

Jules LEHODEY  
Leo LAUGIER  
Thomas LYLE  
groupe TD A2

# Dossier de conception

## Protocole de communication :

Nous avons choisi d'utiliser le protocole TCP car, malgré la possibilité que le destinataire du message ne soit pas connecté à l'application au moment de l'envoie, il doit être connecté au serveur pour recevoir les messages. De plus avec le protocole TCP nous avons une garantie que le destinataire reçoive les messages dans l'ordre dans lequel ils ont été envoyés.

## Interfaces :

### Interface utilisateur :

A hand-drawn sketch of a login page. It features a large rectangular form with two input fields labeled "Username" and "Password". Below these fields are two buttons labeled "mdp oublié" and "envoyer".

Page de connexion

A hand-drawn sketch of a password recovery page. It has a large rectangular form with an input field labeled "mail" and a button labeled "envoyer".

Mot de passe oublié

A hand-drawn sketch of a user profile page. It shows a grid layout with a header row containing a search icon, a "+" sign, and a user icon. The main body contains a table with three rows labeled "Ticket 1", "Ticket 2", and "Ticket 3". To the right of the table is a user icon and a list of profile information: "nom", "prenom", "adresse", and "groupes".

Page d'accueil

A hand-drawn sketch of a ticket creation page. It has a large rectangular form with an input field labeled "Titre ticket", an input field labeled "Message", and a button labeled "envoyer".

Création d'un ticket

A hand-drawn sketch of a discussion page. It shows a grid layout with a header row containing a search icon, a "+" sign, and a user icon. The main body contains a table with three rows labeled "Ticket 1", "Ticket 2", and "Ticket 3". To the right of the table is a speech bubble icon and a "refus" button.

Page de discussion

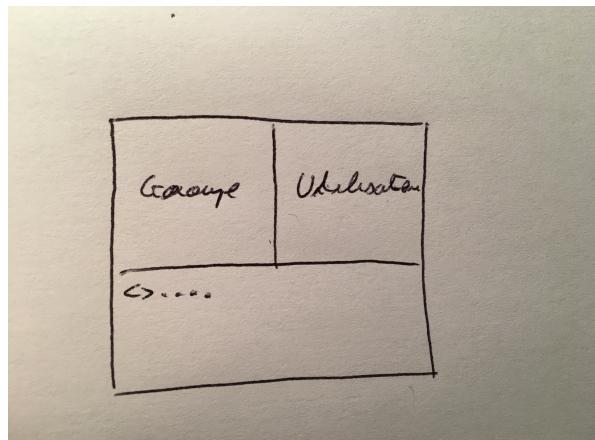
Pour l'interface du client, nous commençons avec une page classique de connexion où on rentre son nom d'utilisateur et son mot de passe. Nous aurons aussi la possibilité de pour pouvoir récupérer notre mot de passe en cas d'oublié.

Une fois connecté, on arrive sur la page d'accueil qui sert aussi aussi de page d'accueil. De la on peut consulter les informations du profil et ce déconnecter.

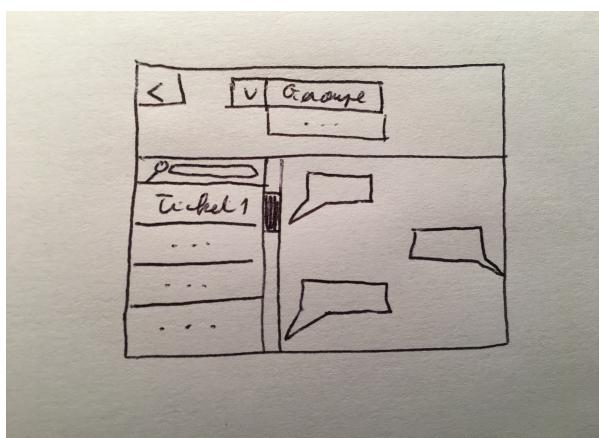
Pour la consultation des différents tickets, nous avons choisi un menu déroulant sur la gauche de la fenêtre, de ce menu, l'utilisateur pourra utiliser le bouton « rechercher » (Loupe) pour chercher un ticket précisément. Le bouton « rafraîchir » (Flèche) permettra d'actualiser la liste des tickets et de recevoir tout les nouveaux messages/tickets. Le bouton « Nouveau » (+) pour accéder la fenêtre de création des tickets.

Pour créer un ticket, il suffira de saisir le nom du ticket ainsi que le message et de l'envoyer.

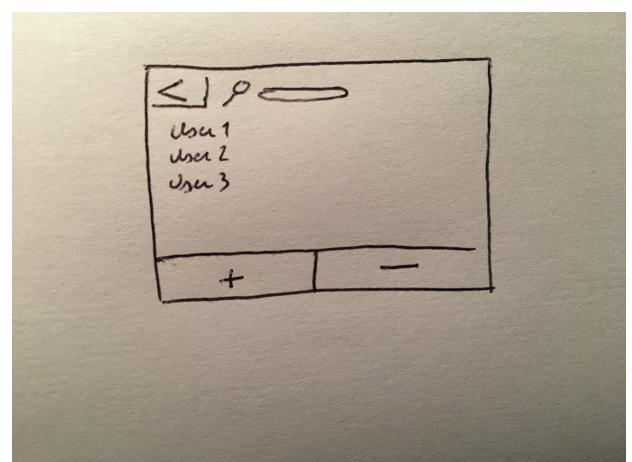
Interface serveur :



Page d'accueil serveur



Page de gestion des groupes



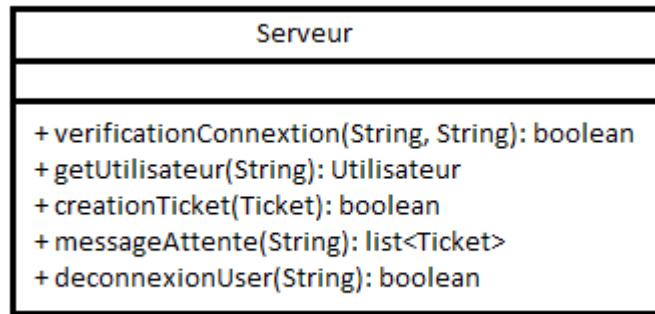
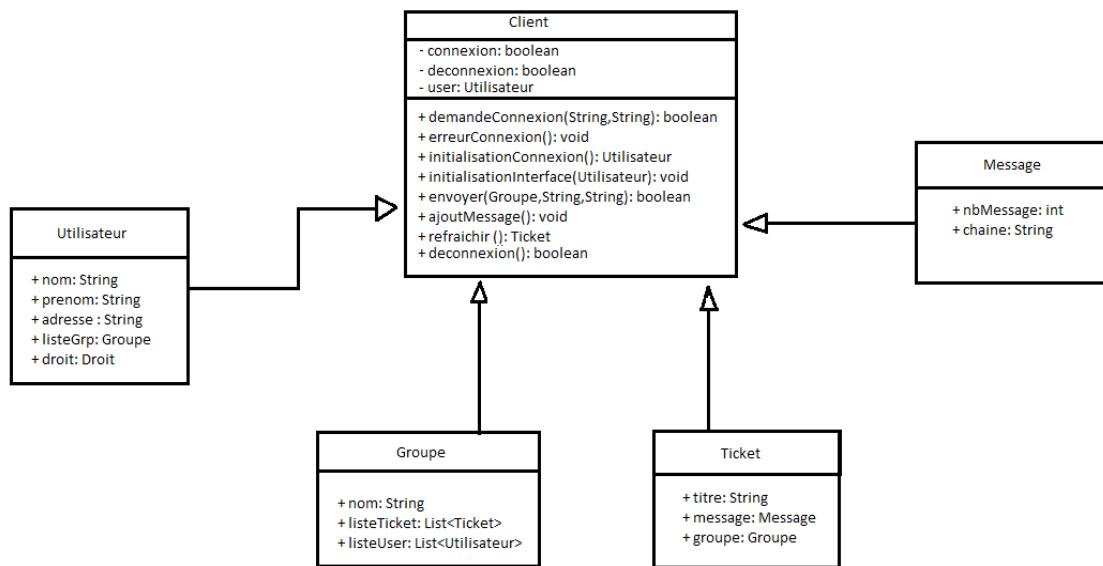
Page de gestion des utilisateurs

Pour l'interface du serveur, nous avons choisi de diviser la fenêtre d'accueil en 3 parties, la première étant un bouton pour accéder à la page de gestion des groupes, de cette page l'administrateur pourra consulter tout les tickets destinés à un groupe. Il pourra choisir le groupe à l'aide d'un menu déroulant.

Le second bouton « Utilisateur » permettra d'accéder la page de gestion des utilisateurs, de cette page, l'administrateur pour voir la liste des tout les utilisateurs, ainsi qu'un bouton (+) pour pouvoir ajouter un nouvel utilisateur, ou ajouter un utilisateur à un groupe, et un bouton (-) pour pouvoir supprimer un utilisateur, ou le supprimer d'un groupe.

La dernière partie est une console qui permettra de voir tout les trafiques entrant et sortant du serveur.

Diagramme de classe :

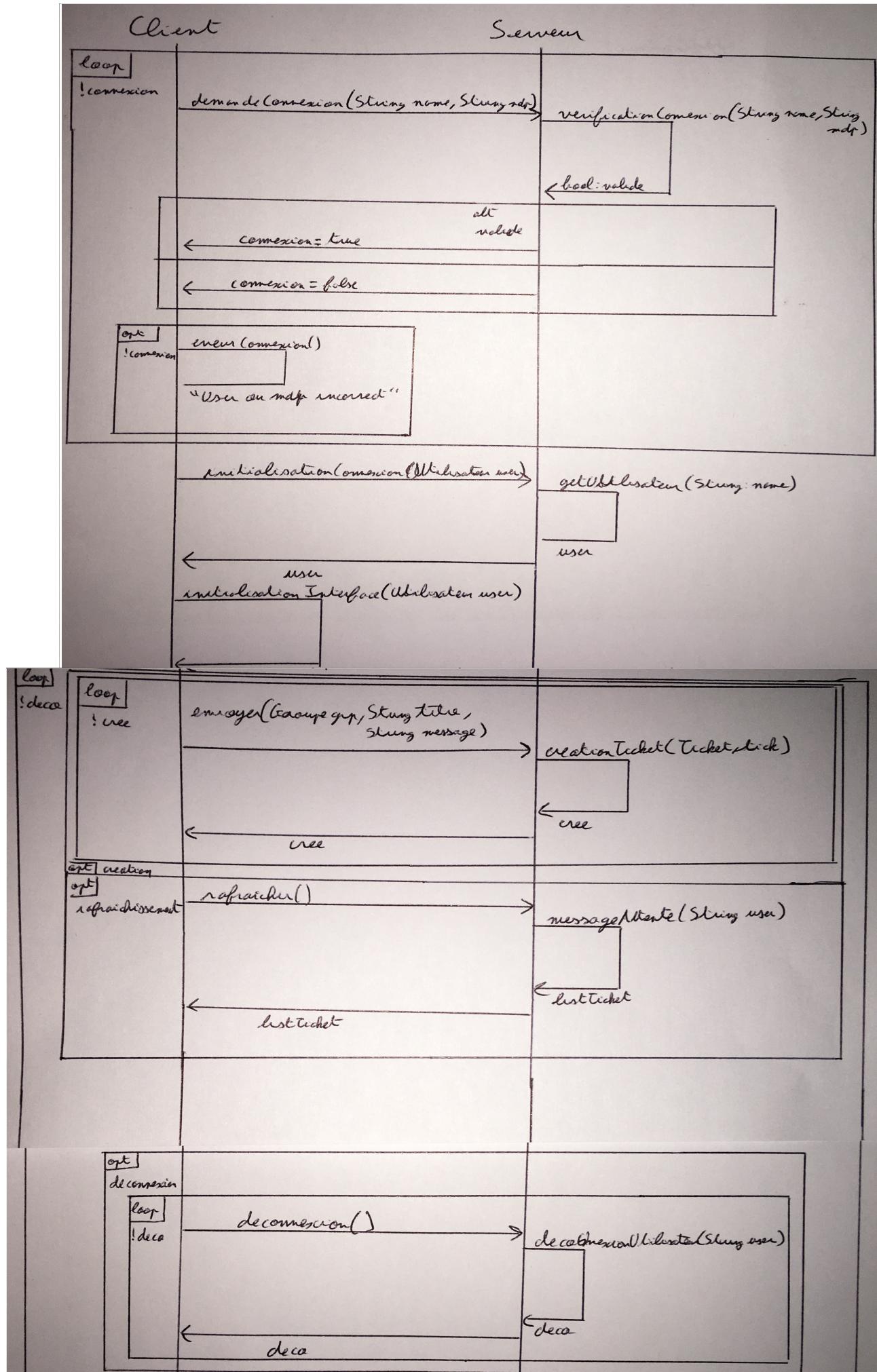


Nous avons décidé de faire 2 diagrammes de classe, un premier pour la partie client, et le second pour le serveur.

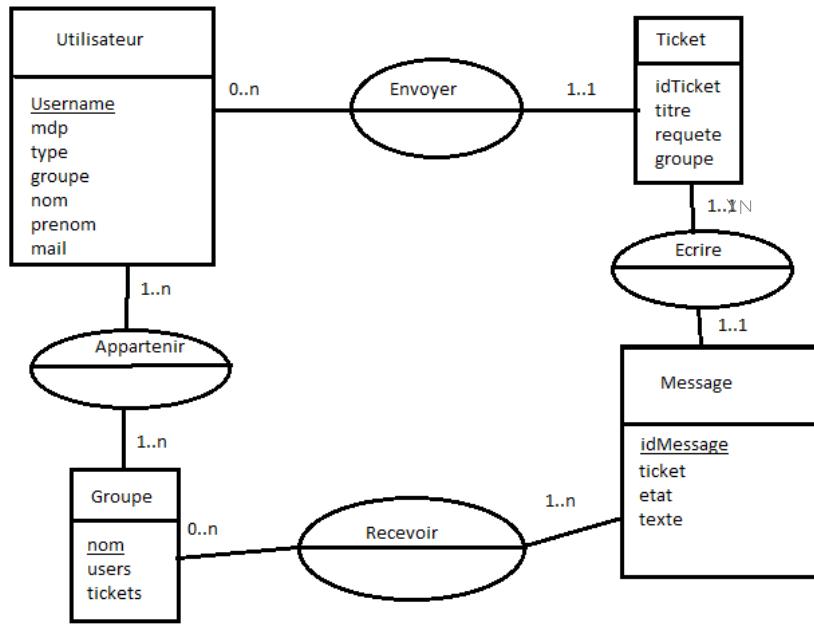
Pour le client, nous avons divisé la programme en 5 classes. Les classes Utilisateur, Groupe, Ticket et Message serviront à récupérer et stocker les informations envoyées par le serveur. La classe Client permettra de communiquer avec le serveur.

La partie serveur communiquera avec la base de données et pourra envoyer ces informations au client.

Diagramme de séquence :



Base de données :



Nous avons choisi de faire une base de donnée avec 4 classes et 4 associations.

La classe utilisateur regroupera toutes les informations sur les utilisateurs de l'application. Les utilisateurs seront reconnaissable grâce à leur « Username », un utilisateur pourra appartenir à 1 ou plusieurs groupe et aura envoyer ou non un ticket.

Les groupes seront identifiés par leur nom, et auront accès à tout les tickets visibles. Les groupes pourront recevoir des messages.

Les messages seront écrits à partir d'un ticket.