

# BoVoyages - projet 1

## Étude de la couche métier

### Objectifs

- modélisation de la couche métier
- codage de la couche métier
  - mise en place des tests (JUnit) et des mocks

### A faire

- lecture du cahier des charges et du diagrammes UML
  - localisation des classes métiers et des interfaces
    - identification de ce que vous pouvez coder
    - identification des parties qui seront "mockées"
      - vous n'avez pas encore abordé la couche DAO
- créez et structurez votre projet
  - packages
  - répertoires
- codez la couche métier (cf. diagramme de classes) et les tests unitaires
  - les objets métier (BO)
  - les interfaces des services
  - les interfaces de la couche DAO
    - mise en place des mocks
- documentation

### Critères d'évaluations

Votre objectifs, outre le codage des classes, est de permettre à une autre équipe de développeurs de reprendre votre projet pour le compléter et/ou le lier à une autre couche, par exemple l'équipe front-end s'appuiera sur votre projet pour créer la couche de présentation.

- Structure du projet (5 points)
  - respect du découpage en package
  - répertoires pour les sources et les tests
- Codage (10 points)
  - le code est-il bien structuré ?
  - les noms (classes, méthodes, propriétés) permettent-ils une auto-documentation
  - les tests permettent-ils de s'assurer que l'ensemble de la couche BO fonctionne ?
    - tests unitaires
    - couverture de test
- Documentation (5 points)
  - schéma général resituant les différentes classes dans une architecture N tiers
  - qualité de la génération de la javadoc de vos classes
    - objectifs de la classe
    - ajout de documentation si l'auto-documentation ne suffit pas

cf. page suivante

## NOTES

- Le DAO utilise des méthodes pour gérer les instances des entités en base de données. Le mnémonique CRUD est souvent employé :
  - Create : crée un enregistrement en base de données
    - l'instance est sauvegardée en base
  - Read : différentes méthodes de lecture en base
    - ces méthodes créent des entités en mémoire, en fonction du résultat de la lecture
  - Update : si une instance est modifiée, l'enregistrement correspondant en base est modifié
  - Delete : l'enregistrement en base de données est supprimé
    - l'ID de l'instance est alors remis à 0 par le DAO
- ID
  - la synchronisation d'une instance en mémoire et de l'enregistrement en base est effectuée par la propriété ID
  - en base chaque enregistrement est identifié par une clé primaire unique
    - souvent, la clé primaire est un entier de type auto-incrément
  - la propriété id de l'entité doit refléter cette clé primaire
    - si l'entité n'est pas présente en base, l'id vaut 0 (zéro)
  - c'est le DAO qui maintient l'id de l'entité
    - en dehors du DAO le développeur n'a pas à toucher à cette propriété