

# Deploy Django em EBS pelo Jenkins

PyCon Amazônia 2017



# Quem sou eu



Julio Melanda

Bacharel em Ciência da  
Computação

Programo em Python desde 2007

[www.programeempython.com.br](http://www.programeempython.com.br)

Toptal

@jcemelanda

jcemelanda@gmail.com



# Objetivos

- Mostrar como fazer um deploy simples usando Jenkins e AWS Elastic Beanstalk
  - Configurar o projeto Django para rodar no EBS
  - Configurar o Jenkins para fazer deploy no EBS
  - Configurar EBS para um projeto Django

# O Projeto Django

- Projeto Django simples
- Pasta `.ebextensions`
- `django.config`

# django.conf

option\_settings:

aws:elasticbeanstalk:container:python:

WSGIPath: palestra/wsgi.py

aws:elasticbeanstalk:application:environment:

DJANGO\_SETTINGS\_MODULE: palestra.settings

container\_commands:

database\_migration:

command: "python manage.py migrate

--noinput"

static\_files:

command: "python manage.py collectstatic

--noinput -v 0"



# Elastic Beanstalk

- O AWS Elastic Beanstalk é um serviço de fácil utilização para implantação e escalabilidade de aplicativos e serviços da web desenvolvidos com Java, .NET, PHP, Node.js, Python, Ruby, Go e Docker em servidores conhecidos, como Apache, Nginx, Passenger e IIS. (Site da AWS)
- Heroku
- Openshift

# Elastic Beanstalk

- Versionamento dos deploys no AWS
- Diferentes environments

# Criando Aplicação

Create New Application ×

**Application Name**

Maximum length of 100 characters, not including forward slash (/).

**Description**

Maximum length of 200 characters.

[Cancel](#) [Create](#)



# Criando Ambiente

## Create a new environment

---

Launch an environment with a sample application or your own code. By creating an environment, you allow AWS Elastic Beanstalk to manage AWS resources and permissions on your behalf. [Learn more](#)

### Environment information

---

Choose the name, subdomain, and description for your environment. These cannot be changed later.

**Application name** palestra

**Environment name**

**Domain**

**Description**

# Criando Ambiente (cont.)

## Base configuration

---

**Tier** Web Server ([Choose tier](#))

**Platform**  Preconfigured platform

Platforms published and maintained by AWS Elastic Beanstalk.

Python ▼

Custom platform <sup>NEW</sup>

Platforms created and owned by you. [Learn more](#)

– Choose a custom platform – ▼

**Application code**  Sample application

Get started right away with sample code.

# Jenkins

Jenkins é um sistema de Integração Contínua.

Ele é usado para contruir projetos automaticamente, executar seus testes e fazer deploy de aplicações.

# Plugins do Jenkins

- AWS Elastic Beanstalk Publisher Plugin
- Github Plugin
- Jenkins Git Plugin


# Configurações Gerais


**General** Gerenciamento de código fonte Trigger de builds Ambiente de build Build Ações de pós-build

Nome do Projeto


Descrição


[HTML escapado] [Visualizar](#)


Descartar builds antigos 


Este build é parametrizado 


GitHub project

Project url  

Throttle builds 

Desabilitar builds 

Execute os builds se necessário 

 **Avançado...**

**Avançado...**

# Repositório

**Gerenciamento de código fonte**

Nenhum

Git

Repositories

Repository URL  ?

Credentials   ?






Branches to build

Branch Specifier (blank for 'any')   ?


Navegar no repositório  ?

# Build

## Trigger de builds

- Dispare builds remotamente (exemplo, a partir dos scripts) 
- Construir após a construção de outros projetos 
- Construir periodicamente 
- GitHub hook trigger for GITScm polling 
- Consultar periodicamente o SCM 

## Ambiente de build

- Delete workspace before build starts
- Abort the build if it's stuck
- Add timestamps to the Console Output
- Use secret text(s) or file(s) 

# Deploy

## Ações de pós-build

### Deploy into AWS Elastic Beanstalk

#### Application Setup

#### Elastic Beanstalk Application

AWS Credentials

palestra : AKIAJVVR52AFNU4XHGCQ

AWS Credentials lookup by name

Use a credentials name that will be resolved during the build.

(de [AWS Elastic Beanstalk Publisher Plugin](#))

Get Credentials Names

AWS Region

us-east-1

AWS Region Text

us-east-1

Application Name

palestra

Get Available Applications



# Deploy (cont.)

EnvironmentLookup

**Get Environments By Name** X

Environment Names  ?

▾

Version Label Format  ?

Fail if any failures  ?

Additional Behaviors

**Deploy to S3** X

S3 Bucket Name  ?

S3 Key Prefix  ?

S3 Overwrite existing file  ?

# Deploy (cont.)

Fail if any failures

Additional Behaviors

**Deploy to S3** X

S3 Bucket Name  ?

S3 Key Prefix

S3 Overwrite existing file  ?

Root Object (File / Directory)  ?

S3 Bucket Region  ?

Includes  ?

Excludes  ?

Adicionar ▼

Adicionar ▼

# Configurando Github

Options

Collaborators

Branches

Webhooks

**Integrations & services**

Deploy keys

Services / **Manage Jenkins (GitHub plugin)**

Test service

Jenkins is a popular continuous integration server.

Using the Jenkins GitHub Plugin you can automatically trigger build jobs when pushes are made to GitHub.

## Install Notes

1. "Jenkins Hook Url" is the URL of your Jenkins server's webhook endpoint. For example: `http://ci.jenkins-ci.org/github-webhook/`.

For more information see <https://wiki.jenkins-ci.org/display/JENKINS/GitHub+plugin>.

## Jenkins hook url

`http://ec2-54-84-234-0.compute-1.amazonaws.com:8080/github-wet`

**Active**

We will run this service when an event is triggered.

Update service

Delete service

# Fazendo Deploy

```
git push origin master
```

# Dúvidas?