

What is Continuous Integration, Delivery and Deployment

Description

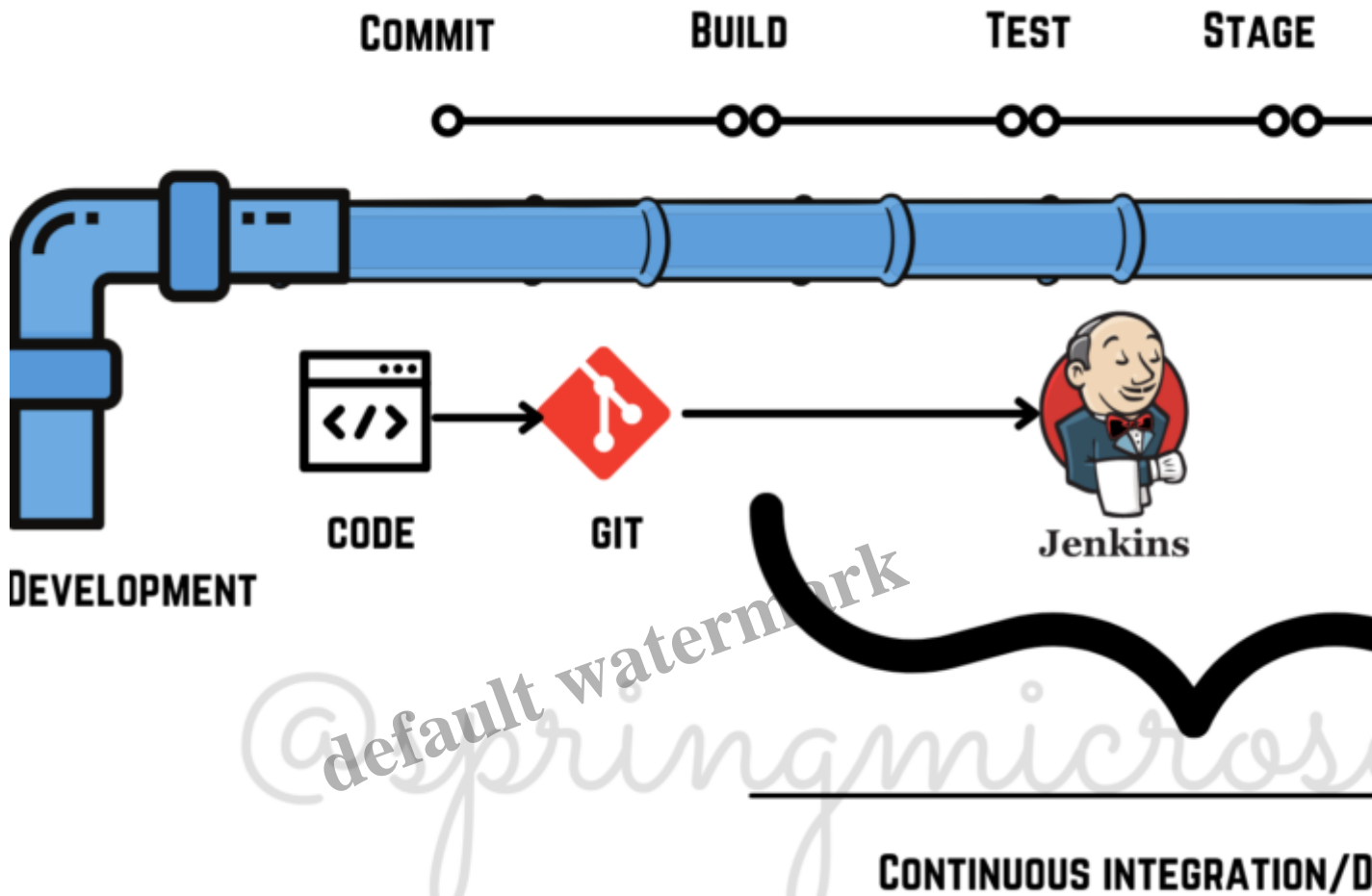
Before going over what is Continuous Integration, Delivery and Deployment let me explain you how the industry worked when it came to delivering the software product.

There are various software development life cycle models. One of them was Water fall model. In this model Customer gave the entire requirement and the organisation would deliver the product when it was fully completed irrespective of whatever time it took. In all the stages of water fall model (analysis, design, coding, testing, deployment), the process moved in a cascade mode. Each stage has concrete deliverable and is strictly documented. The next stage could not be started before the previous one was fully completed. For example, software requirements cannot be re-evaluated further in development. We could not try the software until the last development stage is finished, hence putting the project at high risks and unpredictable results. If the developers crossed the time that was allotted to them, then the testing team would get very less time to test and then errors would arise every now and then.

It is not like Water fall model has all drawbacks. This models suits certain category of projects.

Enough of criticizing Water fall model, let get back into the topic.

default watermark



Above image shows the delivery pipeline. To sum up in a sentence, the developer commits the changes, the build get triggered and the tests are run, after that the changes are deployed to the staging and higher environment and then if all satisfactory then the product goes to the production.

Continuous Integration

In Continuous Integration, a developer should commit their code as often as possible. As soon as the code is checked in, automatic build gets triggered and the tests are run. Continuous Integration pretty much stops at the build stage.

Continuous Delivery

Continuous Delivery is an extension of the Continuous Integration. It basically runs all the production like tests in the stage environment. In most of the cases Stage environment is the production like environment. The product can also be delivered into the production at this stage. Usually a change

management procedure is followed here like raising a change request and getting the approval from the testing domain and the higher management.

Continuous Deployment

This is the final stage where the product is delivered to the production without manual intervention. In this stage, once you commit your code, the changes will be delivered to the production by making sure the build is successful, the test cases are successful and then the change will be delivered to the production.

One of the end goals is to get the quality product within timeline or as soon as possible to the customer.

Category

1. AWS
2. Cloud

Date Created

December 25, 2021

Author

kk-ravi144gmail-com

default watermark