Dev-Ops

DevOps?

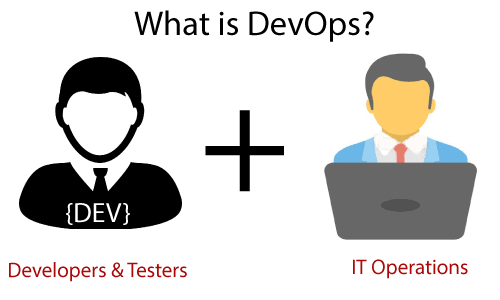
The DevOps is the combination of two words, one is **Development** and other is **Operations**. It is a culture to promote the development and operation process collectively.

What is DevOps?

DevOps is a software development methodology that improves the collaboration between developers and operations teams using various automation tools. These automation tools are implemented using various stages which are a part of the DevOps Lifecycle.

Or

The DevOps is a combination of two words, one is software Development, and second is Operations. This allows a single team to handle the entire application lifecycle, from development to **testing, deployment**, and **operations**. DevOps helps you to reduce the disconnection between software developers, quality assurance (QA) engineers, and system administrators.

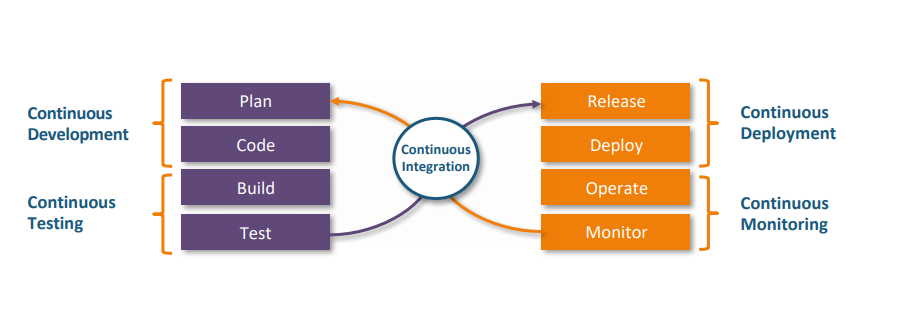


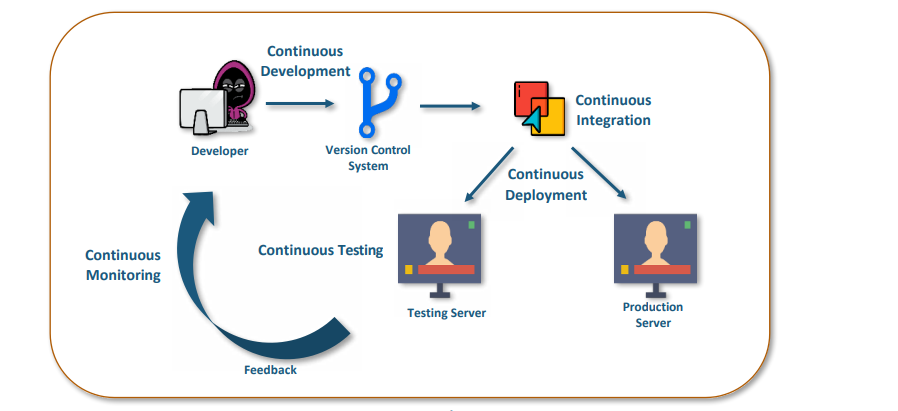
DevOps promotes collaboration between Development and Operations team to deploy code to production faster in an automated & repeatable way.

DevOps helps to increase organization speed to deliver applications and services. It also allows organizations to serve their customers better and compete more strongly in the market.

**How DevOps Works?**

The DevOps Lifecycle divides the SDLC lifecycle into the following stages:





*Automated CI/CD Pipeline*

### ****1. Continuous Development:****

This stage involves committing code to version control tools such as Git or SVN for maintaining the different versions of the code, and tools like Ant, Maven, Gradle for building/packaging the code into an executable file that can be forwarded to the QAs for testing.

### ****2. Continuous Integration:****

The stage is a critical point in the whole DevOps Lifecycle. It deals with integrating the different stages of the DevOps lifecycle and is, therefore, the key in automating the whole DevOps Process.

### ****3. Continuous Deployment:****

In this stage the code is built, the environment or the application is containerized and is pushed onto the desired server. The key processes in this stage are Configuration Management, Virtualization, and Containerization.

### ****4.**** ****Continuous Testing:****

The stage deals with automated testing of the application pushed by the developer. If there is an error, the message is sent back to the integration tool, this tool, in turn, notifies the developer of the error, If the test was a success, the message is sent to Integration-tool which pushes the build on the production server.

### ****5. Continuous Monitoring:****

The stage continuously monitors the deployed application for bugs or crashes. It can also be set up to collect user feedback. The collected data is then sent to the developers to improve the application.

Now we know some basic about DevOps and how it works.

Lets we will know about ADVT of DevOps :

• Shorter cycle time

• Faster, better product delivery

•Faster issue resolution and reduced complexity. and Etc…………..

This is about some basic of DevOps. Now, we know some tools which are used in Dev

Tools we use are :

1. Git/Github
2. Jenkins
3. Docker
4. K8s
5. Splunk

**What is Git:**

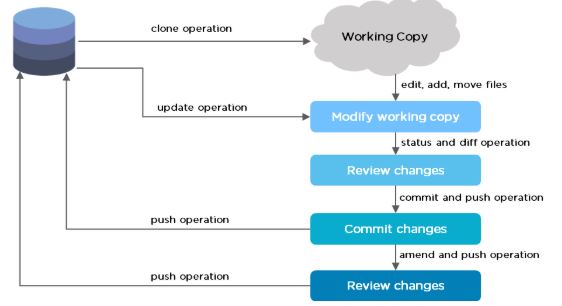
Git is a version control system used for tracking changes in computer files. It is generally used for source code management in software development.

* Git is used to tracking changes in the source code
* The distributed version control tool is used for source code management
* It allows multiple developers to work together
* It supports non-linear development through its thousands of parallel branches
* Note: vc: version control (v1, v2 ,v3) : Centralized  and distributed version control. Git is distributed V C.

## **Features of Git**

* Tracks history
* Free and open source
* Supports non-linear development
* Creates backups
* Scalable
* Supports collaboration
* Branching is easier
* Distributed development

**Git workflow:**



**Git vs github:**

Git is a version control system that lets you manage and keep track of your source code history. GitHub is a cloud-based hosting service that lets you manage Git repositories. If you have open-source projects that use Git, then GitHub is designed to help you better manage them.