

Es la puerta de entrada

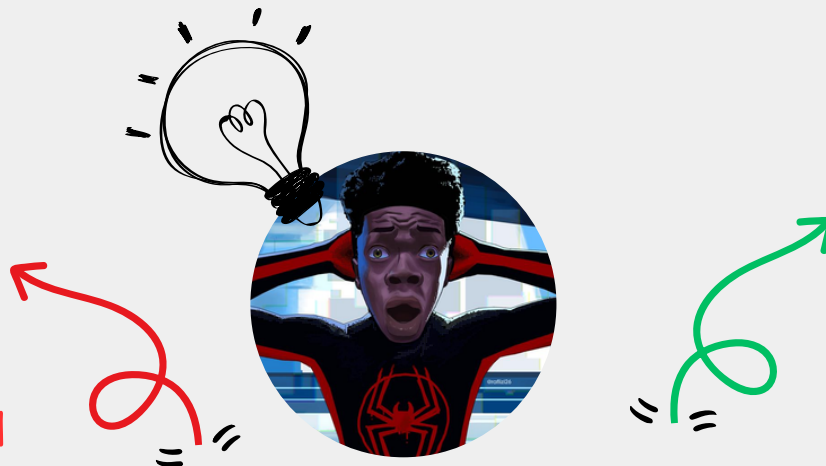
## Controller

```
ContarStock(){
ContarStock();
}
TSave(T){
S.save();
}
```

Implementa el JPA

## IRepository

JPA → {  
(A, ID) -All  
(B, ID) -findById  
(D, ID) -save  
@Query -delete  
ContarStock -update



Se relacionan indirectamente con el IRepository

## IService

T ← Save()  
List<T> ← all()  
Optional<T> ← findById()  
Void ← update()  
Void ← delete()  
Void ← delete Logic()  
List<T> ← List()

Implementa los metodos del IService

## Service

```
Save(){...}
all(){...}
findById(){...}
update(){...}
delete(){...}
deleteLogic(){...}
List(){...}
ContarStock(){...}
```

## Entity

A, B, D

//Se crean las tablas y columnas para la base de gatos//

# PROYECTO SPRING BOOT



Interface: Qué hacer (DTO, IRepository, IService)  
Class abstrac: Cómo hacer pero no permite (AGeneric, Service, Controller)  
Class: cómo (Entity, DTO, Service)

## API

HTTP://localhost:9000/API/

/\*API responsable de dar una URL  
Por ejemplo:  
/api/users  
\*/