



RAPPORT PROJET JEE

GSI 2

ELWAN BAZIZI & YASSINE ES-SADANY
CHARLES CAMBOULIN & SOUFIANE BAKRIM

Sommaire :

Introduction :.....	3
Partie 1 :	4
Partie 2 :	6
Difficultés rencontrées :	10

Introduction :

On souhaite créer une application E-Commerce permettant d'effectuer des transactions d'achat sur Internet. L'application contiendra deux interfaces : une première interface pour le client et une autre pour l'administrateur.

Plusieurs actions sont possibles lorsque l'on est connecté en tant qu'administrateur :

- Action sur les produits : consulter des produits, ajouter des produits avec une quantité au panier, supprimer un produit du panier, consulter toutes les catégories, consulter les produits d'une catégorie, chercher des produits par mot clés.
- Action sur les clients : accorder le droit aux clients inscrit de pouvoir laisser un commentaire sur un produit.

Plusieurs actions sont possibles en tant que client du site :

- Action en tant que client (non connecté) : consulter toutes les catégories, consulter les produits d'une catégorie, chercher des produits par mot clés, ajouter des produits avec une quantité au panier, supprimer un produit du panier, possibilité de s'inscrire.
- Action en tant que client (connecté) : consulter toutes les catégories, consulter les produits d'une catégorie, chercher des produits par mot clés, ajouter des produits avec une quantité au panier, supprimer un produit du panier, passer une commande puis payer la commande et consulter ses commandes passées.

Dans la suite de ce rapport, nous détaillerons dans une première partie les moyens mis en place pour la réalisation de ce site (une représentation du diagramme de classe, un schéma de la base de données avec le détail des tables associées, structure utilisée dans la construction du code et les classes utilitaires utilisées).

Dans une seconde partie nous détaillerons le fonctionnement et l'intérêt des pages web de notre site ainsi que l'interaction qu'il y a entre client du site et administrateur.

Partie 1 :

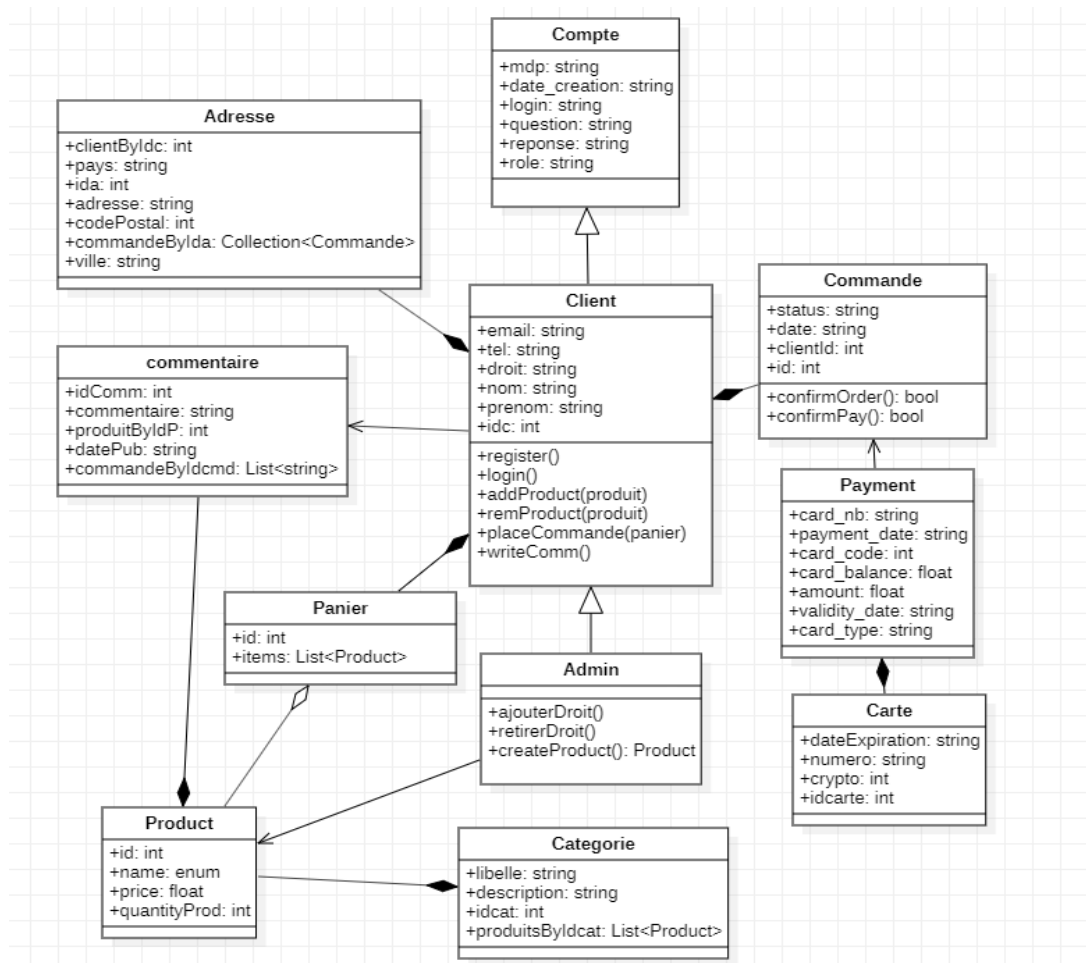


Figure 1 Diagramme de classe final

Afin de bien concevoir le code de notre projet, pour avoir une approche sur la structure du code nous avons commencé par réaliser un diagramme de classe modélisant les interactions entre les classes qui constituent notre projet.

Ce diagramme nous a permis d'établir la première conception de notre projet. A travers ce diagramme nous avons pu comprendre et établir les différentes actions du client comme s'inscrire, se connecter, ajouter ou supprimer un produit du panier et passer une commande. Notamment comment nous devons répertorier les produits en créant des catégories. Et comment gérer une commande en créant une sous-commande pour chaque produit du panier en fonction de la quantité entrée. Enfin, l'administrateur dispose en plus des actions du client, la possibilité de créer des produits et de les consulter. Il peut aussi agir sur le client en lui ajoutant ou supprimant des droits liés à l'écriture de commentaire.

The image displays a database schema, a Java package structure, and a list of classes.

Database Schema:

- compte**
 - login varchar(50)
 - mdp varchar(50)
 - role varchar(50)
 - question varchar(255)
 - reponse varchar(255)
 - date_creation date
- client**
 - idc int(10)
 - nom varchar(255)
 - prenom varchar(255)
 - email varchar(50)
 - tel varchar(30)
 - regDate date
 - droit varchar(25)
 - login varchar(50)
- adresse**
 - ida int(8)
 - adresse varchar(100)
 - ville varchar(50)
 - codepostale int(6)
 - pays varchar(50)
 - idc int(8)
- commande**
 - idcmd int(10)
 - idc int(10)
 - datecmd date
 - methode_pay varchar(30)
 - etat varchar(30)
 - ida int(8)
- produit**
 - idP int(10)
 - idcat int(10)
 - libelle varchar(255)
 - Marque varchar(255)
 - prix int(12)
 - Description varchar(1255)
 - fraisExpedition int(12)
 - disponibilite tinyint(1)
 - qtestck int(4)
 - date_pub date
- image**
 - idimg int(8)
 - url varchar(200)
 - isprincipal int(1)
 - idp int(8)
- lignecommande**
 - idlc int(8)
 - idcmd int(8)
 - idp int(8)
 - qte int(8)
 - prix_achat int(10)
- commentaire**
 - idComm int(10)
 - idP int(10)
 - idcmd int(8)
 - commentaire varchar(255)
 - date_pub date
- categorie**
 - idcat int(10)
 - libelle varchar(30)
 - description varchar(200)
- carte**
 - idcarte int(10)
 - numero varchar(16)
 - dateExpiration varchar(5)
 - crypto int(3)

Java Package Structure:

- main
 - java
 - com.ecommerce
 - controller
 - AddProduit
 - DroitClient
 - GererPanier
 - Inscription
 - Login
 - Logout
 - Paieement
 - SuppProduit
 - dao
 - HibernateSession
 - HibernateUtil
 - metier
 - Adresse
 - Bloque
 - Carte
 - Categorie
 - Client
 - Commande
 - Commentaire
 - Compte
 - Image
 - Lignecommande
 - LignePanier
 - Panier
 - Produit
 - HelloServlet
- resources
- webapp

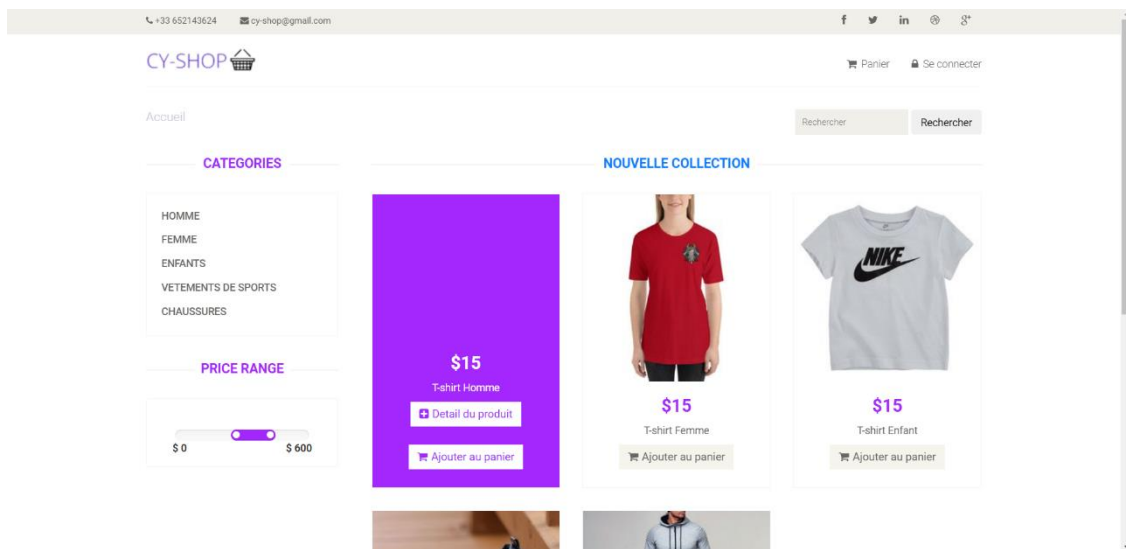
Figure 3/ Schéma de la base de données

Figure 2 Structure du code

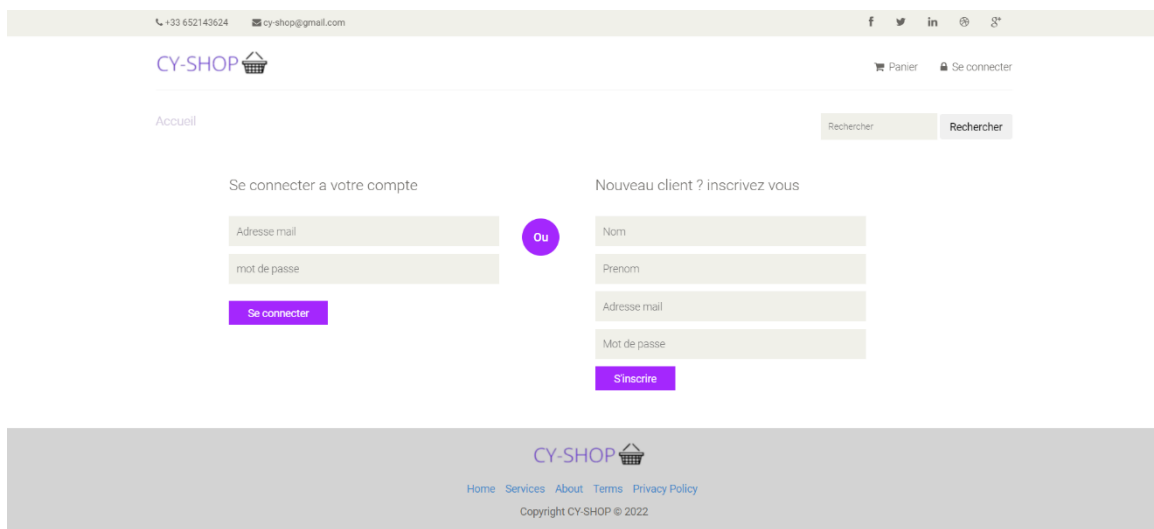
Partie 2 :

Au niveau du fonctionnement du site nous allons voir les interfaces principales nécessaire utilisé par le client pour effectuer une commande :

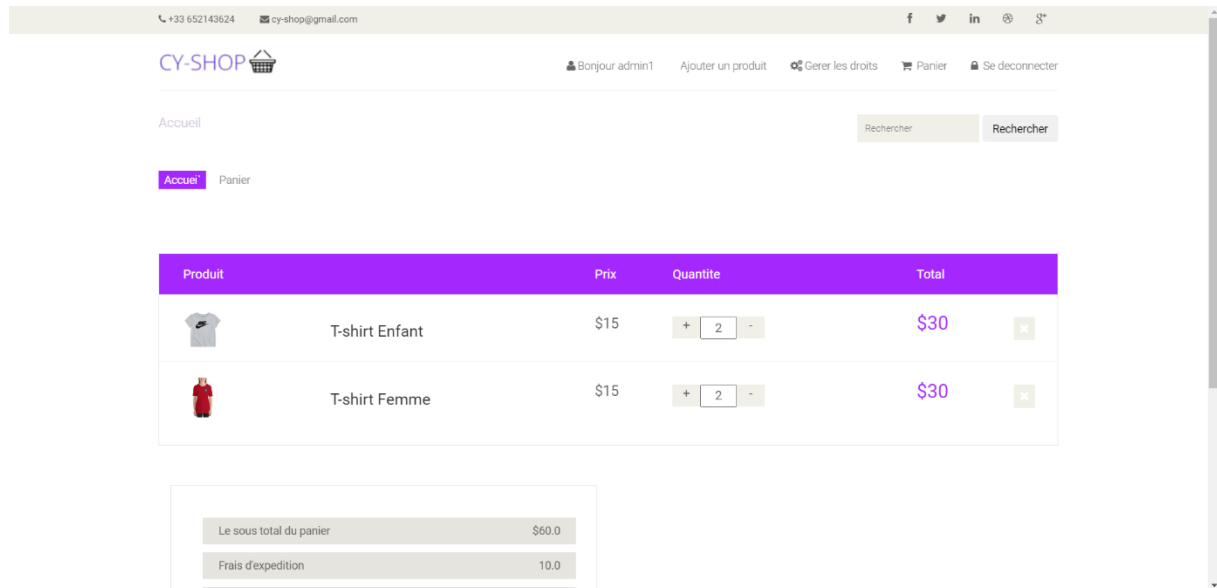
La page “index.jsp” notre page d'accueil du site, ci-dessous nous présente les différents produits et catégories ou nous pouvons retrouver ces produits classés. Le client (non connecter) peut ajouter à son panier des produits, consulter les produits et se connecter ou bien s’inscrire via le bouton se connecter.



Une fois sur la page “formLogin.jsp” ci-dessous, le client peut se connecter ou s’inscrire. Lors ajoutées à la BDD en créant une instance Client et Compte qui seront nécessaires pour vérifier ses données lorsqu’il souhaitera se connecter.



Une fois connecté et que le client a ajouté des produits à son panier, il peut effectuer une commande. Sur la page ci-dessous “panier.jsp”, le client peut choisir la quantité et même supprimer les produits. Il peut ensuite cliquer sur payer ou il devra entrer ses informations afin de poursuivre le paiement.



Sur la page ci-dessous “checkout.jsp” le client doit entrer ses informations qui seront traitées par le controller (/Paieement) et créera par la suite une instance de la classe Adresse qui contiendra les informations entrées par le client. -> La première étape du paiement.

Accueil | Gérer mes commandes

Etape 1

Information du client

admin1

admin1

Expédier à :

Address 1 *



Address 2

Zip / Postal Code *

- Pays -

- ville -

Revision de votre panier & paiement

Produit	Description	Prix	Quantité	Total
	T-shirt Enfant	\$15	+ 2 -	\$30
	T-shirt Femme	\$15	+ 2 -	\$30

☐ Carte Visa

Concernant la deuxième étape du paiement (paiement.jsp) ci-dessous, le client devra entrer ses informations bancaires. Ses données seront traitées par le controller (/paiement) et comparer avec notre BDD, pour accepter ou non son paiement.

Donnees de votre carte visa

1234567891234567

12/22

123

Payer

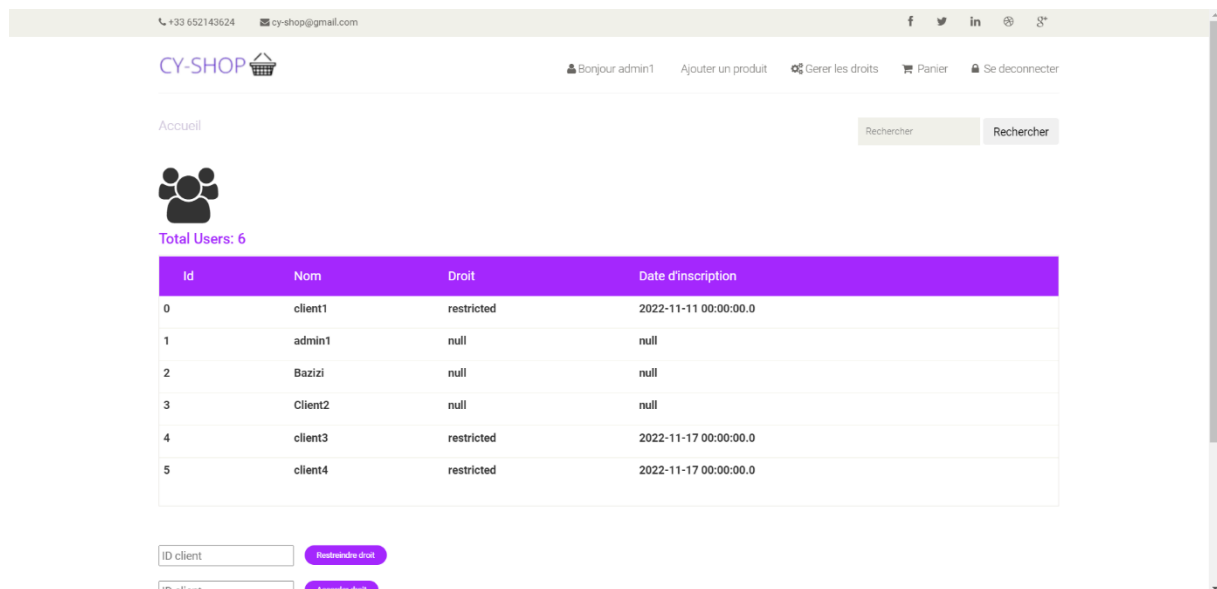
Si les données entrées sont correctes, après traitement au niveau du controller une instance Lignecommande sera créée pour chaque produit du panier ainsi qu'une instance de la classe commande qui contiendra un récapitulatif de toute la commande. Le client aura donc réussi à commander les produits contenus dans le panier.

Votre commande est paye avec succes et sera expedier prochainement!

No de commande	Date	etat	Total	
0	2022-11-17 00:00:00.0	en cours	\$15.0	
2	2022-11-17 00:00:00.0	en cours	\$15.0	
3	2022-11-17 00:00:00.0	en cours	\$15.0	
5	2022-11-17 00:00:00.0	en cours	\$75.0	
7	2022-11-17 00:00:00.0	en cours	\$30.0	
9	2022-11-17 00:00:00.0	en cours	\$30.0	

Au niveau du fonctionnement du site partie administrateur, toutes les actions énoncées précédemment sont faisables par l'administrateur. De plus, il peut interagir avec les produits en les ajoutant au site. Il peut notamment agir sur le droit lié aux commentaires des clients.

Après avoir rempli le formulaire "formAddProduct.jsp", les informations sont traitées par le contrôleur (/AddProduct) ou sont instanciés les produits créés avec leur image associés.

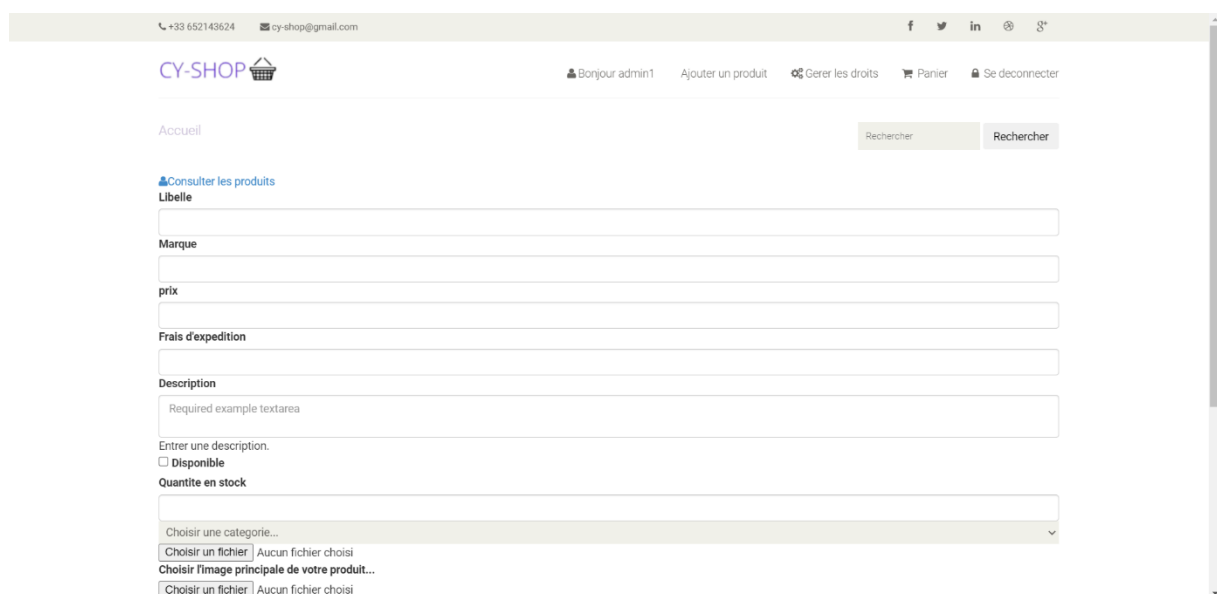


The screenshot shows the administrator interface of CY-SHOP. At the top, there's a header with contact information (+33 652143624, cy-shop@gmail.com) and social media links. The main navigation bar includes "Bonjour admin1", "Ajouter un produit", "Gérer les droits", "Panier", and "Se déconnecter". The "Accueil" (Home) section features a "Rechercher" (Search) button. Below this, a "Total Users: 6" is displayed with a group of people icon. A table lists the users with columns for Id, Nom, Droit, and Date d'inscription.

Id	Nom	Droit	Date d'inscription
0	client1	restricted	2022-11-11 00:00:00.0
1	admin1	null	null
2	Bazizi	null	null
3	Client2	null	null
4	client3	restricted	2022-11-17 00:00:00.0
5	client4	restricted	2022-11-17 00:00:00.0

Below the table, there are two input fields: "ID client" and "Restrindre droit" (with a dropdown arrow), and "ID client" and "Accrocher droit" (with a dropdown arrow).

Sur la page ci-dessous "adminPanel.jsp" l'administrateur a accès aux personnes inscrites sur le site et peut choisir ou non de restreindre leur droit de poster des commentaires.



The screenshot shows the administrator interface of CY-SHOP, specifically the "adminPanel.jsp" page. The header and navigation bar are identical to the previous screenshot. The "Accueil" section has a "Rechercher" button. Below this, there's a link "Consulter les produits". The main form is for adding or editing a product, with fields for "Libelle", "Marque", "prix", "Frais d'expédition", "Description", "Entrez une description.", "Disponible" (checkbox), "Quantité en stock", "Choisir une catégorie...", "Choisir un fichier" (with a dropdown arrow), "Choisir l'image principale de votre produit...", and "Choisir un fichier" (with a dropdown arrow).

Difficultés rencontrées :

- La première difficulté rencontrée, a été la configuration de l'ide pour que nous puissions utiliser Hibernate/JPA et pouvoir générer nos classes à partir de notre base de données. Nous avons perdu beaucoup de temps à faire fonctionner eclipse dans un premier temps car on avait de multiple erreur rendant impossible son utilisation. Puis nous sommes passés sur intellij ou l'on avait moins de problèmes dans l'implémentation du code.
- D'autre part travailler en groupe n'a pas été facile non plus car chacun programmait de son coter et dès lors qu'on devait mettre en commun, ils nous étaient impossible de compiler car nous avions chacun une version différente.