

Continuous Delivery

CI/CD Proposal for *UDAPEOPLE*



1. What is CI/CD

2. Why CI/CD

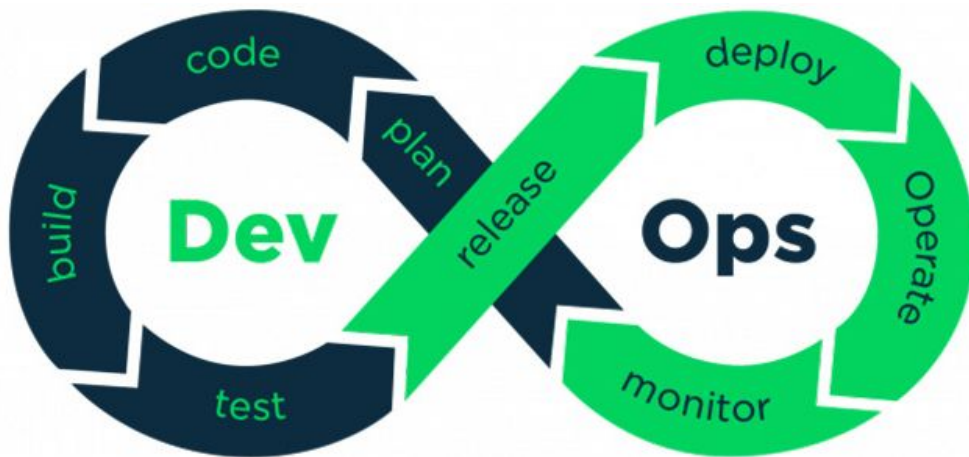
3. CI/CD Tools

Agenda



What is CI/CD

Continuous integration (CI) and **continuous delivery (CD)** are the processes that are used to build, package, and deploy your application. Basically, it lays out some practices to follow in order for the code you write to more quickly and safely get to your users and ultimately generate value.



What is CI/CD

Continuous Integration (CI) is a development practice that helps ensure that software components work together. CI allows you to continuously integrate code into a single shared and easy to access repository.

Continuous delivery (CD) is the ability to deploy your integrated code into production without the need of human intervention. CD allows you to take the code stored in the repository and continuously delivery it to production.

Why CI/CD

- **Reduce costs:** reduce the number of errors that can take place in the many repetitive steps of CI and CD.
- **Smaller code changes:** allows you to integrate small pieces of code at one time. This helps developers to recognize a problem before too much work is completed afterward.
- **Faster release rate:** failures are detected faster and as such, can be repaired faster, leading to increasing release rates.
- **Fault isolations:** ensures that fault isolations are faster to detect and easier to implement.
- **More test reliability:** test reliability improves due to the bite-size and specific changes introduced to the system, allowing for more accurate positive and negative tests to be conducted.

CI/CD Tools



Jenkins



circleci



Bamboo



TeamCity



GitLab

THANK
YOU!
