



**ESCOLA SUPERIOR DA TECNOLOGIA DA INFORMAÇÃO  
REDES DE COMPUTADORES**

**Projeto de bloco: Arquitetura de Infraestrutura de Aplicações**

**ASSESSMENT**

**Professor: Fabiano Gisbert**

**Aluno: Vinícius Xavier**

**07/12/21**

<b>1. Introdução.....</b>	<b>4</b>
<b>2. Justificativa.....</b>	<b>5</b>
<b>3. Aplicação distribuída.....</b>	<b>5</b>
3.1 Criação de um playbook Ansible.....	6
3.1.1 Teste do playbook criado.....	6
3.1.2 Aplicação do playbook criado.....	6
3.2 Criação do repositório público no Github.....	7
3.2.1 Commit e push do código criado.....	7
3.2.2 Alteração, novo commit e push do código criado.....	7
3.2.3 URL do repositório para acesso via internet.....	8
<b>4. Organização da Infraestrutura .....</b>	<b>9</b>
4.1 Instalação do Docker.....	10
4.1.1 Commit e push do código criado.....	10
4.1.2 Comando para instalações de pacotes de suporte.....	10
4.1.3 Chave de assinatura digital do Projeto Docker.....	11
4.1.4 Comando para identificação da chave recém criada.....	11
4.1.5 Repositório de pacotes do Projeto Docker ao sistema.....	11
4.1.6 Download da lista de pacotes disponíveis nos repositórios configurados.....	12
4.1.7 Instalação do sistema Docker.....	12
4.2 Criação de containers.....	13
4.2.1 Container de banco de dados e verificação do status do mesmo.....	14

4.2.2 WordPress e verificação do status de ambos containers recém-criados.....	15
4.2.3 Ambiente WordPress que receberá as aplicações.....	15
4.3 Criação de containers utilizando o Ansible.....	16
4.3.1 Criação do arquivo hosts dentro do diretório /wordpress-docker.....	16
4.3.2 Arquivo playbook.yml dentro do diretório /wordpress-docker.....	17
4.3.3 Instalação do gerenciador de bibliotecas Python3-PIP.....	17
4.3.4 Instalação da biblioteca pysphere.....	18
4.3.5 Teste do playbook criado.....	18
4.3.6 Acesso ao ambiente WordPress (localhost).....	19
4.3.7 Criação de um novo repositório no GitHub.....	19
4.3.8 Exportar playbook para repositório Github.....	19
4.3.9 URL para acesso via internet.....	20
5. Cronograma de Atividades.....	20
5.1 Cronograma atual.....	20
5.1.1 Possível bloqueio.....	21
5.1.2 URL para acessar o cronograma.....	22
6. Conclusão.....	22
7. Referências bibliográficas.....	26

## 1. INTRODUÇÃO

A aplicação que será criada é um blog fictício dedicado à gastronomia brasileira, com o intuito de disponibilizar receitas, modos de preparo e outros conteúdos informativos sobre culinária para cidadãos holandeses e outros cidadãos estrangeiros residentes nos Países Baixos. O objetivo é auxiliar na difusão da gastronomia brasileira nos Países Baixos.

O projeto passará por diferentes fases, desde a teorização, passando pela implementação com auxílio de ferramentas de integração e repositório de código-fonte, monitoramento do fluxo de trabalho através de ferramenta web utilizando a metodologia ágil (scrum) e periódica revisão da infraestrutura virtual construída dentro de uma infraestrutura física.

O acesso a esta aplicação será feito através da Internet. Após o deploy do código-fonte do Docker, o cliente receberá três pares de credenciais (username e senha): o primeiro par para o sistema operacional (Linux Ubuntu), o segundo par para acessar o contêiner da aplicação no Docker e o terceiro par será para acessar o blog através da URL fictícia `brfoods.nl/wp-admin`. Lembrando que o cliente já possui acesso ao VMWare Fusion que foi instalado em sua máquina.

## 2. JUSTIFICATIVA

A criação do blog WordPress, com nome de domínio fictício `brfoods.nl`, desejado pelo cliente teve três fatores principais: dificuldade de inserção no mercado local de trabalho, paixão pela gastronomia brasileira e amplo acesso à internet dos residentes dos Países Baixos.

Apesar de ser um país conhecido por sua inclinação e abertura ao mercado global, para advogados expatriados que não atuam ou não possuem especialização em Direito Internacional (Público e Privado) ou em Comércio Exterior, têm dificuldades iniciais para se adaptarem à legislação e práticas locais. Portanto, o blog tornar-se-à uma forma alternativa de fonte de renda para o cliente.

Além da paixão e apreço do cliente pela gastronomia brasileira, principalmente a culinária mineira, o cliente entendeu que poderia ser algo promissor tendo em vista as poucas variedades gastronômicas locais nos Países Baixos.

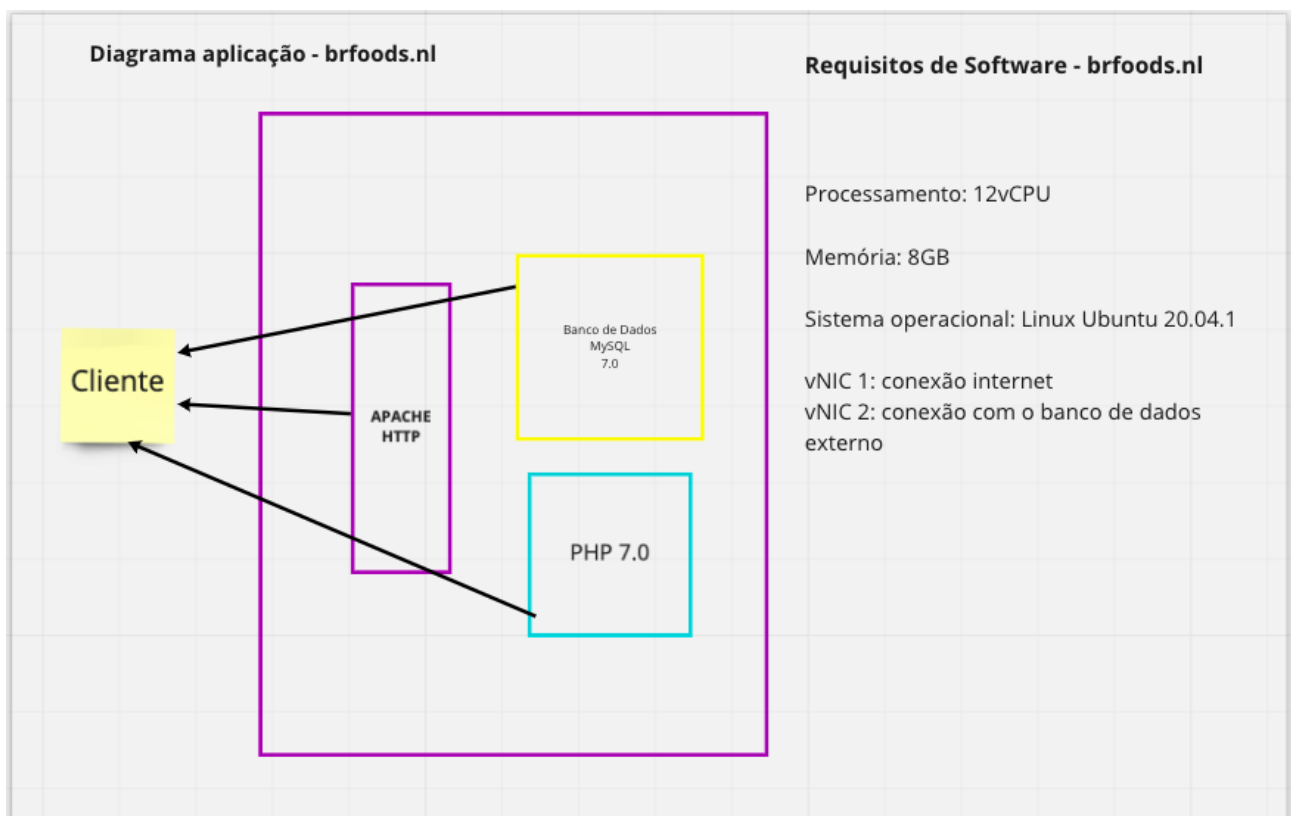
Ainda, em recente pesquisa realizada pela Statista Research Development (2021) foi estimado que cerca de 97% das casas holandesas possuem acesso à internet através de provedores de

serviço especializados em banda larga. E em não tão distante pronunciamento oficial do Governo Holandês (2018), foi exposta a intenção de atingir o número de 100% dos cidadãos e residentes holandeses com acesso à internet via banda larga em suas casas. Logo, vê-se a necessidade da presença online e disponibilização de conteúdo e serviços através da internet.

### 3. APLICAÇÃO DISTRIBUÍDA

A aplicação será um blog de culinária, focado em gastronomia brasileira criado no WordPress. O código-fonte da aplicação a ser criado será armazenado em um repositório da plataforma GitHub e sua implantação será realizada de forma automatizada com o auxílio da ferramenta Ansible, que também terá o código armazenado no GitHub.

Requisitos de software para aplicação:



### 3.1 Criação de um playbook Ansible

- através da linha de comando enquanto root user: touch nano playbook.yml

```
viniciusxavier@ubuntu:~/Desktop$ nano playbook.yml
viniciusxavier@ubuntu:~/Desktop$ ansible-playbook playboo
k.yml
```

#### 3.1.1 Teste do playbook criado

```
viniciusxavier@ubuntu:~/Desktop$ ansible wordpress -m ping -u viniciusxavier
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:riaC2LjrTMkMe82X0p4HwgIPCfvNQKywBwumaH5NXHc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
127.0.0.1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
viniciusxavier@ubuntu:~/Desktop$ ansible wordpress -m ping -u viniciusxavier
127.0.0.1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

```
viniciusxavier@ubuntu:~/Desktop$ ansible-playbook playbook.yml

PLAY [wordpress] *****

TASK [Gathering Facts] *****
ok: [127.0.0.1]

TASK [NTP Installation Process (time sync)] *****
changed: [127.0.0.1]

TASK [NTP Up and Running (confirmation)] *****
ok: [127.0.0.1]

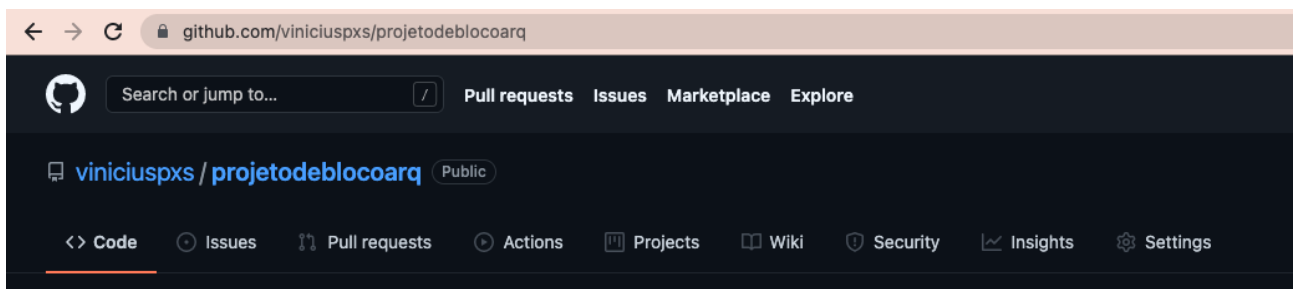
PLAY RECAP *****
127.0.0.1 : ok=3 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

#### 3.1.2 Aplicação do playbook criado

```
GNU nano 4.8                                playbook.yml
- hosts: wordpress
  remote_user: viniciusxavier
  become: yes
  tasks:
  - name: NTP Installation Process (time sync)
    apt: name=ntp state=present
  - name: NTP Up and Running (confirmation)
    service: name=ntp state=started enabled=yes

[ Read 8 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line
```

## 3.2 Criação do repositório público no Github



### 3.2.1 Commit e push do código criado

- 6.1) git init
- 6.2) git add \*
- 6.3) git config --global user.email viniciusp.xavier@gmail.com
- 6.4) git config --global user.name vinciuspxs
- 6.5) git remote add origin <https://github.com/viniciuspxs/projetodeblocoarq.git>

### 3.2.2 alteração, novo commit e push do código criado

comando: git push -u origin master

```
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git init
Reinitialized existing Git repository in /home/viniciusxavier/Desktop/wordpress-ansible2/.git/
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git add *
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git commit -m "importando playbook WordPress"
[master (root-commit) 3990cd2] importando playbook WordPress
39 files changed, 618 insertions(+)
create mode 100644 README.md
create mode 100644 hosts
create mode 100644 playbook.yml
create mode 100644 roles/mysql/.travis.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/.travis.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/.travis.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/.travis.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
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git config --global user.email viniciusp.xavier@gmail.com
```

```
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git config --global user.email viniciusp.xavier@gmail.com
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git config --global user.name viniciusp.xs
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git commit -m "importando playbook WordPress"
On branch master
nothing to commit, working tree clean
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git remote add origin https://github.com/viniciusp.xs/projetodeblocoarq.git
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$ git push -u origin master
Username for 'https://github.com': viniciusp.xs
Password for 'https://viniciusp.xs@github.com':
Enumerating objects: 55, done.
Counting objects: 100% (55/55), done.
Delta compression using up to 2 threads
Compressing objects: 100% (25/25), done.
Writing objects: 100% (55/55), 6.05 KiB | 476.00 KiB/s, done.
Total 55 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/viniciusp.xs/projetodeblocoarq.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
viniciusxavier@ubuntu:~/Desktop/wordpress-ansible2$
```

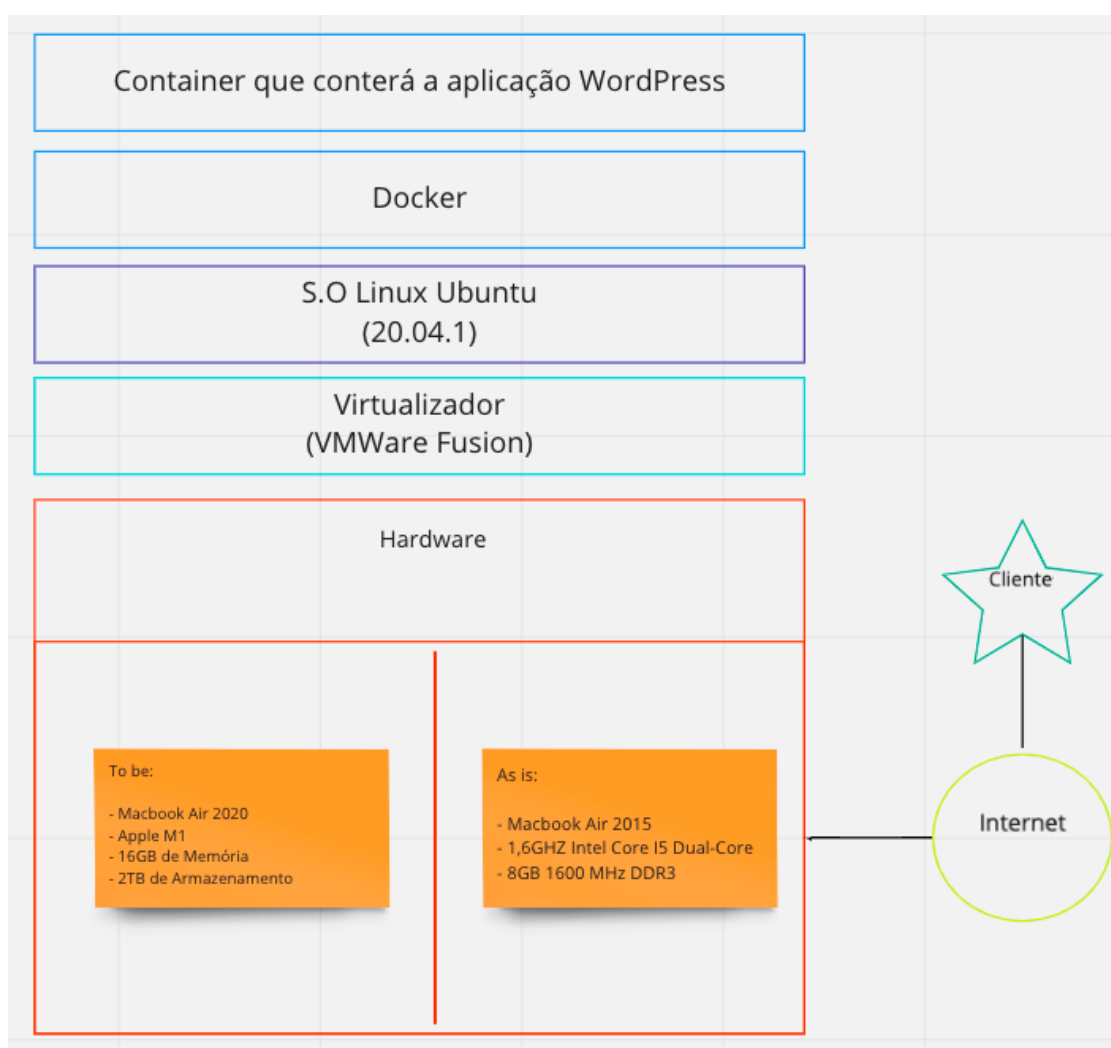
### 3.2.3 URL do repositório para acesso via internet

<https://github.com/viniciusp.xs/projetodeblocoarq>



## 4. ORGANIZAÇÃO DA INFRAESTRUTURA

Uma infraestrutura virtual será criada a partir de uma infraestrutura física. A infraestrutura virtual terá um virtualizador logo acima da infraestrutura física, que será um VMWare Fusion. Na infraestrutura virtual será instalado uma máquina virtual com Sistema Operacional Linux Ubuntu - versão 20.04.1. No servidor Linux Ubuntu será instalado a plataforma Docker e aplicação WordPress será armazenada em um container.



Como o blog é de caráter pessoal e temporário, o cliente optou por realizar a implementação da infraestrutura em um computador pessoal (on-premises). Tendo em vista que o sistema

operacional da máquina física é um OS X, foi escolhido o virtualizador VMWare Fusion em detrimento a outras soluções de virtualização no mercado, pois além de possuir uma versão gratuita que atende as necessidades da aplicação e da infraestrutura virtual, concluiu-se que o VMWare Fusion é o melhor apresenta compatibilidade com o OS X, além de fácil acesso à sua documentação nos portais oficiais da VMWare.

O blog WordPress (aplicação) será desenvolvido em um container Docker. Por mais que a infraestrutura física seja de uso pessoal do cliente e não há o intuito inicial de manter o blog por mais de dois anos, o cliente não descarta a possibilidade de mantê-lo em um cenário improvável e incrementá-lo caso haja uma considerável demanda pelos conteúdos e serviços oferecidos.

Portanto, como o Docker é uma plataforma flexível que permite a mobilidade dos seus contêineres para outra infraestrutura virtual desenvolvida com recursos semelhantes e mesmo sistema operacional (Linux Ubuntu), no caso de uma possível expansão das atividades do cliente relacionadas ao blog, ele terá a possibilidade de locomover o seu container para uma solução de nuvem pública ou outra virtualização on-premises.

## 4.1 Instalação do Docker

4.1.1 Remoção do MySQL e Apache através da linha de comando para evitar possíveis conflitos com as portas de comunicação a serem usadas nos containers:

`sudo apt-get remove apache2 mysql-server`

```
riniciusxavier@ubuntu:~$ mkdir wordpress-docker
riniciusxavier@ubuntu:~$ cd wordpress-docker/
riniciusxavier@ubuntu:~/wordpress-docker$ sudo apt-get remove apache2 mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  apache2-bin apache2-data apache2-utils libaio1 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libbcgi-fast-perl libbcgi-pm-perl libbcgi-perl
  libhtml-template-perl liblvm1 liblua5.2-0 libmecab2 linux-headers-5.11.0-34-generic linux-hwe-5.11-headers-5.11.0-34 linux-image-5.11.0-34-generic
  linux-modules-5.11.0-34-generic linux-modules-extra-5.11.0-34-generic mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0
  mysql-server-8.0 mysql-server-core-8.0
Use 'sudo apt autoremove' to remove them.
The following packages will be REMOVED:
  apache2 mysql-server
0 upgraded, 0 newly installed, 2 to remove and 39 not upgraded.
After this operation, 654 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 236898 files and directories currently installed.)
Removing apache2 (2.4.41-4ubuntu3.6) ...
Removing mysql-server (8.0.26-0ubuntu0.20.04.3) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for ufw (0.36-6) ...
```

4.1.2 Executar comando para instalações de pacotes de suporte que serão necessários para assinatura digital e download de pacotes que serão necessários em um momento futuro:

```

viniciusxavier@ubuntu:~$ sudo apt-get install \
> apt-transport-https \
> ca-certificates \
> curl \
> software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20210119~20.04.2).
ca-certificates set to manually installed.
software-properties-common is already the newest version (0.98.9.5).
software-properties-common set to manually installed.
The following packages were automatically installed and are no longer required:
  libllvm11 linux-headers-5.11.0-34-generic linux-hwe-5.11-headers-5.11.0-34
  linux-image-5.11.0-34-generic linux-modules-5.11.0-34-generic
  linux-modules-extra-5.11.0-34-generic
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  apt-transport-https curl
0 upgraded, 2 newly installed, 0 to remove and 39 not upgraded.
Need to get 166 kB of archives.
After this operation, 573 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/universe amd64 apt-transport-https all 2.0.6 [4,680 B]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.7 [161 kB]
Fetched 166 kB in 1s (225 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 235568 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.0.6_all.deb ...
Unpacking apt-transport-https (2.0.6) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.68.0-1ubuntu2.7_amd64.deb ...
Unpacking curl (7.68.0-1ubuntu2.7) ...
Setting up apt-transport-https (2.0.6) ...
Setting up curl (7.68.0-1ubuntu2.7) ...
Processing triggers for man-db (2.9.1-1) ...

```

4.1.3 Executar comando para download da chave de assinatura digital do Projeto Docker:

```

viniciusxavier@ubuntu:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK

```

4.1.4 Executar comando para identificação da chave recém criada

```

viniciusxavier@ubuntu:~$ sudo apt-key fingerprint 0EBFCD88
pub      rsa4096 2017-02-22 [SCEA]
          9DC8 5822 9FC7 DD38 854A  E2D8 8D81 803C 0EBF CD88
uid          [ unknown] Docker Release (CE deb) <docker@docker.com>
sub      rsa4096 2017-02-22 [S]

```

4.1.5 Adicionar o repositório de pacotes do Projeto Docker ao sistema

```

viniciusxavier@ubuntu:~/wordpress-docker$ sudo add-apt-repository \
> "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
> $(lsb_release -cs) \
> stable"
Hit:1 http://ppa.launchpad.net/ansible/ansible/ubuntu focal InRelease
Hit:2 https://download.docker.com/linux/ubuntu focal InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [29.0 kB]
Get:8 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [62.5 kB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2,464 B]
Get:10 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [283 kB]
Get:11 http://us.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [361 kB]
Get:12 http://us.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:13 http://us.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [10.4 kB]
Fetched 1,078 kB in 4s (259 kB/s)
Reading package lists... Done

```

#### 4.1.6 Download da lista de pacotes disponíveis nos repositórios configurados

```

viniciusxavier@ubuntu:~/wordpress-docker$ sudo apt-get update
Hit:1 http://ppa.launchpad.net/ansible/ansible/ubuntu focal InRelease
Hit:2 https://download.docker.com/linux/ubuntu focal InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:5 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:6 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Reading package lists... Done

```

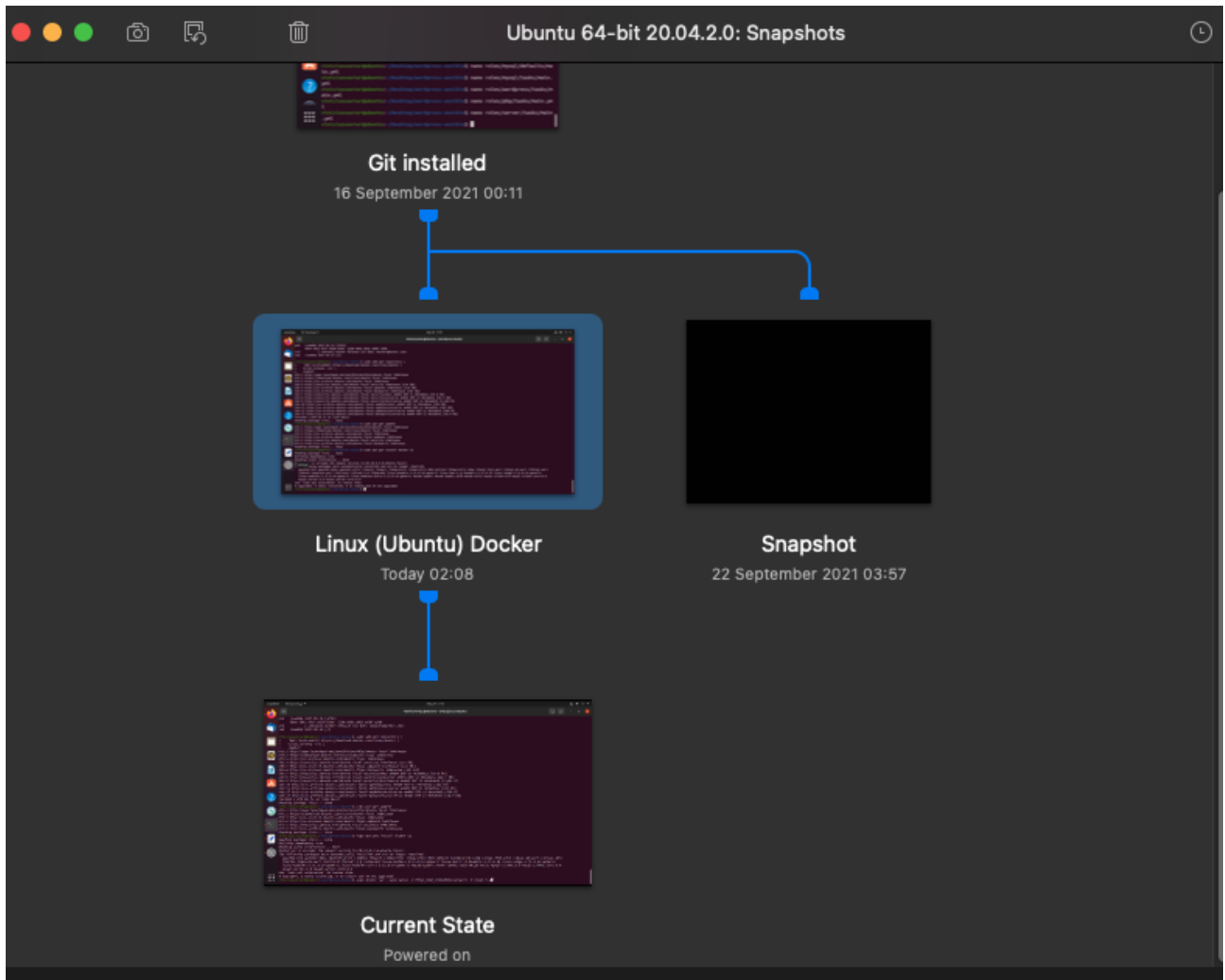
#### 4.1.7 Instalação do sistema Docker através do comando sudo apt-get install docker-ce

```

rinciusxavier@ubuntu:~$ sudo apt-get install docker-ce
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  liblvm1 linux-headers-5.11.0-34-generic linux-hwe-5.11-headers-5.11.0-34 linux-image-5.11.0-34-generic linux-modules-5.11.0-34-generic
  linux-modules-extra-5.11.0-34-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  containerd.io docker-ce-cli docker-ce-rootless-extras docker-scan-plugin pigz slurp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following NEW packages will be installed:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-scan-plugin pigz slurp4netns
0 upgraded, 7 newly installed, 0 to remove and 39 not upgraded.
Need to get 96.7 MB of archives.
After this operation, 406 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 https://download.docker.com/linux/ubuntu focal/stable amd64 containerd.io amd64 1.4.9-1 [24.7 MB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 slurp4netns amd64 0.4.3-1 [74.3 kB]
Get:4 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce-cli amd64 5:20.10.8~3-0-ubuntu-focal [38.8 MB]
Get:5 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce amd64 5:20.10.8~3-0-ubuntu-focal [21.2 MB]
Get:6 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce-rootless-extras amd64 5:20.10.8~3-0-ubuntu-focal [7,917 kB]
Get:7 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-scan-plugin amd64 0.8.0-ubuntu-focal [3,889 kB]
Fetched 96.7 MB in 9s (10.2 MB/s)
Selecting previously unselected package pigz.
(Reading database ... 235579 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_amd64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../1-containerd.io_1.4.9-1_amd64.deb ...
Unpacking containerd.io (1.4.9-1) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../2-docker-ce-cli_5%3a20.10.8~3-0-ubuntu-focal_amd64.deb ...
Unpacking docker-ce-cli (5:20.10.8~3-0-ubuntu-focal) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../3-docker-ce_5%3a20.10.8~3-0-ubuntu-focal_amd64.deb ...
Unpacking docker-ce (5:20.10.8~3-0-ubuntu-focal) ...
Selecting previously unselected package docker-ce-rootless-extras.
Preparing to unpack .../4-docker-ce-rootless-extras_5%3a20.10.8~3-0-ubuntu-focal_amd64.deb ...
Unpacking docker-ce-rootless-extras (5:20.10.8~3-0-ubuntu-focal) ...
Selecting previously unselected package docker-scan-plugin.

```

## 4.2 Criação de containers



#### 4.2.1 Criação do container de banco de dados e verificação do status do mesmo

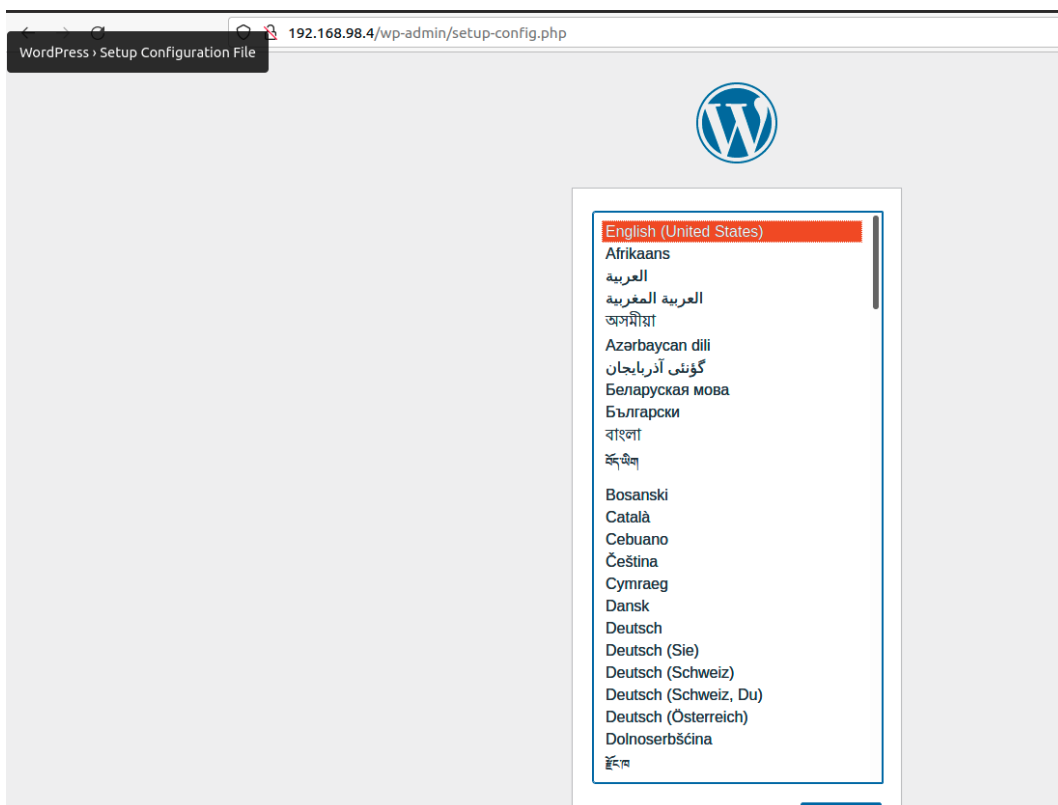
```
viniciusxavier@ubuntu:~/wordpress-docker$ sudo docker run --name banco -e MYSQL_ROOT_PASSWORD=senha123 -d mysql:5.7
Unable to find image 'mysql:5.7' locally
5.7: Pulling from library/mysql
07aded7c29c6: Pull complete
f68b8cbd22de: Pull complete
30c1754a28c4: Pull complete
1b7cb4d6fe05: Pull complete
79a41dc56b9a: Pull complete
0a75f2042fb: Pull complete
b Text Editor .7: Pull complete
5e11fe494f45: Pull complete
9c7de1f889a7: Pull complete
cf6a13d05a76: Pull complete
fc5aa81f393a: Pull complete
Digest: sha256:360c7488c2b5d112804a74cd272d1070d264eef4812d9a9cc6b8ed68c3546189
Status: Downloaded newer image for mysql:5.7
98d9bcbfedb18bc124654dd6b7e4992f8ad6d1b4f6470e1d294a3210b63b758
viniciusxavier@ubuntu:~/wordpress-docker$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
98d9bcbfedb1   mysql:5.7 "docker-entrypoint.s..." 35 seconds ago Up 33 seconds 3306/tcp, 33060/tcp      banco
viniciusxavier@ubuntu:~/wordpress-docker$
```

#### 4.2.2 Criação do container do WordPress e verificação do status de ambos containers recém-criados

```
viniciusxavier@ubuntu:~/wordpress-docker$ sudo docker run --name meusite --link banco:mysql -p 80:80 -d wordpress
Unable to find image 'wordpress:latest' locally
latest: Pulling from library/wordpress
f8416d8bac72: Pull complete
2259392b425a: Pull complete
cfb39fc3daf5: Pull complete
5c501de24ca4: Pull complete
ccf5f97ffc5c: Pull complete
a408db913f46: Pull complete
43600da0ccdc: Pull complete
55db4904bf5d: Pull complete
ab673d231350: Pull complete
341045b54845: Pull complete
9602a80d4b1d: Pull complete
030541c23d91: Pull complete
9f845a70c89d: Pull complete
3dd2d3411943: Pull complete
a32a7f3ec191: Pull complete
864d321b5301: Pull complete
4740a7290d89: Pull complete
9bf80f14cc3a: Pull complete
01605ae73ede: Pull complete
ca00a0f9ead6: Pull complete
720eba734fe9: Pull complete
Digest: sha256:13176e07ef0e43617d871be62f08a3b4e12459952fa905bbd3d63f7ae8c194d2
Status: Downloaded newer image for wordpress:latest
15632690d4e1d14b715a78350245a641d9a0ce0af87ac8adc354027bb57375d0
viniciusxavier@ubuntu:~/wordpress-docker$ sudo docker ps
```

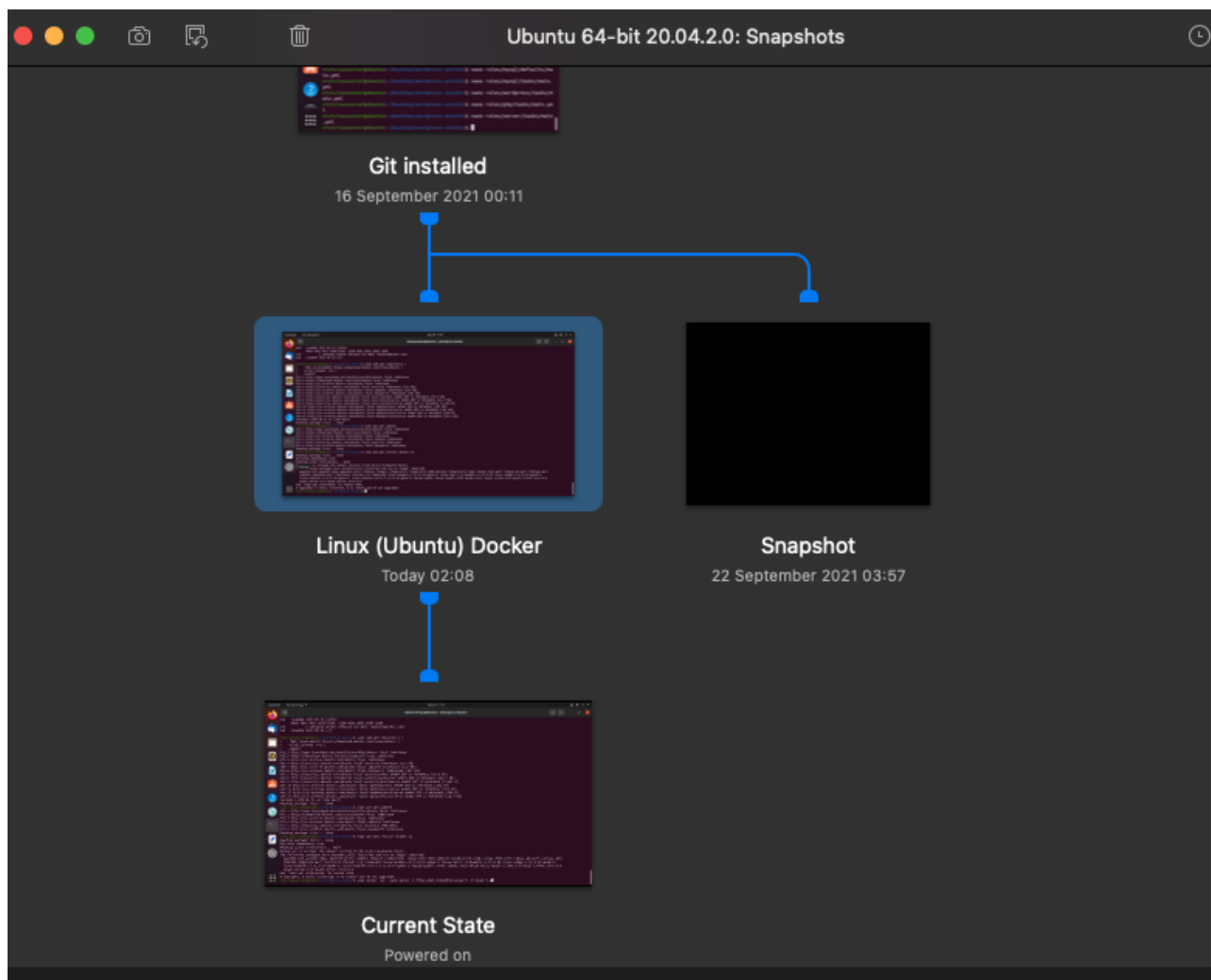
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
15632690d4e1	wordpress	"docker-entrypoint.s..."	About a minute ago	Up About a minute	0.0.0.0:80->80/tcp, :::80->80/tcp	meusite
98d9bcbfedb1	mysql:5.7	"docker-entrypoint.s..."	9 minutes ago	Up 9 minutes	3306/tcp, 33060/tcp	banco

#### 4.2.3 Acesso ao ambiente WordPress que receberá as aplicações, através do IP 192.168.98.4




### 4.3 Criação de containers utilizando o Ansible

Antes de prosseguir com o processo, reverteremos para um Snapshot feito após o término da instalação do Docker:



4.3.1 Criação do arquivo hosts dentro do diretório /wordpress-docker  
comando: touch hosts

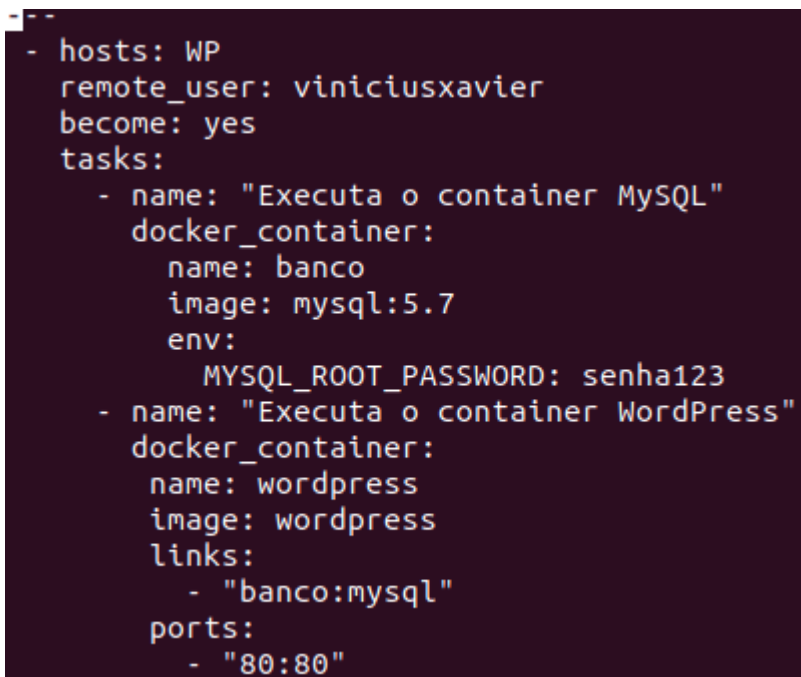




```
[WP]
127.0.0.1
```

4.3.2 Criação do arquivo `playbook.yml` dentro do diretório `/wordpress-docker` que conterá o container MySQL e container WordPress.

comando: `touch playbook.yml`



```
--
- hosts: WP
  remote_user: viniciusxavier
  become: yes
  tasks:
    - name: "Executa o container MySQL"
      docker_container:
        name: banco
        image: mysql:5.7
        env:
          MYSQL_ROOT_PASSWORD: senha123
    - name: "Executa o container WordPress"
      docker_container:
        name: wordpress
        image: wordpress
        links:
          - "banco:mysql"
        ports:
          - "80:80"
```

4.3.3 Instalação do gerenciador de bibliotecas Python3-PIP dentro do diretório `/wordpress-docker`

comando: `sudo apt install python3-pip`

```

viniciusxavier@ubuntu:~/wordpress-docker$ sudo apt install python3-pip
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  apache2-bin apache2-data apache2-utils libatol1 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libcgf-fast-perl libcgf-pm-perl libcgf-perl
  libhtml-template-perl liblvm1 liblua5.2-0 libmecab2 linux-headers-5.11.0-34-generic linux-hwe-5.11.0-34 linux-image-5.11.0-34-generic
  linux-modules-5.11.0-34-generic linux-modules-extra-5.11.0-34-generic mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0
  mysql-server-8.0 mysql-server-core-8.0
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev fakeroot g++ g++-9 gcc gcc-9 libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libasan5 libatomic1 libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0 libexpat1-dev libfakeroot libgcc-9-dev libitm1
  liblsan0 libpython3-dev libpython3.8-dev libquadmath0 libstdc++-9-dev libtsan0 libubsan1 linux-libc-dev make manpages-dev python-pip-whl python3-dev
  python3-setuptools python3-wheel python3.8-dev zlib1g-dev
Suggested packages:
  binutils-doc debian-keyring g++-multilib g++-9-multilib gcc-9-doc gcc-multilib autoconf automake libtool flex bison gcc-doc gcc-9-multilib gcc-9-locales glibc-doc
  libstdc++-9-doc make-doc python3-setuptools-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev fakeroot g++ g++-9 gcc gcc-9 libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libasan5 libatomic1 libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0 libexpat1-dev libfakeroot libgcc-9-dev libitm1
  liblsan0 libpython3-dev libpython3.8-dev libquadmath0 libstdc++-9-dev libtsan0 libubsan1 linux-libc-dev make manpages-dev python-pip-whl python3-dev python3-pip
  python3-setuptools python3-wheel python3.8-dev zlib1g-dev
0 upgraded, 42 newly installed, 0 to remove and 39 not upgraded.
Need to get 41.5 MB of archives.
After this operation, 184 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 binutils-common amd64 2.34-6ubuntu1.1 [207 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libbinutils amd64 2.34-6ubuntu1.1 [475 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libctf-nobfd0 amd64 2.34-6ubuntu1.1 [47.1 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libctf0 amd64 2.34-6ubuntu1.1 [46.6 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 binutils-x86-64-linux-gnu amd64 2.34-6ubuntu1.1 [1,613 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 binutils amd64 2.34-6ubuntu1.1 [3,380 B]
Get:7 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libc-dev-bin amd64 2.31-0ubuntu9.2 [71.8 kB]
Get:8 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 linux-libc-dev amd64 5.4.0-88.99 [1,116 kB]
Get:9 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libcrypt-dev amd64 1:4.4.10-10ubuntu4 [104 kB]
Get:10 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libc6-dev amd64 2.31-0ubuntu9.2 [2,520 kB]
Get:11 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libitm1 amd64 10.3.0-1ubuntu1-20.04 [26.2 kB]
Get:12 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libatomic1 amd64 10.3.0-1ubuntu1-20.04 [9,284 B]
Get:13 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libasan5 amd64 9.3.0-17ubuntu1-20.04 [394 kB]
Get:14 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 liblsan0 amd64 10.3.0-1ubuntu1-20.04 [835 kB]
Get:15 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libtsan0 amd64 10.3.0-1ubuntu1-20.04 [2,009 kB]
Get:16 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libubsan1 amd64 10.3.0-1ubuntu1-20.04 [784 kB]
Get:17 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libquadmath0 amd64 10.3.0-1ubuntu1-20.04 [146 kB]
Get:18 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libgcc-9-dev amd64 9.3.0-17ubuntu1-20.04 [2,360 kB]
Get:19 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 gcc-9 amd64 9.3.0-17ubuntu1-20.04 [8,741 kB]

```

#### 4.3.4 Instalação da biblioteca pysphere dentro do diretório /wordpress-docker:

comando: `sudo pip3 install docker`

```

viniciusxavier@ubuntu:~/wordpress-docker$ sudo pip3 install docker
Collecting docker
  Downloading docker-5.0.2-py2.py3-none-any.whl (145 kB)
    |#####| 145 kB 3.3 MB/s
Collecting websocket-client>=0.32.0
  Downloading websocket-client-1.2.1-py2.py3-none-any.whl (52 kB)
    |#####| 52 kB 1.1 MB/s
Requirement already satisfied: requests!=2.18.0,>=2.14.2 in /usr/lib/python3/dist-packages (from docker) (2.22.0)
Installing collected packages: websocket-client, docker
Successfully installed docker-5.0.2 websocket-client-1.2.1

```

#### 4.3.5 Teste do playbook criado

comando: `ansible-playbook -i hosts playbook.yml`

```

viniciusxavier@ubuntu:~/wordpress-docker$ ansible-playbook -i hosts playbook.yml

PLAY [WP] *****

TASK [Gathering Facts] *****
ok: [127.0.0.1]

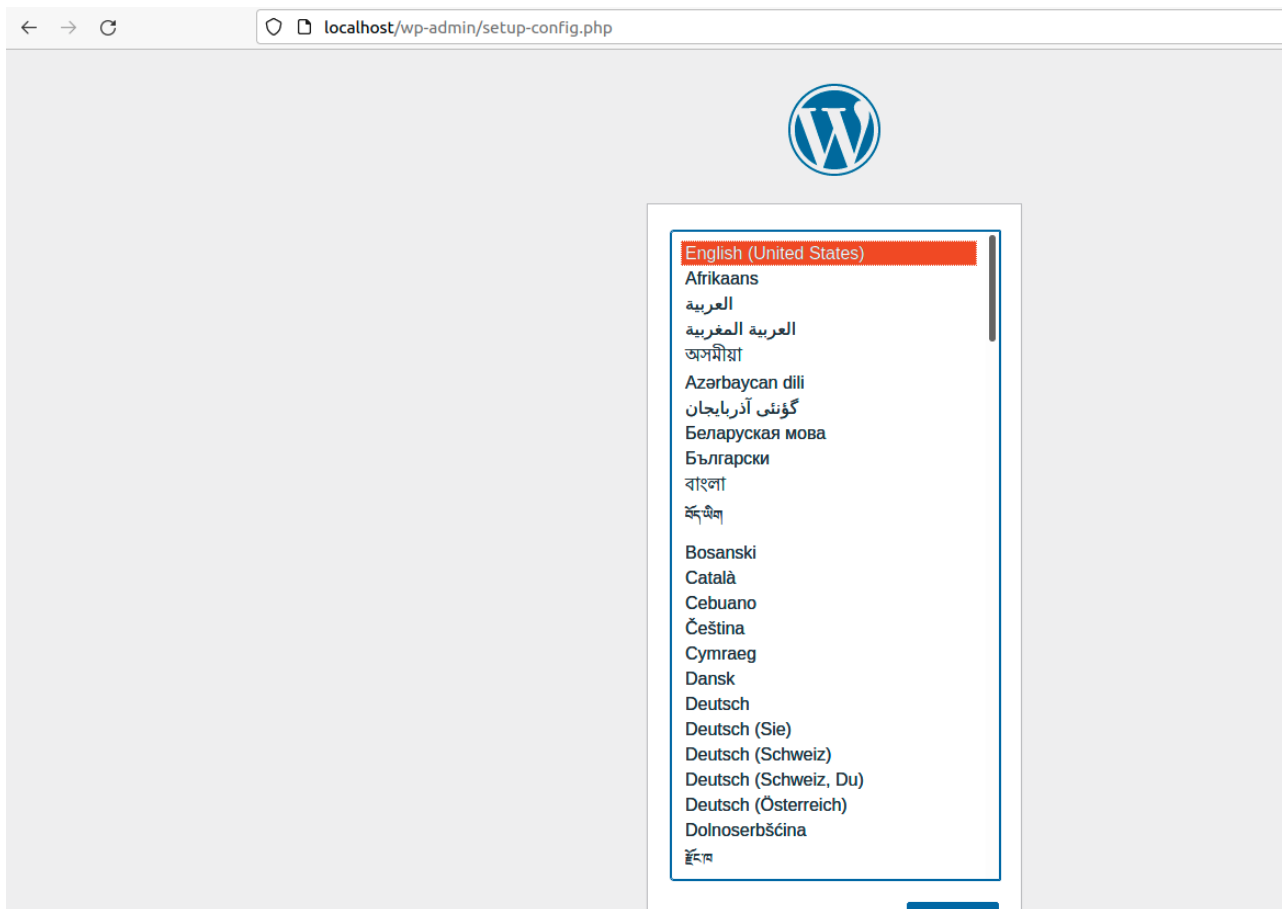
TASK [Executa o container MySQL] *****
changed: [127.0.0.1]

TASK [Executa o container WordPress] *****
changed: [127.0.0.1]

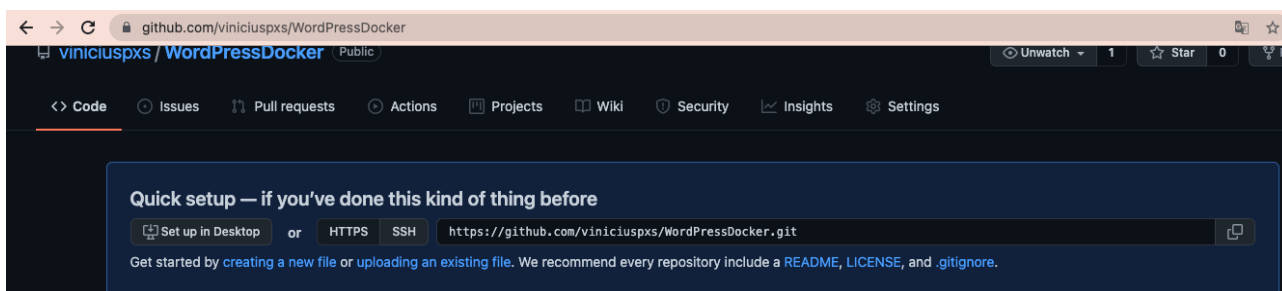
PLAY RECAP *****
127.0.0.1 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

```

#### 4.3.6 Acesso ao ambiente WordPress que receberá as aplicações, através do localhost:



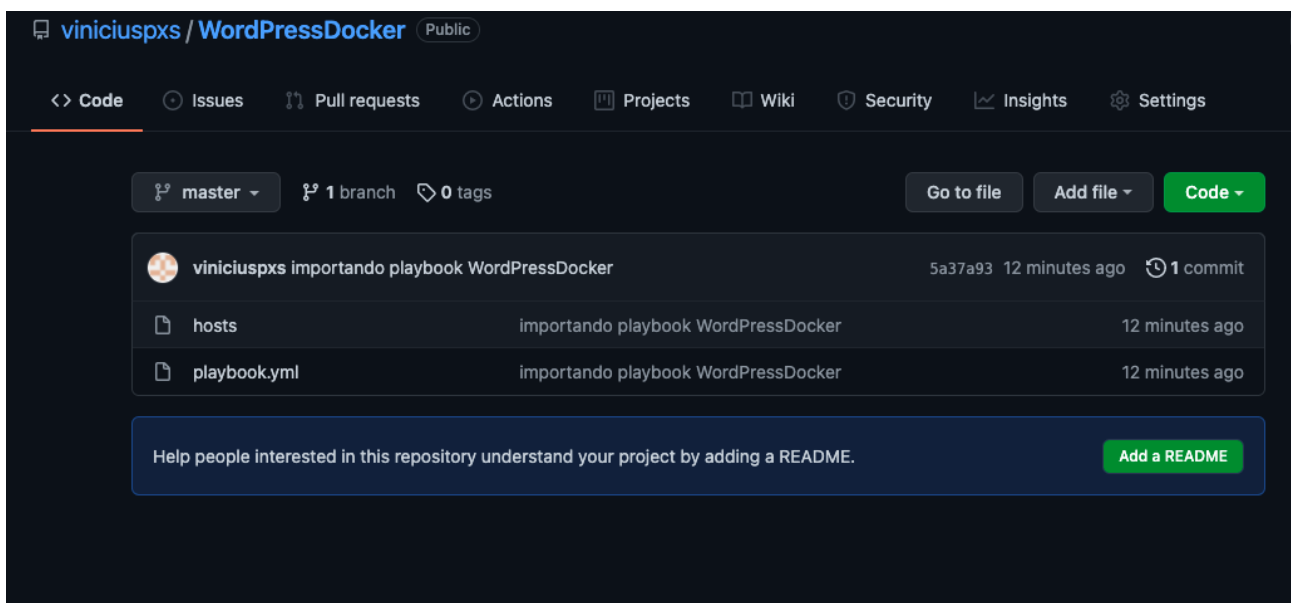
#### 4.3.7 Criação de um novo repositório no GitHub



URL: <https://github.com/viniciuspxs/WordPressDocker.git>

#### 4.3.8 Exportar playbook para repositório Github

```
viniciusxavier@ubuntu:~/wordpress-docker$ git push -u origin master
Username for 'https://github.com': viniciuspxs
Password for 'https://viniciuspxs@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 515 bytes | 257.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/viniciuspxs/WordPressDocker.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```




#### 4.3.9 URL para acesso via internet

<https://github.com/viniciuspxs/WordPressDocker>

## 5. Cronograma de Atividades


A fim de apresentar um melhor controle sobre as atividades, aumentar o grau de transparência e viabilizar a participação do cliente possibilitando-o a realizar comentários nas tarefas a serem realizadas, entregues e possíveis bloqueios, a ferramenta Trello foi escolhida como plataforma para a gestão de cronograma e tarefas.

### 5.1 Cronograma atual

Quadro ▾ **Projeto de Bloco - Arquitetura de aplicações** ☆ Área de trabalho do Trello Visível à Área de trabalho  Convidar

Power-up do Calendário ⚡ Autom


**Backlog** ...

+ Adicionar um cartão 

**Pendente** ...

**A fazer**




- Código-fonte WordPress (a ser realizado por um terceiro)


+ Adicionar um cartão 

**Bloqueio** ...

**Prioridade**


Fatores limitantes



  0/1 

+ Adicionar um cartão 


**Concluído** ...


**Entregue**

Implementação da Infraestrutura Virtual (17/11/21) 


 


**Entregue**

- Playbook Ansible (previsto para 19/11/21) 





**Entregue**


- Código-fonte Docker (previsto para 19/11/21) 



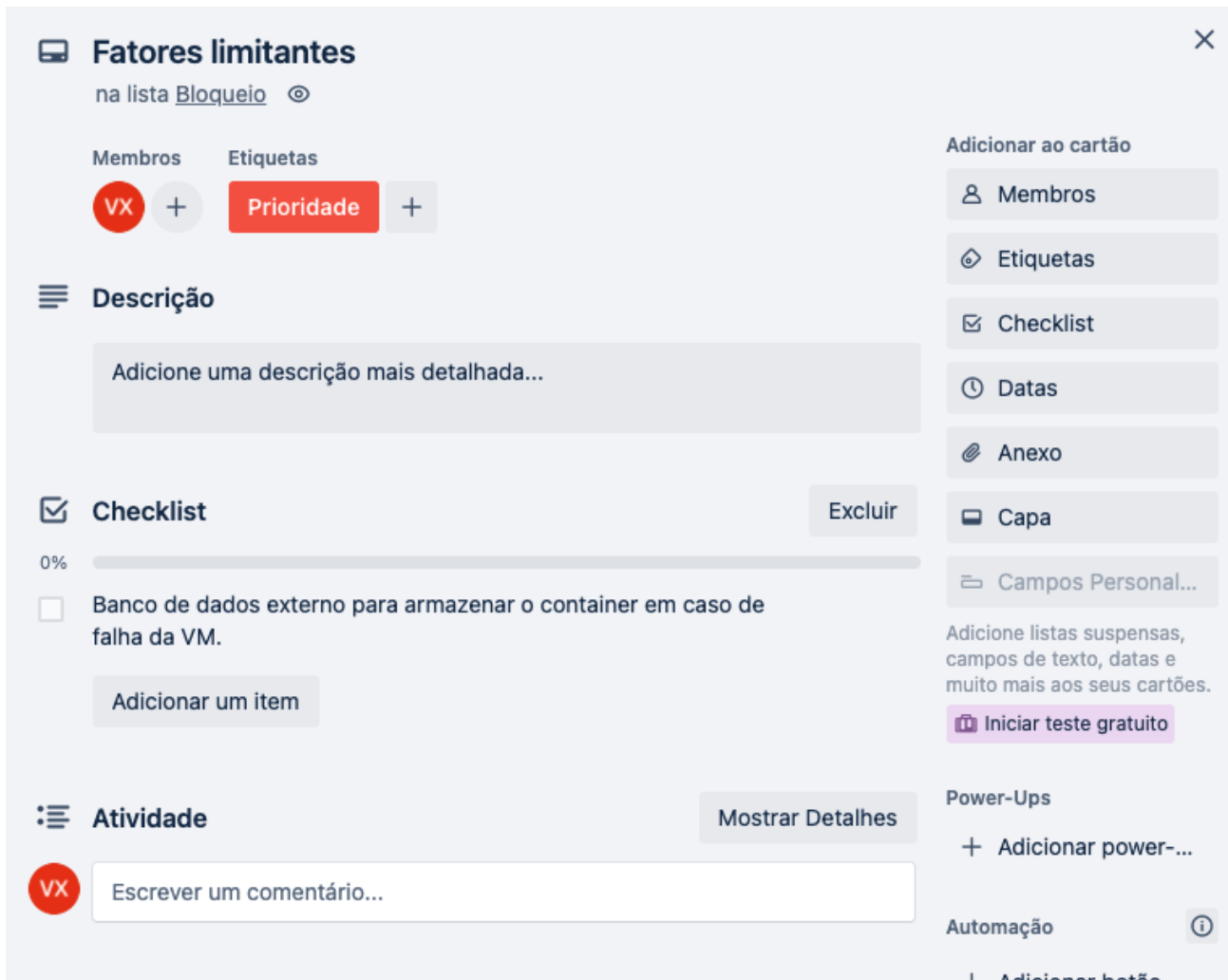
**Entregue**

- Entrega do Projeto ao cliente (06/12/21) 



+ Adicionar um cartão 

### 5.1.1 Possível bloqueio



### 5.1.2 URL para acessar o cronograma

<https://trello.com/b/xfwQclJl/projeto-de-bloco-arquitetura-de-aplica%C3%A7%C3%B5es>

## 6. Conclusão

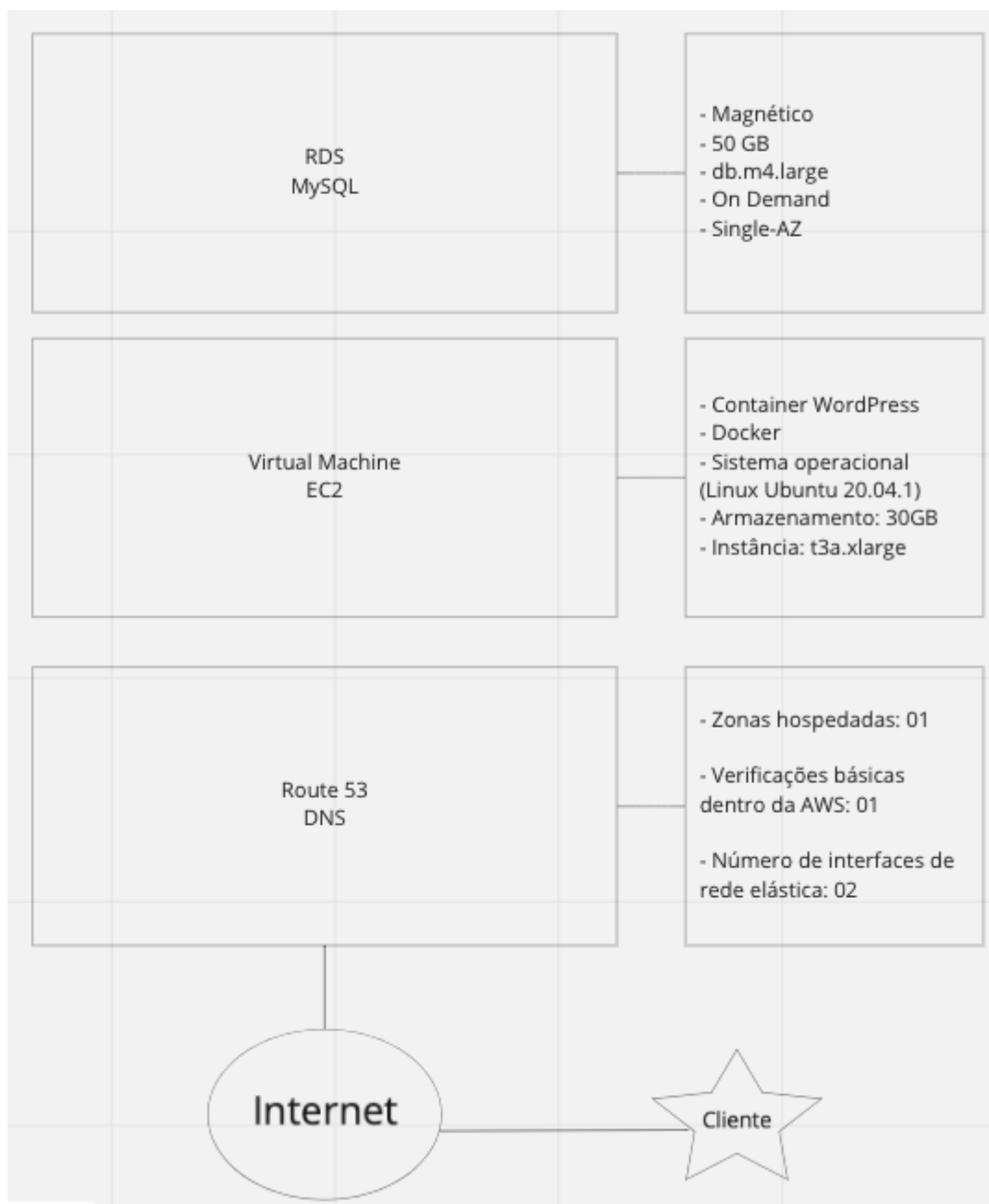
O prazo final estabelecido pelo cliente para a entrega do Projeto foi o suficiente para realizar a implantação da infraestrutura e o desenvolvimento inicial da aplicação. A implantação do código-fonte para aplicação foi acordado junto ao cliente que será desenvolvido e implementado na infraestrutura por um desenvolvedor que será contratado em um momento futuro pelo próprio

cliente.

Os recursos planejados para a infraestrutura física, nos quais já estavam disponíveis na máquina pessoal do cliente, foram suficientes para o desenvolvimento da infraestrutura lógica e para o suporte necessário do desenvolvimento da aplicação desejada, um blog Wordpress de cunho gastronômico.

No entanto, como o cliente não descarta a possibilidade de incrementar o blog caso esse tenha êxito e se por um motivo de força maior o próprio cliente não siga a carreira jurídica na Holanda, foi calculado uma estimativa de preço de uma solução nuvem pública AWS, na qual é mais robusta e mais escalável pois apresenta maior disponibilidade de recursos em comparação à solução atual desenvolvida no Projeto.

Tendo em vista a possibilidade de escalabilidade e a implantação da infraestrutura virtual, na qual conteria a instalação da aplicação, foi apresentado ao cliente a seguinte solução em nuvem pública AWS (To be):



Estimativa de custos baseada nos serviços a serem utilizados na AWS:

### Amazon Route 53

Zonas hospedadas (1), Verificações básicas dentro da AWS (1), Número de interfaces de rede elástica (2)

Valor Mensal: 183,00 USD



### **RDS para MySQL**

Armazenamento para cada instância do RDS (Magnético (geração anterior)), Quantidade de armazenamento (50 GB), Quantidade (1), Tipo de instância (db.m4.large), Utilização (somente Sob demanda) (100 %Utilized/Month), Opção de implantação (Single-AZ), Modelo de definição de preço (OnDemand)

Valor Mensal: 132,75 USD

### **EC2**

Sistema operacional (Linux Ubuntu LTS 20.04.1), Quantidade (1), Estratégia de definição de preço (EC2 Instance Savings Plans 1 ano Sem pagamento adiantado), Quantidade de armazenamento (30 GB), Tipo de instância (t3a.xlarge)

Valor Mensal: 124,07 USD

### **Resumo da estimativa**

Custo inicial	0,00 USD
Custo mensal	439,82 USD
Custo total (12 meses)	5.277,84 USD

Fonte: AWS Pricing Calculator

Portanto, dado os critérios estabelecidos pelo cliente, a disponibilidade de recursos da infraestrutura e recursos financeiros disponíveis do próprio no atual contexto, a solução atual cumpre todas as necessidades momentâneas. Contudo, vale ressaltar que dependendo do código-fonte desenvolvido para aplicação Wordpress pelo desenvolvedor que será contrato em segundo momento, a necessidade de transferência da aplicação criada no Docker para a solução de nuvem pública apresentada pode ser requisitada tanto pelo cliente quanto pelo desenvolvedor antes do previsto. Por fim, o projeto final foi armazenado no repositório Github e pode ser acessado através da seguinte URL: <https://github.com/viniciuspxs/projetodeblocoarq>.

## Referência bibliográfica

<https://calculator.aws/#/estimate>

<https://docs.ansible.com/>

<https://github.com/viniciuspxs/projetodeblocoarq>

<https://government.nl/latest/news/2018/07/06/a-fast-fixed-internet-connection-for-all-dutch-citizens-by-2023>

[https://miro.com/app/board/o9J\\_kmZgl1U=](https://miro.com/app/board/o9J_kmZgl1U=/)

<https://statista.com/statistics/377747/household-internet-access-in-the-netherlands/>

<https://trello.com/b/xfwQclJl/projeto-de-bloco-arquitetura-de-aplica%C3%A7%C3%B5es>