

Manual para Deploy de Projetos



FRONTEND
34.41.164.86



BACKEND
34.31.132.210



BANCO
34.31.132.210

Sumário

1.	Considerações Iniciais	3
2.	Configuração da chave de acesso	4
3.	Configurações dos projetos	4
3.1.	Configurações para o BACKEND	4
3.2.	Configurações para o FRONTEND	4
4.	Acesso Remoto.....	5
5.	Operações com GitHub.	5
6.	Subir o Back.....	6
7.	Subir o Front.....	7
8.	Outros Comandos	8
9.	Versões de Java, Maven e Node	8

1. Considerações Iniciais

Este documento serve como manual para executar o deploy de projetos em um servidor na nuvem.

Existem outros tipos de servidores, bem como outros métodos para fazer deploy. A combinação entre eles gera uma grande diversidade de formas diferentes de fazer o deploy, cada uma indicada para um tipo de aplicação.

Esta combinação foi escolhida por tratar o assunto de forma facilitada, não muito diferente do que foi ensinado durante a residência.

Agora vamos a algumas definições para facilitar o entendimento do texto do manual.

TERMINAL - O terminal é uma interface de linha de comando que permite a interação direta com o sistema operacional, executando comandos de texto para realizar diversas tarefas, como navegação, manipulação de arquivos e execução de programas.

S.O. - Acrônimo para Sistema Operacional

PORTA – Uma porta em um S.O. é definida por número de identificação que é usado para direcionar o tráfego de rede para um aplicativo. Duas aplicações não podem usar a mesma porta para garantir que o tráfego de rede seja corretamente roteado para o aplicativo correto.

SAIR DO TERMINAL – Par sair do terminal de forma correta no Linux, similar ao desligamento correto de um computador, basta digitar o comando **exit** ou pressionar as teclas **Ctrl + D**.

PROCESSO - Um processo é uma instância de um programa em execução, que inclui o código do programa, seus dados, e o estado de execução atual.

PID - O process ID (PID) é um número único atribuído pelo sistema operacional a cada processo em execução, usado para identificar e gerenciar processos.

NOHUP - O nohup no Linux permite que um comando ou script continue a ser executado mesmo após o fechamento do terminal ou logout do usuário, ignorando o sinal hang-up (SIGHUP).

FOREGROUND - Um processo em foreground é um processo que está sendo executado e que mantém o controle sobre o terminal, bloqueando-o até que sua execução seja concluída ou interrompida.

& - O caractere **&** colocado no final de um comando executa o comando em background, permitindo que o terminal continue disponível para outros comandos.

BACKGROUND - Um processo em background é um processo que executa sem bloquear o terminal, permitindo que você continue usando o terminal para outras tarefas.

HOME DO USUÁRIO - É o diretório principal associado a um usuário específico em um sistema operacional. É onde os arquivos pessoais, configurações e dados do usuário são armazenados por padrão.

- **Windows:** É encontrado em C:\Users\nomedousuário. Se o nome de usuário for **collares**, o diretório seria C:\Users\collares.
- **Linux:** É encontrado em /home/nomedousuário. Se o nome de usuário for **collares**, o diretório seria /home/collares.
- O uso de \ no path do Windows e / no path do Linux é uma questão de convenção e compatibilidade histórica que facilita a organização e a interpretação de caminhos de arquivos em cada um dos **S.Os**, mas que gera muita confusão para os desenvolvedores.

2. Configuração da chave de acesso

As instruções do manual levarão em conta que o terminal está posicionado no home do usuário, que é o padrão.

Você receberá um arquivo **PROJXX** no e-mail, coloque-o no diretório **.ssh** no home do seu usuário do Windows, que fica em **C:\Users\nomeDoSeuUsuario**. Se o diretório **.ssh** não existir, pode criá-lo.

[IMPORTANTE] Esse arquivo é a **chave** de acesso para o servidor, não modifique seu nome ou seu conteúdo.

3. Configurações dos projetos

Essas configurações deverão ser feitas de forma definitiva e “**commitadas**” para o repositório, para que não gere problemas com uso de portas de outros projetos.

3.1. Configurações para o BACKEND

Editar o arquivo **application.properties**, acrescentando essa instrução, substituindo XX pelo número do projeto:

```
server.port=80XX
```

3.2. Configurações para o FRONTEND

Editar o arquivo **package.json**, acrescentando as informações destacadas em amarelo no trecho correspondente, respeitando os espaços e substituindo XX no final pelo número do projeto:

```
"scripts": {  
  "dev": "vite",  
  "build": "vite build",  
  "preview": "vite preview --host 0.0.0.0 --strictPort true --port 30XX"  
}
```

4. Acesso Remoto

Conexão com o servidor de **BACK**. Utilize a chave com o nome correto do seu projeto(**projXX**).

```
ssh -i ~/.ssh/projXX projXX@34.31.132.210
```

Conexão com o servidor de **FRONT**. Utilize a chave com o nome correto do seu projeto(**projXX**).

```
ssh -i ~/.ssh/projXX projXX@34.41.164.86
```

5. Operações com GitHub.

Nos servidores, tanto de BACK quanto de FRONT, o repositório deverá ser usado apenas para atualização do código no servidor, **apenas pull**. Não edite arquivos diretamente no servidor, isso é uma prática perigosa e só deverá ser feita em casos específicos e por pessoas experientes.

Todos deverão gerar suas respectivas senhas de desenvolvimento no GitHub, e cada um deverá usar a sua **SENHA** para atualizar o git. A senha deverá ser parecida com isso: ghp_Rbv4uDjyAWzGcQzLDsqegsURBePvnBRPXxo6.

Será solicitado usuário e **SENHA**, aquela que foi gerada no GitHub.

```
git clone enderecoGit
```

Navegar para o diretório do projeto.

```
cd diretorioDoProjeto
```

Atualizar o código do projeto, puxando do repositório do git.

```
git pull
```

6. Subir o Back

[IMPORTANTE] É necessário matar o processo anterior antes de subir o novo.

Este comando serve para pesquisar o **PID (Process Id)** do processo, utilizando a **porta** como filtro para pesquisa.

```
netstat -anp | grep :80XX
```

Para matar o processo, copie o **PID** retornado pelo comando anterior e cole em **pid**.

```
kill -9 pid
```

Apagar o arquivo **nohup.out** com os logs da execução anterior, para que a máquina não fique lenta.

```
rm -f nohup.out
```

Subir projeto com Maven.

[IMPORTANTE] Executar o item 3.1 para configurar o **application.properties** antes de rodar esse comando .

```
nohup mvn spring-boot:run &
```

Leitura do log continuamente no terminal, pode ser feita a qualquer momento dentro do diretório.

```
tail -f nohup.out
```

Sair da leitura contínua do log.

```
Ctrl + C
```

7. Subir o Front

[IMPORTANTE] É necessário matar o processo anterior antes de subir o novo.

Este comando serve para pesquisar o **PID (Process Id)** do processo, utilizando a **porta** como filtro para pesquisa.

```
netstat -anp | grep :::30XX
```

Para matar o processo, copie o **pid (Process Id)** retornado pelo comando anterior e cole em **pid**.

```
kill -9 pid
```

Na primeira utilização ou quando adicionar pacotes novos, rodar o comando de instalação.

```
npm install
```

Para rodar o build e gerar a versão de produção do front o comando é esse.

```
npm run build
```

Apagar arquivo **nohup.out** com os logs da execução anterior, para que a máquina não fique lenta.

```
rm -f nohup.out
```

Subir o projeto do front.

[IMPORTANTE] Executar o item 3.2 para configurar o **package.json** antes de rodar esse comando.

```
nohup npm run preview
```

Leitura do log continuamente no terminal.

```
tail -f nohup.out
```

Sair da leitura contínua do log.

```
Ctrl + C
```

8. Outros Comandos

Navegar para o diretório do projeto.

```
cd diretorioDoProjeto/
```

Listar conteúdo do diretório.

```
ls ou ls -l ou ll
```

Voltar para um diretório anterior.

```
cd ..
```

Ver a versão do NODE instalada. Apenas no servidor de Frontend.

```
node -v
```

Ver a versão do NPM instalada. Apenas no servidor de Frontend.

```
npm -v
```

Ver a versão do JAVA JRE instalada. Apenas no servidor de Backend.

```
java -version
```

Ver a versão do JAVA JDK instalada. Apenas no servidor de Backend.

```
javac -version
```

Ver a versão do MAVEN instalada. Apenas no servidor de Backend.

```
mvn -v
```

9. Versões de Java, Maven e Node

JAVA JDK	MAVEN	NODE
17	3.8.8	20.15.0