

# Relatório de Deploy em Produção - CalculatorPlus

Este documento descreve o processo de deploy da aplicação **CalculatorPlus** em um ambiente de produção na AWS, utilizando Docker. O processo difere do ambiente de homologação, que utiliza um deploy manual, pela containerização da aplicação, o que garante consistência e portabilidade.

## Passo 1: Configuração do Dockerfile

O processo começa com a definição de um **Dockerfile** multi-stage. Esta abordagem otimiza a imagem final, separando o ambiente de build do ambiente de execução.

- **Stage 1 (Build):** Usa uma imagem Maven para compilar o código-fonte Java e gerar o arquivo `.jar`.
- **Stage 2 (Final):** Usa uma imagem JRE (Java Runtime Environment) leve, para a qual apenas o `.jar` compilado é copiado. Isso resulta em uma imagem de produção significativamente menor e mais segura.

```
# Stage 1: Build do projeto
FROM maven:3.9.4-eclipse-temurin-11 AS build

# Diretório de trabalho dentro do container
WORKDIR /app

# Copia apenas o pom.xml primeiro para aproveitar cache do Docker
COPY pom.xml .

# Baixa as dependências do Maven (cacheável)
RUN mvn dependency:go-offline -B

# Copia todo código-fonte
COPY src ./src

# Faz o build do projeto (produz o JAR)
RUN mvn clean package -DskipTests

# Stage 2: Imagem final mais leve
FROM eclipse-temurin:11-jre

WORKDIR /app

# Copia o JAR do stage de build
COPY --from=build /app/target/CalculatorPlus-1.0.0.jar app.jar

# Porta que a aplicação vai expor
EXPOSE 9000

# Comando para rodar a aplicação
ENTRYPOINT ["java", "-jar", "app.jar"]
```

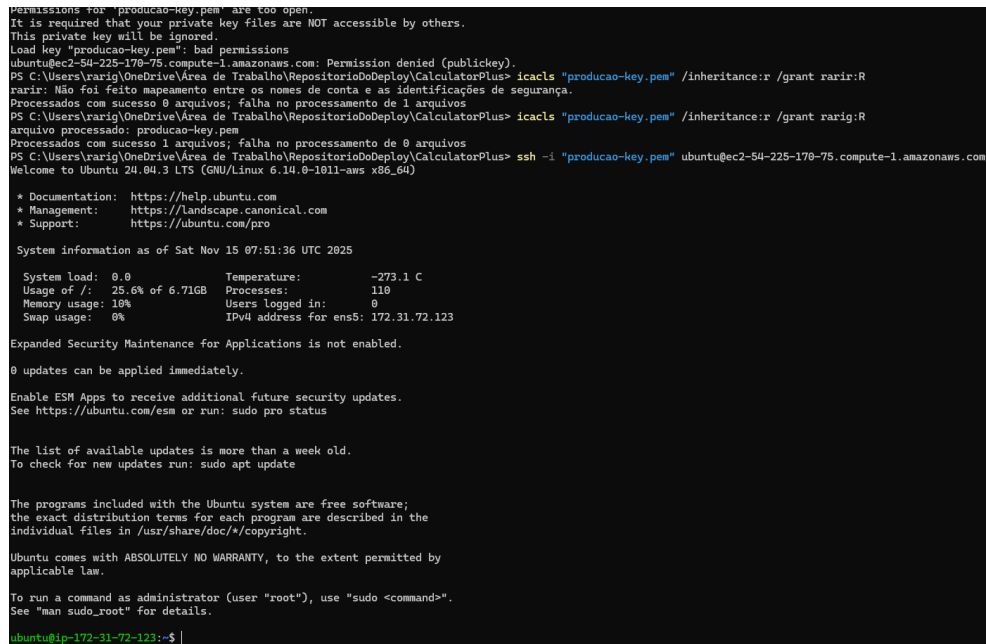
Figura 1: Dockerfile

## Passo 2: Acesso ao Servidor de Produção

O deploy é realizado em uma instância EC2 na AWS. O acesso é feito de forma segura via SSH, utilizando uma chave privada (.pem).

Comandos:

```
# Ajuste de permissão da chave
icacls "producao-key.pem" /inheritance:r /grant:r Rari:R
# Acesso SSH
ssh -i "producao-key.pem" ubuntu@ec2-54-225-178-75.compute-1.amazonaws.com
```



```
Permissions for 'producao-key.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "producao-key.pem": bad permissions
ubuntu@ec2-54-225-178-75.compute-1.amazonaws.com: Permission denied (publickey).
PS C:\Users\rarig\OneDrive\Área de Trabalho\RepositorioDoDeploy\CalculatorPlus> icacls "producao-key.pem" /inheritance:r /grant rarig:R
rarig: Não foi feito mapeamento entre os nomes de conta e as identificações de segurança.
Processados com sucesso 0 arquivos; falha no processamento de 1 arquivos
PS C:\Users\rarig\OneDrive\Área de Trabalho\RepositorioDoDeploy\CalculatorPlus> icacls "producao-key.pem" /inheritance:r /grant rarig:R
arquivo processado: producao-key.pem
Processados com sucesso 1 arquivos; falha no processamento de 0 arquivos
PS C:\Users\rarig\OneDrive\Área de Trabalho\RepositorioDoDeploy\CalculatorPlus> ssh -i "producao-key.pem" ubuntu@ec2-54-225-178-75.compute-1.amazonaws.com
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1011-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sat Nov 15 07:51:36 UTC 2025

System load:  0.0           Temperature:   -273.1 C
Usage of /:   25.6% of 6.71GB Processes:      110
Memory usage: 10%          Users logged in: 0
Swap usage:   0%           IPv4 address for ens5: 172.31.172.123

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-72-123:~$ |
```

Figura 2: Acesso SSH

## Passo 3: Preparação do Ambiente do Servidor

Uma vez conectado ao servidor, os pacotes do sistema operacional são atualizados para garantir que todas as correções de segurança e atualizações de software estejam aplicadas.

Comando:

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

## Passo 4: Instalação do Docker

O Docker Engine é instalado no servidor. Ele é a peça central do deploy em produção, responsável por gerenciar o ciclo de vida dos containers.

Comando:

```
sudo apt install -y docker.io
```

```

ubuntu@ip-172-31-72-123:~$ sudo apt update && sudo apt upgrade -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [126 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1309 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [828 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1585 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [299 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [175 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [15.7 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1489 kB]

```

Figura 3: Atualização de Pacotes

```

ubuntu@ip-172-31-72-123:~$ sudo apt install -y docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan

```

Figura 4: Instalação do Docker

## Passo 5: Ativação e Verificação do Docker

Após a instalação, o serviço do Docker é habilitado para iniciar automaticamente com o sistema e é iniciado. A versão é verificada para confirmar que a instalação foi bem-sucedida.

**Comandos:**

```

sudo systemctl enable docker
sudo systemctl start docker
docker --version

```

```

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-72-123:~$ sudo systemctl enable docker
ubuntu@ip-172-31-72-123:~$ sudo systemctl start docker
ubuntu@ip-172-31-72-123:~$ docker --version
Docker version 28.2.2, build 28.2.2-0ubuntu1-24.04.1
ubuntu@ip-172-31-72-123:~$

```

Figura 5: Verificação do Docker

## Passo 6: Instalação do Git

O Git é instalado para permitir a clonagem do código-fonte da aplicação diretamente do repositório para o servidor de produção.

**Comandos:**

```

sudo apt install git -y
git --version

```

## Passo 7: Clonagem do Repositório e Checkout

O código-fonte da aplicação é clonado. Em seguida, é feito o checkout para a branch producao-dockerFile, que contém o Dockerfile e o código a ser “containerizado”.

**Comandos:**

```

# (Após clonar o repositório)
git checkout producao-dockerFile

```

```

ubuntu@ip-172-31-72-123:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Fetched 126 kB in 1s (244 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
ubuntu@ip-172-31-72-123:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.3).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-72-123:~$ git --version
git version 2.43.0
ubuntu@ip-172-31-72-123:~$ |

```

Figura 6: Instalação do Git

```
cd CalculatorPlusProject/
```

```

ubuntu@ip-172-31-72-123:~/CalculatorPlus$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
ubuntu@ip-172-31-72-123:~/CalculatorPlus$ git checkout producao-dockerfile
branch 'producao-dockerfile' set up to track 'origin/producao-dockerfile'.
Switched to a new branch 'producao-dockerfile'
ubuntu@ip-172-31-72-123:~/CalculatorPlus$ git pull
Already up to date.
ubuntu@ip-172-31-72-123:~/CalculatorPlus$ ls
CalculatorPlusProject
ubuntu@ip-172-31-72-123:~/CalculatorPlus$ cd CalculatorPlusProject/
ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$ ls
Dockerfile  pom.xml  src

```

Figura 7: Navegação até o Dockerfile

## Passo 8: Build da Imagem Docker

Com o Dockerfile e o código-fonte no lugar, o comando `docker build` é executado. Este comando lê o Dockerfile e executa os passos definidos para construir a imagem da aplicação, que é tagueada como `calculatorplus:1.0`.

**Comando:**

```
sudo docker build -t calculatorplus:1.0 .
```

```

ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$ sudo docker build -t calculatorplus:1.0 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/

Sending build context to Docker daemon 8.957MB
Step 1/11 : FROM maven:3.9.4-eclipse-temurin-11 AS build
3.9.4-eclipse-temurin-11: Pulling from library/maven
43f89b94cd7d: Pulling fs layer
39769250f2be: Pulling fs layer
7d78bd96cc6f: Pulling fs layer
609b859ad36b: Pulling fs layer
442b4ebcd69d: Pulling fs layer

```

Figura 8: Início do Build

O processo de build multi-stage é executado, resultando em uma imagem Docker final e otimizada, como mostra a confirmação final do build.

## Passo 9: Execução do Container

Com a imagem pronta, o comando `docker run` é usado para iniciar um container. O container é nomeado `calculatorplus_prod`, executado em background (`-d`), e a porta 9000 da aplicação é mapeada para a porta 9000 do servidor.

```

ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
e3747e3548d3   calculatorplus:1.0   "java -jar app.jar"     22 seconds ago   Up 21 seconds   0.0.0.0:9000->9000/tcp, [::]:9000->9000/tcp   calculatorplus_prod
ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$

```

Figura 9: Fim do Build

Comando:

```
sudo docker run -d -p 9000:9000 --name calculatorplus_prod calculatorplus:1.0
```

```

ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$ sudo docker run -d \
-p 9000:9000 \
--name calculatorplus_prod \
calculatorplus:1.0
e3747e3548d3e045a22274b0c1d02bea19c4499aa5c8865727abde8ebce43b847
ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$

```

Figura 10: Iniciando o Container

## Passo 10: Confirmação Final

Finalmente, o comando `docker ps` é utilizado para listar os containers em execução e confirmar que a aplicação `calculatorplus_prod` está rodando (Up) e acessível na porta especificada.

Comando:

```
sudo docker ps
```

```

--> Removed intermediate container d2d7a8bb2450
--> 79231e058dd2
Step 7/11 : FROM eclipse-temurin:11-jre
--> 87af0f08113e3
Step 8/11 : WORKDIR /app
--> Using cache
--> 8429576570bb
Step 9/11 : COPY --from=build /app/target/CalculatorPlus-1.0.0.jar app.jar
--> 568f248c634f
Step 10/11 : EXPOSE 9000
--> Running in f98be61dfdea
--> Removed intermediate container f98be61dfdea
--> 0e51a2b4dfe3
Step 11/11 : ENTRYPOINT ["java", "-jar", "app.jar"]
--> Running in b426d6d5efeb
--> Removed intermediate container b426d6d5efeb
--> 364e7d7d6a6b
Successfully built 364e7d7d6a6b
Successfully tagged calculatorplus:1.0
ubuntu@ip-172-31-72-123:~/CalculatorPlus/CalculatorPlusProject$

```

Figura 11: Confirmação Final