# **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.

# **How DevOps Works?**

### 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.

# **DevOps Architecture Features**

**1) Automation**

Automation can reduce time consumption, especially during the testing and deployment phase. The productivity increases, and releases are made quicker by automation. This will lead to catching bugs quickly so that it can be fixed easily. For contiguous delivery, each code is defined through automated tests, cloud-based services, and builds. This promotes production using automated deployments

.

**2) Collaboration**

The Development and Operations team collaborates as a DevOps team, which improves the cultural model as the teams become more productive with their productivity, which strengthens accountability and ownership. The teams share their responsibilities and work closely in sync, which in turn makes the deployment to production faster.

**3) Integration**

Applications need to be integrated with other components in the environment. The integration phase is where the existing code is combined with new functionality and then tested. Continuous integration and testing enable continuous development. The frequency in the releases and microservices leads to significant operational challenges. To overcome such problems, continuous integration and delivery are implemented to deliver in a quicker, safer, and reliable manner.

**4) Configuration management**

It ensures the application can interact with only those resources that are concerned with the environment in which it runs. The configuration files are not created where the external configuration to the application is separated from the source code. The configuration file can be written during deployment, or it can be loaded at the run time, depending on the environment in which it is running.

# **Advantages and Disadvantages of DevOps?**

| **Advantages** | **Disadvantages** |
| --- | --- |
| DevOps is an excellent approach for quick development and deployment of applications. | DevOps professional or expert's developers are less available. |
| It responds faster to the market changes to improve business growth. | Developing with DevOps is so expensive. |
| DevOps escalate business profit by decreasing software delivery time and transportation costs | Adopting new DevOps technology into the industries is hard to manage in a short time. |
| DevOps simplifies collaboration and places all tools in the cloud for customers to access. | Lack of DevOps knowledge can be a problem in the continuous integration of automation projects. |