

[LABORATOIRES](#) > [PAYROLL](#)

# Department

## Intention pédagogique

Afin de faire le lien avec la théorie "Les ORMs", nous allons enrichir fonctionnellement notre projet "Payroll" et ainsi continuer l'étude du framework Spring.

Nous pourrons ainsi voir par la pratique la modélisation spécifique pour un ORM, approche un peu différente du MCD/MLD classique.

Nous allons également nous entraîner à l'utilisation de Git-Flow depuis l'ouverture d'une nouvelle "branch/feature" jusqu'à la publication d'une release.

## Voici le point de départ



GitHub - ETML-ES-FWBE/payroll: warm up spring with a payroll project  
GitHub



## Objectifs opérationnels

### Cahier des charges

A cette étape du laboratoire, il s'agit d'ajouter la notion de "département" à notre projet. Dans l'idée de pouvoir engager des employés au sein d'un département.

Nous réaliserons cela en deux étapes. L'une pour l'ajout d'un département, puis l'engagement d'un employée.

⚠ Nous ne traiterons ici que de la notion de département. Le mécanisme de "hire" sera traité plus tard.

## Etapes à suivre

### Etape 001 - Comprendre le backlog

Prenez le temps de réfléchir aux impacts que cette nouvelle étape va induire sur notre projet actuel. Quelle stratégie de branches ? Comment découper le backlog.

#### Complément besoin métier

- Un employé
  - ne peut travailler que pour un département.
  - est identifié de manière unique par une adresse email.
- Un département
  - peut être créé sans avoir d'employé.
  - est identifié de manière unique par son nom.

#### Impacts architecturaux

- Toutes les couches (layers) Spring vont être impactées

### Etape 002 - Architecture

Mettez à jour votre diagramme de classe en intégrant "department".

- livraison sur la branche "develop"

## Etape 003 - POJO department

Ecrire le POJO décrivant le département. Omettez volontairement les annotations spécifiques pour hybernate.

- livraison sur la branche "feature/department"

## Etape 004 - Entité department


Modifier votre POJO pour le faire évoluer en Entité et le Repository nécessaire.

 [Spring et les annotations JPA](#) ➤

## Etape 005 - Controller

Adapter les routes pour pouvoir manipuler la ressource "department".

- Débutez par écrire les appels curl pour les différents besoins.
- Créer le controller et valider chacun des méthodes.
- Livraison sur la branche "feature/department".

 Attention [aux verbe HTTP](#) ➤ ainsi qu'au [nommage des ressources](#) ➤

## Etape 006 - Intégrer la nouvelle fonctionnalité

Une fois que votre "department" est fonctionnel, intégrez cette fonctionnalité dans la branche "develop".

Previous  
L'architecture Spring

Next  
Folder Structure

Last updated 1 month ago