Gérer les codes de réponse HTTP

Dans le standard REST, les codes de retours HTTP sont importants car ils ont une sémantique. Il convient donc de retourner les bons codes de réponse HTTP dans chacun de nos endpoints.

Cas nominal

Pour le cas nominal, on peut utilise l'annotation `@ResponseStatus` sur la méthode endpoint. On lui passe en paramètre le status à l'aide de l'annotation `HttpStatus`.

Exemple :

```
@GetMapping
@ResponseStatus(HttpStatus.CREATED)
public TodoDTO createTodo(@RequestBody CreateTodoDTO dto){
...
}
```

Cas d'erreur

Pour gérer les cas d'erreur on peut lancer des exceptions spécifiques définies par le framework, comme par exemple :

- `BadRequestException` : pour le status 400 Bad Request
- `ResourceNotFoundException` : pour le status 404 not found.

Il est également possible d'attribuer des codes de retours à des exception personnalisées qui étende `RuntimeException` avec l'annotation `@ResponseStatus` :

```
@ResponseStatus(value = HttpStatus.I_AM_A_TEAPOT)
public class IamATeapotException extends RuntimeException {
    public MyResourceNotFoundException() {
        super();
```

```
    }
    public MyResourceNotFoundException(String message, Throwable cause) {
        super(message, cause);
    }
    public MyResourceNotFoundException(String message) {
        super(message);
    }
    public MyResourceNotFoundException(Throwable cause) {
        super(cause);
    }
}
```

Revision #2
Created 2021-02-03 17:07:28 UTC by Arsène Lapostolet
Updated 2021-02-03 18:25:56 UTC by Arsène Lapostolet