

# Service API Design Lab.

## TeamForce

Étudiant :

- BENATHMANE Ayoub, benathmane.ab@gmail.com, Master 2 IFI parcours WEB

## I. Implémentation

La mise en œuvre de RPC expose trois services: (i) un service **CustomerRPC** dédié aux actions liées aux clients (ii) un service **EmployeeRPC** regroupe les opérations liées à la manipulation de clients et des commandes et (iii) un service **ManagerRPC** s'occupe de tous ce qu'est gestion des catalogues et stock.

CustomerRPC WSDL : *http://localhost:8080/webservices/TeamForce-Customer-RPC?wsdl*  
(rpc.CustomerRPC.java)

EmployeeRPC WSDL : *http://localhost:8080/webservices/TeamForce-Employee-RPC?wsdl*  
(rpc.EmployeeRPC.java)

ManagerRPC WSDL : *http://localhost:8080/webservices/TeamForce-Manager-RPC?wsdl*  
(rpc.ManagerRPC.java)

Les opérations suivantes sont exposés par les trois services.

- Les services destinés **aux clients** :
  - `create_account` : Créer un nouveau compte client et retourne l'objet Client crée.
  - `Assign_addresses_to_customer` : Modifier les 2 adresses liées au Client et retourne l'objet client modifié.
  - `create_order` : Créer une commande et la liée au Client, retourne l'objet Ordre.
  - `add_event_to_order` : Ajouter des commentaires à une commande, retourne un nouvel objet ordre.
- Les services destinés **aux employés** :
  - `getAllAddressesForCustomer` : Retourne la liste des adresses pour un client donné.
  - `getAllOrdersForCustomer` : Retourne la liste des commandes pour un client donné.
  - `getAllCustomer` : Retourne la liste de tous les clients.
  - `modify_statut` : Modifier le statut d'une commande, prend en paramètre l'id de la commande et le nouveau statut.
  - `modify_step` : Modifier l'étape d'une commande, prend en paramètre l'id de la commande et la nouvelle étape.
  - `findCustomerbyphone` : Rechercher un client à partir de son numéro de téléphone, retourne un objet Client.
- Les services destinés **aux gestionnaires** :
  - `create_product` : Créer un produit
  - `create_catalogue` : Créer un Catalogue
  - `assign_product_to_categorie` : Associer un produit à une catégorie
  - `assign_product_to_catalogue` : Associer un produit à une catalogue

## II. The business Object

Le projet est organisé selon la structure de répertoire suivante.

- **business** : Contient les Data model gérée par le projet TeamForce.
- **rpc** : une mise en œuvre du service de calculatrice à l'aide du paradigme de la RPC, au-dessus de SOAP.

Dans le dossier "business", nous trouvons les classes java :

- **Customer** : Contient le constructeur "Customer" et les différentes getters et setters.
- **Order** : Contient le constructeur "Ordre" et les différentes getters et setters.
- **Product** : Contient le constructeur "Product" et les différentes getters et setters.
- **Catalogue** : Contient le constructeur "Catalogue" et les différentes getters et setters.
- **Adresse** : Contient le constructeur "Adresse" et les différentes getters et setters.
- **Event** : Contient le constructeur "Event" et les différentes getters et setters.
- **Categorie** : Contient le constructeur "Catégorie" et les différentes getters et setters.
- **Status** : Classe de type énumération contient un liste des Status déjà préfini.
- **Steps** : Classe de type énumération contient un liste des Etats déjà préfini.
- **DataAccessObjectCstm** : Classe contenant différentes méthodes et fonctions permettant la manipulation des classes citées ci-dessus, et aussi un exemple de données (initialisation d'une liste des objets : Customer, Ordre, Product,... liées entre eux par référence) permettant le test de nos web services.

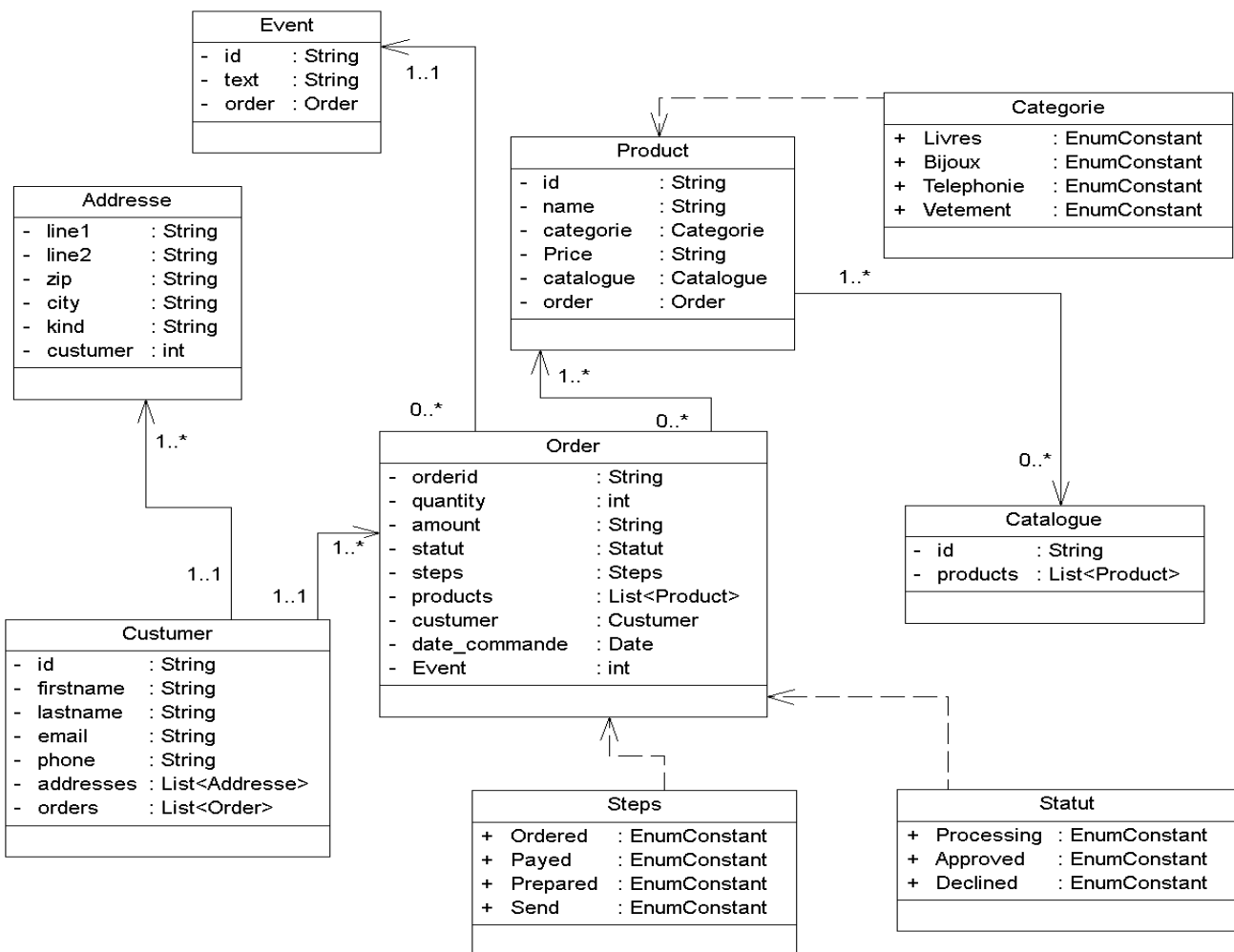
Dans le dossier "rpc", nous trouvons les classes java liées aux web services :

- **CustomerRPC** : Contient les différents service liés aux Clients.
- **ManagerRPC** : Contient les différents service liés aux Managers.
- **EmployeeRPC** : Contient les différents service liés aux Employées.
- **UnknownCustomerFault** : Gère les erreurs liées à un client qui n'existe pas.
- **UnknownOrderFault** : Gère les erreurs liées à une commande qui n'existe pas.

L'application TeamForce vise trois catégories d'utilisateurs :

- **Customer** : C'est le client de notre application qui peut s'inscrire, créer ou modifier ses adresses, et créer des commande.
- **Manager** : C'est le manager du stock qui peut créer des catalogues et des produits, assigner de ces derniers a des catégories et des catalogues.
- **Employee** : Gère tout ce qu'est Ordre (changer statut, suivi, etc ...) et gestion des Clients.

### III. Conception



Relation		Explication
Customer	Adresse	Le Client contient une liste d'adresses (2 : personnelle et professionnelle), il peut être livré dans une des deux, par contre l'adresse est liée qu'un ou plusieurs clients dans le cas même membre de famille ayant la même adresse.
Customer	Ordre	Un client à une liste des commandes, la commande est liée à un seul client.
Ordre	Product	La commande contient une liste de produit (1..*), et chaque produit peut être lié a une ou plusieurs commande.
Ordre	Event	Une commande peut avoir une liste des events (une ou plusieurs notes attribuées à chaque commande), et cette event est liée à une seule commande.
Ordre	Steps	La commande contient une variable de type énumération "Steps" dans les valeurs sont déjà définies.
Ordre	Status	La commande contient une variable de type énumération "Status" dans les valeurs sont déjà définies.

Product	Categorie	Le produit contient une variable de type énumération "Categorie" dans les valeurs sont déjà définies.
Product	Catalogue	Un produit appartient à un ou plusieurs catalogue, par contre le catalogue peut contenir 0 ou plusieurs produits, (N.B : 0 dans le cas du scenario 3).