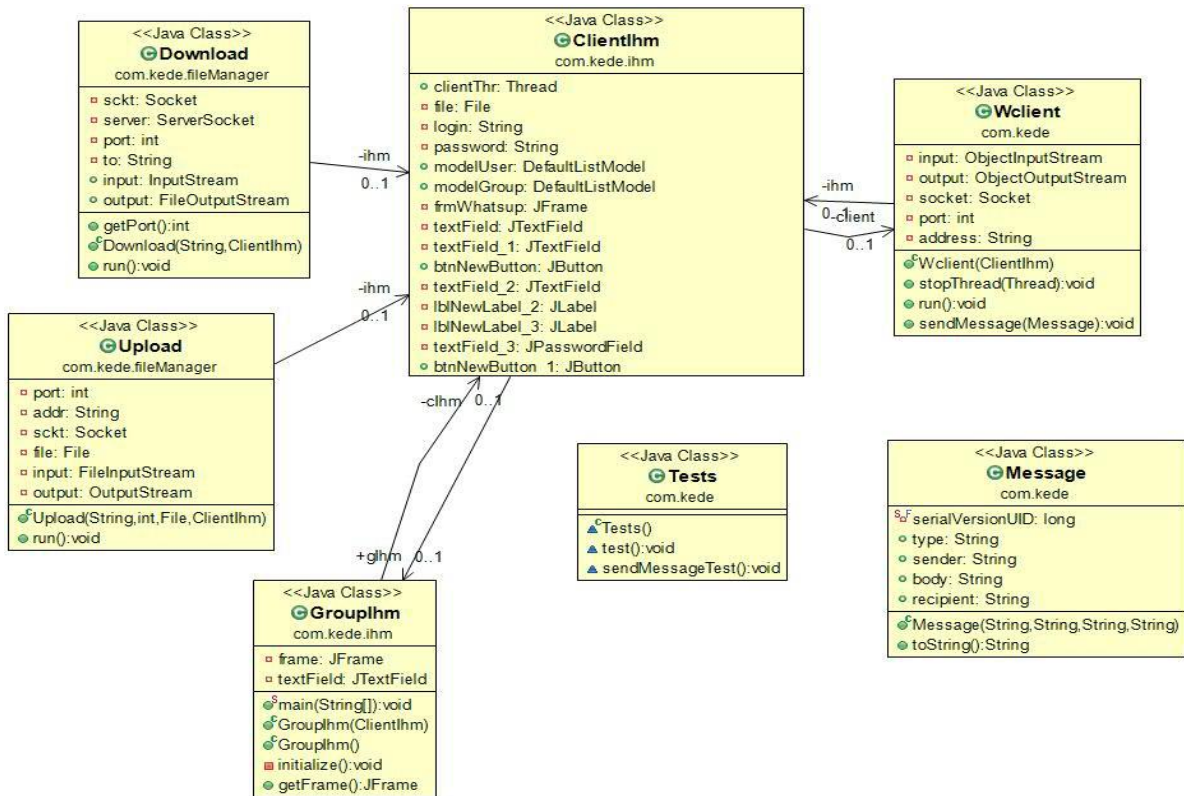
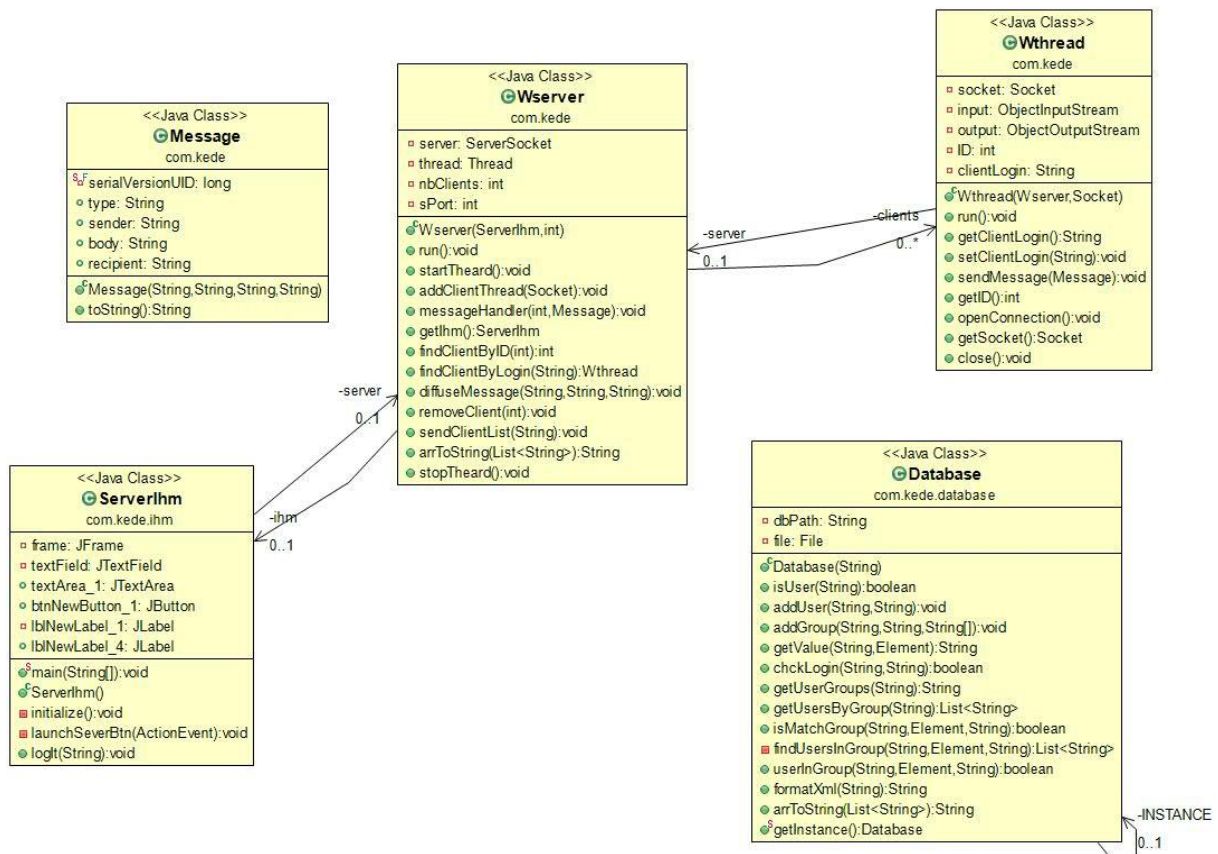


FICHE TECHNIQUE

CLIENT UML



SERVER UML



1. ARCHITECTURE

L'architecture de whatsapp a été un choix décisif dans la conception de celui-ci, au vu de problématique et l'objectif de l'application, nous avons décidé de nous tourner vers une architecture en deux grandes couches

- Une Couche CLIENT/SERVEUR
- Une Pseudo-MVC

Ce choix nous a permis de séparer l'application en deux projets indépendants, Un projet CLIENT et un projet Serveur, et chaque'un de ses projets est basé sur une architecture Pseudo-MVC.

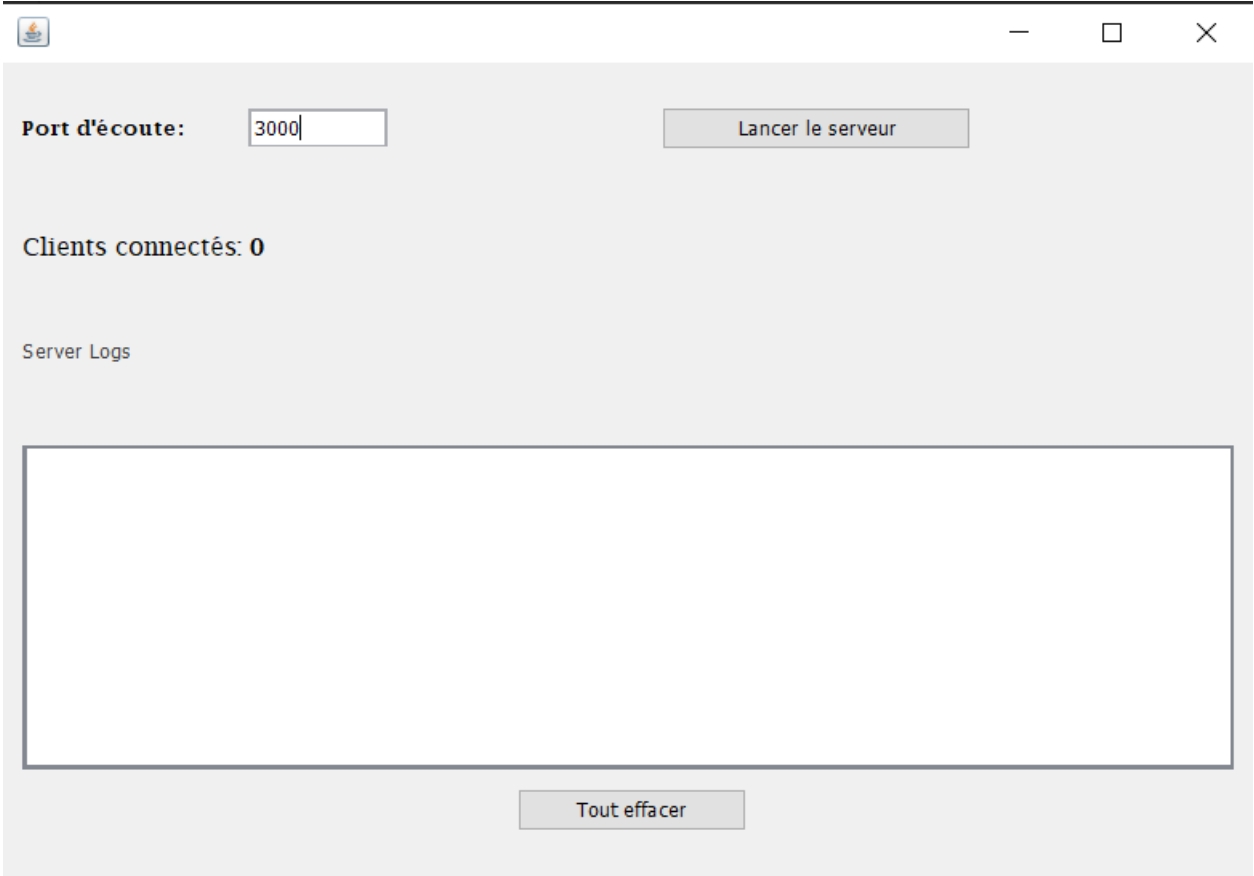
L'interaction entre ces deux parties se fait via des appels réseaux.

Ce choix d'architecture était selon nous l'une des meilleurs facon d'implémenter whatsapp, on pouvait facilement se répartir les taches, la maintenance applicative était beaucoup plus simples.

L'un des points fort de ce choix est le fait que l'on pouvait partager seulement la partie CLIENT de notre application aux utilisateurs.

2. SERVEUR

Le serveur est le coeur de notre application, c'est celui-ci qui va gérer les connexions des différents utilisateurs ainsi que les différents échanges de données (Messages extuels et fichiers)



The screenshot shows a window titled "SERVEUR" with a standard Windows-style title bar (minimize, maximize, close buttons). The main content area is light gray and contains the following elements:

- Port d'écoute:** A label followed by a text input field containing the value "3000".
- Lancer le serveur:** A button located to the right of the port input field.
- Clients connectés: 0**: A text label indicating the current number of connected clients.
- Server Logs:** A label above a large, empty rectangular box intended for displaying log messages.
- Tout effacer:** A button located at the bottom center of the window.

2.1 Lancer le serveur

Pour lancer le serveur, on renseigne un port d'écoute. Ici, nous avons pré-enregistrer le port 3000. Puis, on lance le serveur en cliquant sur "lancer le serveur".

Port d'écoute:

Port d'écoute:

Serveur lancer. IP : DESKTOP-209DU6K/192.168.1.100, Port : 3000

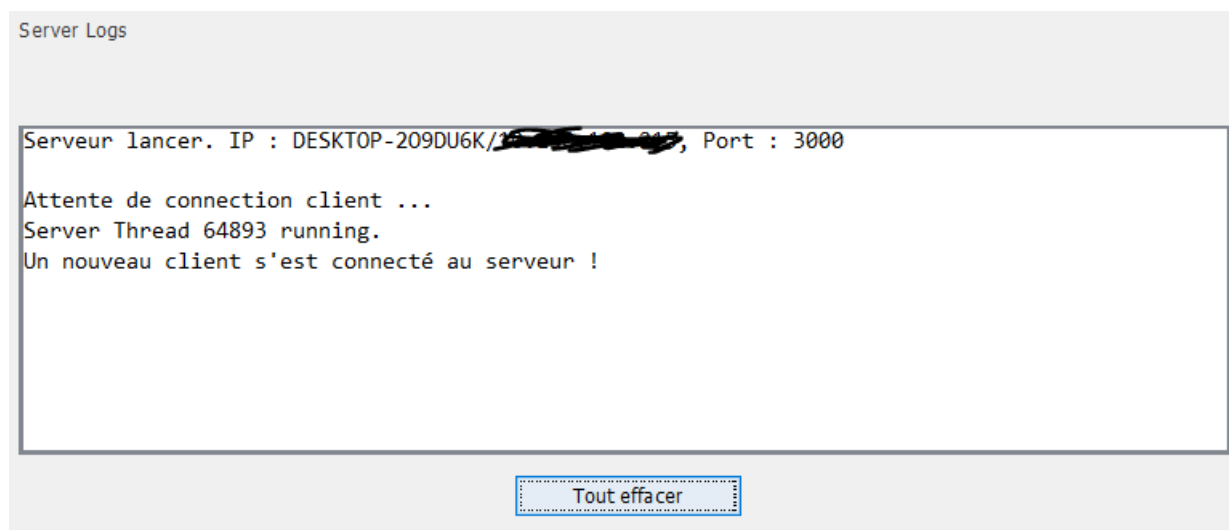
On peut voir ci-dessus la confirmation du lancement du serveur.

2.2 Le nombre de clients connectés

Clients connectés: 1

Cette partie nous montre le nombre de client connecté au serveur.

2.3 Les logs



Dans cette partie, on renseigne toutes les actions réalisées sur le serveur. On peut voir ici par exemple que le serveur est en état de marche et qu'un nouveau client s'est connecté.

3. CLIENT

Le client doit se connecter au serveur. Pour se faire, il doit renseigner l'adresse IP et le port du serveur puis il doit cliquer sur "Joindre"

Pour une première connexion le client doit définir son login et mot de passe ensuite cliquer sur "S'inscrire"

Ainsi pour ses futures connexions le client doit juste renseigner son login et mot de passe et cliquer sur "Se connecter"

Le client peut choisir le destinataire du message dans la partie "Membres"

Le bouton "Nouveau groupe" permet de créer un groupe et de le gérer c'est-à-dire ajouter des membres ou les supprimer. Quand tous les membres y sont le client clique sur "Créer le groupe"

Le bouton "Pièce jointe" permet au client d'envoyer des fichiers à différents destinataires.

Whatsup

Host: 127.0.0.1

Port: 3000

Joindre

Login: jj

Mot de passe:

Se connecter

S'inscrire

Membres

TOUT LE MONDE

[SERVER> MOI] : VOUS AVEZ REJOINT LE SERVEUR AVEC SUCCES, VEUILLEZ VOUS LOGGER
[SERVER> MOI] : LOGIN déjà pris!
[SERVER> MOI] : LOGIN en cours...
[SERVER> MOI] : CONNECTION REUSSITE, VOUS POUVEZ COMMENCER A CHATTER
[jj> TOUT LE MONDE] : hello

Nouveau groupe

hello

Envoyer

Piece-jointe

Nouveau groupe

Nom du groupe :

Ajouter des membres :

chris

Maintener la touche Ctrl
pour selectioner
plusieurs membres

Creer le groupe

4. BASE DE DONNEE

Dans la partie base de donnée est stockée les différentes données c'est-à-dire login, mot de passe, nom des groupes et composition des groupes

Database - Bloc-notes

Fichier Edition Format Affichage Aide

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><data>
```

```
  <user>
    <login>
      chris
    </login>
    <password>
      kede
    </password>
  </user>
  <user>
    <login>
      junior
    </login>
    <password>
      kede
    </password>
  </user>
  <group>
    <createdBy>
      chris
    </createdBy>
    <name>
      fkf
    </name>
    <userLogin>
      chris
    </userLogin>
    <userLogin>
      tot
    </userLogin>
    <userLogin>
      junior
    </userLogin>
  </group>
  <group>
    <createdBy>
      chris
    </createdBy>
    <name>
      test
    </name>
    <userLogin>
```