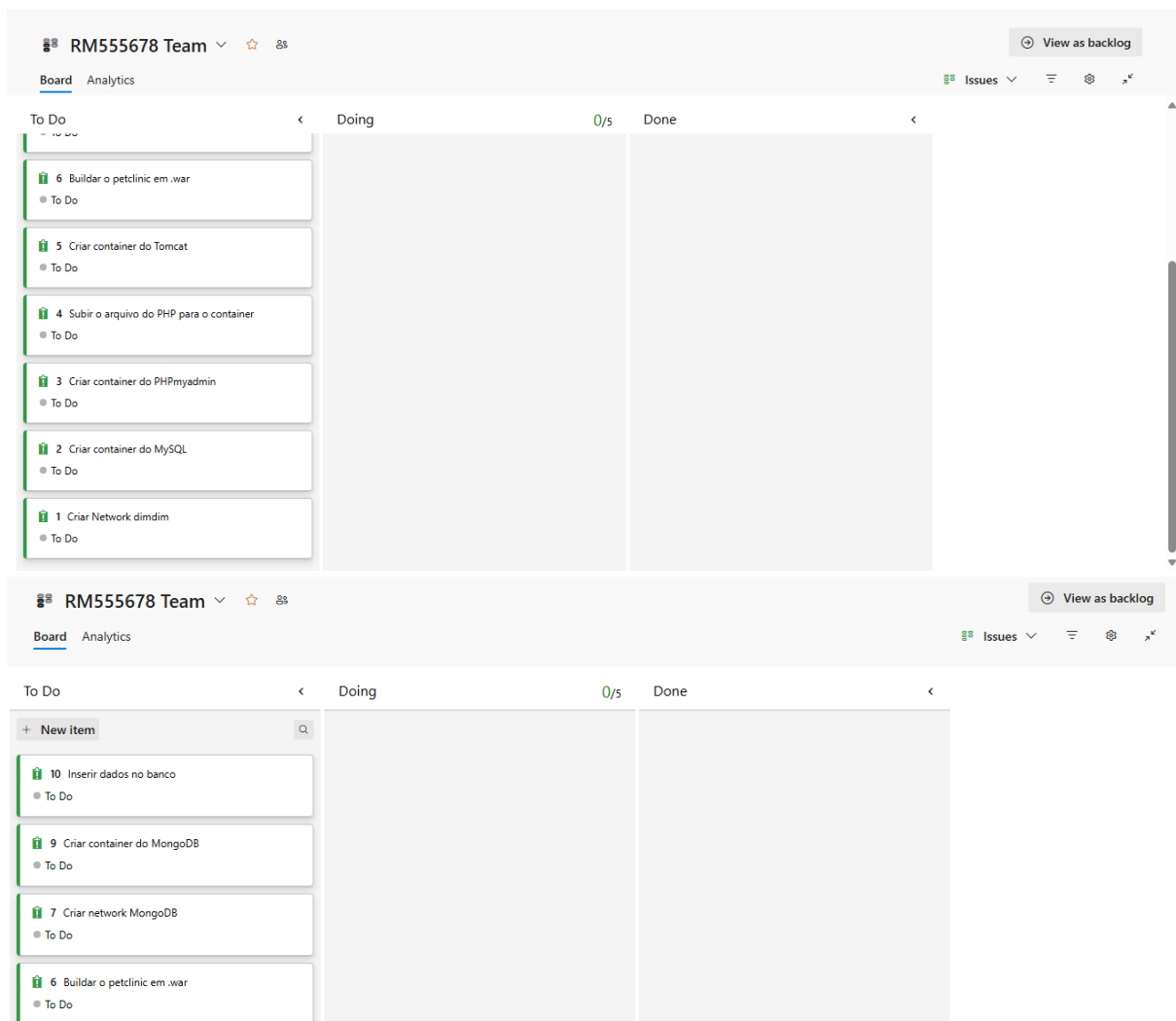


# Checkpoint 2

Nome: João Victor Michaeli de Bem

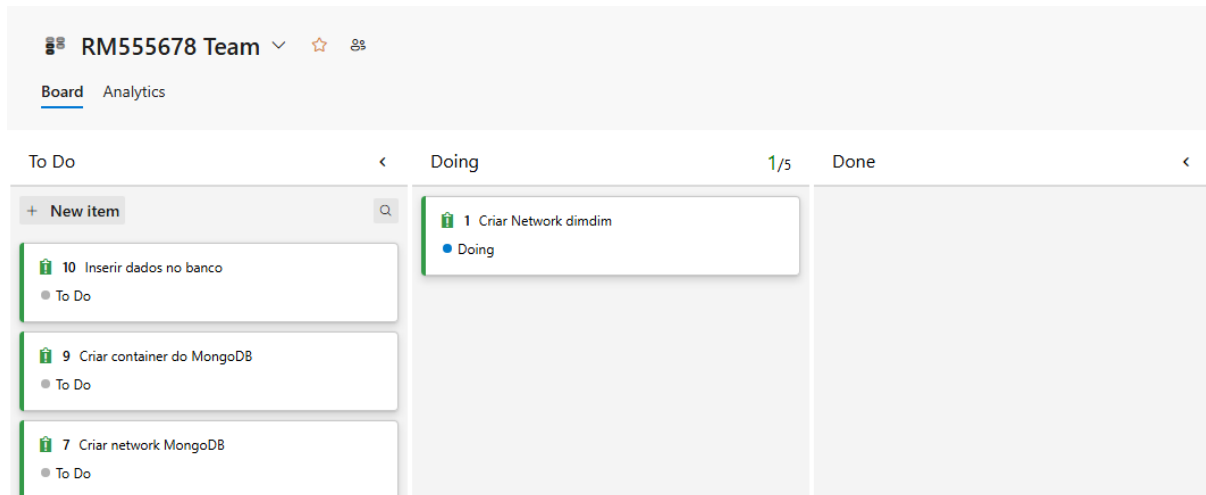
Rm: 555678

Iniciamos o projeto criando um diagrama de arquitetura no Azure Board, que nos ajuda a organizar e visualizar todas as etapas do desenvolvimento.



Criamos uma rede personalizada no Docker, que permitirá que os containers se comuniquem entre si de forma isolada e organizada.

## Azure board

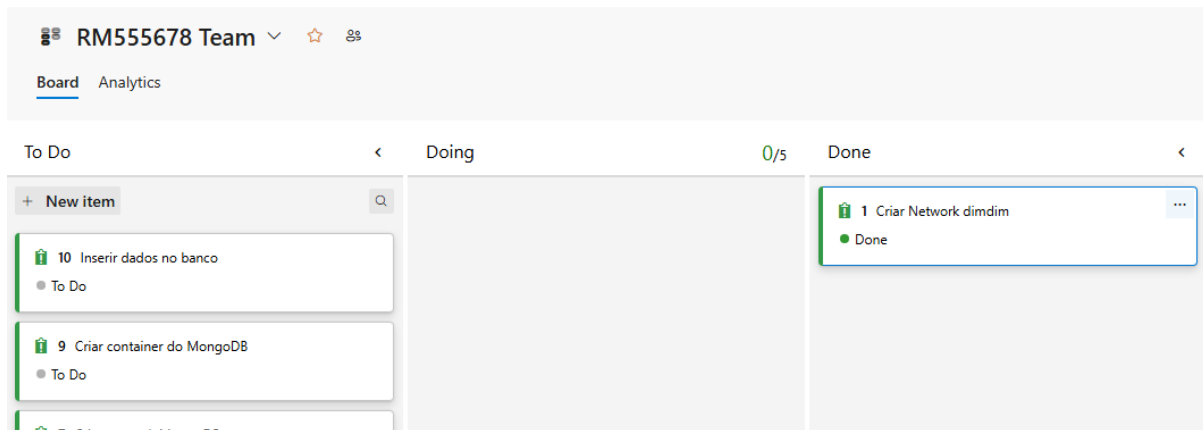


Neste passo, iniciamos a criação dos containers no Docker, configurando cada um conforme as necessidades da aplicação.

```
Terminal
PS C:\Users\João> docker network create dimdim-network
e07c51ccbd80ebe4fdf9bb58661e6ba5661a1535d25176e12aa722d79061d530
```

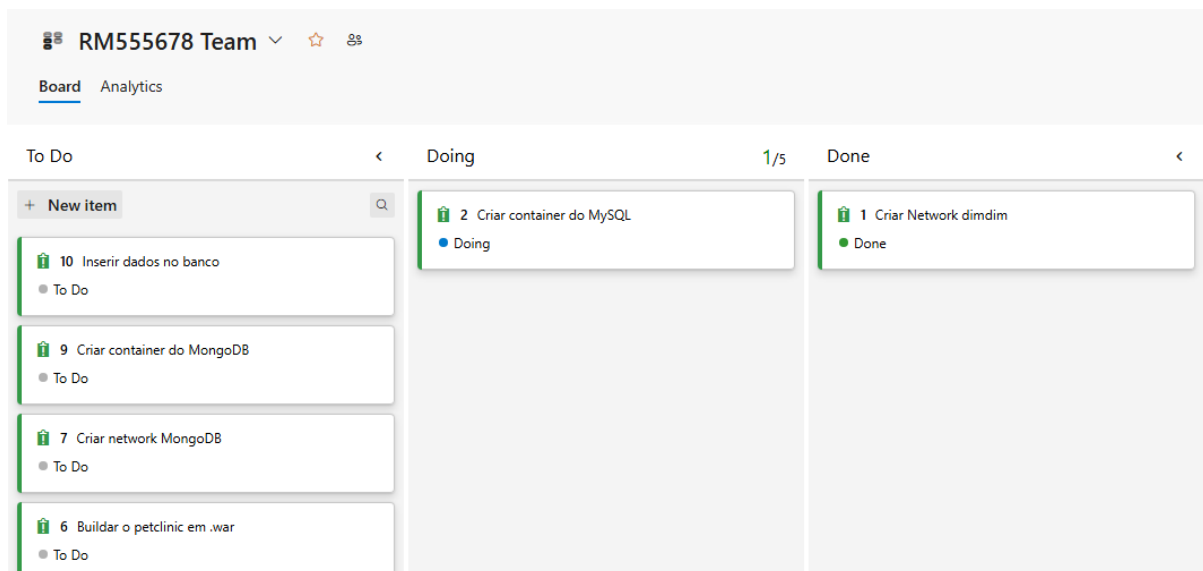
```
PS C:\Users\João> docker network ls
NETWORK ID        NAME                DRIVER              SCOPE
f5adcc6aa07d     bridge              bridge              local
e07c51ccbd80     dimdim-network      bridge              local
8f9ff56a2349     host                host                local
911a90e40ace     none                null                local
```

## Atualização board



Aqui realizamos o deploy do container MySQL, que será responsável por armazenar os dados da aplicação.

### Atualização board



### Docker

```
PS C:\Users\João> docker run -d `
>> --name dindim-mysql `
>> --network dindim-network `
>> -e MYSQL_ROOT_PASSWORD=rootpass `
>> -e MYSQL_DATABASE=dindimdb `
>> -e MYSQL_USER=dimuser `
>> -e MYSQL_PASSWORD=dimpass `
>> mysql/mysql-server:latest
```

```
>> mysql/mysql-server:latest
Unable to find image 'mysql/mysql-server:latest' locally
latest: Pulling from mysql/mysql-server
b6b576315b62: Pull complete
6a4a3ef82cdc: Pull complete
5518b09b1089: Pull complete
c7668948e14a: Pull complete
abe8d2406c31: Pull complete
349b52643cc3: Pull complete
c7e93886e496: Pull complete
Digest: sha256:d6c8301b7834c5b9c2b733b10b7e630f441af7bc917c74dba379f24eeeb6a313
Status: Downloaded newer image for mysql/mysql-server:latest
d4ec22ca25685902b8e2a88e6f4233458e0ace399cc69486613f73f1aac56036
```

Verificamos os logs do container MySQL para garantir que ele foi iniciado corretamente e está funcionando como esperado.

Terminal

```
PS C:\Users\João> docker logs -f dindin-mysql
[Entrypoint] MySQL Docker Image 8.0.32-1.2.11-server
[Entrypoint] Initializing database
2025-05-09T00:00:37.010932Z 0 [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and will be removed in a future release. Please use SET GLOBAL host_cache_size=0 instead.
2025-05-09T00:00:37.011121Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.32) initializing of server in progress as process 17
2025-05-09T00:00:37.049335Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2025-05-09T00:00:37.738814Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2025-05-09T00:00:39.929900Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.
[Entrypoint] Database initialized
2025-05-09T00:00:44.055447Z 0 [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and will be removed in a future release. Please use SET GLOBAL host_cache_size=0 instead.
2025-05-09T00:00:44.060067Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.32) starting as process 60
2025-05-09T00:00:44.105861Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2025-05-09T00:00:44.414524Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2025-05-09T00:00:44.868766Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2025-05-09T00:00:44.868863Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
2025-05-09T00:00:44.893863Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqlx.sock
2025-05-09T00:00:44.894177Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.32' socket: '/var/lib/mysql/mysql.sock' port: 0 MySQL Community Server - GPL.
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/tzdata.zi' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it.

[Entrypoint] ignoring /docker-entrypoint-initdb.d/*

2025-05-09T00:00:49.145143Z 14 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version : 8.0.32).
2025-05-09T00:00:50.618666Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.32) MySQL Community Server - GPL.
[Entrypoint] Server shut down
```

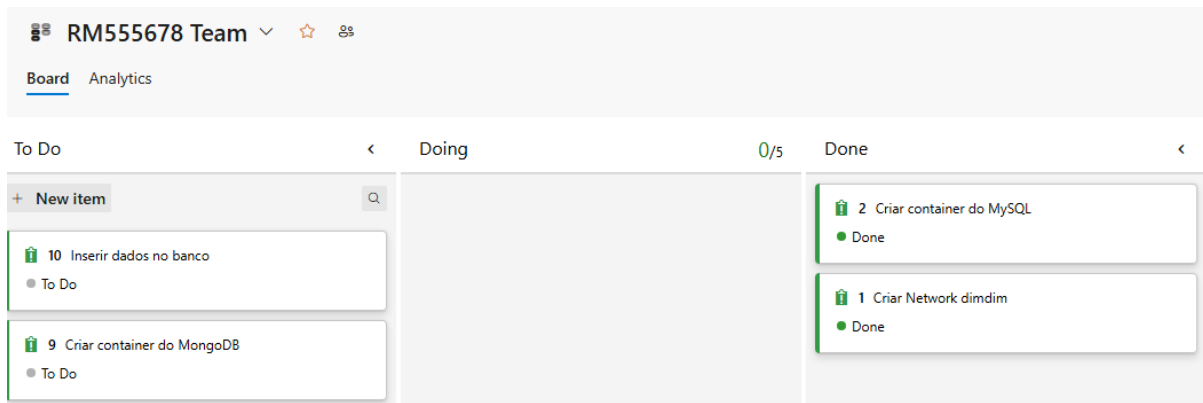
Terminal

```
ons are now supported for this channel.
2025-05-09T00:00:44.893863Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqlx.sock
2025-05-09T00:00:44.894177Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.32' socket: '/var/lib/mysql/mysql.sock' port: 0 MySQL Community Server - GPL.
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/tzdata.zi' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it.

[Entrypoint] ignoring /docker-entrypoint-initdb.d/*

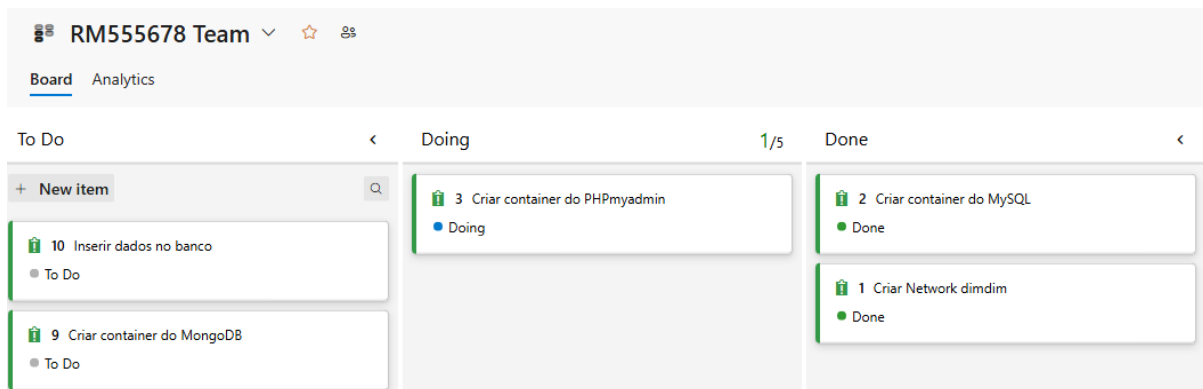
2025-05-09T00:00:49.145143Z 14 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version : 8.0.32).
2025-05-09T00:00:50.618666Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.32) MySQL Community Server - GPL.
[Entrypoint] Server shut down
```

Atualização board



Agora iniciamos o container do PHPMyAdmin, que fornece uma interface gráfica para gerenciar o banco de dados MySQL de forma mais intuitiva.

## Board



## Docker

```

Terminal

PS C:\Users\João> docker run -d `
>> --name dimdim-phpmyadmin `
>> --network dimdim-network `
>> -p 8080:80 `
>> -e PMA_HOST=dimdim-mysql `
>> -e PMA_USER=dimuser `
>> -e PMA_PASSWORD=dimpass `
>> phpmyadmin
  
```

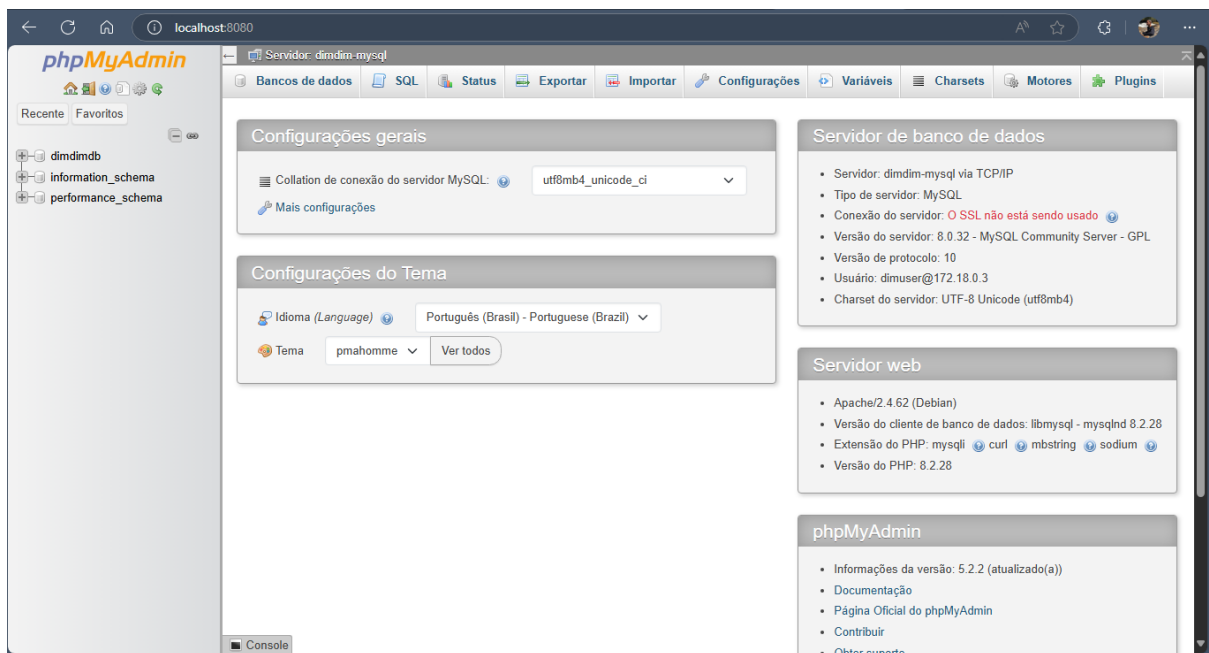
```

Digest: sha256:68d7f9dc247b1b10c3525a244e0688979437f062d253edd5d7bff2019df2a063
Status: Downloaded newer image for phpmyadmin:latest
0a2176649ec6b17a76e176780316504454b100d1e4ab33a96ece4f68075c85ce
  
```

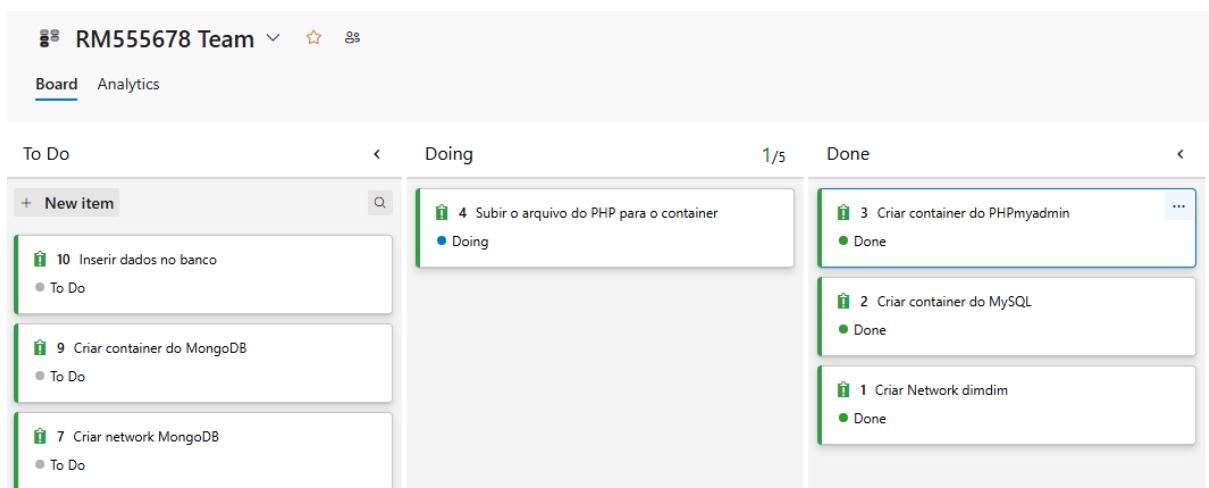
Analisamos os logs do PHPMyAdmin para garantir que ele também foi iniciado sem erros e está pronto para uso.

```
PS C:\Users\João> docker logs -f dimdim-phpmyadmin
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.18.0.3. Set the 'ServerName' directive globally to suppress this message
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.18.0.3. Set the 'ServerName' directive globally to suppress this message
[Fri May 09 00:10:13.651390 2025] [mpm_prefork:notice] [pid 1:tid 1] AH00163: Apache/2.4.62 (Debian) PHP/8.2.28 configured - resuming normal operations
[Fri May 09 00:10:13.651474 2025] [core:notice] [pid 1:tid 1] AH00094: Command line: 'apache2 -D FOREGROUND'
```

Acessamos o PHPMyAdmin através do navegador utilizando o endereço localhost, confirmando que está funcionando corretamente.



Azure board



Selecionamos o arquivo PHP que será executado no ambiente Docker, indicando o caminho correto onde ele está salvo.

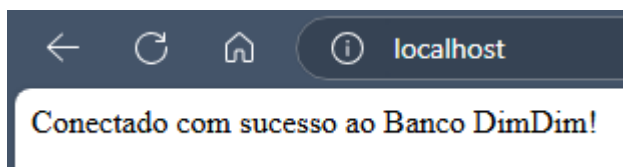
```
index.php X
C: > Users > João > Documents > FIAP > DevOps > dimdim-backend > index.php
1  <?php
2
3  $mysqli = new mysqli("dimdim-mysql", "dimuser", "dimpass", "dimdimdb");
4
5  if ($mysqli->connect_error) {
6      die("Erro na conexão: " . $mysqli->connect_error);
7  }
8
9  echo "Conectado com sucesso ao Banco DimDim!";
10
11  ?>
12
```

Neste passo, colocamos o arquivo PHP dentro do container Docker, para que ele possa ser executado no servidor local.

```
Terminal
PS C:\Users\João> cd .\Documents\FIAP\DevOps\dimdim-backend\
PS C:\Users\João\Documents\FIAP\DevOps\dimdim-backend> docker run -d `
>> --name dimdim-php-app `
>> --network dimdim-network `
>> -p 80:80 `
>> -v "${PWD}:/var/www/html" `
>> php:8.2-apache `
>> bash -c "apt update && \
>> apt install -y libpng-dev libjpeg-dev libonig-dev libxml2-dev libzip-dev unzip zip libmcrypt-dev && \
>> docker-php-ext-install mysqli && \
>> docker-php-ext-enable mysqli && \
>> apachectl -D FOREGROUND"

Unable to find image 'php:8.2-apache' locally
8.2-apache: Pulling from library/php
Digest: sha256:e2408924aac97ed8dce0ba54adff30443fe7a940a87d7b0d083b36941d8aa431
Status: Downloaded newer image for php:8.2-apache
37cb4cc6d6cdb5d4a1f1404f1599c1b8fb738535f5dca1d0b9c1c383bc0bd058
```

Acessamos o navegador na porta 80 (localhost:80) para visualizar o resultado do código PHP rodando dentro do container.



Aqui listamos todas as imagens que foram instaladas no Docker até o momento, mostrando os serviços disponíveis localmente.

**Images** [Give feedback](#)

View and manage your local and Docker Hub images. [Learn more](#)

**Local** Docker Hub repositories

1.14 GB / 1.43 GB in use 3 images Last refresh: 2 seconds ago

Search

<input type="checkbox"/>	Name	Tag	Image ID	Created	Size	Actions
<input type="checkbox"/>	mysql/mysql-server	latest	d6c8301b7834	2 years ago	698.21 MB	<a href="#">▶</a> <a href="#">⋮</a> <a href="#">🗑️</a>
<input type="checkbox"/>	phpmyadmin	latest	68d7f9dc247b	3 months ago	813.22 MB	<a href="#">▶</a> <a href="#">⋮</a> <a href="#">🗑️</a>
<input type="checkbox"/>	php	8.2-apache	e2408924aac9	2 months ago	708.83 MB	<a href="#">▶</a> <a href="#">⋮</a> <a href="#">🗑️</a>

Verificamos os containers que estão em execução no ambiente local para garantir que todos os serviços estão ativos.

**Containers** [Give feedback](#)

View all your running containers and applications. [Learn more](#)

Container CPU usage 1.34% / 800% (8 CPUs available) Container memory usage 508.97MB / 7.21GB [Show charts](#)

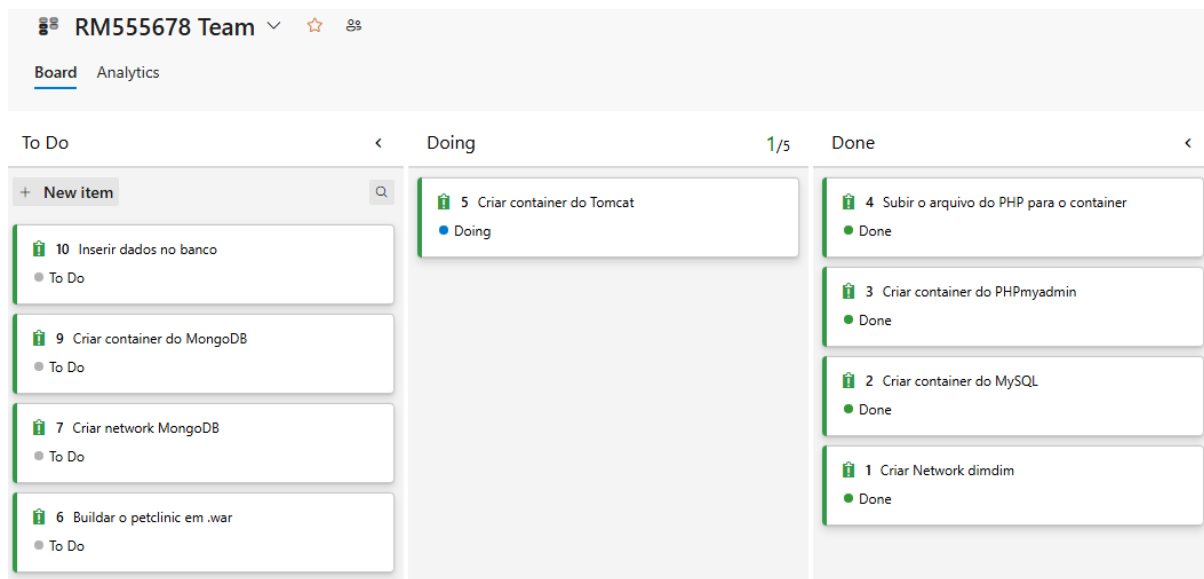
Search ☐ Only show running containers

<input type="checkbox"/>	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	dimdim-mysql	d4ec22ca2568	<a href="#">mysql/mysql-se</a>		1.32%	23 minutes ago	<a href="#">▶</a> <a href="#">⋮</a> <a href="#">🗑️</a>
<input type="checkbox"/>	dimdim-phpmya	0a2176649ec6	<a href="#">phpmyadmin</a>	<a href="#">8080:80</a>	0.01%	14 minutes ago	<a href="#">▶</a> <a href="#">⋮</a> <a href="#">🗑️</a>
<input type="checkbox"/>	dimdim-php-app	37cb4cc6d6cd	<a href="#">php:8.2-apache</a>	<a href="#">80:80</a>	0.01%	4 minutes ago	<a href="#">▶</a> <a href="#">⋮</a> <a href="#">🗑️</a>

A seguir, realizamos o deploy de dois serviços importantes: Tomcat (para aplicações Java) e MongoDB (banco de dados NoSQL).



Começamos com a instalação e configuração do Tomcat, que será usado para hospedar uma aplicação web em Java.



Fizemos o download da imagem oficial do Tomcat 10 a partir do Docker Hub, para utilizarmos como base na criação do container.

```
Terminal
PS C:\Users\João> docker pull tomcat:10
10: Pulling from library/tomcat
d0cc8efa6ea7: Pull complete
4f4fb700ef54: Pull complete
f526e30db5f3: Pull complete
0622fac788ed: Pull complete
b0eec1e8367a: Pull complete
0f6b7f7c07f4: Pull complete
d4bb17fdf011: Pull complete
4223556ed03c: Pull complete
Digest: sha256:76edc30a01a4b99dd42e6ec1f842f76e600c52773d91f979485ba24527f4436f
Status: Downloaded newer image for tomcat:10
docker.io/library/tomcat:10
```

Mapeamos as portas e volumes do container Tomcat para garantir que ele possa ser acessado externamente e armazenar os dados corretamente.

```
PS C:\Users\João> docker run -d --name tomcat10 -p 8080:8080 tomcat:10
a0599725e6c95fb3ca97a7aff61046c1e11ee514456eb7bf7965a2fe85c23587
PS C:\Users\João> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
a0599725e6c9	tomcat:10	"catalina.sh run"	53 seconds ago	Up 52 seconds	0.0.0.0:8080->8080/tcp	tomcat10

Compilamos o arquivo WAR da aplicação Java, que será implantado no Tomcat para execução.

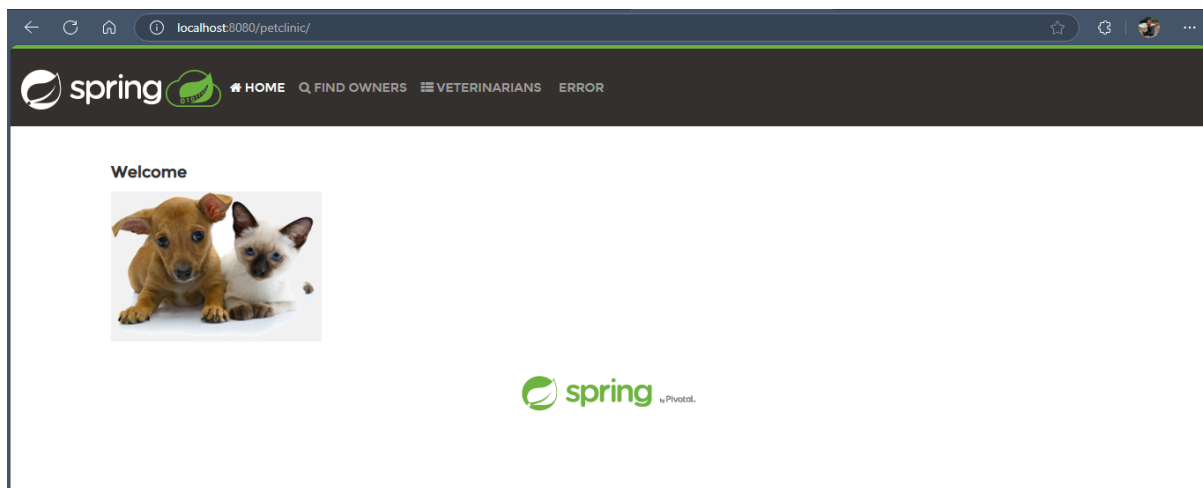
```
Terminal

PS C:\Users\João\Documents\FIAP\DevOps\warCp2\spring-framework-petclinic> cd .\target\
PS C:\Users\João\Documents\FIAP\DevOps\warCp2\spring-framework-petclinic\target> docker cp petclinic.war tomcat10:/usr/local/tomcat/webapps/
Successfully copied 45.9MB to tomcat10:/usr/local/tomcat/webapps/

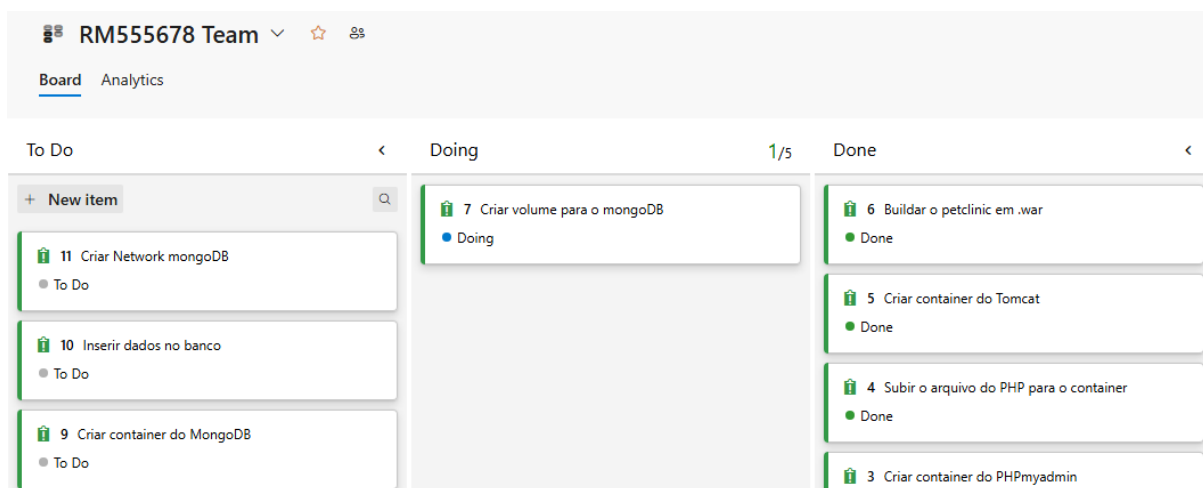
--release 17 is recommended instead of -source 17 -target 17 because it sets the location of system modules automatically
[INFO] --- surefire:3.5.0:test (default-test) @ spring-framework-petclinic ---
[INFO] Tests are skipped.
[INFO] --- jacoco:0.8.12:report (report) @ spring-framework-petclinic ---
[INFO] Skipping JaCoCo execution due to missing execution data file.
[INFO] --- war:3.4.0:war (default-war) @ spring-framework-petclinic ---
[INFO] Packaging webapp
[INFO] Assembling webapp [spring-framework-petclinic] in [C:\Users\João\Documents\FIAP\DevOps\warCp2\spring-framework-petclinic\target\petclinic]
[INFO] Processing war project
[INFO] Copying webapp resources [C:\Users\João\Documents\FIAP\DevOps\warCp2\spring-framework-petclinic\src\main\webapp]
[INFO] Building war: C:\Users\João\Documents\FIAP\DevOps\warCp2\spring-framework-petclinic\target\petclinic.war
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 7.083 s
[INFO] Finished at: 2025-05-09T10:19:55-03:00
[INFO] -----

João@DESKTOP-2D2NBFJ MINGW64 ~/Documents/FIAP/DevOps/warCp2/spring-framework-petclinic (main)
```

Após o deploy, acessamos a aplicação com sucesso no caminho <http://localhost:8080/petclinic>, validando que tudo funcionou corretamente.



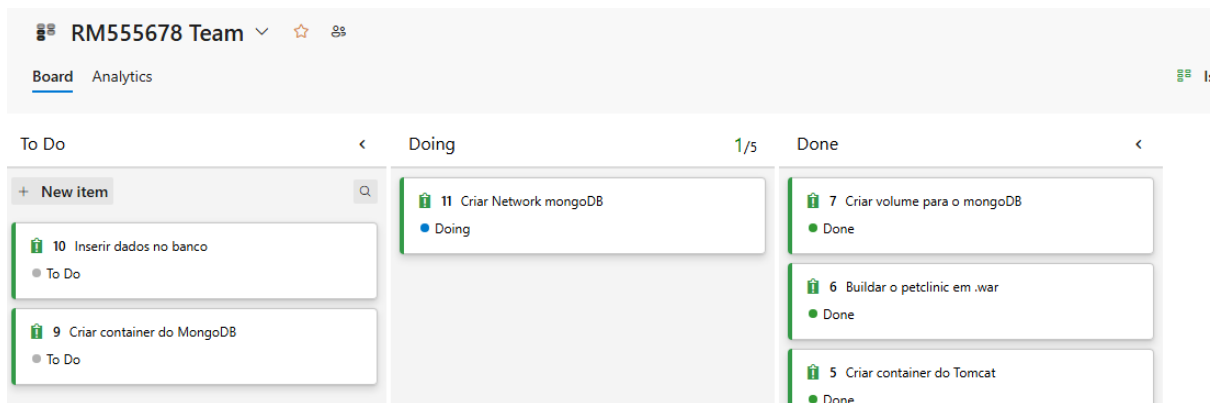
## Atualização board



Agora passamos para o MongoDB, um banco de dados NoSQL ideal para trabalhar com documentos em formato JSON.

Criamos um volume no Docker para garantir que os dados armazenados no MongoDB não sejam perdidos mesmo que o container seja reiniciado.

## Board

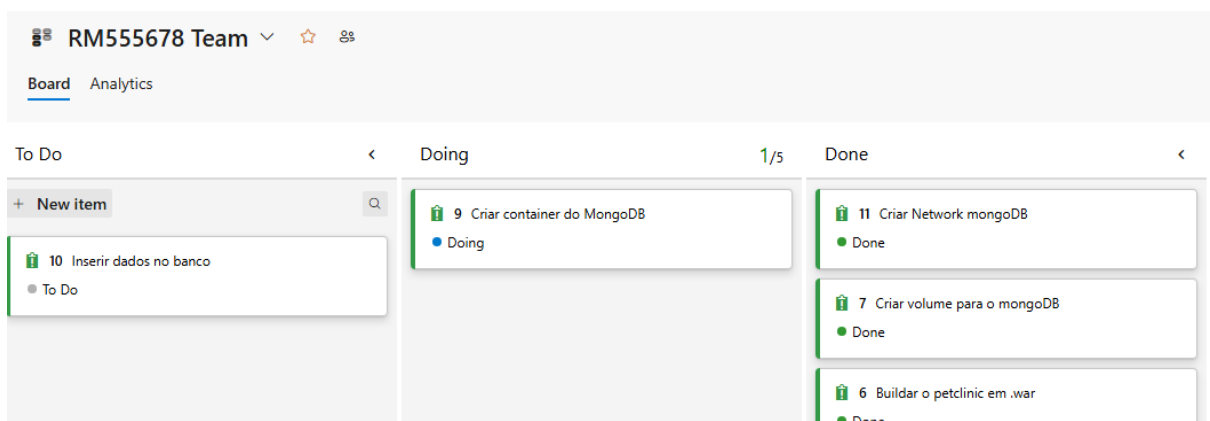


Montamos uma nova rede no Docker para isolar o MongoDB e permitir sua comunicação segura com outros containers, como o backend da aplicação.

```
Terminal

PS C:\Users\João> docker network create networkmongo
aaffdd76fc20dab22fcadf9dddc886314af302d4ff8604f2486da111ade75d96
PS C:\Users\João> docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a77c4f60b93e        bridge              bridge              local
e07c51ccbd80        dindim-network      bridge              local
8f9ff56a2349        host                host                local
aaffdd76fc20        networkmongo         bridge              local
911a90e40ace        none                null                local
```

## Board



Inicializamos o container MongoDB, configurando as portas e o volume criado anteriormente.

```
Terminal

PS C:\Users\João> docker run -d --network networkmongo -h mongo --name mongo -p 27017:27017 -v volumemongo:/data/db mongo
Unable to find image 'mongo:latest' locally
latest: Pulling from library/mongo
54db8e3d8732: Pull complete
ca015e545994: Pull complete
Digest: sha256:2e018e386e891d2e4239aca6035fb7701dac51b72891247ecd2f95ff8a167859
Status: Downloaded newer image for mongo:latest
3f02cc93fb9fdcc701d0e49c4abd39517f14bbd041c3211a2223b9892086437e

PS C:\Users\João> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS
3f02cc93fb9f   mongo    "docker-entrypoint.s..." 39 seconds ago Up 37 seconds 0.0.0.0:27017->27017/tcp
mongo
```

Entramos no terminal do container MongoDB para executar comandos diretamente e gerenciar o banco de dados.

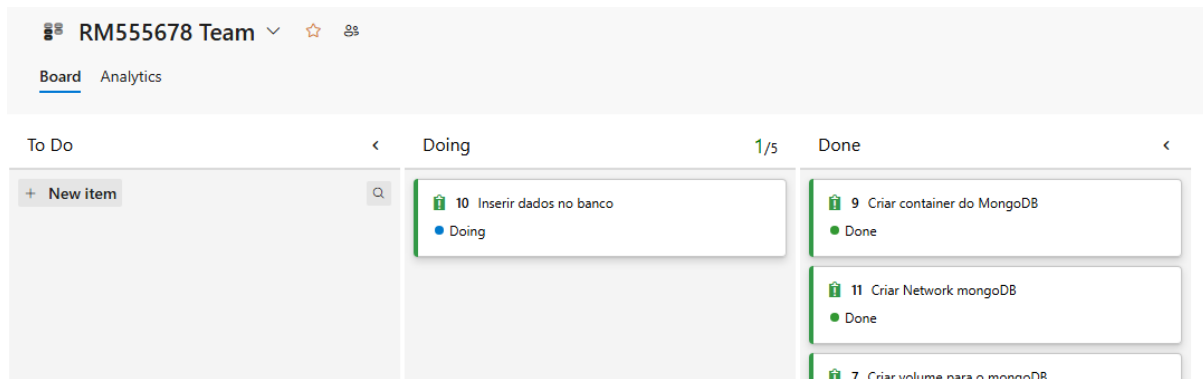
```
PS C:\Users\João> docker container exec -it mongo bash
root@mongo:/# mongosh
Current Mongosh Log ID: 681e2319ed98aed426d861df
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.0
Using MongoDB:      8.0.9
Using Mongosh:       2.5.0

For mongosh info see: https://www.mongodb.com/docs/mongosh-shell/

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

-----
The server generated these startup warnings when booting
2025-05-09T15:42:42.279+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2025-05-09T15:42:43.684+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
2025-05-09T15:42:43.685+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2025-05-09T15:42:43.685+00:00: We suggest setting the contents of sysfsFile to 0.
2025-05-09T15:42:43.685+00:00: vm.max_map_count is too low
2025-05-09T15:42:43.685+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----
test>
```

Atualização board



Transferimos o arquivo eletronicos.json para dentro do container MongoDB. Ele contém os dados que serão inseridos no banco.

```
PS C:\Users\João\Documents\FIAP\DevOps\Cp2.2> docker container ls
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
3f02cc93fb9f   mongo    "docker-entrypoint.s..." 15 minutes ago Up 15 minutes 0.0.0.0:27017->27017/tcp   mongo
PS C:\Users\João\Documents\FIAP\DevOps\Cp2.2> docker cp eletronicos.json mongo:/eletronicos.js
Successfully copied 3.07kB to mongo:/eletronicos.js
```

Utilizamos comandos para importar os dados do arquivo eletronicos.json para uma coleção no MongoDB.

```
PS C:\Users\João> docker exec -it mongo bash
root@mongo:/# mongosh < /eletronicos.js
Current Mongosh Log ID: 681e26f184fc86e47bd861df
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.0
Using MongoDB:      8.0.9
Using Mongosh:       2.5.0

For mongosh info see: https://www.mongodb.com/docs/mongosh-shell/

-----
The server generated these startup warnings when booting
2025-05-09T15:42:42.279+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See h
http://dochub.mongodb.org/core/prodnotes-filesystem
2025-05-09T15:42:43.684+00:00: Access control is not enabled for the database. Read and write access to data and configur
ation is unrestricted
2025-05-09T15:42:43.685+00:00: For customers running the current memory allocator, we suggest changing the contents of th
e following sysfsFile
2025-05-09T15:42:43.685+00:00: We suggest setting the contents of sysfsFile to 0.
2025-05-09T15:42:43.685+00:00: vm.max_map_count is too low
```

```

test> use produtos
switched to db produtos
produtos>

produtos> db.eletronicos.insertMany([
...   {
...     produto: "Smartphone",
...     tipo_produto: "Celular",
...     quantidade_vendida: 100,
...     valor_venda: 599.99
...   },
...   {
...     produto: "TV LED",
...     tipo_produto: "Televisor",
...     quantidade_vendida: 50,
...     valor_venda: 799.99
...   },
...   {
...     produto: "Notebook",

```

Por fim, acessamos o MongoDB e visualizamos os dados da coleção, confirmando que a importação foi bem-sucedida.

```

produtos> db.eletronicos.find()
[
  {
    _id: ObjectId('681e26f484fc86e47bd861e0'),
    produto: 'Smartphone',
    tipo_produto: 'Celular',
    quantidade_vendida: 100,
    valor_venda: 599.99
  },
  {
    _id: ObjectId('681e26f484fc86e47bd861e1'),
    produto: 'TV LED',
    tipo_produto: 'Televisor',
    quantidade_vendida: 50,
    valor_venda: 799.99
  },
  {
    _id: ObjectId('681e26f484fc86e47bd861e2'),
    produto: 'Notebook',

```

```
{
  _id: ObjectId('681e26f484fc86e47bd861e3'),
  produto: 'Tablet',
  tipo_produto: 'Dispositivo Móvel',
  quantidade_vendida: 30,
  valor_venda: 1200.5
},
{
  _id: ObjectId('681e26f484fc86e47bd861e4'),
  produto: 'Fone Bluetooth',
  tipo_produto: 'Acessório',
  quantidade_vendida: 200,
  valor_venda: 199.9
},
{
  _id: ObjectId('681e26f484fc86e47bd861e5'),
  produto: 'Smartwatch',
  tipo_produto: 'Relógio Inteligente',
  quantidade_vendida: 40,
```

```
    quantidade_vendida: 80,
    valor_venda: 299.99
  },
  {
    _id: ObjectId('681e26f484fc86e47bd861e8'),
    produto: 'Câmera Digital',
    tipo_produto: 'Fotografia',
    quantidade_vendida: 20,
    valor_venda: 1500
  },
  {
    _id: ObjectId('681e26f484fc86e47bd861e9'),
    produto: 'Monitor Gamer',
    tipo_produto: 'Acessório de PC',
    quantidade_vendida: 25,
    valor_venda: 1899.9
  }
]
produtos>
```