

Projet de ICQ NEXT

ISY Messagerie (I Seek You)

Yann Catanese

4 novembre 2025

1 Présentation de la messagerie instantanée en ligne de commande : ISY

1.1 Analyse fonctionnelle

1.1.1 Schéma général

On souhaite réaliser une messagerie instantanée en ligne de commande qui permettent les fonctions décrites dans le schéma UML suivant :

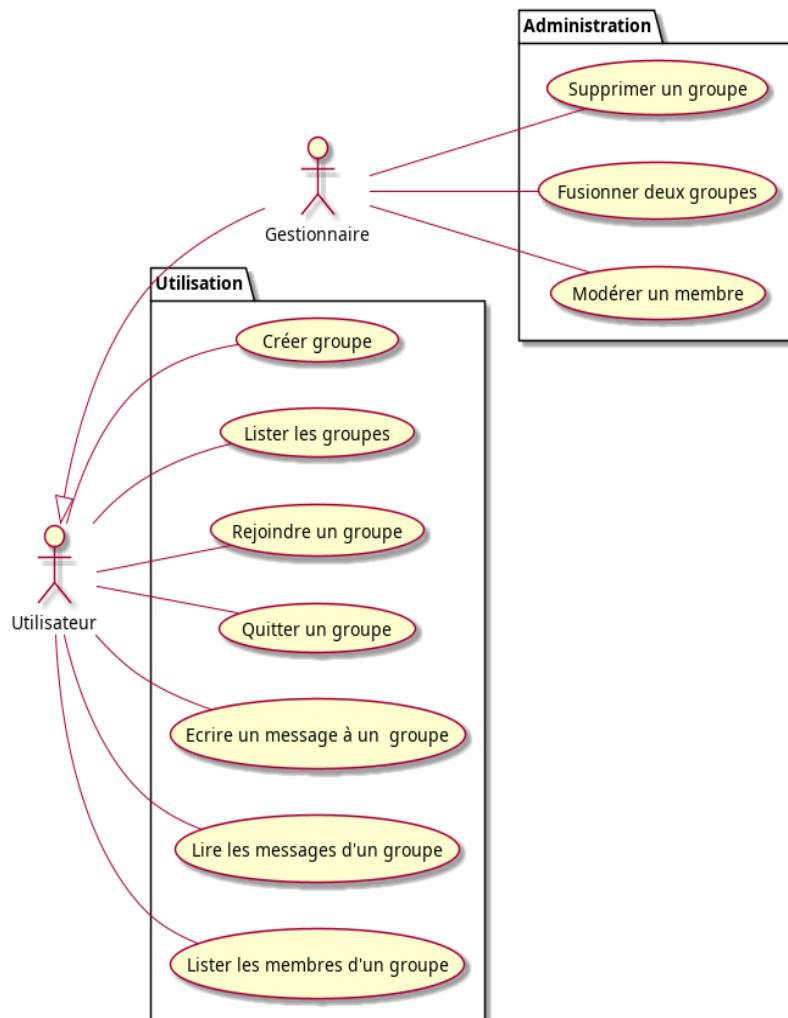


FIGURE 1 – Cas d'utilisation : Général

Détails des fonctions :

- **Créer groupe** : Permet à un utilisateur de créer un groupe de discussion, il en devient ainsi le gestionnaire ;
- **Supprimer un groupe** : Permet à un gestionnaire de supprimer un groupe de discussion ;
- **Fusionner deux groupes** : Permet à un gestionnaire de deux groupes de les fusionner en un seul ;
- **Modérer un membre** : Permet à un gestionnaire de supprimer un membre d'un groupe et d'empêcher qu'il ne rejoigne le groupe de nouveau ;
- **Lister les groupes** : Permet à un utilisateur de lister les groupes de discussion en cours sur le serveur ;
- **Rejoindre un groupe** : Permet à un utilisateur de rejoindre un groupe de discussion ;
- **Quitter un groupe** : Permet à un utilisateur de quitter un groupe de discussion ;
- **Ecrire un message à un groupe** : Permet à un utilisateur faisant partie d'un groupe de discussion, de publier un message ;
- **Lire les messages d'un groupe** : Permet aux utilisateurs d'un même groupe de discussion de visualiser tous les messages postés sur le groupe ;
- **Lister les membres d'un groupe** : Permet à un utilisateur de lister les utilisateurs actuellement connectés au groupe.

OBJECTIF 1 : Il vous est demandé de détailler tous ces cas d'utilisation par des schémas dédiés comme l'exemple du paragraphe suivant et la figure 2

1.1.2 Cas d'utilisation : Créer un groupe

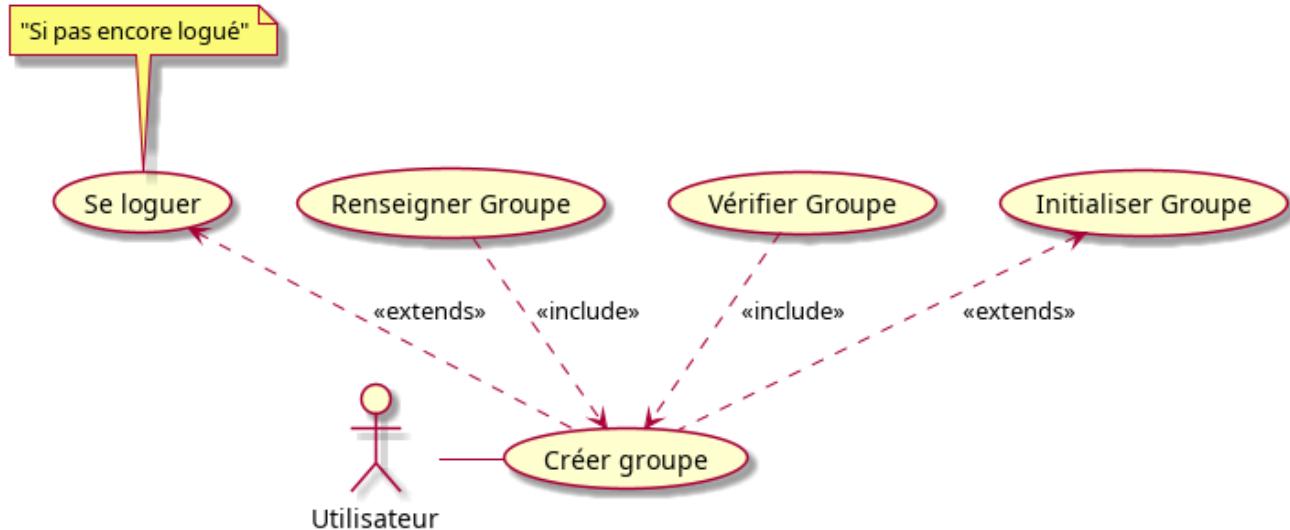


FIGURE 2 – Cas d'utilisation : Créer un groupe

Détails des fonctions :

- **Se loguer** : Si l'utilisateur n'est pas encore identifié (login, mot de passe), il lui est demandé de s'identifier ;
- **Renseigner Groupe** : L'utilisateur fournit les informations pour la création du groupe (nom, nombre maximum d'utilisateur, password, nombre de groupes maximum etc.) ;
- **Vérifier Groupe** : Le serveur vérifie si ce nom de groupe n'existe pas déjà ;

- **Initialiser Groupe** : Si toutes les informations sont correctes, le nouveau groupe est initialisé.

Chaque groupe devra proposer une idée novatrice et l'implémenter dans sa solution Côté serveur les programmes échangent et se synchronisent via SHM et sémaphores

1.2 Architecture logicielle

1.2.1 Découpage et rôle des fichiers source

L' objectif est de développer en C une messagerie instantanée via trois programmes C qui dialoguent entre eux en réseau :

- Un premier programme côté serveur qui attend les connexions clientes pour la création de groupes de messagerie instantanée et qui les gèrent.
- Un deuxième programme côté serveur qui est executé par le premier programme à chaque groupe de messagerie et qui gère les échanges de messages.
- Un troisième programme côté client pour se connecter échanger des messages, créer des groupes de discussions etc. Il se décompose en 2 teminaux bash, un pour l'affichage et un pour la saisie des commandes qui dialoguent via SHM.
- Un Quatrième programme côté client pour se connecter échanger des messages, créer des groupes de discussions etc. Il se décompose en 2 teminaux bash, un pour l'affichage et un pour la saisie des commandes qui dialoguent via SHM.

Pour les besoins du projet, on échangera ces messages via plusieurs machines ou sur la même machine avec des docker ou des VMs différentes, et le tout en mode ligne de commande.

On nommera respectivement :

- **ServeurISY.c** : le code source du serveur qui gère les groupes ;
- **GroupeISY.c** : le code source qui gère les discussions sur un groupe ;
- **ClientISY.c** : le code source côté client qui gère les commandes envoyées au serveur et les messages envoyés aux groupes dont on est membre ;
- **AffichageISY.c** : le code source côté client qui gère l'affichage.

Ces quatre fichiers feront référence au fichier à implémenter **Commun.h** qui comportera les définitions communes. Dans ce dernier par exemple est défini une structure qui servira de définition des messages échangés.

Un **makefile** est demandé pour compiler l'ensemble du projet

1.3 Messages

Les messages échangés possèdent la structure suivante :

- 4 octets pour l'ordre
- 20 octets pour le nom de l'émetteur
- 100 octets pour les textes des messages.

La structure suivante est imaginée pour ces messages (faites la évoluer au besoin) :

```
struct struct_message
{
    char     Ordre[4];
    char     Emetteur[20];
    char     Texte[100];
};
```

Dans notre cas, on imagine les ordres non exhaustifs suivants :

- CON : Pour une demande de connexion ;

- DEC : Pour une demande de déconnexion ;
- MES : Pour un message ;
- CMD : Pour une commande ;
- etc.

1.4 Qualité logicielle

Il est demandé également de respecter les règles de qualité logicielle données en cours, à savoir :

- Indentation de votre code ;
- Un certain nombre de commentaires pour l'aide à la compréhension ;
- Des noms de variables intelligibles ;
- Utilisation des tests des retours des fonctions systèmes ;
- etc.

Il est demandé de traiter tous les cas d'erreurs pour que votre code soit robuste. Par exemple les erreurs concernant l'utilisation des fonctions systèmes comme :

- Une erreur sur le gestionnaire de signal ;
- Une erreur à la création d'une socket ;
- Une Saisie de message vide ;
- Etc.

1.4.1 Schéma d'architecture logicielle

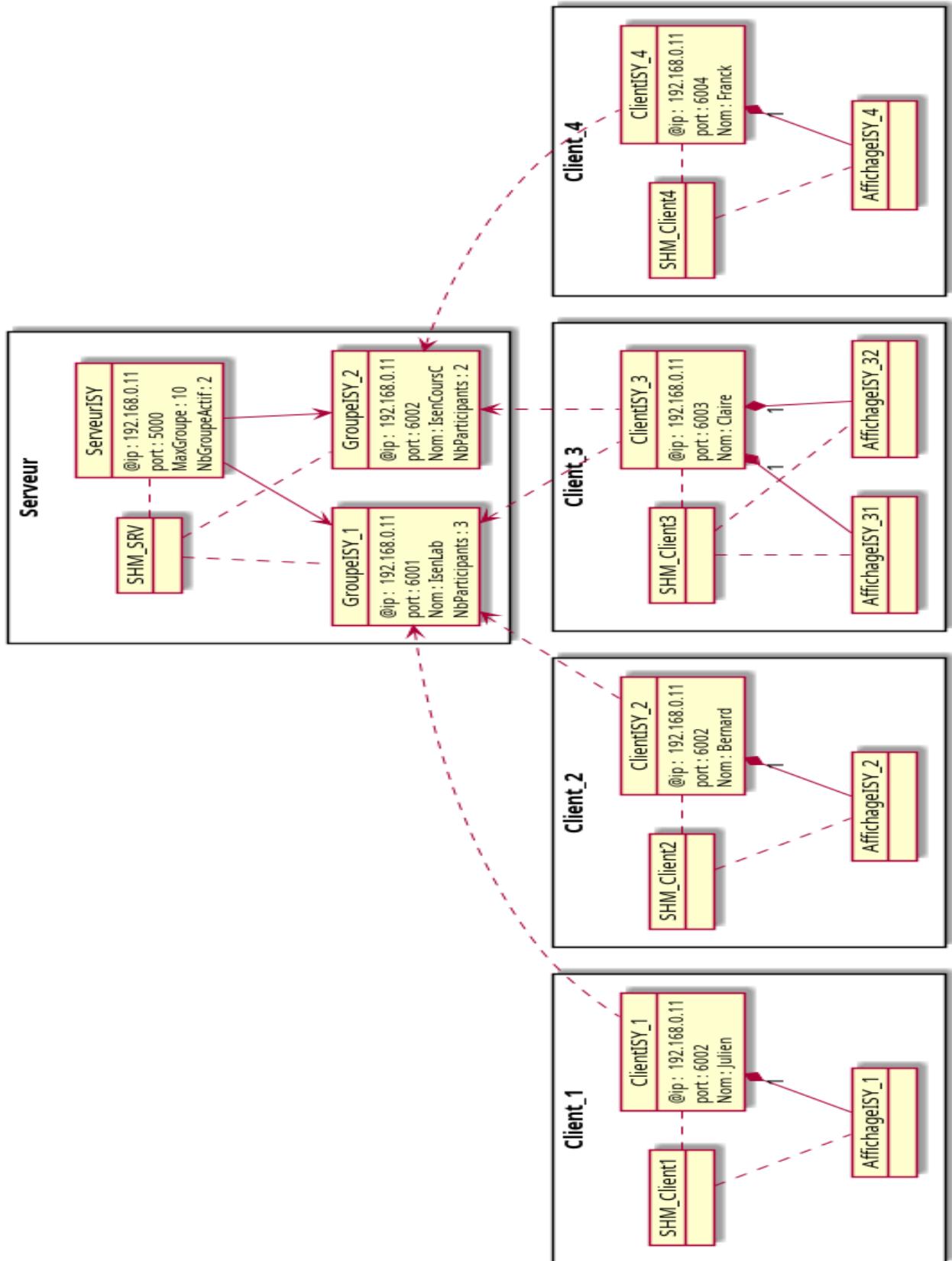


FIGURE 3 – Processus en mémoire

OBJECTIF 2 : Il vous est demandé de respecter cette architecture logicielle de la figure : 3

1.5 Le processus ServeurISY

Ce programme initialise un socket réseau sur lequel il attend les messages suivants :

- Demandes de création de groupe ;
- Demandes de destruction de groupe ;
- Demandes de connexion ;
- Demandes de déconnexion ;
- Demande de listes de groupe de discussion ;
- Etc.

Ce programme ouvre un socket réseau avec les informations suivantes :

- protocole UDP/UNICAST
- Adresse IP de la machine serveur
- Port Reception 8000

Un fichier de paramétrage est lu à l'initialisation du processus pour lire ces valeurs

Ce programme peut se terminer sur l'appui de CTRL-C, il met alors fin à tous les groupes de discussions, attend leur terminaison et se termine également proprement.

A chaque demande de création d'un nouveau groupe de discussion, il mémorise le nom du créateur (seul habilité à gérer les membres du groupe). De manière générale, il contrôle les actions sur les groupes, par exemple la redondance des noms de groupes de discussion.

Ensuite il lance un processus **GroupeISY** pour gérer les messages de ce groupe. Les échanges de données entre le processus **ServeurISY** et mes processus **GroupeISY** se font par SHM.

BASH : ServeurISY

```
>./ServeurISY
Lecture du fichier de configuration OK
Franck : Reception d une demande de listing des groupes de discussion
Envoi Franck : liste des groupes de discussions
Franck : Reception d une demande de creation de groupe ISEN
Groupe ISEN cree
Franck : Reception d une demande de connexion au groupe de discussion ISEN
Envoi Franck : Informations de connexion au groupe
Ludovic : Reception d une demande de connexion au groupe de discussion ISEN
Envoi Ludovic : Informations de connexion au groupe
Bertrand : Reception d une demande de connexion au groupe de discussion
ISEN
Envoi Bertrand : Informations de connexion au groupe
Sophie : Reception d une demande de connexion au groupe de discussion ISEN
Envoi Sophie : Informations de connexion au groupe
Franck : Reception d une demande d exclusion de Ludovic
Ludovic exclu du groupe ISEN
```

1.6 Le programme GroupeISY

Ce programme ouvre un socket réseau avec les informations suivantes :

- protocole UDP/UNICAST
- Adresse IP de la machine serveur
- Port Reception 80xx (avec XX comme numéro de groupe)

Ce programme gère les communications au sein d'un groupe de discussion. Il reçoit tous les messages des membres du groupe et les retransmet à l'ensemble du groupe. Il communique avec le

processus **ServeurISY** par le biais d'une mémoire partagée pour certaines actions comme une commande de la part du modérateur (créateur du groupe) d'interdire un membre ou autre informations sur le groupe (nombre de messages échangés, nombre de membres actifs, numéro de port du groupe etc.).

BASH : GroupeISY ISEN

```
Lancement du groupe de discussion ISEN, modérateur Franck
Reception Message Franck : Hello tout le monde
Redistribution message a Franck
Reception Message Franck : Hello tout le monde
Redistribution message a Franck, Sophie, Bertrand, Ludovic
Reception Message Sophie : Bienvenue dans l ISY de l ISEN Bertrand
Redistribution message a Franck, Sophie, Bertrand, Ludovic
Reception Message Bertrand : Merci Sophie
Redistribution message a Franck, Sophie, Bertrand, Ludovic
Reception Message Ludovic : Tu n es pas le bienvenu ici !
Redistribution message a Franck, Sophie, Bertrand, Ludovic
Reception Message Franck : Pas de critique sur le groupe
Redistribution message a Franck, Sophie, Bertrand, Ludovic
Reception Message Ludovic : Franck non plus n es pas le bienvenu ici !
Redistribution message a Franck, Sophie, Bertrand, Ludovic
Reception Message Sophie : On est mieux sans critique negative
Redistribution message a Franck, Sophie, Bertrand
Reception Message Franck : Oui en effet
Redistribution message a Franck, Sophie, Bertrand
```

1.7 Le programme AffichageISY

Une fois lancé le programme attend les messages transmis par le processus **GroupeISY** auquel il est connecté et affiche les messages reçus sur son terminal dédié.

L'affichage doit ressembler à cet exemple pour AffichageISY :

BASH : Groupe de discussion ISEN modérateur franck

```
Message de Franck : Hello tout le monde
Message de Bertrand : Hello tout le monde
Message de Sophie : Bienvenue dans l ISY de l ISEN Bertrand
Message de Bertrand : Merci Sophie
Message de Ludovic : Tu n es pas le bienvenu ici !
Message de Franck : Pas de critique sur le groupe
Message de Ludovic : Franck non plus n es pas le bienvenu ici !
Action de Franck : Membre ludovic supprime du groupe ISEN
Message de Sophie : On est mieux sans critique negative
Message de Franck : Oui en effet
```

1.8 Le programme ClientISY

Un fichier de paramétrage est lu à l'initialisation du processus pour connaître les informations sur le socket serveur sur lequel communiquer ainsi que les informations du membre. Une fois lancé le processus est lié à un terminal qui attend les commandes de l'utilisateur (demande de création d'un groupe, demande de fourniture de la liste des groupes, demande pour rejoindre un groupe etc.). Une fois un groupe rejoint (en tant qu'utilisateur ou modérateur), il lance un processus **AffichageISY** pour l'affichage des messages du groupe. Un processus **AffichageISY** est lancé pour chaque groupe que l'utilisateur a rejoint, en revanche il n'y a qu'un seul terminal ClientISY pour les commandes.

Ce programme se termine sur l'appui de CTRL-C, il met alors fin à tous les affichages des différents groupes de discussions ouverts, attends leurs terminaisons et se termine également proprement en faisant une demande de déconnexion au serveur.

Vous pouvez lancer le programme **ClientISY** plusieurs fois avec un fichier de configuration différent sur la même machine.

OBJECTIF 3 : Il vous est demandé de développer l'ensemble de ces programmes

BASH : ClientISY

```
>./ClientISY
Lecture du fichier de configuration OK, utilisateur franck
Choix des commandes :
 0 Creation de groupe
 1 Rejoindre un groupe
 2 Lister les groupes
 3 Dialoguer sur un groupe
 4 Quitter
Choix :
>2
Aucun groupe
Choix des commandes :
 0 Creation de groupe
 1 Rejoindre un groupe
 2 Lister les groupes
 3 Dialoguer sur un groupe
 4 Quitter
Choix :
0
Saisire le nom du groupe
ISEN
Envoi de la demande au serveur
Groupe cree !
Choix des commandes :
 0 Creation de groupe
 1 Rejoindre un groupe
 2 Lister les groupes
 3 Dialoguer sur un groupe
 4 Quitter
Choix :
1
Saisire le nom du groupe
ISEN
Connexion au groupe ISEN realisee, lancement de l affichage
Tapez quit pour revenir au menu, cmd pour entrer une commande, msg pour
    revenir aux messages
Message : Bonjour a tous
Message : Pas de critique sur le groupe
Message : cmd
Commande : list
Bertrand
Sophie
Ludovic
Commande : Delete ludovic
Commande : msg
Message : Oui en effet
Message : quit
Choix des commandes :
 0 Creation de groupe
 1 Rejoindre un groupe
 2 Lister les groupes
 3 Dialoguer sur un groupe
 4 Quitter
Choix :
4
Demande de deconnexion du groupe
Attente de fermeture de l affichage
Affichage clos
Fin du programme
$>
```