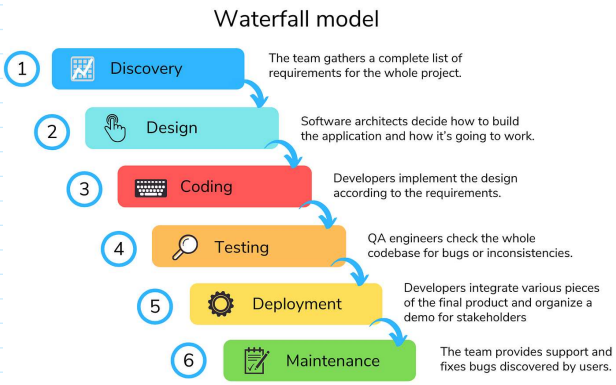


DevOps
DevSecOps
SysOps
SysSecOps
CloudOps
GitOps = IaaC - Terraform
DataOps
MLOps
SecOps
AIOps

SDLC

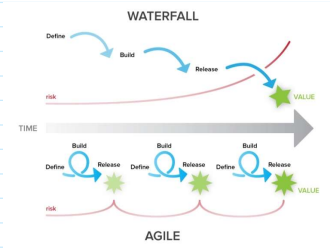
- Software Development Life Cycle

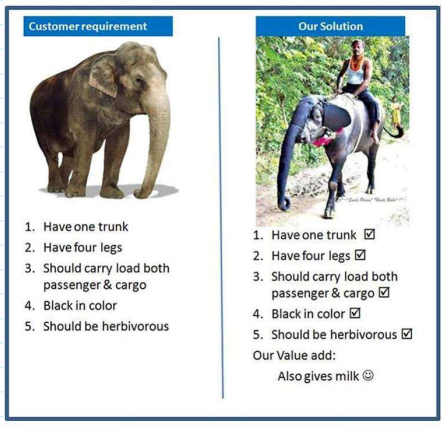
- Phases:
  - Analysis/Requirement
  - Design/Define
  - Development
  - Testing
  - Deployment
  - Maintenance
- Oldest SDLC Methodology : Waterfall Methodology

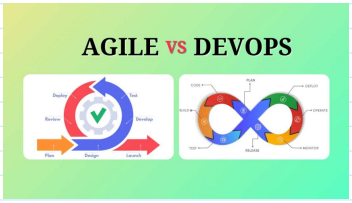


Agile Methodology:

Management - Agile
Technical - DevOps







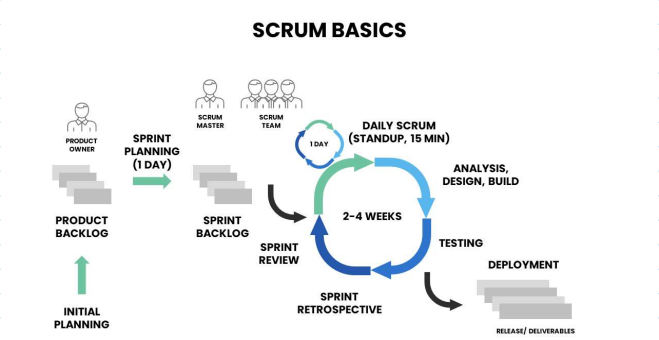
Focus	**DevOps**: Collaborate between dev and Ops team **Agile**: Frequent Customer feedback
Iterations	DevOps: Rapid Feedback Loops Agile: Iterative Cycle

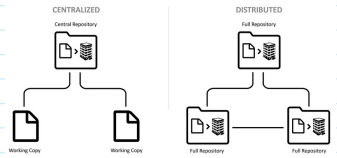
Duration of Sprint = 2-4 weeks

Daily stand up calls = 24 hrs

Meetings Duration:

1. Daily Stand up : 15 min.
2. Road Map/Planning Meeting:
    a. Decide Sprints Planning
    b. Review Meeting





Ticketing Tool:(Issues/Task)

1. Zoho Sprint
2. Bitrix24
3. Quick Scrum
4. Atlassian **Jira**
5. Service Now
6. Trello
7. ITSM

VCS Tool: (Project Code)

a. CVCS - Central Version Control System
b. **DVCS** - Distributed VCS
    a. **GIT**= **Github** --> GitLab
    b. mg=Atlassian BitBucket

Build Tool: (Generate Code to Output)

1. Apache Ant(Old)
2. Apache **Maven**(Stable) (CB)
3. Apache Gradle (Latest)
4. SBT

Testing Tool:

1. **Selenium** (CT)

8. Github Issues

CI Tool:
1. Open Source: **Jenkins**
2. AWS CodePipeline
3. Azure Pipeline

C. Deployment Tool:
1. **Ansible**
2. AWS System Manager

Infra Tool/Platform:
1. AWS CloudFormation
2. **AWS**
3. **Terraform**

Containerization:
1. **Docker**
   a. **CORE**
   b. **Compose**

Container Orchestration:
1. OS: **K8S**- Kubernetes (2000X)
2. Redhat - Open shift
3. AWS - ECS, EKS
4. Azure - AKS
5. GCP - GKS

CM/Visualization:
1. **Prometheus**
2. **Grafana**
3. AWS CloudWatch/CloudTrail