

TD3

Question 1 :

JPA signifie Java Persistence API, son intérêt est de permettre la gestion entre l'objet java et la base de données

Question 2 :

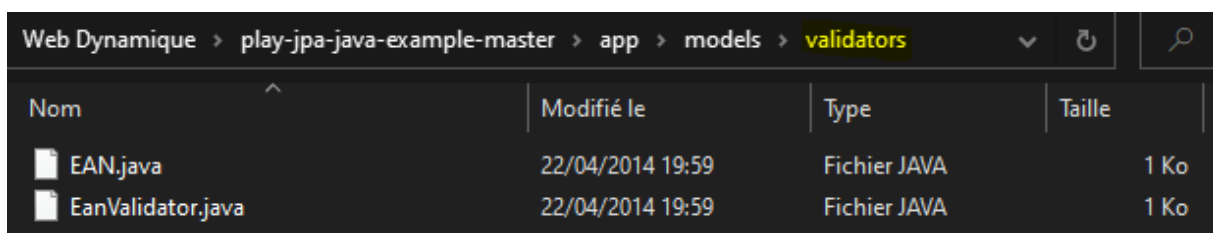
Les deux modèles sont :



- User
- Product

Question 3 :

Le « @ID » dans la classe Product sert à identifier un produit

Question 4 :



Web Dynamique > play-jpa-java-example-master > app > models > validators			
Nom	Modifié le	Type	Taille
 EAN.java	22/04/2014 19:59	Fichier JAVA	1 Ko
 EanValidator.java	22/04/2014 19:59	Fichier JAVA	1 Ko

Le dossier validators contient des fichiers qui valide l'EAN (« European Article Numbering » code barre européen)

Question 5 :

Les 5 vues sont les suivants : details, list, login, main et registration.

Question 6 :

Ceci est un formulaire de produit

Question 7 :

C'est un bouton qui nous envoie vers la page détails du produit

Question 8 :

C'est un bouton qui supprime le produit

Question 9 :

Login nous permet de nous connecter et registration nous permet de s'enregistrer

Question 10 :

Les méthodes sont les suivantes :

- *Secret Key* : sert à crypter les données
- *The application languages* : La langue de l'application
- *Global object class* : L'ensemble des objets utilisé pour l'application
- *Router* : définit toutes les routes utilisé pour l'application
- *Database configuration* : permet de configurer la base de données
- *Evolutions* : permet de suivre et organiser les évolutions de notre schéma de base données.
- *Logger* : Utile pour la surveillance, le débogage, le suivi des erreurs et la veille économique.

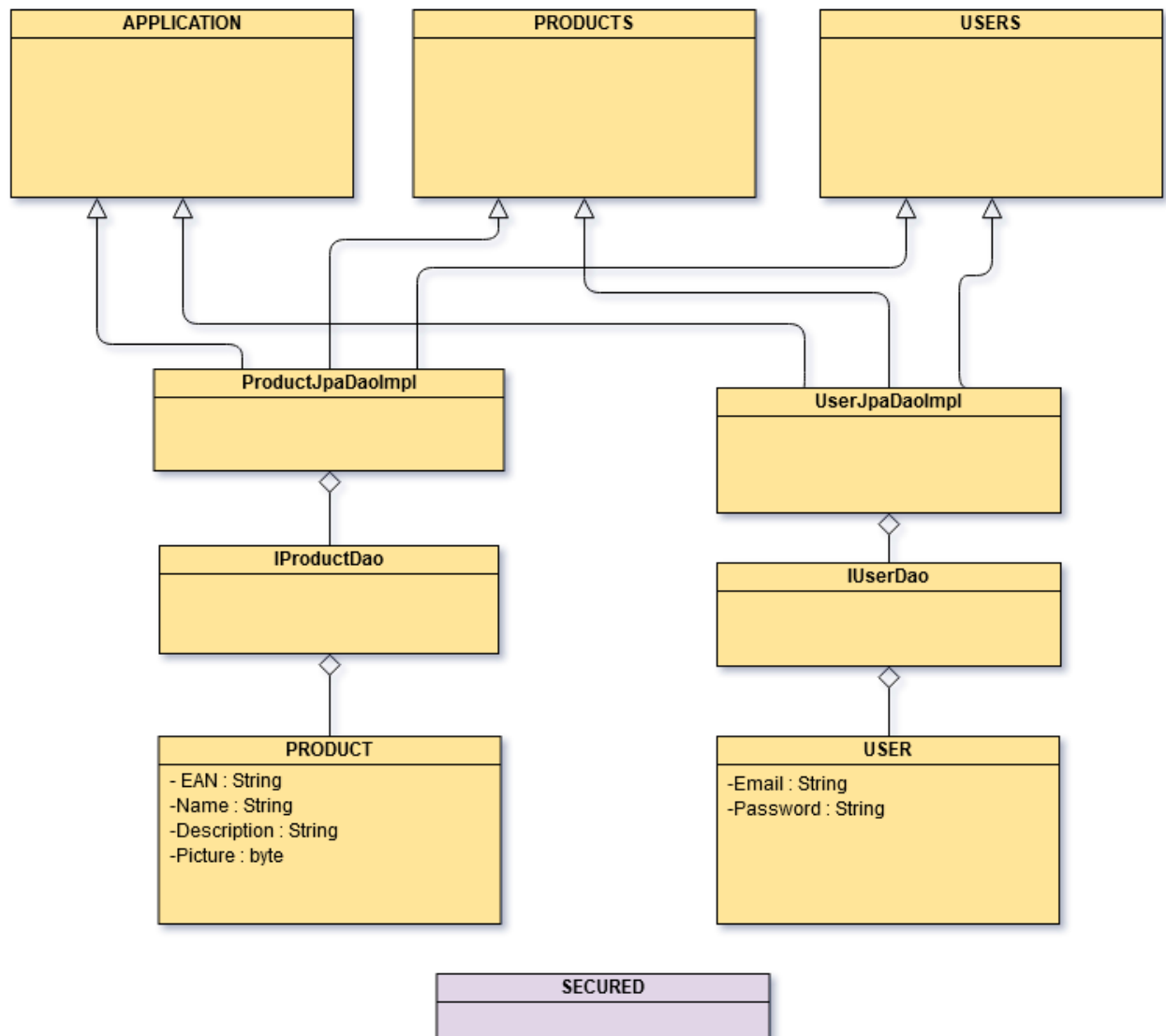
Question 11 :

Les produits sont contenue dans des liste, l'avantage des listes est que leur taille est infinie contrairement au tableau, les vecteurs quand a eux sont beaucoup plus complexe à mettre en place.

Question 12 :

C'est comme un accusé de réception, il retourne un message pour dire que ça a été ajouté avec succès.

Question 13 (UML) :



Question 13 :

```
GET /products/:ean controllers.Products.details(
  ean: String)
```

L'utilité de / :ean dans listing 4 sert à aller chercher le code barre dans le dossier products.

Question 14 :

On utilise **GET** pour obtenir des données, et **POST** pour transmettre des données.

Question 15 :

```
1 DELETE /products/:ean controllers.Products.delete(ean  
   : String)
```

Cette route supprime le code barre dans le dossier products.

Question 16 :

Le fichier « message » dans conf sert à nous renvoyer des erreurs :

```
error.form=Please correct the form below  
error.form.file=Error reading input file  
error.form.pass=Invalid password or user  
success.add=Successfully added %s  
error.invalid.ean=Ean should be a 13 digits number  
error.prod.notExist=Product %s does not exist  
error.user.exist=The user with the same email already exist
```

```
error.invalid.ean=Ean doit être un nombre de 13 chiffres
```

Question 17 :

Le dossier dao regroupe les interfaces (structures qui regroupe plusieurs données ensembles)

