

# ESCOLA SUPERIOR DE TECNOLOGIA DA INFORMAÇÃO



Graduação:	<b>GESTÃO DA TECNOLOGIA DA INFORMAÇÃO</b>
Bloco:	<b>Arquitetura de Infraestrutura e Aplicações</b>
Matéria:	<b>Projeto de Bloco</b>
Referência:	<b>TP8 e TP9</b>
Turma:	<b>GRLGTI01C2-N2-L1</b>
Aluno:	<b>Silney Alvarenga Lopes</b>

## Sumário

1. Introdução.....	3
2. TP8.....	3
3. TP9.....	13
4. Conclusão .....	44

## 1. Introdução

Objetivo neste documento é responder as questões colocadas no TP8 do Projeto de Bloco.

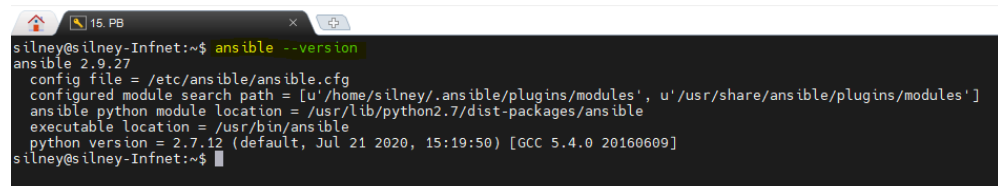
## 2. TP8

Faça upload de uma versão inicial do capítulo 3/Implementação de seu Projeto de Bloco.

Você deve contemplar:

### a. Referências dos downloads e/ou versões de todos os componentes da solução implementada.

Versão do Ansible:




```
silney@silney-Infnet:~$ ansible --version
ansible 2.9.27
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/silney/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.12 (default, Jul 21 2020, 15:19:50) [GCC 5.4.0 20160609]
silney@silney-Infnet:~$
```

Versão dos componentes de servidor:

```
- name: Instalar pacotes do servidor
  apt: name={{ item }} state=present
  become: yes
  with_items:
    - apache2
    - mysql-server
    - php8.1-mysql
    - php8.1
    - libapache2-mod-php8.1
    - python3-pymysql
```

Versão do Docker:

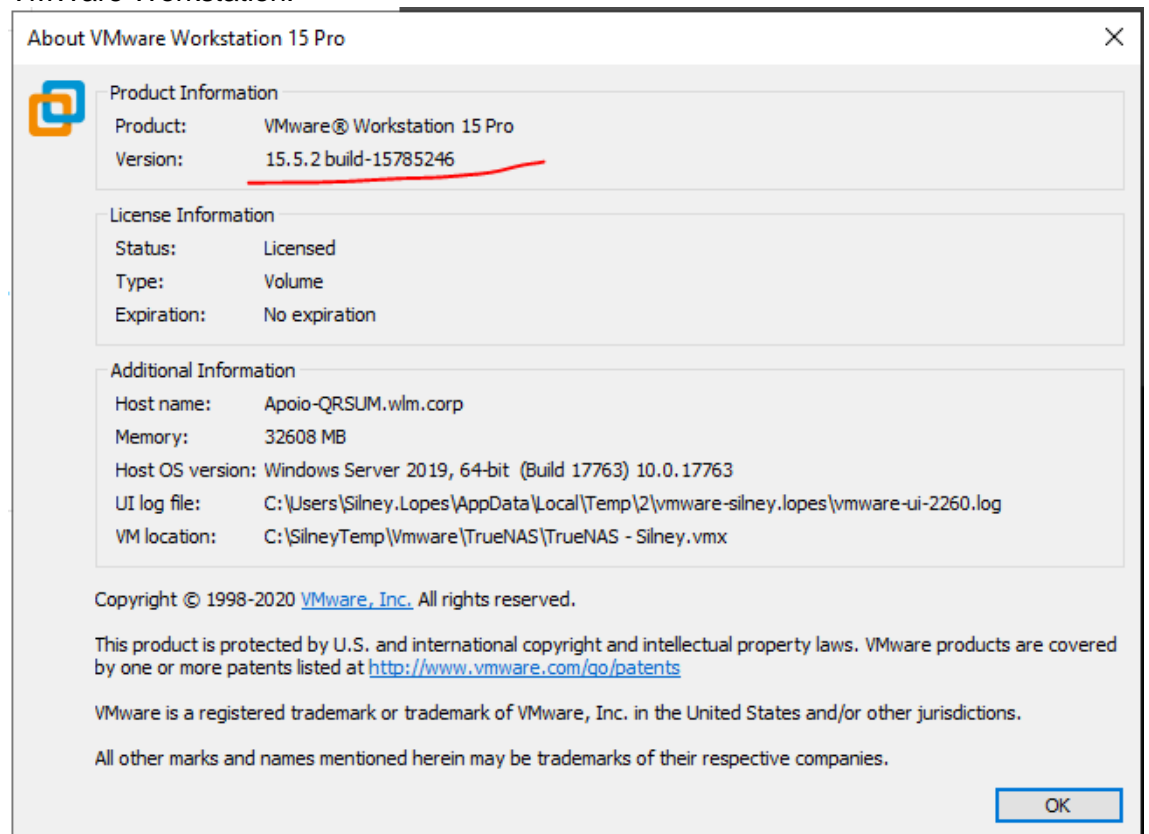


```
silney@silney-infnet:~$ docker --version
Docker version 23.0.1, build a5ee5b1
```

Versão do Ubuntu:

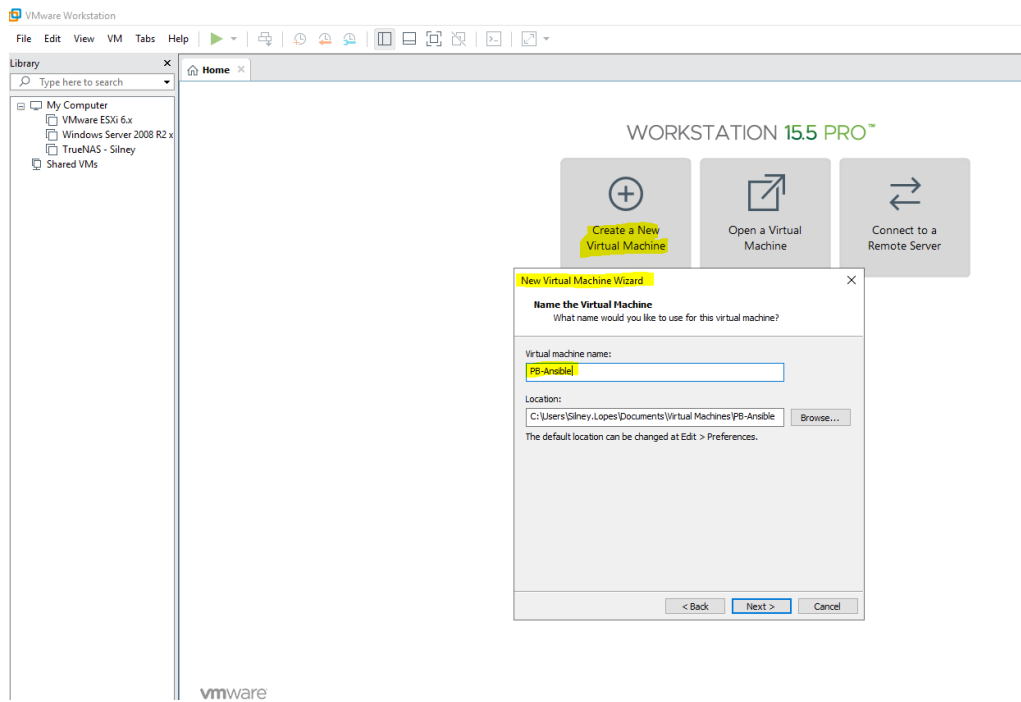
```
silney@silney-infnet:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.2 LTS
Release:        22.04
Codename:       jammy
silney@silney-infnet:~$
```

VMWare Workstation:

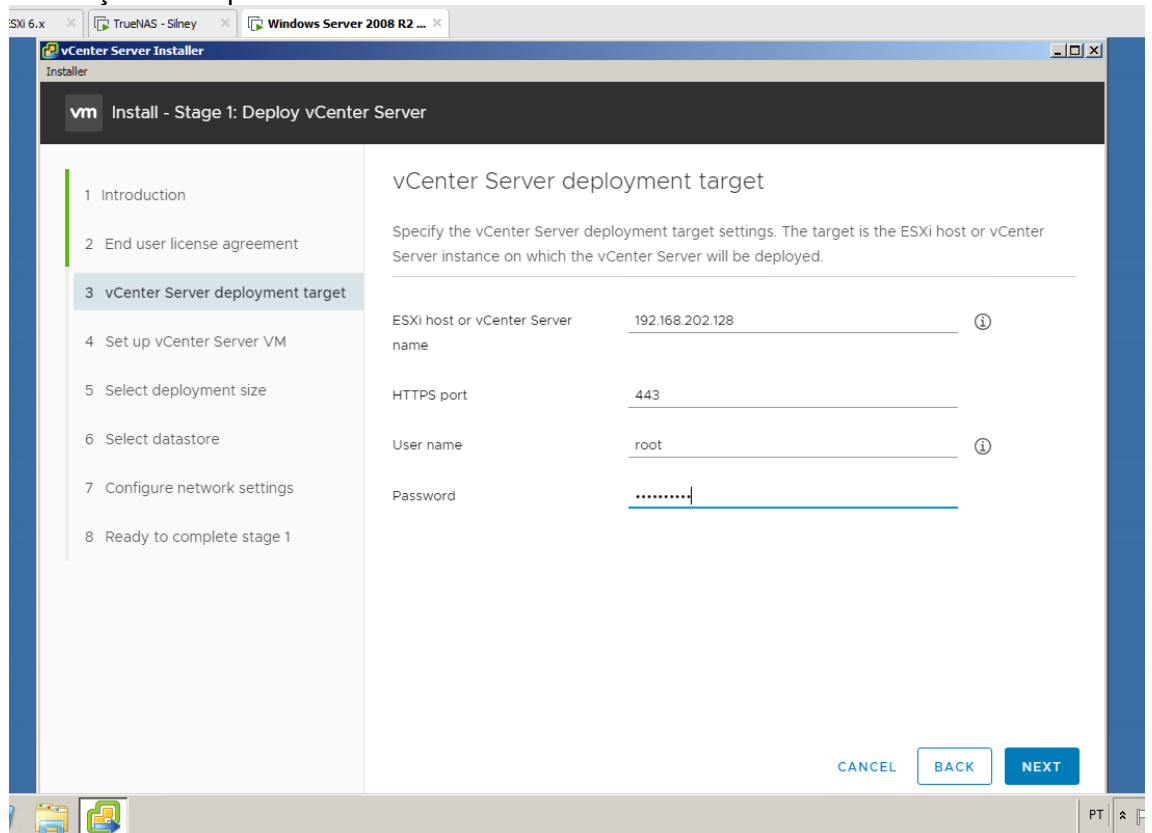


**b. Todos os passos/capturas de tela de implementação de sua infraestrutura de virtualização.**

Criando máquinas virtuais



## Instalação do vSphere



## Instalação do SSH

```
Terminal
silney@silney-infnet: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

silney@silney-Infnet:~$ sudo apt-get install openssh-server
[sudo] senha para silney:
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Pacotes sugeridos:
  ssh-askpass rssh molly-guard monkeysphere
Os NOVOS pacotes a seguir serão instalados:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 pacotes atualizados, 4 pacotes novos instalados, 0 a serem removidos e 188 não
```

## Instalação do Ansible

```
silney@silney-Infnet:~$ sudo apt-add-repository ppa:ansible/ansible
[sudo] senha para silney:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications—au
tomate in a language that approaches plain English, using SSH, with no agents to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
Mais informações: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Pressione [ENTER] para continuar ou ctrl+c para cancelar a adição

gpg: chaveiro '/tmp/tmp1h2295b/secring.gpg' criado
gpg: chaveiro '/tmp/tmp1h2295b/pubring.gpg' criado
gpg: requisitando chave 78B9C367 de servidor hkp - keyserver.ubuntu.com
gpg: /tmp/tmp1h2295b/trustdb.gpg: banco de dados de confiabilidade criado
gpg: chave 78B9C367: chave pública "Launchpad PPA for Ansible, Inc." importada
gpg: Número total processado: 1
gpg:      importados: 1 (RSA: 1)

silney@silney-Infnet:~$ sudo apt-get update
Obter:1 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial InRelease [18.0 kB]
Attingido:2 http://br.archive.ubuntu.com/ubuntu xenial InRelease
Obter:3 http://security.ubuntu.com/ubuntu xenial-security InRelease [99.8 kB]
Obter:4 http://br.archive.ubuntu.com/ubuntu xenial-updates InRelease [99.8 kB]
Obter:5 http://br.archive.ubuntu.com/ubuntu xenial-backports InRelease [97.4 kB]
Obter:6 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 DEP-11 Metadata [327 kB]
Obter:7 http://br.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 DEP-11 Metadata [281 kB]
Obter:8 http://br.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 DEP-11 Metadata [5.956 B]
Obter:9 http://br.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Metadata [3.288 B]
Obter:10 http://br.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-11 Metadata [6.608 B]
Obter:11 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main amd64 Packages [704 B]
Obter:12 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main i386 Packages [704 B]
Obter:13 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main Translation-en [472 B]
Baixados 940 kB em 2s (422 kB/s)
Lendo listas de pacotes... Pronto
silney@silney-Infnet:~$ sudo apt-get install ansible
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  python-cffi-backend python-crypto python-cryptography python-ecdsa python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2 python-markupsafe python-paramiko
  python-pkg-resources python-pyasn1 python-setuptools python-six python-yaml sshpass
Pacotes sugeridos:
  python-crypto-dbg python-crypto-doc python-cryptography-doc python-cryptography-vectors python-enum34-doc python-jinja2-doc python-setuptools-doc
Os NOVOS pacotes a seguir serão instalados:
  ansible python-cffi-backend python-crypto python-cryptography python-ecdsa python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2 python-markupsafe python-paramiko
  python-pkg-resources python-pyasn1 python-setuptools python-six python-yaml sshpass
0 pacotes atualizados, 16 pacotes novos instalados, 0 a serem removidos e 188 não atualizados.
É preciso baixar 7.155 kB de arquivos.
Depois desta operação, 65,5 MB adicionais de espaço em disco serão usados.
Você quer continuar? [S/n] s
Obter:1 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-markupsafe amd64 0.23-2build2 [15,5 kB]
Obter:2 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-jinja2 all 2.8-1ubuntu0.1 [106 kB]
Obter:3 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-yaml amd64 5.11-2build1 [105 kB]
Obter:4 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-crypto amd64 2.6.1-6ubuntu0.16.04.3 [246 kB]
Obter:5 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-six all 1.10.0-3 [10,9 kB]
Obter:6 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-ecdsa all 0.13-2ubuntu0.16.04.1 [36,2 kB]
Obter:7 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-paramiko all 1.16.0-1ubuntu0.2 [110 kB]
Obter:8 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-httplib2 all 0.9.1+dfsg-1 [34,2 kB]
Obter:9 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-pkg-resources all 20.7.0-1 [108 kB]
Obter:10 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-setuptools all 20.7.0-1 [169 kB]
Obter:11 http://br.archive.ubuntu.com/ubuntu xenial/universe amd64 sshpass amd64 1.05-1 [10,5 kB]
```

## Instalação do Docker

```
silney@silney-infnet:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-03-23 23:39:37 -03; 18s ago
     TriggeredBy: ● docker.socket
        Docs: https://docs.docker.com
       Main PID: 3397 (dockerd)
         Tasks: 8
        Memory: 25.7M
         CPU: 895ms
        CGroup: /system.slice/docker.service
                └─3397 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.760841562-03:00" level=info msg="[core] [Channel #4] SubC
mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.761476929-03:00" level=info msg="[core] [Channel #4] Cha
mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.908418910-03:00" level=info msg="Loading containers: sta
mar 23 23:39:37 silney-infnet dockerd[3397]: time="2023-03-23T23:39:37.343150706-03:00" level=info msg="Default bridge /dockers
```

### c. Todos os passos/capturas de tela de configuração de sua aplicação.

#### Construindo Playbook Ansible para Wordpress

```
silney@silney-infnet:~/PBInfnet$ ls
{changed: hosts 'MicrosoftTeams-image (66).png' playbook.yml README.md roles
```

#### Estrutura Playbook Ansible para Wordpress

```
silney@silney-infnet:~$ cd PBInfnet/
silney@silney-infnet:~/PBInfnet$ ls
hosts  playbook.yml  README.md  roles
silney@silney-infnet:~/PBInfnet$ cd roles/
silney@silney-infnet:~/PBInfnet/roles$ ls
mysql  php  server  wordpress
silney@silney-infnet:~/PBInfnet/roles$
```

- PBInfnet/hosts.yml:  
[wordpress]  
192.168.31.35
- PBInfnet/playbook.yml:  
---  
- hosts: wordpress  
 become: yes  
 remote\_user: silney  
  
 roles:  
 - server  
 - php  
 - mysql  
 - wordpress
- PBInfnet/roles/server/tasks/main.yml:  
---  
# tasks file for server  
  
- name: Atualiza apt cache  
 apt: update\_cache=yes cache\_valid\_time=3600  
 become: yes  
  
- name: Instalar pacotes do servidor  
 apt: name={{ item }} state=present  
 become: yes  
 with\_items:

- apache2
- mysql-server
- php8.1-mysql
- php8.1
- libapache2-mod-php8.1
- python3-pymysql

- PBlInfnet/roles/php/tasks/main.yml:

---

# tasks file for php

- name: Instala as extensoes do PHP 7.4
- apt: name={{ item }} state=present
- become: yes
- with\_items:
  - php8.1-gd
  - php-ssh2

- PBlInfnet/roles/mysql/defaults/main.yml:

---

# defaults file for mysql

wp\_mysql\_db: wordpress

wp\_mysql\_user: wordpress

wp\_mysql\_password: wp\_db\_password

- PBlInfnet/roles/mysql/tasks/main.yml:

---

# tasks file for mysql

- name: Cria banco de dados do wordpress

mysql\_db: name={{ wp\_mysql\_db }} state=present

- name: Cria o usuario do banco de dados wordpress

mysql\_user:

name={{ wp\_mysql\_user }}

password={{ wp\_mysql\_password }}

priv=\*.\*:ALL

- PBlInfnet/roles/wordpress/handlers/main.yml:

---

# handlers file for wordpress

- name: restart apache

become: true

service: name=apache2 state=restarted

- PBlInfnet/roles/wordpress/tasks/main.yml:

---

# tasks file for wordpress

- name: Baixar Wordpress

become: true

get\_url:

url=https://wordpress.org/latest.tar.gz

dest=/tmp/wordpress.tar.gz

validate\_certs=no



```

- name: Extrair WordPress
  become: true
  unarchive:
    src=/tmp/wordpress.tar.gz
    dest=/var/www/
    copy=no

- name: Atualizar site Apache
  become: true
  lineinfile:
    dest=/etc/apache2/sites-enabled/000-default.conf
    regexp="(.)+DocumentRoot /var/www/html"
    line="DocumentRoot /var/www/wordpress"
  notify:
    - restart apache

- name: Copy sample config file
  become: true
  command: mv /var/www/wordpress/wp-config-sample.php
/var/www/wordpress/wp-config.php creates=/var/www/wordpress/wp-
config.php

- name: Update WordPress config file
  become: true
  lineinfile:
    dest=/var/www/wordpress/wp-config.php
    regexp="{{ item.regex }}"
    line="{{ item.line }}"
  with_items:
    - {regex: 'database_name_here', line: "define('DB_NAME',
'{{wp_mysql_db}}');"}
    - {regex: 'username_here', line: "define('DB_USER',
'{{wp_mysql_user}}');"}
    - {regex: 'password_here', line: "define('DB_PASSWORD',
'{{wp_mysql_password}}');"}
# - {regex: "define\\('DB_NAME', '(.)+\\';", 'line': "define('DB_NAME',
'{{wp_mysql_db}}');"}
# - {regex: "define\\('DB_USER', '(.)+\\';", 'line': "define('DB_USER',
'{{wp_mysql_user}}');"}
# - {regex: "define\\('DB_PASSWORD', '(.)+\\';", 'line':
"define('DB_PASSWORD', '{{wp_mysql_password}}');"}

```

## Executando o Playbook

```
silney@silney-infnet:~$ cd PBInfnet/
silney@silney-infnet:~/PBInfnet$ ansible-playbook playbook.yml --ask-become-pass
BECOME password:

PLAY [wordpress] *****

TASK [Gathering Facts] *****
ok: [192.168.31.35]

TASK [server : Atualiza apt cache] *****
ok: [192.168.31.35]

TASK [server : Instalar pacotes do servidor] *****
ok: [192.168.31.35] => (item=apache2)
ok: [192.168.31.35] => (item=mysql-server)
ok: [192.168.31.35] => (item=php8.1-mysql)
ok: [192.168.31.35] => (item=php8.1)
ok: [192.168.31.35] => (item=libapache2-mod-php8.1)
ok: [192.168.31.35] => (item=python3-pymysql)

TASK [php : Instala as extensoes do PHP 7.4] *****
ok: [192.168.31.35] => (item=php8.1-gd)
ok: [192.168.31.35] => (item=php8.1-ssh2)

TASK [mysql : Cria banco de dados do wordpress] *****
changed: [192.168.31.35]

TASK [mysql : Cria o usuario do banco de dados wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Baixar WordPress] *****
changed: [192.168.31.35]

TASK [wordpress : Extrair WordPress] *****
changed: [192.168.31.35]

TASK [wordpress : Atualizar site Apache] *****
changed: [192.168.31.35]

TASK [wordpress : Copy sample config file] *****
changed: [192.168.31.35]

TASK [wordpress : Update WordPress config file] *****
changed: [192.168.31.35] => (item={'regex': 'database_name_here', 'line': "define('DB_NAME', 'wordpress');"})
changed: [192.168.31.35] => (item={'regex': 'username_here', 'line': "define('DB_USER', 'wordpress');"})
changed: [192.168.31.35] => (item={'regex': 'password_here', 'line': "define('DB_PASSWORD', 'wp_db_password');"})

RUNNING HANDLER [wordpress : restart apache] *****
changed: [192.168.31.35]

PLAY RECAP *****
192.168.31.35 : ok=12 changed=8 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

silney@silney-infnet:~/PBInfnet$ █
```

## Construindo Playbook Ansible para Docker

```
silney@silney-infnet:~$ cd /etc/ansible/
silney@silney-infnet:/etc/ansible$ nano hosts
```

```
192.168.31.254
[docker]
192.168.31.254
[docker:vars]
ansible_user=silney
ansible_password=maicjc
ansible_become_password=maicjc
ansible_python_interpreter=/usr/bin/python3
█

^G Help      ^O Write Out  ^W Where Is
^X Exit      ^R Read File  ^_ Replace
```

silney@silney-infnet:~/TP4\_TP5\$ nano playbook.yml

```
--
- hosts: docker
  vars:
    db_volume: mariadb
    wordpress: wordpress
  tasks:
    - name: Deploy MariaDB server
      docker_container:
        image: mariadb
        name: mariadb
        volumes:
          - "{{db_volume}}:/var/lib/mysql"
        env:
          MYSQL_ROOT_PASSWORD: password
          MYSQL_PASSWORD: infnet
          MYSQL_DATABASE: db_TP4_TP5
          MYSQL_USER: infnet

    - name: Deploy WordPress
      docker_container:
        image: wordpress
        name: wordpress
        restart_policy: always
        ports:
          - "8080:80"
        links:
          - "{{db_volume}}:/var/lib/mysql"
        volumes:
          - "{{wordpress}}:/var/www/html"
        env:
          MYSQL_PASSWORD: infnet
          MYSQL_DATABASE: db_TP4_TP5
          MYSQL_USER: infnet
          MYSQL_HOST: mariadb
```

## Execução do código

```
silney@silney-infnet:~/TP4_TP5$ export ANSIBLE_HOST_KEY_CHECKING=False
silney@silney-infnet:~/TP4_TP5$ ansible-playbook playbook.yml -l docker --ask-become-pass
BECOME password:

PLAY [docker] *****
TASK [Gathering Facts] *****
ok: [192.168.31.254]

TASK [Deploy MariaDB server] *****
changed: [192.168.31.254]

TASK [Deploy WordPress] *****
changed: [192.168.31.254]

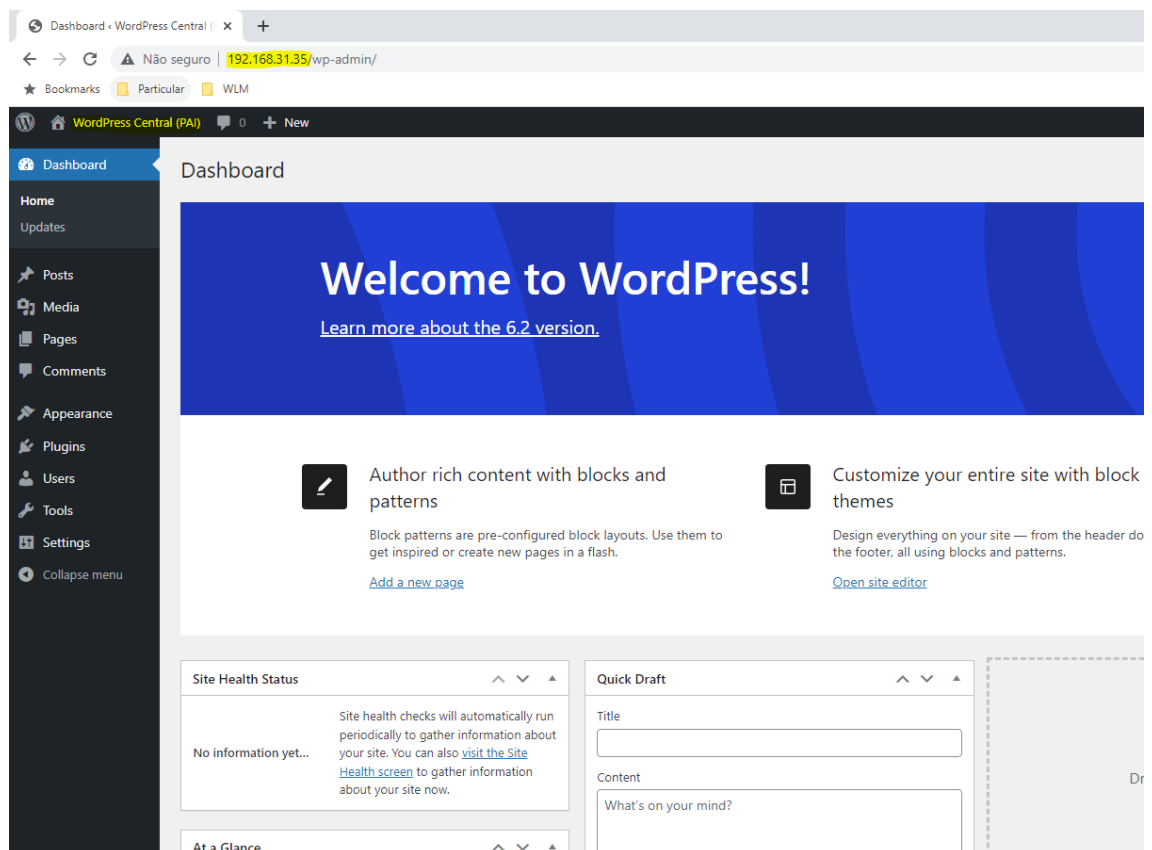
PLAY RECAP *****
192.168.31.254 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

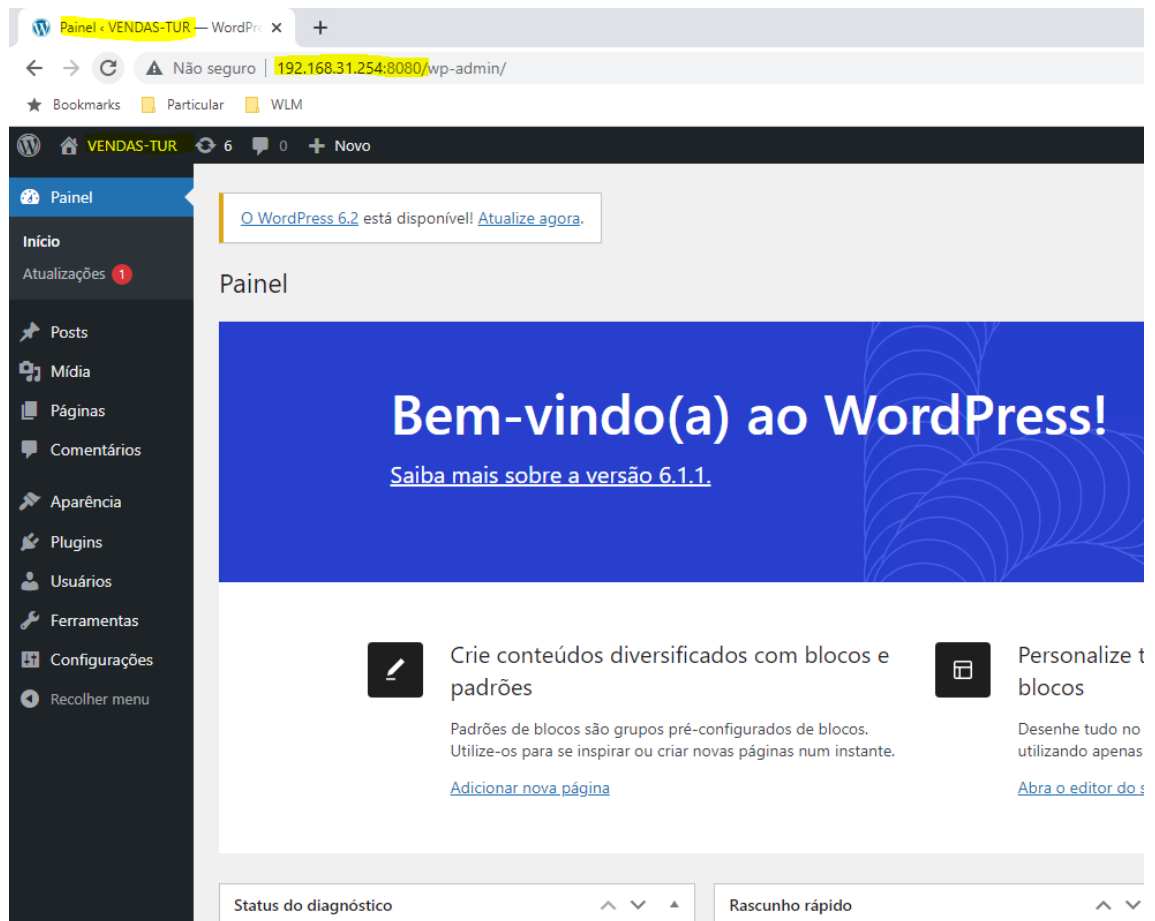
## Consulta da Container Docker

```
For more help on how to use Docker, head to https://docs.docker.com/go/guides/

silney@silney-infnet:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
silney@silney-infnet:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
aebe85500a30   wordpress "docker-entrypoint.s..." 17 seconds ago   Up 14 seconds   0.0.0.0:8080->80/tcp   wordpress
589b33d63eed   mariadb   "docker-entrypoint.s..." 21 seconds ago   Up 18 seconds   3306/tcp             mariadb
silney@silney-infnet:~$
```

### d. Uma captura de tela de sua aplicação em funcionamento.





### 3. TP9

**Faça upload do conteúdo completo de seu Projeto de Bloco até aqui, incluindo todos os capítulos apresentados nos TPs 6, 7 e 8, mais uma versão inicial do capítulo 4/Conclusões.**

**Além dos requisitos apresentados para os capítulos anteriores, suas conclusões devem incluir:**

- a. **Se o prazo estabelecido no início do projeto foi adequado para a execução (não há problema se levou mais tempo que o planejado, mas se levou, explique as razões).**

O prazo me pareceu bem ajustado, dá para executar as tarefas confortavelmente, com exceção da programação em Ansible, que demorei para conseguir pegar, acredito que por falta de afinidade.

- b. **Se os recursos planejados (quantidade de memória, disco, etc.) foram suficientes para colocar a solução em funcionamento. Se não foram, explique as razões e sugira uma nova configuração.**

Meu equipamento é um notebook corporativo com 8GB de memória, era impraticável subir as máquinas virtuais na minha estação, consegui ligar um servidor na empresa em que trabalho e utilizei para a realização do

bloco e disciplinas. Sugiro que nos próximos blocos as lições já sejam diretamente em nuvem, dispensando recurso de máquina e em sinergia com as tecnologias do mercado.

- c. **Se as funcionalidades previstas na solução original funcionaram a contento. Se não foram, explique as razões.**

Sim, consegui executar exatamente o que tinha planejado inicialmente.

- d. **Quais seriam as melhorias futuras que poderiam ser feitas no projeto executado, adotando soluções adicionais, aperfeiçoando instalações ou tornando o cenário mais complexo.**

Esse projeto foi feito utilizando uma nuvem privada, seria muito interessante, inclusive para o Sr. João (o dono da empresa de turismo), que fosse migrado para uma nuvem pública, teríamos imensa redução de custos e os serviços ficariam melhores, visto que estariam em um ambiente altamente seguro e disponível.

- e. **Antes da entrega final, todo o conteúdo de seu trabalho (mesmo que sejam apenas documentos do Word e PDFs) deve ser armazenado em um repositório GitHub, cujo link deve estar contido no corpo do trabalho.**

<https://github.com/silneylopes/Projeto-de-Bloco/>

## **Uploads conteúdo completo projeto de bloco:**

### **➤ TP1**

#### **Aplicação**

A aplicação escolhida para ser implantada, é um ERP. A organização está aderindo ao movimento de ir para a nuvem, mas por hora ainda está com sua nuvem privada, junto a isso, a empresa quer deixar de ficar presa a softwares de prateleira e a diretoria solicitou que a Equipe de programação iniciasse a construção de um ERP para atender a todo o negócio.

#### **Desenvolvimento**

Definir claramente os requisitos do software, a fim de garantir que o produto final atenda às expectativas do cliente, fazer um bom design, para que o sistema seja fácil de usar, eficiente e escalável, levando em consideração arquitetura do sistema e tecnologias utilizadas, codificar o sistema (transformar o design em códigos de programação) de forma organizada e documentada, testes em todas as etapas do processo de desenvolvimento e manutenção para garantir que ela continue funcionando atendendo as necessidades do negócio.

#### **Infraestrutura**

A infraestrutura do ERP vai rodar inicialmente em uma nuvem privada da vmware esxi mas mirando em um Datacenter como software SDDC, no Microsoft Azure.

Requisitos:

- 3 Virtual Machines, Linux, Distribuição Ubuntu 22.04, 2vCPUs 2.0Ghz, 8GB RAM, 100GB SSD
- Estar na mesma sub-rede, se comunicarem.

### **Implantação**

Três máquinas virtuais sendo uma para nossa estação Ansible, outra para nossa estação WordPress Central e a última uma distribuição Docker com containers WordPress menores para cada setor/departamento. Programadores que farão a intercomunicação dos departamentos para a construção central.

### **Configuração**

Na Configuração devemos tratar das tabelas que serão utilizadas pelo banco de dados, disponibilização da aplicação na web, configurações de segurança como SSL/TLS por exemplo, configurações de integração como API por exemplo, configurações dos processos de negócio e personalizações da experiência do usuário.

## **Detalhes**

### **Tratamento de Demandas**

O Negócio possui um sistema de chamados, onde recebe solicitações de demandas de todos os tipos, foi cadastrado um novo menu de serviços sobre o novo ERP que esta sendo desenvolvido e lá temos funções de novas ideias e correções, assim permitindo que todos possam contribuir. Ao entrar uma nova solicitação desse teor, ela é enviada a um grupo de pessoas que são os stakeholders do ERP, esses vão se reunir e analisar as demandas, verificar a viabilidade das ideias e correções, se forem aprovadas, entrarão na fila dos desenvolvedores.

### **Gerenciamento de código-fonte**

Os códigos fonte serão gerenciados e documentados pela plataforma GitHub, que é uma plataforma de hospedagem de códigos fonte e gerenciamento de projetos que permite a colaboração de desenvolvedores e controle de versões.

### **Ciclo de Vida**

O ciclo de vida do desenvolvimento de software será gerenciado pela plataforma Azure DevOps, com ela acompanharemos o gerenciamento de projetos e tarefas, o controle de versões, as compilações e implantações, os testes e validações e gerenciamento de artefatos.

### **Compilação e Unidade Instalável**

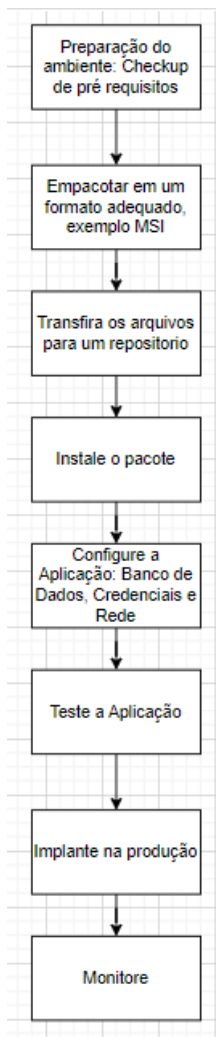
Preparação do ambiente, como instalação de bibliotecas e compiladores, verificação de código-fonte com depuradores, compilação de arquivos-fonte em arquivos executáveis, testes e implantação. É importante que utilizemos a integração contínua e a entrega contínua.

### **Gerenciamento de Configurações**

A ferramenta que permite automatizar o provisionamento, a configuração e o gerenciamento de sistemas e aplicativos que vamos utilizar é o ANSIBLE. Ele usa uma linguagem de configuração simples e tem suporte para vários sistemas operacionais.

### **Plano de Implantação**





## ➤ TP2 e TP3

Continuando nosso projeto, a instalação de um Wordpress fará parte da implementação de um ERP, portanto nosso projeto será instalar o Ansible e subir o Wordpress Central com Ansible e hospedar seus códigos no repositório público do GitHub.

**Instalamos o SSH nos dois servidores (PB-Ansible e PB-WordPress)**

```
Terminal
silney@silney-Infnet: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

silney@silney-Infnet:~$ sudo apt-get install openssh-server
[sudo] senha para silney:
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Pacotes sugeridos:
  ssh-askpass rssh molly-guard monkeysphere
Os NOVOS pacotes a seguir serão instalados:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 pacotes atualizados, 4 pacotes novos instalados, 0 a serem removidos e 188 não
```

## Instalando o ANSIBLE e conferindo a versão

```

silney@silney-Infinet:~$ sudo apt-add-repository ppa:ansible/ansible
[sudo] senha para silney:
Ansible é um radicalmente simples IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications-
au
tomate in a language that approaches plain English, using SSH, with no agents to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
Mais informações: https://launchpad.net/ansible/+archive/ubuntu/ansible
Pressione [ENTER] para continuar ou ctrl-c para cancelar a adição

gpg: chaveiro '/tmp/tmp1h2295b/securig.gpg' criado
gpg: chaveiro '/tmp/tmp1h2295b/pubring.gpg' criado
gpg: requisitando chave 78B9C367 de servidor hkp - keyserver.ubuntu.com
gpg: /tmp/tmp1h2295b/trustdb.gpg: banco de dados de confiabilidade criado
gpg: chave 78B9C367: chave pública "Launchpad PPA for Ansible, Inc." importada
gpg: Número total processado: 1
gpg:      importados: 1 (RSA: 1)

silney@silney-Infinet:~$ sudo apt-get update
Oter01: http://ppa.launchpad.net/ansible/ansible/ubuntu xenial InRelease [18.0 kB]
Ater02: http://br.archive.ubuntu.com/ubuntu xenial InRelease
Oter03: http://security.ubuntu.com/ubuntu xenial-security InRelease [99.8 kB]
Oter04: http://br.archive.ubuntu.com/ubuntu xenial-updates InRelease [99.8 kB]
Oter05: http://br.archive.ubuntu.com/ubuntu xenial-backports InRelease [97.4 kB]
Oter06: http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 DEP-11 Metadata [327 kB]
Oter07: http://br.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 DEP-11 Metadata [281 kB]
Oter08: http://br.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 DEP-11 Metadata [5.956 B]
Oter09: http://br.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Metadata [3.328 B]
Oter10: http://br.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-11 Metadata [6.608 B]
Oter11: http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main amd64 Packages [704 B]
Oter12: http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main i386 Packages [204 B]
Oter13: http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main Translation-en [472 B]
Baixados 940 kB em 2s (422 kB/s)
Lendo listas de pacotes... Pronto
silney@silney-Infinet:~$ sudo apt-get install ansible
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informações de estado... Pronto
The following additional packages will be installed:
python-cffi-backend python-crypto python-cryptography python-ecdsa python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2 python-markupsafe python-paramiko
python-pkg-resources python-pyasn1 python-setuptools python-six python-yaml sshpass
Pacotes sugeridos:
python-crypto-dbg python-crypto-doc python-cryptography-doc python-cryptography-vectors python-enum34-doc python-jinja2-doc python-setuptools-doc
Os novos pacotes a serem instalados são:
ansible python-cffi-backend python-crypto python-cryptography python-ecdsa python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2 python-markupsafe python-paramiko
python-pkg-resources python-pyasn1 python-setuptools python-six python-yaml sshpass
O pacote atualizado, 16 pacotes novos instalados, 0 a serem removidos e 188 não atualizados.
É preciso baixar 7.155 kB de arquivos.
Depois desta operação, 65,5 MB adicionais de espaço em disco serão usados.
Você quer continuar? [S/n] s
Oter01: http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-markupsafe amd64 0.23-2build2 [15.5 kB]
Oter02: http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-jinja2 all 2.8-1ubuntu0.1 [106 kB]
Oter03: http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-yaml amd64 3.11-3build1 [105 kB]
Oter04: http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-crypto amd64 2.6.1-5ubuntu0.16.04.3 [246 kB]
Oter05: http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-six all 1.10.0-3 [10.9 kB]
Oter06: http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-ecdsa all 0.13-2ubuntu0.16.04.1 [36.2 kB]
Oter07: http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-paramiko all 1.16.0-1ubuntu0.1 [110 kB]
Oter08: http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-httplib2 all 0.9.1-1ubuntu1 [34.2 kB]
Oter09: http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-pkg-resources all 20.7.0-1 [108 kB]
Oter10: http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-setuptools all 20.7.0-1 [169 kB]

```

```
silney@silney-Infnct:~$ ansible --version
ansible 2.9.27
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/silney/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.12 (default, Jul 21 2020, 15:19:50) [GCC 5.4.0 20160609]
silney@silney-Infnct:~$
```

## Construindo playbook

O playbook foi construído e enviado para o diretório chamado **PBInfnet** via GitHub (será demonstrado)

```
silney@silney-infnets:~/PBInfnets$ ls
{changed: hosts 'MicrosoftTeams-image (66).png'  playbook.yml  README.md  roles
```

## Executando o playbook

```

silney@silney-infnet:~$ cd PBInfnet/
silney@silney-infnet:~/PBInfnet$ ansible-playbook playbook.yml --ask-become-pass
BECOME password:

PLAY [wordpress] *****

TASK [Gathering Facts] *****
ok: [192.168.31.35]

TASK [server : Atualiza apt cache] *****
ok: [192.168.31.35]

TASK [server : Instalar pacotes do servidor] *****
ok: [192.168.31.35] => (item=apache2)
ok: [192.168.31.35] => (item=mysql-server)
ok: [192.168.31.35] => (item=php8.1-mysql)
ok: [192.168.31.35] => (item=php8.1)
ok: [192.168.31.35] => (item=libapache2-mod-php8.1)
ok: [192.168.31.35] => (item=python3-pymysql)

TASK [php : Instala as extensoes do PHP 7.4] *****
ok: [192.168.31.35] => (item=php8.1-gd)
ok: [192.168.31.35] => (item=php-ssh2)

TASK [mysql : Cria banco de dados do wordpress] *****
changed: [192.168.31.35]

TASK [mysql : Cria o usuario do banco de dados wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Baixar Wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Extrair Wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Atualizar site Apache] *****
changed: [192.168.31.35]

TASK [wordpress : Copy sample config file] *****
changed: [192.168.31.35]

TASK [wordpress : Update WordPress config file] *****
changed: [192.168.31.35] => (item={'regex': 'database_name_here', 'line': "define('DB_NAME', 'wordpress');"})
changed: [192.168.31.35] => (item={'regex': 'username_here', 'line': "define('DB_USER', 'wordpress');"})
changed: [192.168.31.35] => (item={'regex': 'password_here', 'line': "define('DB_PASSWORD', 'wp_db_password');"})

RUNNING HANDLER [wordpress : restart apache] *****
changed: [192.168.31.35]

PLAY RECAP *****
192.168.31.35      : ok=12  changed=8  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0


silney@silney-infnet:~/PBInfnet$ █

```

## Testando se o Wordpress está no ar

Não seguro | 192.168.31.35/wp-admin/install.php

Particular WLM



### Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

### Information needed

Please provide the following information. Do not worry, you can always change these settings later.

**Site Title**  
WordPress Central (PAI)

**Username**  
silney  
Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

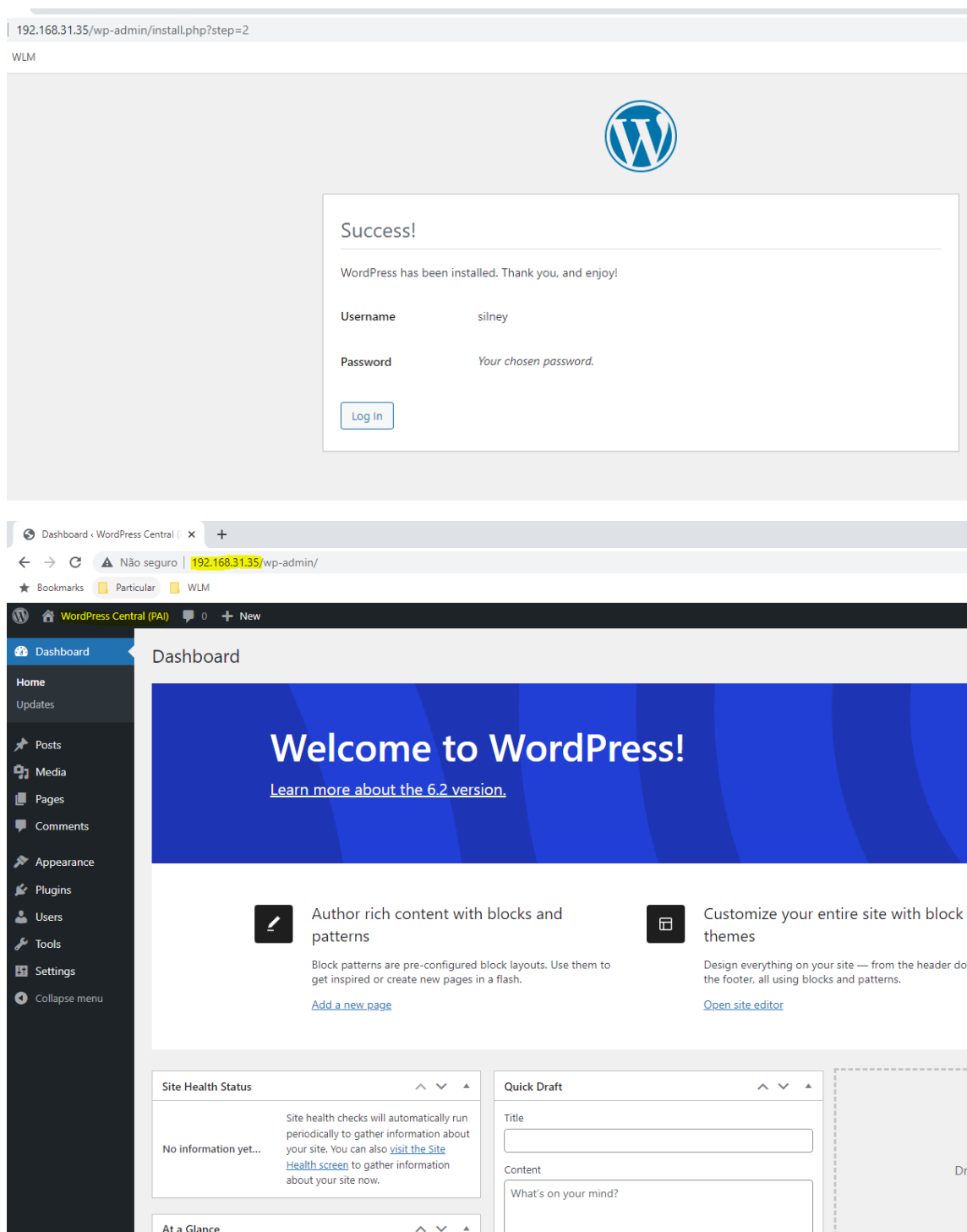
**Password**  
\*\*\*\*\*  
Very weak  
**Important:** You will need this password to log in. Please store it in a secure location.

**Confirm Password**  
☒ Confirm use of weak password

**Your Email**  
silneylopes@hotmail.com  
Double-check your email address before continuing.

**Search engine visibility**  
☒ Discourage search engines from indexing this site  
It is up to search engines to honor this request.

[Install WordPress](#)



## GitHub



Agora iremos criar um repositório no GitHub com o nome PBInfnct:

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

---


Owner \*      Repository name \*


 silneylopes / PBInfnet 

Great repository names are short and memorable. Need inspiration? How about **potential-octo-robot**?

Description (optional)

---

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

---

**Initialize this repository with:**  
Skip this step if you're importing an existing repository.

☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more.](#)


**Add .gitignore**  
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

**Choose a license**  
A license tells others what they can and can't do with your code. [Learn more.](#)

License: None ▾

---

 You are creating a public repository in your personal account.

---

[Create repository](#)

Os códigos foram construídos no github e feito o PULL para o servidor

```
silney@silney-infnet:~/PBInfnet$ git push
Enumerating objects: 45, done.
Counting objects: 100% (40/40), done.
Delta compression using up to 2 threads
Compressing objects: 100% (19/19), done.
Writing objects: 100% (23/23), 2.23 KiB | 380.00 KiB/s, done.
Total 23 (delta 8), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (8/8), completed with 5 local objects.
To https://github.com/silneylopes/PBInfnet.git
   f347fed..b4187e8  main -> main
silney@silney-infnet:~/PBInfnet$ git pull
Already up to date.
silney@silney-infnet:~/PBInfnet$
```

Para conferir os códigos, segue o link do repositório:

<https://github.com/silneylopes/PBInfnet.git>

## ➤ TP4 e TP5

### Preparando ambiente

A empresa observou, que é uma estratégia interessante construir mini ERPs com WordPress, segregado por departamentos e a equipe de programação cuidar da intercomunicação, backup e funcionamento entre eles (como módulos), devido ao baixo custo com a utilização de docker e recurso de máquina.

Baseado nisso, foi preciso desenvolver um código Ansible que automatize a instalação de um banco de dados e um wordpress para futuras implantações de mini ERPs de homologação e produção.

Começamos aqui o desenvolvimento:

### Instalando o Docker

Foi criada uma VM dedicada ao Docker, os comandos utilizados para a instalação foram:

- `sudo apt update`
- `sudo apt install apt-transport-https ca-certificates curl software-properties-common`
- `curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg`
- `echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null`
- `sudo apt update`
- `sudo apt install docker-ce`

### Evidências da Instalação:

```
silney@silney-infnet:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-03-23 23:39:37 -03; 18s ago
     TriggeredBy: ● docker.socket
        Docs: https://docs.docker.com
      Main PID: 3397 (dockerd)
         Tasks: 8
        Memory: 25.7M
           CPU: 895ms
      CGroup: /system.slice/docker.service
              └─3397 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.760841562-03:00" level=info msg="[core] [Channel #4 SubC
mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.761476929-03:00" level=info msg="[core] [Channel #4] Cha
mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.908418910-03:00" level=info msg="Loading containers: sta
mar 23 23:39:37 silney-infnet dockerd[3397]: time="2023-03-23T23:39:37.343459706-03:00" level=info msg="Default bridge (docke

silney@silney-infnet:~$ docker --version
Docker version 23.0.1, build a5ee5b1
```

## Configurações VM Ansible

Criei uma pasta chamada TP4\_TP5 vazia no GitHub e fiz o git clone para baixa-la na VM de Ansible.

```
silney@silney-infnet:~$ ls
Desktop Documents Downloads Music PBInfnet Pictures Public snap Templates Videos
silney@silney-infnet:~$ git clone https://github.com/silneylopes/TP4_TP5
Cloning into 'TP4_TP5'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
silney@silney-infnet:~$ ls
Desktop Documents Downloads Music PBInfnet Pictures Public snap Templates TP4_TP5 Videos
silney@silney-infnet:~$
```

## Instalei o modulo Python Docker

```
sudo apt install python3-pip
pip3 install docker
```

```
silney@silney-infnet:~/TP4_TP5$ pip3 install docker
Defaulting to user installation because normal site-packages is not writeable
Collecting docker
  Downloading docker-6.0.1-py3-none-any.whl (147 kB)
    147.5/147.5 KB 3.9 MB/s eta 0:00:00
Requirement already satisfied: packaging>=14.0 in /usr/lib/python3/dist-packages (from docker) (21.3)
Collecting websocket-client>=0.32.0
  Downloading websocket_client-1.5.1-py3-none-any.whl (55 kB)
    55.9/55.9 KB 5.6 MB/s eta 0:00:00
Requirement already satisfied: urllib3>=1.26.0 in /usr/lib/python3/dist-packages (from docker) (1.26.5)
Collecting requests>=2.26.0
  Downloading requests-2.28.2-py3-none-any.whl (62 kB)
    62.8/62.8 KB 7.2 MB/s eta 0:00:00
Collecting charset-normalizer<4,>=2
  Downloading charset_normalizer-3.1.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (199
    199.3/199.3 KB 8.0 MB/s eta 0:00:00
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3/dist-packages (from requests>=2.26.0->
Requirement already satisfied: certifi>=2017.4.17 in /usr/lib/python3/dist-packages (from requests>=2.
Installing collected packages: websocket-client, charset-normalizer, requests, docker
WARNING: The script wsdump is installed in '/home/silney/.local/bin' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-scr
WARNING: The script normalizer is installed in '/home/silney/.local/bin' which is not on PATH.
```

Altereí o arquivo hosts da pasta Ansible inserindo o grupo docker com o endereço do servidor virtual que irá receber o Docker e suas variáveis.

```
silney@silney-infnet:~$ cd /etc/ansible/
silney@silney-infnet:/etc/ansible$ nano hosts
```

```
[docker]
192.168.31.254
[docker:vars]
ansible_user=silney
ansible_password=maicjc
ansible_become_password=maicjc
ansible_python_interpreter=/usr/bin/python3
^G Help      ^O Write Out  ^W Where Is
^X Exit      ^R Read File  ^_ Replace
```

## Construção do Playbook

```
silney@silney-infnet:~/TP4_TP5$ nano playbook.yml
```

```

--
- hosts: docker
  vars:
    db_volume: mariadb
    wordpress: wordpress
  tasks:
    - name: Deploy MariaDB server
      docker_container:
        image: mariadb
        name: mariadb
        volumes:
          - "{{db_volume}}:/var/lib/mysql"
        env:
          MYSQL_ROOT_PASSWORD: password
          MYSQL_PASSWORD: infnet
          MYSQL_DATABASE: db_TP4_TP5
          MYSQL_USER: infnet

    - name: Deploy WordPress
      docker_container:
        image: wordpress
        name: wordpress
        restart_policy: always
        ports:
          - "8080:80"
        links:
          - "{{db_volume}}:/var/lib/mysql"
        volumes:
          - "{{wordpress}}:/var/www/html"
        env:
          MYSQL_PASSWORD: infnet
          MYSQL_DATABASE: db_TP4_TP5
          MYSQL_USER: infnet
          MYSQL_HOST: mariadb

```

Execução do código na VM Ansible para instalar na VM Docker  
(192.168.31.254)

```

silney@silney-infnet:~/TP4_TP5$ export ANSIBLE_HOST_KEY_CHECKING=False
silney@silney-infnet:~/TP4_TP5$ ansible-playbook playbook.yml -l docker --ask-become-pass
BECOME password:

PLAY [docker] *****

TASK [Gathering Facts] *****
ok: [192.168.31.254]

TASK [Deploy MariaDB server] *****
changed: [192.168.31.254]

TASK [Deploy WordPress] *****
changed: [192.168.31.254]

PLAY RECAP *****
192.168.31.254      : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```



## Resultado na VM Docker


```
For more help on how to use Docker, head to https://docs.docker.com/go/guides/

silney@silney-infnet:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
silney@silney-infnet:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
aebe85500a30   wordpress "docker-entrypoint.s..." 17 seconds ago   Up 14 seconds   0.0.0.0:8080->80/tcp   wordpress
589b33d63eed   mariadb   "docker-entrypoint.s..." 21 seconds ago   Up 18 seconds   3306/tcp             mariadb
silney@silney-infnet:~$
```

## Teste via browser


seguro 192.168.31.254:8080/wp-admin/setup-config.php?step=1&language=pt\_BR

lar WLM



Abaixo você deve digitar os detalhes de conexão do banco de dados. Se você não tem certeza sobre isso, entre em contato com sua hospedagem.


Nome do banco de dados	<input type="text" value="db_TP4_TP5"/>	O nome do seu banco de dados que você deseja utilizar com o WordPress.
Nome de usuário	<input type="text" value="infnet"/>	Usuário do seu banco de dados.
Senha	<input type="text" value="infnet"/>	Senha do seu banco de dados.
Servidor do banco de dados	<input type="text" value="mariadb"/>	Você deve ser capaz de obter esta informação no seu servidor de hospedagem, caso localhost não funcione.
Prefixo da tabela	<input type="text" value="wp_infnet_pb"/>	Se quiser rodar várias instalações WordPress em um único banco de dados, mude isto.



Muito bem! Você concluiu esta parte da instalação. Agora o WordPress pode se comunicar com seu banco de dados. Se você estiver pronto, é hora de...

Não seguro | 192.168.31.254:8080/wp-admin/install.php?step=1

Particular WLM



## Bem-vindo (a)

Bem-vindo (a) à famosa instalação do WordPress em cinco minutos! Basta preencher as informações abaixo e você estará a poucos passos de usar a plataforma de publicação mais extensível e poderosa do mundo.

### Informação necessária

Forneça as seguintes informações. Não se preocupe, você pode alterar estas configurações mais tarde.

**Título do site**

**Nome de usuário**   
Nomes de usuário podem ter somente caracteres alfanuméricos, espaços, sublinhados, hífens, pontos e o símbolo @.

**Senha**  [Show](#)  
Very weak

**Confirme a senha** ☒ Confirmar o uso de uma senha fraca

**O seu e-mail**   
Verifique o seu endereço de e-mail antes de prosseguir.


**Visibilidade nos motores de busca** ☒ Evitar que mecanismos de busca indexem este site  
Cabe aos mecanismos de busca atender esta solicitação.


[Instalar WordPress](#)

Painel - VENDAS-TUR - WordPress

Não seguro | 192.168.31.254:8080/wp-admin/

Bookmarks Particular WLM

**VENDAS-TUR** 6 0 + Novo

**Painel**

Início

Atualizações 1

Posts

Mídia

Páginas

Comentários

Aparência

Plugins

Usuários

Ferramentas

Configurações


Recolher menu

[WordPress 6.2](#) está disponível! [Atualize agora.](#)

## Painel


# Bem-vindo(a) ao WordPress!

[Saiba mais sobre a versão 6.1.1.](#)

**Crie conteúdos diversificados com blocos e padrões**

Padrões de blocos são grupos pré-configurados de blocos. Utilize-os para se inspirar ou criar novas páginas num instante.

[Adicionar nova página](#)

**Personalize t blocos**

Desenhe tudo no utilizando apenas

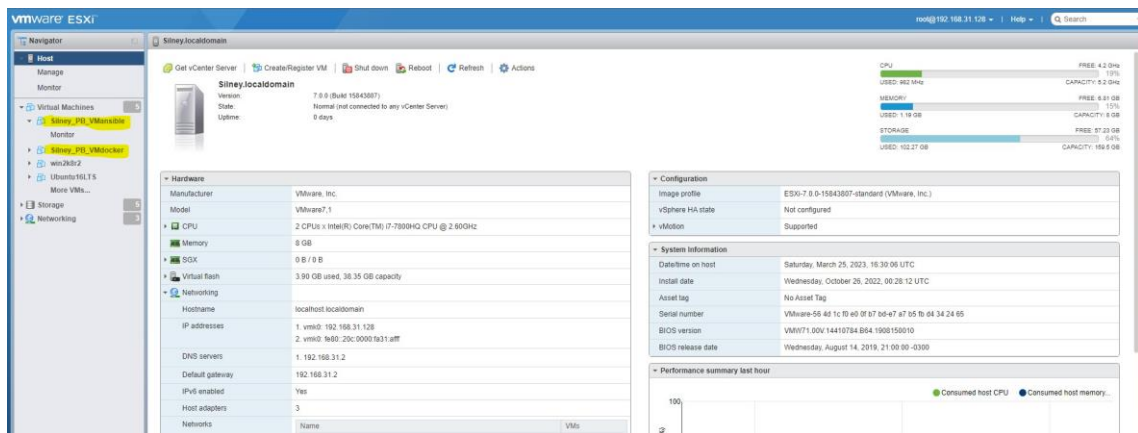
[Abra o editor do s](#)

Status do diagnóstico ^ v ^

Rascunho rápido ^ v

pág. 26

Tudo Executado em uma Nuvem Privada da VMware ESXi



Para finalizar estes TPs, executamos um git push para enviar os códigos para o repositório TP4\_TP5 do GitHub

```
silney@silney-infnet:~/TP4_TP5$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 676 bytes | 676.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/silneylopes/TP4_TP5.git
2dec170..9b3ab17 main -> main
silney@silney-infnet:~/TP4_TP5$
```

Segue link: [https://github.com/silneylopes/TP4\\_TP5](https://github.com/silneylopes/TP4_TP5)

## ➤ TP6 e TP7

### TP6

Faça upload de uma versão inicial do capítulo de Introdução de seu Projeto de Bloco. Você deve contemplar:

- O tipo de negócio/processo/problema que pretende tratar a partir de uma aplicação distribuída rodando sobre uma infraestrutura com virtualização.**

Em nosso caso hipotético, o tipo de negócio é de uma empresa de turismo, a empresa está no mercado a bastante tempo, sua marca esta

em alguns estados no Brasil, mas ainda com processos manuais, cadernos, digitalização, e-mail e uma equipe de Marketing para lançar promoções em sites. Porém o proprietário, que aqui vamos tratar como Sr. João, é bem controlado financeiramente e conseguiu fazer um bom caixa na empresa e começou a entender a importância da tecnologia no mundo atual e sente que se nada for feito poderá ver seus concorrentes engolirem o que construiu. Seus funcionários sempre trazem ideias de como melhorar cada setor da empresa, mas todas elas sempre envolveram tecnologia, que o Sr. João sempre disse ser “caro demais” e sempre teve medo. O plano agora que o Sr. João se convenceu que precisa da tecnologia, é informatizar a empresa e melhorar os sistemas, caso tenha algum, principalmente entender as demandas dos departamentos e construir soluções que atendam totalmente ao negócio. Segue um resumo dos departamentos da empresa:

Departamento de vendas: Responsável por promover os pacotes de viagem e serviços turísticos da empresa e fechar negócios com os clientes.

Departamento de marketing: Responsável pela criação de campanhas publicitárias e promoção da empresa e seus produtos.

Departamento de operações: Responsável por coordenar e executar todas as operações relacionadas aos pacotes de viagem, como transporte, hospedagem, alimentação, passeios etc.

Departamento de reservas: Responsável por receber e gerenciar as reservas dos clientes.

Departamento financeiro: Responsável pela gestão financeira da empresa, incluindo faturamento, contas a pagar e a receber, folha de pagamento, entre outros.

Departamento de recursos humanos: Responsável pela gestão dos colaboradores da empresa, incluindo recrutamento, seleção, treinamento, desenvolvimento, entre outros.

Departamento de atendimento ao cliente: Responsável por prestar suporte e atendimento aos clientes antes, durante e após a viagem, solucionando eventuais problemas e dúvidas.

Departamento de planejamento: Responsável por planejar e desenvolver novos produtos e serviços turísticos, além de definir metas e estratégias de crescimento da empresa.

**b. Uma justificativa de porque este problema é relevante.**

Esse problema é altamente relevante, pois os concorrentes estão cada vez mais ágeis e competitivos, com estratégias melhores, KPI, metas, análises de dados, conseguindo encontrar os clientes utilizando bem a tecnologia, se o Sr. João ficar para trás a empresa irá a falência. Além disso, os funcionários devem estar se sentindo mal por trabalharem de forma arcaica sabendo de novas tecnologias no mercado e não sendo atendidos nos seus pedidos ao Sr. João. É uma situação que precisa ser resolvida de forma rápida, para atender a todos os funcionários, departamentos e ainda afetar diretamente o faturamento da empresa.

Como foi pensado?

Com programadores e Wordpress, o wordpress pode ser utilizado como ERP e mini ERP, então a ideia é subir um Wordpress que vamos chamar de Central, esse será o Wordpress PAI que irá receber as regras de negócio e códigos validados dos wordpress filhos que serão instalados nos departamentos mapeados, desta forma, cada departamento poderá “montar” a melhor forma do seu departamento trabalhar e depois integramos tudo no sistema PAI.

- c. Uma descrição da aplicação distribuída que você pretende implementar com detalhes de sua arquitetura, como o gerenciamento do código-fonte, do processo de desenvolvimento, pré-requisitos para instalação, servidores necessários etc.

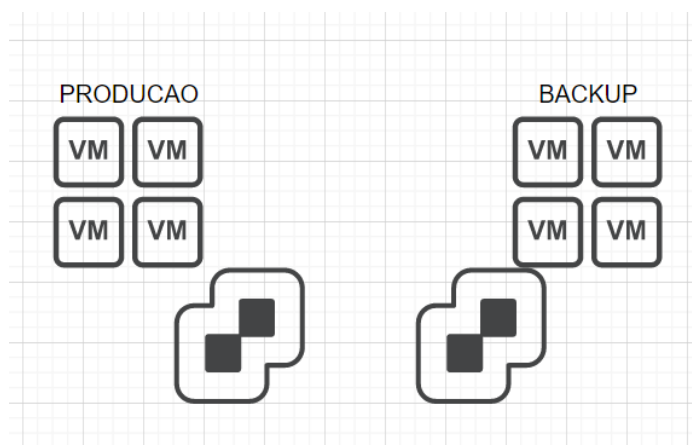
Os código-fonte serão todos gerenciados pelos programadores utilizando a plataforma GitHub, não só para o código, mas também para a colaboração em um mesmo projeto, para gerenciamento de projetos e integração contínua. Os pré-requisitos para nossa aplicação funcionar são:

Para os usuários:

- Ter comunicação de rede
- Ter um navegador de internet
- Ter permissões de acesso

Para a TI:

- Ter dois hardwares com no mínimo:
  - Disco: 1TB
  - Espaço Livre no Disco: 256GB
  - Memoria: 32GB
  - Processador: Xeon E2560 2.60Ghz
- Ter dois sistemas operacionais baseados em Linux
- Ter permissões de administrador
- Ter estrutura de comunicação de Rede (interna e externa)



Quanto ao processo de desenvolvimento da aplicação, estamos na fase de levantamento de requisitos e análise de design, onde é necessário entender quais são as necessidades do cliente e quais os objetivos da aplicação e como a aplicação deve ser estruturada e projetada bem como ferramentas e tecnologias utilizadas.

**d. Uma proposta inicial de como organizar a infraestrutura de sua aplicação, de acordo com o sistema de virtualização em que você pretende implementá-la.**

A solução de virtualização escolhida foi da Vmware, para uma boa organização, pretendo implementar as seguintes práticas:

- Criar pastas, tags e cores para agrupar máquinas virtuais por aplicação, projeto ou departamento
- Nomes Objetivos nas VMs e nos recursos de disco, no meu caso PB-Ansible, PB-WordPress e PB-Docker.
- Separar por Grupos, por exemplo Homologação e Produção
- Documentar as configurações de cada máquina para futuras resoluções de problemas, treinamentos e plano de continuidade
- Montar Scripts de Automação para as VMs, para evitar erros humanos
- Sempre fazer backup, definir a periodicidade com o negócio e executar testes de restore.

## TP7

Faça upload de uma versão inicial do capítulo 2/Proposta de Solução de seu Projeto de Bloco. Você deve contemplar:

**Uma pequena descrição textual teórica sobre a arquitetura da solução de virtualização que você pretende utilizar.**

Pretendo utilizar VMware, que é uma solução de virtualização que permite executar vários sistemas operacionais em um computador físico (host). É criada uma camada de abstração entre o hardware e o S.O. permitindo que vários sejam executados simultaneamente. As

vantagens de se usar essa solução é redução de custo com compra de host físicos, escalabilidade, já que podemos alocar os recursos das máquinas de acordo com a necessidade, elas são gerenciadas independente, oferece recursos de migração de máquinas virtuais em tempo real para balanceamento de carga, failover, provisionamento de recursos, Snapshots e Backups. Utilizarei também a plataforma Docker que tem por finalidade a criação, gerenciamento e execução de aplicativos em containers.

### **Comparações entre a forma como você pretende implantar seu ambiente e outras abordagens, usando outras ferramentas.**

Pretendo implementar a ferramenta de implantação, gerenciamento e orquestração de infraestrutura e aplicativos chamada Ansible, com ela vou instalar os componentes de servidores e o WordPress diretamente, que é conhecido neste projeto como WordPress Central ou PAI. Na sequência, também com Ansible, teremos um código que subirá um outro wordpress, só que desta vez em contêiner docker, que é uma unidade e software leve e portátil que tem tudo que um aplicativo precisa para ser executado, podendo facilmente ser movido entre ambientes e escalados.

### **Um planejamento passo a passo (com descrições de cada etapa) de como será feita a implantação da aplicação distribuída virtualizada.**

Etapas de Implantação da Aplicação:

1. Alocar recursos de Máquina Virtual no VMware – VM Ansible
2. Instalar o Sistema Operacional Linux Distribuição Ubuntu 22.04
3. Instalar Ansible
4. Construir um Playbook que:
  - 4.1. Instale pacotes no servidor  
Apache, mysql,php,python
  - 4.2. Instale as extensões PHP]
  - 4.3. Crie o Banco de Dados  
Crie o Usuario do Banco de Dados
  - 4.4. Baixa o WordPress  
Instale o WordPress  
Atualize site Apache  
Reinicie Apache  
Copie o arquivo Config  
Atualize o arquivo config do Wordpress
5. Executamos o Playbook
6. Subir códigos para o GitHub
7. Alocar recursos de Máquina Virtual no VMWare – VM WordPress
8. Instalar o Sistema Operacional Linux Distribuição Ubuntu 22.04
9. Rodar o Script Ansible para instalar o WordPress Central.
10. Validação e Testes
11. Alocar recursos de Máquina Virtual no VMWare – VM Docker
12. Instalar o Sistema Operacional Linux Distribuição Ubuntu 22.04



13. Instalar o Docker
14. Construir um Playbook que:
  - 13.1. Declaramos variáveis
  - 13.2. Declaramos nome e imagens dos containers de Banco de Dados
  - 13.3. Enviamos instruções de Usuários e Senha de Banco de Dados
  - 13.4. Declaramos nome e imagens dos containers de WordPress
  - 13.5. Enviamos instruções instalação do WordPress
14. Executamos o Playbook
15. Validação e Testes
16. Subir códigos para o GitHub

Um cronograma estimado com o prazo para execução de cada atividade.



Tarefas	Início	Término
1. Alocar recursos de Máquina Virtual no VMware – VM Ansible	02/04/2023	03/04/2023
2. Instalar o Sistema Operacional Linux Distribuição Ubuntu 22.04	02/04/2023	03/04/2023
3. Instalar Ansible	02/04/2023	03/04/2023
4. Construir um Playbook que:	04/04/2023	06/04/2023
4.1. Instale pacotes no servidor	04/04/2023	06/04/2023
4.2. Apache, mysql,php,python	04/04/2023	06/04/2023
4.3. Instale as extensões PHP	04/04/2023	06/04/2023
4.4. Crie o Banco de Dados	04/04/2023	06/04/2023
4.5. Crie o Usuario do Banco de Dados	04/04/2023	06/04/2023
4.6. Baixa o WordPress	04/04/2023	06/04/2023
4.7. Instala o WordPress	04/04/2023	06/04/2023
4.8. Atualiza site Apache	04/04/2023	06/04/2023
4.9. Reinicia Apache	04/04/2023	06/04/2023
4.10. Copia o arquivo Config	04/04/2023	06/04/2023
4.11. Atualiza o Arquivo config do WordPress	04/04/2023	06/04/2023
5. Executamos o Playbook	04/04/2023	06/04/2023
6. Subir códigos para o GitHub	04/04/2023	06/04/2023
7. Alocar recursos de Máquina Virtual no VMWare – VM WordPress	08/04/2023	09/04/2023
8. Instalar o Sistema Operacional Linux Distribuição Ubuntu 22.04	08/04/2023	09/04/2023
9. Rodar o Script Ansible para instalar o WordPress Central.	08/04/2023	09/04/2023
10. Validação e Testes	08/04/2023	09/04/2023
11. Alocar recursos de Máquina Virtual no VMWare – VM Docker	10/04/2023	11/04/2023
12. Instalar o Sistema Operacional Linux Distribuição Ubuntu 22.04	10/04/2023	11/04/2023
13. Instalar o Docker	10/04/2023	11/04/2023
14. Construir um Playbook que:	13/04/2023	15/04/2023
13.1. Declaramos Variaveis	13/04/2023	15/04/2023
13.2. Declaramos nome e imagens dos containers de Banco de Dados	13/04/2023	15/04/2023
13.3. Enviamos instruções de Usuarios e Senha de Banco de Dados	13/04/2023	15/04/2023
13.4. Declaramos nome e imagens dos containers de WordPress	13/04/2023	15/04/2023
13.5. Enviamos instruções de instalação do WordPress	13/04/2023	15/04/2023
14. Executamos o Playbook	13/04/2023	15/04/2023
15. Validação e Testes	13/04/2023	15/04/2023
16. Subir códigos para o GitHub	13/04/2023	15/04/2023

## ➤ TP8

Faça upload de uma versão inicial do capítulo 3/Implementação de seu Projeto de Bloco.

Você deve contemplar:

**Referências dos downloads e/ou versões de todos os componentes da solução implementada.**

Versão do Ansible:

```

silney@silney-Infnet:~$ ansible --version
ansible 2.9.27
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/silney/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.12 (default, Jul 21 2020, 15:19:50) [GCC 5.4.0 20160609]
silney@silney-Infnet:~$

```

Versão dos componentes de servidor:

```
- name: Instalar pacotes do servidor
  apt: name={{ item }} state=present
  become: yes
  with_items:
    - apache2
    - mysql-server
    - php8.1-mysql
    - php8.1
    - libapache2-mod-php8.1
    - python3-pymysql
```

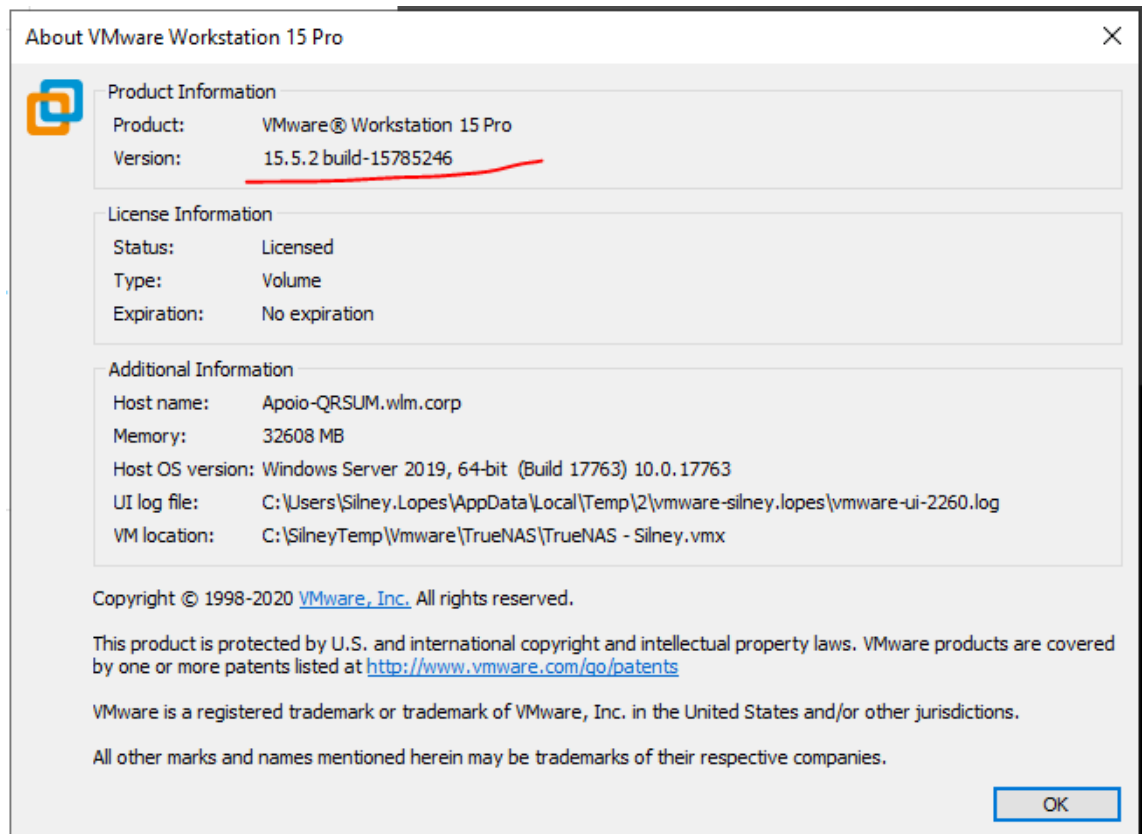
Versão do Docker:

```
silney@silney-infnet:~$ docker --version
Docker version 23.0.1, build a5ee5b1
```

Versão do Ubuntu:

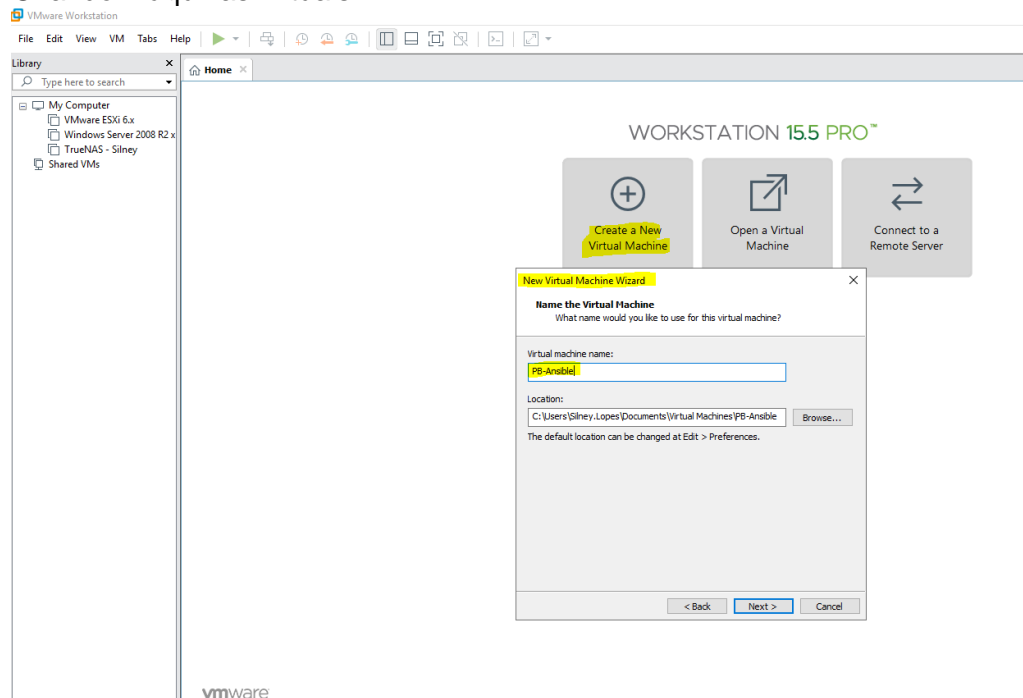
```
silney@silney-infnet:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.2 LTS
Release:        22.04
Codename:       jammy
silney@silney-infnet:~$
```

VMWare Workstation:

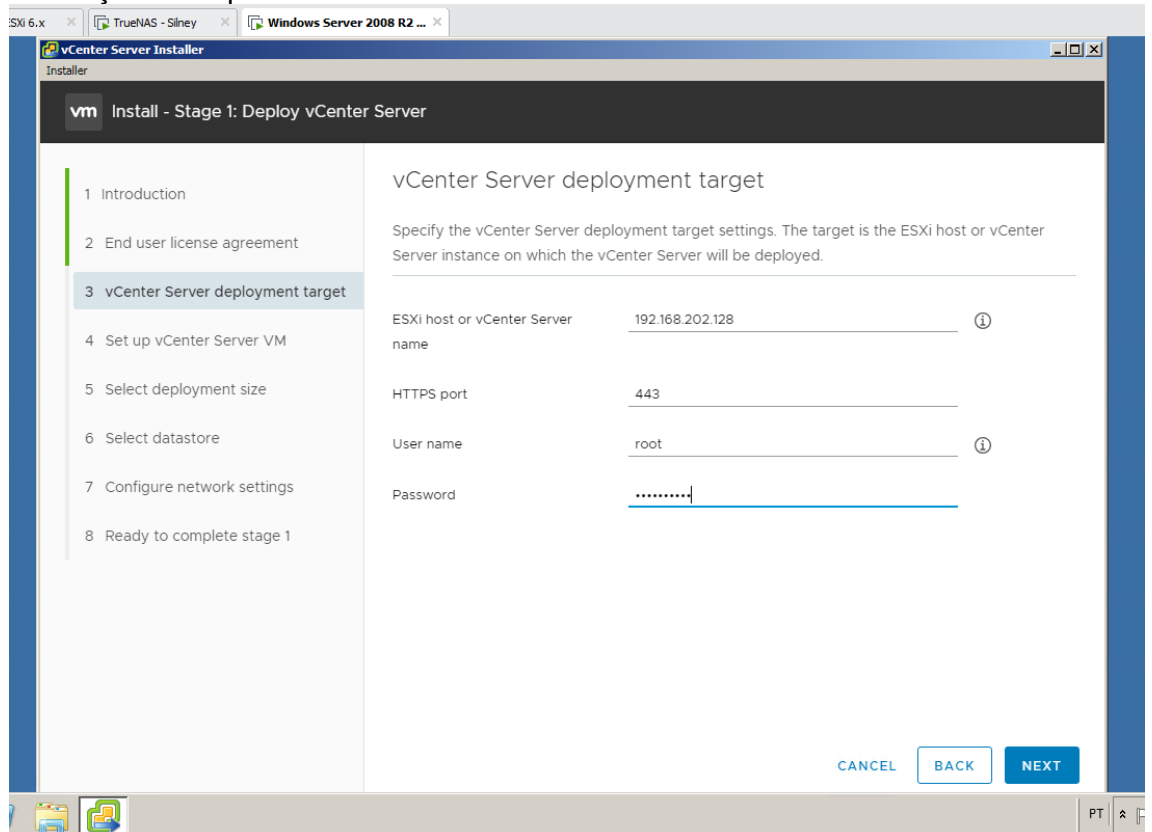


**Todos os passos/capturas de tela de implementação de sua infraestrutura de virtualização.**

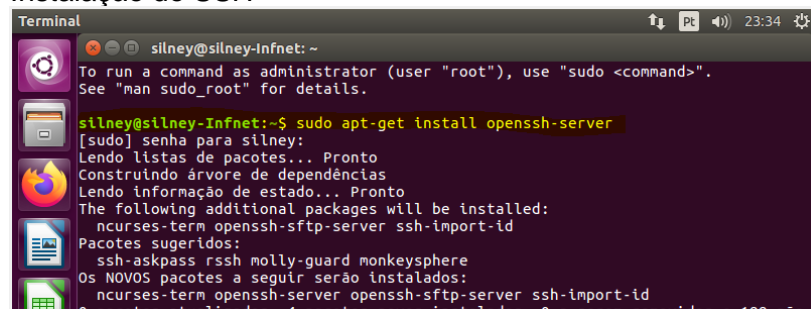
### Criando máquinas virtuais



## Instalação do vSphere



## Instalação do SSH



## Instalação do Ansible

```
silney@silney-Infnet:~$ sudo apt-add-repository ppa:ansible/ansible
[sudo] senha para silney:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications—au
tomate in a language that approaches plain English, using SSH, with no agents to install on remote systems.
http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
Mais informações: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Pressione [ENTER] para continuar ou ctrl-c para cancelar a adição

gpg: chaveiro /tmp/tmp1h2295b/secring.gpg criado
gpg: chaveiro /tmp/tmp1h2295b/pubring.gpg criado
gpg: requisitando chave 78B9C367 de servidor hkp - keyserver.ubuntu.com
gpg: /tmp/tmp1h2295b/trustdb.gpg: banco de dados de confiabilidade criado
gpg: chave 78B9C367: chave pública "Launchpad PPA for Ansible, Inc." importada
gpg: Número total processado: 1
gpg:      importados: 1 (RSA: 1)
OK
silney@silney-Infnet:~$ sudo apt-get update
Obter:1 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial InRelease [10.0 kB]
Atendo:2 http://br.archive.ubuntu.com/ubuntu xenial InRelease
Obter:3 http://security.ubuntu.com/ubuntu xenial-security InRelease [99,8 kB]
Obter:4 http://br.archive.ubuntu.com/ubuntu xenial-updates InRelease [99,8 kB]
Obter:5 http://br.archive.ubuntu.com/ubuntu xenial-backports InRelease [97,4 kB]
Obter:6 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 DEP-11 Metadata [327 kB]
Obter:7 http://br.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 DEP-11 Metadata [281 kB]
Obter:8 http://br.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 DEP-11 Metadata [5.956 B]
Obter:9 http://br.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Metadata [3.228 B]
Obter:10 http://br.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-11 Metadata [6.608 B]
Obter:11 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main amd64 Packages [704 B]
Obter:12 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main i386 Packages [704 B]
Obter:13 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial/main Translation-en [472 B]
Baixados 940 kB em 2s (422 kB/s)
Lendo listas de pacotes... Pronto
silney@silney-Infnet:~$ sudo apt-get install ansible
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informações de estado... Pronto
The following additional packages will be installed:
python-cffi-backend python-crypto python-cryptography python-ecdsa python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2 python-markupsafe python-paramiko
python-pkg-resources python-pyasn1 python-setuptools python-six python-yaml sshpass
Pacotes sugeridos:
python-crypto-dbg python-crypto-doc python-cryptography-doc python-cryptography-vectors python-enum34-doc python-jinja2-doc python-setuptools-doc
Os NOVOS pacotes a seguir serão instalados:
ansible python-cffi-backend python-crypto python-cryptography python-ecdsa python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2 python-markupsafe python-paramiko
python-pkg-resources python-pyasn1 python-setuptools python-six python-yaml sshpass
9 pacotes atualizados, 16 pacotes novos instalados, 0 a serem removidos e 188 não atualizados.
É preciso baixar 7.155 kB de arquivos.
Depois desta operação, 65,5 MB adicionais de espaço em disco serão usados.
Você quer continuar? [S/n]
Obter:1 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-markupsafe amd64 0.23-2build2 [15,5 kB]
Obter:2 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-jinja2 all 2.8-1ubuntu1 [106 kB]
Obter:3 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-yaml amd64 3.11-3build1 [103 kB]
Obter:4 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-crypto amd64 2.6.1-6ubuntu0.16.04.3 [246 kB]
Obter:5 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-six all 1.10.0-3 [10,9 kB]
Obter:6 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-ecdsa all 0.13-2ubuntu16.04.1 [36,2 kB]
Obter:7 http://br.archive.ubuntu.com/ubuntu xenial-updates/main amd64 python-paramiko all 1.16.0-1ubuntu2 [110 kB]
Obter:8 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-httplib2 all 0.9.1+dfsg-1 [34,2 kB]
Obter:9 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-pkg-resources all 20.7.0-1 [108 kB]
Obter:10 http://br.archive.ubuntu.com/ubuntu xenial/main amd64 python-setuptools all 20.7.0-1 [169 kB]
Obter:11 http://br.archive.ubuntu.com/ubuntu xenial/universe amd64 sshpass amd64 1.05-1 [10,5 kB]
```

## Instalação do Docker

```
silney@silney-infnet:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-03-23 23:39:37 -03; 18s ago
 TriggeredBy: ● docker.socket
    Docs: https://docs.docker.com
   Main PID: 3397 (dockerd)
     Tasks: 8
    Memory: 25.7M
      CPU: 895ms
   CGroup: /system.slice/docker.service
           └─3397 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.760841562-03:00" level=info msg="[core] [Channel #4] SubC
mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.761476929-03:00" level=info msg="[core] [Channel #4] Cha
mar 23 23:39:36 silney-infnet dockerd[3397]: time="2023-03-23T23:39:36.908418910-03:00" level=info msg="Loading containers: sta
mar 23 23:39:37 silney-infnet dockerd[3397]: time="2023-03-23T23:39:37.342150706-03:00" level=info msg="Default bridge (docke
```

Todos os passos/capturas de tela de configuração de sua aplicação.

Construindo Playbook Ansible para Wordpress

```
silney@silney-infnet:~$ ls
{changed: hosts 'MicrosoftTeams-image (66).png' playbook.yml README.md roles
```

Estrutura Playbook Ansible para Wordpress

```

silney@silney-infnet:~$ cd PBInfnet/
silney@silney-infnet:~/PBInfnet$ ls
hosts  playbook.yml  README.md  roles
silney@silney-infnet:~/PBInfnet$ cd roles/
silney@silney-infnet:~/PBInfnet/roles$ ls
mysql  php  server  wordpress
silney@silney-infnet:~/PBInfnet/roles$ █

```

- PBInfnet/hosts.yml:
 

```

[wordpress]

192.168.31.35

```
- PBInfnet/playbook.yml:
 

```

---
- hosts: wordpress
  become: yes
  remote_user: silney

  roles:
    - server
    - php
    - mysql
    - wordpress

```
- PBInfnet/roles/server/tasks/main.yml:
 

```

---
# tasks file for server

- name: Atualiza apt cache
  apt: update_cache=yes cache_valid_time=3600
  become: yes

- name: Instalar pacotes do servidor
  apt: name={{ item }} state=present
  become: yes
  with_items:
    - apache2
    - mysql-server
    - php8.1-mysql
    - php8.1
    - libapache2-mod-php8.1
    - python3-pymysql

```
- PBInfnet/roles/php/tasks/main.yml:
 

```

---
# tasks file for php

- name: Instala as extensoes do PHP 7.4
  apt: name={{ item }} state=present
  become: yes
  with_items:
    - php8.1-gd
    - php8.1-ssh2

```

- PBIInfnet/roles/mysql/defaults/main.yml:  

```

---
# defaults file for mysql
wp_mysql_db: wordpress
wp_mysql_user: wordpress
wp_mysql_password: wp_db_password

```
- PBIInfnet/roles/mysql/tasks/main.yml:  

```

---
# tasks file for mysql
- name: Cria banco de dados do wordpress
  mysql_db: name={{ wp_mysql_db }} state=present

- name: Cria o usuario do banco de dados wordpress
  mysql_user:
    name={{ wp_mysql_user }}
    password={{ wp_mysql_password }}
    priv=*.*:ALL

```
- PBIInfnet/roles/wordpress/handlers/main.yml:  

```

---
# handlers file for wordpress

- name: restart apache
  become: true
  service: name=apache2 state=restarted

```
- PBIInfnet/roles/wordpress/tasks/main.yml:  

```

---
# tasks file for wordpress
- name: Baixar Wordpress
  become: true
  get_url:
    url=https://wordpress.org/latest.tar.gz
    dest=/tmp/wordpress.tar.gz
    validate_certs=no

- name: Extrair WordPress
  become: true
  unarchive:
    src=/tmp/wordpress.tar.gz
    dest=/var/www/
    copy=no

- name: Atualizar site Apache
  become: true
  lineinfile:
    dest=/etc/apache2/sites-enabled/000-default.conf
    regexp="(.)+DocumentRoot /var/www/html"
    line="DocumentRoot /var/www/wordpress"
  notify:
    - restart apache

- name: Copy sample config file
  become: true

```



```
command: mv /var/www/wordpress/wp-config-sample.php
/var/www/wordpress/wp-config.php creates=/var/www/wordpress/wp-
config.php
```

```
- name: Update WordPress config file
  become: true
  lineinfile:
    dest=/var/www/wordpress/wp-config.php
    regexp="{{ item.regex }}"
    line="{{ item.line }}"
  with_items:
    - {regex: 'database_name_here', line: "define('DB_NAME',
'{{wp_mysql_db}}');"}
    - {regex: 'username_here', line: "define('DB_USER',
'{{wp_mysql_user}}');"}
    - {regex: 'password_here', line: "define('DB_PASSWORD',
'{{wp_mysql_password}}');"}
    # - {regex: "define\\('DB_NAME', '(.)+'\\);", 'line': "define('DB_NAME',
'{{wp_mysql_db}}');"}
    # - {regex: "define\\('DB_USER', '(.)+'\\);", 'line': "define('DB_USER',
'{{wp_mysql_user}}');"}
    # - {regex: "define\\('DB_PASSWORD', '(.)+'\\);", 'line':
"define('DB_PASSWORD', '{{wp_mysql_password}}');"}

```

## Executando o Playbook

```
silney@silney-infnet:~$ cd PBInfnet/
silney@silney-infnet:~/PBInfnet$ ansible-playbook playbook.yml --ask-become-pass
BECOME password:

PLAY [wordpress] *****

TASK [Gathering Facts] *****
ok: [192.168.31.35]

TASK [server : Atualiza apt cache] *****
ok: [192.168.31.35]

TASK [server : Instalar pacotes do servidor] *****
ok: [192.168.31.35] => (item=apache2)
ok: [192.168.31.35] => (item=mysql-server)
ok: [192.168.31.35] => (item=php8.1-mysql)
ok: [192.168.31.35] => (item=php8.1)
ok: [192.168.31.35] => (item=libapache2-mod-php8.1)
ok: [192.168.31.35] => (item=python3-pymysql)

TASK [php : Instala as extensoes do PHP 7.4] *****
ok: [192.168.31.35] => (item=php8.1-gd)
ok: [192.168.31.35] => (item=php-ssh2)

TASK [mysql : Cria banco de dados do wordpress] *****
changed: [192.168.31.35]

TASK [mysql : Cria o usuario do banco de dados wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Baixar Wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Extrair Wordpress] *****
changed: [192.168.31.35]

TASK [wordpress : Atualizar site Apache] *****
changed: [192.168.31.35]

TASK [wordpress : Copy sample config file] *****
changed: [192.168.31.35]

TASK [wordpress : Update WordPress config file] *****
changed: [192.168.31.35] => (item={'regex': 'database_name_here', 'line': "define('DB_NAME', 'wordpress');"})
changed: [192.168.31.35] => (item={'regex': 'username_here', 'line': "define('DB_USER', 'wordpress');"})
changed: [192.168.31.35] => (item={'regex': 'password_here', 'line': "define('DB_PASSWORD', 'wp_db_password');"})

RUNNING HANDLER [wordpress : restart apache] *****
changed: [192.168.31.35]

PLAY RECAP *****
192.168.31.35      : ok=12  changed=8  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

silney@silney-infnet:~/PBInfnet$
```

## Construindo Playbook Ansible para Docker

```
silney@silney-infnet:~$ cd /etc/ansible/  
silney@silney-infnet:/etc/ansible$ nano hosts
```

```
[docker]  
192.168.31.254  
[docker:vars]  
ansible_user=silney  
ansible_password=maicjc  
ansible_become_password=maicjc  
ansible_python_interpreter=/usr/bin/python3  
^G Help      ^O Write Out  ^W Where Is  
^X Exit      ^R Read File  ^\ Replace
```

```
silney@silney-infnet:~/TP4_TP5$ nano playbook.yml
```

```
--  
- hosts: docker  
  vars:  
    db_volume: mariadb  
    wordpress: wordpress  
  tasks:  
    - name: Deploy MariaDB server  
      docker_container:  
        image: mariadb  
        name: mariadb  
        volumes:  
          - "{{db_volume}}:/var/lib/mysql"  
        env:  
          MYSQL_ROOT_PASSWORD: password  
          MYSQL_PASSWORD: infnet  
          MYSQL_DATABASE: db_TP4_TP5  
          MYSQL_USER: infnet  
  
    - name: Deploy WordPress  
      docker_container:  
        image: wordpress  
        name: wordpress  
        restart_policy: always  
        ports:  
          - "8080:80"  
        links:  
          - "{{db_volume}}:/var/lib/mysql"  
        volumes:  
          - "{{wordpress}}:/var/www/html"  
        env:  
          MYSQL_PASSWORD: infnet  
          MYSQL_DATABASE: db_TP4_TP5  
          MYSQL_USER: infnet  
          MYSQL_HOST: mariadb
```

## Execução do código

```
silney@silney-infnet:~/TP4_TP5$ export ANSIBLE_HOST_KEY_CHECKING=False
silney@silney-infnet:~/TP4_TP5$ ansible-playbook playbook.yml -l docker --ask-become-pass
BECOME password:

PLAY [docker] *****

TASK [Gathering Facts] *****
ok: [192.168.31.254]

TASK [Deploy MariaDB server] *****
changed: [192.168.31.254]

TASK [Deploy WordPress] *****
changed: [192.168.31.254]

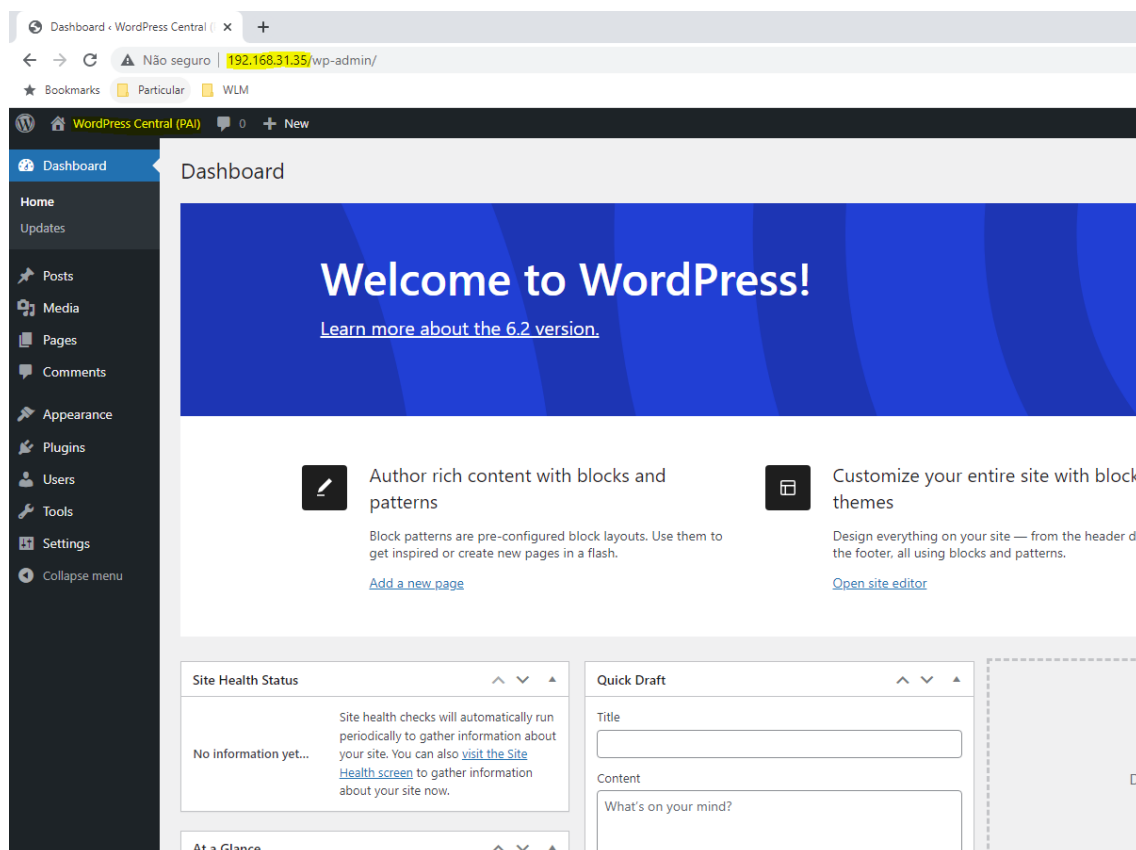
PLAY RECAP *****
192.168.31.254 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

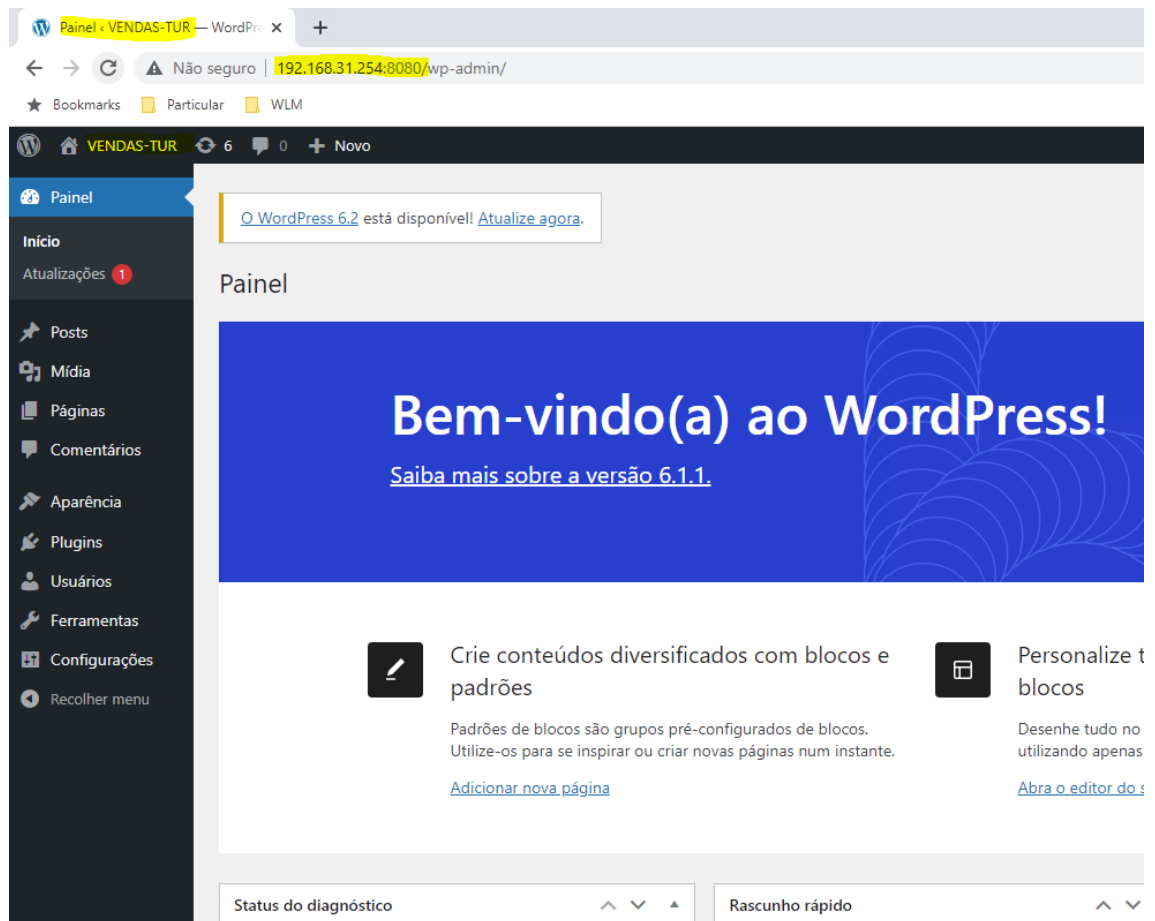
## Consulta da Container Docker

```
For more help on how to use Docker, head to https://docs.docker.com/go/guides/

silney@silney-infnet:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
silney@silney-infnet:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
aebe85500a30   wordpress "docker-entrypoint.s..." 17 seconds ago Up 14 seconds 0.0.0.0:8080->80/tcp   wordpress
589b33d63eed   mariadb   "docker-entrypoint.s..." 21 seconds ago Up 18 seconds 3306/tcp          mariadb
silney@silney-infnet:~$
```

### e. Uma captura de tela de sua aplicação em funcionamento.





## 4. Conclusão

Neste TPs, coletamos as evidências de versões, virtualizações e aplicações que implementamos ao decorrer do projeto de bloco.