

## TD6 : collection, filtrage et pagination

L'objectif du TD est de programmer des requêtes portant sur des collections et comportant du filtrage de valeurs et de la pagination.

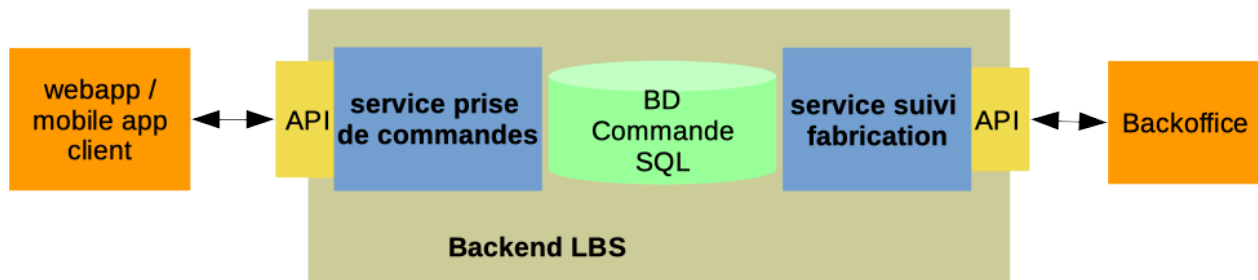
### 0. préparation

Les requêtes prévues dans ce TD portent sur les listes de commandes **mais sont à réaliser dans un service différent du service de prise de commandes** développé dans les TD précédents.

Il s'agit du service de suivi de la fabrication/préparation des commandes, qui est utilisé pour réaliser le backoffice point de vente de l'application.

Ce service utilise la même base de données, mais propose une API totalement indépendante. Il est programmé de manière entièrement distincte et ne partage aucun code avec le service de prise de commandes. Il peut donc être développé, déployé et maintenu indépendamment.

D'un point de vue pratique, il faut donc déclarer un nouveau service dans la composition docker, et lui permettre d'accéder au service de gestion de données pour les commandes.



### 1. collection

Programmer la requête qui permet d'obtenir la collection complètes des commandes.

L'objet json retourné doit contenir des méta-données incluant notamment le nombre d'éléments dans la collection, et un tableau de commandes. Chaque commande est décrite par ses id, nom du client, date de création et date de livraison, montant et contient un lien vers la ressource commande correspondante.

Les commandes sont triées par date de livraison.

Le json aura donc la forme suivante :

```
{
  "type": "collection",
  "count": 1750,
  "commands": [ {
    "command": {
      "id": "30d5909a-a273-4220-a19a-e2c907410ccf",
      "nom": "Marcel.Renard",
      "created_at": "2020-12-17 15:09:48",
      "livraison": "2020-12-17 18:29:01",
      "status": 5 },
    "links": {
      "self": { "href": "/commands/30d5909a-a273-4220-a19a-e2c907410ccf/" }
    }
  }, {
    "command": {
      "id": "6133d128-9781-4f18-92da-dc691b7fe2c4",
      "nom": "Gilles.Tessier",
      "created_at": "2020-12-17 23:28:37",
      "livraison": "2020-12-19 10:43:34",
      "status": 5 },
    "links": {
```

```

        "self": { "href": "/commands/6133d128-9781-4f18-92da-dc691b7fe2c4/" }
      }
    }, { ... }
  ]
}

```

## 2. filtrage

On souhaite pouvoir filtrer les commandes selon leur état. Pour cela, on met en place un paramètre optionnel dans l'uri collection sur les commandes, permettant de sélectionner un état.

Ainsi, la requête :

GET /commandes retourne toutes les commandes enregistrées, et la requête :

GET /commandes?s=3 retourne toutes les commandes dont le status est "3" .

L'objet json retourné aura la même forme que dans le cas de l'uri non filtrée.

## 3. pagination

On souhaite mettre en place un mécanisme de pagination permettant de parcourir la collection lorsqu'il y a un grand nombre de données. Le mécanisme de pagination doit être utilisable en même temps que les filtres.

Le principe consiste à ajouter dans l'uri un numéro de page. Par défaut, on considère que les pages contiennent 10 éléments.

Ainsi, l'uri :

- GET /commandes : par défaut, retourne la 1ère page, c'est à dire les 10 premiers éléments,
- GET /commandes?page=3 : retourne la 3ème page, c'est à dire, 10 éléments à partir du 21ème.

Les méta-données de la réponse indiquent :

- count : le nombre d'éléments total répondant à la requête (indépendamment de la pagination)
- size : le nombre d'éléments dans la page courante

## 4. pagination avancée

Compléter le mécanisme de pagination avec les fonctionnalités suivantes :

1. on peut indiquer la taille des pages souhaitées dans la requête, qui remplace alors la taille par défaut : GET /commandes?page=3&size=15 retourne 15 éléments à partir du 31ème.
2. si le numéro de page demandé est < 0, retourne la 1ère page,
3. si le numéro de page demandé est supérieur au nombre de pages, retourne la dernière page.
4. des liens permettant de parcourir les pages sont ajoutés dans la réponse.

```
GET /commandes?page=4&size=15
```

```

{
  "type": "collection",
  "count": 1502,
  "size": 15,
  "links": {
    "next": {
      "href": "/commandes/?page=5&size=15"
    },
    "prev": {
      "href": "/commandes/?page=3&size=15"
    }
  }
}

```

```
    },
    "last": {
      "href": "/commandes/?page=94&size=15"
    },
    "first": {
      "href": "/commandes/?page=1&size=15"
    }
  },

  "commandes": [
    ...
  ]
}
```