JENKINS

* Jenkin is totally plugin based tool it mostly contains 1000+plugins.
* Without plugin we cannot do any project.
* For each module we may require 3-4 plugins.
* We need to update the plugins daily.
* For example containerization- deploy to container-tomcat.

CI/CD:

MANUALLY

* Unit testing
* Functional testing
* Regression testing on every time.
* For all these it will take months of time.
* To overcome we came with automation.

AUTOMATION

* Unit testing
* Static code analysis
* Code equality-vulnerability
* Automation testing (end-to-end)
* Reports
* Deployments.

CI/CD Pipeline Procedure:

customers

feedback feedback

Approve & deploy

CD

Build &test

CI

Source code control server

Developers



feedback

Feed-

Back

* This will repeat until we get positive feedback.
* Made procedure too fast than normal CI/CD automation.
* Deliver the project in less time.

CI/CD Pipeline key features.

* Automatic build & tests
* Rapid feedback
* In CI it provides rapid feedback like code is reliable or not.
* Culture of agility🡪reduce wastage & quick delivery .

CI/CD Pipeline key features.

* Manual deployment: after success completion of testing the release is packaging and waiting for scripted code for triggered manually deployment.
* Deploy-ready code: after production for final check point only we will deploy.

BEFORE CI:

Integration/build

Git hub

Testing

AFTER CI:

Build-test-deploy

Github

feedback

Build trigger:

1. Poll scm

* Cron expression based🡪schedule🡪dev modify🡪automatically runs the jobs🡪 build 🡪deploy.

1. Build Periodically

* We will set the time 🡪 it will check for changes for given time & it will build.
* Cron based expression
* \*\* \*\* \*\* \*\*

1. Webhooks

* Event based.
* Dev changes🡪immediately action🡪runs the jobs🡪build 🡪deploy.
* Webhooks can also be created in github.

Plugins:

* External/ additional features added to our project.
* In order to install plugins we need to utilize Jenkins dashboard.
* Jenkins dashboard🡪plugins🡪type of search plugins🡪.
* Jenkins is installed in 3 ways.

1. In Jenkins dashboard , manage Jenkins 🡪manage plugins --.available🡪plugin name🡪 install.
2. We need to download the plugin in our local laptop🡪the download plugin needs to be uploaded into Jenkins dashboard manage(Jenkins).

* 🡪advanced🡪upload plugin🡪choose downloaded file🡪select the downloaded file🡪open🡪upload.

1. We need to download the plugin in our local laptop 🡪copy that plugin to Jenkins instance.

* Set the path: /var/lib/Jenkins/plugins.

Tools configuration in Jenkins:

* Setting up tools or global tools like jre or maven in Jenkins to use these tools across multiple jobs without needing to configure them individually for each job.
* Imagine you have multiple Jenkins jobs that need to build and run java applications. Each of these jobs requires a jre and a build tool like maven.

Without global tools:

* For each job, you would need to manually specify the path to the jre and maven every time you create or configure a job.
* If you ever need to update the jre or maven version you’d have to go into each job and update the paths individually which is time – consuming and error prone.