# Deploying Apps using CI/CD
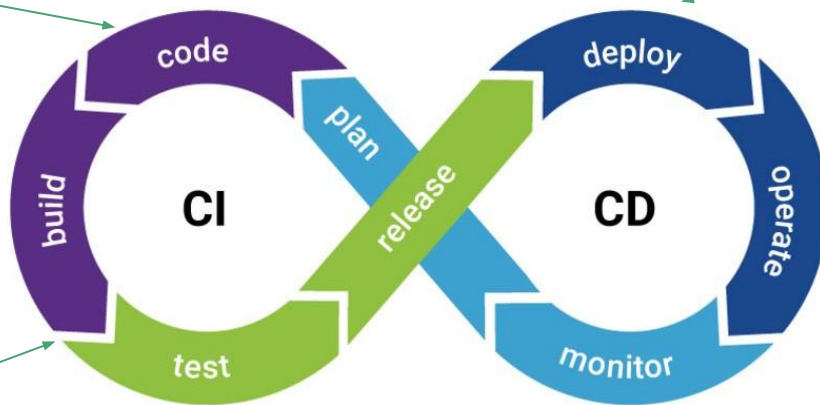
# What is CI/CD?

**CI/CD** is a method to **frequently** deliver apps to customers by introducing **automation** into the stages of app development.

# What is CI/CD?

- Version control
- Peer Code Reviews

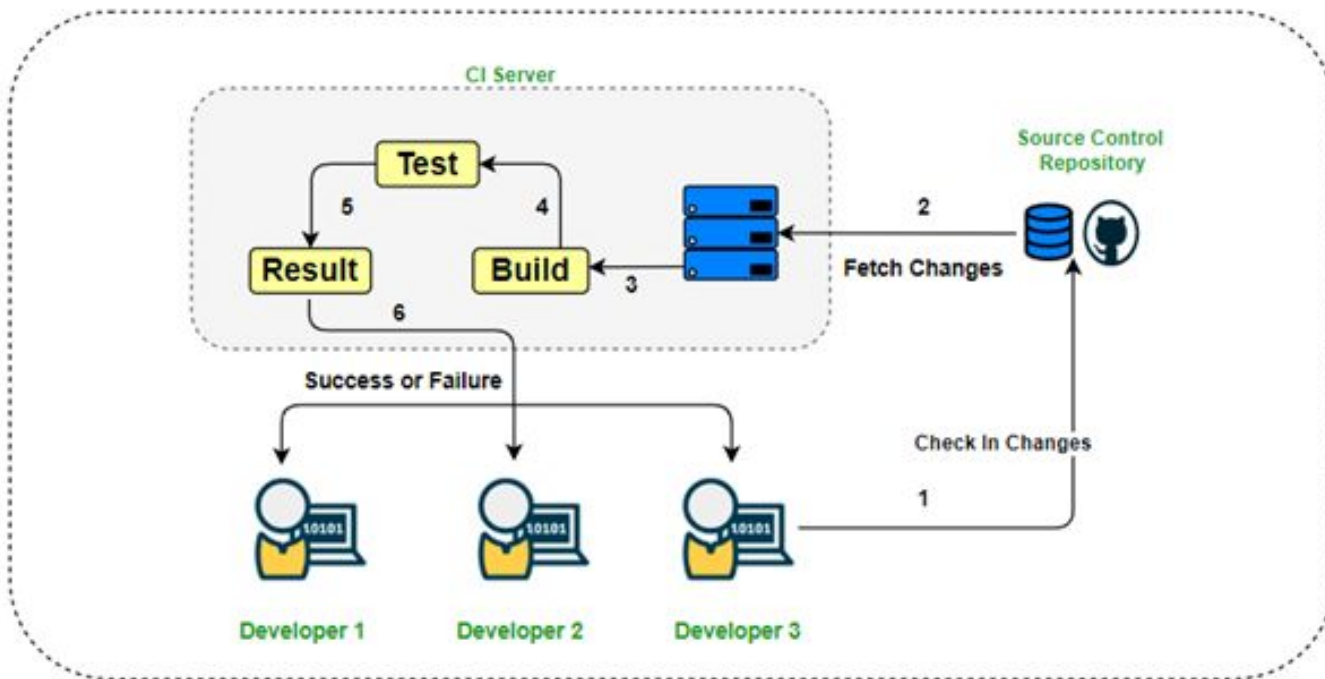- Compile code
- Unit tests
- Style checkers

**Strategies**
- Canaries
- Blue/Green

- Integration tests
- UI Testing

# CI

The "CI" in CI/CD always refers to **C**ontinuous **I**ntegration, which is an **automation** process for **developers**.

Successful CI means new code changes to an app are regularly **built**, **tested**, and **merged** to a shared **repository**.

# CI

# CD

The "CD" in CI/CD refers to **C**ontinuous **D**elivery and/or **C**ontinuous **D**eployment.

These two terms are often considered as same or confused.

Both are about automating further stages of the pipeline, however the **difference** is actually **how much automation is happening**.

# CD

Continuous Delivery means a developer's changes are automatically built, tested and uploaded to a repository, where they can then be deployed to production environment by the operations team (**Manual deployment**).

Continuous Deployment (the other possible "CD") can refer to **automatically** releasing a developer's changes from the repository to production (**Automatic deployment**).
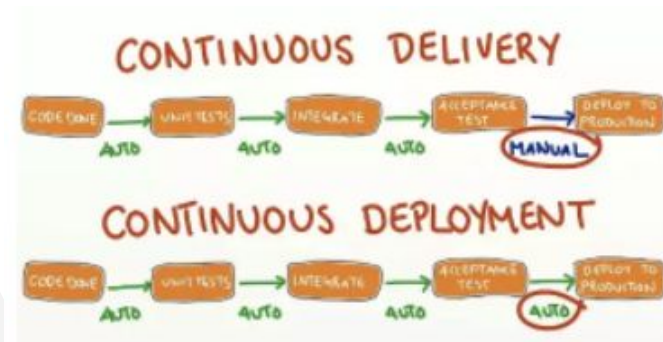
# Release vs Deploy

**Release** is about publishing an artifact (jar, docker image, library) to a binary repository (maven, docker hub, npm).

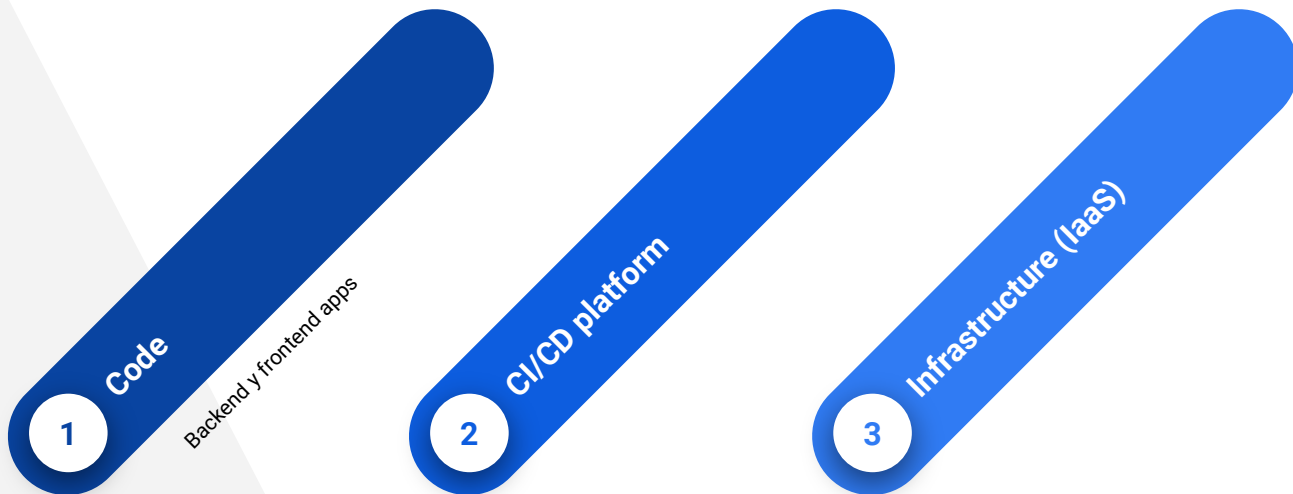**Deploy** is about launching the binary on the **infrastructure**.

*java -jar build/libs/ddd-ecommerce-0.1-all.jar*

# Let's do it

# Basic ingredients

**1** Code
Backend y frontend apps

**2** CI/CD platform

**3** Infrastructure (IaaS)

# Common CI/CD tools

# IaaS

# Create free accounts

- [https://github.com/](https://github.com/)
- [https://www.heroku.com/](https://www.heroku.com/)
- [https://www.netlify.com/](https://www.netlify.com/)
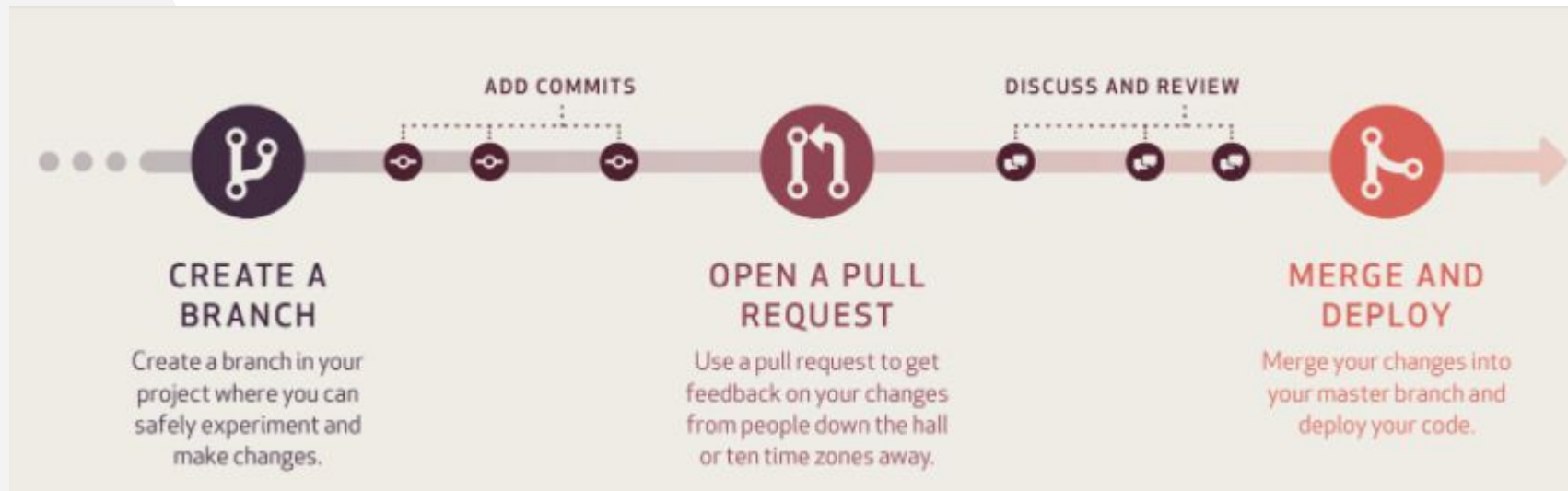
# Setup Backend & Front

- Create a new empty repo (*Fullstack-backend*)
- Clone the repo
- Push our backend code


- Create a new empty repo (*Fullstack-frontend*)
- Clone the repo
- Push our frontend code

# Git Workflow



ADD COMMITS

DISCUSS AND REVIEW

**CREATE A BRANCH**

Create a branch in your project where you can safely experiment and make changes.

**OPEN A PULL REQUEST**

Use a pull request to get feedback on your changes from people down the hall or ten time zones away.

**MERGE AND DEPLOY**

Merge your changes into your master branch and deploy your code.

iTi BOOTCAMP

# THANKS!

**Any questions?**