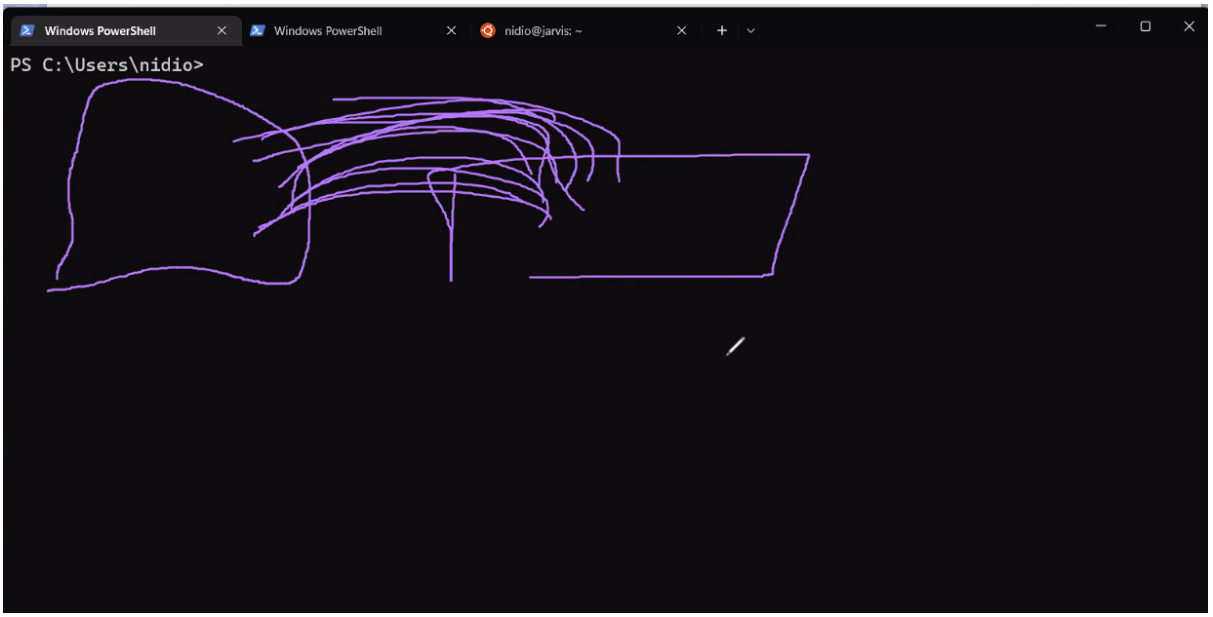
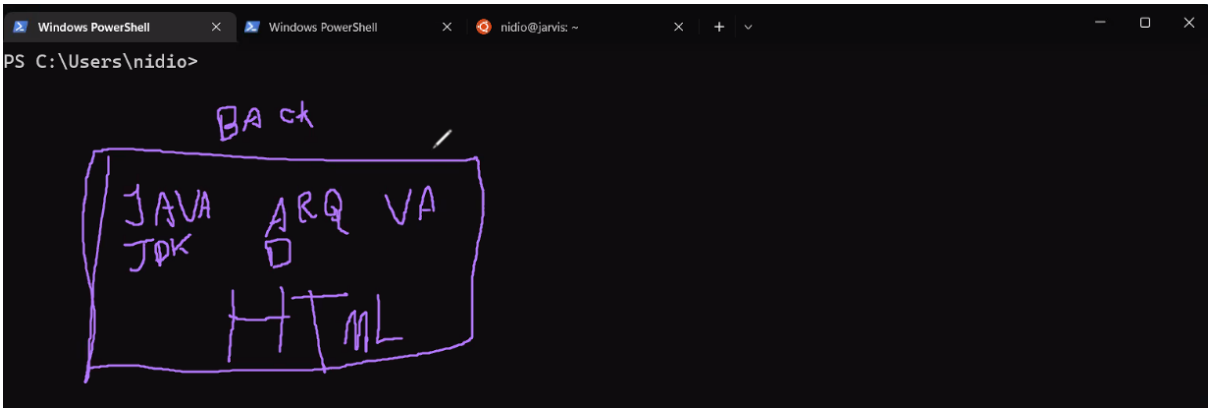
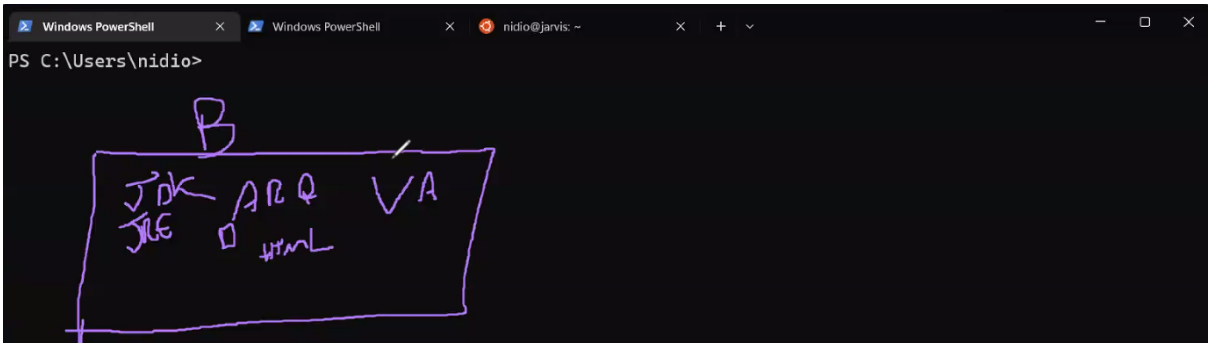
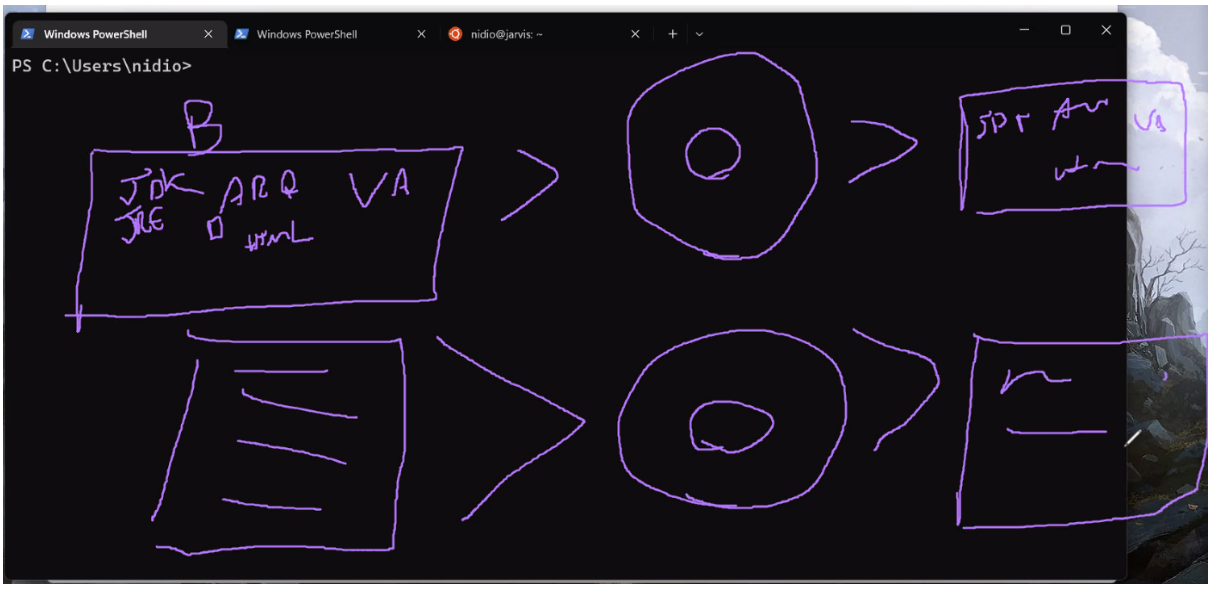
Aula 3

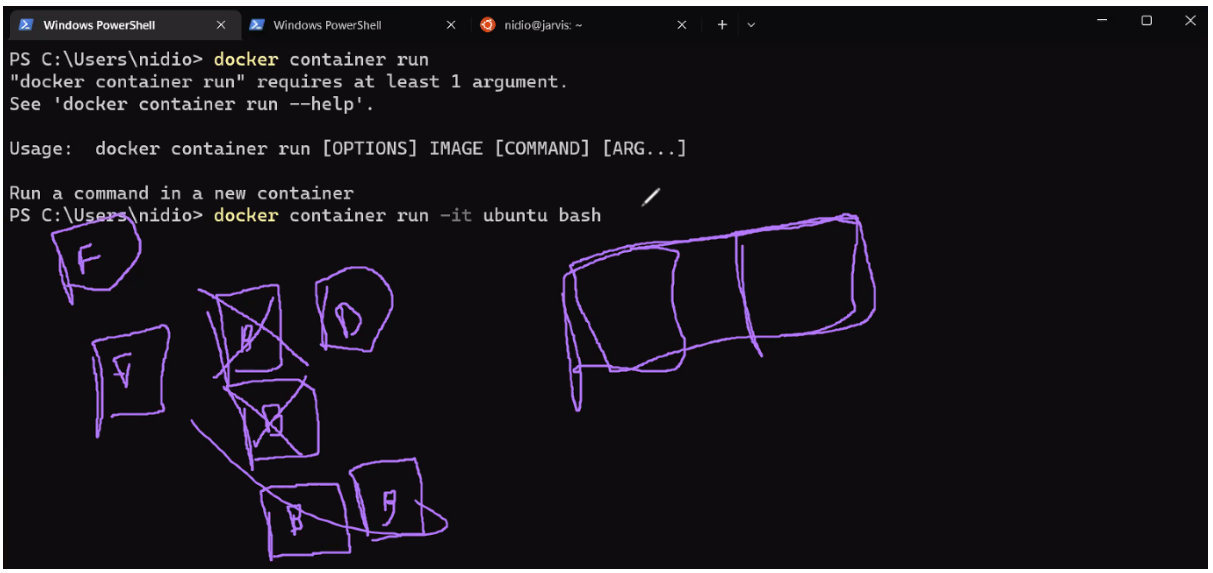

Windows PowerShell
Windows PowerShell
nidio@jarvis: ~
PS C:\Users\nidio>


Windows PowerShell
Windows PowerShell
nidio@jarvis: ~
PS C:\Users\nidio>
BA ck
JAVA
JDK
ARQ
VA
HTML


Windows PowerShell
Windows PowerShell
nidio@jarvis: ~
PS C:\Users\nidio>
B
JDK
JRE
ARQ
VA
HTML



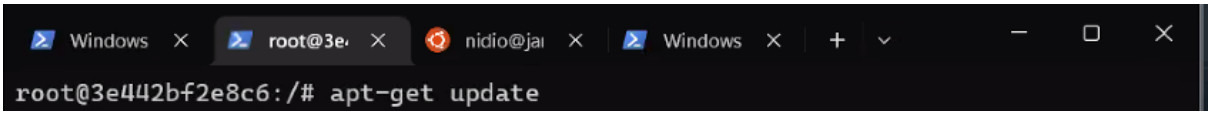
- (1) Docker file
- (2) Image ISO
- (3) Funciona



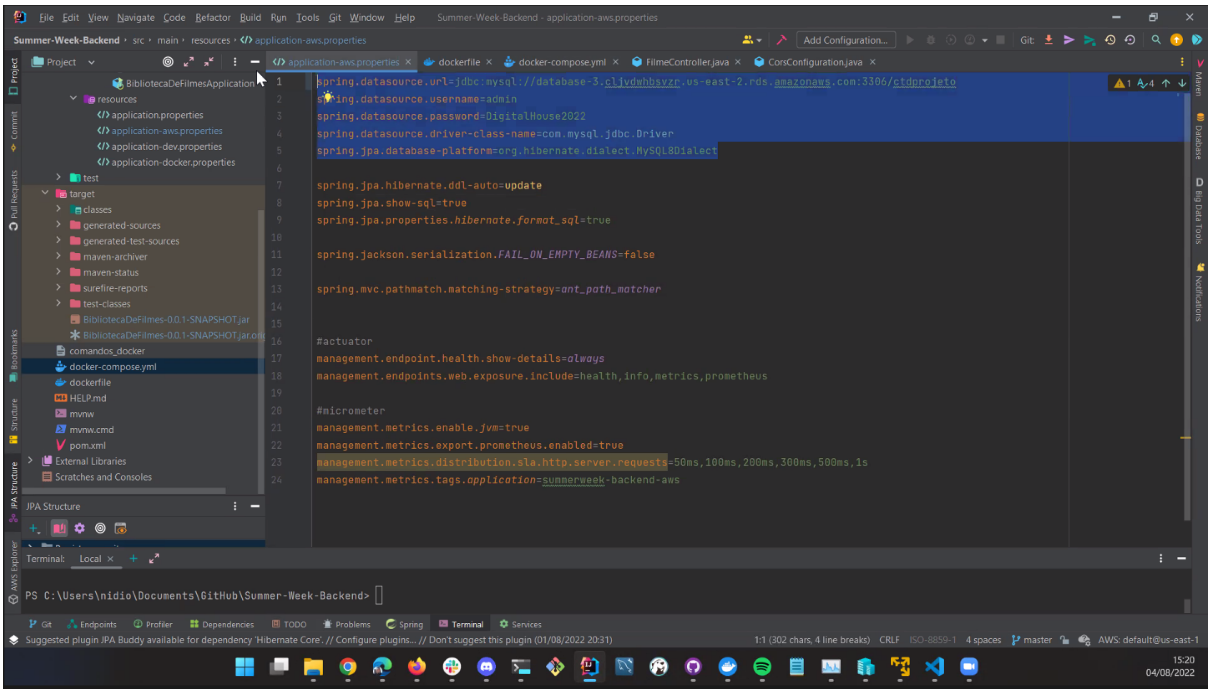
```
docker container run -it ubuntu bash
```

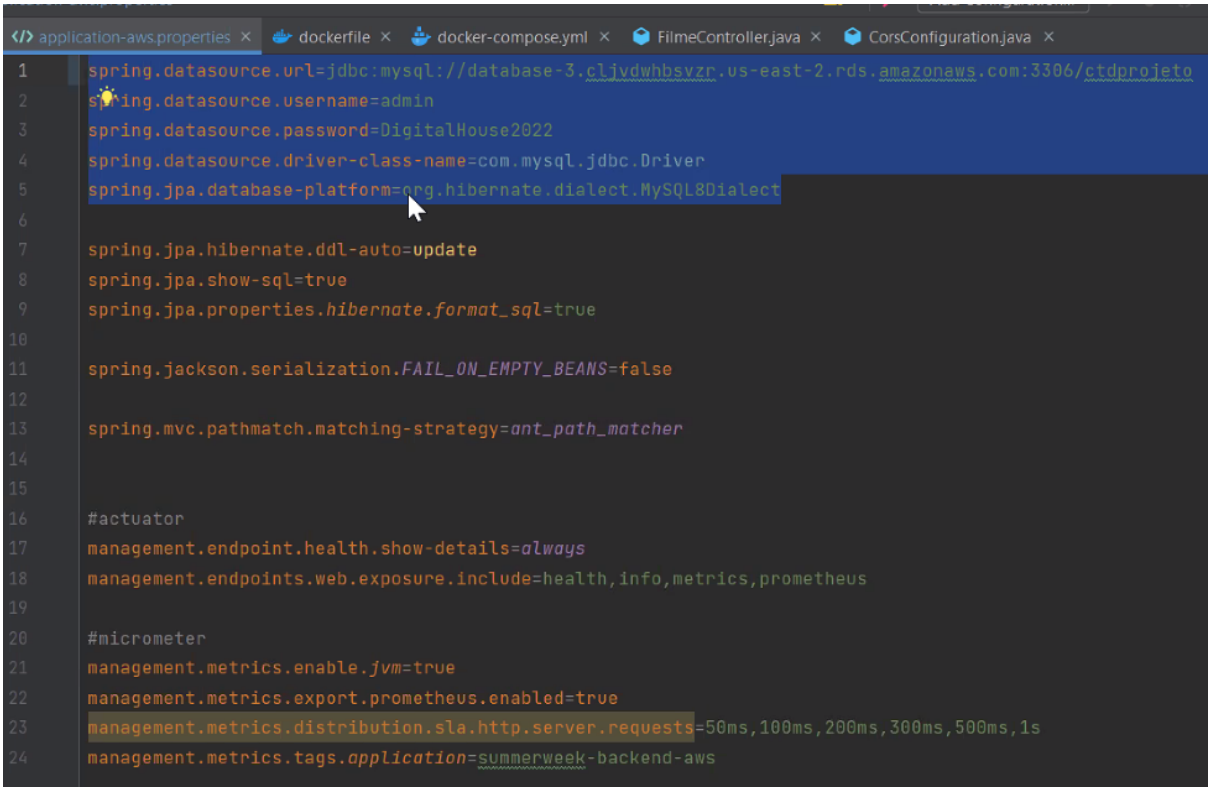
```
docker run -it alpine /bin/sh
```

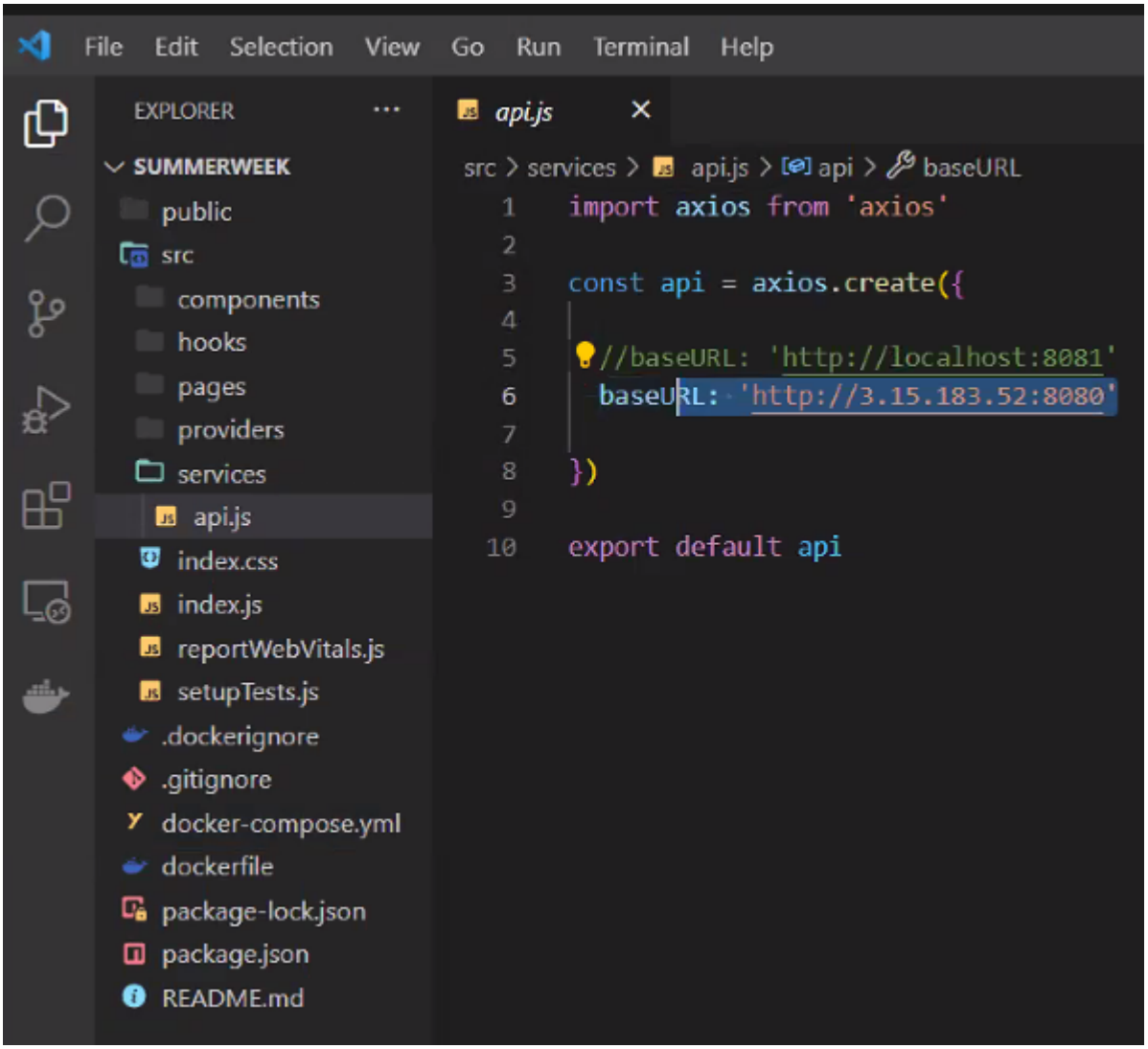
- i = interativo (vai dar a possibilidade de conseguir digitar alguma coisa)
- t = terminal (aloca um pseudo terminal para o containe)
- it = terminal interativo



- Docker in Docker: quando usa um Docker para fazer a criação de um outro Docker (por exemplo: no GitHub, o qual usa o Docker como servidor de pacote e compilador, daí chega um momento que você vai criar uma Image ISO de Docker)
- Ele não é recomendado para cenários de produção, de teste… porque a chance de ele dar erro é muito grande.




application-aws.properties
dockerfile
docker-compose.yml
FilmeController.java
CorsConfiguration.java
spring.datasource.url=jdbc:mysql://database-3.cljvdwhbsvzr.us-east-2.rds.amazonaws.com:3306/ctdprojeto
spring.datasource.username=admin
spring.datasource.password=DigitalHouse2022
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true
spring.jackson.serialization.FAIL_ON_EMPTY_BEANS=false
spring.mvc.pathmatch.matching-strategy=ant_path_matcher
#actuator
management.endpoint.health.show-details=always
management.endpoints.web.exposure.include=health,info,metrics,prometheus
#micrometer
management.metrics.enable.jvm=true
management.metrics.export.prometheus.enabled=true
management.metrics.distribution.sla.http.server.requests=50ms,100ms,200ms,300ms,500ms,1s
management.metrics.tags.application=summerweek-backend-aws


File Edit Selection View Go Run Terminal Help
EXPLORER
SUMMERWEEK
public
src
components
hooks
pages
providers
services
api.js
index.css
index.js
reportWebVitals.js
setupTests.js
.dockerignore
.gitignore
docker-compose.yml
dockerfile
package-lock.json
package.json
README.md
api.js
src > services > api.js > api > baseURL
1 import axios from 'axios'
2
3 const api = axios.create({
4
5 //baseURL: 'http://localhost:8081'
6 baseURL: 'http://3.15.183.52:8080'
7
8 })
9
10 export default api