

**CRÉEZ UN WEB SERVICE EXPOSANT UNE API**

**PROJET 7**

**DÉVELOPPEUR D'APPLICATION - PHP / SYMFONY**

# Sommaire

1	Introduction .....	3
1.1	Contexte.....	3
2	Définition des acteurs .....	4
2.1	Acteurs principaux.....	4
2.2	Acteurs secondaire.....	4
3	Diagramme de contexte.....	5
4	Diagramme de Package .....	6
5	Les diagrammes de cas d'utilisation.....	6
5.1	Gestion d'utilisateur .....	8
5.2	Gestion des Clients .....	8
6	Diagrammes de séquence .....	9
6.1	Ajout d'un utilisateur (lié à un client) .....	9
6.2	Suppression d'un user.....	10
6.3	Mise à jour d'un user .....	10
6.4	Lister des Utilisateurs .....	11
6.5	Lecture d'un utilisateur.....	11
6.6	Lister les produits .....	11
6.7	Lecture d'un produit .....	12
7	Diagramme de classes .....	13
8	Modèle physique de données.....	14
9	Conclusion.....	15
10	Annexe.....	15

# 1 Introduction

---

## 1.1 Contexte

BileMo est une entreprise offrant toute une sélection de téléphones mobiles haut de gamme.

Le business modèle de BileMo n'est pas de vendre directement ses produits sur le site web, mais de fournir à toutes les plateformes qui le souhaitent l'accès au catalogue via une API (Application Programming Interface). Il s'agit donc de vente exclusivement en B2B (business to business).

Considéré comme un développeur PHP/Symfony, je suis en charge du développement de la vitrine de téléphones mobiles de l'entreprise BileMo.

Il va falloir que j'expose une API pour que les applications des autres plateformes web puissent effectuer des opérations.

## Description du besoin

Le premier client a enfin signé un contrat de partenariat avec BileMo ! C'est le branle-bas de combat pour répondre aux besoins de ce premier client qui va permettre de mettre en place l'ensemble des API et de les éprouver tout de suite.

Après une réunion dense avec le client, il a été identifié un certain nombre d'informations. Il doit être possible de :

- Consulter la liste des produits BileMo ;
- Consulter les détails d'un produit BileMo ;
- Consulter la liste des utilisateurs inscrits liés à un client sur le site web ;
- Consulter le détail d'un utilisateur inscrit lié à un client ;
- Ajouter un nouvel utilisateur lié à un client ;
- Supprimer un utilisateur ajouté par un client.

Seuls les clients référencés peuvent accéder aux API. Les clients de l'API doivent être authentifiés via OAuth ou JWT.

J'ai le choix entre mettre en place un serveur OAuth et y faire appel (en utilisant le [FOSOAuthServerBundle](#)), et utiliser Facebook, Google ou LinkedIn.




Si je décide d'utiliser JWT, il vous faudra que je [vérifie la validité du token](#) ; l'usage d'une librairie est autorisé.

## 2 Définition des acteurs

---

Dans le cadre de ma réflexion j'ai pu définir les acteurs suivant

### 2.1 Acteurs principaux

 Administrateur	Personne avec tous les privilèges
 Client	Personne avec droit de consultation de produits et de gestion de ses propres utilisateurs
 Utilisateur	Personne avec droit de consultation de produits

### 2.2 Acteurs secondaire

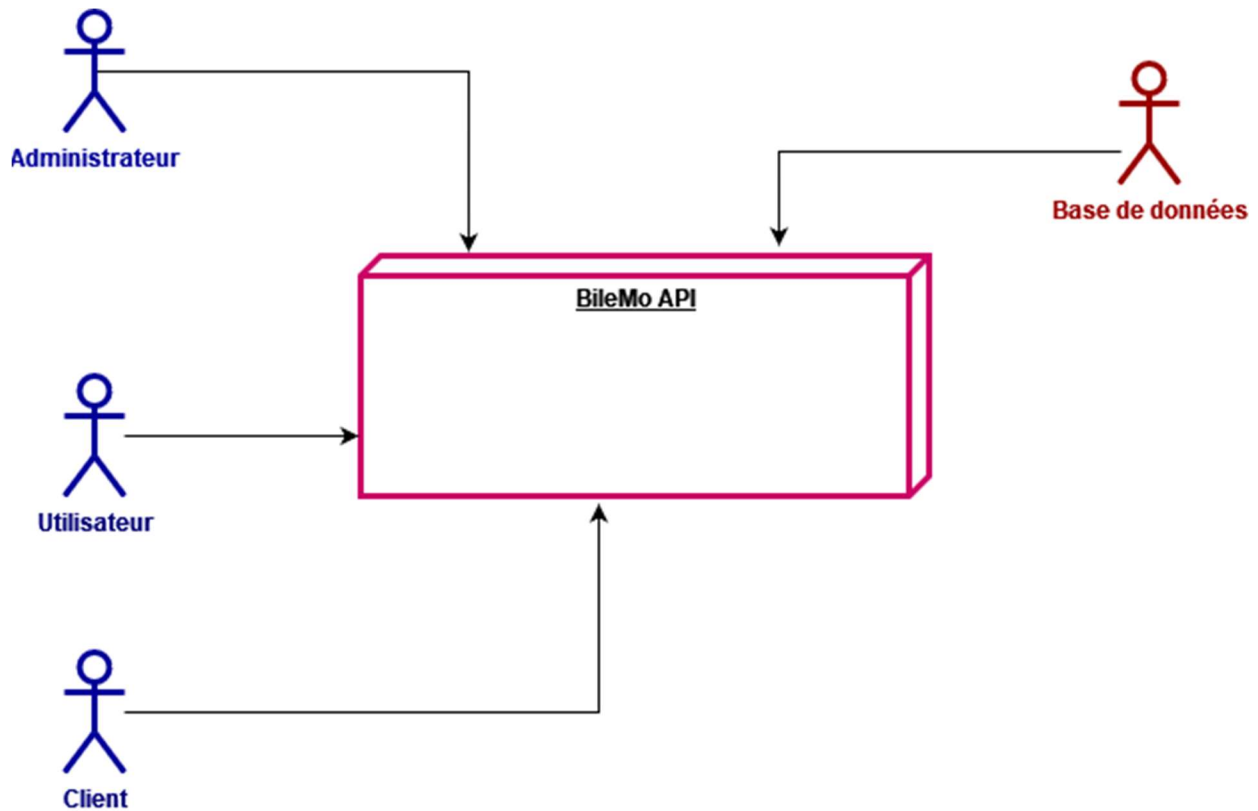


« Système » permettant la manipulation des données

### 3 Diagramme de contexte

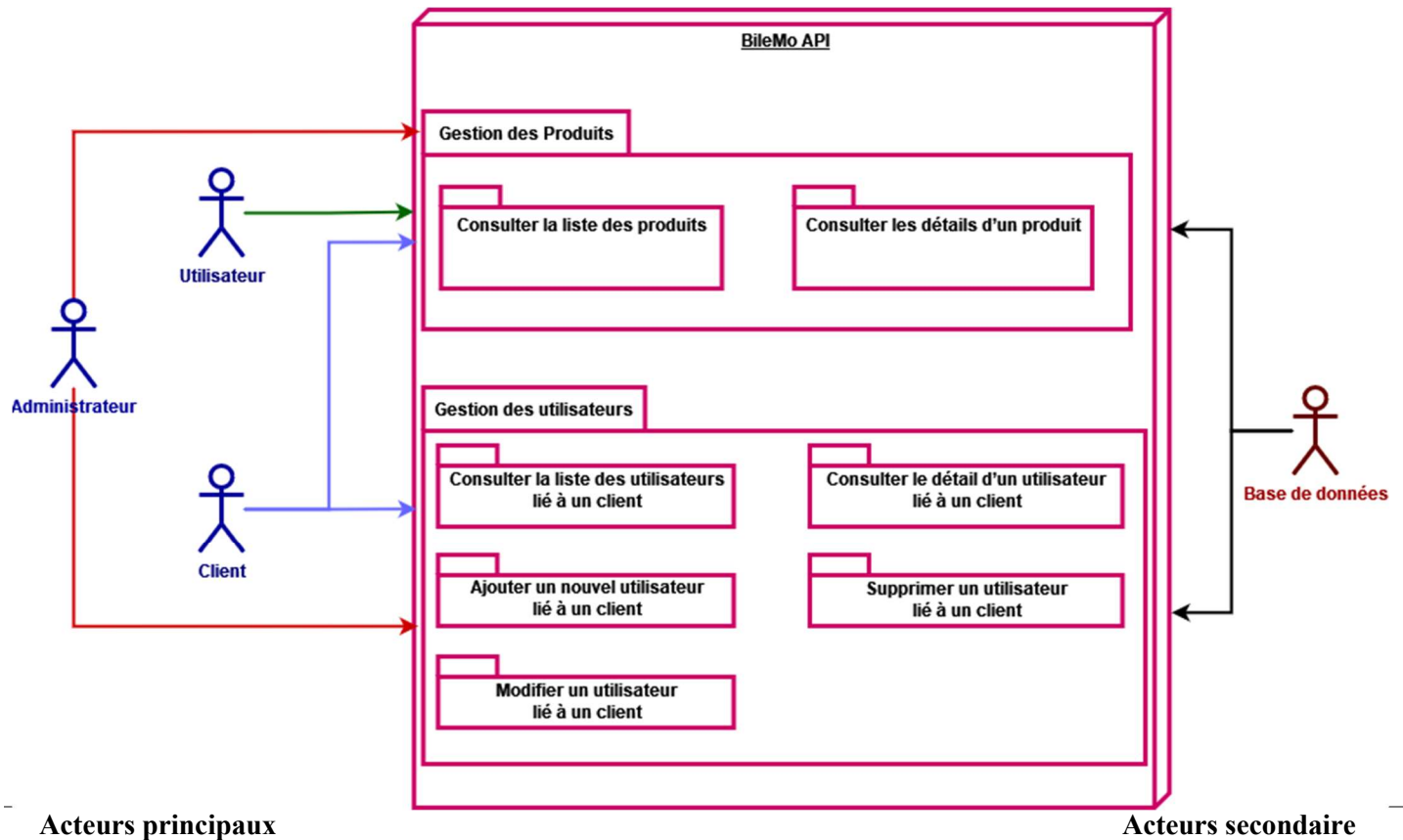
---

Le modèle de contexte met en évidence, à un niveau très général, les flux d'information échangés entre les acteurs.



**\* Le système est notre application(API) qui est en cours de développement. \***

## 4 Diagramme de Package



En avançant dans notre analyse, nous pouvons maintenant identifier les familles de fonctionnalités du système soit :

Gestion des Produits :

- Lister les produits
- Lecture d'un produit

Gestion des utilisateurs

- Création d'utilisateurs
- Lecture d'un utilisateur
- Lister les d'utilisateurs
- Mise à jour d'un utilisateur
- Suppression d'un utilisateur

Nous remarquons les droits Acteurs suivants :

Administrateur : tous les droits

Clients : Tous les droits sur ses propres utilisateurs

Utilisateur (lié à un client) : Lecture(seule) de produits.

## 5 Les diagrammes de cas d'utilisation

Les familles de fonctionnalités du système citées précédemment me permettent de lister les diagrammes de cas suivant

J'utiliserai pour la suite de mon travail les termes suivants :

- 1- CRUD : du terme anglais (CREATE READ UPDATE DELETE) signifiant respectivement (Lire Ecrire Mettre à jour et Suppression)

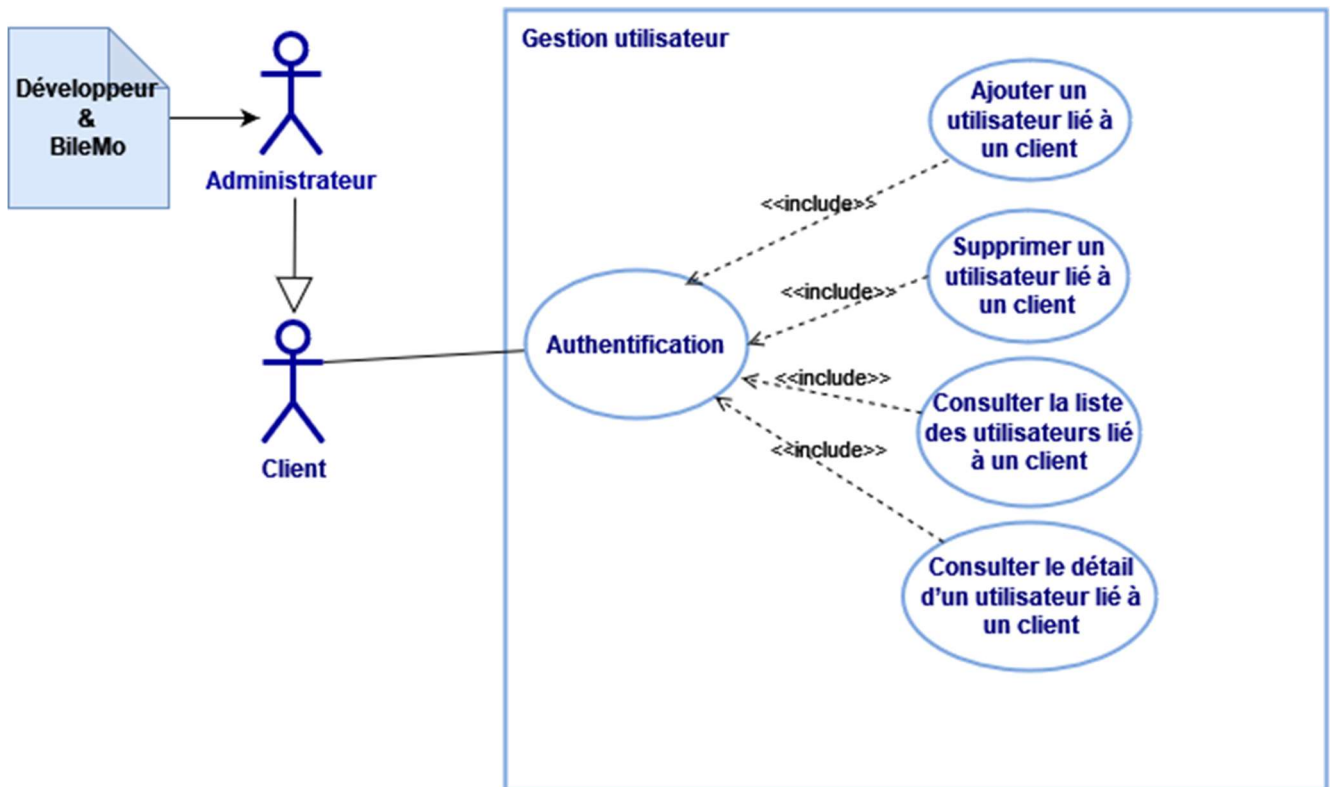
#### Partie Client (CRUD)

- Création d'un utilisateur
- Lecture d'un utilisateur
- Mise à jour d'utilisateurs
- Suppression d'un utilisateur
- Lecture d'un produit
- Lecture de produits

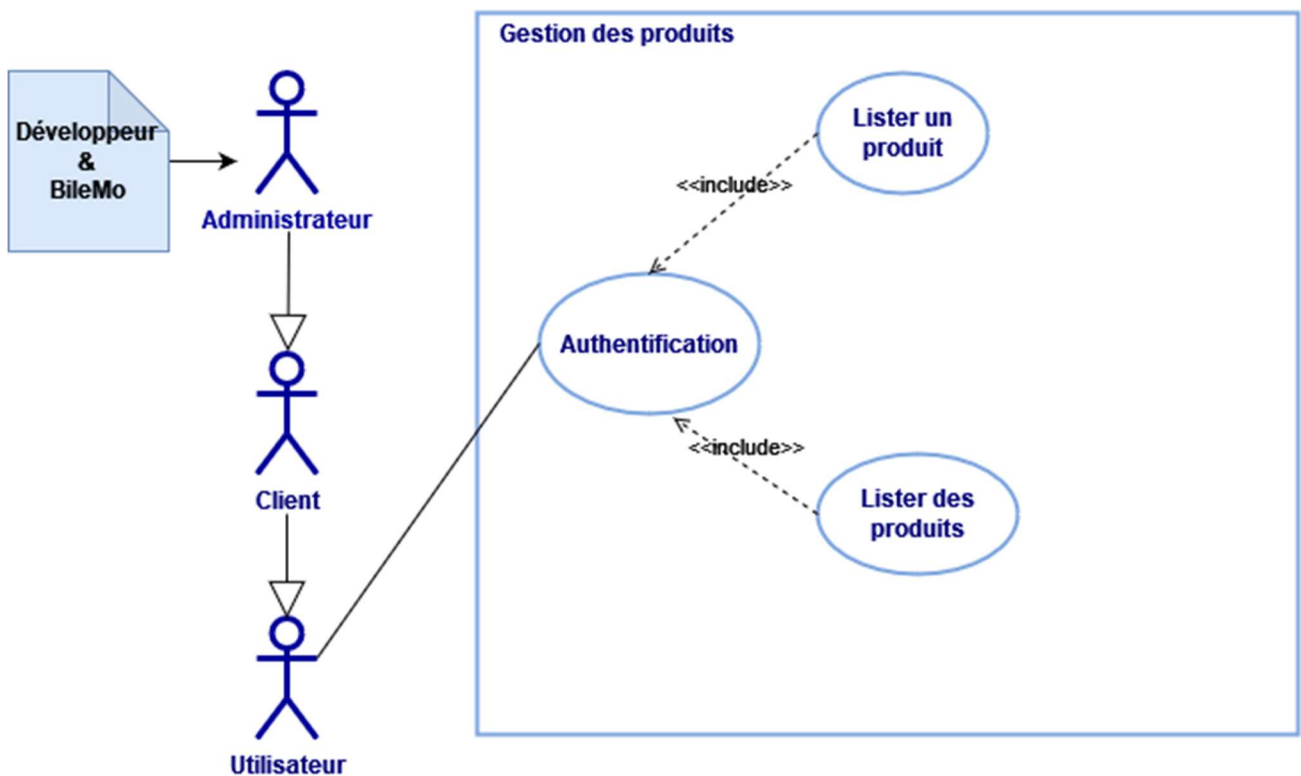
#### Partie Utilisateurs (lié à un client)

- Lecture de produits
- Lecture d'un produit

## 5.1 Gestion d'utilisateur



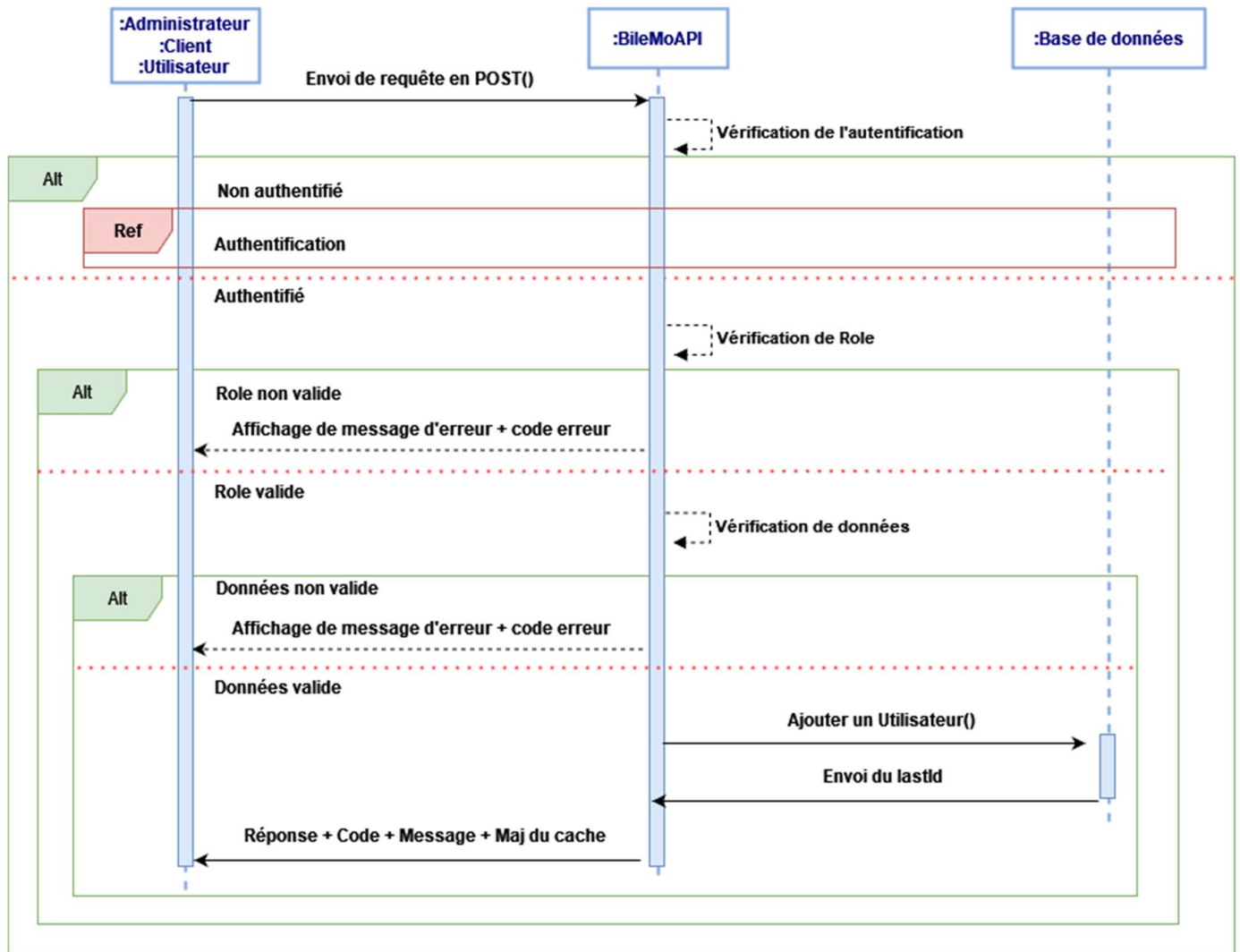
## 5.2 Gestion des Produits



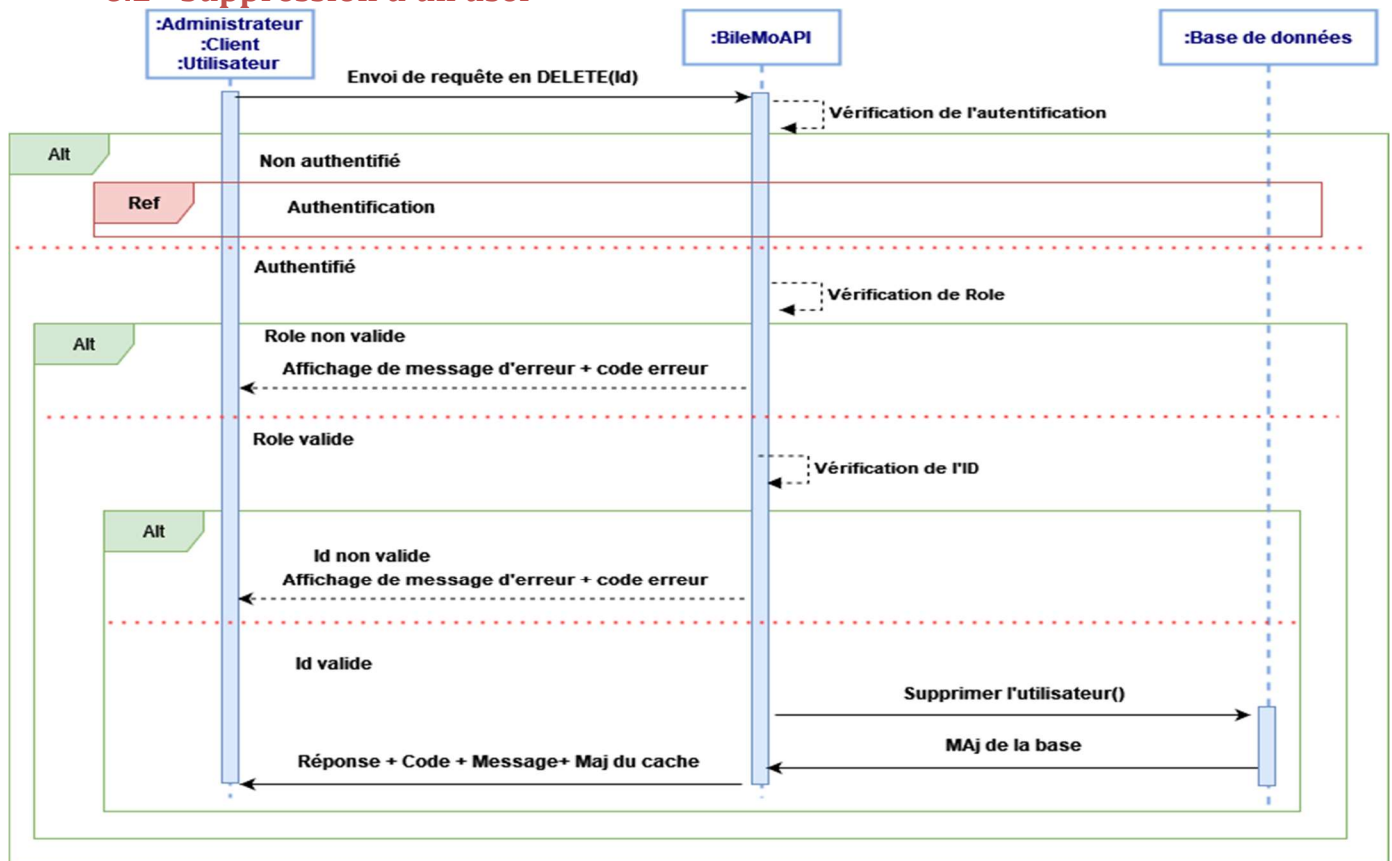


## 6 Diagrammes de séquence

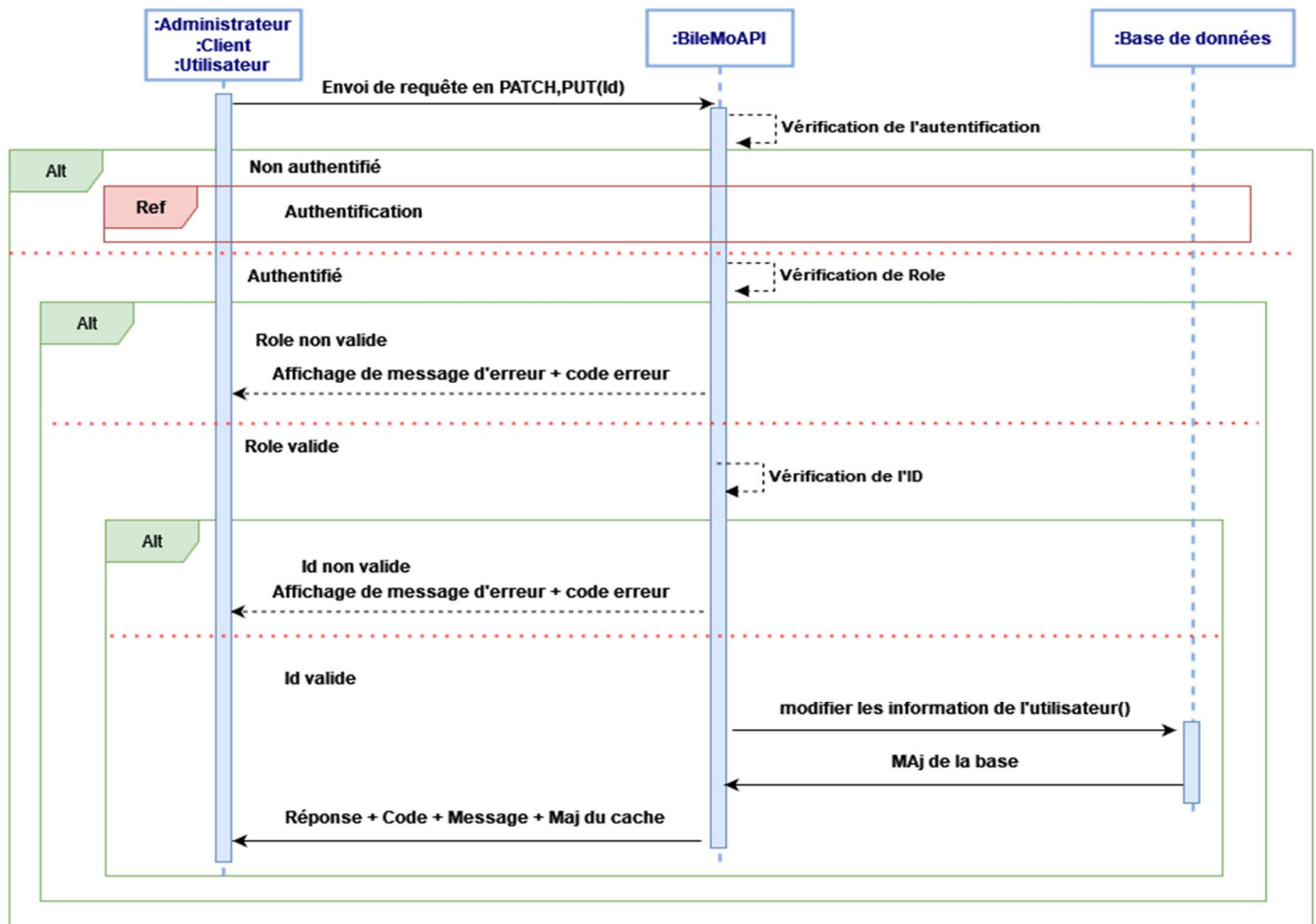
### 6.1 Ajout d'un utilisateur



## 6.2 Suppression d'un user

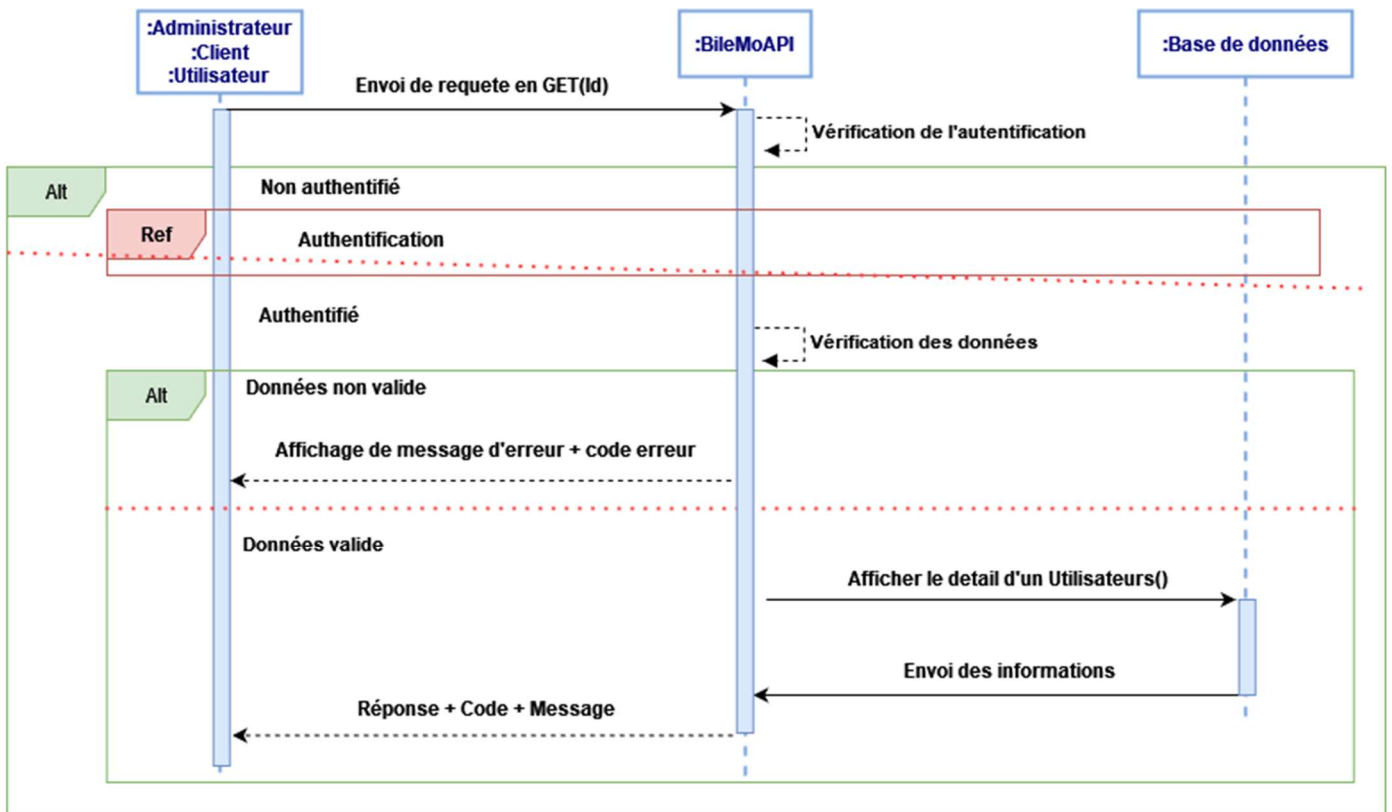


## 6.3 Mise à jour d'un utilisateur

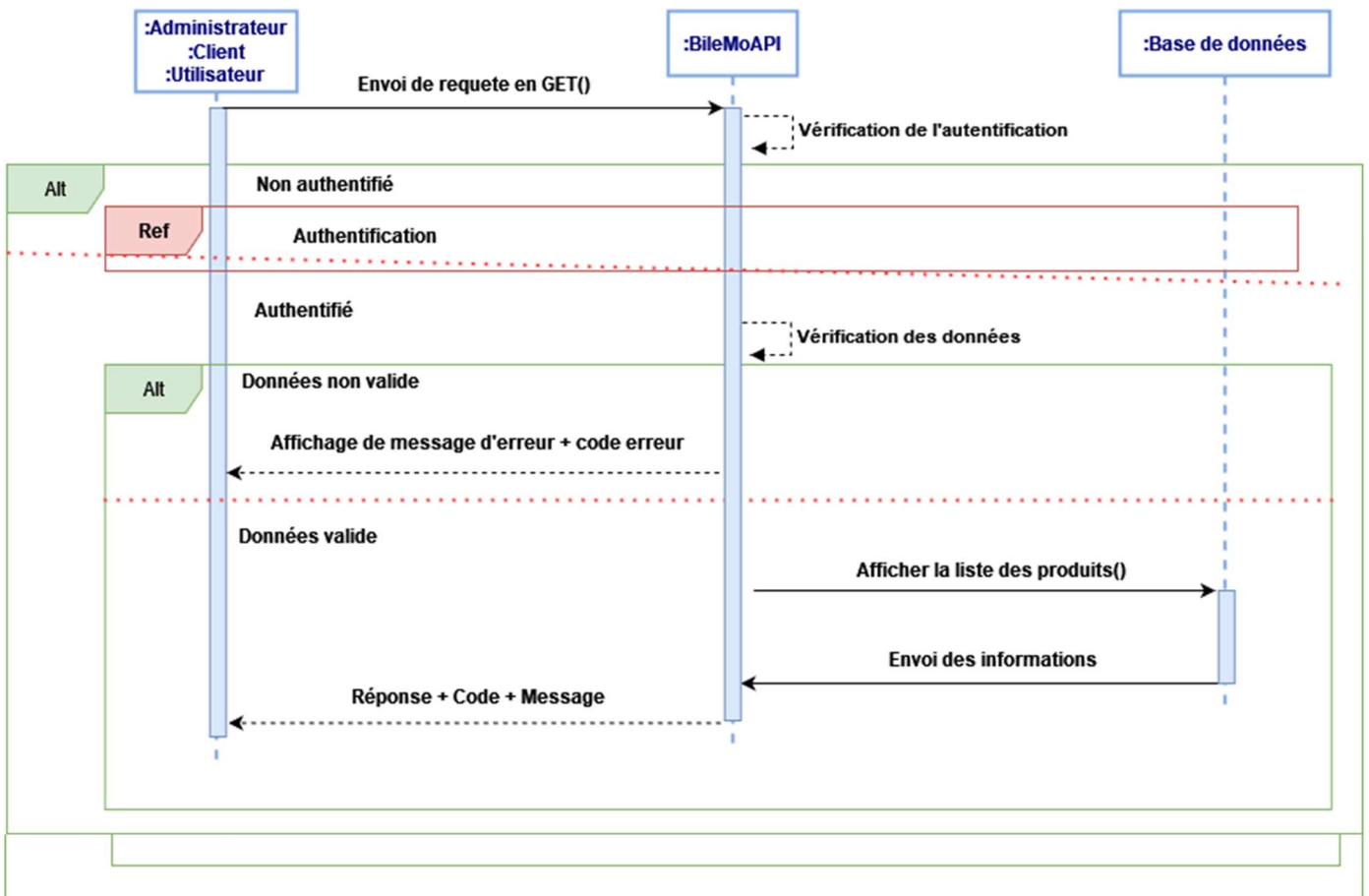


## 6.4 Lister des utilisateurs

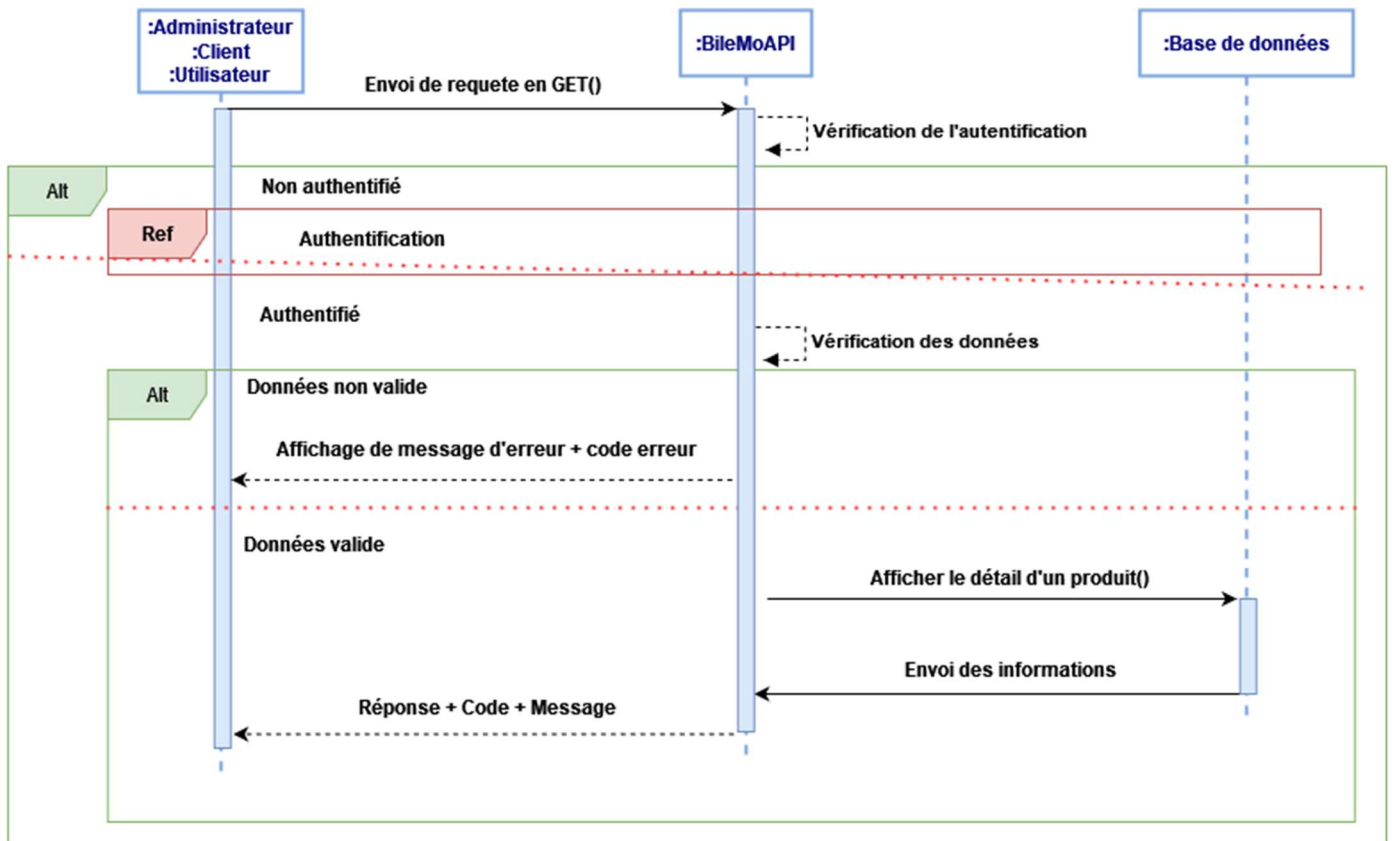
## 6.5 Lecture d'un utilisateur



## 6.6 Lister les produits

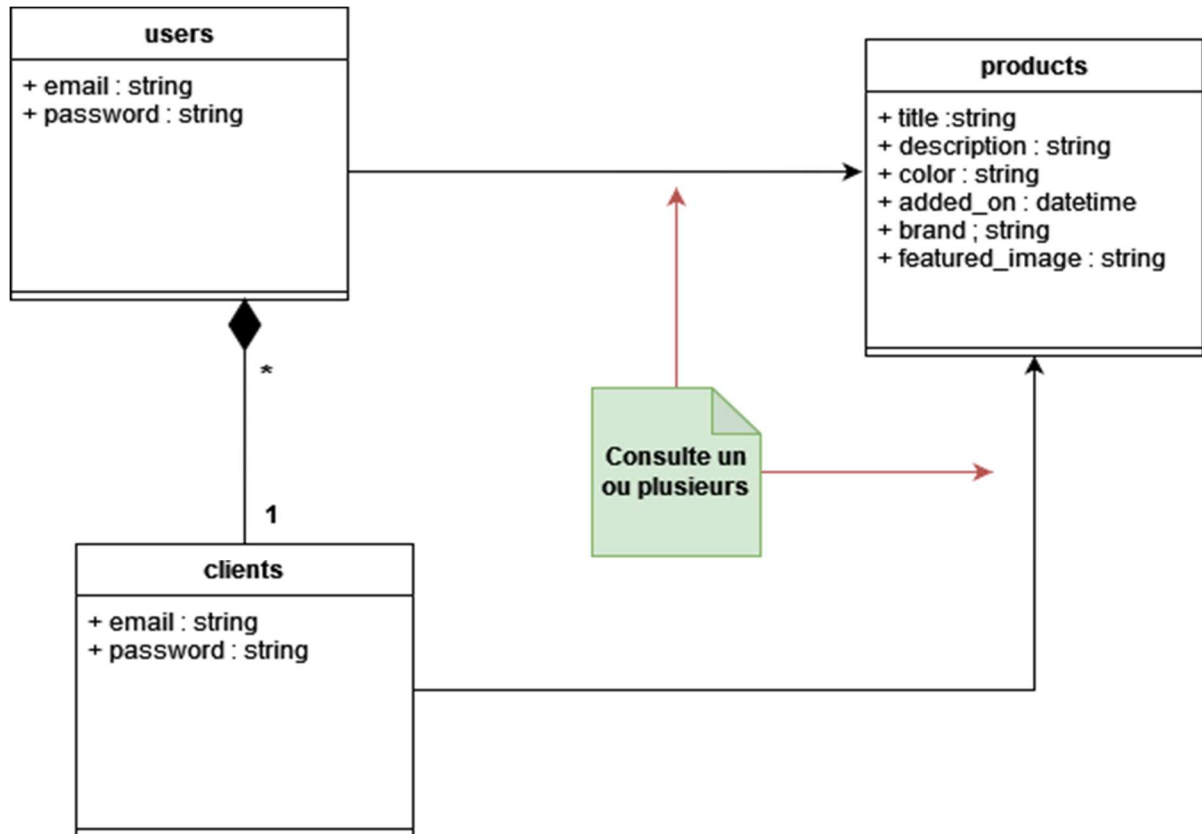


## 6.7 Lecture d'un produit

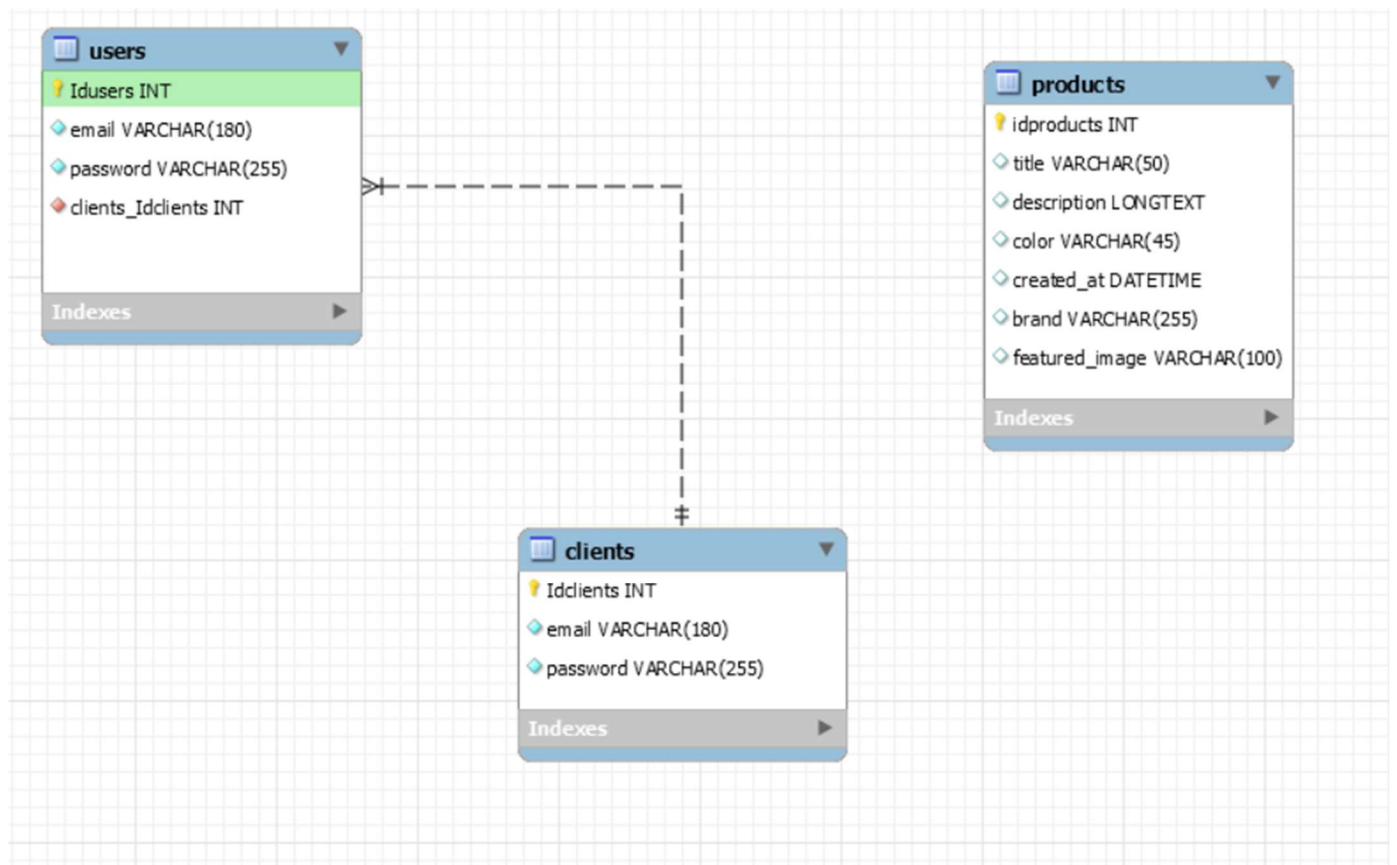


## 7 Diagramme de classes

---



## 8 Modèle physique de données



## 9 Conclusion

---

Certaines zones d'ombres resteront à éclaircir, néanmoins ce document pourra être considéré comme un préambule tant aux demandes formulées par notre client qu'aux éventuelles idées d'améliorations pouvant survenir

Un échange constructif est donc plus que demandé avant de passer à la phase **conception**

De ce fait, j'ai modélisé une **base de données « MySQL » avec un jeu de données de démonstration** tel que demandé par le client à des fins de démonstration

## 10 Annexe

---

- 1- <https://app.diagrams.net> a été utilisé dans le but de créer mes diagrammes de cas, les diagrammes de séquences ainsi que le diagramme de classe
- 2- **MySQL Workbench** a été utilisé dans le but de créer mon MPD « Model Physique de données »
- 3- Mes sources seront dans un dossier nommé « **P7\_Mecili\_Nabil** » dans lequel il y'aura quatre (04) sous dossiers nommés respectivement « **P7\_01\_BileMoAPI\_CodeSource** » pour le code « **P7\_02\_Diagrammes** » ainsi que « **P7\_03\_Presentation** » et « **P7\_04\_Liens** » pour ma base de donnée « » et quelques requêtes SQL à titre d'exemple dans des fichiers « \*.sql »