



SÃO  
PAULO  
TECH  
SCHOOL

# Sistemas Operacionais

LAB

Docker

Container + MySQL

Professor Marcio Santana

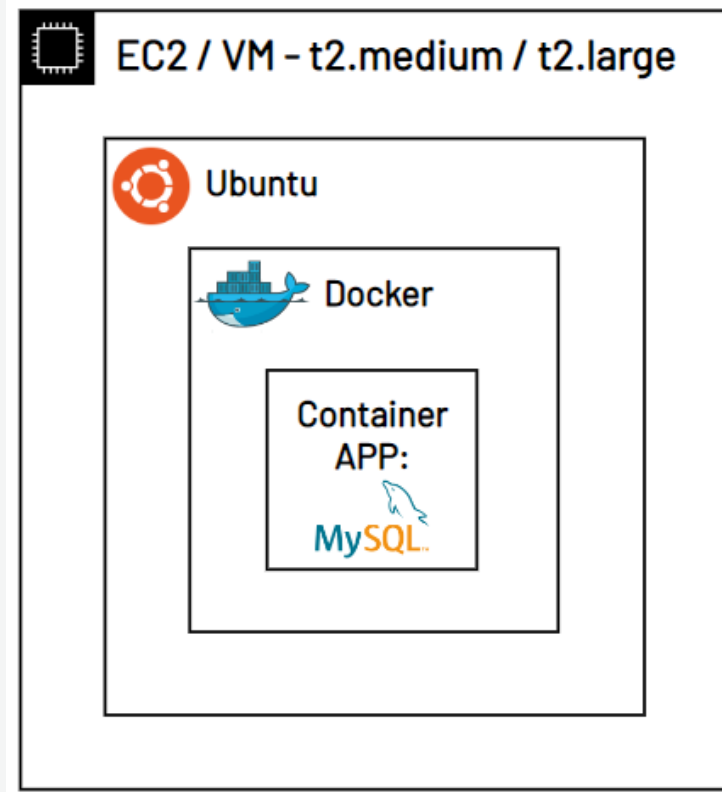
[marcio.santana@sptech.school](mailto:marcio.santana@sptech.school)

# Tópicos da Aula

1. Atualizar pacotes do sistema e instalar Docker Engine
2. Criar container com aplicação do MySQL
3. Acessar container de MySQL para utilizar aplicação
4. Encerrando atividades
5. **Desafios**

# Objetivo

Utilizando uma EC2 pela conexão SSH, vamos instalar e configurar o software Docker, com o objetivo de criar um container de MySQL.



# **1. Atualizar pacotes do sistema, instalar e configurando Docker Engine**

# Preparando EC2



1. Acesse o terminal da EC2\*, pode ser via protocolo SSH.
2. Atualize os pacotes do sistema:

```
sudo apt update && sudo apt upgrade -y
```

```
ubuntu@ip-172-31-24-71:~$ sudo apt update && sudo apt upgrade -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
```

3. Instale o Docker na EC2:

```
sudo apt install docker.io
```

```
ubuntu@ip-172-31-24-71:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
```

\*Se a EC2 for nova, não esqueça de fazer o setup inicial

# Configurando Docker



4. Verifique se o Docker foi instalado corretamente:

```
docker --version
```

```
ubuntu@ip-172-31-24-71:~$ docker --version
Docker version 20.10.21, build 20.10.21-0ubuntu1~22.04.3
```

5. Ative o serviço do Docker no sistema operacional, para que ele fique disponível para uso.

```
sudo systemctl start docker
```

```
ubuntu@ip-172-31-24-71:~$ sudo systemctl start docker
ubuntu@ip-172-31-24-71:~$
```

6. Habilite o serviço do Docker para ser iniciado junto ao sistema operacional:

```
sudo systemctl enable docker
```

```
ubuntu@ip-172-31-24-71:~$ sudo systemctl enable docker
ubuntu@ip-172-31-24-71:~$
```

**systemctl** é um utilitário de linha de comando usado para gerenciar serviços

## **2. Criar container com aplicação do MySQL**



# Criando Container



7. Para criar um container Docker, será necessário uma imagem da aplicação que será instalada no container para execução.

Faça pull da imagem do MySQL 5.7 que fica armazenada no repositório remoto Docker Hub

```
sudo docker pull mysql:5.7
```

```
ubuntu@ip-172-31-24-71:~$ sudo docker pull mysql:5.7
5.7: Pulling from library/mysql
```

8. Confirme o download da imagem:

```
sudo docker images
```

```
ubuntu@ip-172-31-24-71:~$ sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mysql	5.7	dd6675b5cfea	2 weeks ago	569MB

# Criando Container



9. Agora com a imagem, vamos criar o container de MySQL.

O comando abaixo cria o container e configurar a aplicação do MySQL dentro do container:

```
sudo docker run -d -p 3306:3306 --name ContainerBD -e "MYSQL_DATABASE=banco1" -e "MYSQL_ROOT_PASSWORD=urubu100" mysql:5.7
```

```
ubuntu@ip-172-31-24-71:~$ sudo docker run -d -p 3306:3306 --name ContainerBD -e "MYSQL_DATABASE=banco1" -e "MYSQL_ROOT_PASSWORD=urubu100" mysql:5.7
633d135a332c27acb87713a10b3d68fac309772a1638a8c94618e9c73b0046c7
```

10. Confirme se o container foi criado com sucesso:

```
sudo docker ps -a
```

```
ubuntu@ip-172-31-24-71:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                                                                 NAMES
633d135a332c   mysql:5.7 "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:3306->3306/tcp, :::3306->3306/tcp, 33060/tcp ContainerBD
```

Observe a coluna "**STATUS**", o container deve está "**UP**".

**Caso contrário, a criação falhou....!**

-p porta de rede que a aplicação dentro do container vai responder  
-d executa o container em background  
-e variável ambiente

# Criando Container



11. Monitore o consumo de recursos que o seu container está consumindo.

```
sudo docker stats ContainerBD
```

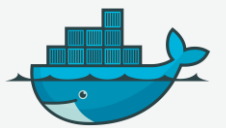
```
ubuntu@ip-172-31-24-71:~$ sudo docker stats ContainerBD
```

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
bc039f704ebf	ContainerBD	0.04%	194.3MiB / 7.76GiB	2.45%	31.9kB / 876B	0B / 291MB	27

Para terminar o monitoramento, pressione “ctrl + c”

### **3. Acessar container de MySQL para utilizar aplicação**

# Acessando Container



12. Para manipular a aplicação que está sendo executada dentro do container, precisamos primeiro acessar o container:

```
sudo docker exec -it ContainerBD bash
```

-i permite interagir com o container  
-t associa um pseudo terminal ao container

```
ubuntu@ip-172-31-24-71:~$ sudo docker exec -it ContainerBD bash
bash-4.2# |
```

Nesse momento, iniciamos o modo iterativo, executando um simples bash dentro do container para executar alguns comandos.

13. Agora podemos iniciar a configuração do MySQL:

```
mysql -u root -p
```

```
bash-4.2# mysql -u root -p
Enter password:
welcome to the MySQL monitor.
```

Password, foi configurado no momento da criação do container = urubu100

# Acessando Container



14. Nesse momento, podemos utilizar os comando de MySQL:

**SHOW DATABASES;**

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| banco1 |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> |
```

## **4. Encerrando atividades**

# Saindo Container



15. Para fechar a conexão com MySQL, execute:

`exit`

```
mysql> exit
Bye
bash-4.2#
```

16. Continuamos no bash dentro do container, para sair e voltar ao shell-bash do Linux, execute:

```
bash-4.2# exit
exit
ubuntu@ip-172-31-24-71:~$
```



# Pausando Container



17. Para fechar a conexão com MySQL, execute:

```
sudo docker stop ContainerBD
```

```
ubuntu@ip-172-31-24-71:~$ sudo docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
bc039f704ebf	mysql:5.7	"docker-entrypoint.s..."	26 minutes ago	Up 26 minutes	0.0.0.0:3306->3306/tcp, :::3306->3306/tcp, 33060/tcp	ContainerBD

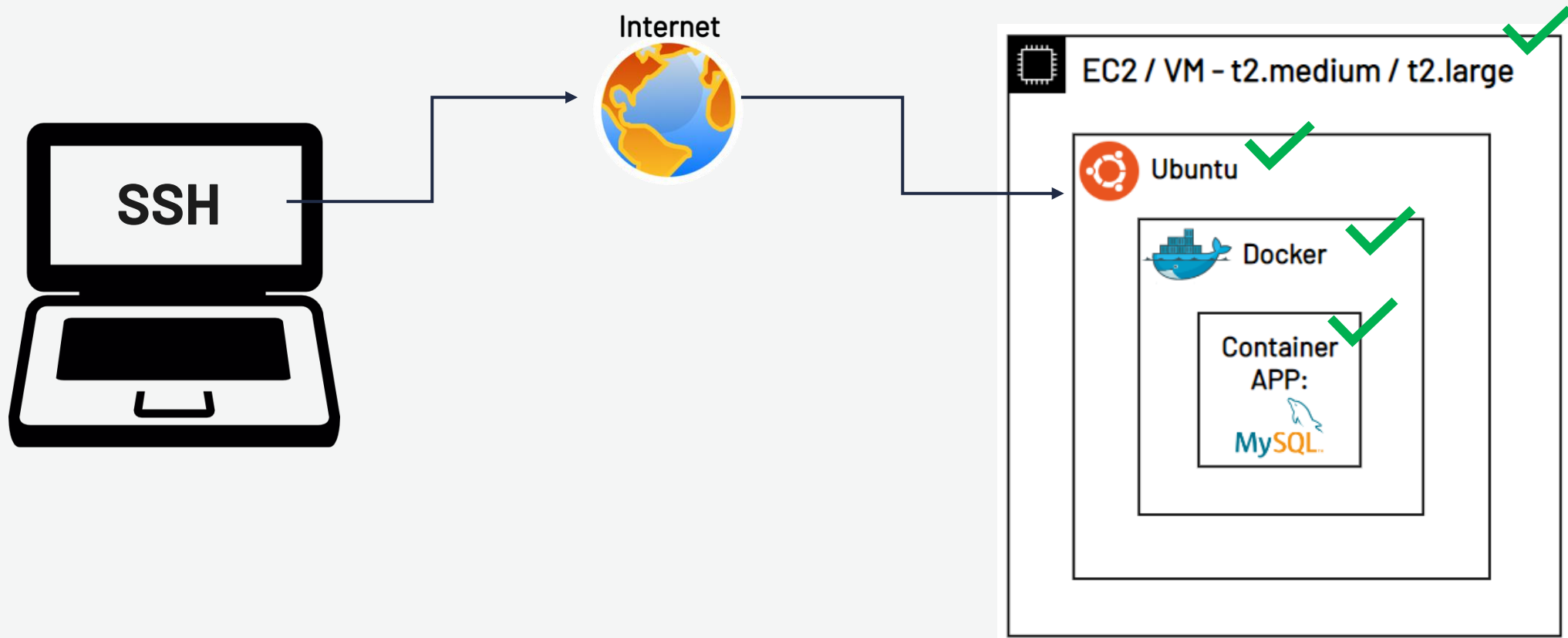
Para manipular o container, você pode utilizar o ID ou Nome.

18. Caso precise iniciar o container novamente, basta executar:

```
sudo docker start ContainerBD
```

# Objetivo

Utilizando uma EC2 pela conexão SSH, vamos instalar e configurar o software Docker, com o objetivo de criar um container de MySQL.

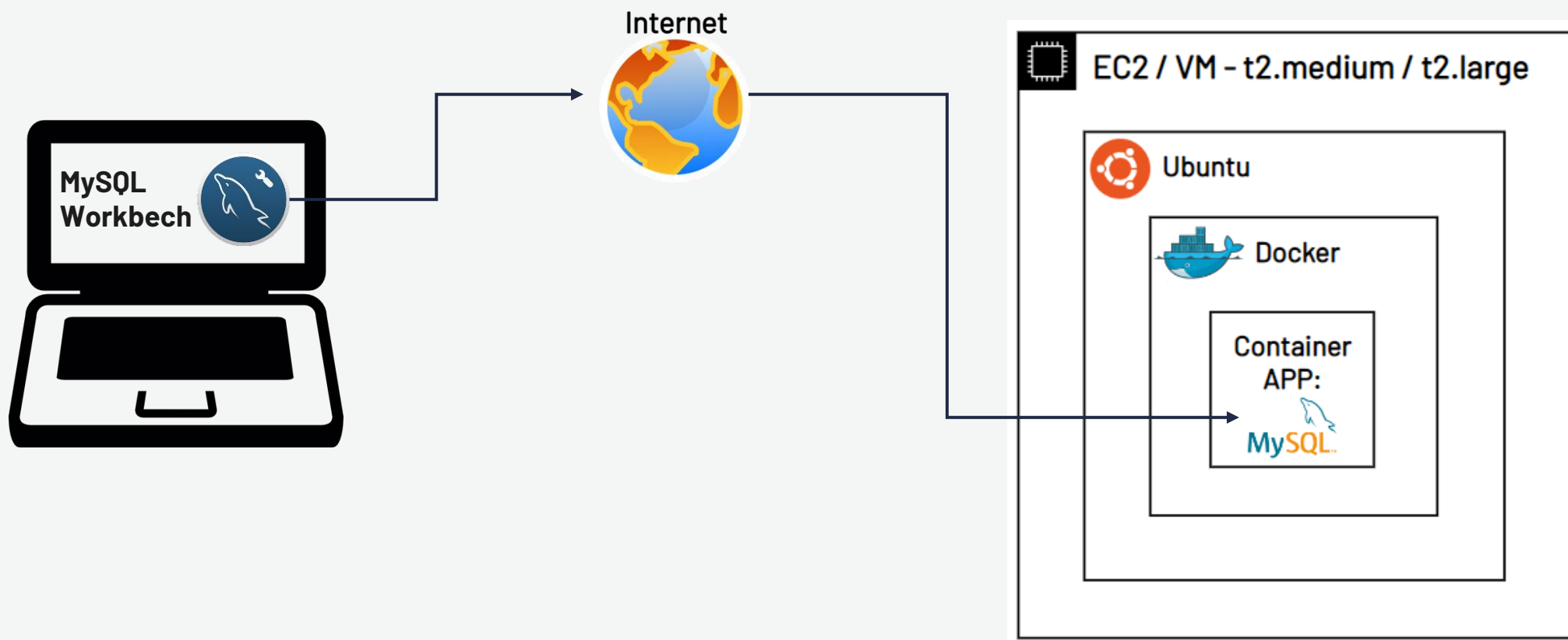


# 5. Desafios

The background is a dark, monochromatic abstract composition. It features a series of thin, white, wavy lines that flow from the left side towards the right, creating a sense of movement and depth. Scattered throughout the image are numerous small, bright white dots and star-like particles, some of which appear to be trailing behind the flowing lines, giving the impression of a cosmic or digital landscape.

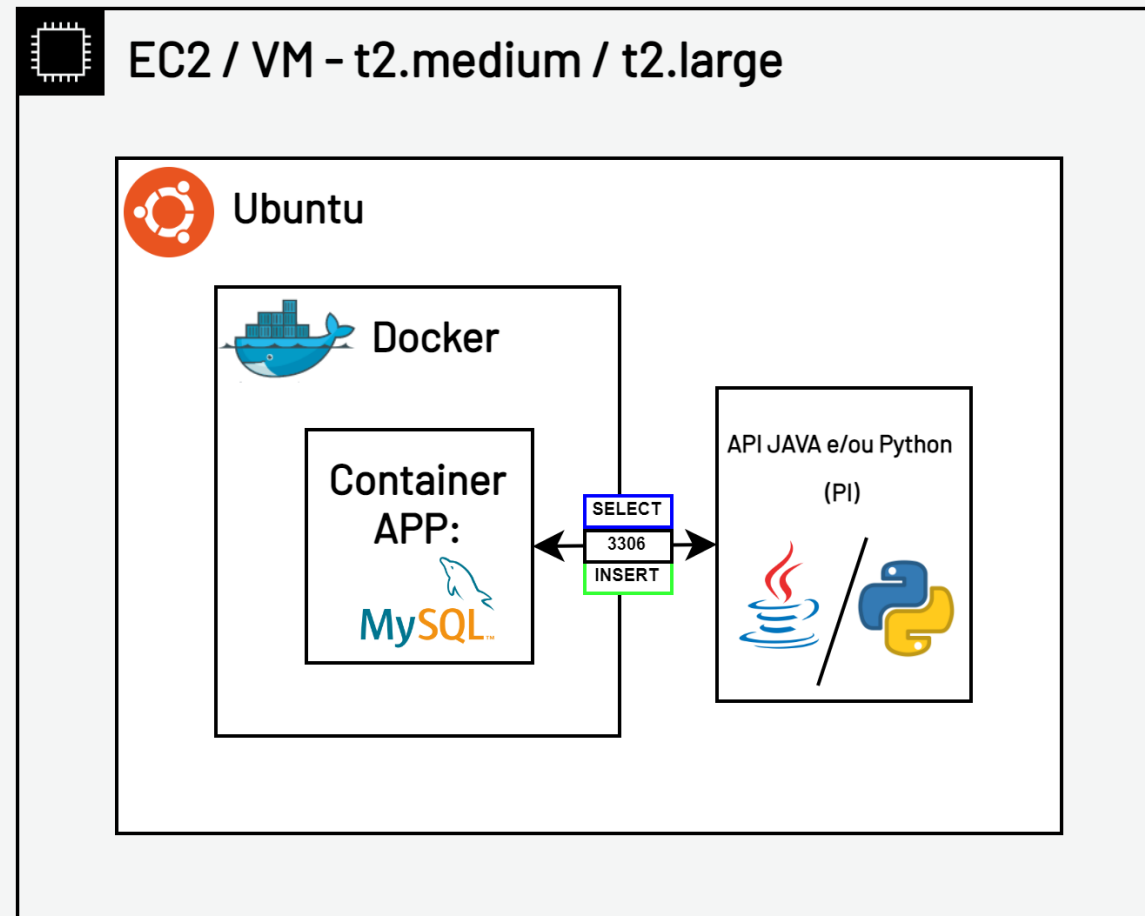
# DESAFIO 1

Para facilitar a gestão do seu mysql, conecte o aplicativo “Workbech” à sua aplicação MySQL que está sendo executada no container!



# DESAFIO 2

Realize a conexão da aplicação desenvolvida com o banco de dados MySQL  
**Local na EC2!**





**Agradeço**  
a sua atenção!

**Marcio Santana**

marcio.santana@sptech.school

SÃO  
PAULO  
TECH  
SCHOOL