

Logiciel « Chat »

Spécifications

Auteur	Contact
Matthias Brun	matthias.brun@eseo.fr
Bastien Potiron	bastien.potiron@e-cosi.com

Relecteur	Contact
Camille Constant	camille.constant@eseo.fr

Version	Date	Commentaire
0.5	01/02/2017	Modification du diagramme de CU ;
0.4	13/01/2017	Modification CU sortir du chat ;
0.3	13/01/2013	Ajout CU Fenetre de connexion ;
0.2	09/01/2013	Modification CU sortir du chat ;
0.1	08/01/2013	Ajout CU sortir du chat ;
		Modification CU lancement client chat → entrer dans le chat ;
		Modification IHM : Messages du serveur.
		Ajout CU lancement client chat ;
		Ajout pré-conditions aux CU ;
		Modification IHM : Messages du serveur.
		Introduction, Déploiement, CU, IHM.

Table des matières

1	Introduction	3
1.1	Objectif des spécifications	3
1.2	Portée des spécifications	3
1.3	Modification des spécifications	3
2	Déploiement	4
3	Cas d'utilisation	5
3.1	Communiquer avec d'autres clients	5
3.2	Connexion au chat	6
3.3	Entrer dans le chat	7
3.4	Obtenir de l'aide	7
3.5	Échanger des messages	8
3.6	Changer de surnom	8
3.7	Afficher des informations	9
3.7.1	Afficher ses informations personnelles	9
3.7.2	Afficher les informations sur les canaux	9
3.8	Gérer des canaux de discussion	10
3.8.1	Créer un canal de discussion	10
3.8.2	Changer de canal de discussion	11
3.8.3	Supprimer un canal de discussion	12
3.9	Sortir du chat	13
4	Interface homme-machine	14
4.1	Interface homme-machine de Felix	14
4.2	Protocole de communication client/serveur	15
4.3	Messages du serveur	15

1 Introduction

1.1 Objectif des spécifications

Ce document spécifie un ensemble logiciel composé d'un serveur et de clients chat. Plusieurs clients chat peuvent se connecter au serveur chat et échanger des messages textuels via ce serveur.

Ce document décrit en particulier :

- **le déploiement** prévu des composants serveur et clients chat ;
- **les cas d'utilisation** attendus pour cet ensemble logiciel ;
- **l'interface homme-machine** envisagée pour un client ainsi que le protocole de communication client/serveur et les messages renvoyés par le serveur.

1.2 Portée des spécifications

Ce document concerne tous les intervenants du projet.

1.3 Modification des spécifications

Les spécifications pourront être soumises à des modifications. Toute nouvelle version sera signalée par mail à l'ensemble des intervenants du projet.

2 Déploiement

La figure 1 décrit le déploiement du serveur et des clients chat.

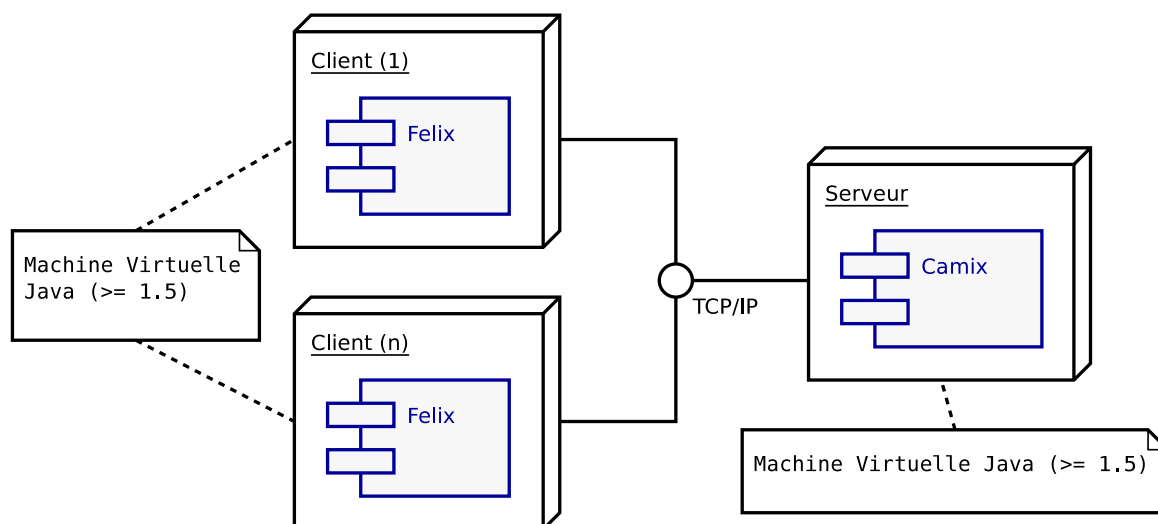


FIGURE 1 – Diagramme de déploiement du chat.

Le composant **Camix** est le composant logiciel serveur du chat. Les composants **Felix** sont les composants clients du chat.

Les composants Camix et Felix sont programmés en Java. Leurs exécutions nécessitent une machine virtuelle Java (ou *Java Virtual Machine* - JVM) en version supérieure ou égale à 1.5.

Les composants Felix et Camix communiquent via un réseau TCP/IP (*Transmission Control Protocol/Internet Protocol*).

3 Cas d'utilisation

La figure 2 présente les cas d'utilisation du chat tels que décrits ci-dessous.

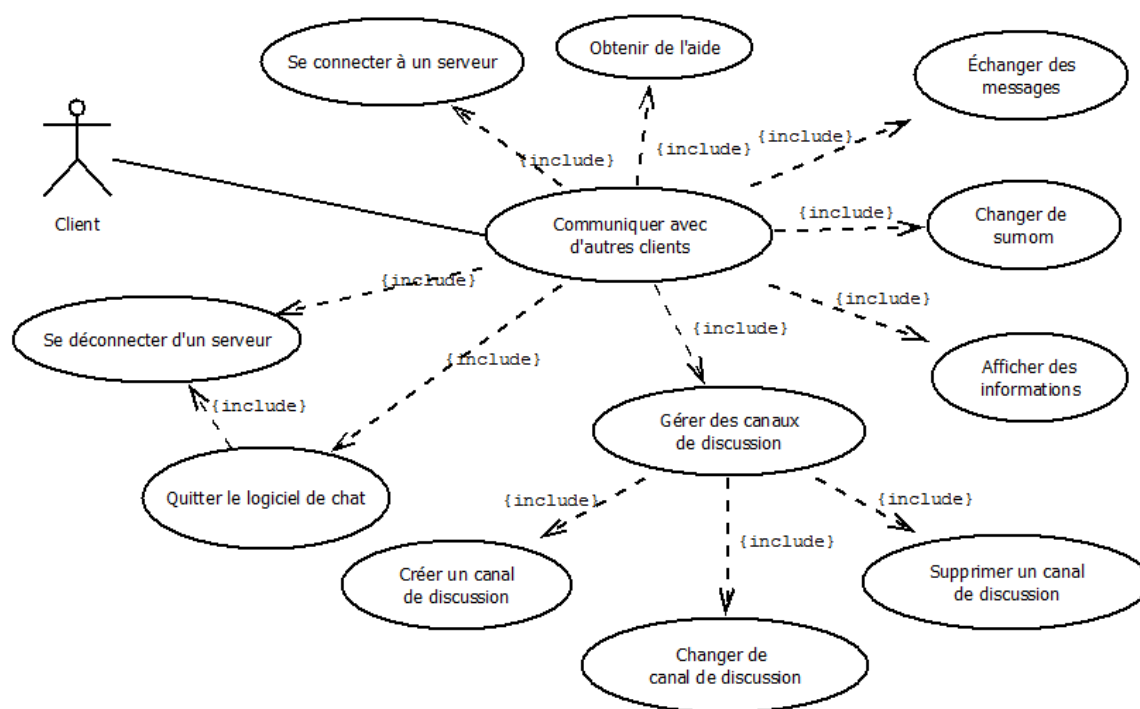


FIGURE 2 – Cas d'utilisation du chat.

Chacun de ces cas d'utilisation est détaillé dans la suite de cette section.

3.1 Communiquer avec d'autres clients

L'objectif du chat est de permettre à des clients de communiquer entre eux. Après être entré dans le chat, cette communication se fait par échanges de messages textuels. Chaque client est identifié par les autres clients via un surnom qu'il peut changer au cours de son utilisation du chat. Les échanges de messages se font au sein de canaux de discussion dont une gestion est possible par chaque client (création, suppression et changement de canal). Enfin, le chat fournit une aide aux utilisateurs ainsi que des informations sur les canaux existants et sur les clients eux-mêmes. La sortie du chat se fait en fermant le client chat (Felix) ou en tapant la commande '/q' qui ramène alors l'utilisateur sur la fenêtre de connexion.

Remarque (1) : Un serveur chat (Camix) compatible avec le client chat (Felix) doit être lancé pour permettre aux différents clients de communiquer.

Remarque (2) : Pour utiliser les différentes fonctionnalités du chat un protocole est à la disposition des clients. Ce protocole (spécifié dans la section 4.2, page 15) définit un ensemble de commandes que peut invoquer un client.

Cas d'utilisation : Communiquer avec d'autres clients.

Résumé : Un client lance un client chat pour entrer dans le chat, il peut ensuite obtenir de l'aide sur les commandes du chat, échanger des messages, changer de surnom, afficher des informations, gérer des canaux de discussion et sortir du chat.

Niveau : Stratégique.

Acteur principal : Client.

Pré-conditions : Un logiciel serveur (Camix) est lancé et accessible par le réseau TCP/IP.

Scénario nominal :

1. Connexion au chat (cf. 3.2, page 6).
2. Entrer dans le chat (cf. 3.3, page 7).
3. Obtenir de l'aide (cf. 3.4, page 7).
4. Échanger des messages (cf. 3.5, page 8).
5. Changer de surnom (cf. 3.6, page 8).
6. Afficher des informations (cf. 3.7, page 9).
7. Gérer des canaux de discussion (cf. 3.8, page 10).
8. Sortir du chat (cf. 3.9, page 13).

3.2 Connexion au chat

Cas d'utilisation : Connexion au chat.

Résumé : Un client tente de se connecter au serveur de chat Camix après avoir indiqué l'adresse IP et le port du serveur.

Acteur principal : Client.

Pré-conditions : Aucune. Pour que la connexion puisse se faire (scénario nominal), un serveur Camix doit être accessible (lancé et accessible via le protocole IP).

Scénario nominal :

1. Le client lance l'exécution du composant Felix.
2. Le client clique sur le bouton de connexion au chat.
3. La fenêtre affiche la tentative de connexion au serveur.
4. Le client Felix se connecte en TCP/IP au serveur Camix via les informations présentées dans le fichier de configuration par défaut.
5. La fenêtre de connexion se ferme.

Alternative [L'utilisateur veut modifier l'adresse IP et le port par défaut de connexion] :

2. L'utilisateur indique l'adresse et/ou le port du serveur auquel il veut se connecter dans la fenêtre de connexion.
3. Le client clique sur le bouton de connexion au chat.

4. La fenêtre affiche la tentative de connexion au serveur.
5. Le client Felix se connecte en TCP/IP au serveur Camix via les informations rentrées par l'utilisateur.
6. La fenêtre de connexion se ferme.

Alternative [La connexion au serveur échoue] :

6. La fenêtre affiche un message d'information stipulant l'impossibilité de se connecter au serveur Camix.
7. L'utilisateur peut retenter de se connecter autant de fois que voulu en reprenant le scénario nominal à l'étape 2.

3.3 Entrer dans le chat

Cas d'utilisation : Entrer dans le chat.

Résumé : Un client entre dans le chat.

Acteur principal : Client.

Acteurs secondaires : Les autres utilisateurs du chat.

Pré-conditions : Le client Felix est connecté en TCP/IP au serveur Camix.

Scénario nominal :

1. La fenêtre de chat s'ouvre.
2. Camix inscrit le client dans le canal par défaut (place publique).
3. Camix informe les composants Felix des autres utilisateurs inscrits dans le canal par défaut que le client arrive dans le chat.
4. Chaque composant Felix concerné affiche un message d'arrivée du client dans le chat.
5. Camix transmet au composant Felix du client un message d'accueil dans le chat.
6. Le composant Felix du client affiche un message d'accueil dans le chat.

3.4 Obtenir de l'aide

Cas d'utilisation : Obtenir de l'aide.

Résumé : Un client peut obtenir de l'aide sur les commandes du chat.

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande d'aide.
2. Felix transmet la commande d'aide à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.5 Échanger des messages

Cas d'utilisation : Échanger des messages.

Résumé : Un client échange des messages avec d'autres clients.

Acteur principal : Client.

Acteurs secondaires : Les autres utilisateurs du chat.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit un message textuel.
2. Felix transmet le message à Camix.
3. Camix transmet le message aux composants Felix des utilisateurs inscrits dans le canal du client.
4. Chaque composant Felix concerné affiche le message (fin).

3.6 Changer de surnom

Les clients peuvent avoir des surnoms pour être plus facilement identifiés dans le chat. Le surnom par défaut d'un client est un point d'interrogation (?).

Cas d'utilisation : Changer de surnom.

Résumé : Un client change de surnom.

Acteur principal : Client.

Acteurs secondaires : Les autres utilisateurs du chat.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande de changement de surnom.
2. Felix transmet la commande de changement de surnom à Camix.
3. Camix change le surnom du client.
4. Camix informe les composants Felix des utilisateurs inscrits dans le canal du client que celui-ci a changé de surnom.
5. Chaque composant Felix concerné affiche un message de changement de surnom du client (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.7 Afficher des informations

Le serveur chat peut fournir à chaque client des informations personnelles (surnom du client et canal occupé) et sur les canaux du chat (noms et nombres d'utilisateurs).

Cas d'utilisation : Afficher des informations.

Résumé : Un client peut faire afficher ses informations personnelles et des informations sur les canaux du chat.

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Afficher ses informations personnelles (cf. 3.7.1, page 9).
2. Afficher les informations sur les canaux (cf. 3.7.2, page 9).

3.7.1 Afficher ses informations personnelles

Cas d'utilisation : Afficher des informations personnelles.

Résumé : Un client peut obtenir ses informations personnelles (surnom et canal occupé).

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande d'informations personnelles.
2. Felix transmet la commande d'informations personnelles à Camix.
3. Camix transmet à Felix les informations sur le client (surnom et canal occupé).
4. Felix affiche les informations personnelles du client (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.7.2 Afficher les informations sur les canaux

Cas d'utilisation : Afficher des informations sur les canaux.

Résumé : Un client peut obtenir des informations sur les canaux de discussion (noms et nombres de clients).

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande d'informations sur les canaux.

2. Felix transmet la commande d'informations sur les canaux à Camix.
3. Camix transmet à Felix les informations sur les canaux (noms et nombres de clients).
4. Felix affiche les informations sur les canaux (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.8 Gérer des canaux de discussion

Le serveur chat offre la possibilité d'échanger des messages dans des canaux de discussion. Les échanges au sein d'un canal restent internes à celui-ci, ils ne sont pas visibles des autres canaux. Le canal par défaut du serveur chat se nomme **place publique**. Le serveur permet de créer et de supprimer des canaux de discussion ainsi que de changer de canal de discussion lors de l'utilisation du chat.

Cas d'utilisation : Gérer des canaux de discussion.

Résumé : Un client peut créer des canaux de discussion, changer de canal de discussion et supprimer des canaux de discussion.

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Créer un canal de discussion (cf. 3.8.1, page 10).
2. Changer de canal de discussion (cf. 3.8.2, page 11).
3. Supprimer un canal de discussion (cf. 3.8.3, page 12).

3.8.1 Créer un canal de discussion

Cas d'utilisation : Créer un canal de discussion.

Résumé : Un client crée un canal de discussion sur le serveur du chat.

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande de création d'un canal.
2. Felix transmet la commande de création du canal à Camix.
3. Camix crée le canal.
4. Camix informe Felix qu'un canal a été créé.
5. Felix affiche un message de création du canal (fin).

Alternative [le canal existe déjà] :

3. Camix informe Felix que le canal existe déjà.
4. Felix affiche un message d'existence du canal (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.8.2 Changer de canal de discussion

Cas d'utilisation : Changer de canal de discussion.

Résumé : Un client change de canal de discussion sur le serveur du chat.

Acteur principal : Client.

Acteurs secondaires : Les autres utilisateurs du chat.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande de changement de canal.
2. Felix transmet la commande de changement de canal à Camix.
3. Camix informe les composants Felix des utilisateurs inscrits dans le canal du client que celui-ci quitte le canal.
4. Chaque composant Felix concerné affiche un message de départ du client du canal.
5. Camix change le client de canal.
6. Camix informe les composants Felix des utilisateurs inscrits dans le nouveau canal du client que celui-ci arrive dans le canal.
7. Chaque composant Felix concerné affiche un message d'arrivée du client dans le canal (fin).

Alternative [le canal n'existe pas] :

3. Camix informe Felix que le canal n'existe pas.
4. Felix affiche un message de non existence du canal demandé (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.8.3 Supprimer un canal de discussion

Cas d'utilisation : Supprimer un canal de discussion.

Résumé : Un client supprime un canal de discussion sur le serveur du chat.

Acteur principal : Client.

Pré-conditions : Le composant client du chat (Felix) est lancé et connecté par TCP/IP au composant serveur du chat (Camix).

Scénario nominal :

1. Le client saisit la commande de suppression d'un canal.
2. Felix transmet la commande de suppression du canal à Camix.
3. Camix supprime le canal.
4. Camix informe Felix qu'un canal a été supprimé.
5. Felix affiche un message de suppression du canal (fin).

Alternative [le canal n'existe pas] :

3. Camix informe Felix que le canal n'existe pas.
4. Felix affiche un message de non existence du canal (fin).

Alternative [le canal n'est pas vide] :

3. Camix informe Felix que le canal n'est pas vide.
4. Felix affiche un message de canal non vide (fin).

Alternative [le canal est place publique] :

3. Camix informe Felix que le canal ne peut pas être supprimé.
4. Felix affiche un message de canal non supprimable (fin).

Alternative [la commande n'est pas valide] :

2. Felix transmet la commande à Camix.
3. Camix transmet à Felix les informations sur les commandes disponibles.
4. Felix affiche un message d'aide sur les commandes disponibles (fin).

3.9 Sortir du chat

Cas d'utilisation : Sortir du chat.

Résumé : Un client sort du chat en fermant le logiciel client du chat ou en insérant '/q' dans le chat.

Acteur principal : Client.

Acteurs secondaires : Les autres utilisateurs du chat dans le cas du scénario nominal.

Pré-conditions : Un logiciel serveur (Camix) est lancé et accessible par le réseau TCP/IP. Le client est connecté sur le logiciel serveur.

Scénario nominal :

1. Le client arrête l'exécution du composant Felix.
2. Felix se déconnecte de Camix (ferme la connexion au serveur).
3. Le processus Felix se termine (fermeture de la fenêtre).
4. Camix informe les composants Felix des utilisateurs inscrits dans le canal du client que celui-ci quitte le chat.
5. Chaque composant Felix concerné affiche un message de départ du client du chat.
6. Camix ferme la connexion du client (et l'enlève donc du canal dans lequel il était).

Scénario alternatif [Le client se déconnecte du chat] :

1. Le client saisit la commande de déconnexion du chat ('/q').
2. Felix transmet la commande de déconnexion du chat à Camix.
3. Camix informe les composants Felix des utilisateurs inscrits dans le canal du client que celui-ci quitte le chat.
4. Chaque composant Felix concerné affiche un message de départ du client du chat.
5. Camix ferme la connexion du client (et l'enlève donc du canal dans lequel il était).
6. La fenêtre de chat du client Felix se ferme et Felix ouvre la fenêtre de connexion au chat (retour à la première étape du scénario nominal du cas d'utilisation "Connexion au chat").

4 Interface homme-machine

4.1 Interface homme-machine de Felix

La figure 3 présente l'interface homme-machine (IHM) du client chat Felix.

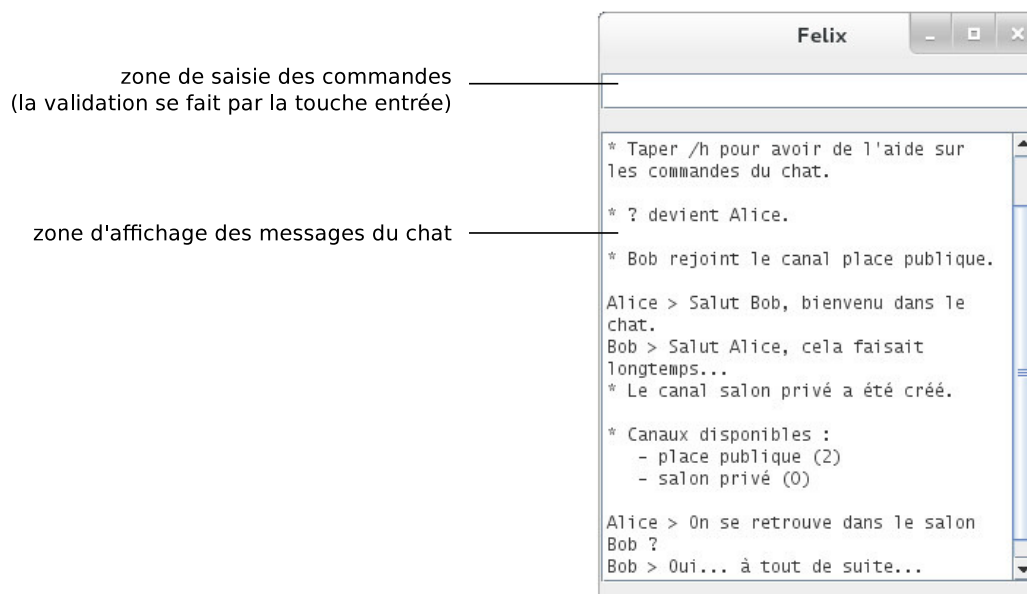


FIGURE 3 – IHM du client chat Felix.

La figure 4 présente l'interface homme-machine (IHM) de la fenêtre de connexion de Felix au serveur de chat.

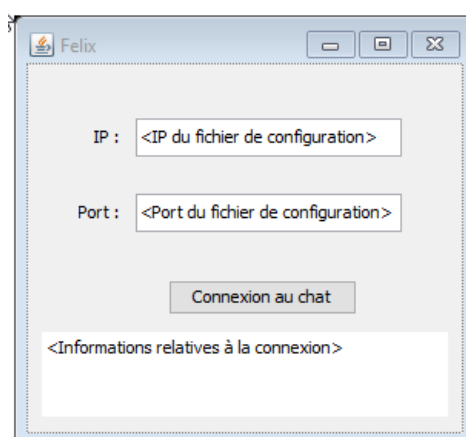


FIGURE 4 – IHM de la fenêtre de connexion de Felix.

4.2 Protocole de communication client/serveur

Pour utiliser les différentes fonctionnalités du chat un protocole est à la disposition des clients. Ce protocole définit un ensemble de commandes que peut invoquer un client.

Le tableau 1 énumère les commandes disponibles dans le chat.

<i>/n surnom</i>	Changer de surnom.
<i>/a nom du canal</i>	Créer un canal de discussion.
<i>/r nom du canal</i>	Supprimer un canal de discussion.
<i>/c nom du canal</i>	Changer de canal de discussion.
<i>/l</i>	Afficher la liste des canaux de discussion.
<i>/?</i>	Afficher ses informations personnelles.
<i>/h</i>	Afficher l'aide sur les commandes du chat.
<i>/q</i>	Déconnexion du chat.

TABLE 1 – Liste des commandes du chat.

4.3 Messages du serveur

L'utilisation du chat prévoit des messages communiqués aux clients par le serveur. Les listings 1, page 15, à 16, page 17 spécifient ces différents messages.

```
* Un nouveau client est dans le chat (place publique).
```

Listing 1 – Message d'arrivée d'un client dans le chat (à destination de la place publique).

```
* Taper /h pour avoir de l'aide sur les commandes du chat.
```

Listing 2 – Message d'accueil dans le chat.

```
* Commandes disponibles :
/n : changer de surnom ;
/c : changer de canal ;
/l : afficher les canaux ;
/a : creer un canal ;
/r : supprimer un canal ;
/? : afficher ses informations ;
/h : afficher l'aide.
/q : deconnexion du chat.
```

Listing 3 – Message d'aide sur les commandes du chat.

```
client_x > Hello World !
```

Remarque : Un message est prefixe par le surnom de l'émetteur suivi de '>'.

Listing 4 – Exemple de message transmis dans le chat.

```
* client_x devient client_y.
```

Listing 5 – Exemple de message de changement de surnom d'un client.

```
* Informations personnelles :  
  Surnom : client_x ;  
  Canal  : place publique ;
```

Listing 6 – Exemple de message sur des informations personnelles.

```
* Canaux disponibles :  
  - canal_x (3)  
  - place publique (2)
```

Remarque : Le nombre de clients dans un canal est donne entre parentheses.

Listing 7 – Exemple de message sur des informations sur les canaux.

```
* Le canal canal_x a ete cree.
```

Listing 8 – Exemple de message de création de canal.

```
* Le canal canal_x existe deja.
```

Listing 9 – Exemple de message de création impossible de canal (le canal existe déjà).

```
* client_x quitte le canal place publique.
```

Listing 10 – Exemple de message de départ d'un canal.

```
* client_x rejoint le canal canal_x.
```

Listing 11 – Exemple de message d’arrivée dans un canal.

```
* Le canal demande n'existe pas.
```

Listing 12 – Message de non existence d’un canal demandé (pour changement de canal)

```
* Le canal canal_x a ete supprime.
```

Listing 13 – Exemple de message de suppression d’un canal.

```
* Le canal canal_y n'existe pas.
```

Listing 14 – Exemple de message de suppression impossible d’un canal (le canal n'existe pas).

```
* Le canal canal_x n'est pas vide.
```

Listing 15 – Exemple de message de suppression impossible d’un canal (le canal n’est pas vide).

```
* Impossible de supprimer le canal par default du chat (place publique).
```

Listing 16 – Message de suppression impossible du canal « place publique ».

```
* client_x quitte le chat.
```

Listing 17 – Exemple de message de départ d’un client du chat.