



# Udapeople: Pulling the trigger(on CI/CD!)

Presentation by: Arantza Flores



# Agenda

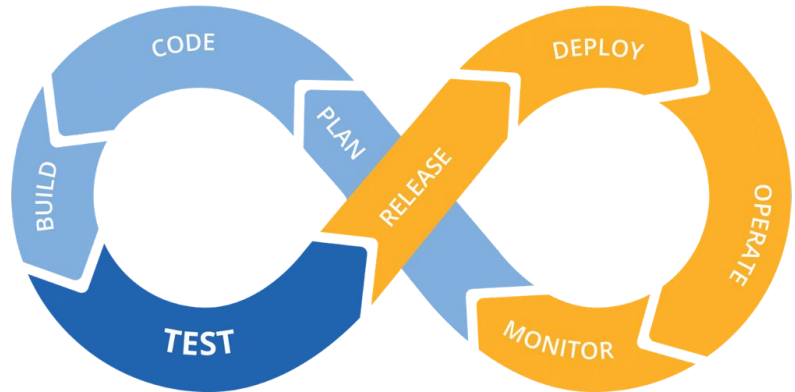
1. What is CI/CD
2. Benefits of CI/CD
3. Cost Benefits
4. Plan Forward

# What is CI/CD?

Our product, Udapeople aims to transform Human Resources for small businesses in a way that's better for their most valuable asset, their people.

And what better way to insure a successful launch and to form a strong relationship with our customers than by making the switch to CI/CD, or Continuous Integration and Continuous Delivery.

CI/CD is the process and practice of automating all aspects of delivering a service, from code pushes to testing and even releases.



# Benefits of CI/CD

- Minimize code conflicts
  - Implementing continuous integration would mean more frequent code pushes and merges to branches, here testing and validation is automated, no more having to wait for other developers to resolve code issues.
- Faster turnaround time for resolving errors
  - Because code is changing so often, we implement validations and testing, this would mean detecting errors and security issues and their source faster. Which then leads to quicker resolutions. Giving developers peace of mind and delivering higher quality code.
- Faster deployments
  - Delivering a working service faster and more often.
  - Newer features go out sooner than they did before, keeping us ahead of the curve and competitive.
  - We can adapt quickly to our customers needs



# Cost Benefits

- Cost Reduction Through Automation
  - Task that were once manual and repetitive, can and should be automated. Our service releases can be build, tested and deployed at almost a push of a button. As a developer that means I can launch this weeks deployment, forget about it and start working on the next feature.
  - Something goes wrong? Quick rollbacks have your back! No more spending money on resources that failed to deploy and sit there until a developer notices.
  - Mitigating human error through testing means fewer bugs get to production. Bugs that would negatively affect the usage of our service are a thing of the past.



# Moving Forward

- Identify the CI/CD tools that align with our product and tools. Some suggestions to start are to move our code to a Gitlab/Github
- Establish best practices between developers
- Training for developers on CI/CD concepts, tools and workflows.
- Gradually implement the 4 stages of CICD pipelines
  - Source - version control of code
  - Build - software builds, containerization, dependencies
  - Test - validation of code and software
  - Deploy - implementing blue-green deployments
- And Finally, release a successful product