

04/05/2023

PAPPE

Projet Java



PORTAL Thierry

Table des matières

1) Présentation du projet.....	2
1.1) Objectif du projet	2
1.2) Rôle des utilisateurs.....	2
1.3) Test de l'application	2
2) Notice d'utilisation détaillée	3
2.1) Connexion.....	3
2.2) Dashboard	3
2.3) Fonctionnalités Client.....	4
2.3.1) Afficher les stocks d'un médicament.....	5
2.3.2) Acheter un lot de médicament.....	6
2.3.3) Consulter l'historique de vos achats.....	7
2.4) Fonctionnalités Usine	8
2.4.1) Afficher les stocks d'un médicament.....	8
2.4.2) Ajouter un médicaments.....	9
2.4.3) Ajouter un lot de médicament	9
2.4.4) Consulter les stocks de votre usine	10
3) Explication du code.....	11
3.1) Sécurisation Login.....	11
3.2) Hibernate	12
3.3) DAO Générique.....	13

1) Présentation du projet

1.1) Objectif du projet

Ce projet est une application réalisée en Java et qui a pour rôle **la mise en relation de la production des usines de médicaments avec les pharmacies qui cherchent à remplir leurs stocks**. La solution comporte, à son lancement, une partie authentification qui permet l'attribution de l'un des deux rôles suivants : Usine ou Client.

1.2) Rôle des utilisateurs

➤ Partie Usine :

Une fois connectée, une usine pourra consulter la totalité des stocks qu'elle a mise en vente. Cela lui permet de voir les différents médicaments qu'elle vend et les stocks en vente qui lui restent, ce qui lui permettra d'adapter sa production. Une usine peut alors ajouter des stocks de médicaments à la mise en vente. Elle peut également créer un médicament qui n'est pas encore commercialisé pour être le premier à proposer sur l'application le médicament. Les prix sont identiques pour toutes les usines car la vente aux pharmacies ne peut pas être concurrentielle.

➤ Partie Client :

Les clients sont des comptes pour chaque pharmacie afin qu'elle achète des médicaments pour les proposer à la vente dans leurs magasins physiques. Les clients peuvent cliquer sur un médicament pour voir toutes les usines qui le vendent et enfin choisir l'usine qu'il préfère pour acheter ce médicament. Le compte client permet également de consulter l'historique des commandes réalisées pour un compte client.

1.3) Test de l'application

Pour se connecter à un compte usine voici les identifiants :

Identifiant : testUsine

Mot de passe : motdepasse

Pour se connecter à un compte client (pharmacie) voici les identifiants :

Identifiant : testClient

Mot de passe : motdepasse

2) Notice d'utilisation détaillée

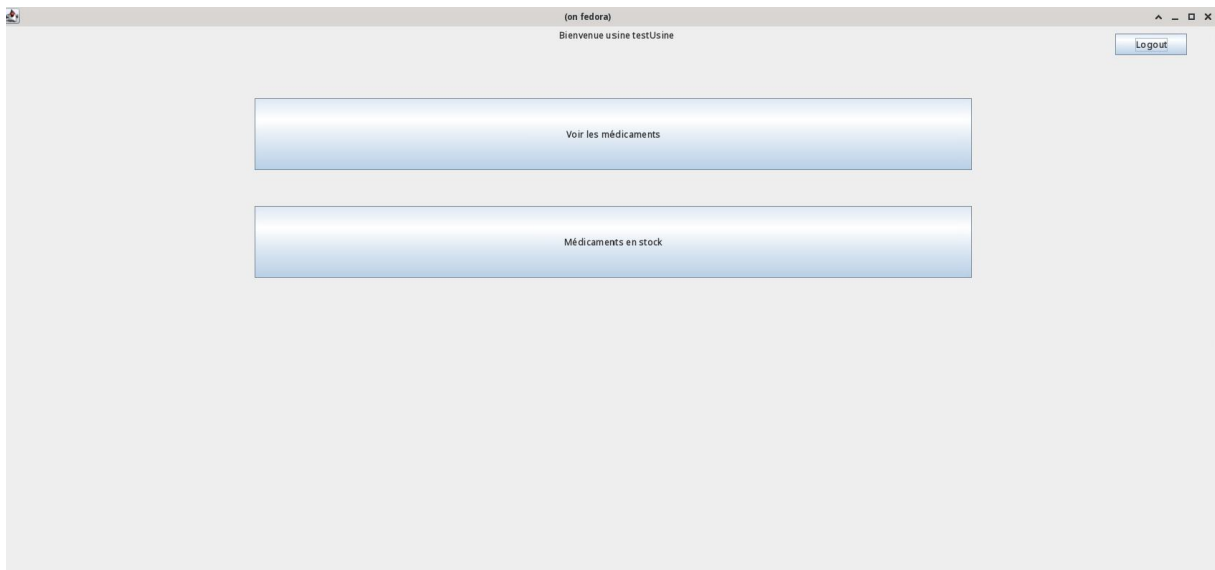
2.1) Connexion

Lorsque vous lancez l'application, vous arrivez sur une page de connexion. Vous devez, à partir de ce moment, rentrer vos identifiants, et choisir si vous voulez être connectés en tant que Usine ou Client.

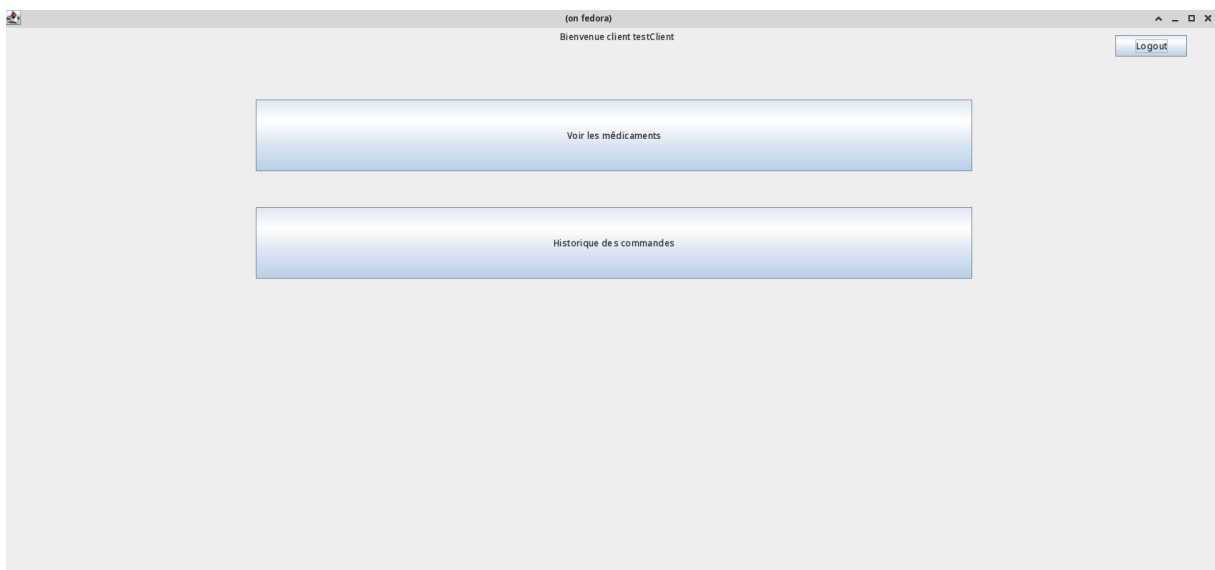
The image shows a screenshot of a web application window. The window has a title bar at the top with the text "(on fedora)" and standard window control buttons (minimize, maximize, close). The main content area has a light gray background. At the top center, the word "Connexion" is displayed in a large, bold, black font. Below this, there are two input fields. The first is labeled "Identifiant" and the second is labeled "Mot de passe". Both labels are in a small, black font. Below the input fields, there is a blue button with the text "Connexion" in white. The button has a slight 3D effect with a darker blue shadow.

2.2) Dashboard

Une fois connecté vous arriverez sur un dashboard où il vous sera proposé d'accéder aux différentes pages. En fonction de vos besoins, nous vous invitons à cliquer sur les carrés de redirection.



Vue Usine

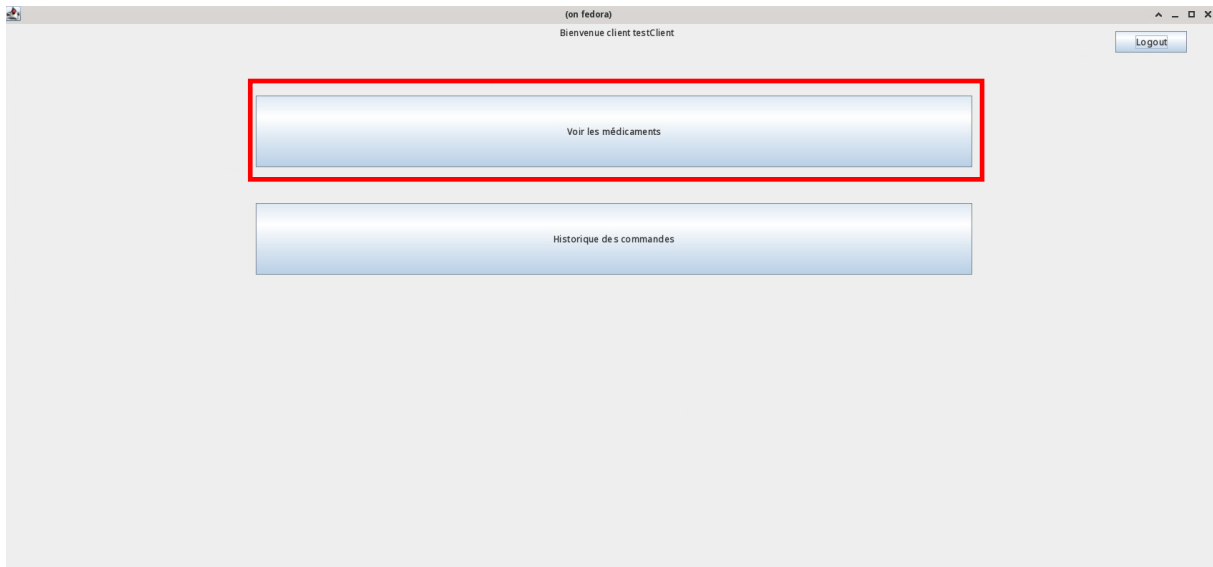


Vue Client

2.3) Fonctionnalités Client

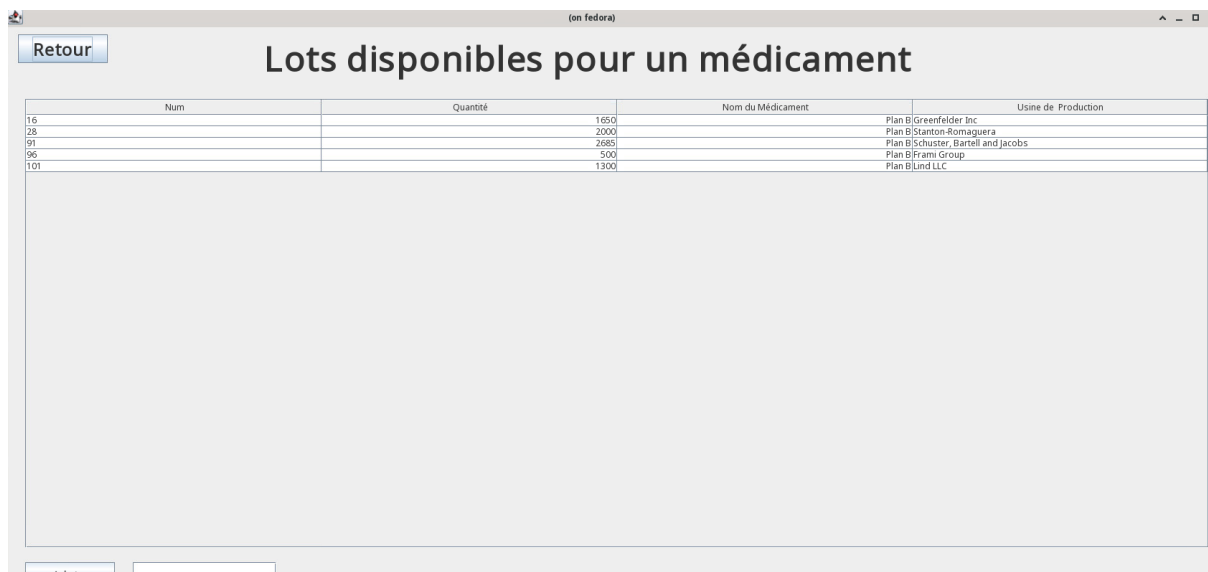
2.3.1) Afficher les stocks d'un médicament

Aller sur l'onglet 'Voir les médicaments'. Vous allez atterrir sur la liste de tous les médicaments. Ici vous pourrez consulter tous les médicaments. Lorsque vous cliquez sur le numéro d'un médicament, vous êtes redirigés sur une page où tous les lots de toutes les usines du médicament concerné seront visibles.



The screenshot shows the same web application window after clicking 'Voir les médicaments'. It features a 'Retour' button in the top left and a search bar with the label 'Recherche :', an 'OK' button, and a text input field. Below the search bar is a table listing various medications.

Id	Nom	Prix
1	Doliprane	17.99
2	CBD	57.99
3	Hi Hand Sanitizer	120.99
4	Amoxicillin and Clavulanate Potassium	50.99
5	Gabapentin	125.99
6	Omeprazole	100.86
7	Plan B	127.24
8	Thyrolar	150.73
9	Clindamycin Hydrochloride	170.40
10	Diltiazem Hydrochloride	35.34
11	Risperidone	73.73
12	Spirolactone	83.72
13	MicroMousse II Alcohol Free Hand Sanitizing Foam	107.75
14	Ultimate Daily Face Wash	92.62
15	Bion3	7.99
16	Med Aid Chest Rub	24.18
17	Nitroglycerin	7.52
18	Ammonia	126.44
19	Run Kang Alcohol Pads - MEDIUM	138.41
20	cool heat	115.46
21	KALIMURBATICUM	98.37
22	Crayola	159.24
23	Cephalexin	52.88
24	Anti-Bacterial Hand	152.21
25	Zolpidem Tartrate	86.39
26	irinotecan Hydrochloride	185.42
27	Flu RELIEF	96.49
28	Ondansetron	152.55
29	smart sense pain relief	85.55
30	Conchaes Argentum	25.10
31	Joek Itch	69.48
32	Carboplatin	187.21
33	ULTRA CONTROL	14.20
34	Treatment Set T5336851	11.06
35	Valium	37.99
36	Povidone-Iodine	141.78

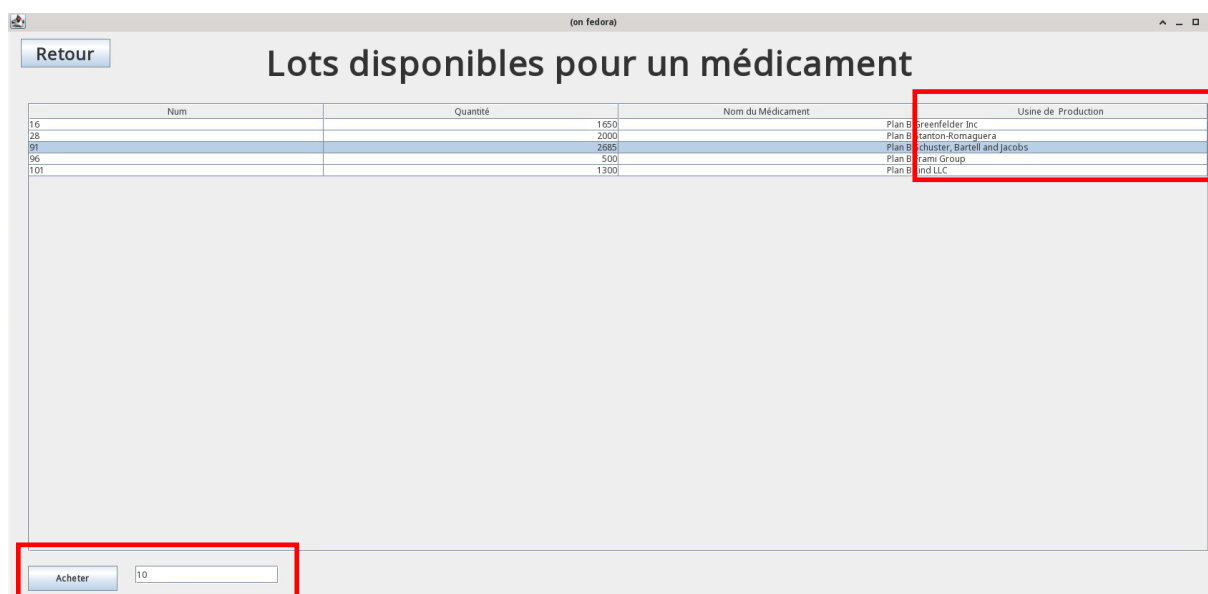


2.3.2) Acheter un lot de médicament

Une fois dans la liste des stocks vous pouvez acheter un lot du médicament à l'usine de votre souhait.

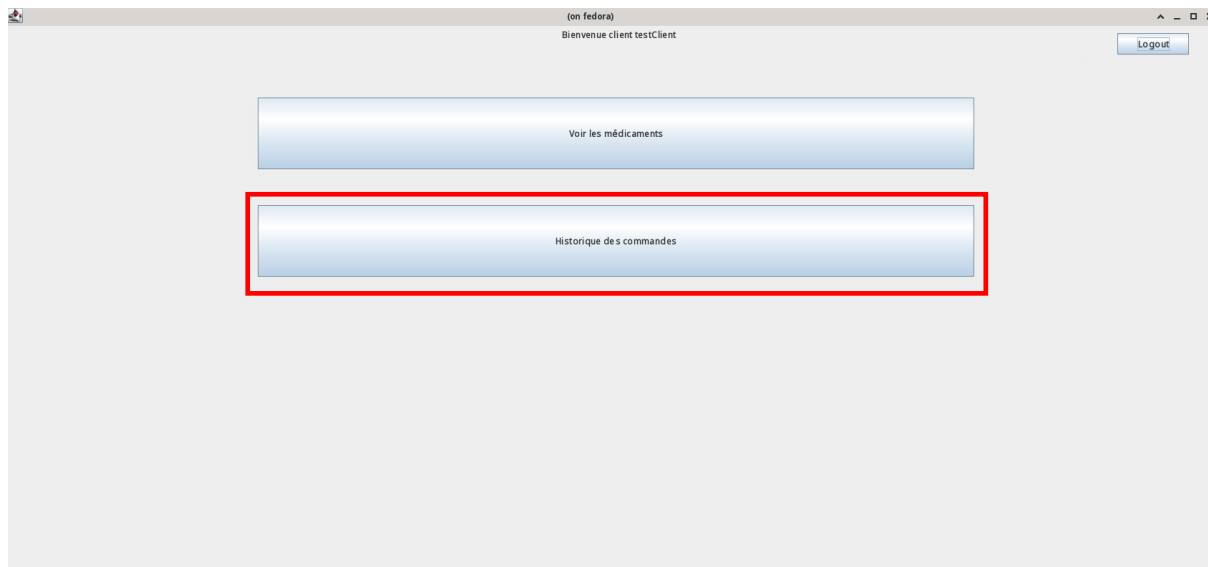
Pour cela, vous devez cliquer sur l'usine à laquelle vous souhaitez acheter votre médicament. Ensuite, rentrer le nombre de lots que vous désirez acheter et enfin cliquer sur le bouton « acheter ».

Vous avez acheté vos médicaments, maintenant vous pouvez vous référer au chapitre suivant pour consulter l'historique et vérifier que vous avez bien réalisé votre commande.



2.3.3) Consulter l'historique de vos achats

Pour accéder à votre historique, vous devez retourner sur le dashboard. Une fois sur cet écran, cliquer sur le bouton « Historique de vos commandes ». Maintenant, vous pouvez consulter votre historique.



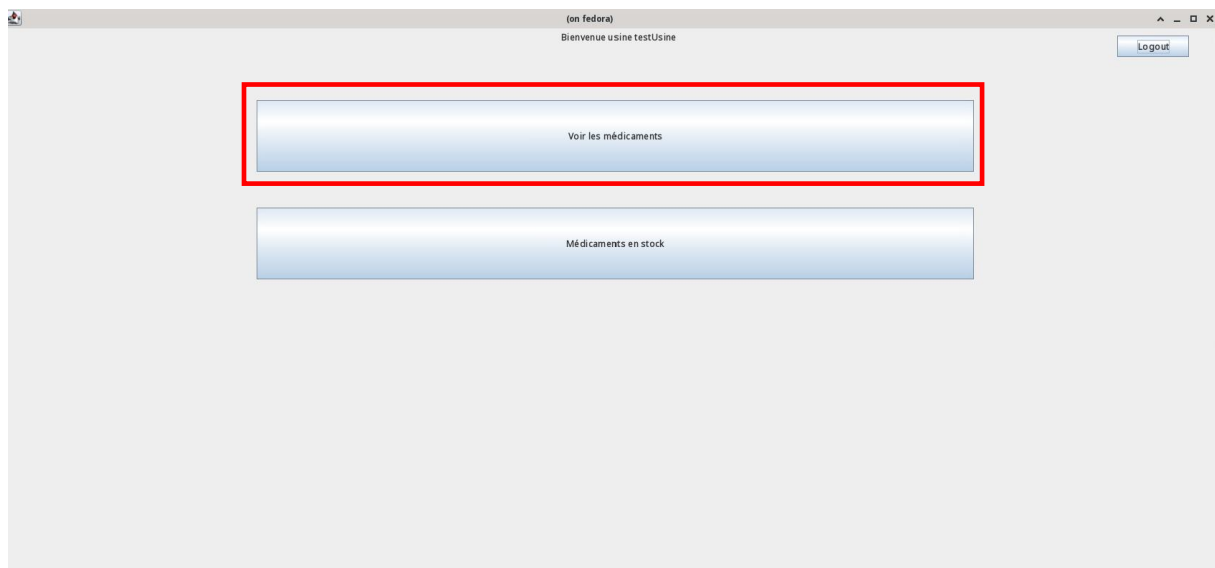
The screenshot shows the 'Historique des commandes' page. It has a title bar '(on fedora)' and a subtitle 'Historique des commandes'. There is a 'Retour' button in the top left corner. Below the title, there is a table with the following data:

Numéro de la Commande	Nom Pharmacie	Nom du médicament	Nom du Fournisseur	Nombre de lots	Prix Total (en €)
2	Rohan Inc	Amoxicillin and Clavulanate Potassium	Veum, Vandervort and McCullough	82	238,429.25
70	Rohan Inc	Diltiazem Hydrochloride	Lund LLC	10	353.4
71	Rohan Inc	Thyrolar	Veum, Vandervort and McCullough	785	301,460
72	Rohan Inc	Flu RILEP	Schuster, Bartell and Jacobs	70	771,192
73	Rohan Inc	Jock Rich	Wacker, Conn and Olson	36	159,804
74	Rohan Inc	Plan B	Lund LLC	10	165,412
75	Rohan Inc	Plan B	Frami Group	100	63,620
76	Rohan Inc	Omeprazole	Freney, Deckow and Schneider	76	70,602
77	Rohan Inc	Omeprazole	Huel-Jast	11	90,774

2.4) Fonctionnalités Usine

2.4.1) Afficher les stocks d'un médicament

Aller sur l'onglet 'Voir les médicaments'. Vous allez atterrir sur la liste de tous les médicaments. Ici vous pourrez ajouter, modifier, supprimer et consulter tous les médicaments. Lorsque vous cliquez sur le nom d'un médicament, vous êtes redirigés sur une page où tous les lots de toutes les usines du médicament concerné seront visibles.



The screenshot shows the 'Médicaments en stock' page. At the top, there are buttons for 'Retour' and 'Ajouter', and a search bar labeled 'Recherche :'. Below the search bar is a table with three columns: 'Id', 'Nom', and 'Prix'. The table contains 36 rows of medication data. At the bottom of the page, there are buttons for 'Cancel', 'Delete', and 'Update'.

Id	Nom	Prix
1	Doligrane	17.99
2	CBD	57.99
3	Hand Sanitizer	120.99
4	Amoxicillin and Clavulanate Potassium	50.99
5	Gabapentin	125.99
6	Omeprazole	100.86
7	Plan B	127.24
8	Thyrolar	150.73
9	Clindamycin Hydrochloride	170.40
10	Diltiazem Hydrochloride	35.34
11	Risperidone	73.73
12	Piromolactone	83.73
13	MicroMousse II Alcohol Free Hand Sanitizing Foam	107.75
14	Ultimate Daily Face Wash	92.62
15	Bion2	7.99
16	Med Aid Chest Rub	24.18
17	Nitroglycerin	7.52
18	Ammonia	126.44
19	Run Kamp Alcohol Pads - MEDIUM	138.41
20	cool heat	115.45
21	KALI MURIA TICUM	98.37
22	Crayola	159.24
23	Cephalexin	52.88
24	Anti-Bacterial Hand	152.21
25	Zolpidem Tartrate	86.39
26	Imotegan Hydrochloride	185.42
27	Flu RELIEF	96.49
28	Ondansetron	152.55
29	smart sense pain relief	85.55
30	condiae Argentum	25.10
31	lock itch	69.48
32	Carboplatin	187.21
33	ULTRA CONTROL	14.20
34	Treatment Set TS336851	11.06
35	Valium	37.39
36	Povidone-Iodine	141.78

2.4.2) Ajouter un médicaments

Une fois sur l'écran précédent, cliquer sur le bouton 'Ajouter'. Vous allez être dirigé vers un formulaire pour ajouter un médicament. Compléter ce dernier puis cliquer sur ajouter. Penser à ajouter les lots que vous voulez mettre en vente car lorsque vous ajoutez un médicament, ce dernier s'affiche sans aucun lot disponible.

The screenshot shows a web application window titled '(on fedora)'. At the top, there are two buttons: 'Retour' and 'Ajouter', with 'Ajouter' highlighted by a red rectangle. To the right of these buttons is a search bar labeled 'Recherche :' with an 'OK' button. Below the buttons is a table with three columns: 'Id', 'Nom', and 'Prix'. The table contains 36 rows of data, including items like 'Doliprane', 'CBD', 'Hand Sanitizer', 'Amoxicillin and Clavulanate Potassium', 'Gabapentin', 'Omeprazole', 'Plan B', 'Thyrolar', 'Clindamycin Hydrochloride', 'Diltiazem Hydrochloride', 'Risperidone', 'Spironolactone', 'MicroMousse II Alcohol Free Hand Sanitizing Foam', 'Ultimate Daily Face Wash', 'Binon2', 'Med Aid Chest Rub', 'Nitroglycerin', 'Ammonia', 'Kang Alcohol Pads - MEDIUM', 'cool heat', 'KALI MURJATICUM', 'Crayola', 'Cephalexin', 'Anti-Bacterial Hand', 'Zolpidem Tartrate', 'Fenofibrate Hydrochloride', 'Flu RELIEF', 'Ondansetron', 'smart sense pain relief', 'Concha Argentum', 'Jock Itch', 'Carboplatin', 'ULTRA CONTROL', 'Treatment Set TS336851', 'Valium', and 'Povidone-Iodine'. At the bottom of the table are three buttons: 'Cancel', 'Delete', and 'Update'.

The screenshot shows a web application window titled '(on fedora)'. At the top left is a 'Retour' button. The main heading is 'Ajout d'un médicament'. Below the heading are three input fields: 'Id : 101', 'Nom :', and 'Prix :'. At the bottom center is a button labeled 'Ajouter', which is highlighted by a red rectangle.

2.4.3) Ajouter un lot de médicament

Une fois dans la liste des stocks vous pouvez ajouter un lot du médicament à votre usine.

Ensuite, rentrer le nombre de lots que vous souhaitez ajouter et enfin cliquer sur le bouton « ajouter ».

Vous avez acheté vos médicaments, maintenant vous pouvez vous référer au chapitre suivant pour consulter l'historique et vérifier que vous avez bien réalisé votre commande.

(on fedora)

[Retour](#)

Lots disponibles pour un médicament

Num	Quantité	Nom du Médicament	Usine de Production
17	3816	Ili Hand Sanitizer	Altenwerth-MacGyver
104	1550	Ili Hand Sanitizer	Lund LLC

2.4.4) Consulter les stocks de votre usine

Pour accéder à votre historique, vous devez retourner sur le dashboard. Une fois sur cet écran, cliquer sur le bouton « Stock de votre usine». Maintenant, vous pouvez consulter votre historique.

(on fedora)

Bienvenue usine testUsine

[Logout](#)

Voir les médicaments

Médicaments en stock

Retour

(on fedora)

Stock des médicaments de l'usine

Nom de l'usine	Nom du médicament	Nombre de lots	Prix du médicament (en €)
Lind LLC	Diltazem Hydrochloride		1035.34
Lind LLC	Plan B		1300127.24
Lind LLC	Amoxicillin and Clavulanate Potassium		100050.99
Lind LLC	Spiroolactone		755983.72
Lind LLC	Bi Hand Sanitizer		1550120.99
Lind LLC	Gabapentin		1211125.99
Lind LLC	Omeprazole		125100.86
Lind LLC	Bion3		507.99
Lind LLC	Risperidone		150073.73
Lind LLC	Thyrolar		0150.73

3) Explication du code

3.1) Sécurisation Login

L'application est sécurisée grâce à une authentification. Cette dernière manipule et vérifie les informations de connexion de l'utilisateur. Lors de la vérification du mot de passe, on compare des hash pour éviter que le mot de passe stocké en base de données soit intercepté et dérobé.

Voici ci-dessous le dépendance utilisé pour le hashage des mots de passe :

```
<!-- Password Encodage -->
<dependency>
  <groupId>org.springframework.security</groupId>
  <artifactId>spring-security-crypto</artifactId>
  <version>3.1.0.RELEASE</version>
</dependency>
```

```

private User doConnexion(String id, String password) {
    BCryptPasswordEncoder passwordEncoder = new BCryptPasswordEncoder();
    fenetreLogin.getLabelIncorrect().setText("Identifiant ou mot de passe incorrect");
    try {
        User user = daou.findUserByLogin(id);
        if (user.getUsername().equals(id)) {
            if (passwordEncoder.matches(password, user.getPassword())) {
                if ("usine".equals(user.getRoleBean().getLibelle())) {
                    FenetreHomeUsine fenetreHomeUsine = new FenetreHomeUsine(user);
                    fenetreLogin.setVisible(false);
                    fenetreHomeUsine.setVisible(true);
                    Fenetre01 fenetre01 = new Fenetre01(user);
                    FenetreStock fenetreStock = new FenetreStock();
                    new HomeController(fenetre01, fenetreLot, fenetreHomeUsine, fenetreLogin, fenetreStock, session, user);
                    return user;
                }
                if ("client".equals(user.getRoleBean().getLibelle())) {
                    FenetreHistoriqueCommande fenetreHistoriqueCommande = new FenetreHistoriqueCommande();
                    FenetreHomeUser fenetreHomeUser = new FenetreHomeUser(user);
                    fenetreLogin.setVisible(false);
                    fenetreHomeUser.setVisible(true);
                    Fenetre01 fenetre01 = new Fenetre01(user);
                    new HomeController(fenetre01, fenetreLot, fenetreHomeUser, fenetreLogin, fenetreHistoriqueCommande, session, user);
                    return user;
                }
            }
        }
    }
}

```

3.2) Hibernate

Hibernate est un framework de persistance objet-relational (ORM) pour les projets Java. Son objectif principal est de faciliter la communication et la manipulation des données entre une application Java et une base de données relationnelle. Au lieu d'écrire des requêtes SQL manuellement et de gérer la correspondance entre les objets Java et les tables de la base de données, Hibernate offre un moyen de travailler avec des objets Java directement.

Dans ce projet, nous avons mis en place ce framework afin de faciliter l'utilisation des objets pour chaque entité.

```

@Entity
@Table(name="MEDICAMENT")
@NamedQuery(name="Medicament.findAll", query="SELECT m FROM Medicament m")
public class Medicament implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    private int id;

    @Column(name="Nom")
    private String nom;

    @Column(name="Prix")
    private BigDecimal prix;

    //bi-directional many-to-one association to Lot
    @OneToMany(mappedBy="medicament")
    private List<Lot> lots;

    public Medicament() {
    }

    public Medicament(int id, String nom, BigDecimal prix) {
        super();
        this.id = id;
        this.nom = nom;
        this.prix = prix;
    }
}

```

3.3) DAO Générique

Un DAO générique, ou Data Access Object générique, est une classe qui fournit des opérations de base pour accéder aux données d'une entité dans une application. Il est conçu pour être réutilisable et flexible, en fournissant une implémentation générique pour effectuer les opérations CRUD (Create, Read, Update, Delete) sur une entité donnée, sans avoir à écrire une DAO spécifique pour chaque entité. Il permet aux développeurs d'éviter la duplication de code en fournissant une implémentation de base pour les opérations courantes sur les entités, telles que l'insertion, la récupération, la mise à jour et la suppression. Cependant, des DAO Spécifiques pour chaque entité reste nécessaires pour effectuer certaines opérations spécifique à un objet

```
package DAO;

import java.sql.PreparedStatement;

public abstract class DAOGeneric<T> {

    protected Session session;
    protected Class<T> entityClass;

    public DAOGeneric(Session session, Class<T> entityClass) {
        super();
        this.session = session;
        this.entityClass = entityClass;
    }

    public T find(int id) {
        return session.get(entityClass, id);
    }

    public void save(T entity) {
        session.getTransaction().begin();
        session.save(entity);
        session.getTransaction().commit();
    }

    public void update(T entity) {
        session.getTransaction().begin();
        session.update(entity);
        session.getTransaction().commit();
    }
}
```

```

public void delete(T entity) {
    session.getTransaction().begin();
    session.delete(entity);
    session.getTransaction().commit();
}

public List<T> findAll() {

    Criteria criteria = session.createCriteria(entityClass);
    List<T> l = criteria.list();

    return l;
}

public int count() {
    Criteria criteria = session.createCriteria(entityClass);
    criteria.setProjection(Projections.rowCount());
    Long ligne = (long) criteria.uniqueResult();

    return (int)( (long) ligne);
}
}

```