



CONTINUOUS DEPLOYMENT

na przykładzie aplikacji w Django



Wojciech Lichota - STX Next

Marzec 2015



1. Wstęp

2. Workflow

3. Narzędzia

4. Wyzwania



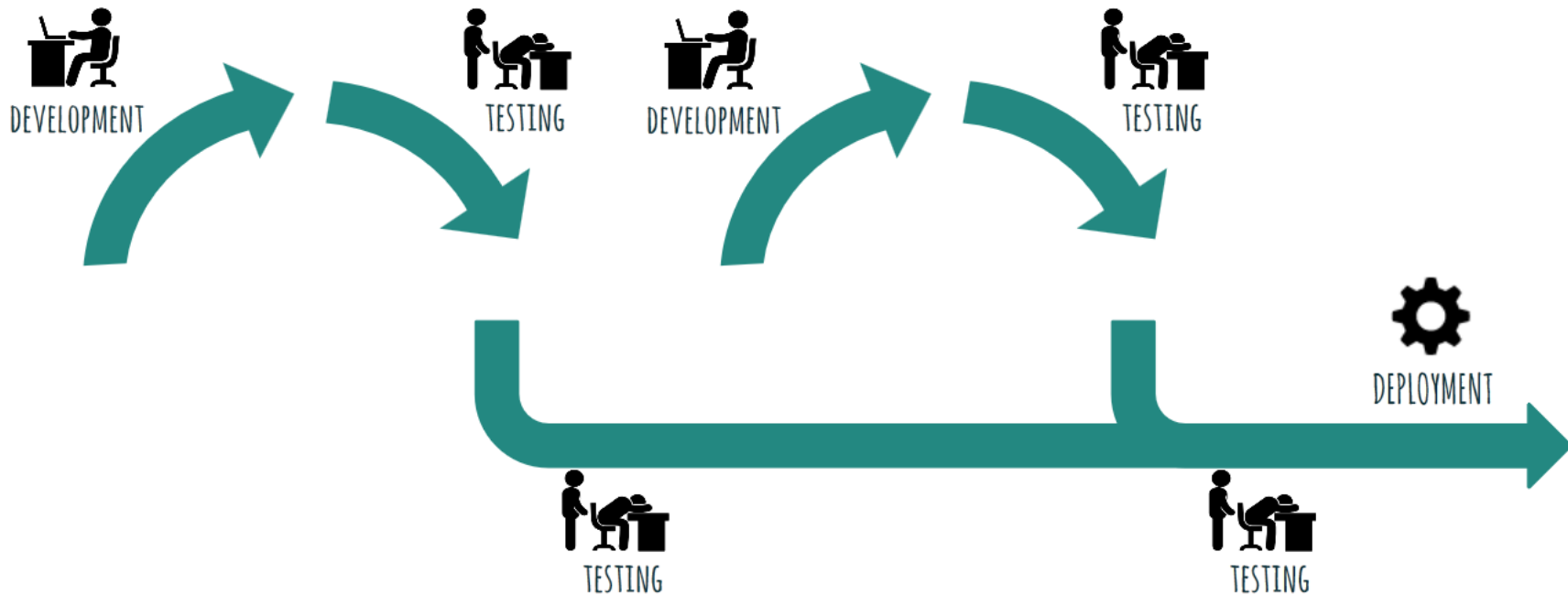
1. Wstęp

2. Workflow

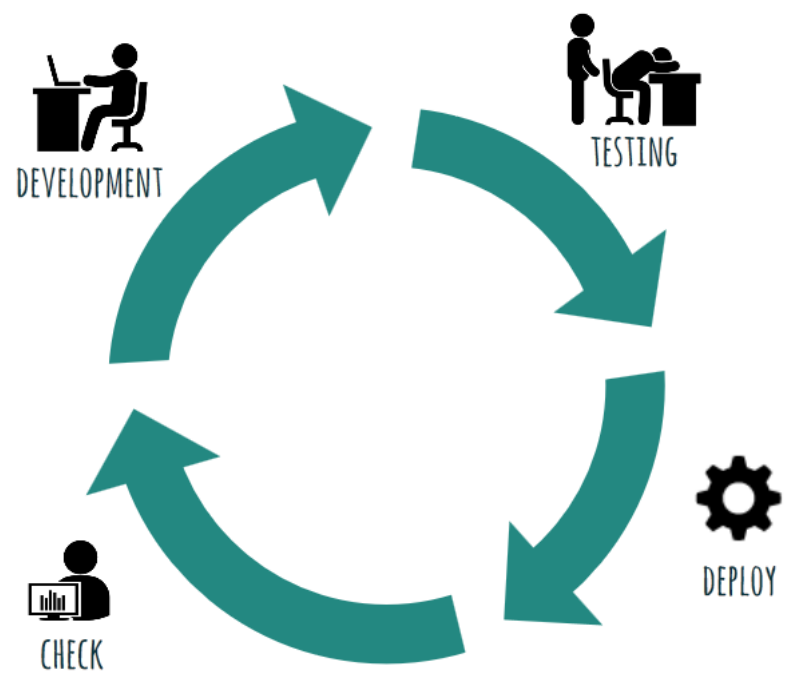
3. Narzędzia

4. Wyzwania

Iterative Deployment



Continuous Deployment



Tags Applicant status All vacancies All cities Age All experiences Current work
Latest activity date Applicant source clear create report

Results: 81

1 2 3 4 5 > »

All (507)

Application (81)

Confirmation (7)

Verification (103)

Interview (95)

Decision (109)

Employment (103)

#	Name	E-Mail / Phone Number	City	Position (Status)	Source	Age	Experience	Creation Date	Latest Activity Date	
1.	Van Hernandez Former Employee django css	test480@stxnext.com 123 880 789	Łódź	Senior Python Developer APPLICATION	Recommendation	39	7	2014-09-30	2014-09-30	2
2.	Nadia Shupe php Employee angular	test478@stxnext.com 123 862 789	Poznań	Front End Developer APPLICATION	Recommendation	25	0	2014-09-30	2014-09-30	1
3.	Marion Bunker php html	test476@stxnext.com 123 179 789	Poznań	QA APPLICATION	Direct search	36	2	2014-09-30	2014-09-30	5
4.	Penny Perze	test465@stxnext.com 123 242 789	Piła	Front End Developer APPLICATION	CV database	39	5	2014-09-30	2014-09-30	1
5.	Patricia Padilla python php html jQuery django	test444@stxnext.com 123 433 789	Łódź	Front End Developer APPLICATION	Job fair	21	3	2014-09-30	2014-09-30	2

STX Next RMS

typeahead.js

Django

CKEditor factory-boy

South Django REST Framework

Bootstrap

python-linkedin

bleach Selectize.js

python-social-auth

Django-REST-Swagger


PostgreSQL

nose AngularJS

Underscore.js

RMS Team


 **jakubjanuzik** #1
470 commits / 18,397 ++ / 9,593 --

 **marek-leszczynski** #2
188 commits / 5,841 ++ / 2,327 --

 **mnowakowska** #3
157 commits / 6,905 ++ / 2,324 --

 **romanlevin** #4
117 commits / 4,480 ++ / 2,291 --

 **dpolcyn** #5
116 commits / 3,950 ++ / 3,667 --

 **KKrisu** #6
113 commits / 20,790 ++ / 20,382 --

 **alubeck** #7
107 commits / 21,785 ++ / 11,395 --

 **ArGroc** #8
95 commits / 6,961 ++ / 4,232 --


 **mrpear** #9
88 commits / 3,340 ++ / 2,390 --

 **ImEagle** #10
82 commits / 3,782 ++ / 1,175 --


 **pkucmus** #11
74 commits / 8,411 ++ / 21,515 --

 **add00** #12
70 commits / 2,593 ++ / 1,341 --

 **mjelonek92** #13
55 commits / 1,789 ++ / 1,068 --

 **lukaszjagodzinski** #14
55 commits / 43,288 ++ / 22,084 --

 **bitrut** #15
42 commits / 1,810 ++ / 947 --

 **avalanchy** #16
38 commits / 11,246 ++ / 1,941 --

 **nihilifer** #17
30 commits / 538 ++ / 136 --

 **Exorades** #18
22 commits / 718 ++ / 835 --

 **romankierzkowski** #19
19 commits / 2,086 ++ / 1,215 --

 **sargo** #20
14 commits / 217 ++ / 129 --

 **Alkemic** #21
10 commits / 339 ++ / 106 --

 **AcidWeb** #22
7 commits / 162 ++ / 407 --

 **Gruszks** #23
7 commits / 348 ++ / 125 --

 **msopocko** #24
1 commit / 16 ++ / 5 --

 **dasm** #25
1 commit / 88 ++ / 84 --



1. Wstęp

2. Workflow

3. Narzędzia

4. Wyzwania

GIT Workflow

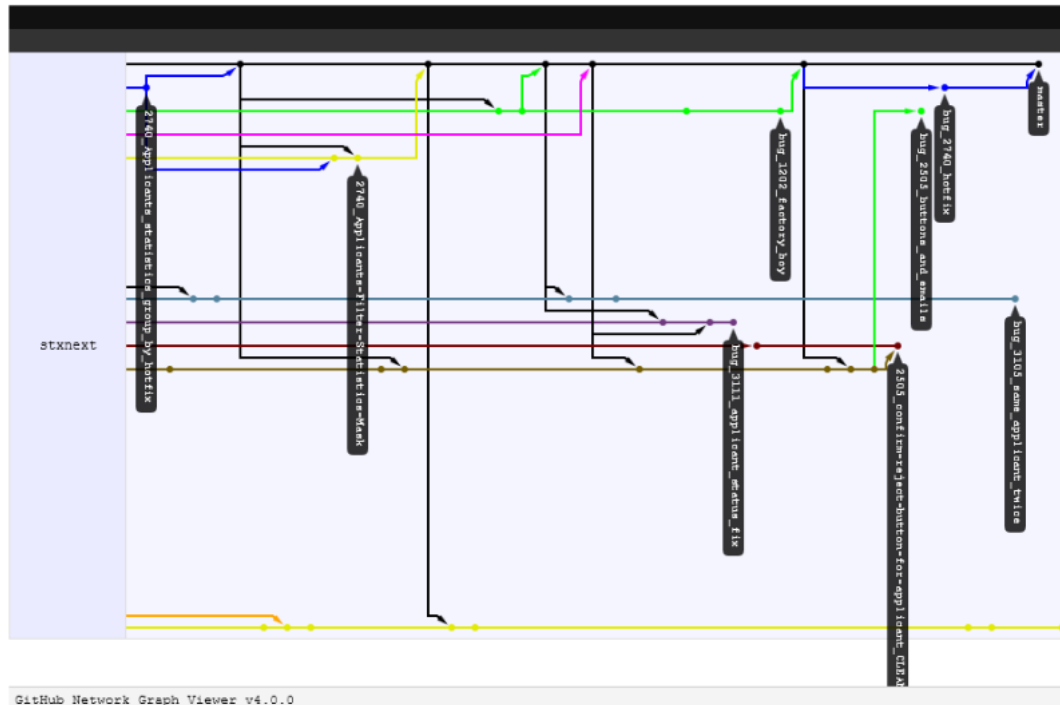
Graph Members

The rms network graph

Keyboard shortcuts available

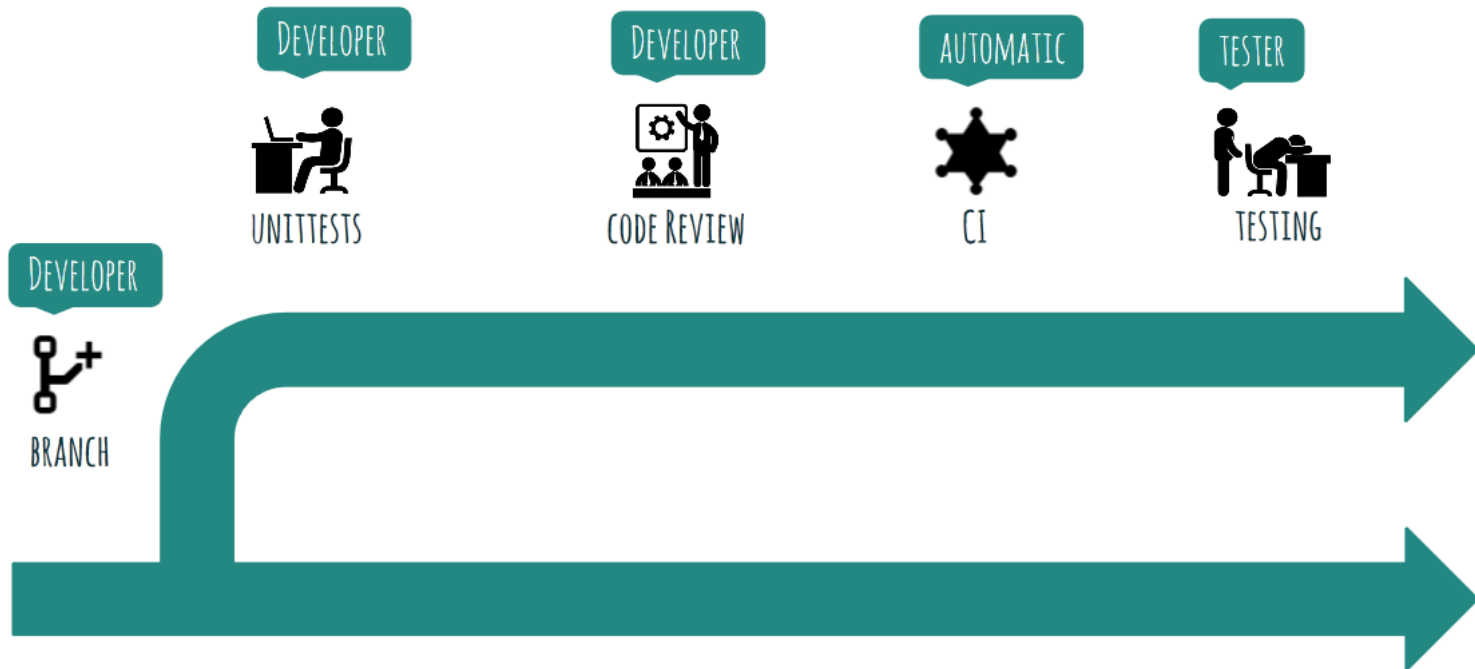
All branches in the network using stxnext/rms as the reference point. Read our blog post about how it works.

Show Help

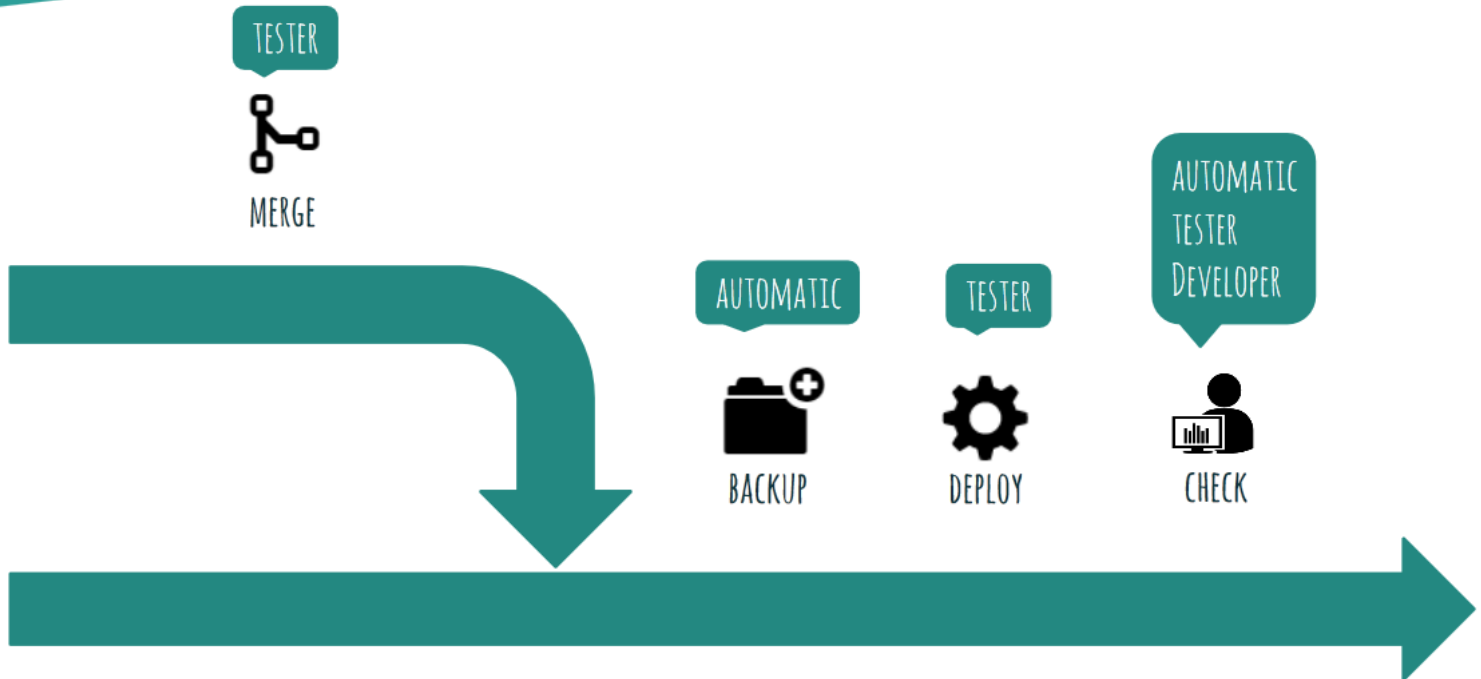


- ★ jeden branch główny (master)
- ★ feature branch (oddzielny branch na każdą historyjkę)
- ★ pull requesty

QA Workflow



Deploy Workflow





1. Wstęp

2. Workflow

3. Narzędzia

4. Wyzwania



JENKINS

continuous
integration

All		RMS				+
S	W	Name ↓	Ostatni Sukces	Ostatni Błąd	Ostatni Czas Trwania	
		RMS-live	1 day 18 hr - #136	—	3 min 43 sec	
		RMS-PR-testing	23 hr - #65	23 hr - #64	1 min 24 sec	
		RMS-staging	1 day 18 hr - #725	—	2 min 21 sec	
		RMS-testing	1 day 18 hr - #10829	—	1 min 58 sec	

Instancje:

- ★ PR-testing - dla każdej z testerek, na feature branch
- ★ staging - z danymi testowymi
- ★ testing - z danymi testowymi (kopia bazy staging)
- ★ live - z danymi produkcyjnymi



JENKINS

continuous
integration

Przydatne wtyczki Jenkinsa

- ★ Violations Plugin - pep8, pylint
 - ★ Post build task Plugin - kroki po prawidłowym buildzie (np. uruchomienie Fabrica)
 - ★ Matrix Authorization Strategy Plugin - uprawnienia (np. kto może wrzucać na produkcję)
 - ★ Github Plugin - automatyczne tagowanie
-

Dodatki do Django wspomagające integrację z Jenkinsem:

- ★ django-nose - raporty odpalenia unittestów (opcja --with-xunit)
- ★ nosecover - raport pokrycia testami (opcja --with-xcover)



BUILDOUT



build system

- ★ Tworzy reprodukowalne środowisko
- ★ Odseparowane sandboxy bez konieczności ich aktywacji/deaktywacji
- ★ Pozwala na uproszczenie komend (opcje uruchomienia w konfiguracji buildouta)

Przydatne recepty:

- ★ `djangorecipe`
- ★ `collective.recipe.template`
- ★ `buildout.recipe.uwsgi`
- ★ `mr.developer`



BOWER
★
GRUNT

Bower - zarządzanie zależnościami dla frontend

Grunt - wykonywanie operacji na zasobach dla frontend

Przykładowe taski Grunta:

- ★ łączenie plików JS/CSS
- ★ kopiowanie plików (np. obrazków z bibliotek do projektu)
- ★ kompresja JS/CSS
- ★ obfuscacja JS (tylko live)
- ★ uruchomienie jshint, compass



FABRIC



remote
execution

- ★ Spięcie kilku komend (np. odpalenie buildout, grunta, restarty serwerów) w jedno polecenie
- ★ Uruchomienie wybranych komend na zdalnych serwerach (poprzez ssh)



GUNNERY

remote
execution

- ★ alternatywa dla Jenkins + Fabric
- ★ zarządzanie poprzez przeglądarkę różnymi instancjami na różnych serwerach

GUNNERY

VW UFO
development

wojciech.lichota
Settings
Help
Log Out

Environment *development*

Dashboard / VW UFO / development

Activity

development /
update_devel_WIOLETTA
Wioletta 18 minutes ago

development /
update_devel_KAROLINA
Karolina 36 minutes ago

development /
update_devel_PIOTR
Piotr 2 hours ago

development /
update_devel_WIOLETTA
Wioletta 3 hours ago

Servers +

stx_dev	10.93.1.30	app		
---------	------------	-----	--	--

Tasks +

update_devel_WIOLETTA		
update_devel_KAROLINA		
update_devel_PIOTR		
update_devel_MARTA		
update_VWL_FO_CLIENT		



1. Wstęp

2. Workflow

3. Narzędzia

4. Wyzwania

Restarty aplikacji

Problem

Kilka instancji na wspólnym **Apache i mod_wsgi**.
Restart jednej instancji wpływa na chwilową niedostępność innych.

Rozwiązanie

Osobny **uWSGI** dla każdej instancji.

PROTIP

Użyj **Master FIFO** do zarządzania procesem **uWSGI**.

```
while lsof uwsgi_ctl; do
    echo Q > uwsgi_ctl
done
```

Baza danych

Problem

Dwa otwarte pull requesty i w obu migracja przez **South**.
Utrzymywanie wstecznych migracji, brak konfliktów.

Rozwiązanie

Instancja PR-testing zawsze na świeżej **kopii** bazy danych.
Plik migrations/version.txt z ostatnim numerem migracji.

PROTIP

```
# drop all tables
psql rms_pr_testing -t -c "select 'drop table \'' || tablename || '\'' cascade;'
from pg_tables where schemaname = 'public'" | psql rms_pr_testing
# copy tables
pg_dump rms_testing | psql rms_pr_testing
```

Revert na live

Problem

Jeżeli wrzutka na **live** spowoduje błędy, potrzebny jest szybki sposób przywrócenia i uruchomienia poprzedniej wersji.

Rozwiązanie

- ★ Tagowanie kodu
- ★ Backup bazy danych
- ★ wrzutki tylko w godzinach gdy admin jest “pod ręką”
- ★ Automatyzacja rewertu

PROTIP

```
# backup db  
pg_dump -C rms > ~/rms_backup_$(date +"%Y%m%d%H%M").psql
```

Używaj narzędzia do monitoringu - np. New Relic, Nagios.

Podsumowanie

- ★ Przedstawione rozwiązanie wystarczające dla niekrytycznych aplikacji
- ★ Szybciej nowa rzecz na produkcji
- ★ Większa ilość pracy dla testera, mniej dla dewelopera
- ★ Zawsze zakładaj problemy (jeżeli coś się może zepsuć - to na pewno się zepsuje)



Wojciech Lichota



STX Next

wojciech@lichota.pl

[@wlichota](#)

<http://lichota.pl>

STX NEXT

OVER 100 TALENTED DEVELOPERS...

Employing over 100 Python programmers, STX Next is one of the largest software development houses in central Europe. We are goal-oriented, agile, and fast-paced.



Pila

Poznań

Wrocław

... IN 3 DIFFERENT LOCATIONS...

We currently have 3 locations in Poland. We know that we are as strong as the people working for us, so we try to maintain a work environment based around honesty, trust, and hard work instead of management and bureaucracy.

... CREATE AWESOME PRODUCTS.

We develop Web Applications in **Python** and **Javascript** and mobile apps for **Android** and **iOS**. We use **agile** methods and continuous delivery. We do projects for clients all over the world, and see how people use our software.

