

# CAHIER D'ANALYSES

## APPLICATION DE SUIVI DES STOCKS ET COMMANDES POUR UNE LIBRAIRIE EN LIGNE



**Nom et Prénoms des étudiants :**

- ❖ **BASSAMA ATOUME STEEVE LOIC**
- ❖ **BILONG KEREN**
- ❖ **NGUEPI ZAMBO LIND**
- ❖ **YASMINE BOUBA**

**Enseignant : M. ATANGANA FUDA**





Année Académique : 2024- 2025


## **Introduction**

Le cahier d'analyse spécifie les besoins fonctionnels et non fonctionnels qui représentent la phase dans le cycle de développement d'un logiciel. C'est ainsi que dans notre cadre des Travaux pratiques dont le thème est conception et réalisation d'une application de suivi des stocks et commande pour une librairie en ligne, nous allons décrire les besoins ainsi que quelques diagrammes permettant de bien spécifier les besoins fondamentaux de l'utilisateur de l'application.

### **1. Présentation générale du projet**

L'application qui sera conçue pour la gestion des commandes et suivi des stocks pour une librairie en ligne est une application client – serveur qui va répondre aux attentes des gérants et des lecteurs passionnés. Elle va répondre d'une manière ou d'une autre aux besoins réels et sera facile à l'utilisation. Cette application comprendra comme modules :

-  **Module d'Inscription** : Permettra aux clients et aux administrateurs de s'inscrire sur la plateforme.
-  **Module Gestion des articles** : Organisera les articles en fonction des catégories (scolaires, éducatifs, loisirs, autres...) dans autres on note les romans, le théâtre, la poésie).
-  **Module de Gestion des clients** : Gèrera tous les clients qui se sont enregistrés dans la plateforme.
-  **Module de Gestion des stocks en temps réel** : Permettra aux administrateurs d'être au courant sur la quantité d'article disponible en stock.

 **Tableau de Bord Administratif** : Fournira aux administrateurs une vue d'ensemble des activités de la librairie (rapport satisfaction client, rapport des ventes).

## **2. Contexte du projet**

Une librairie en ligne en pleine croissance rencontre des difficultés dans la gestion de ses stocks et commandes. Les ruptures de stock fréquentes et les erreurs dans les livraisons nuisent à sa réputation. Le gérant souhaite une application pour gérer les stocks en temps réel, alerter sur les seuils critiques et optimiser les réapprovisionnements.

## **3. Cas d'utilisation du système**

### **Les acteurs et leurs rôles**

Un acteur représente est une entité qui définit le rôle joué par un utilisateur ou par un système qui interagit avec le système modélisé. Généralement un acteur est représenté par un bonhomme en fil de fer (en anglais stick man).

#### **3.2.1. Acteurs internes**

Un acteur interne est un acteur faisant partie du système, jouant un rôle à l'exécution des services que le système propose.

Dans notre cas nous n'avons qu'un seul acteur interne à savoir **l'administrateur** qui sera celui chargé de la supervision de tout le système

#### **3.2.2. Acteurs externes**

L'acteur externe est un acteur ne faisant pas partie du système et bénéficiant des différents services que le système propose.

Nous n'avons qu'un seul acteur externe dans notre application à savoir **le client** qui sera chargé de passer des commandes.

## DIAGRAMME DE CAS D'UTILISATION

Model1::UseCaseDiagram1

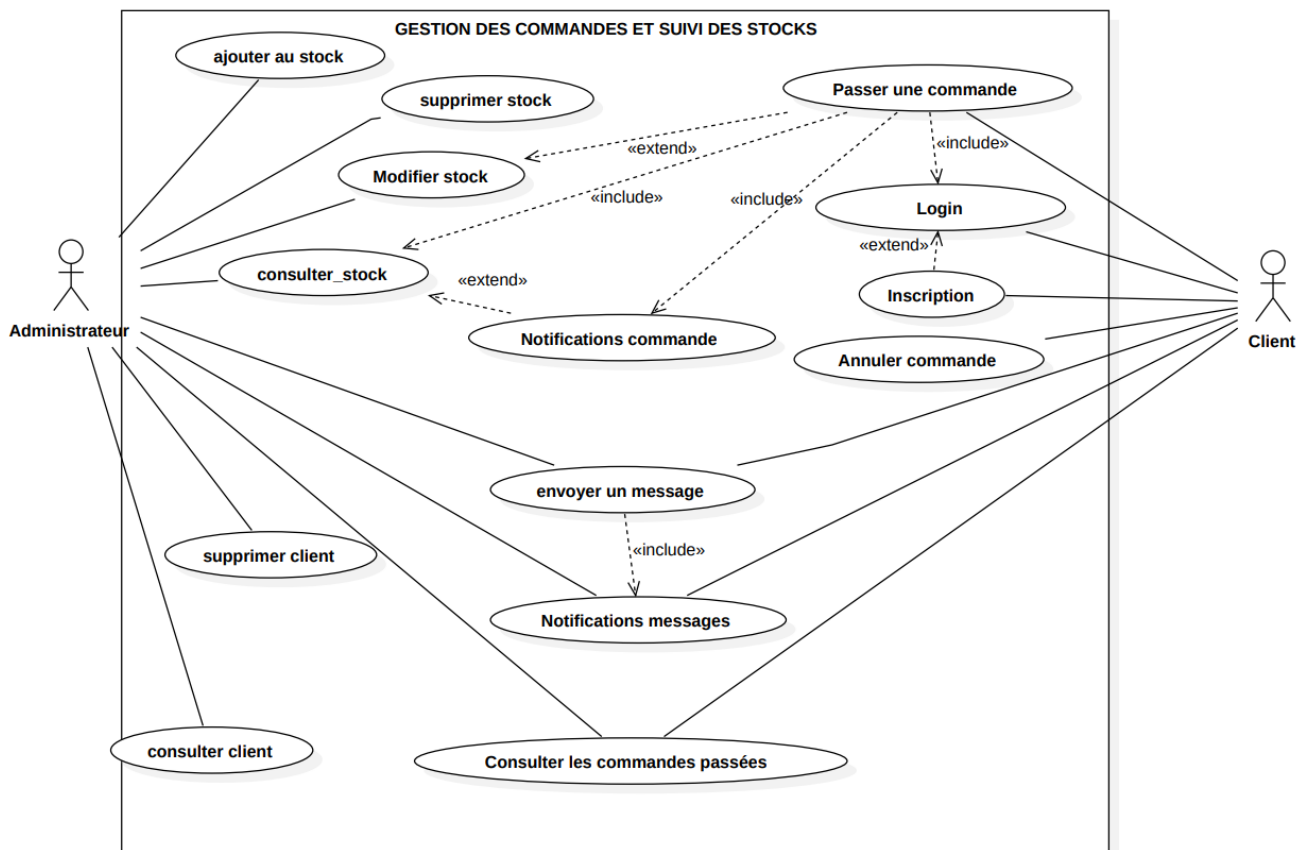


Figure 1: Diagramme des cas d'utilisation général

## EXPLICATION DU DIAGRAMME DE CAS D'UTILISATION

### Acteurs

**Administrateur :** Cet acteur a des responsabilités étendues, notamment la gestion des stocks et des clients.

Client : Cet acteur interagit principalement avec le système pour passer des commandes, consulter des informations, et recevoir des notifications.

### **Cas d'utilisation**

Les cas d'utilisation représentent les différentes fonctionnalités que le système offre aux acteurs. Voici les principaux cas d'utilisation identifiés :

- **Pour le Client**

Passer une commande : Permet au client de créer une nouvelle commande.

Annuler commande : Donne au client la possibilité d'annuler une commande existante.

Consulter les commandes passées : Permet au client de voir l'historique de ses commandes.

Envoyer un message : Permet au client de communiquer avec le système ou l'administrateur.

Notifications commande : Le client reçoit des alertes concernant l'état de ses commandes.

Notifications messages : Le client est informé des nouveaux messages reçus.

- **Pour l'Administrateur**

Ajouter au stock : Permet à l'administrateur d'ajouter de nouveaux articles au stock.

Modifier stock : L'administrateur peut modifier les informations sur les articles en stock.

Supprimer stock : Permet la suppression d'articles du stock.

Inscription : L'administrateur peut inscrire de nouveaux clients dans le système.

Login : Accès au système pour l'administrateur.

Consulter client : Permet à l'administrateur de voir les informations sur les clients.

Supprimer client : Donne à l'administrateur la capacité de supprimer un client du système.

## **Relations**

Le diagramme inclut également des relations entre les cas d'utilisation :

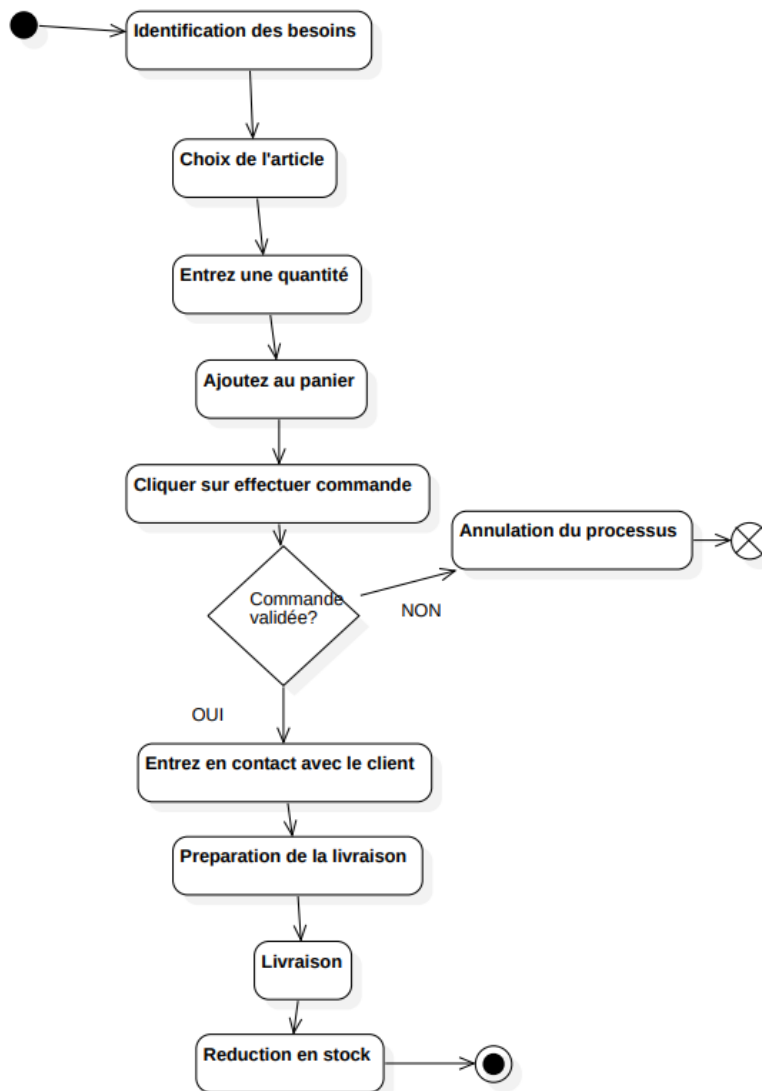
**«include»** : Indique qu'un cas d'utilisation inclut toujours un autre cas. Par exemple, "Passer une commande" pourrait inclure "Login" pour s'assurer que le client est authentifié avant de passer une commande.

**«extend»** : Indique qu'un cas d'utilisation peut être étendu par un autre cas dans certaines conditions. Cela permet d'ajouter des fonctionnalités optionnelles sans modifier le cas principal

## **DIAGRAMME D'ACTIVITE**

Le diagramme d'activité en UML (Unified Modeling Language) est un outil de modélisation comportementale qui permet de représenter les flux de travail et les processus au sein d'un système. Nous allons le représenter pour 03 cas d'utilisation.

- **PASSER UNE COMMANDE**



## **EXPLICATION**

### Choix de l'article

L'utilisateur sélectionne un ou plusieurs articles qu'il souhaite acheter.

### Entrer une quantité

Après avoir choisi les articles, l'utilisateur saisit la quantité désirée pour chaque article.

### Ajouter au panier

Les articles et les quantités sont ensuite ajoutés au panier d'achat.

#### Cliquer sur effectuer commande

L'utilisateur passe à l'étape suivante en cliquant sur le bouton pour effectuer la commande.

#### Commande validée ?

À ce stade, une décision est prise : la commande est-elle validée par le système ?

OUI : Si la commande est validée, le processus se poursuit.

NON : Si la commande n'est pas validée, l'utilisateur est redirigé vers une étape où il peut entrer en contact avec le service client pour résoudre les problèmes.

#### Préparation de la livraison

Si la commande est validée, l'étape suivante consiste à préparer la livraison des articles commandés.

#### Livraison

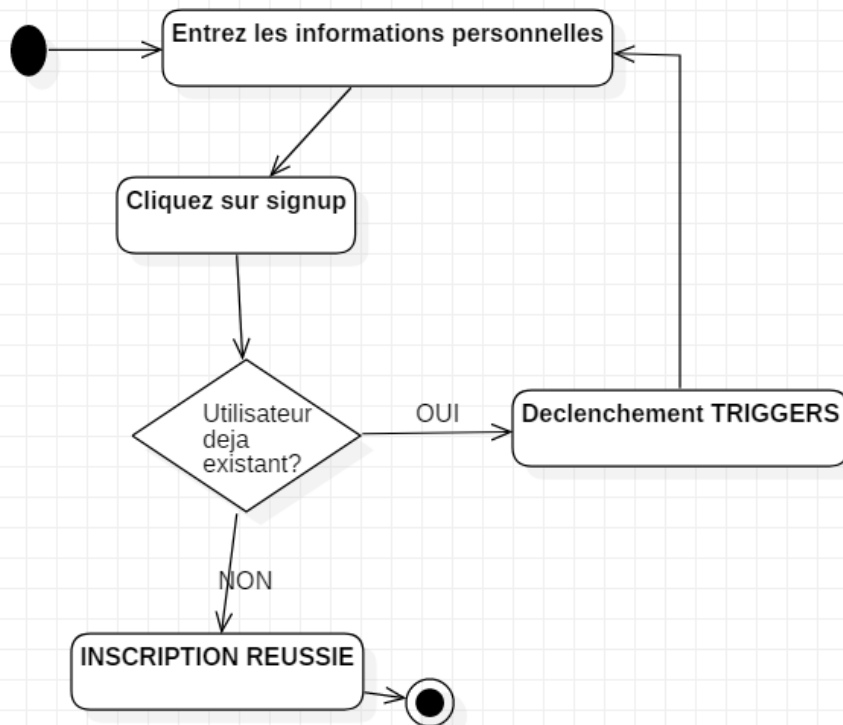
Les articles sont ensuite livrés au client.

#### Réduction en stock

Enfin, après la livraison, le stock est mis à jour pour refléter les articles qui ont été vendus.

- **INSCRIPTION**





## EXPLICATION

### ENTREZ LES INFORMATIONS PERSONNELLES

Début du processus, ici l'utilisateur entre ses informations personnelles (nom, prénom, date de naissance, numéro, sexe etc...) à fin de pouvoir être enregistré dans notre base de données.

### CLIQUEZ SUR SIGNUP

Le bouton signup est un bouton événementiel qui effectuera l'action d'inscription et qui stockera l'utilisateur dans la base de données.

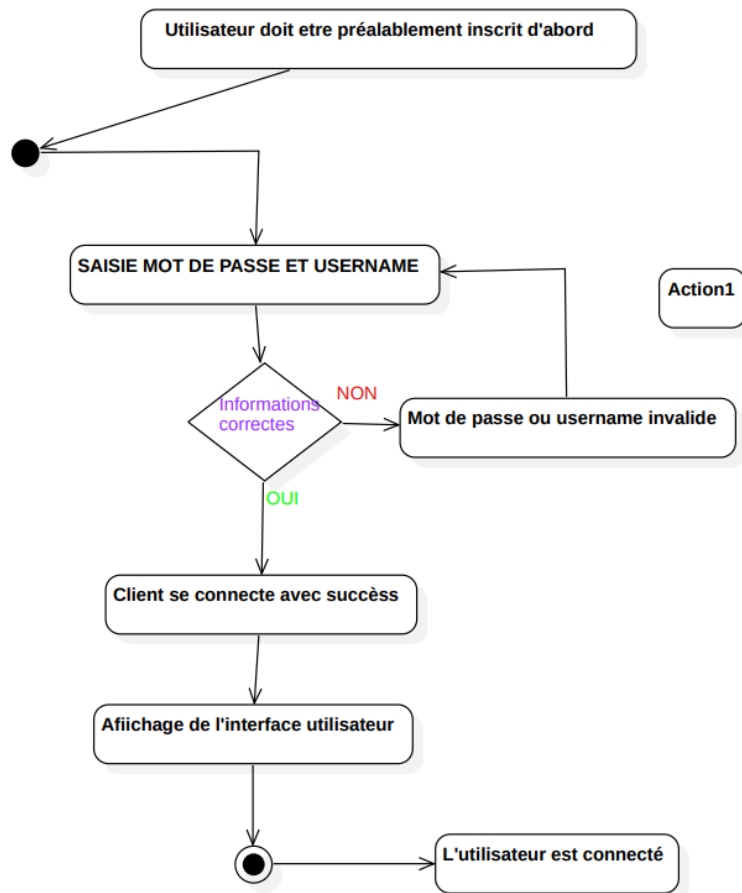
### CONDITION : UTILISATEUR DÉJÀ EXISTANT ?

Dans cette condition on vérifie si les données entrées ( principalement l'adresse mail ) sont déjà présentes dans la base de données.

SI OUI : un trigger se déclenche empêchant l'inscription et forçant à réeffectuer la saisie des données

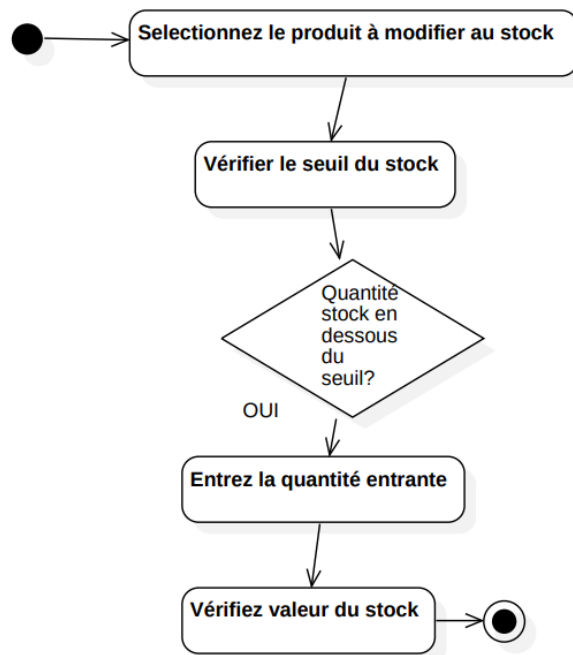
SI NON : l'inscription est réussie.

- **LOGIN**



## EXPLICATION

- MODIFIER STOCK



## **EXPLICATION DU DIAGRAMME**

- **Sélectionner le produit à ajouter au stock**

L'utilisateur ou le système choisit le produit dont on souhaite modifier le stock.

- **Vérifier le seuil du stock**

Une vérification est effectuée pour déterminer si le niveau actuel du stock est inférieur à un certain seuil critique.

- **Quantité stock en dessous du seuil ?**

Si la quantité en stock est inférieure au seuil, il est nécessaire d'ajouter des produits au stock (processus continue).

Si le stock est suffisant (réponse "NON"), le processus peut être arrêté ou ajusté selon d'autres paramètres.

- **Entrer la quantité entrante**

L'utilisateur entre la quantité de produits qui va être ajoutée au stock.

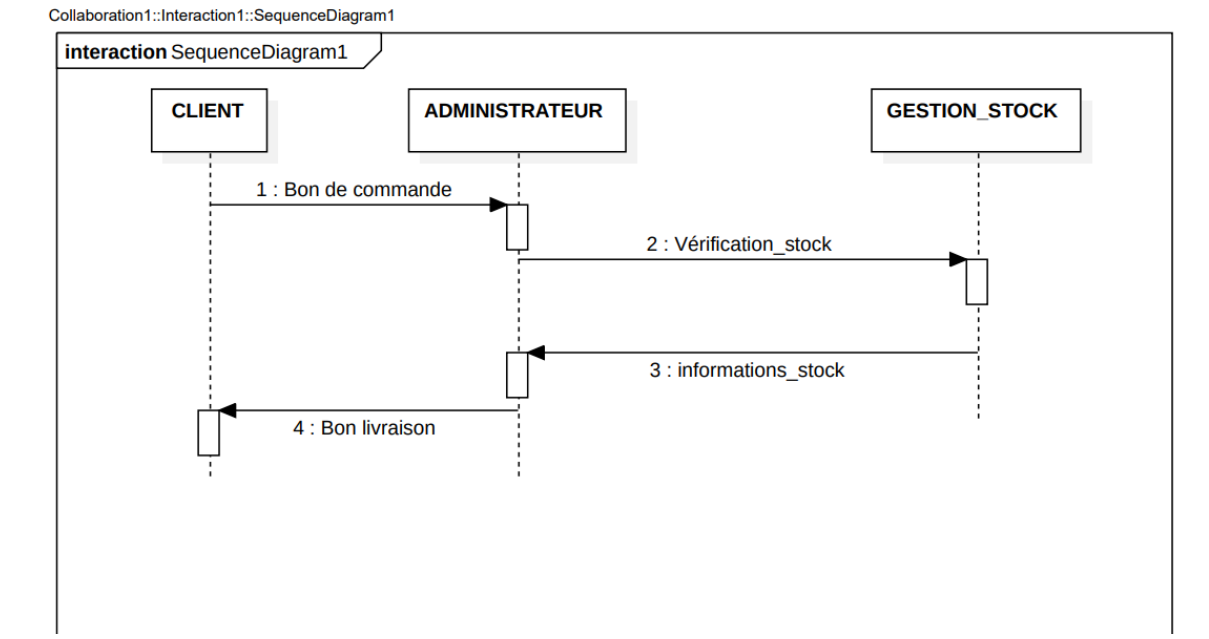
- **Vérifiez la valeur du stock**

Une validation finale est faite pour confirmer que la valeur du stock est correcte après la mise à jour.

## **DIAGRAMME DE SEQUENCE**

Un diagramme de séquence est un type de diagramme UML (Unified Modeling Language) qui illustre les interactions entre les objets d'un système dans le temps.

- **PASSER COMMANDE**



## Acteurs

Client : celui chargé d'effectuer des commandes.

Administrateur celui chargé de la supervision de tout le système

Gestion stock : Gestion responsable du stock des livres.

## Étapes du Processus

### 1. Bon de commande

Cette étape représente l'action où le client passe une commande auprès de l'administrateur.

### 2. Vérification stock

Une fois la commande passée, l'administrateur effectue une vérification du stock des produits dans la gestion des stocks.

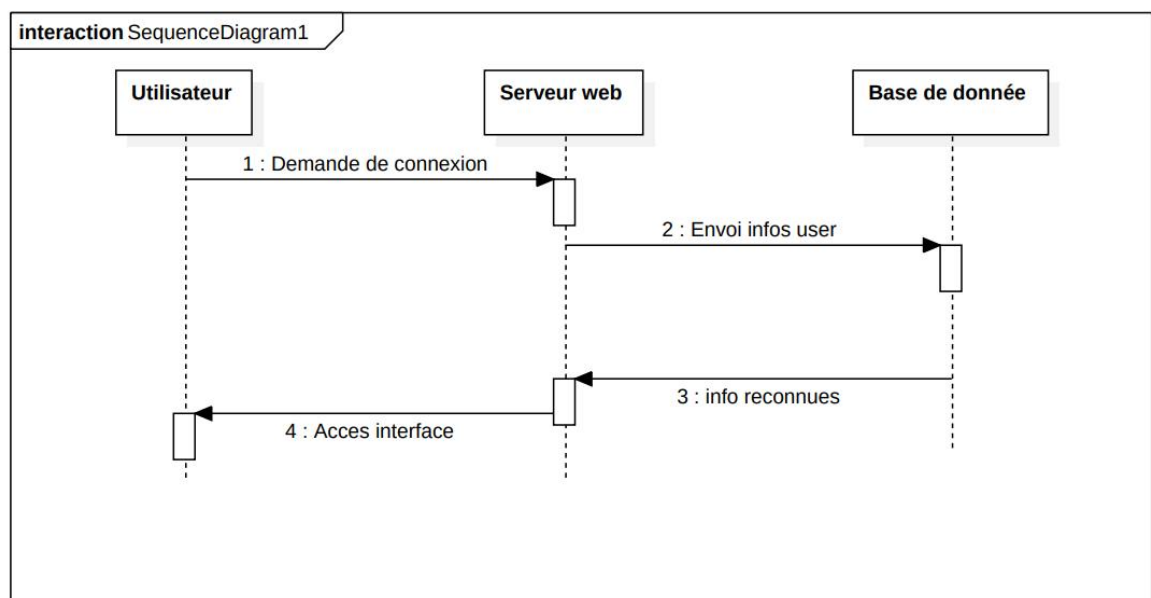
### 3. informations stock

Le département du stock renvoie donc les informations relatives à la quantité et disponibilité des produits de la commande à l'administrateur.

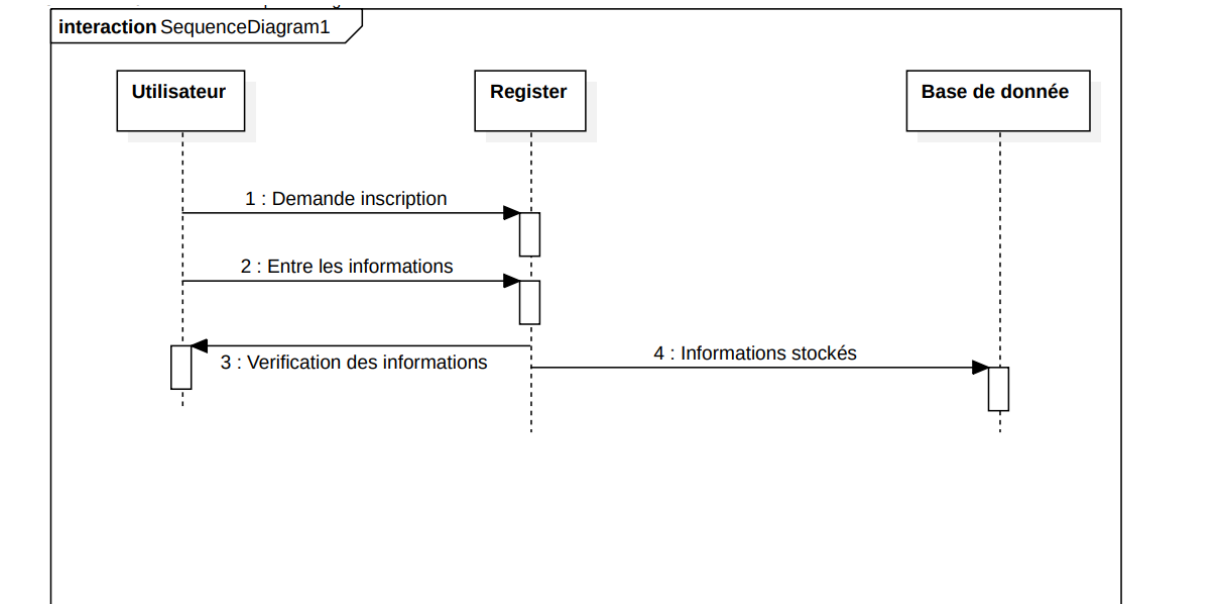
#### 4. Bon livraison

Une fois les informations du stock reçues on envoie un bon de livraison auprès du client.

- **LOGIN**



- **INSCRIPTION**



## **EXPLICATION**

### **Acteurs**

Utilisateur : La personne qui souhaite s'inscrire.

Register : Le système d'enregistrement qui gère les demandes d'inscription.

Base de données : Système de stockage des informations.

### **Description des interactions**

Demande d'inscription : L'utilisateur initie une demande pour s'inscrire au système.

Entrée des informations : L'utilisateur fournit les informations nécessaires, comme son nom, email, etc.

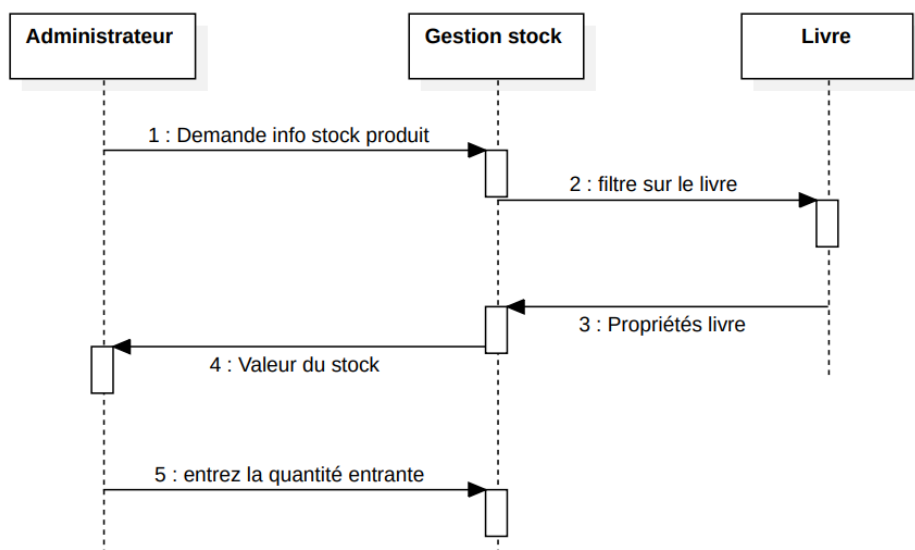


Vérification des informations : Le système vérifie les données fournies (par exemple, pour s'assurer qu'elles sont valides et que l'utilisateur n'existe pas déjà).

Informations stockées : Si la vérification réussit, les informations de l'utilisateur sont sauvegardées dans la base de données.

- **MISE A JOUR STOCK**

Collaboration1::Interaction1::SequenceDiagram1



## **EXPLICATION**

## **Acteurs**

Administrateur : Personne qui gère les stocks.

Livre : Objet représentant un produit dans le stock.

## **Description des interactions**

Demande d'information sur le stock produit : L'administrateur initie le processus en demandant des informations sur un produit spécifique dans le stock.

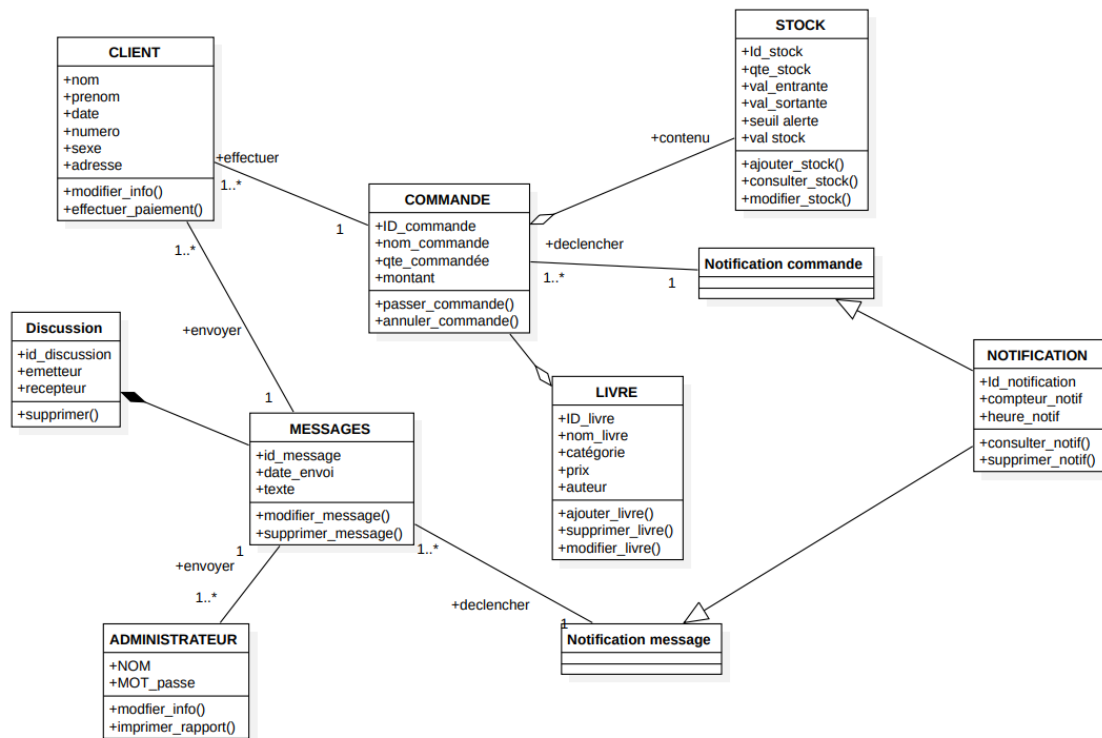
Filtre sur le livre : Une recherche est effectuée pour filtrer les données afin de localiser le produit concerné (le livre dans ce cas).

Propriétés du livre : Le système renvoie les propriétés détaillées du livre sélectionné.

Valeur du stock : L'administrateur reçoit la valeur actuelle du stock du produit.

Entrée de la quantité entrante : L'administrateur ajoute une nouvelle quantité du produit au stock, mettant à jour les données.

## **DIAGRAMME DE CLASSE**



## EXPLICATION DU DIAGRAMME DE CLASSE

### Classes et Attributs

#### Commande

##### Attributs :

ID\_commande : Identifiant unique de la commande.

nom\_commande : Nom associé à la commande.

qte\_commandée : Quantité d'articles commandés.

montant : Montant total de la commande.

##### Méthodes :

passer\_commande() : Méthode pour passer une nouvelle commande.

annuler\_commande() : Méthode pour annuler une commande existante.

## **Stock**

### Attributs:

Id\_stock : Identifiant du stock.

qte\_stock : Quantité d'articles en stock.

val\_entrante : Valeur des articles entrants.

val\_sortante : Valeur des articles sortants.

seuil\_alerte : Seuil d'alerte pour le stock faible.

### Méthodes :

ajouter\_stock() : Méthode pour ajouter des articles au stock.

consulter\_stock() : Méthode pour consulter l'état du stock.

modifier\_stock() : Méthode pour modifier les informations du stock.

## **Client**

### Attributs :

nom, prenom, date, numero, sexe, adresse : Informations personnelles du client.

### Méthodes :

modifier\_info() : Méthode pour modifier les informations du client.

## **Livre**

### Attributs :

ID\_livre, nom\_livre, catégorie, prix, auteur : Informations sur les livres disponibles dans le système.

### Méthodes :

ajouter\_livre(), supprimer\_livre(), modifier\_livre() : Méthodes pour gérer les livres dans le système.

### **Administrateur**

#### Attributs :

NOM, MOT\_passe : Informations d'identification de l'administrateur.

#### Méthodes :

modifier\_info() : Méthode pour modifier les informations de l'administrateur.

### **Notification**

#### Attributs :

Id\_notification, compteur\_notif, heure\_notif : Informations sur les notifications envoyées au client ou à l'administrateur.

#### Méthodes :

consulter\_notif(), supprimer\_notif() : Méthodes pour gérer les notifications.

### **Messages**

#### Attributs :

id\_message, date\_envoi, texte : Informations sur les messages échangés dans le système.

#### Méthodes :

modifier\_message(), supprimer\_message() : Méthodes pour gérer les messages.

### **Discussion**

#### Attributs :

id\_discussion, emetteur, recepateur : Informations sur les discussions entre utilisateurs (clients, administrateurs).

### Méthodes :

supprimer() : Méthode pour supprimer une discussion.

### **Relations entre Classes**

Les relations entre les classes sont indiquées par des lignes connectant les différentes entités, avec des notations qui montrent la multiplicité (par exemple, 1..\* indique qu'une classe peut être liée à plusieurs instances d'une autre classe).

Par exemple, une commande peut être associée à plusieurs notifications et messages, ce qui montre comment ces entités interagissent dans le système.