

Sommaire

1. Identification des acteurs

1.1. Primaires

1.2. Secondaires

2. Diagramme des cas d'utilisation

3. Diagramme des classes

3.1. Schéma

3.2. Détails et typage

4. Diagramme de séquence

4.1 Connexion

4.2 Déconnexion

4.3 Envoi Message

4.4 Changement de pseudo

5. Diagramme de composants

6. Diagramme de déploiement (Architecture visée)

7. Schéma de la BD
8. Maquettes des GUI
9. Diagramme des composants

1. Identification des acteurs

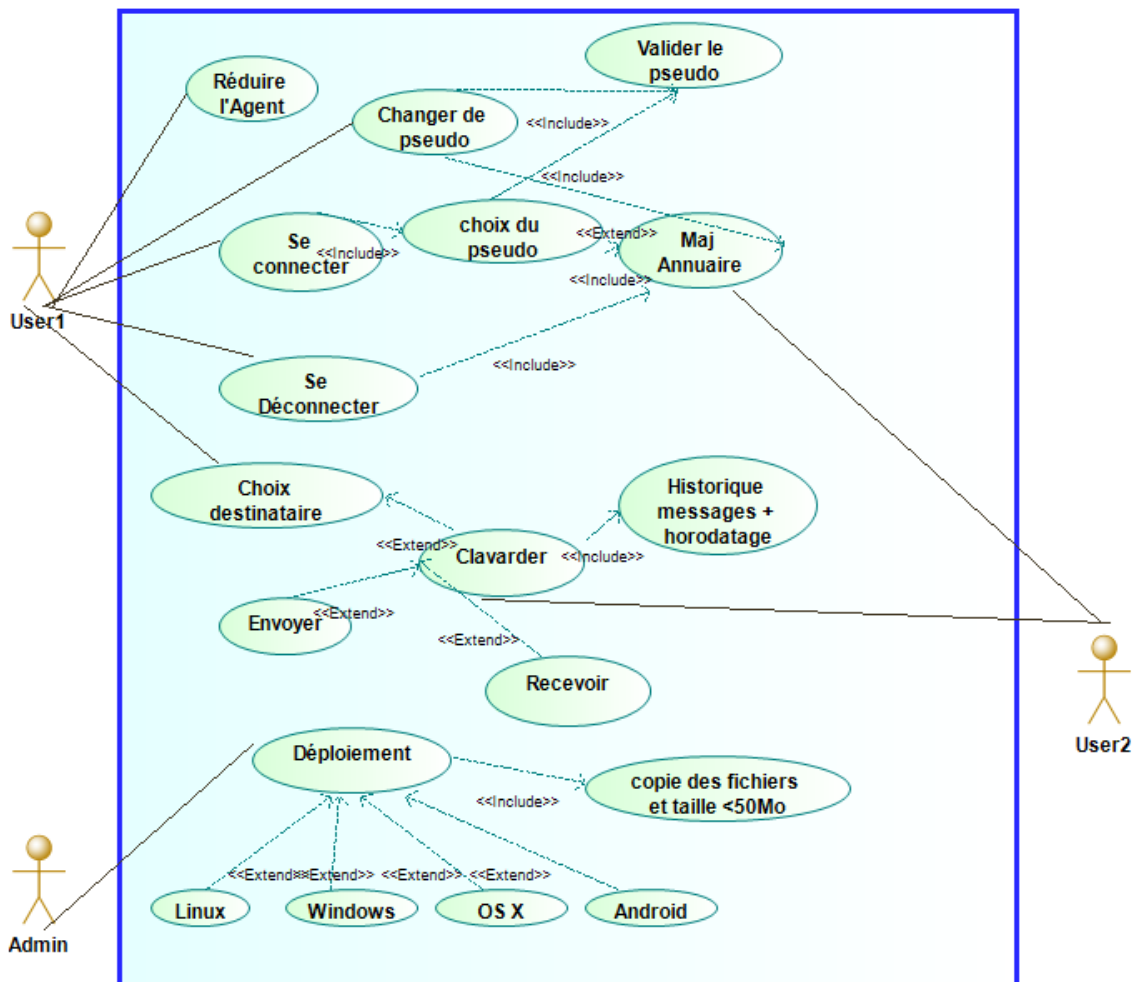
1.1. Primaires

Les acteurs principaux sont l'utilisateur du service de message et l'administrateur.

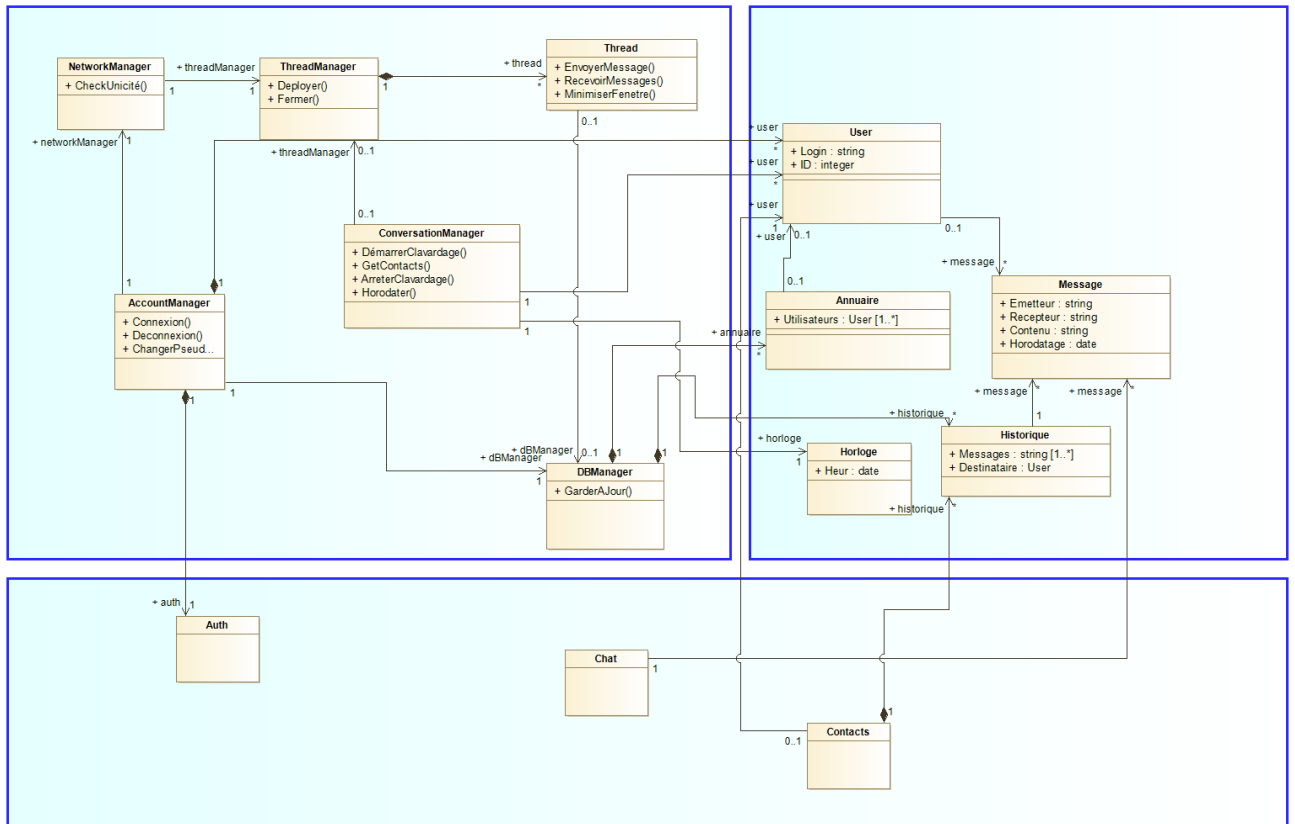
2.2 Secondaires

L'autre utilisateur avec lequel communique le premier utilisateur peut être considéré comme un acteur secondaire.

2. Diagramme des cas d'utilisation

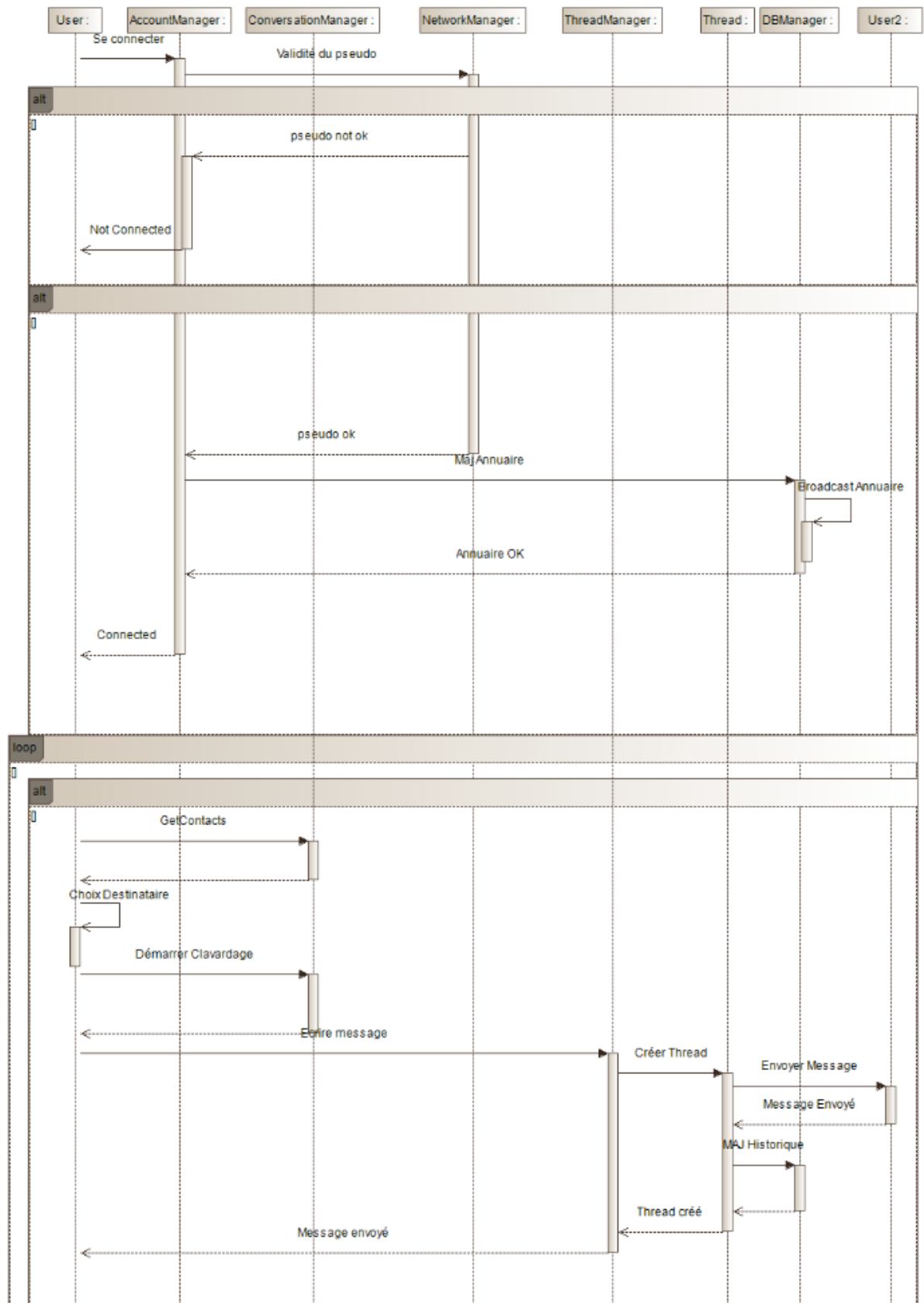


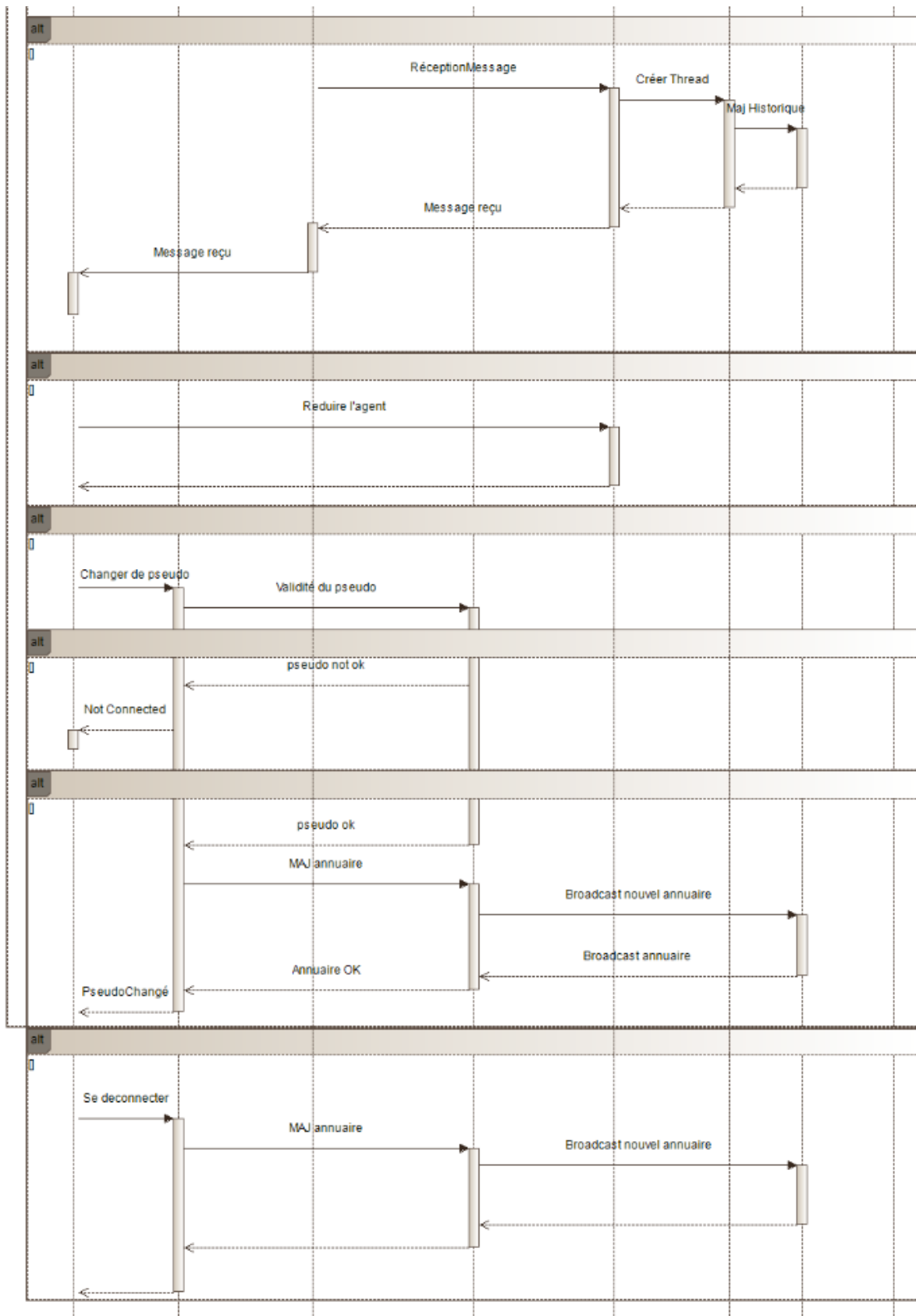
3. Diagramme de classes



On peut voir ci-dessus le diagramme de classe avec en haut à gauche les contrôleurs, en haut à droite les modèles et en bas la vue.

4. Diagramme de séquence



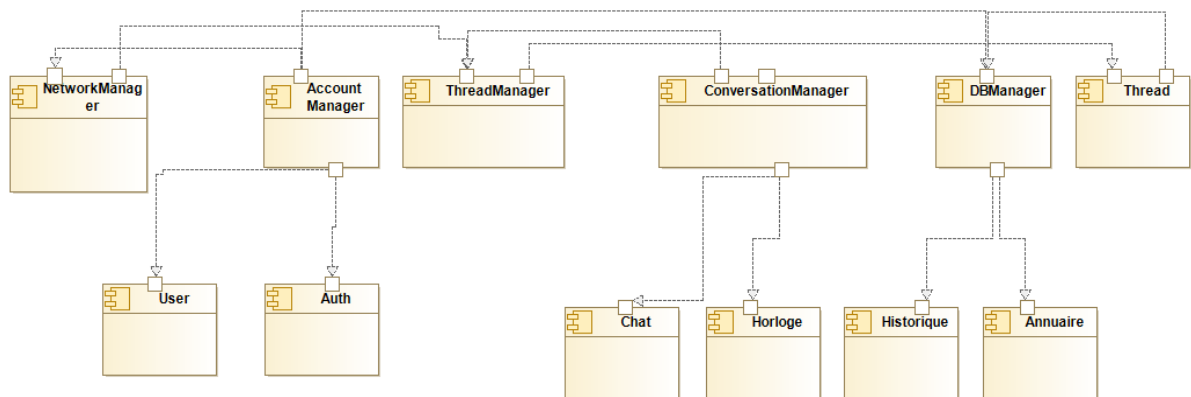


On a décidé d'expliquer les situations :

- Validation du pseudo, soit il est valide, soit il ne l'est pas. Lorsqu'il est valide, l'annuaire est mis à jour et on le broadcast.

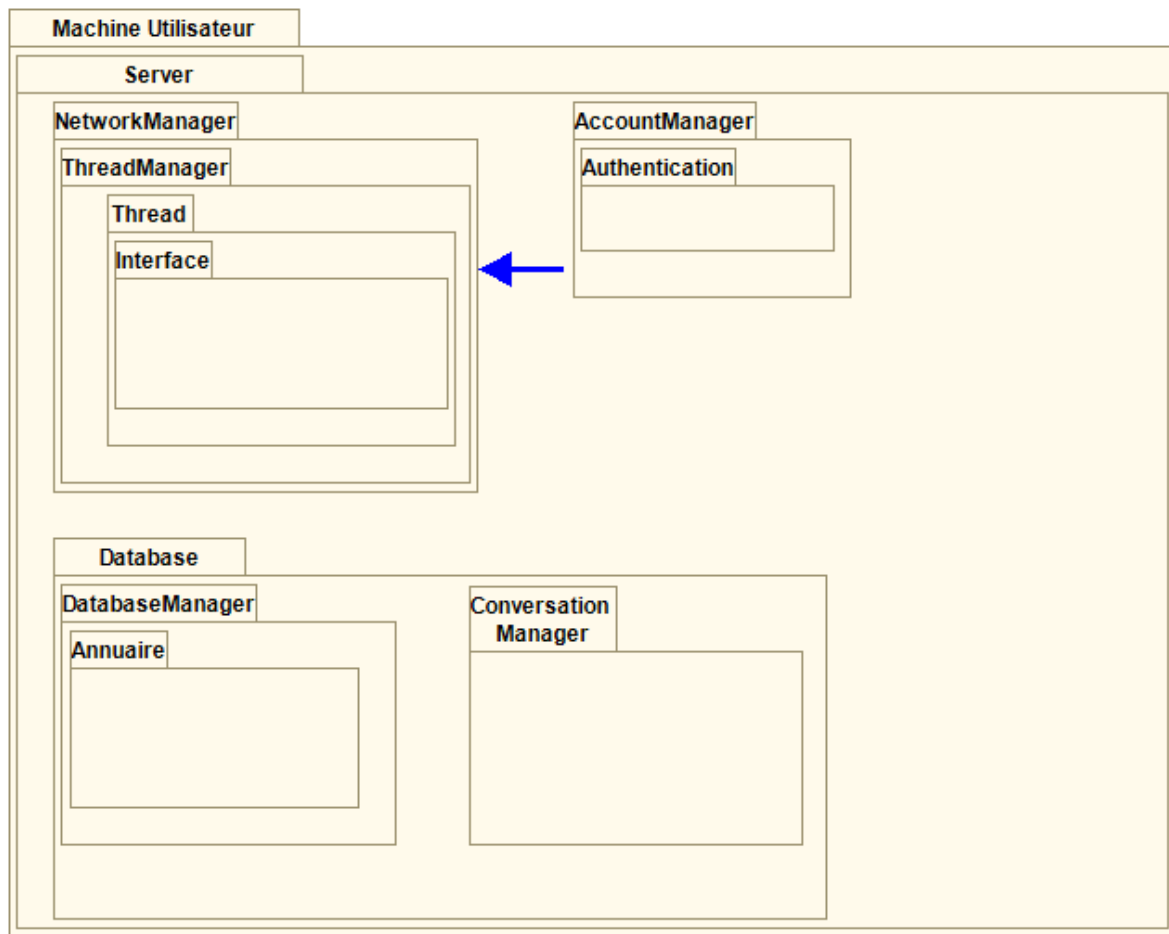
- L'envoi d'un message, pour cela, le manager des conversations nous renvoie les contacts, on crée une thread "envoi" et on rajoute le message à l'historique.
- La réception d'un message, pour cela, on crée une thread "réception" et on rajoute le message à l'historique.
- Lors d'une déconnexion, l'annuaire est mis à jour et on le broadcast.

5. Diagramme de composites



On peut voir l'interconnexion entre les différentes classes avec au-dessus les contrôleurs et en bas la vue et les modèles.

6. Diagramme de déploiement (Architecture visée)

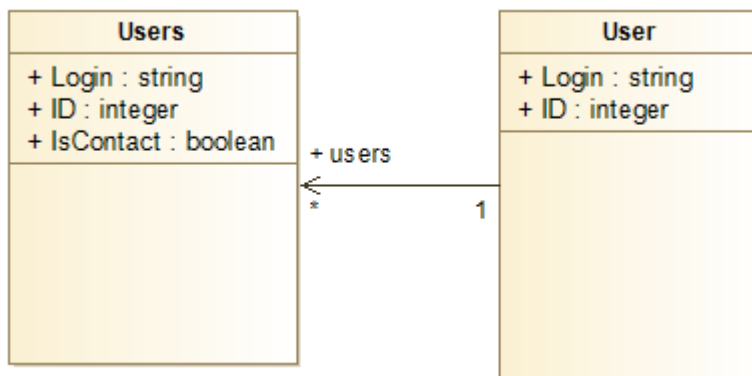


Le diagramme de déploiement indique l'imbrication de nos modèles et contrôleurs.

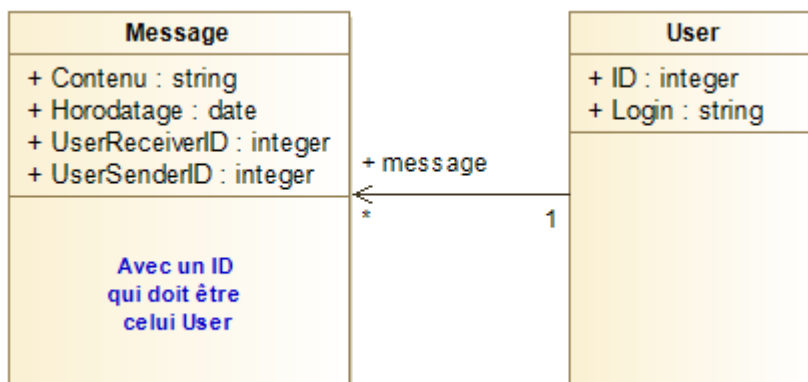
7. Schéma de la BD

La base de données sera un recueil de conversations. A son tour, chaque conversation est une liste de messages. Chaque message contient l'information nécessaire pour restituer la conversation avec des requêtes simples. Un message sera composé de qui émet le message, qui le reçoit, le contenu du message et l'horodatage. Pour la gestion des contacts, on aura aussi une conversation, qui aura un attribut isEmpty pour indiquer que la conversation est vide. Tout contact aura donc une conversation, vide ou pas.

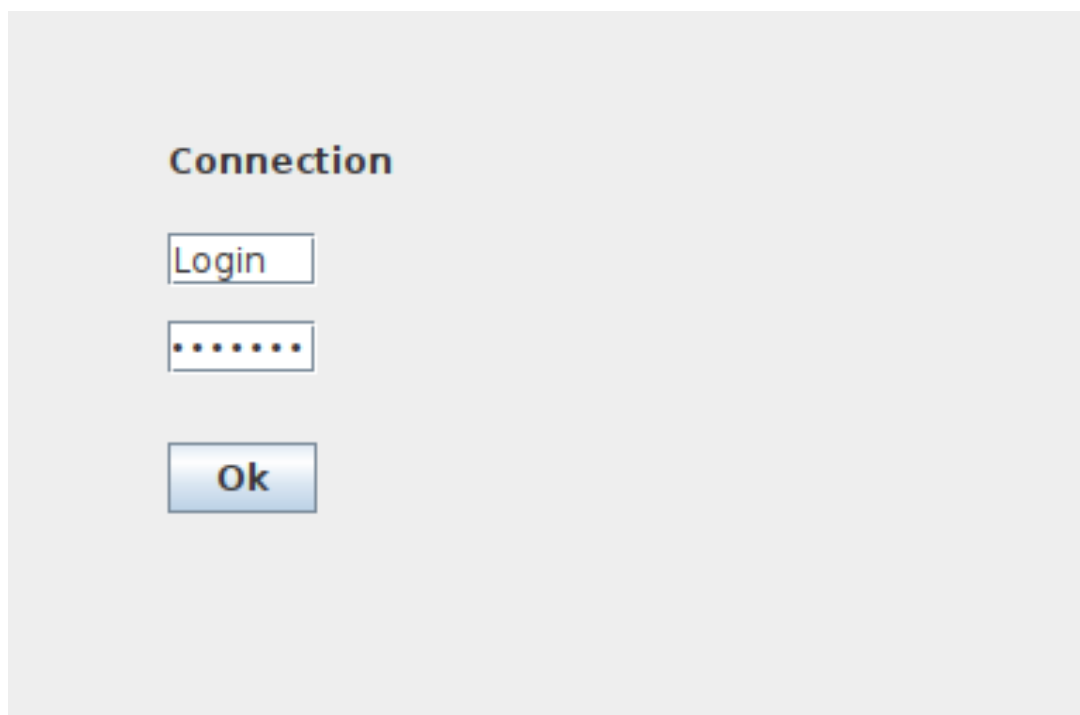
Annuaire:



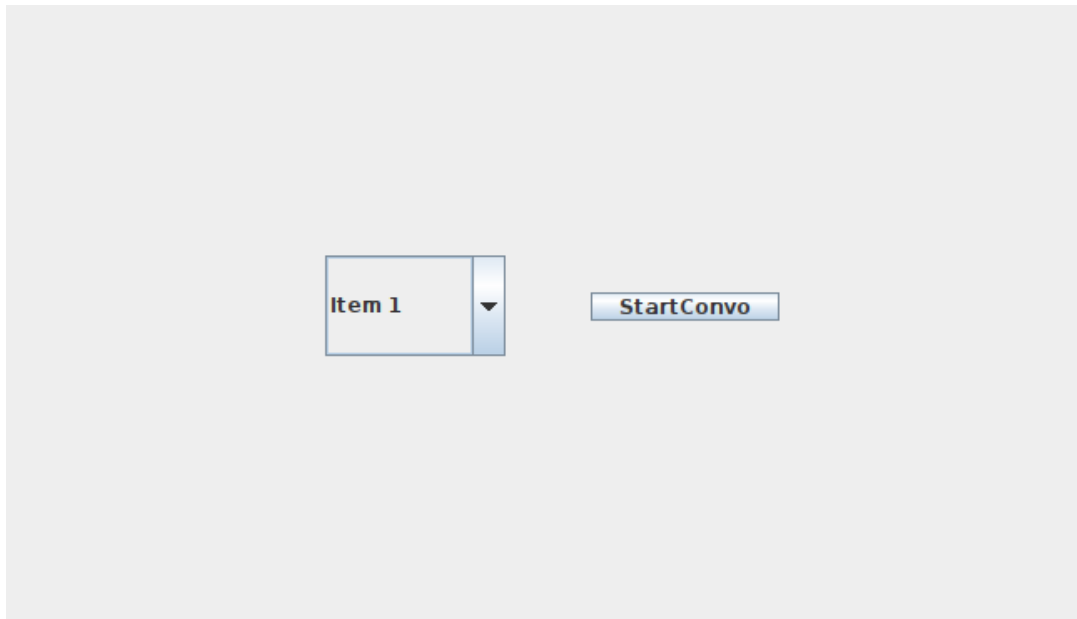
Historique:



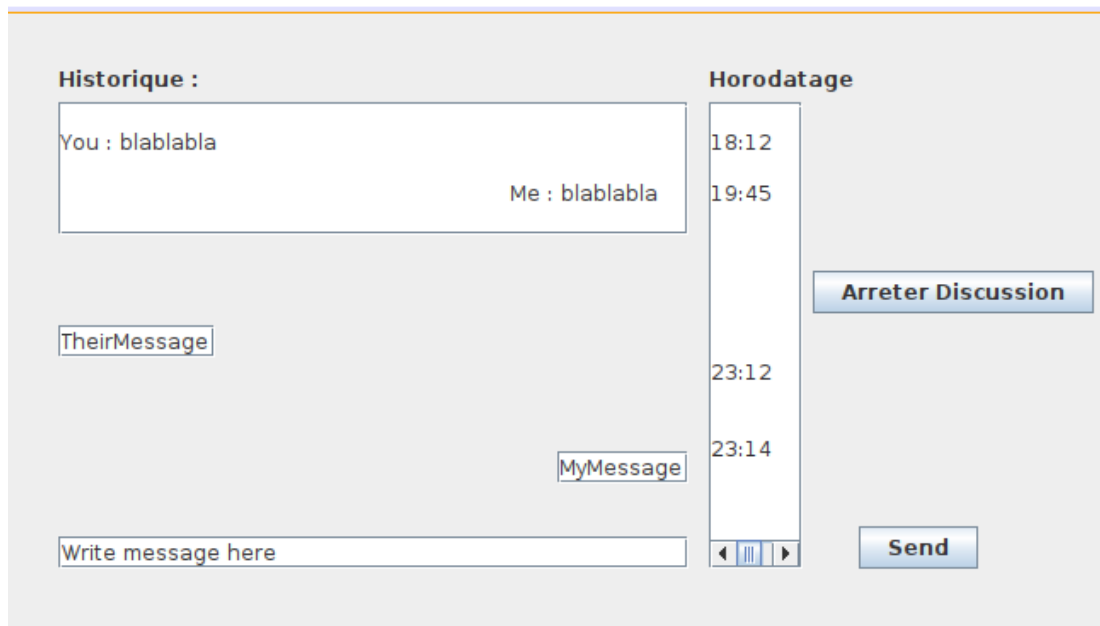
8. Maquette du GUI



Ecran de connexion initial



Annuaire



Interface d'envoi et de réception de message



Interface de changement de pseudo et de déconnexion

9. Diagramme des composants

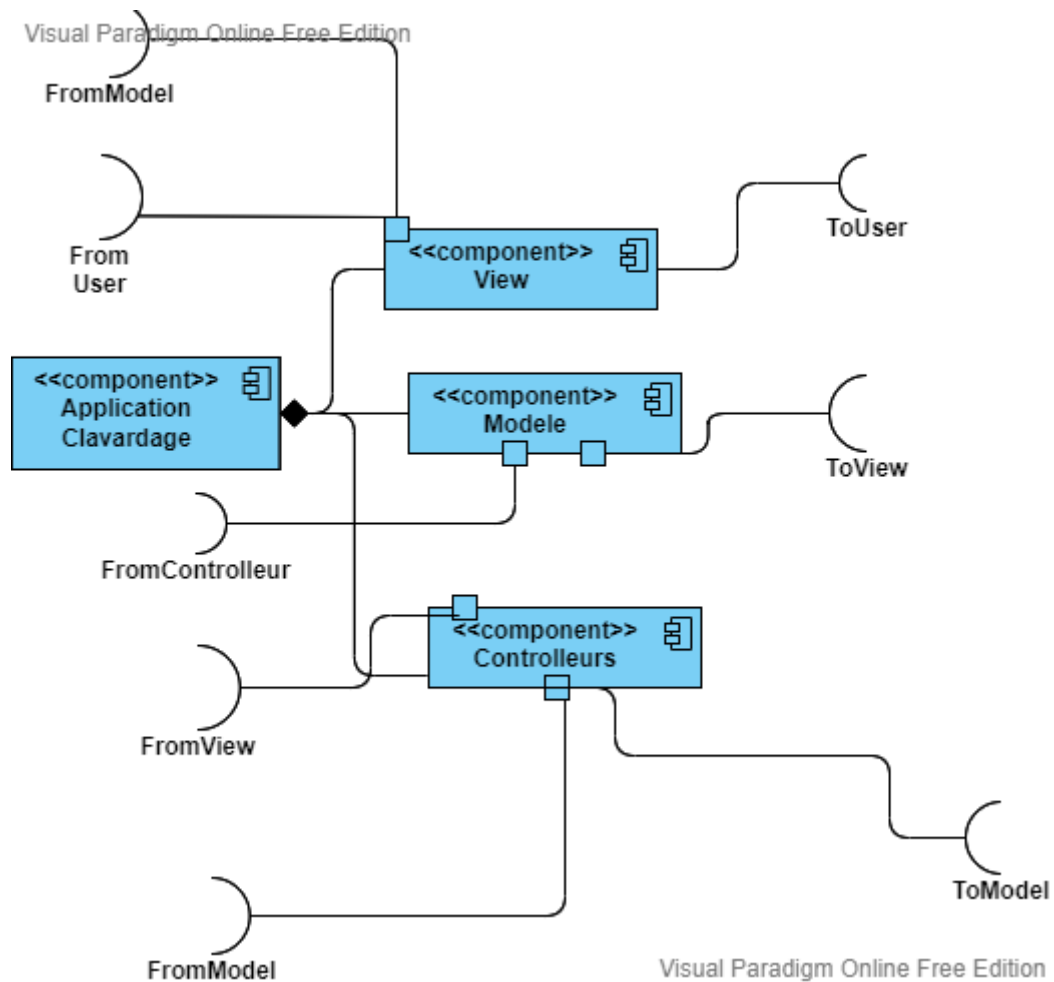


Diagramme des composants fait en plus du plan prévu initialement spécifiant les interactions entre MVC (modèle, vue, Contrôleur).