

# Jenkins na froncie

i na front'cie





# Arek Gutkowski

[a.gutkowski@outlook.com](mailto:a.gutkowski@outlook.com)  
[github.com/ArkadioG](https://github.com/ArkadioG)

TRENER

info Share  
<academy/>



Teaching Python programming

# COMMUNITY

django girls



# AGENDA

- Continuous Integration / Continuous Delivery
- DevOps
- Jenkins
  - instalacja
  - konfiguracja
  - projekt Jenkins
- Dobre Praktyki

# 1. CI – CD

# Continuous Integration

**CI** – to praktyka mergowania wszystkich gałęzi developerskich do głównej gałęzi kodu co najmniej raz dziennie.

Praktyka XP (eXtreme Programming) postuluje integrowanie kodu nawet kilkanaście razy dziennie

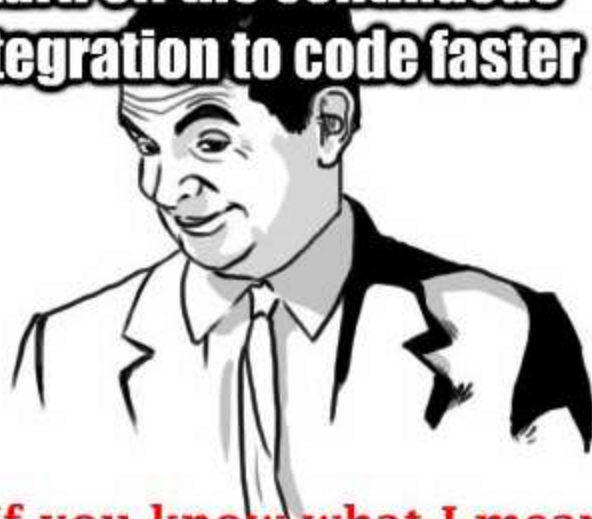
# Continuous Delivery

**CD** - Podejście do wytwarzania oprogramowania, w którym zespół developerski produkuje kod w krótkich cyklach, zapewniając **gotowość** kodu do wdrożenia w każdym czasie.

Osiaga się to poprzez automatyzację budowania, testowania i realese'owania kodu.



**I turn off the continuous  
integration to code faster**



**If you know what I mean**

©2015-2020 infoShare Academy

## 2. DevOps

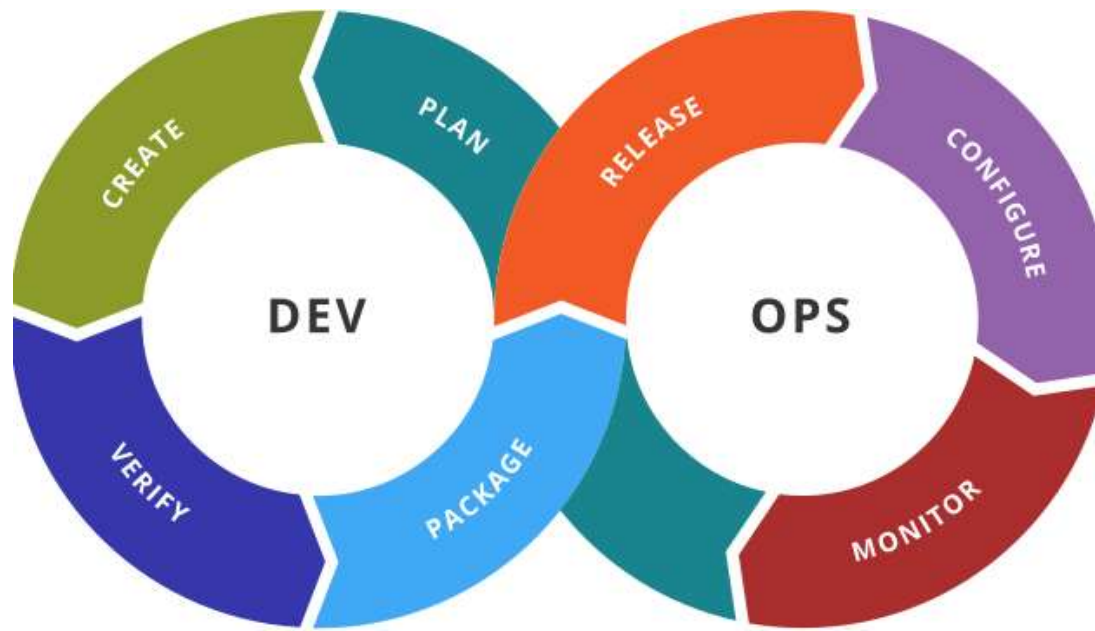
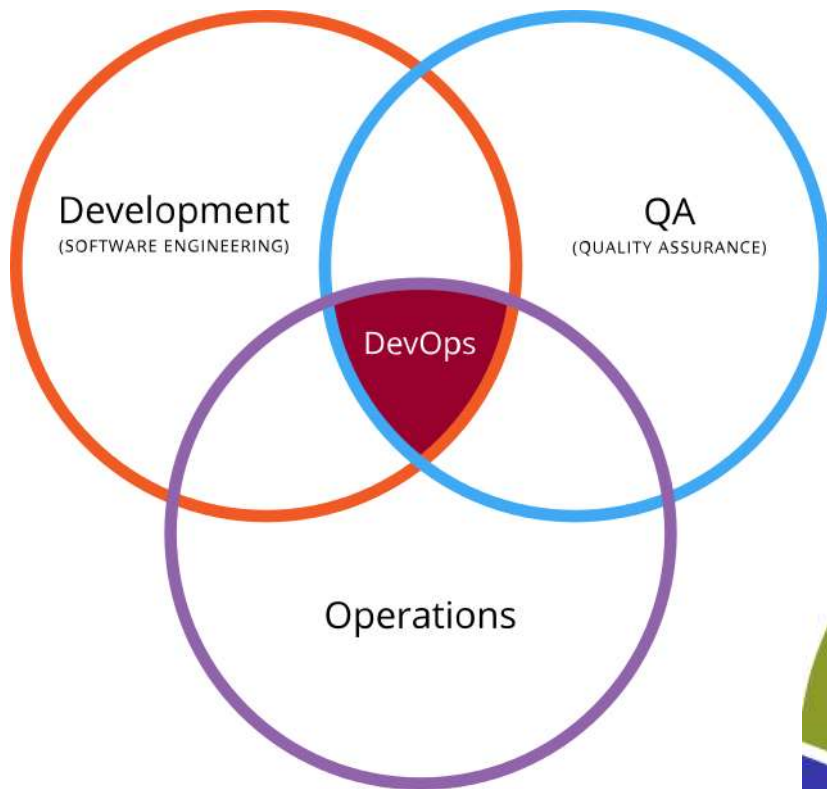
# DevOps

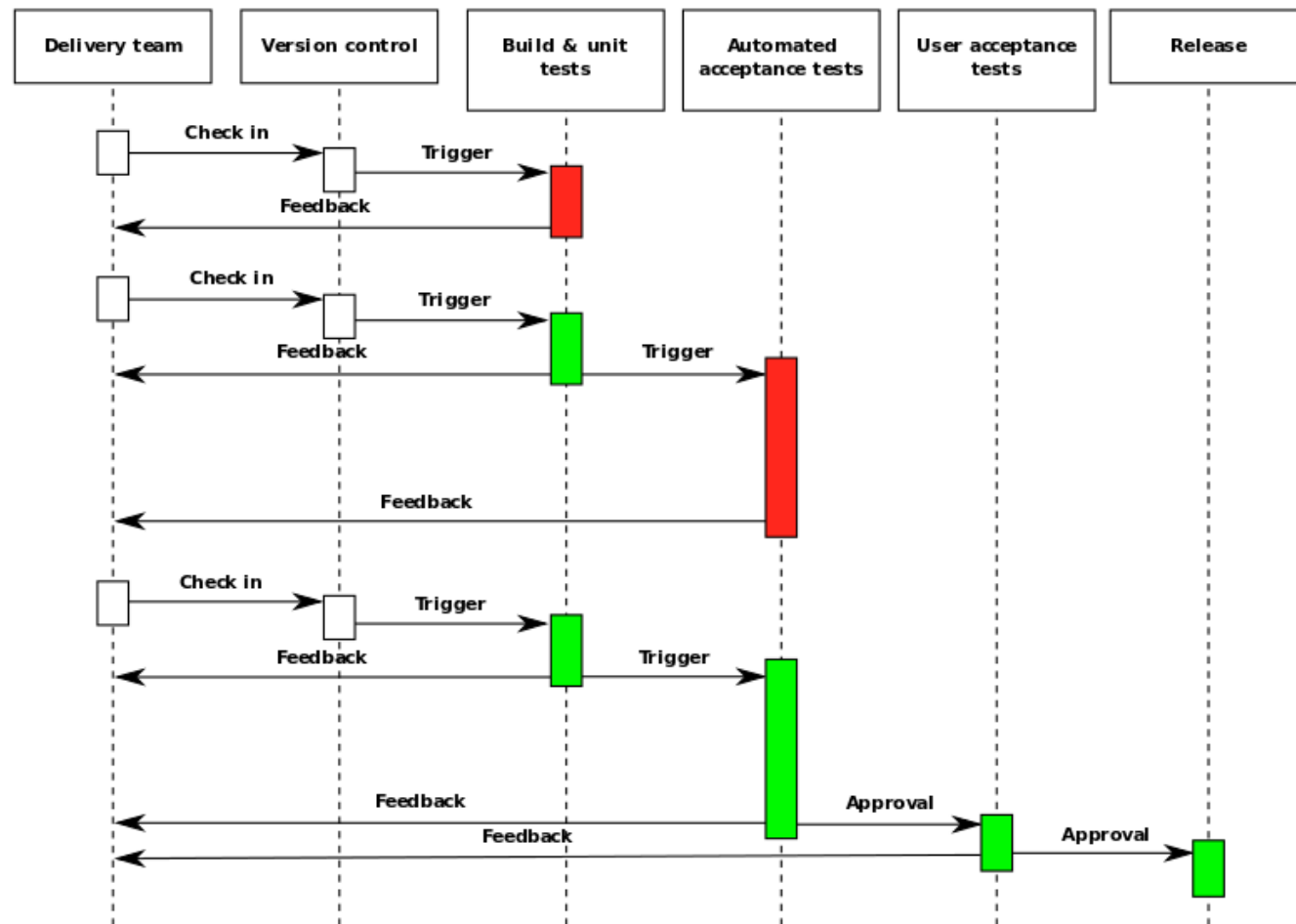
## Development and Operations

Proces wytwarzania kodu, kładący nacisk na komunikację i współpracę między managerami, zespołami developerskimi oraz zespołami operacyjnymi (tech)

Osiąga się to przez automatyzację i monitorowanie procesów integracji, testowania, wydawania (deployment) oprogramowania oraz zmianami infrastruktury.

DevOps to nie tylko proces ale "kultura" organizacji.







## 3. Jenkins

# Przygotowanie

```
git clone https://github.com/ArkadioG/isa-jenkins
```



# Wymagania - Java

- Linux:

polecenia w terminalu

```
java -version          -sprawdzamy wersję  
sudo apt-get install default-jre    - instalacja
```

- Windows:

Pobieramy instalator: <https://www.java.com/pl/download/>

**Uważamy aby odznaczyć śmieci, które proponuje nam zainstalować Oracle!**



# Jenkins - Instalacja

<https://jenkins.io/doc/book/getting-started/installing/>

hard way – wpisujemy ręcznie polecenia:

```
wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add -  
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ >  
/etc/apt/sources.list.d/jenkins.list'  
sudo apt-get update  
sudo apt-get install jenkins
```

**easy way** - uruchamiamy skrypt install\_jenkins.sh pobrany z repo:

```
sudo sh install_jenkins.sh
```

Po instalacji sprawdzamy czy usługa Jenkinsa działa

```
sudo service jenkins status
```

**Folder instalacji:**

```
/var/lib/jenkins/
```

# Jenkins – przed uruchomieniem

Z Jenkinsa korzysta się poprzez przeglądarkę – domyślna instalacja użyje portu **8080**. (localhost:8080 ; 127.0.0.1:8080)

Jeśli na tym porcie już mamy jakąś usługę to należy zmienić konfigurację Jenkinsa aby korzystał z innego portu. Zmieniamy pliki:

**Linux:**

**/etc/default/jenkins**

zmieniamy wartość klucza **HTTP\_PORT=8080**

**Win:**

**c:\...ścieżka...\jenkins.xml**

zmieniamy wartość **-httpPort=8080**

# Jenkins – uruchomienie

## 1

W przeglądarce:

localhost:8080   127.0.0.1:8080   nazwa\_komputera:8080

## Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

# Jenkins – uruchomienie

## 3

Postępujemy zgodnie z poleceniami:

- kopiujemy ze wskazanego pliku hasło do pierwszego logowania
- instalujemy domyślne pluginy
- tworzymy własnego użytkownika – wchodzimy do J

# Jenkins - konfiguracja

konfigurujemy najczęściej w dwóch miejscach

- configure Jenkins - > Global Settings
- configure Jenkins -> Global Tool Options

Przy zdecydowanej większości opcji jest czerwony znak zapytania, wystarczy go kliknąć aby uzyskać odpowiedź.

# Jenkins - Projekt

- Add Project – Freestyle Project, podajemy nazwę – bez spacji!
- konfigurujemy projekt - workspace
- repozytorium – Git – podajemy dane github
- wyzwalacz – Poll SCM
- budowanie
- kroki po budowaniu – publishing, powiadomienia

# Jenkins - **możliwości**

- pobieranie kodu z wielu rodzajów SCM
- zadania mogą uruchamiać się nawzajem, warunkowanie zachowania
- wykonywanie skryptów w różnych językach
- wykonywanie testów, historia testów, code coverage
- generowanie dokumentacji
- prezentacja danych
- powiadomienia (Slack, mail, skype etc.)



## 3. Dobre Praktyki

# Dobre praktyki

- ❑ Jedno repozytorium dla wszystkich, miej strategię branchowania
- ❑ automatyzacja budowania
- ❑ uruchamianie testów automatycznie z buildami
- ❑ wszyscy komitują jak najczęściej (codziennie)
- ❑ każdy commit do głównego brancha powinien być zbudowany
- ❑ optymalizacja procesu CI – ma być szybko!
- ❑ testuj w klonie środowiska dev
- ❑ wizualizuj stan buildów
- ❑ zautomatyzuj deployment



# Thanks!!

Arkadiusz Gutkowski  
[a.gutkowski@outlook.com](mailto:a.gutkowski@outlook.com)  
[github.com/ArkadioG](https://github.com/ArkadioG)