



Spring MVC - Validation des formulaires

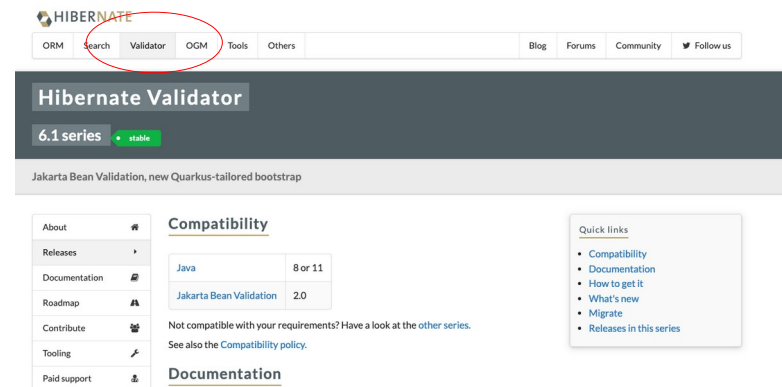
Validation standard

- Il existe une spécification JSR 303/309 puis JSR 380 specification of the Java API for bean validation
- depuis sa version 4, Spring donne accès aux fonctionnalités de validation (à condition de fournir les librairies de validation)
- Valider les saisies de l'utilisateur est devenu un standard dans les pratiques de développement , ce qui semble plutôt logique

Hibernate pour la validation

- Ce qui est moins intuitif ,c 'est que c'est hibernate qui va proposer une implémentation qui fait référence.
- Télécharger les librairies

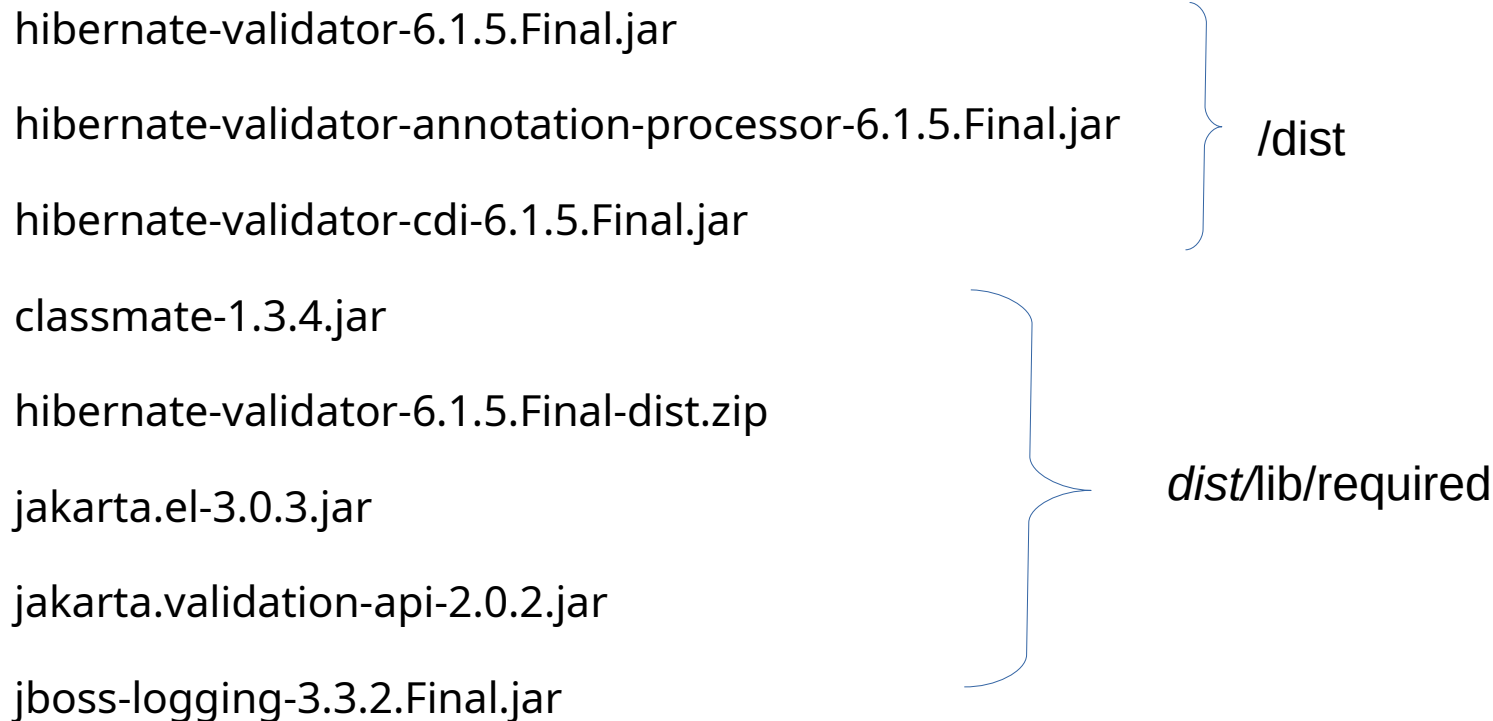
www.hibernate.org



Télécharger l'archive Hibernate Validator : [hibernate-validator-6.1.5.Final-dist.zip](#)

<https://sourceforge.net/projects/hibernate/files/hibernate-validator/6.1.5.Final/hibernate-validator-6.1.5.Final-dist.zip/download>

hibernate-validator-6.15.Final



Extraire les jars de l'archive .zip téléchargée, et les placer dans le répertoire des dépendances de votre projet (l'emplacement *WEB-INF/lib*)

Vérifier que ces librairies sont incluses au classpath de votre application.

La validation est maintenant disponible dans notre application

Les besoins en validation

Vérifier les saisies d'un utilisateur signifie :

- Les champs requis sont bien non null (required)
- Les valeurs numériques sont comprises dans un intervalle (min & max)
- Les code postaux par exemple, respectent un format attendu (expression régulières)
- Les règles de validation personnalisées sont respectées sur les champs concernés (custom validation)

<http://www.beanvalidation.org>

Quelques annotations de validation

Annotation	Description
@NotNull	Vérifier que la valeur annotée n'est pas null
@Min	Le champ doit être un nombre \geq à une valeur
@Max	Le champ doit être un nombre \leq à une valeur
@Size	La taille du champ doit correspondre à une taille donnée
@Pattern	Le champ doit vérifier l'expression régulière donnée
@Future / @Past	Le champ doit être une Date future ou passé par rapport à une date donnée.
et d'autres ...	



Au programme de ce paragraphe

- On mis en place notre environnement
- Valider champ requis
- Validation d'un nombre dans un intervalle
- Validation par regex
- Règle de validation personnalisée

Hibernate n'est pas qu'un ORM

- Hibernate Validator est un projet indépendant compatible avec Java Bean Validation API.
- Hibernate fourni une implementation d'une specification JSR
- JSR – specification
- Implementations = vendor

Le fonctionnement que nous voulons

Champ requis

Remplir le formulaire SVP.

Prénom :

Nom (*) :

Valider

(*) signifie que le champ est obligatoire



Remplir le formulaire SVP.

Prénom :

Nom (*) : Champ obligatoire

Valider

(*) signifie que le champ est obligatoire

Flow diagram

