

Team K

Emma ALLAIN
Roxane BACON
Antoine FADDA RODRIGUEZ
Clément LEFEVRE

512 Eats

Rendu O2

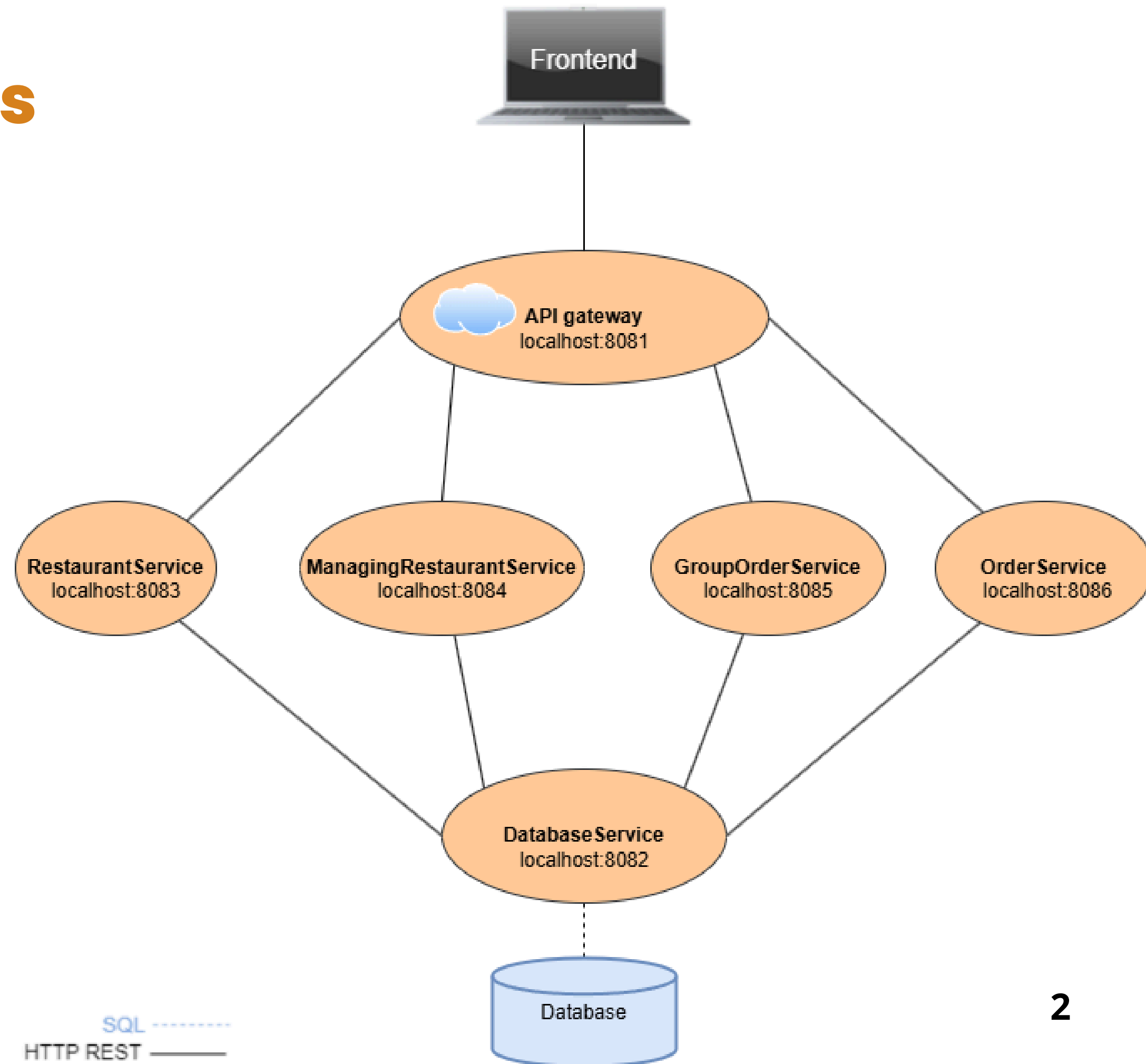


Architecture des codes

Architecture initiale visée

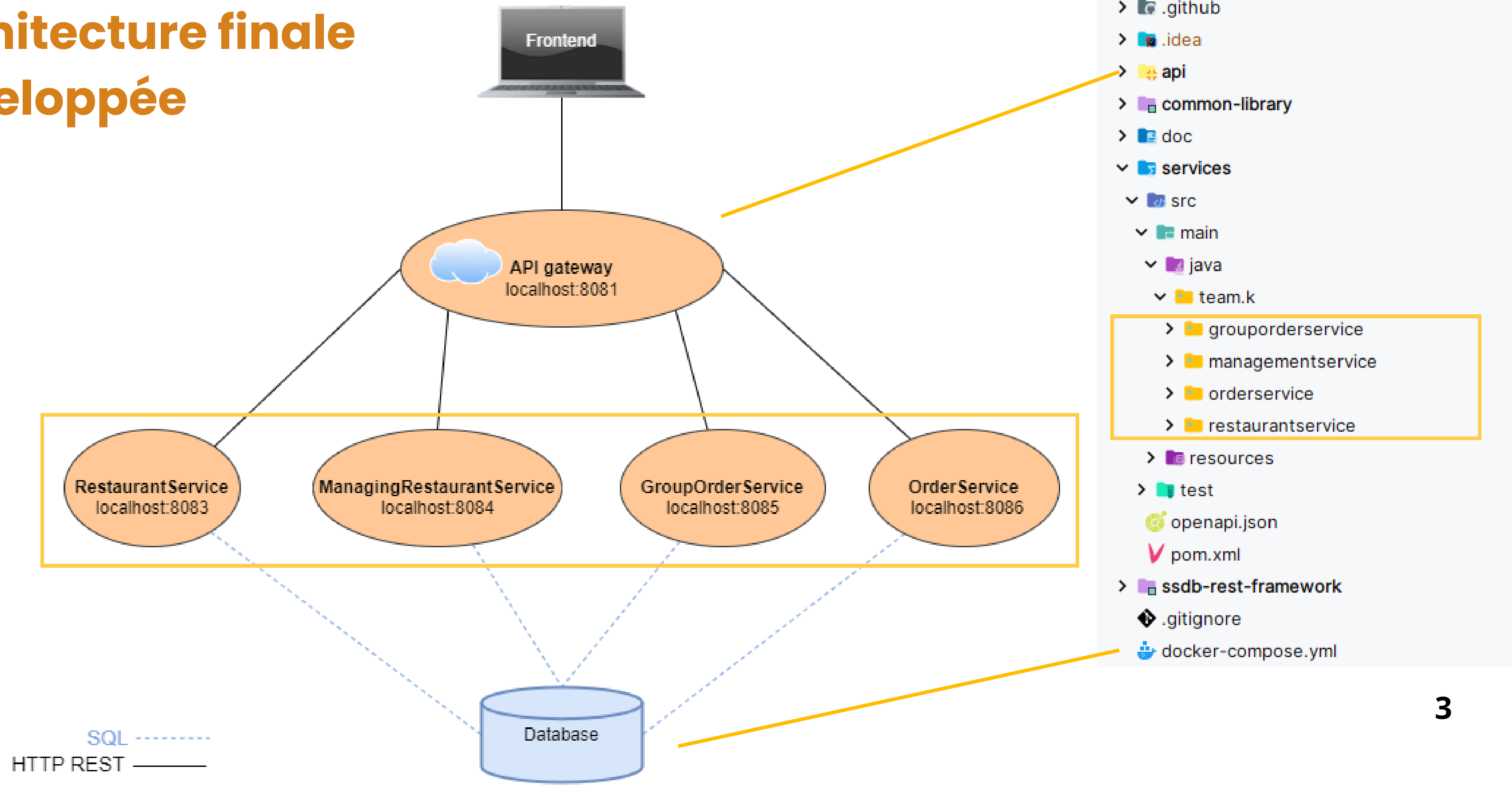
Inconvénients

- Moteur de base de données inadapté (SQLite)
 - Mauvaise gestion des dates
- ORM mal choisi (Ebean)
 - Fonctionnalités trop simples
 - Peu de documentation
- Ré-implémentation de SQL en REST

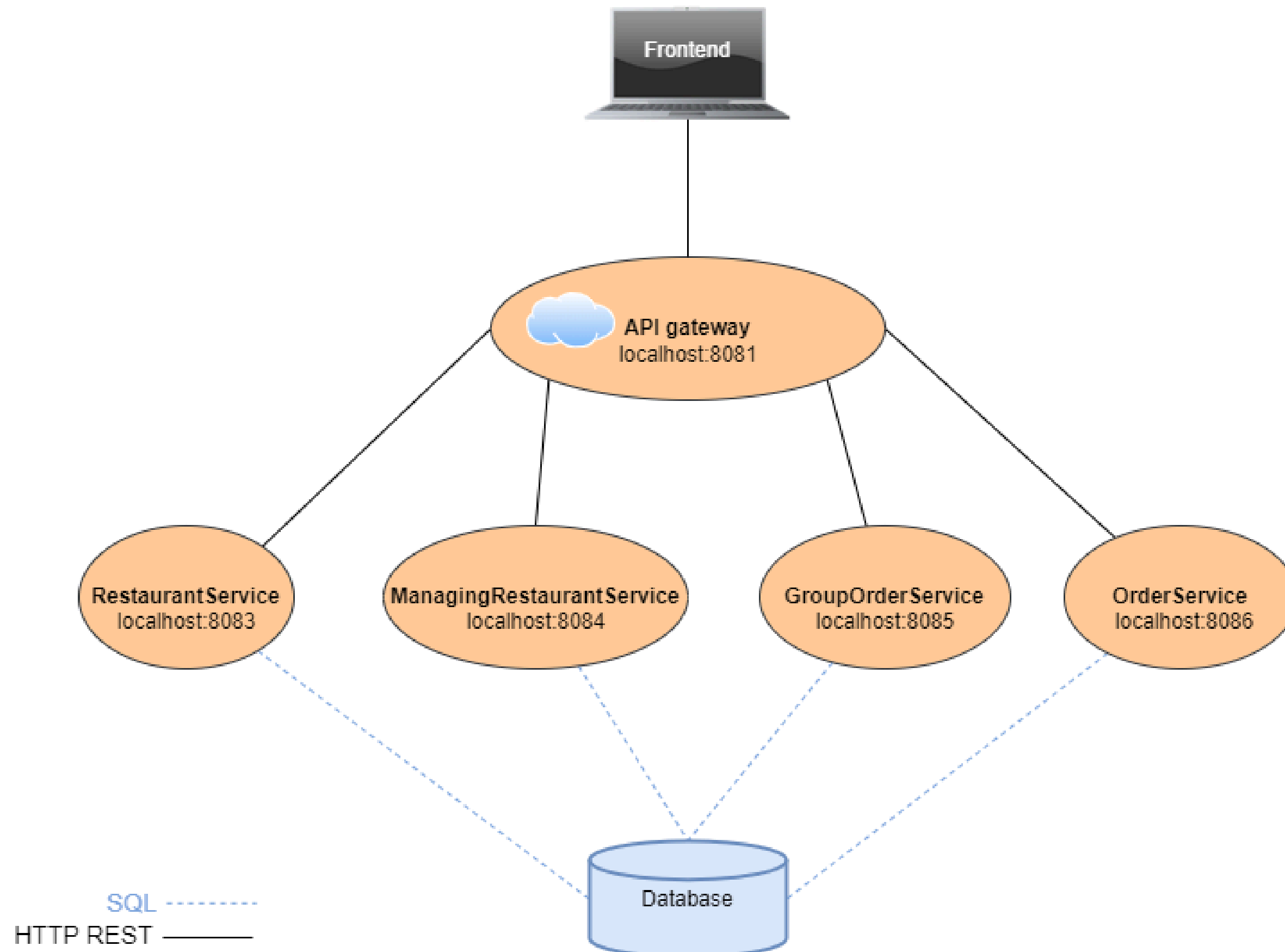


Architecture des codes

Architecture finale développée



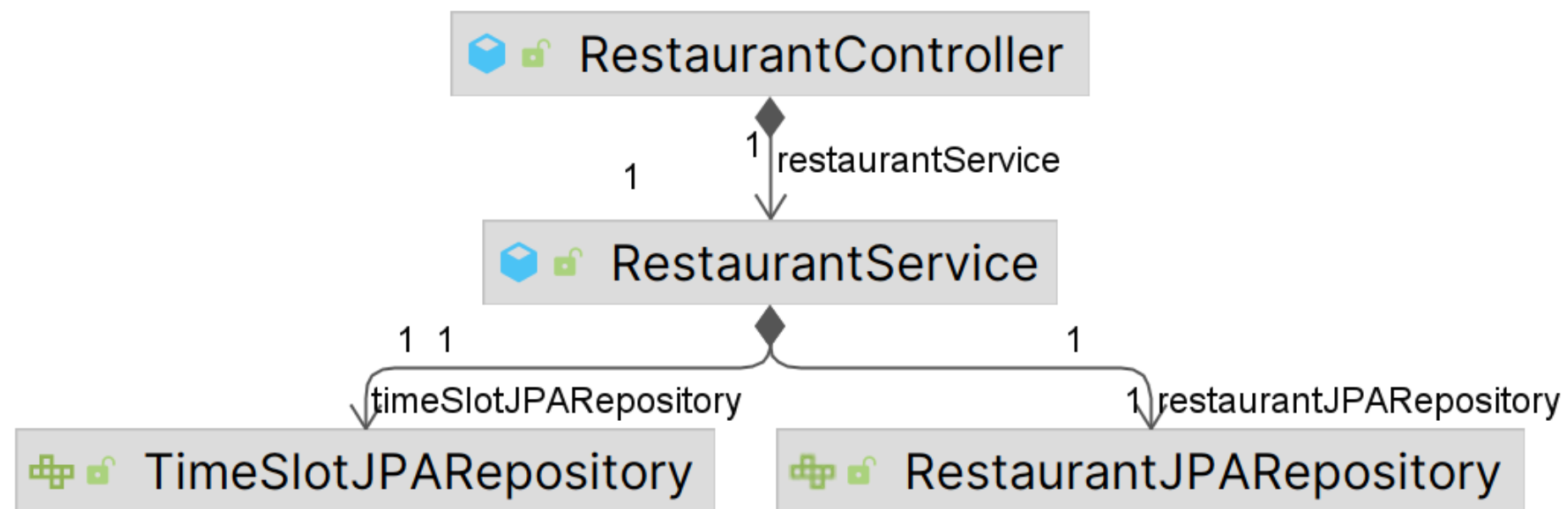
Architecture des codes



Architecture des codes

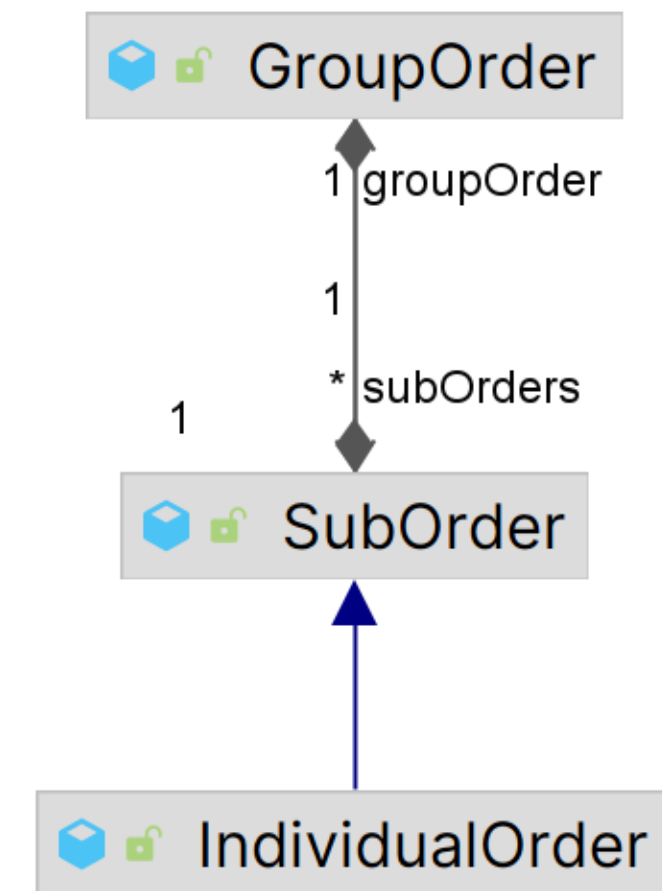
Bon découpage

- Simplicité d'ajout d'autres fonctionnalités



Mauvais héritage

- Lien contraignant entre SubOrder et IndividualOrder



Tests et qualité des codes



Tests
Cucumber



Tests
Unitaires



Gestion
d'erreur



Objectif du
projet



Couverture

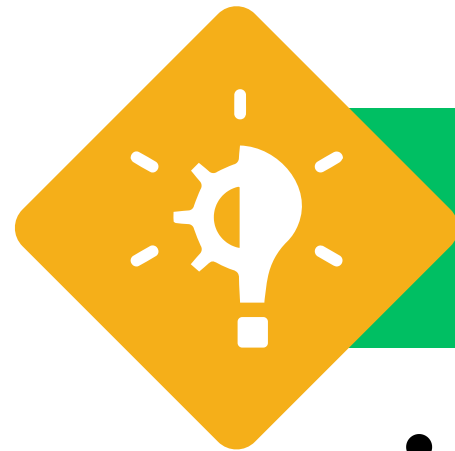
team	60% (12/20)	52% (65/124)	40% (214/532)
k	60% (12/20)	52% (65/124)	40% (214/532)
controller	0% (0/7)	0% (0/28)	0% (0/102)
repository	100% (8/8)	67% (36/53)	66% (42/63)
DishRepository	100% (1/1)	42% (3/7)	37% (3/8)
GroupOrderRepository	100% (1/1)	75% (3/4)	75% (3/4)
IndividualOrderRepository	100% (1/1)	66% (4/6)	77% (7/9)
LocationRepository	100% (1/1)	80% (4/5)	66% (4/6)
RegisteredUserRepository	100% (1/1)	66% (4/6)	66% (4/6)
RestaurantRepository	100% (1/1)	90% (10/11)	86% (13/15)
SubOrderRepository	100% (1/1)	57% (4/7)	57% (4/7)
TimeSlotRepository	100% (1/1)	57% (4/7)	50% (4/8)
service	100% (4/4)	80% (29/36)	76% (172/224)
GroupOrderService	100% (1/1)	85% (6/7)	75% (41/54)
ManageRestaurantService	100% (1/1)	100% (7/7)	93% (31/33)
OrderService	100% (1/1)	80% (8/10)	72% (73/101)
RestaurantService	100% (1/1)	66% (8/12)	75% (27/36)
ServicesAPI	0% (0/1)	0% (0/7)	0% (0/143)

```
Feature: Browse available dishes of one restaurant

Scenario:
  Given A restaurant "512EatRestaurant" with a dish "Strawberry salad" and a dish "Candy apple"
  When The user wants to have dishes of the restaurant "512EatRestaurant"
  Then the user gets the dishes "Strawberry salad" and "Candy apple" registered in the restaurant selected

Scenario:
  Given A restaurant "512BurgerRestaurant" with no dishes registered
  When The user wants to have dishes non registered of the restaurant "512BurgerRestaurant"
  Then The user gets no dishes registered in the restaurant selected and have this message : "No dishes available"
```

Tests et qualité des codes



Code de bonne qualité

- Une bonne lisibilité du code, des scénarios clairs et divers recouvrant la gestion des erreurs
- La responsabilité des services est mieux répartie
- Bonne ergonomie de l'interface



Code à améliorer

- Fonctionnalités à implémenter dans le front :
 - Profil internet user
 - Consulting bloqué
 - Gestion de la localisation dans la group order
- Nos endpoints ne sont pas conventionnels

```
@Endpoint(path = "/get/sub-order", method = HttpMethod.GET)
```

Gestion de projet

Gestion de projet

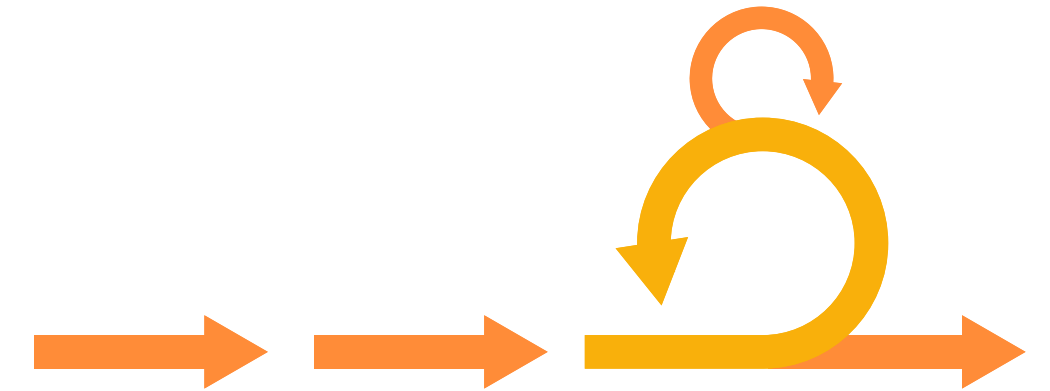
1 Sprint = 1 Semaine

Création de 2 binômes avec fonctionnement de pair-programming afin d'avancer le front et le back

User story

Conservation de l'usage des user story pour construire le front end afin de correspondre aux besoins utilisateurs

Méthode SCRUM



Gestion des repos

2 projets distincts :

- Backend
- Frontend

Agilité

Remise en question tardive de la 1ère architecture afin de rebondir sur une plus efficace.



512[®]

Team K

Emma ALLAIN
Roxane BACON
Antoine FADDA RODRIGUEZ
Clément LEFEVRE

Merci

De votre attention