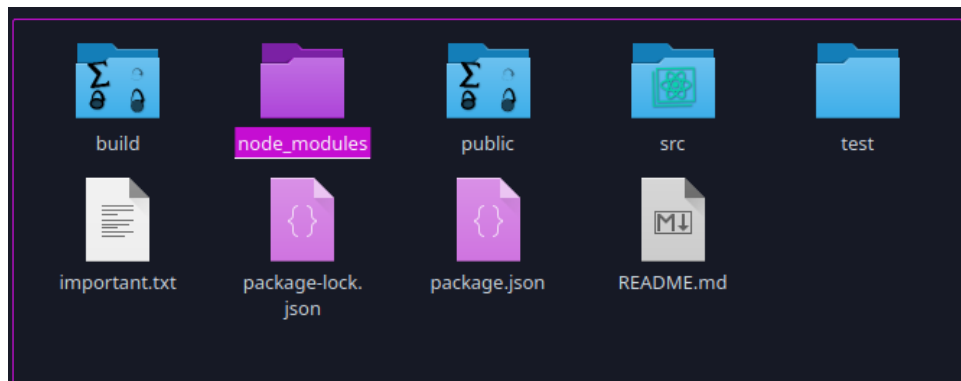


# Manual DevOps para LFAWeb

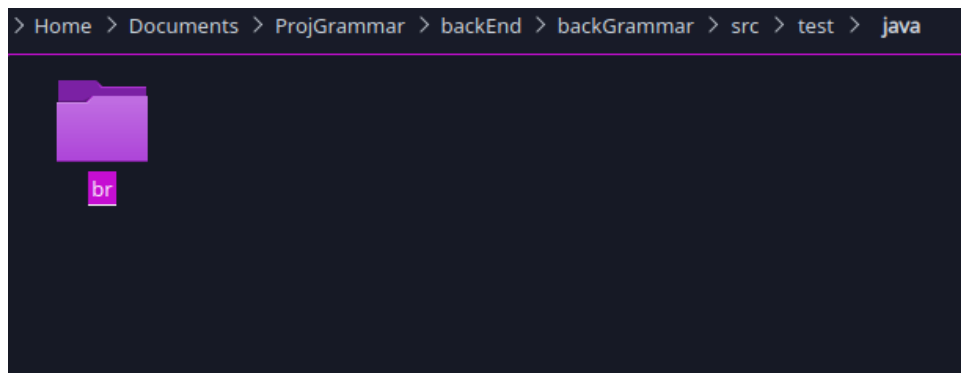
## 1. Preparação dos projetos para transferência :

- a. Copie as pastas dos projeto o backend e o frontend para alguma pasta  
(não se esqueça que antes de realizar a cópia dos arquivos tem está na branch develop).
- b. Após a cópia vá a pasta do frontEnd e apague a pasta “node\_modules”.

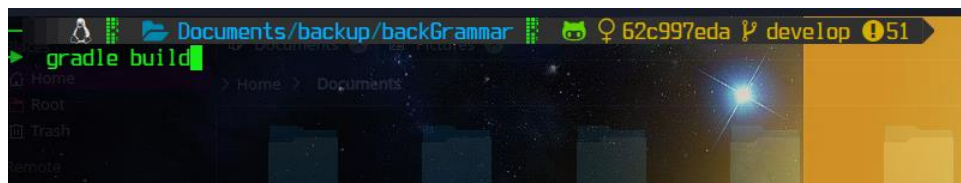


- c. Edite os seguintes arquivos :
  - i. FrontEnd:
    1. in package.json
      - a. "start": "PORT=80 react-scripts start",
      - b. remover a biblioteca robotJs (caso tiver).
    2. in file DataService.js
      - a. const URL = "<http://lfaweb.dcc.ufla.br:8080>";
  - ii. SpringBoot Project:
    1. UserController.java, WebSecurityConfig.java, GrammarController.java:
      - a. @CrossOrigin(origins = {"<http://lfaweb.dcc.ufla.br>"})

d. Não esqueça de retirar a pasta de teste do projeto do backend :



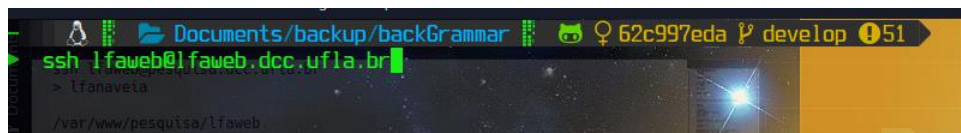
e. Na pasta do Backend execute o seguinte comando :



f. Feito isto, faça da pasta do frontEnd um zip (comprimir a pasta).

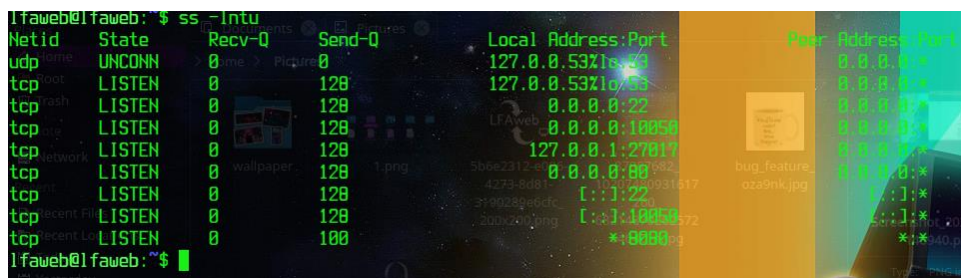
## 2. Preparação do servidor

a. Para acessar o servidor use o seguinte comando :



i. Senha: lfanaveia

b. No servidor execute o seguinte comando, para verificar as portas ativas:



- c. Após, se as portas 80 e/ou 8080 estiverem ativas execute os seguinte comando :

- i. Para derrubar as portas para substituir os arquivos:

1. `sudo kill -9 $(sudo lsof -t -i:80)`
2. `sudo kill -9 $(sudo lsof -t -i:8080)`

- d. Não se esqueça de apagar todos os arquivos contidos na pasta /applications:



```
lfaweb@lfaweb:~/applications$ ls
frontGrammar  frontGrammar.zip  Ifaweb-1.1.9-SNAPSHOT.jar
lfaweb@lfaweb:~/applications$
```

- i. `sudo rm <arquivos>`
- ii. `sudo rm -rf <pasta>`

### 3. Transferência de arquivos para o servidor

- a. Vá pasta que se fez a cópia do arquivos, então na pasta do backend :

- i. build/libs, copie o transfirar o arquivo jar com o comando:

1. `scp <nome_do_arquivo>.jar`

lfaweb@lfaweb.dcc.ufla.br:applications/

- b. Faça isto também para o arquivo zip criado usando a pasta do frontEnd:

- i. `scp <nome_do_arquivo>.zip lfaweb@lfaweb.dcc.ufla.br:applications/`

### 4. Rodar a aplicação no servidor

- a. Por recomendação reinicie o servidor para evitar problemas de falhas de termino de processos que vão ser criado.

- i. `sudo reboot now`

- ii. Espere um pouco para o servidor reiniciar e volte acessá-lo pelo ssh.

b. Agora vá para a pasta /applications:

- i. `cd applications/`

c. Backend:

- i. Execute este comando para testar o arquivo e já iniciar os próximos comando sem pedir a senha : `sudo java -jar <nome_do_arquivo>.jar`
- ii. `sudo java -jar <nome_do_arquivo>.jar &> /dev/null &`

d. FrontEnd:

- i. Descompacte o zip do frontEnd com o comando : `unzip <nome_do_arquivo>.zip`
- ii. Na pasta do frontEnd execute para reinstalar as bibliotecas:  
`sudo npm i`
- iii. Então execute o projeto do frontEnd, por recomendação use o comando screen:
  - 1. `sudo screen -S lfawebjs npm start`
  - 2. Espere terminar a compilação e `ctrl+a+d`

e. Faça o teste se está funcionando testando pelo browser e terminado.