

Documentation du projet iBuySU.com

Membres de l'équipe:

TEYEB Nidhal - MOUTYMBU Ugo - GASSA Naidath - KANDJEE Rizvan - ORUC Sinem - WEHBE Chifaa

Manuel pour lancer l'application:

Dézippez le fichier code.zip, et déplacez vous dans un terminale LINUX (/code)

Pour compiler : javac src /*.java -d bin

Pour exécuter : java -cp bin Systeme

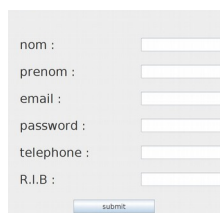
Interface Utilisateur:

L'interface utilisateur est composée de plusieurs pages générées automatiquement selon l'action effectuée par l'utilisateur (méthode addActionListener()). Toutes les pages de l'application héritent de la classe Page

1. **PageAccueil:** Page visible par tous les utilisateurs (connectés ou non) affichant tous les articles en ligne.
2. **PageArticle:** Page affichée après avoir cliqué sur l'article choisi par l'acheteur
3. **PageAuthenticate:** Page permettant de s'inscrire ou de se connecter (choix entre "Sign in" et "Log in").



4. **PagePoster:** Formulaire d'information à remplir pour qu'un vendeur puisse poster l'article à vendre (titre de l'article, prix, description, état...).
5. **PageResultSearch:** Page affichée après avoir effectué une recherche par catégorie ou par mot-clé dans la barre de recherche.
6. **PageSignin :** Page permettant de s'inscrire sur le site en tant que vendeur ou acheteur. Cette page contient 2 boutons "acheteur" et "vendeur" permettant de choisir sur quel type de profil on souhaite s'inscrire.
7. **PageSigninAcheteur:** Page permettant de s'inscrire en tant qu'Acheteur sur l'application. Cette page apparaît après avoir cliqué sur le bouton "acheteur" sur la page "PageSignin".
8. **PageSigninVendeur:** Page permettant de s'inscrire en tant que Vendeur sur l'application. Cette page apparaît après avoir cliqué sur le bouton "vendeur" sur la page "PageSignin".
9. **PageUser:** Page profil affichant toutes informations personnelles de l'utilisateur connecté.

A screenshot of a registration form. It contains six text input fields labeled 'nom', 'prenom', 'email', 'password', 'telephone', and 'R.I.B'. Below these fields is a light blue button labeled 'submit'.

Barre de menu:

Une barre de menu est constamment affichée en haut des pages contenant:

1. Le **logo du site** (en haut tout à gauche) est un bouton permettant de retourner à la page d'accueil.
2. Un **menu** déroulant "Catégorie" affichant les catégories d'articles et permettant de choisir quelle catégorie que l'on souhaite voir.
3. La **barre de recherche** permettant d'effectuer des recherches par mots-clés.
4. Un **bouton "Authenticate"** (en haut tout à droite) permettant de se connecter à l'application. Ce bouton se transforme en un bouton "Profil" permettant à l'utilisateur connecté d'accéder à sa page profil (PageUser).



Stockage des informations:

Le stockage des informations est géré dans la classe Memory à travers des HashSet et HashMap. Un HashMap est un dictionnaire utilisé dans le langage Java associant une clé à une valeur. Cette classe comporte:

1. Un **HashSet users**: c'est un ensemble qui contient tous les utilisateurs connectés de l'application.
2. Un **HashSet articles**: c'est un ensemble qui contient tous les articles postés sur l'application.
3. Un **HashMap ventes**: c'est un dictionnaire associant un vendeur à l'ensemble des articles postés par ce vendeur.
4. Un **HashMap mots_cles**: ce dictionnaire associe un mot-clé aux articles correspondant à ce mot-clé.
5. Un **HashMap evaluations**: permet d'évaluer un utilisateur. Ce dictionnaire associe une paire d'utilisateurs à une note et un commentaire.
6. Un **HashMap catégories**: ce dictionnaire associe une catégorie aux articles correspondant à cette catégorie.

Côté Système:

La classe Systeme représente le point d'entrée du programme, qui est le lien entre l'humain et la machine. Cette classe vérifie :

1. La connexion/déconnexion d'un utilisateur
2. Le format des données entrées lors de l'inscription (e-mail valide ou non)

Les erreurs (erreur de format, mot de passe incorrect,...) sont affichées par des popups gérés par la classe Fenetre.

De plus, on remplit le site avec les articles postés (visibles sur la page d'accueil).

La classe Fenetre gère le visuel de l'application. Elle permet d'afficher toutes les pages en fonction du clic de l'utilisateur. La méthode keyPressed, par exemple, gère les boutons et la redirection des pages.