CI & CD
Adoption into Udapeople

A better guarantee for growth and fast delivery

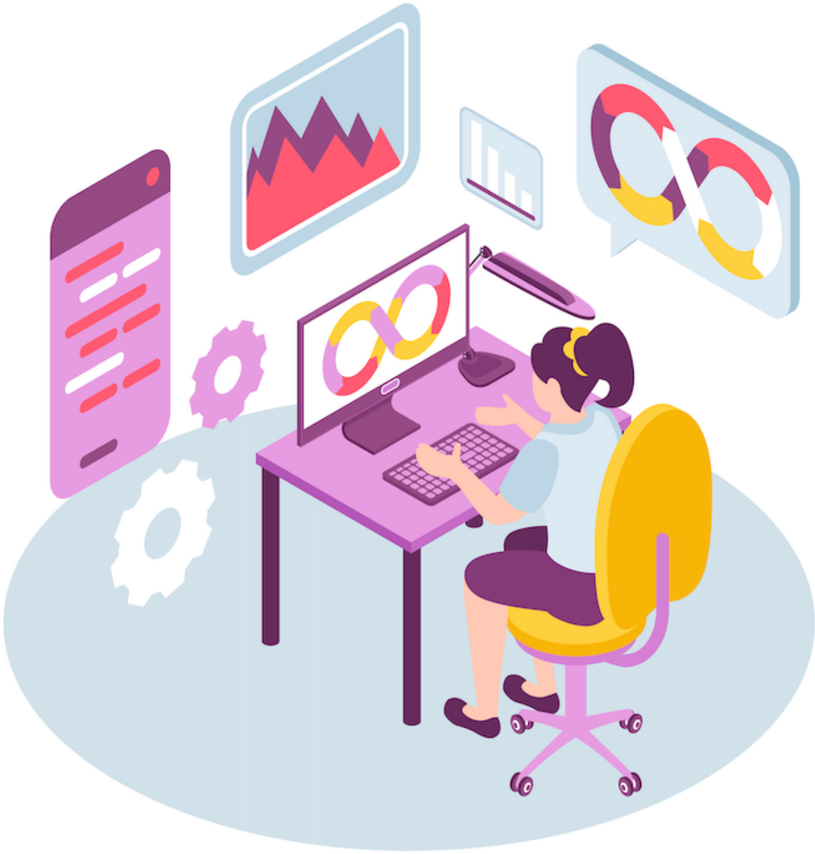
Contents

CI/CD ? Concept explained

Continuous Integration + Continuous Deployment = Continuous Delivery

CI/CD involve 3 main concept which are :-

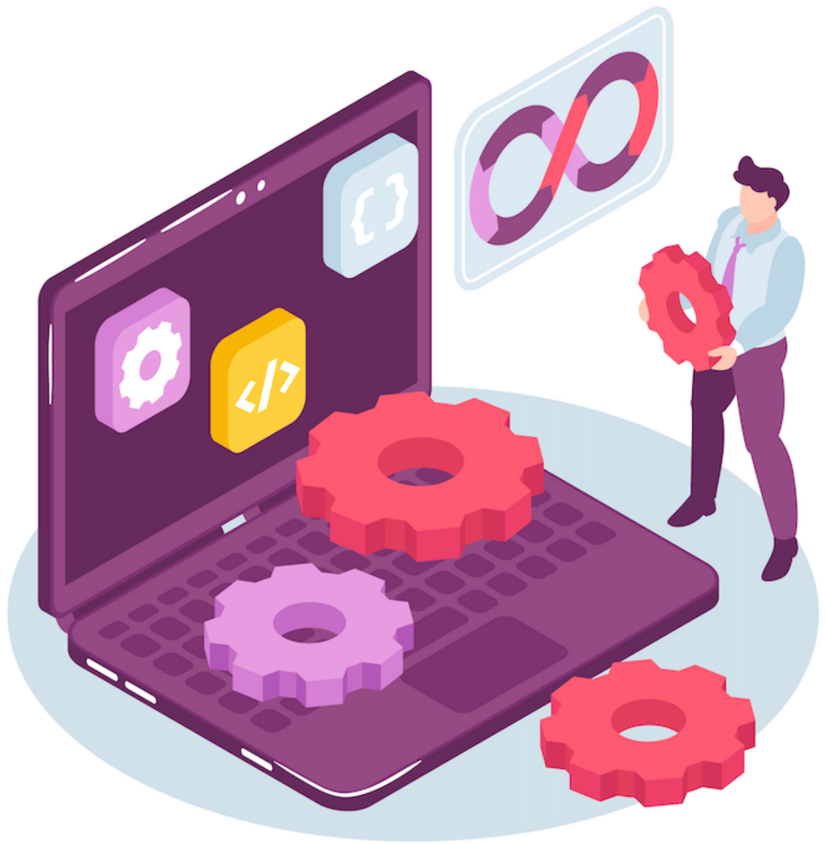
- **Continuous integration (CI)** is the practice of merging all developers' working copies to a shared mainline several times a day. This is just a short term for joint collaboration in building a whole product though contribution of individual developer work into a main production line eliminating code conflict , code duplication and also promoting unity during development .
- **Continuous Deployment (CD)** is an approach in which software functionalities are delivered frequently through automated deployments . In another word , Continuous Deployment consists of instantiation of a new version of an application and retiring the old version once it the new one is ready.
- **Continuous Delivery** is discipline where our team at build software in such a way that the software can be released to production at any time while still including new feature at a fast pace .



Current Pain point ?

- Current release involves manual process which is error prone without much details , leading to delay to deployment to production.
- We are currently experience poor analysis in our code which leads poor software quality
- Late feedback from the business side of the company which lead to no flexible solution
- Complexity in deployment which is a result of having only few people in the company that can deploy our application using scripts resulting to little or no smoke test or rollback

What will Continuous Integration adoption do to benefit Udapeople ?



Current release involves manual process which is error prone without much details , leading to delay to deployment to production.

Reduce Cost

Less developer time on issues from new developer code after catching compile time errors after subsequent merge

Avoid Cost

- Less bugs in production and less time in testing by catching unit test failures
- Prevent embarrassing or costly security holes by detecting security vulnerability

Benefit of CD

Although , continuous deployment is not very easy to setup . It is very productive in the long run

Complex deployments and handcrafted automation which often fail. Missing smoke tests and rollback mechanisms

Avoid Cost

Automating infrastructure / Resources during creation lead to less human error and Faster deployments .

Reduce Cost

Automating infrastructure cleanup help reduce infrastructure costs from unused resources

Increase Revenue

- Faster and more frequent production deployments leads to new value-generating features been released more quickly.
- Deploy to production without manual checks in less time to market

Conclusion

In general CI & CD help promote the following :

- Increase accountability and transparency
- Increase customer satisfaction
- Increase team satisfaction as a result of easy maintenance and updates
- Smaller backlog and code changes
- Faster feedback from customers lead to higher customer satisfaction rates since they are involved right from the beginning of feature development/deployment and not just at a fixed release date

Challenges involved in creating CI/CD pipeline development .

- The establishment of a CI/CD pipeline comes with a high amount of initial cost and learning. At first sight this might seem overwhelming compared to current best practices
- Delivery of CI/CD pipelines is not a one time effort, but requires constant support and maintenance as well as continuous development and improvement .