

# Rapport projet JEE

**Groupe :** CAPATINA Vadim / GUTAN Nicolae

**Professeur :** BRAIKI Faycal

## Contexte du projet

À la suite du remporte de notre entreprise de l'appel d'offre moi et Nicolae, on a eu la responsabilité de conception et de création de l'application web reposant sur les technologies JEE pour le revendeur informatique LDXL. L'application a comme but la gestion des ventes, du stock, des références produits et de ses clients. Le cahier de charge étant fourni au préalable par le client.

## Cahier de charges

- L'application doit permettre la connexion à différents utilisateurs par le biais d'une authentification sécurisée de type formulaire ;
- L'utilisateur aura la possibilité d'accéder à l'ensemble des fonctionnalités de l'application à partir d'un menu ;
- L'application doit permettre les opérations de type CRUD sur la base de données ;
- Chaque vue indiquera le nom d'utilisateur et elle aura un bouton qui permettra à l'utilisateur la déconnexion de l'application ;

### Spécification fonctionnelles générales :

#### Module Client :

- Les opérations CRUD sur l'ensemble des données des clients ;
- La recherche des clients à partir de son nom ou de son ID ;
- La récupération des achats d'un clients.

#### Sécurisation :

- L'accès à l'application se fait de manière sécurisée.

## Le choix technologique

Compte tenu de désirs du clients, l'application web est réalisé avec la technologie JEE et utilisant la framework Spring et l'ORM Hibernate.

## L'architecture du projet

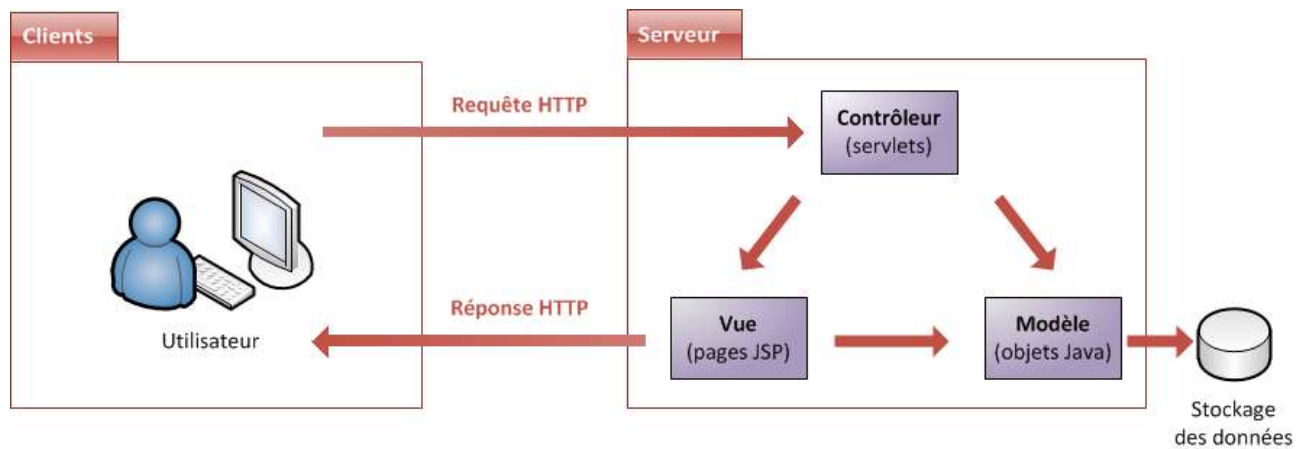
Une fois les choix technologiques faites, l'architecture de l'application est évidente. Elle repose sur le modèle MVC.

- **Le modèle** est la partie relative aux données traitées par l'application ;
- **La vue** est partie relative aux interactions avec l'utilisateur ;
- **Le contrôleur** est la partie relative à la gestion de l'ensemble de l'application.

Dans une application web java, l'utilisation des servlets, des JSP et des JavaBeans permettent d'implémenter MVC en respectant toutefois certaines règles :

- **Les Servlets** jouent le rôle de **contrôleur** ;
- **Les JSP** jouent le rôle de **vue** ;
- **Les Beans** permettent l'accès au **modèle**.

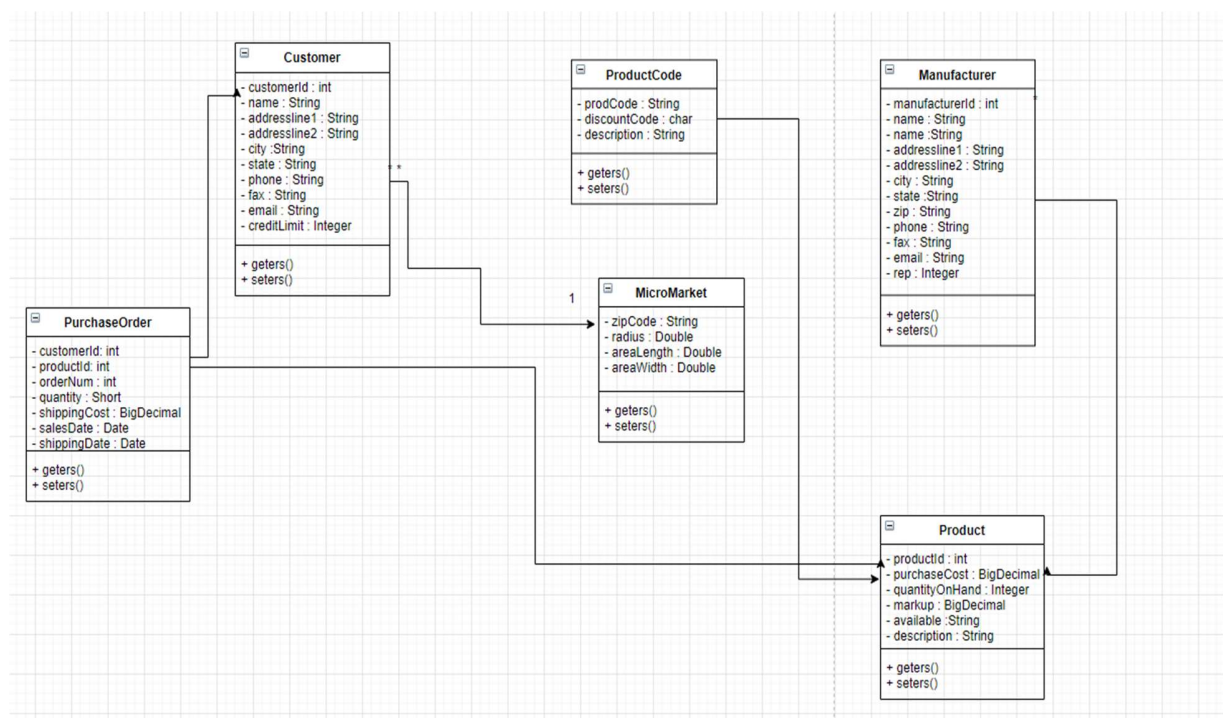
La figure qui suite permettra visualiser et comprendre au mieux l'architecture choisie.



### La base de données

JAVA permet l'accès aux bases de données via son API JDBC (Java DataBase Connectivity). Elle permet d'opérer les opérations de type «CRUD» (Create, Read, Update, Delete) par le biais d'une connexion à la base de données. À partir d'un servlet, la connexion via JDBC à la base de données donnera donc la possibilité de manipuler les données stockées.

La compréhension du produit final qu'on doit livrer et aussi les besoins du client nous a permis de construire les classes qui représenteront le squelette des tableaux de la base de données.



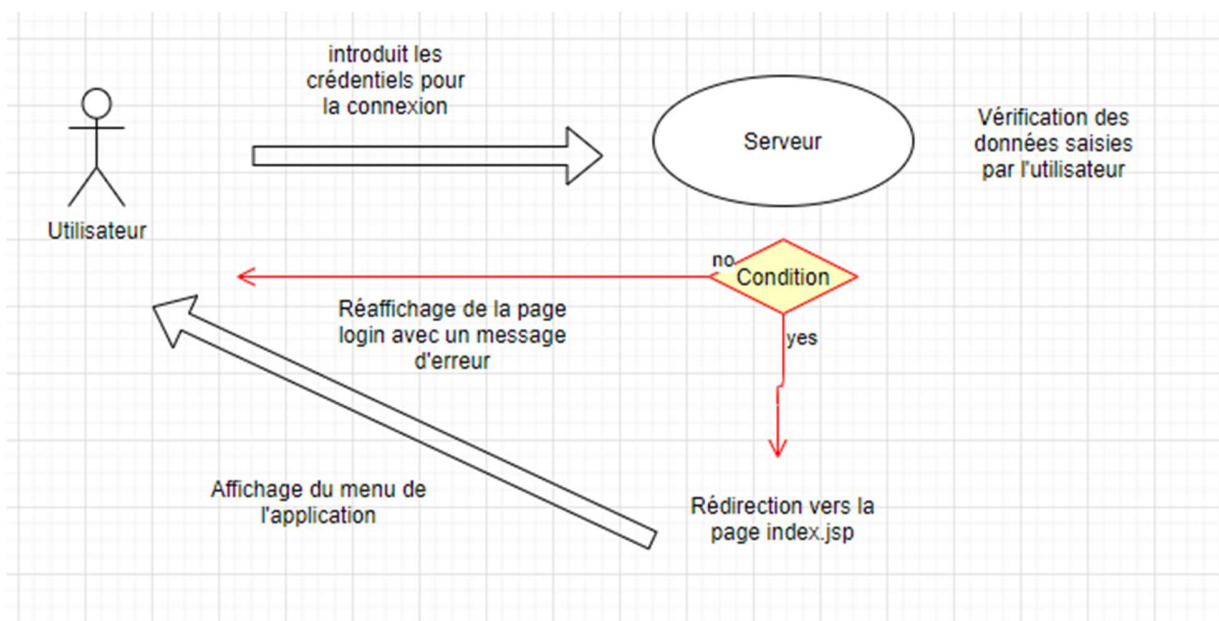
Pour réaliser le mapping des classes aux tableaux de la base de données, nous avons utilisé la solution ORM(Object Relational Mapping) Hibernate qui enfaite nous permet :

- Le mapping des tables avec les classes, des champs avec les attributs, des relations et des cardinalités ;
- Une interface qui permette de facilement mettre en œuvre des actions de type CRUD ;
- Un langage de requêtes indépendant de la base de données cible et assurer une traduction en SQL natif selon la base utilisée.
- Le support de différentes formes d'identifiants générés automatiquement par les bases de données (identity, sequence) ;

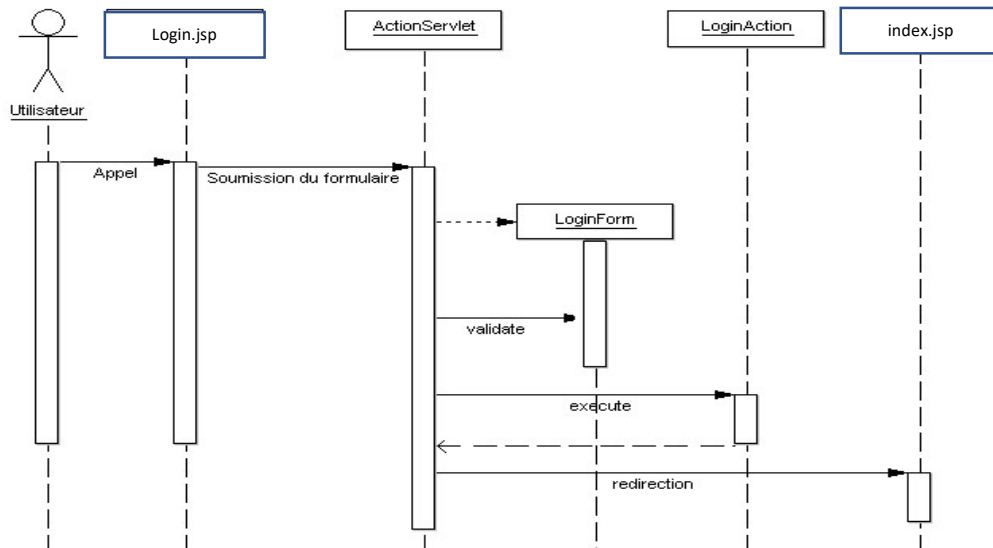
### Gestion du projet

Pour la gestion du projet on a choisi la méthode Agile. Au départ on avait commencé avec l'analyse et la compréhension des besoins du client. Par la suite on a listé les fonctions principales que l'application doit contenir. Une fois cette étape terminée on a passé à la rédaction des diagrammes de cas d'utilisation. Cette étape nous a permis de se mettre d'accord sur le principe du fonctionnement des certains fonctionnalités de l'application. Dans ce rapport on a choisi de vous montrer le diagramme de cas d'utilisation et de séquence de la connexion de l'utilisateur à notre application.

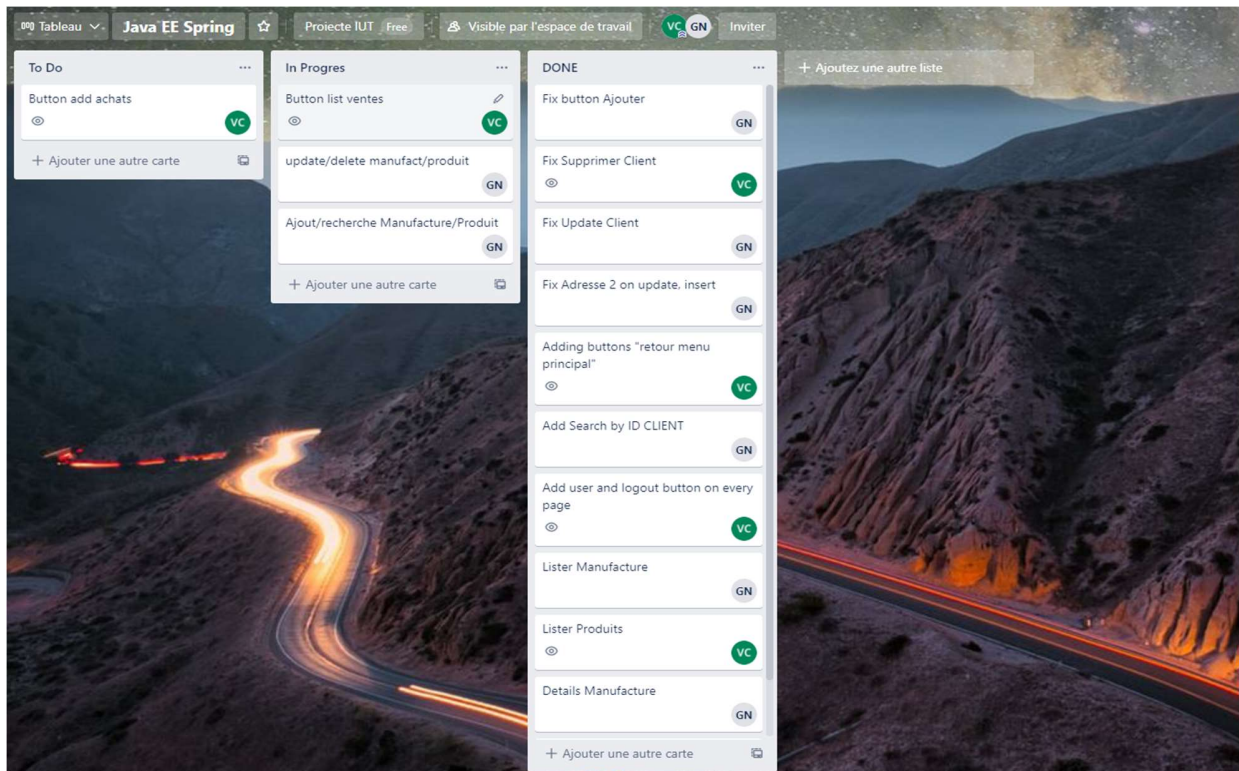
#### Diagramme cas d'utilisation connexion de l'utilisateur.



## Diagramme de séquence connexion de l'utilisateur



Tout au long du projet nous avons utilisé l'outil « TRELLO » pour la répartition et la réalisation des tâches.



## **Conclusion**

Du point de vue personnel, ce projet a été pour nous un bon exercice pour la mise en pratique des connaissances acquises pendant les cours de JEE. Nous avons approfondi nos compétences dans le domaine de développement des applications web avec la technologie Java EE, la framework Spring et l'ORM Hibernate. Tout au long du projet on avait rencontré des problèmes suite à la nouvelle technologie utilisé mais avec beaucoup de patience et persévérance on a réussi à chercher et trouver les bonnes solutions.

De même le travail en autonomie, nous a apporté beaucoup d'expérience professionnelle. On a cherché la meilleure méthode d'organisation du travail pour être productif chaque jour et réaliser un bon travail.