

# **GRADUAÇÃO EM REDE DE** **COMPUTADORES**



*Projeto de Bloco - Arquitetura e Infraestrutura de Aplicações [18E3\_5] – Teste de Performance – TP03*

*Por: Gustavo de Alcantara Amaral*

*Aluno do Curso RDC, Turma 1, Manhã*

*Professor: Alexandre Carneiro*

*Professor orientador de Projeto de Bloco: Alexandre Carneiro*

*Tutor:*

*Rio de Janeiro*

*02/09/2018*

## Sumário

TP03 – Introdução:.....	3
Descrição .....	3
TP03 – Parte 1: .....	3
TP03 – Rubrica (Competência):.....	7
TP03 – Referências : .....	8

## TP03 – Introdução:

### Descrição

#### PARTE I - PRÁTICA

##### TP02

Escreva um playbook Ansible que instale a aplicação escolhida por você para realização do TP1. Seu playbook deve automatizar o máximo possível de passos de instalação e configuração, e deve incluir comentários (e diretivas "name") que permitam compreender facilmente o que está sendo realizado.

Caso algum passo do processo não possa ser documentado, sem problemas, mas é preciso que você inclua em seu trabalho uma justificativa e uma pequena descrição desses passos, exatamente como faria em um projeto do mundo real. Por fim, você deve incluir capturas de tela que mostrem uma aplicação bem-sucedida do seu playbook.

##### TP03

Documente em um pequeno relatório a criação de um repositório público para armazenar o playbook desenvolvido no TP da etapa 2. Você também precisa capturar as telas do processo de inicialização, do primeiro commit e do primeiro push do seu código. Faça também uma alteração simples no seu código (pode ser até mesmo a modificação de um README.md) e documente o processo de commit e push. Por fim, inclua no seu relatório o link para seu repositório, que será consultado pelo professor e precisa refletir as operações documentadas em seu texto.

## TP03 – Parte 1:

#### PARTE I - PRÁTICA

##### Introdução:

Este documento tem o propósito de demonstrar a aplicabilidade de soluções distribuídas que serão implementadas e utilizadas futuramente em soluções de virtualização.

##### Objetivos:

Este documento tem como principal objetivo colocar em prática a utilização da ferramenta de automação de rotinas e comandos conhecida como Ansible.

Para alcançar este objetivo será demonstrada a instalação do Wordpress em um servidor com o sistema operacional Ubuntu 16.4 (recomendado pela orientação acadêmica).

##### Desenvolvimento:

#### 1- Exemplo de playbook a ser utilizado no Ansible:

```
- hosts: wordpress
  remote_user: gustavo

  become: yes

  roles:
    - server
    - php
    - mysql
```

- wordpress


## 2- Diretório criado utilizando o Ansible Galaxy:

```
gustavoa@ubuntu:~/playbooks$ cd..
cd..: command not found
gustavoa@ubuntu:~/playbooks$ cd ..
gustavoa@ubuntu:~$ ls
Desktop  Documents  Downloads  examples.desktop  Music  Pictures  playbooks  Public  Templates  Videos
gustavoa@ubuntu:~$ mkdir wordpress-ansible
gustavoa@ubuntu:~$ cd wordpress-ansible
gustavoa@ubuntu:~/wordpress-ansible$ touch playbook.yml
gustavoa@ubuntu:~/wordpress-ansible$ touch hosts
gustavoa@ubuntu:~/wordpress-ansible$ mkdir roles
gustavoa@ubuntu:~/wordpress-ansible$ cd roles
gustavoa@ubuntu:~/wordpress-ansible/roles$ ansible-galaxy init server
- server was created successfully
gustavoa@ubuntu:~/wordpress-ansible/roles$ ansible-galaxy init php
- php was created successfully
gustavoa@ubuntu:~/wordpress-ansible/roles$ ansible-galaxy init mysql
- mysql was created successfully
gustavoa@ubuntu:~/wordpress-ansible/roles$
```

Figura 01 – Utilizando o Ansible Galaxy

## 3- Playbox 100% funcional e implementado no Ansible:

```
gustavoa@ubuntu:~/wordpress-ansible/roles$ gedit ~/wordpress-ansible/hosts
```

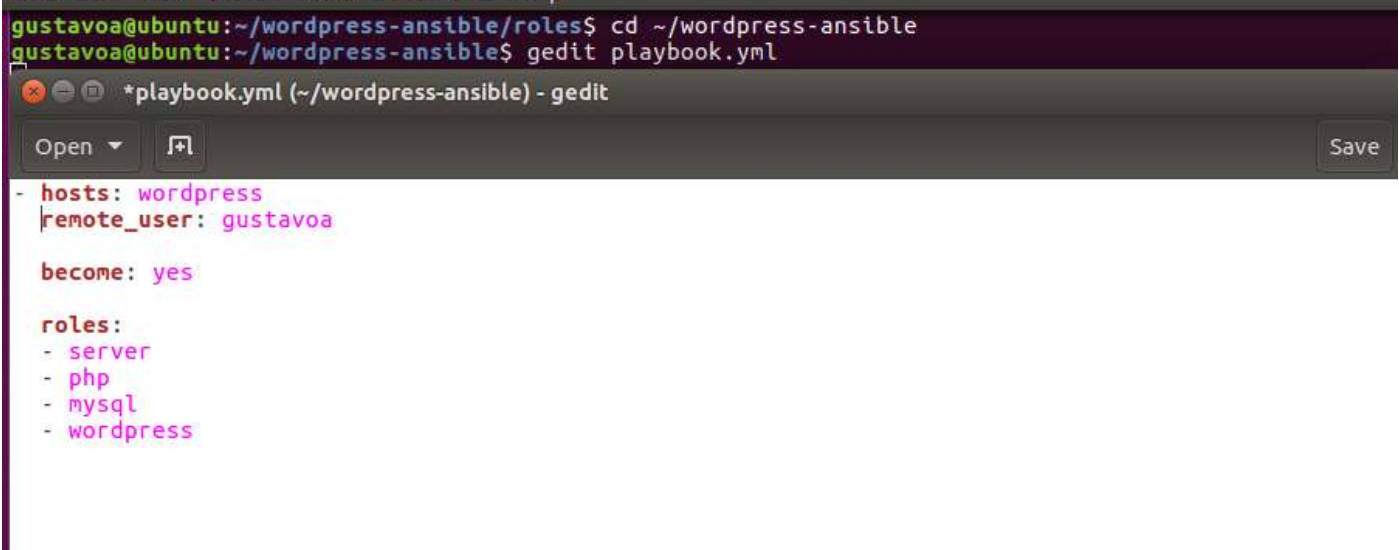


The screenshot shows the gedit editor window titled '\*hosts (~/.wordpress-ansible) - gedit'. The file content is:

```
[wordpress]
127.0.0.1
```

Figura 02 – Preenchendo o arquivo hosts:

```
File Edit View Search Tools Documents Help
gustavoa@ubuntu:~/wordpress-ansible/roles$ cd ~/wordpress-ansible
gustavoa@ubuntu:~/wordpress-ansible$ gedit playbook.yml
```



The screenshot shows the gedit editor window titled '\*playbook.yml (~/.wordpress-ansible) - gedit'. The file content is:

```
- hosts: wordpress
  remote_user: gustavoa

  become: yes

  roles:
    - server
    - php
    - mysql
    - wordpress
```

Figura 03 – Preenchendo o playbook:

#### 4- Criação do repositório para utilização do GitHub:

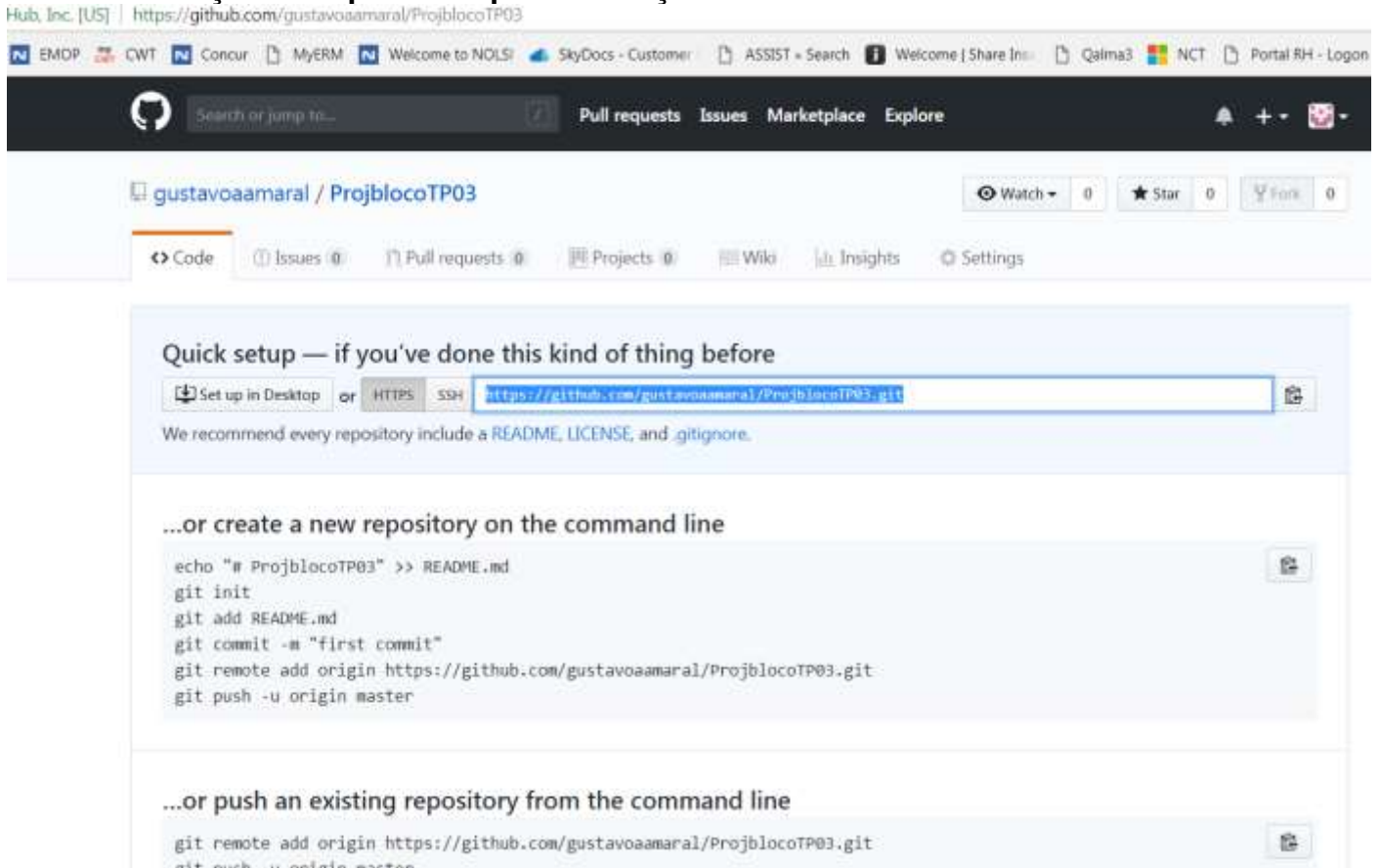


Figura 03 - <https://github.com/gustavoamaral/ProjblocoTP03.git>

#### 5- Demonstração do Commit e Push do código:

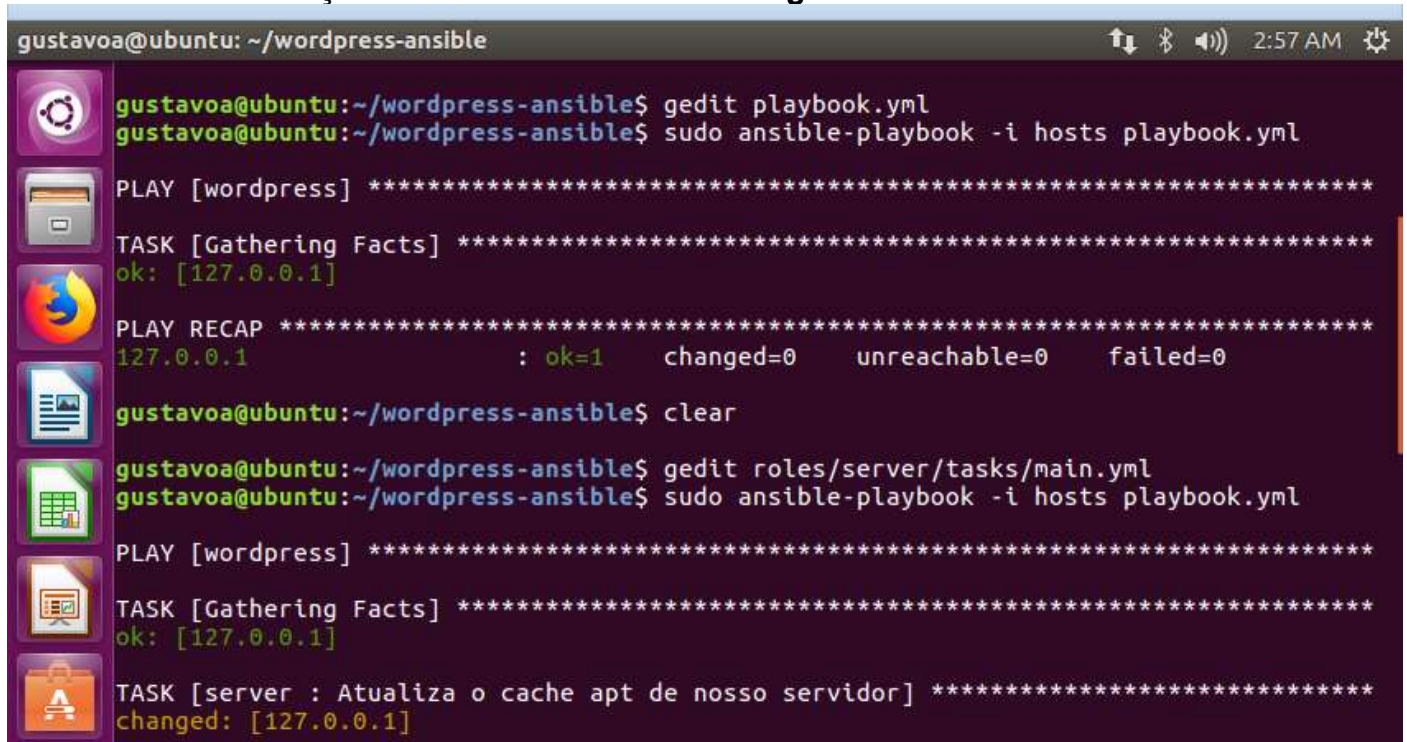


Figura 04 – Código em funcionamento



```

gustavoa@ubuntu:~/wordpress-ansible$ git push -u origin master
Username for 'https://github.com': gustavoaamaral
Password for 'https://gustavoaamaral@github.com':
Counting objects: 54, done.
Compressing objects: 100% (24/24), done.
Writing objects: 100% (54/54), 5.50 KiB | 0 bytes/s, done.
Total 54 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/gustavoaamaral/ProjblocoTP03.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
gustavoa@ubuntu:~/wordpress-ansible$

```

Figura 05 – Push funcionando

```

gustavoa@ubuntu:~/wordpress-ansible$ git commit -m "importando playbook WordPress"
[master (root-commit) 337ec62] importando playbook WordPress
35 files changed, 499 insertions(+)
create mode 100755 hosts
create mode 100644 playbook.retry
create mode 100644 playbook.yml
create mode 100644 roles/mysql/README.md
create mode 100644 roles/mysql/defaults/main.yml
create mode 100644 roles/mysql/handlers/main.yml
create mode 100644 roles/mysql/meta/main.yml
create mode 100644 roles/mysql/tasks/main.yml
create mode 100644 roles/mysql/tests/inventory
create mode 100644 roles/mysql/tests/test.yml
create mode 100644 roles/mysql/vars/main.yml
create mode 100644 roles/php/README.md
create mode 100644 roles/php/defaults/main.yml
create mode 100644 roles/php/handlers/main.yml
create mode 100644 roles/php/meta/main.yml
create mode 100644 roles/php/tasks/main.yml
create mode 100644 roles/php/tests/inventory
create mode 100644 roles/php/tests/test.yml
create mode 100644 roles/php/vars/main.yml
create mode 100644 roles/server/README.md
create mode 100644 roles/server/defaults/main.yml
create mode 100644 roles/server/handlers/main.yml
create mode 100644 roles/server/meta/main.yml
create mode 100644 roles/server/tasks/main.yml
create mode 100644 roles/server/tests/inventory
create mode 100644 roles/server/tests/test.yml
create mode 100644 roles/server/vars/main.yml
create mode 100644 roles/wordpress/README.md
create mode 100644 roles/wordpress/defaults/main.yml
create mode 100644 roles/wordpress/handlers/main.yml
create mode 100644 roles/wordpress/meta/main.yml
create mode 100644 roles/wordpress/tasks/main.yml
create mode 100644 roles/wordpress/tests/inventory
create mode 100644 roles/wordpress/tests/test.yml
create mode 100644 roles/wordpress/vars/main.yml

```

Figura 06 – Commit funcionando

## 6- Relatório de instalação:

- a. Etapa 01 – Criar a pasta wordpress-ansible:
  - # mkdir wordpress-ansible
  - # cd wordpress-ansible

- b. Etapa 02 – Criar o arquivo “playbook” e o arquivo “hosts” dentro do diretório wordpress-ansible:  
 # touch playbook.yml  
 # touch hosts
- c. Etapa 03 – Criar as “roles” utilizando o Ansible Galaxy:  
 # ansible-galaxy init server  
 # ansible-galaxy init php  
 # ansible-galaxy init mysql  
 # ansible-galaxy init wordpress
- d. Etapa 04 – Configurar o arquivo “hosts”:  
 # gedit ~/wordpress-ansible/hosts
- e. Etapa 05 – Dentro do arquivo “hosts” inserir:  
 [wordpress]  
 127.0.0.1
- f. Etapa 06 – Inserindo os comandos dentro do playbook do Ansible:  
 # cd ~/wordpress-ansible  
 # gedit playbook.yml
- g. Etapa 07 – Conteúdo do playbook do Ansible:  
 - hosts: wordpress  
 remote\_user: gustavo  
  
 become: yes  
  
 roles:  
 - server  
 - php  
 - mysql  
 - wordpress
- h. Etapa 08 – Execução do playbook para primeiro teste:  
 # ansible-playbook -i hosts playbook.yml

## 7- Conclusão:

Com os tópicos propostos neste documento, espera-se que todos objetivos propostos no mesmo, sejam alcançados, e que a implementação desta solução esteja totalmente funcional de acordo com as premissas abordadas.

## TP03 – Rubrica (Competência):

TP2  
 2. Utilizar ferramentas de automação de instalação de aplicações  
 O aluno produziu um playbook válido, ainda que não tenha conseguido automatizar toda a configuração da aplicação?

TP2

2. Utilizar ferramentas de automação de instalação de aplicações

O aluno conseguiu aplicar seu playbook com sucesso usando a ferramenta ansible-playbook?

TP2

2. Utilizar ferramentas de automação de instalação de aplicações

O aluno produziu um playbook de acordo com as melhores práticas, com diretórios inicializados pela ferramenta ansible-galaxy e um arquivo de inventário específico para seu projeto?

TP2

2. Utilizar ferramentas de automação de instalação de aplicações

O aluno conseguiu produzir um playbook capaz de automatizar com sucesso 100% do processo de instalação e configuração da aplicação pretendida?

TP3

3. Documentar um projeto de infraestrutura para aplicações

O aluno demonstrou a criação de um repositório corretamente?

TP3

3. Documentar um projeto de infraestrutura para aplicações

O aluno conseguiu fazer o commit e o push do seu código corretamente?

TP3

3. Documentar um projeto de infraestrutura para aplicações

O aluno documentou o processo de alteração, novo commit e push do código?

TP3

3. Documentar um projeto de infraestrutura para aplicações

A URL de repositório fornecida pelo aluno tem um conteúdo consistente com os passos apresentados no relatório?

## TP03 – Referências :

Acesso em: 10/08/2018

- <https://lms.infnet.edu.br/moodle/mod/page/view.php?id=94736>
- <https://lms.infnet.edu.br/moodle/mod/assign/view.php?id=94738>
- <https://lms.infnet.edu.br/moodle/mod/assign/view.php?id=94746>
- <https://www.youtube.com/watch?v=o3Cz1jyEahE>
- <http://helton.de/Blog/article/pt/267>
- <https://lms.infnet.edu.br/moodle/mod/page/view.php?id=94744>
- <https://www.ansible.com/resources/videos/quick-start-video>
- <https://github.com/gustavoamaral/ProjblocoTP03>