



SYSTEME DE CLAVARDAGE DISTRIBUE INTERACTIF MULTI-UTILISATEUR TEMPS REEL

Programmation orientée objet

SETTELEN Antoine et TIMANI Mario

UF COO et POO



SYSTEME DE CLAVARDAGE DISTRIBUE INTERACTIF MULTI-UTILISATEUR TEMPS REEL

Programmation orientée objet

SETTELEN Antoine et TIMANI Mario
UF COO et POO

Table des matières

INTRODUCTION	4
MOTIVATION DE CHOIX TECHNOLOGIQUES ET CONCEPTUELS.....	5
I. Choix techniques	5
1. Choix sur le stockage de données applicatives	5
2. Choix sur la signalisation entre les utilisateurs du Chat System	5
a. Façon décentralisée :.....	5
b. Façon centralisée (Serveur de présence) :.....	5
II. Choix conceptuels.....	6
1. Choix de création et de modification du compte utilisateur	6
MANUEL D'UTILISATION DU PRODUIT	7
I. Installation et configuration du Chat System	7
1. Création du dossier d'installation du Chat System	7
2. Configuration du fichier de paramétrage <i>settings.ini</i>	7
3. Configuration et utilisation d'un serveur de présence (exemple de configuration avec Tomcat)	8
a. Installation des données du serveur :	8
b. Lancement et test du serveur :	9
c. Sources d'erreurs de déploiement possibles :.....	10
II. Utilisation du chat	10
1. Cas normal d'utilisation de l'application de chat	10
a. Lancement de l'application :	10
b. Création du compte utilisateur :.....	11
c. Connexion au chat :	11
d. Utilisation du chat :.....	11
2. Modification des informations d'utilisateur	12

INTRODUCTION

Ce rapport présente en profondeur l'ensemble des choix techniques et conceptuels effectués qui nous ont permis de concevoir et implémenter le système de clavardage.

Ce système va permettre aux utilisateurs de la compagnie de communiquer en s'envoyant et en réceptionnant des messages lorsqu'ils seront connectés sur leur propre réseau d'entreprise. Les messages seront textuels mais pourront aussi se composer d'images, de programmes et de tout type de documents.

Le répertoire GitHub <https://github.com/mtimani/POO.git> contient tout le code source écrit pour créer une application de clavardage dans le cadre de l'UF POO.

Il contient l'ensemble des fichiers JAVA (dossier `src`), des ressources (dossier `res`) et des données pour tester le serveur de présence. Le dossier `jars` contient, lui, les librairies utilisées lors de la réalisation du projet.

Le fichier `chatsystem.jar` correspond à l'application prête à être utilisée.

MOTIVATION DE CHOIX TECHNOLOGIQUES ET CONCEPTUELS

I. Choix techniques

1. Choix sur le stockage de données applicatives

Chat System est une application de clavardage avec un système de stockage qui doit être décentralisé, conformément au cahier des charges. Lors de la conception de l'application, nous avons fait le choix de ne pas déployer une base de données pour stocker les données applicatives. La solution que nous avons décidée de mettre en place pour le stockage de données est plus simple et efficace du point de vue du déploiement de l'application et de l'espace mémoire occupé par les données utilisateurs par la non-gestion d'une base de données externe. La méthode est : **sérialisation d'objets et leur stockage en binaire dans des conteneurs .bin.**

Un des désavantages majeurs des fichiers .bin est le fait qu'il faut reconvertir l'intégralité du fichier afin de pouvoir récupérer ne serait-ce qu'un bout d'information stockée dessus. Néanmoins, dans le cadre de cette application il nous faut tout de même charger l'intégralité des données afin de la faire fonctionner correctement. Ainsi, le fait de sauvegarder les données applicatives sous la forme de conteneurs .bin ne présente aucun désavantage, mais au contraire est plus optimal du point de vue du stockage.

2. Choix sur la signalisation entre les utilisateurs du Chat System

a. Façon décentralisée :

La signalisation de connexion, confirmation de connexion, déconnexion, changement de username se fait via des paquets UDP envoyés sur le réseau par chaque utilisateur souhaitant communiquer.

b. Façon centralisée (Serveur de présence) :

Si un utilisateur souhaite utiliser l'application avec un serveur de présence (le déploiement et la configuration du serveur Tomcat sont décrites dans la partie « Installation et configuration du Chat System »), il peut simplement l'indiquer dans le fichier settings.ini. Dans ce cas, la signalisation de connexion, confirmation de connexion, déconnexion, changement de username se fera via des requêtes http envoyés au serveur de présence par chaque utilisateur souhaitant communiquer. De manière périodique (paramètre *update_interval* dans le fichier settings.ini) la machine utilisatrice du Chat System va demander la liste d'utilisateurs connectés au Chat System auprès du serveur Tomcat.

La communication par messages, fichiers ou images entre les utilisateurs de l'application s'effectue toujours en utilisant le protocole TCP.

II. Choix conceptuels

1. Choix de création et de modification du compte utilisateur

Un utilisateur qui possède un compte sur une machine est protégé grâce à un mot de passe. De ce fait, si un utilisateur souhaite modifier son username ou son mot de passe, il lui faut d'abord se connecter à l'application avec son ancien login/mot de passe et modifier son login/mot de passe par la suite. Tous les utilisateurs de l'application verront que l'username a changé. La création d'un nouveau compte sur l'application entraînera une suppression instantanée des données utilisateur. Toutes les sessions de clavardages seront ainsi supprimées. Ceci est un choix conceptuel que nous avons voulu mettre en place (sachant que si jamais il nous ne faut pas supprimer ces données, nous aurons besoin de commenter trois lignes).

MANUEL D'UTILISATION DU PRODUIT

I. Installation et configuration du Chat System

1. Création du dossier d'installation du Chat System

Veillez noter que la version jdk utilisée est 11.0.5 et la version du serveur Tomcat est 7.0

L'application de clavardage est un fichier *.jar* (*ChatSystem.jar*). Il est nécessaire de copier ce fichier au sein du dossier créé pour l'application. Si vous souhaitez personnaliser l'application et régler quelques paramètres, placez le fichier *settings.ini* à l'intérieur du dossier créé pour l'application.

Veillez noter qu'un dossier *data* sera automatiquement créé par l'application lors de la première création de compte.

Si le dossier que vous avez créé pour l'application s'appelle *ChatSystem*, voici l'arborescence que vous devez obtenir à la fin de la première utilisation de l'application (le dossier *data* n'est pas obligatoire pour démarrer l'application, il sera automatiquement créé par l'application) :

```
ChatSystem/  
| — ChatSystem.jar  
| — data/  
|   | — groups.bin  
|   | — messages.bi  
|   | — user.bin  
| — settings.ini
```

2. Configuration du fichier de paramétrage *settings.ini*

Le fichier *settings.ini* permet à l'utilisateur de paramétrer l'utilisation de l'application, ainsi que de l'adapter à ses besoins. Voici un exemple possible de la structure de ce fichier :

```
[general]  
theme=dark  
use_server=0
```

```
[udp]  
port=7000
```

```
[server]  
ip=10.1.5.299  
port=8080  
timeout=5000
```

update_interval=500

Le paramètre **theme** de la section *[general]* permet à l'utilisateur de sélectionner le thème utilisé dans l'application. Deux thèmes sont disponibles : un thème sombre (*dark*) et un thème clair (*light*, utilisé par défaut si le paramètre theme n'est pas spécifié dans le fichier settings.ini).

Le paramètre **use_server** de la section *[general]* permet à l'utilisateur de sélectionner le mode de fonctionnement de l'application.

- En mettant la valeur 1, le mode centralisé et plus spécifiquement le **mode avec serveur** sera utilisé par l'application ;
- Tout autre valeur indiquée permettra d'utiliser le **mode UDP** pour la transmission d'annonces utilisatrices.

La section *[udp]* permet à l'utilisateur de configurer l'utilisation du service UDP pour les annonces utilisatrices sur le réseau. Vous pouvez ainsi sélectionner le port à utiliser à l'aide du paramètre *port*. Il est important de noter que si ce paramètre est spécifié, il faut que ce paramètre (numéro de port) soit identique dans les paramètres de chaque machine sur le réseau qui utilise l'application, afin que ces utilisateurs puissent communiquer entre eux.

Si vous avez utilisé la valeur 1 pour le paramètre **use_server**, vous devez détailler d'avantage quelques paramètres dans le fichier settings.ini afin de pouvoir utiliser le serveur à l'aide de la section *[server]*.

Cette section contient 4 paramètres :

- *ip* : correspond à l'adresse IP de la machine sur laquelle est déployé le serveur Tomcat. Si un utilisateur souhaite utiliser l'application sur la machine sur laquelle ce serveur Tomcat est déployé, il peut spécifier localhost ou 127.0.0.1 pour ce paramètre ;
- *port* : c'est le port sur lequel le serveur est déployé. Par défaut, Tomcat se déploie sur le port 8080. Il est fortement conseillé de changer ce port lors du déploiement du serveur Tomcat dans le cas où votre machine héberge plusieurs serveurs Web comme Apache, NGINX ou Django ;
- *timeout* : c'est le délai maximal que peut prendre la tentative de connexion au serveur, en millisecondes (ms). Passé ce délai, la connexion échouera et l'application renverra une erreur indiquant que le serveur est indisponible ;
- *update_interval* : il s'agit de la durée écoulée entre deux requêtes que l'application de l'utilisateur effectue auprès du serveur, en millisecondes (ms).

3. Configuration et utilisation d'un serveur de présence (exemple de configuration avec Tomcat)

*Cette section indique comment configurer et utiliser un serveur de présence **Tomcat**.*

*La variable **\$CATALINA_HOME** correspond au répertoire d'installation de **Tomcat**.*

a. Installation des données du serveur :

Pour utiliser l'application avec la présence d'un serveur Tomcat, il faut qu'une machine du réseau soit configurée pour faire office de serveur.

Le dossier *Chat* fourni contient toutes les données nécessaires pour faire fonctionner l'application ChatSystem avec **Tomcat**. Voici le contenu partiel de ce dossier :

```
Chat/
|— WEB-INF/
|   |— lib/
|       | - ChatSystem.jar
|       | - ChatSystem.war
|       |— gson-2.6.2.jar
|       | - ini4j-0.5.4.jar
|       | - javax.servlet.jar
|   |— web.xml
| - META-INF/
| - MANIFEST.MF
```

Le répertoire *Chat* doit être copié dans son intégralité au sein du dossier *webapps* du dossier contenant l'installation de Tomcat.

- Le sous-dossier *lib* contient le .jar de l'application, le .war de l'application, ainsi que les diverses bibliothèques externes utilisées par l'application et le serveur ;
- Enfin, le fichier *web.xml* contient les informations de configuration de la servlet. Ce fichier est utilisé par le serveur Tomcat par la suite.

Si vous utilisez Tomcat, vous ne devriez pas avoir à faire plus de configurations.

b. Lancement et test du serveur :

Dès que le serveur est installé et correctement configuré, vous pouvez le lancer grâce à la commande :

```
sudo $CATALINA_HOME/bin/startup.sh
```

Une option plus facile est d'importer l'intégralité du projet dans Eclipse JEE et lancer le serveur, ainsi que l'application depuis Eclipse.

Une fois le serveur lancé, vous pouvez tester son fonctionnement. Pour cela, ouvrez un navigateur internet et allez à l'adresse suivante :

- <http://X.X.X.X:8080/chat/ChatServer?test> (où X.X.X.X est l'adresse IP de la machine supportant le serveur ou localhost si le ChatSystem sera lancé sur la même machine) depuis toute machine souhaitant utiliser l'application.

Remarque : le port 8080 est utilisé par défaut par le serveur Tomcat, si vous avez déployé le serveur Tomcat en utilisant un autre numéro de port, utilisez celui que vous avez choisi. N'oubliez pas de spécifier le bon numéro de port du serveur dans le fichier settings.ini.

Vous devriez avoir comme réponse l'élément *json* suivant :

```
{"code":0,"dataFormat":"json","data":"Le serveur fonctionnement correctement."}
```

Si vous souhaitez arrêter le serveur, utilisez la commande suivante :

```
sudo $CATALINA_HOME/bin/shutdown.sh
```

c. Sources d'erreurs de déploiement possibles :

Si après avoir configuré et démarré le serveur Tomcat, vous n'êtes pas parvenu à obtenir l'affichage *json* correct, vérifiez les points suivants :

- Assurez-vous d'avoir copié l'intégralité du dossier *Chat* dans le dossier *webapps* de **Tomcat** ;
- Vérifiez que les configurations de **Tomcat** et du chat sont les mêmes ;
- Si une erreur *404* survient, c'est que le fichier *.jar* n'est pas présent dans le dossier *webapps* ou que le descripteur de la servlet (*web.xml*) contient des données incorrectes ;
- Si une erreur *500* survient (et que tous le *.jar* de l'application a correctement été copié), il est probable que le fichier *gson-2.6.2.jar* soit manquant dans le dossier *\$CATALINA_HOME/webapps/Chat/WEB-INF/lib/* ;
- Si le déploiement du serveur fonctionne sur une machine mais pas sur une autre, vérifiez les paramètres du firewall, qui pourrait être à l'origine du filtrage de certaines connexions (notamment pour le port *8080* ou pour le port sur lequel vous avez essayé de déployer votre serveur Tomcat).

II. Utilisation du chat

Cette section détaille l'utilisation de l'application ChatSystem, après installation et configuration.

1. Cas normal d'utilisation de l'application de chat

Pour démarrer le chat et discuter avec d'autres utilisateurs :

a. Lancement de l'application :

1. Si vous souhaitez utiliser l'application avec le **mode serveur de présence**, lancez d'abord le serveur Tomcat sur une machine ;
2. Modifiez la configuration du fichier *settings.ini* afin que le mode serveur soit activé (veuillez voir la partie "**Installation et configuration préalable**" du serveur de présence)

Remarque : une fois que cette modification est faite il ne sera plus nécessaire dans le futur de remodifier ce fichier ;

3. Lancez l'application ;

b. Création du compte utilisateur :

4. Lors du premier lancement de l'application, il vous faut créer un nouvel utilisateur en cliquant sur le bouton « CREER UTILISATEUR » et ensuite indiquer un nom d'utilisateur et un mot de passe dans la fenêtre apparue ;

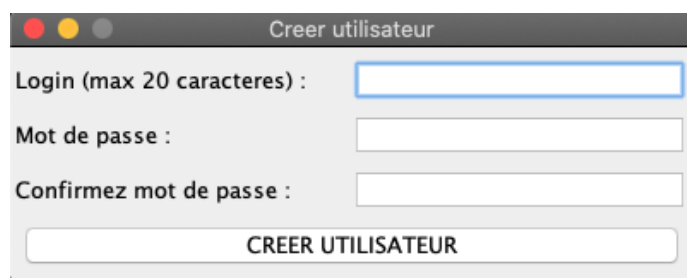


Figure 1 : Image illustrant la fenêtre de création de compte de l'application

c. Connexion au chat :

5. Sur la fenêtre de connexion, choisissez l'adresse IP du réseau souhaité (ignorez cette option si vous n'êtes connecté qu'à un seul réseau) puis rentrez votre login ainsi que votre mot de passe, que vous avez choisi lors de la création de votre compte ;

d. Utilisation du chat :

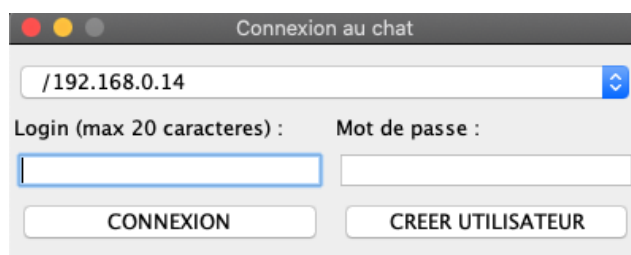


Figure 2 : Image illustrant la fenêtre de connexion de l'application de chat

6. Connectez-vous (en appuyant sur le bouton « CONNEXION ») ;

7. Vous pouvez désormais communiquer avec les autres utilisateurs connectés à l'application de chat en les sélectionnant dans la liste des utilisateurs connectés.

Remarque : vous pouvez également échanger des fichiers et images en appuyant sur le bouton « Envoyer un fichier » ;

8. Vous pouvez supprimer l'historique d'une conversation démarrée en appuyant sur le bouton « Supprimer la conversation » ;

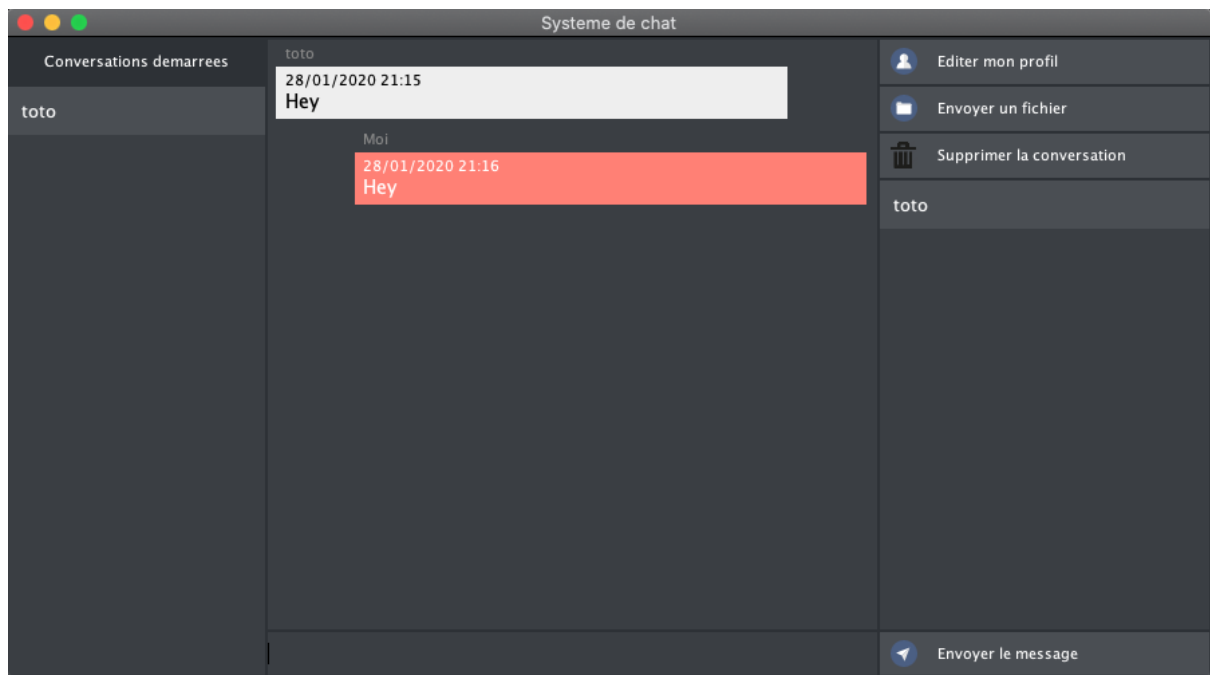


Figure 3 : Image illustrant l'application après connexion et après démarrage d'une conversation

2. Modification des informations d'utilisateur

Afin de changer de nom d'utilisateur et/ou de mot de passe :

1. Cliquez sur le bouton "Editer mon profil" lorsque vous vous êtes connecté ;
2. Vous avez ensuite la possibilité de modifier les informations vous concernant.

Remarque : veuillez noter que le login et mot de passe modifiés sont à mémoriser, car lors de la prochaine connexion se sont ces derniers qui seront utilisés.

Veuillez noter que si vous créez un nouveau compte, toutes les données vont être supprimées. Si vous souhaitez modifier votre login et votre mot de passe sans perdre les discussions il vous faut d'abord vous connecter en utilisant votre ancien login et mot de passe et modifier ces derniers grâce à la forme « Editer mon profil ».

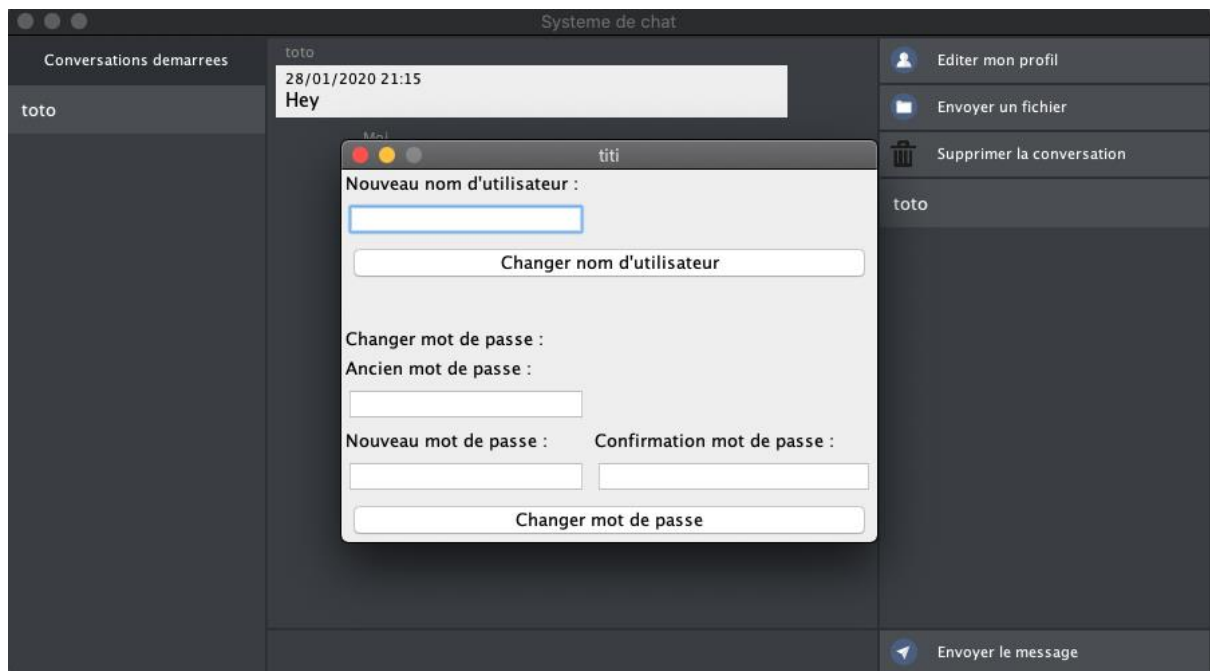


Figure 4 : Image illustrant la fenêtre d'édition de login et mot de passe

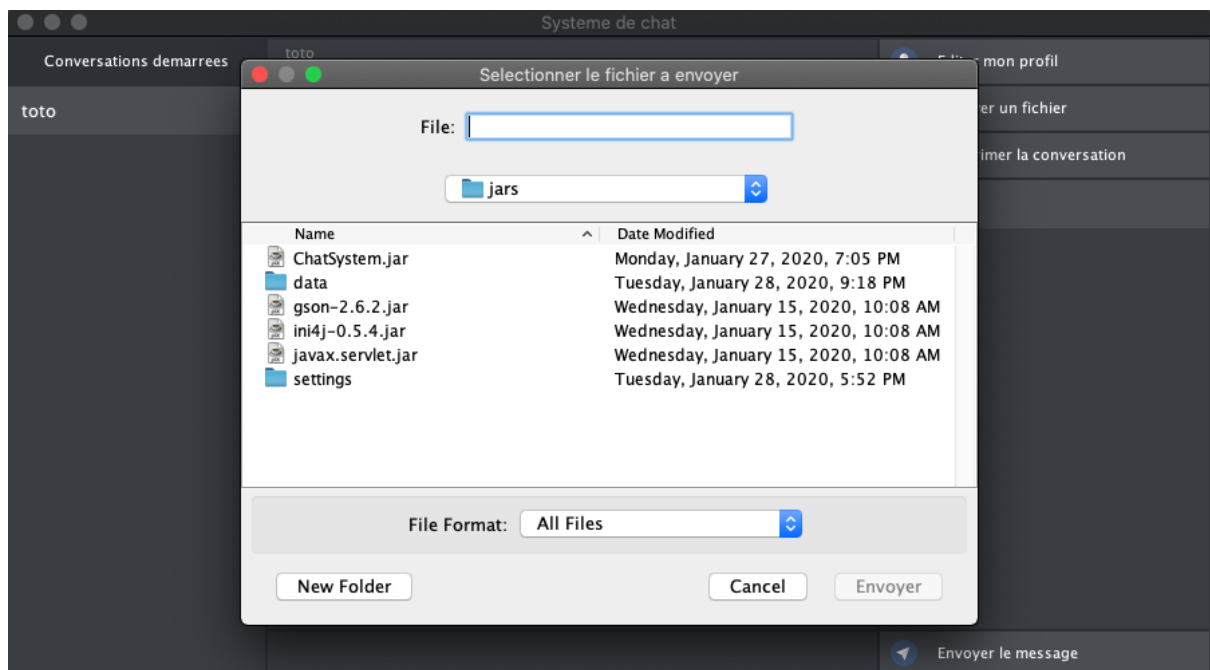


Figure 5 : Image illustrant la fenêtre de choix de fichier ou image à envoyer

INSA Toulouse

135, avenue de Rangueil
31077 Toulouse Cedex 4 - France
www.insa-toulouse.fr



MINISTÈRE
DE L'ÉDUCATION NATIONALE,
DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE