**Tutoriel développer des Micro-Services avec Spring Boot et Spring Cloud dans l'IDE ECLIPSE**

Rédigé le : 20/10/2020

Table des matières

[1 Micro-Service de CONFIGURATION 2](#_Toc54114006)

[1.1 Développement du Micro-Service de CONFIGURATION CENTRALISEE 2](#_Toc54114007)

[1.1.1 Créer un nouveau projet SPRING-STARTER-PROJECT dans l'IDE 2](#_Toc54114008)

[1.2 Test du Micro-Service de CONFIGURATION CENTRALISEE 2](#_Toc54114009)

[1.3 Vérifier la couverture des tests du Micro-Service de CONFIGURATION CENTRALISEE 3](#_Toc54114010)

# Micro-Service de CONFIGURATION

**Objectif** : créer un micro-service TECHNIQUE **UNIQUE** chargé de gérer *toutes* les configurations de *tous* les micro-services utilisés par une application Web.

## Développement du Micro-Service de CONFIGURATION CENTRALISEE

### Créer un nouveau projet SPRING-STARTER-PROJECT dans l'IDE

1. **Créer un nouveau SPRING STARTER PROJECT** dans l'IDE (ECLIPSE) .

*MAVEN est automatiquement embarqué dans le projet ECLIPSE ainsi que le Conteneur WEB TOMCAT*.

**Le package choisi doit être le même quelque soit le micro-service** si on veut pouvoir créer une application *monolithique* en faisant juste glisser les packages ou classes provenant de différents projets ECLIPSE sous un explorateur Windows.

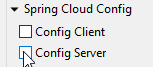
Je mets *systématiquement* levy.daniel.application à la fois comme groupId Maven et comme package Maven pour tous les micro-services que je développe.

* Le groupId Maven indiqué dans le pom.xml du projet ECLIPSE sert à **nommer les classes buildées par Maven (espace de nommage)**. Par exemple, si on met un groupId Maven = "toto" et que l'on crée une classe 'Exemple.java', le nom complet (aussi appelé nom qualifié en Java) de la classe buildée sera toto.Examplejava.
* Le Package Maven indiqué dans le pom.xml du projet ECLIPSE sert à **créer automatiquement les packages des classes *sources*** puis **binaires .class** lors de la génération automatique des packages dans le projet ECLIPSE par Maven.
* Par exemple, ECLIPSE créera automatiquement les packages : src.main.java.levy.daniel.application et src.test.java.levy.daniel.application

dans le projet ECLIPSE si package = levy.daniel.application.

### Choisir les dépendances SPRING CLOUD d'un Micro-Service de Configuration

Dans Spring Cloud Config > Config Server



Dans Spring Cloud Discovery > Eureka Discovery Client



|  | Groupe de dépendances | Dépendance |
| --- | --- | --- |
| 1 | Spring Cloud Config | Config Server |
| 2 | Spring Cloud Discovery | Eureka Discovery Client |
| 3 | Developer Tools | Spring Boot Dev Tools |
|  |  |  |

### Particularités du POM d'un Micro-Service de Configuration

Il peut être utile de sauter les tests unitaires lors du build Maven.

Dans l'élément <properties>, rajouter :

<maven.test.skip>true</maven.test.skip>

Documentation Swagger2

http://localhost:9000/v2/api-docs

## Test du Micro-Service de CONFIGURATION CENTRALISEE

## Vérifier la couverture des tests du Micro-Service de CONFIGURATION CENTRALISEE

Micro-Service

Micro-Service

Micro-Service

Micro-Service