

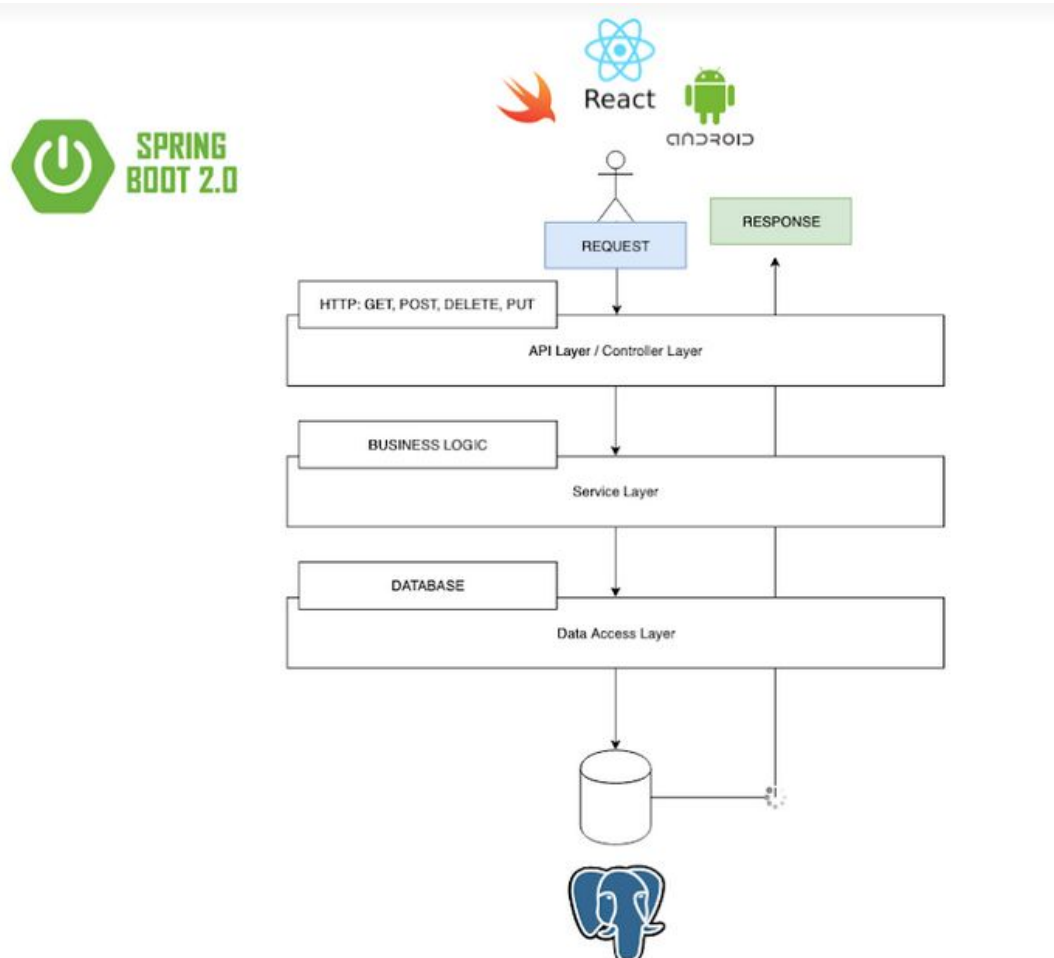
Spring Amigos Code partie 1:

Le but :

Pendant une entretien, on m'a posé une question sur Spring et ne sachant pas répondre, j'ai décidé de me lancer dessus pour y répondre.

Le tutoriel :

[Spring Boot 2.1 Tutorial | 2019](#)



Problème rencontré:

Comment savoir si mon port 80 est utilisé ?

ouvrir mon cmd et lancer les différentes commandes pour voir les différents processus.

```
netstat -aon | findstr 8080
tasklist | findstr <pid>
tasklist | findstr <processus_name/application_name>
netstat -aon | findstr <pi>
```

Impossible de lancer mon postman, comment tester mon api ?

J'ai dû utiliser Swagger ce qui a été très facile voir
https://www.youtube.com/watch?v=gduKpLW_vdY

Raccourcis clavier getter/setters : Alt + Shift +S,R
Raccourcis clavier du constructeur: Alt + Shift +S,O

Une classe n'est pas chargé ?

Cliquez sur le projet
 >Allez sur Maven
 > Update Maven (alt + f5)

Interrogation :

Les conventions de nommage des DAO :

<https://developer.ibm.com/tutorials/j-genericdao/>
<https://stackoverflow.com/questions/12575773/standard-naming-of-dao-and-hibernate-classes>

Comment fonctionne l'injection de dépendance :

<https://www.youtube.com/watch?v=oqYRI06DNHQ&>

Comment bien créer son projet Spring ?

<https://developer.ibm.com/technologies/spring/tutorials/j-spring-boot-basics-perry>

Pense bête :

A chaque début de projet, il faut créer les **packages** suivants :

- config
- api
- service
- dao
- model

Exemple de design pattern pour Spring :

> Model : *Person*
> Dao : *<interface> PersonDao*
> Api : *PersonController* : *PersonService* personService
> Service : *PersonService* : *PersonDao* personDao

Les **annotations** utilisées pendant ce projet :

Api
<pre>@RequestMapping("URL") @RestController @Autowired @PostMapping @GetMapping @GetMapping(path = "{A}") @PathVariable("A") @DeleteMapping(path = "{B}") @PathVariable("B") @PutMapping @PathVariable("C") XX @Valid @NotNull @RequestBody XX</pre>

Service
<pre>@Service @Autowired</pre>

```
@Qualifier("BDD")
```

```
Model
```

```
@NotBlank
```

```
@JsonProperty("A") @JsonProperty("B")
```

```
Model
```

```
@Repository("BDD")
```

```
@SpringBootApplication
```

Comment **déployer** son projet avec Maven ?

Clique droit sur le projet.

-> Allez sur *Maven*.

-> Cliquez sur *Maven install* ce qui va créer mon fichier .jar et les fichiers dans le dossier target.

Allez dans le dossier target puis écrire `java -jar LENOMDUTRUC.jar`