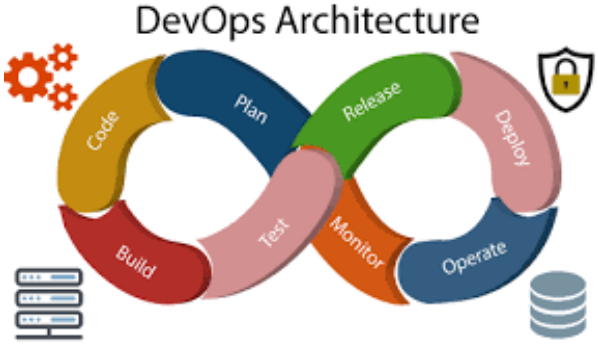
DEVOPS

- DevOps is a methodology.
- It is a combination of Development + Operations
- DevOps is a combination of tools, practices and philosophies which enables organizations to release more features, services at high velocity with more frequency
- DevOps helps you to reduce the disconnection between s/w developers, Quality Assurance (QA) engineers, and system administrators.
- This allows a single team to handle the entire application lifecycle, from development to testing, deployment and operations.
- With the help of DevOps, quality and speed of the application delivery has improved to a great extent.

DEVOPS ARCHITECTURE FEATURES



It is used for the application hosted on the cloud platform and large distributed applications. Agile development is used here. So, that integration and delivery can be contiguous.

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 in catching bugs quickly. So, we can fix easily.

Collaboration

- The Development and Operations team collaborates as a DevOps team, which improves cultural model as the teams become more productive with their productivity, which strengthens accountability and ownership

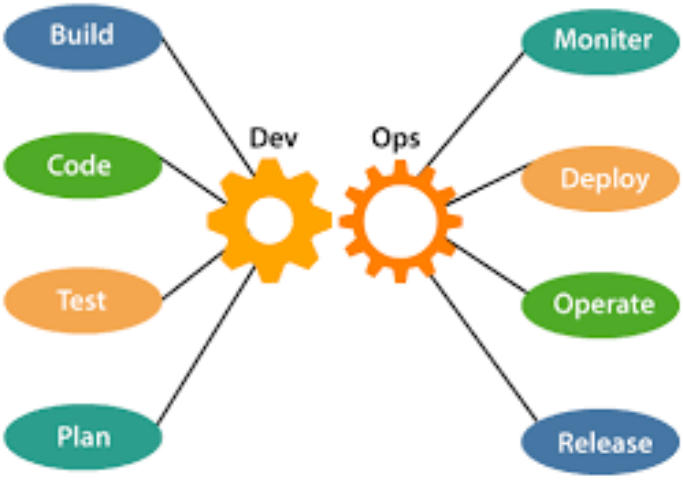
Integration

- 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.

Configuration Management

- It ensure/make sure the application to 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 files can be written during deployment, or they can be loaded at the run time, depending on the environment in which it is running.

COMPONENTS



These below components are done by development team.

1. Plan
2. Code
3. Build
4. Test

These below components are done by Operations team.

1. Deploy
2. Monitor
3. Operate
4. Release

DEVOPS-PHILOSOPHY

- Automate repetitive tasks and always thing for improvements
- Don't work in silos make sure you collaborate and communicate on time with other members and teams.
- Keep shorter software development lifecycles for example 2 weeks releases.

DEVOPS - LIFE CYCLE

DevOps defines an agile relationship between operations and development. It is a process that is practiced by the development team and operational engineers together from beginning to the final stage of the product.

Seven Phases (7) :-

- Continuous Development
- Continuous Integration
- Continuous Testing
- Continuous Monitoring
- Continuous Feedback
- Continuous Deployment
- Continuous Operations

DEVOPS - PRACTICES

1.Continuous Integration

It is a development practice where developers should integrate changes to their central repositories like Git daily to reduce conflicts and bugs.

2.Continuous Delivery

- It expands upon CI, it is automated process to deploy the code all the way from development to production
- Approval is required before promoting changes to production environment

3.Continuous Deployment

- It does not require approval
- 99% of customers don't follow this

4.Configuration Management

- Is a process where we maintain infrastructure and software changes through automation and the goal is to maintain them in desired state.
- For instance I wanna set up a tomcat server on 100 servers. This activity can be automated using configuration management tool
- It saves lots of time and it makes it easy to fix defects if any.
- Popular configuration management tools are
  - [ ] Ansible
  - [ ] Puppet
  - [ ] Chef

- [ ] Saltstack

5.Microservices Architecture

In this architecture big application(single code base) is divided into smaller services so that we get following benefits

- We can release more services
- Development will be much faster
- We can scale any specific service alone

6.Monitoring

It is very important to monitor applications and services after it is deployed into production.