# Package Business Object

Utilisateur

-noUtilisateur: Integer

-pseudo: String //TODO: à verifier dans BLL

-nom: String

-prenom: String

-email: String

-telephone: String

-rue: String

-codePostal: String

-ville: String

-motDePasse: String //TODO: à verifier dans BLL

-credit: Integer

-administrateur: Boolean

-listArticlesVendus : List<ArticleVendu>

-listEncheres : List<Enchere>

+Utilisateur()

+Utilisateur(String pseudo, String nom, String prenom, String email, String telephone, String rue, String codePostal, String ville, String motDePasse, Integer credit, Boolean administrateur)

+getter, setter

+ToString

Encheres

-noEncher : Integer

-dateEnchère : LocalDate

-montant\_enchère: Integer

-no\_article : Integer

-no\_utilisateur : Integer

Retrait

-rue: String

-codePostal: String

-ville: String

-noArticle: Integer

ArticleVendu

noArticle : Integer (type d’objet) [ int (type primitive) ]

nomArticle : String

description : String

dateDebutEncheres : LocalDate

dateFinEncheres : LocalDate

miseAPrix : Integer

prixVente : Integer

etatVente : String //TODO : ajouter dans la BDD

noUtilisateur: Integer

noCategorie : Integer

lieuRetrait : Retrait

+ ArticleVendu ()

+ ArticleVendu (nomArticle : String, description : String, dateDebutEncheres : LocalDate, dateFinEncheres : LocalDate, miseAPrix : Integer, prixVente : Integer, etatVente : String )

+getter, setter

+ToString

Categorie

-noCategorie : Integer  
-libelle : String

-listArticlesVendus : List<ArticleVendu>

# Package BLL

**UtilisateurManager**

**Fonctionnalités**

(CRUD: Create, Read, Update and Delete)

+ajouter un utilisateur dans la base de données

+afficher les utilisateurs dans la base de données

+afficher un utilisateur dans la base de données

+mettre à jour l’utilisateur

+supprimer des utilisateurs

**Contraintes :**

+ Le pseudo doit être unique sur toute la plateforme

+ Le mail doit être unique sur toute la plateforme

+ Le pseudo n’accepte que des caractères alphanumériques

+ Le code postal ne contient que les chiffres

+ le nom, prénom, ville ne contiennent pas de chiffre

+ le mail doit contenir @, .

+ le crédit ne peut pas être négatif

+ tous les attributs sont obligatoires sauf téléphone et noUtilisateur

**ArticlesVendusManager**

**Fonctionnalités**

(CRUD: Create, Read, Update and Delete)

+ajouter un article dans la base de données

+afficher les articles dans la base de données

+afficher un article dans la base de données

+mettre à jour l’article

+supprimer des articles

**Contraintes :**

+ Le pseudo doit être unique sur toute la plateforme

+ Le mail doit être unique sur toute la plateforme

+ Le pseudo n’accepte que des caractères alphanumériques

+ Le code postal ne contient que les chiffres

+ le nom, prénom, ville ne contiennent pas de chiffre

+ le mail doit contenir @, .

+ le crédit ne peut pas être négatif

+Si pas d’adresse de retrait, l’adresse du vendeur

EnchereManager

Fonctionnalités

-Lister les enchères de tout le monde

-Lister les enchères d’un utilisateur

-Recréditer l’enchérisseur

-Vérifier si le montant de l’enchère est valable

-Faire une enchère

-Modifier une enchère

-Supprimer une enchère

Contraintes

-L’utilisateur qui vend l’article ne peut enchérir sur son article

-L’utilisateur ne peut pas proposer d’enchère supérieur à son crédit

-L’utilisateur ne peut pas proposer de montant d’enchère déjà proposé ou inférieur à l’enchère la plus importante

-L’utilisateur ne peut pas proposer d’enchère inférieur à la mise à prix

RetraitManager

Fonctionnalités

-Ajouter un retrait

-Modifier un retrait

-Supprimer un retrait

CategorieManager

Fonctionnalités

-Ajouter un retrait

-Modifier un retrait

-Supprimer un retrait

Package DAL

+Interface: UtilisateurDAO: (CRUD: Create, Read, Update and Delete) (Data Access Objet)

+UtilisateurDAOimpl implements UtilisateurDAO

+DAOFactory:

+getUtilisateurDAO: UtilisateurDAO

+connection à la base de données :

+ProviderConnection

+Interface: ArticleVenduDAO: (CRUD: Create, Read, Update and Delete) (Data Access Objet)

+ArticleVenduDAOimpl implements ArticleVenduDAO

+DAOFactory:

+getArticleVenduDAO: ArticleVenduDAO

+connection à la base de données :

+ProviderConnection

+Interface: EnchereDAO: (CRUD: Create, Read, Update and Delete) (Data Access Objet)

+EnchereDAOimpl implements EnchereDAO

+DAOFactory:

+getEnchereDAO: EnchereDAO

+connection à la base de données :

+ProviderConnection

+Interface: RetraitDAO: (CRUD: Create, Read, Update and Delete) (Data Access Objet)

+RetraitDAOimpl implements RetraitDAO

+DAOFactory:

+getRetraitDAO: RetraitDAO

+connection à la base de données :

+ProviderConnection

+Interface: CategorieDAO: (CRUD: Create, Read, Update and Delete) (Data Access Objet)

+CategorieDAOimpl implements CategorieDAO

+DAOFactory:

+getCategorieDAO: CategorieDAO

+connection à la base de données :

+ProviderConnection

-Classe IHM (Interface Humain Machine)

Création d’utilisateur

- formulaire jsp

-Le Manager affichera le message d’erreur correspondant à l’exception levée.

-Si le formulaire de creation utilisateur est validé, l’utilisateur navigue vers la page “Liste des enchères”

-En cas d’annulation, on revient vers la page d’accueil “Liste des enchères non connecté”

Création d’article

- formulaire jsp

-Le Manager affichera le message d’erreur correspondant à l’exception levée.

-Si le formulaire de creation article est validé, l’utilisateur navigue vers la page “Liste des enchères”

-Par défaut, l’adresse affiché est celle du vendeur.

-En cas d’annulation, on revient vers la page d’accueil “Liste des enchères non connecté”

Création d’enchères

- formulaire jsp

-Le Manager affichera le message d’erreur correspondant à l’exception levée.

-Si le formulaire de creation encheres est validé, l’utilisateur navigue vers la page “Liste des enchères”

-Si l’utilisateur n’est pas connecté, il ne peut faire d’enchères.

-Si l’utilisateur n’est pas connecté, il n’a pas accès au détail des enchères

-Par défaut, l’adresse affiché est celle du vendeur.

-En cas d’annulation, on revient vers la page d’accueil “Liste des enchères”

- ihm pour Categorie et Retrait est dans ihm de ArticleVendu.

Nommer des Packages

fr.ENI.HiddenFigures.NomProjet.NomClasse

fr.ENI.HiddenFigures.Enchere.ihm