**Devops**

* **Lecture 1-7 :**

**Lecture 3 :**

**What is Devops?**

Dev writes the code and provides the ops team for deploying to the server.

Devops Lifecycle –

1. Code – Developers commits the code
2. Code Build – Deployable software : Artifact
3. Code Test – Unit and Integration Test
4. Code analysis – Vulnerability, best practices
5. Delivery – Deploy changes into Staging
6. DB/Sec Changes – Every other ops changes
7. Software Testing – QA/Functional, load, performance test
8. Deploy to Prod
9. Go live – User Traffic diverted to new changes
10. User Approval - Feedback
11. Keep Monitoring

**Software Development Process (SDLC)** – It is an organised and well-defined process.

* Requirement Gathering and Analysis
* Planning - what do we want?
* Design – Roadmap for developers
* Development – Developers write code as per design
* Testing – Will solve all the issue after then reach to production
* Deployment – Responsible for app to always up and running
* Maintanence – Requires for system health, Performances

Waterfall, Agile, Spiral, BigBang Model in SDLC.

**Lecture 5:**

**What is Continuous Integration (CI) :**  It is an automated process in Devops, which generates software and features quickly and efficiently.

* Code is placed at Github – Everyday code changes and will be deployed to build server and everyday it is build , tested and evaluated.
* The code is packed in artifact format they are archeive of a file – In Java – WARJAR, In windows -DLL/EXE/MSI and or in ZIP/TAR
* From respository artifact is moved to server after deploying testers test the application and after approved, it can be shipped to production servers

Goal of CI is to detect the bug at early stage so it does not get multiply.

Version Control – Git

Build Tools – Maven, Ant, Gradle, MSBUILD etc

S/w repo for storing artifacts – Sonatype nexus, Jfrog artifactory, Archiva

CI Tools – Jenkins, Circle CI, Bamboo CI

**Lecture 6:**

**What is Continuous Delivery (CD) :** It is an automated process of delivering code changes to the server quickly and efficiently. Extension of CI process.

* Ops Team request to deploy the artifacts for further testing.
* They deploy to server
* Deployment gets failed so the developer and ops team need to work
* Deployment – Server provisioning, dependencies, configuration changes, network, artifact deploy.
* All the above steps should be automated.

Tools – Ansible, Puppet, chef – for system automation

* Terraform, cloud formation – for cloud automation
* Jenkins, octupus deploy for CI/CD automation