Devops curriculum using This Tools

Overview of Devops Architecture

DEVOPS OVERFLOW: Design:

* Definition goal of Devops
* Devops architecture
* " workflow.

* DEFINITION GOAL OF DEVOPS:

The main goals of Devops are to improve the speed, efficiency and quality of software development and delivery

✓ Increase Deployment frequency

✓ Improve Deployment Quality

✓ Reduce Lead time for changes

✓ Enhance collaboration & communication

✓ Improve recovery time

✓ Automate and streamline

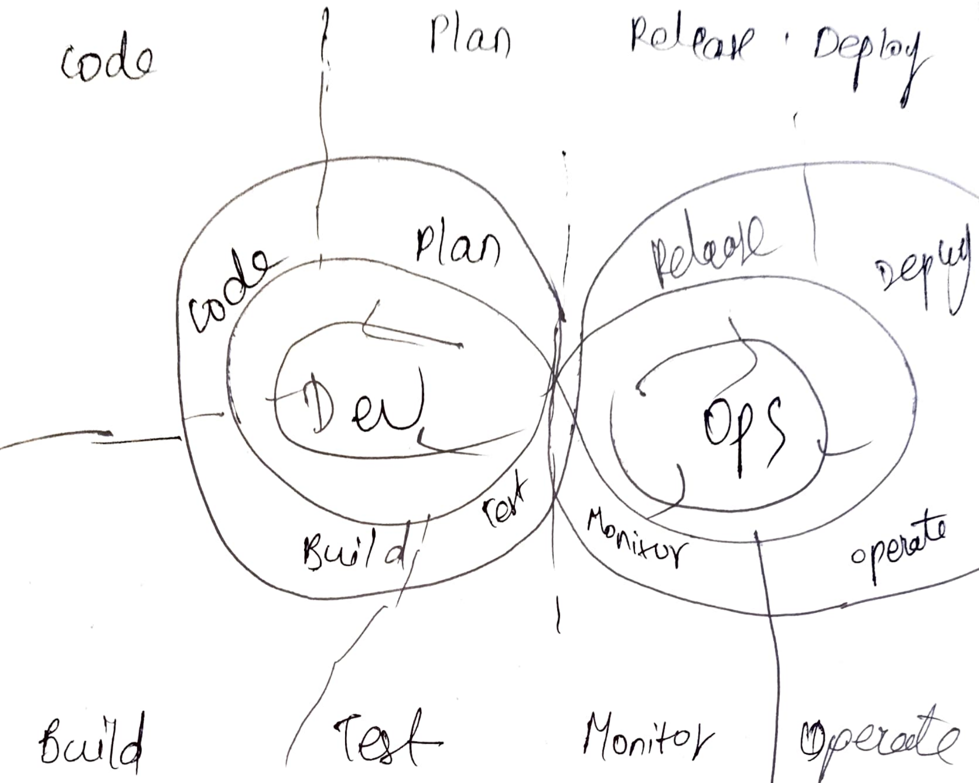
Processes –

Enhance collaboration &
munication

Improve recovery time

Automate and streamline

Processes -



KEY COMPONENTS OF DEVOPS ARCHITECTURE:

- Version Control System VCS)

Manage code versions, track changes, facilitates collaboration among developers.

- Continuous Integration (CI).

Automates the process of integrating code changes from multiple contributors into a single software Project.

- Continuous Delivery (Deployment:

(CD):

Automates the deployment of code changes to Various environments, ensuring that software can be released reliably at any time

CONFIGURATION MANAGEMENT:

Manges and maintains consistency in software environments (development, testing, production)

INFRASTURE AS CODE (IaC):

Manages and provisions computing infrasture through machine-readable definition files, rather than physical hardware or interactive configuration tools.

CONTAINERIZATION & ORCHESTRATION:

Packages. applications, and their dependencies into containers to ensure consistency across environments and simplifies deployment.

CONTINOUS MONITORING & LOGGING:

Monitors applications and infrastucture to detect performance issues, errors and security threats.

COLLABORATION & COMMUNICATION TOOLS:

and simplifies deployment

CONTINOUS MONITORING & LOGGING!

Monitors applications and infrastucture to detect performance issues, errors and security threats.

COLLABORATION & COMMUNICATION TOOLS!

Facilitates communication and Collaboration among team members enabling faster decision-making and issue resolution.

DEVELOP OVERFLOW!

Code: Developers write and commit code to a version control system. (eg: Git)

Build: The CI server automatically builds the code into executable files, creating artifacts that can be deployed.

Test: Automated ~~tests~~ are run to ensure the quality of the code. This includes unit tests, Integration tests and sometimes security checks.

Release: If all tests pass, the code is packaged and prepared for deployment.

Deploy: The code is automatically developed. to the target environment. eg: staging, production. continuous Deployment involves deploying to production automatically, continuous Delivery might require manual approval.

Operate: The deployed applications are monitored for performance, reliability and security. continuous monitoring tools collect metrics and logs, providing insights into the application's behaviour.

Monitor: feedback is collected from monitoring and users, providing data for continuous improvement. Any issues detected are fed back into the development process for resolution.

DEVOPS Vs TRADITIONAL IT OPERATIONS:

* Diff between Devops and traditional software development and IT operations.

* Benefits of adopting Devops Practices

* Building a culture of collaboration & Communication between development & operation teams.

* The role of automation and Monitoring in enhancing team efficiency.

DIFFERENCE BETWEEN DEVOPS AND TRADITIONAL SOFTWARE DEVELOPMENT AND IT. OPERATION

⇒ Collaboration and communication

* Traditional Approach: Development and IT operations team work in silos. Developers focus on writing code and operations team are responible for deploying and maintaining

waterfall model:

It can make ur projects flow smoothly, avoid bottlenecks, help hit dead lines, deliverables are met before next phase begin, allow team with perfection. This in-depth guide analyses the advantages of methodology.

⇒ Requirement gathering & Analysis

↓

⇒ system Design

↓

Implementation

↓

Testing

↓

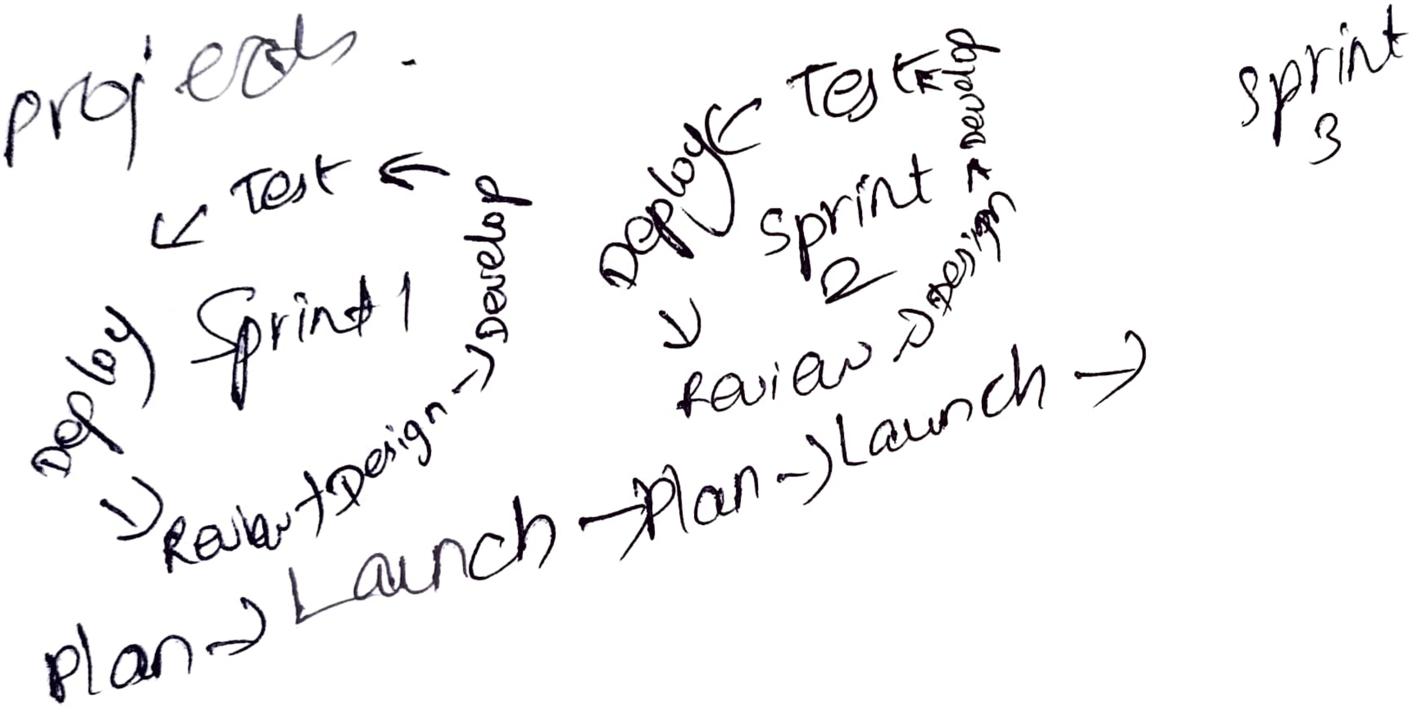
Deployment

↓

maintenance

AGILE:

Agele development is important because it helps to development teams complete projects on time and within budget . helps the improve communication between the development team and producer owner. Agily development methodology can help reduce the risks associated with complex projects .


Sprint 1
Test
Develop
Design
Review
Deploy
Sprint 2
Test
Develop
Design
Review
Deploy
Sprint 3
Plan
Launch
Plan
Launch