Clientes

– CRUD genérico de clients y direcciones (considerando histrix, autenticación, monitoreo y creando cliente retro)

Al crear y borrar una dirección considerar la existencia del cliente

– Al borrar un cliente se debe validar si tiene direcciones o pedidos.

Pedidos

– CRUD genérico de pedidos (considerando histrix, autenticación, monitoreo y creando cliente retro)

– Al crear un pedido, se debe validar que exista el cliente y la dirección.

Orden:

Entorno funcionando

Base de datos

Contenedores

eclipse

crud

Cliente retrofix

Histrix

Autenticacion

Monitoreo

Consideraciones histrix:

Solo considerar las invocaciones interservicios.

Invocaciones interservicios (crud, cliente y histrix):

Pedido por cliente.

Pedido por dirección.

Validar cliente.

Validar direccion.

InvalidAddressException

InvalidAddressExceptionMapper

AddressService

AddressServiceClient

AddressService

AddressServiceRemoteInvokerCommand (2 en el caso de order)

Registrar en app y en configuration app

94 de la casa

30 de adelanto para la otra semana

50 para la pitahaya

exportar la variable del host de Postgres

export DATABASE\_HOST=localhost

Levantamos el Docker Compose

./up.sh

Validanmos con

docker-compose ps

#Asegurarnos que corren todos los contenedores (Up)

Generamos los tags de la migración

java -jar ./build/libs/service.jar db tag ./src/main/resources/config.yaml 2018\_08\_15\_19\_42

Aplicamos la migración a la base de datos

java -jar ./build/libs/service.jar db migrate ./src/main/resources/config.yaml

============== Comandos utilitarios =================

Usar el cliente de Postgres, el password es c7

psql -h localhost -U c7 -W c7\_clientes

Listar volumenes de datos que Docker gestiona

docker volume ls

Limpiar los volumenes (si deseo borrarlos, no se recuperan)

docker volume rm id\_del\_volumen