**What is DevOps?**

DevOps is a culture which promotes collaboration between Development and Operations Team to deploy code to production faster in an automated & repeatable way. The word 'DevOps' is a combination of two words 'development' and 'operations.'

DevOps helps to increases an organization's speed to deliver applications and services. It allows organizations to serve their customers better and compete more strongly in the market.

In simple words, DevOps can be defined as an alignment of development and IT operations with better communication and collaboration.

**Why is DevOps is Needed?**

* Before DevOps, the development and operation team worked in complete isolation.
* Testing and Deployment were isolated activities done after design-build. Hence they consumed more time than actual build cycles.
* Without using DevOps, team members are spending a large amount of their time in testing, deploying, and designing instead of building the project.
* Manual code deployment leads to human errors in production
* Coding & operation teams have their separate timelines and are not in synch causing further delays.

There is a demand to increase the rate of software delivery by business stakeholders. As per Forrester Consulting Study, Only 17% of teams can use delivery software fast enough. This proves the pain point.

# The benefits of DevOps

Companies that incorporate DevOps practices get more done, plain and simple. With a single team composed of cross-functional members all working in collaboration, DevOps organizations can deliver with maximum speed, functionality, and innovation.

There are technical benefits:

* Continuous software delivery
* Less complexity to manage
* Faster resolution of problems

There are cultural benefits:

* Happier, more productive teams
* Higher employee engagement
* Greater professional development opportunities

And there are business benefits:

* Faster delivery of features
* More stable operating environments
* Improved communication and collaboration
* More time to innovate (rather than fix/maintain)

## DevOps Lifecycle

Before we jump into DevOps, let ’s understand why DevOps came into the picture.

It all started with the first software lifecycle model, the waterfall model.

**WATERFALL MODEL**

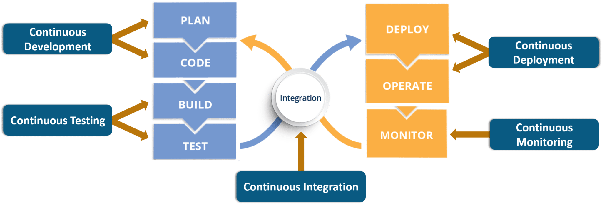
The waterfall model can be defined as sequential process in the development of a system or software that follows a top-down approach. The waterfall model had various phases such as Requirement Definition, Software Design, Implementation, Testing, Deployment, Maintenance, Retirement. Software development companies using the waterfall model approach spend so much time getting their product because unless you complete a particular phase, you are not able to proceed to further phases.

**AGILE**

Agile methodology is a practice that promotes continuous iteration of development and testing throughout the software development life cycle of the project. Both development and testing activities are concurrent unlike the Waterfall model. While the Agile approach brought agility to development, it was lost on Operations which did not come up to speed with agile practices.

**DEVOPS**

* The drawbacks of the agile methodology gave birth to DevOps approach. DevOps enables continuous software delivery with less complex problems to fix and faster resolution of problems.
* The term DevOps is a combination of two words namely Development and Operations. It is a practice which allows a single team to manage the entire application development life cycle, that is, development, testing, deployment, operations.
* DevOps is a software development approach through which superior quality software can be developed quickly and with more reliability. It consists of various stages such as continuous development, continuous integration, continuous testing, continuous deployment and continuous monitoring.



<https://www.quora.com/What-is-%E2%80%9CDevOps%E2%80%9D-and-how-does-%E2%80%9CDevOps%E2%80%9D-work>

<https://www.janbasktraining.com/blog/what-is-devops-lifecycle/>

<https://www.guru99.com/devops-tutorial.html#2>

<https://www.janbasktraining.com/blog/devops-tutorial/>