heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

PROJET DE SEMESTRE 2015

Chator - Gestion

Mélanie Huck
Jan Purro
Bastien Rouiller
Miguel Santamaria
Benoist Wolleb

Professeur René Rentsch

TABLE DES MATIÈRES

1 CAHIER DES CHARGES

1.1 PAIRS

Miguel Santamaria (Chef de groupe)
Mélanie Huck
Jan Purro
Bastien Rouiller
Benoist Wolleb
René Rentsch (Professeur et « client » du projet).

1.2 DESCRIPTION DU PROJET

1.2.1 Généralités

Ce projet représente une application de messagerie instantanée (chat) de type client-serveur. Des utilisateurs peuvent communiquer via des canaux de discussion (salles) privés ou publics, modérés par des personnes possédant des droits d'administration.

Le projet est donc constitué d'un logiciel client et d'un logiciel serveur.

L'application cliente fournit un module d'inscription et de connexion, un module de gestion de salles, ainsi que le module *principal* qui représente les flux de discussion.

L'application serveur s'occupera de gérer les transactions de données (messages, …), ainsi que les lectures/écritures dans une base de données, de manière sécurisée et optimisée.

Les points qui suivent décrivent de manière rigoureuse les différentes fonctionnalités du programme.

1.2.2 Fonctionnalités / Configuration

1.1.1.1 Salles de discussion

Tout utilisateur est libre de rejoindre une salle publique. Quant aux salles privées, elles bénéficieront de chiffrement. Il y a deux types de salles privées : visible ou non visible. Pour les salles visibles, l'utilisateur doit faire une demande d'adhésion alors que pour les salles non visibles, c'est à l'administrateur de l'ajouter.

2.1.1.1 Administration

L'administrateur est l'utilisateur qui a créé une salle.

Lors de la création d'une salle, l'administrateur donne un nom, un nombre maximal de message stockés, le type de salle et éventuellement une image. Il peut à tout moment modifier ces paramètres ainsi qu'exclure un utilisateur d'une discussion. Il peut également se faire épauler par un autre utilisateur en le nommant coadministrateur, ce dernier possédera ainsi les mêmes droits sur la salle.

3.1.1.1 Processus de connexion à un serveur

Lorsque l'utilisateur lance l'application, il fournit un nom d'utilisateur et un mot de passe. Il peut également s'inscrire s'il ne possède pas compte. Il doit ensuite saisir l'IP du serveur manuellement ainsi qu'un numéro de port.

4.1.1.1 Fonctionnalités utilisateur

Une fois dans une salle de discussion, un utilisateur peut envoyer un message. Après l'envoi de ce message, il sera toujours possible de le modifier ou de le supprimer. Un utilisateur peut faire partie de plusieurs salles. L'utilisateur peut voir les différents membres dans les salles dans lesquelles il se trouve. Il est avisé des nouveaux messages survenant dans une salle différente de laquelle il consulte actuellement. L'utilisateur peut décider d'afficher ou non l'heure d'arrivée des messages. Il est en mesure de modifier les informations relatives à son compte.

Lors de la reconnexion d'un utilisateur, l'application cliente récupère les messages envoyés durant son absence dans les différentes salles de chat auxquelles l'utilisateur est abonné.

1.2.3 Sécurité

1.1.1.1 *Chiffrement des messages*

L'application offre la possibilité, au travers des salles privées, de chiffrer les messages échangés.

Un message est chiffré à l'envoi par l'utilisateur le postant et déchiffré à la réception par les autres utilisateurs. Le serveur transmet et stocke le message sans le déchiffrer.

Pour que tout le monde puisse déchiffrer les messages à tout moment il faut donc que la clé utilisée pour déchiffrer soit la même pour tous. Pour cette raison ainsi que pour des raisons de performances, un mécanisme de chiffrement à clé symétrique (secrète) sera utilisé pour chiffrer les messages.

Chaque salle possédera donc une clé de chiffrement. Lorsqu'un utilisateur est accepté dans une salle privée, l'administrateur qui l'a accepté partage la clé avec le nouveau venu. Cet échange doit se faire de manière sécurisée. Un mécanisme de chiffrement utilisant une clé publique permet de le faire efficacement.

2.1.1.1 Génération des clés

La clé secrète d'une salle est générée aléatoirement au moment de la création de celle-ci par son administrateur.

Chaque utilisateur possède également une clé publique pour permettre les échanges de clés secrètes. Il possède donc également une clé privée. La paire de clé est générée aléatoirement à la création de l'utilisateur, il peut en générer une nouvelle s'il l'estime nécessaire.

3.1.1.1 Échange des clés

Quand un nouvel utilisateur rejoint une salle privée il doit pouvoir lire les messages et chiffrer les siens. Il doit donc connaître la clé.

L'administrateur qui accepte la demande de l'utilisateur de rejoindre la salle partagera la clé avec le nouvel utilisateur. Il chiffrera la clé de la salle à l'aide de la clé publique du nouveau venu. Celui-ci pourra ensuite la déchiffrer à l'aide de sa clé privée.

4.1.1.1 Stockage des clés

Il faut stocker les clés secrètes des salles privées dont un utilisateur est membre ainsi que sa clé privée et publique, afin qu'il puisse lire et écrire des messages à tout moment. Ce stockage doit être fait de manière sécurisée.

Il s'agira de chiffrer les clés secrètes et la clé privée à l'aide du mot de passe de l'utilisateur. Les clés chiffrées seront ensuite stockées dans la base de données. Cela permettra à l'utilisateur d'avoir accès aux salles privées même s'il change de machine.

La clé publique sera stockée en clair dans la base de données, afin d'être accessible à tous à tout moment par les autres utilisateurs.

5.1.1.1 Gestion des mots de passe

Le mot de passe d'un utilisateur est utilisé pour l'authentifier et comme clé pour le chiffrement de sa clé privée et des clés secrètes qui lui sont connues.

Il sera salé et hashé avant d'être stocké dans la base de donnée.

Il sera également nécessaire de vérifier que les mots de passe choisi par les utilisateurs soient robustes.

6.1.1.1 Connexion sécurisée au serveur

La connexion entre les clients (utilisateurs) et le serveur est sécurisée au moyen de TLS ce qui permettra un échange sécurisé des données ainsi que l'authentification du serveur auprès des clients. Ce dernier point requiert donc la génération d'un certificat pour le serveur.

1.2.4 Serveur

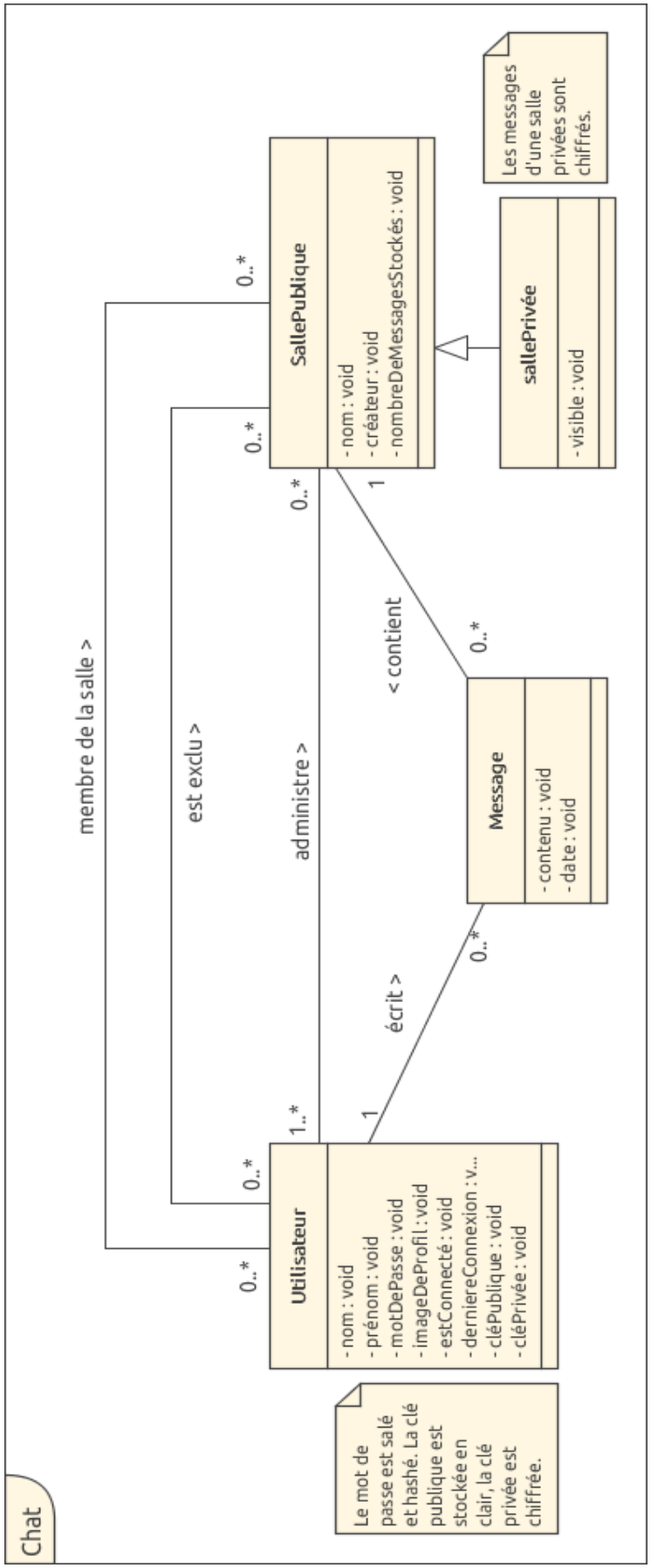
Nous ne réaliserons pas d'interface graphique pour le serveur ; en effet, une simple console d'administration suffit amplement pour les besoins.

Lors du lancement du serveur, l'utilisateur devra éventuellement indiquer certains arguments, comme le port, ou le nombre de connexions maximum par exemple.

Certains logs seront affichés en temps réel (inscription d'un utilisateur, création d'une salle, …), afin de pouvoir debugger facilement l'application, et de pouvoir garder une trace de ce qui s'est déroulé durant les derniers instants.

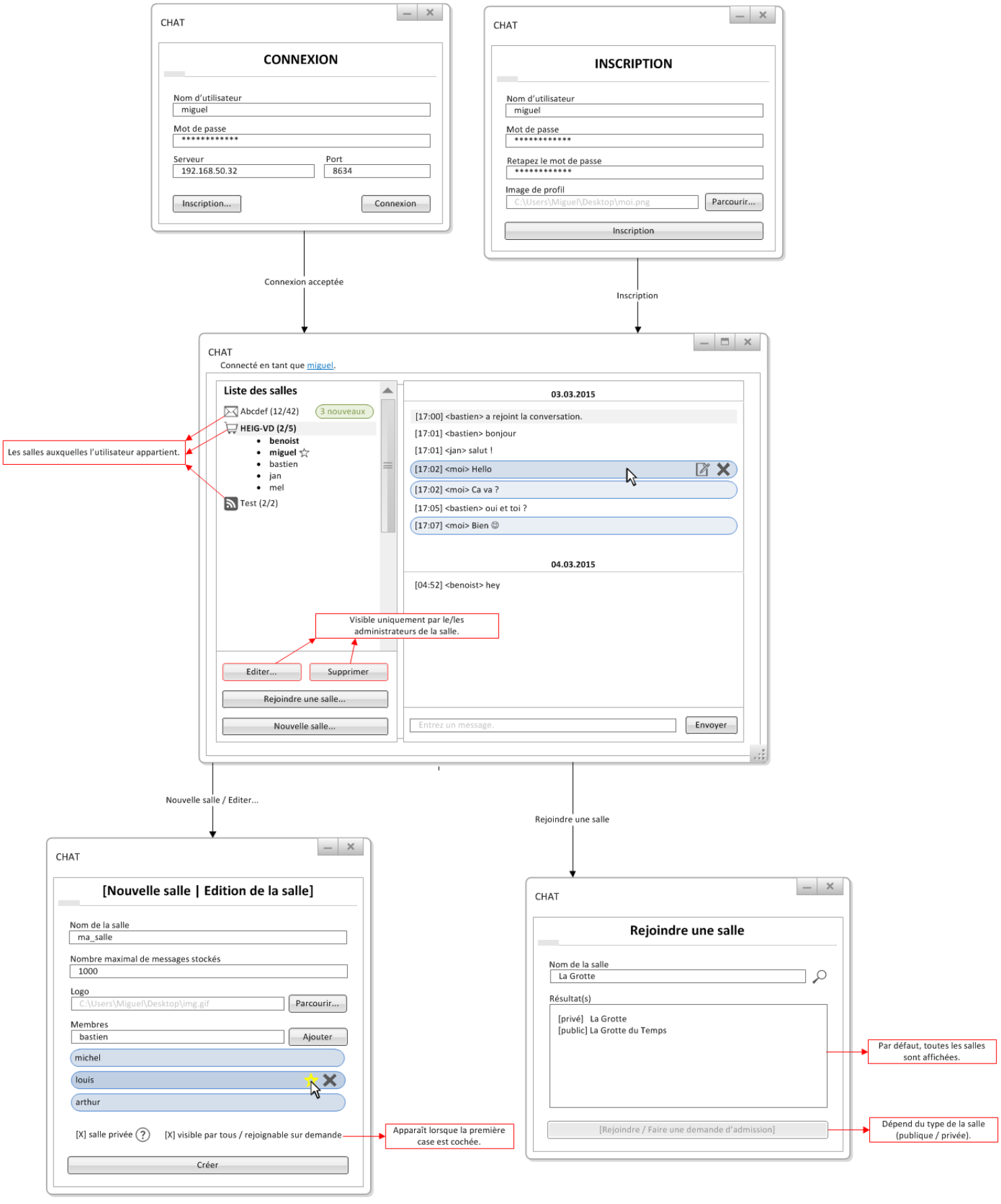
1.2.5 Base de données

Nous allons utiliser une base de données SQLite afin de stocker les données entant que fichier indépendant et permettre ainsi de lancer le serveur sans posséder de serveur plus lourd. La base de données est constituée des éléments suivants. Ce diagramme va certainement être modifié par la suite.


Chat
Utilisateur
- nom : void
- prénom : void
- motDePasse : void
- imageDeProfil : void
- estConnecté : void
- dernièreConnexion : v...
- clépublique : void
- cléPrivée : void
SallePublique
- nom : void
- créateur : void
- nombreDeMessagesStockés : void
sallePrivée
- visible : void
Message
- contenu : void
- date : void
Le mot de passe est salé et hashé. La clé publique est stockée en clair, la clé privée est chiffrée.
Les messages d'une salle privées sont chiffrés.
membre de la salle >
est exclu >
administre >
< contient
écrit >
0..*
1..*
1

1.2.6 Interface

Voici une ébauche de l'interface utilisateur (GUI), qui nous permettra de nous orienter plus facilement dans le projet, en partant sur des bases propres et solides. A noter que rien n'est définitif, et que ce schéma peut changer à tout moment ; toutefois, les idées générales y sont présentes.



1.2.7 Protocole de communication

La communication se fait toujours d'un client vers le serveur, jamais de manière décentralisée. Le serveur sert alors de relais.

La transmission des messages entre les clients et le serveur s'effectue via un socket TCP (un seul par client).

Le serveur reçoit les connexions des clients sur un port supérieur à 1023 afin de ne pas dépendre des droits d'administration de la machine pour s'exécuter.

Les données échangées dans le socket sont binaires et contiennent un code d'opération ou code d'erreur et les éventuelles données à traiter :

Longueur du message	Code d'opération / erreur	Longueur donnée 1	Donnée 1	...	Longueur donnée n	Donnée n
Toujours présents		*Optionnels*				

Les messages échangés entre un client et le serveur peuvent être de natures différentes:

- Informations de login ou d'enregistrement (nom/prénom, pseudo, mot de passe hâché)
- Informations concernant les salles de discussion rejointes par l'utilisateur ou disponibles (publiques ou privées-visibles)
- Requête de création de salles de discussion
- Requête de modification de salles de discussion
- Requête d'abonnement à une salle de discussion
- Acceptation d'un utilisateur dans une salle de discussion
- Requête de modification de login (pseudonyme, mot de passe)
- Transmission de messages dans les salles de discussion
- Transmission de méta-informations des utilisateurs (pseudonyme, avatar, date de dernière connexion, clé publique, salles jointes, salles administrées)
- Transmission des clés entre utilisateurs
- Requête d'édition de messages d'une salle de discussion
- Code d'erreur

1.2.8 Extensions

Bien que très motivés par ce projet, nous ne pourrons évidemment pas réaliser toutes les idées sur lesquelles nous avons cogité. Pour cela, une priorité plus basse a été attribuée à certains points, qui restent parfaitement réalisables si le temps nous le permet :

- Lors du démarrage du client, scanner le réseau pour rechercher les serveurs disponibles, au lieu de demander l'adresse IP.
- Pouvoir envoyer des messages privés à un autre utilisateur ; actuellement cela reste possible en créant une salle pour deux utilisateurs.
- Gérer l'envoi de fichiers.
- La possibilité de changer la clé utilisée pour chiffrer les messages d'une salle. Cela nécessitera de déchiffrer l'ensemble de l'historique et de le chiffrer à nouveau avec la nouvelle clé.

1.2.9 Technologies utilisées

Pour nos différents besoins nous avons fait le choix de différentes technologies :

- Le C++ (norme 2011) comme langage de production
- Qt comme framework graphique et réseau
- SQLite comme moteur de base de données
- La bibliothèque OpenSSL pour gérer la sécurité de l'application (TLS, hâchage, algorithme de chiffrement symétrique, algorithme de chiffrement asymétrique)

2 PLANIFICATIONS

Les deux planifications sont jointes en annexe, pour plus de lisibilité.

2.1 PLANIFICATION INITIALE

Pour rappel, voici la planification qui était initialement prévue :

GANTT

Tâche	Description	Durée	Semaine 4	5	6	7	8	9	10	11	12	13	14	15
Analyse		**2**												
T1	Répartition des tâches	1												
T2	Etablissement des conventions de travail	1												
T3	Schéma DB	2												
T4	Diagramme d'utilisation / Design	2												
T5	Schéma UML de l'application	2												
T6	Spécification du protocole	2												
T7	Aspects sécuritaires	2												
T8	Familiarisation avec les environnements	1												
Conception des fonctionnalités de base		**3**												
T9	Implémentation de la DB	1												
T10	Serveur	**2**												
T10-1	Initialisation	1												
T10-2	Gestion sockets	2												
T11	Module utilisateur	**3**												
T11-1	Inscription	2												
T11-2	Connexion	1												
T12	Module messagerie	**3**												
T12-1	Envoi/Réception de messages	2												
T12-2	Synchronisation	2												
T13	Module gestion de salles	**1**												
T13-1	Création de salles	1												
Conception des fonctionnalités avancées		**4**												
T14	Serveur	**2**												
T14-1	TLS	2												
T15	Module utilisateur	**1**												
T15-1	Edition du profil	1												
T16	Module messagerie	**4**												
T16-1	Chiffrement	3												
T16-2	Edition / Suppression de messages	1												
T17	Module gestion de salles	**4**												
T17-1	Edition	1												
T17-2	Suppression	1												
T17-3	Adhésion	2												
T17-4	Exclusion / Promotion / Droits	3												
T17-5	Partage de clés	2												
Tests														
T18	Tests unitaires	7												
T19	Tests fonctionnels	5												
T20	Tests finaux	2												
Documentation														
T21	Journal de bord	8												
T22	Rapport	11												
T23	Instructions de déploiement	2												
T24	Manuel d'utilisation	2												

Figure 1 : Planification initiale.

2.2 PLANIFICATION APPLIQUÉE

Et voici maintenant la planification réelle du projet qui a été appliquée :

GANTT

Tâche	Description	Durée	Semaine 4	5	6	7	8	9	10	11	12	13	14	15
Analyse		4												
T1	Répartition des tâches	3												
T2	Etablissement des conventions de travail	3												
T3	Schéma DB	1												
T4	Diagramme d'utilisation / Design	1												
T5	Schéma UML de l'application	3												
T6	Spécification du protocole	1												
T7	Aspects sécuritaires	1												
T8	Familiarisation avec les environnements	4												
Conception des fonctionnalités de base		9												
T9	Implémentation de la DB	2												
T10	Serveur	3												
T10-1	Initialisation	2												
T10-2	Gestion sockets	3												
T11	Module utilisateur	7												
T11-1	Inscription	6												
T11-2	Connexion	4												
T12	Module messagerie	4												
T12-1	Envoi/Réception de messages	2												
T12-2	Synchronisation	3												
T13	Module gestion de salles	3												
T13-1	Création de salles	3												
Conception des fonctionnalités avancées		6												
T14	Serveur	6												
T14-1	TLS	1												
T14-2	Adaptations	5												
T15	Module utilisateur	2												
T15-1	Informations de profil (anciennement édition)	2												
T16	Module messagerie	3												
T16-1	Chiffrement	2												
T16-2	Edition / Suppression de messages	2												
T17	Module gestion de salles	4												
T17-1	Edition	3												
T17-2	Suppression	1												
T17-3	Adhésion	3												
T17-4	Exclusion / Promotion / Droits	3												
T17-5	Partage de clés	3												
Tests														
T18	Tests unitaires	7												
T19	Tests fonctionnels	5												
T20	Tests finaux	2												
Documentation														
T21	Journal de bord	8												
T22	Rapport	11												
T23	Instructions de déploiement	2												
T24	Manuel d'utilisation	2												

Figure 2 : Planification réelle.

Légendes :

- *Nombres en orange : tâches plus longues que ce qui avait été planifié.*
- *Nombres en vert : tâches plus courtes que ce qui avait été planifié.*
- *Nombres en bleu : tâche qui n'était pas planifiée dans la planification initiale (T14-2).*
- *Nombres en noir : tâche qui a suivi la planification initiale sans encombre particulière.*

2.3 COMPARAISON

Certaines tâches se sont étendues en longueur (comme le module utilisateur dans la conception des fonctionnalités de bases), car il a fallu se mettre d'accord sur certains points et y implémenter la sécurité qui nous a rapidement pris du temps. Il s'agit aussi quelquefois de choses que nous avions mis de côté, car moins importantes.

Les tâches nous ayant donné le plus de fil à retordre sont l'inscription, ainsi que l'adhésion à une salle.

Même si le schéma pourrait paraître très (trop) orange à premier vue, il ne s'agit pas forcément d'un point néfaste : il s'agit pour beaucoup de tâches de la partie sécurité qui nous a pris un peu plus de temps que prévu ; nous avons aussi comptabilisé les formations, dans ces tâches. Nous sommes tout de même arrivés à nos fins sans véritable encombre.

La partie "Conception des fonctionnalités avancées -> Serveur -> Adaptations" a dû être ajoutée, afin de pouvoir afficher les adaptations qui ont dû être faites au serveur, du côté sécuritaire.

3 JOURNAL DE BORD

3.1 SEMAINE 1 – 17.02.2015

Présentation du projet et composition du groupe.

3.2 SEMAINE 2 – 03.03.2015

Réflexion sur le sujet du projet + réalisation du cahier des charges.

3.3 SEMAINE 3 – 10.03.2015

1. Finalisation du cahier des charges.
2. Réalisation de la planification.
3. Répartition des tâches d'analyse :
    a. **Miguel** : finalisation de l'interface ; réalisation du diagramme d'utilisation et de tests sur Qt.
    b. **Jan** : recherches plus profondes sur la sécurité ; réalisations de tests en C++.
    c. **Benoist** : spécification du protocole.
    d. **Bastien + Mélanie** : mise à jour de la base de données.

3.4 SEMAINE 4 – 17-03.2015

1. Discussion et validation du cahier des charges, avec M. Rentsch.
2. Mise en commun des tâches attribuées à la fin de la séance passée :
    a. Interface : OK, mettre à jour pour les fonctionnalités de bannissement.
    b. Protocole : OK.
    c. Sécurité : analyse des tests effectués en C++ ; OK.
    d. Base de données : mise à jour en commun, afin de l'adapter à la compréhension et aux besoins de chacun.
3. Conventions de codage : voir PDF sur ENSI.
4. Codage des sources : UTF-8.
5. Répartition des nouvelles tâches :
    a. Installation de OpenSSL sous Windows ; réalisation de tests sur les fichiers de tests de Jan.
    b. Finaliser l'analyse.

3.5 SEMAINE 5 – 24.03.2015

1. Réalisation du schéma de classe.
2. Répartition des tâches :
    a. **Benoist** : serveur.
    b. **Bastien** : base de données.
    c. **Mélanie** : interface « utilisateur » (connexion, inscription, édition du compte).
    d. **Miguel** : interface principale (fenêtre de chat).
    e. **Jan** : interface « salle » (création/édition de salle, adhésion).

3.6 SEMAINE 6 – 31.03.2015

1. Mise au point des réalisations : OK.
2. Création d'un code de base, en commun.
3. Réponse aux questions.
4. Mise à jour de la documentation.
5. Validation de la répartition des tâches :
    a. **Benoist** : serveur.
    b. **Jan** : salles.
    c. **Mélanie** : utilisateurs.
    d. **Miguel** : chat.
    e. **Bastien** : DB + modèle + documentation.

3.7 SEMAINE 7 – 14.04.2015

1. Mise en commun : nous avons pris un peu de retard durant la semaine de relâche. Nous devions nous voir, mais cela n'a malheureusement pas pu être fait, à cause de diverses raisons (vacances à l'étranger, plusieurs travaux en parallèles, …). Il faudra rattraper cela durant cette semaine.
2. De ce fait, et après discussion, les tâches ont été redistribuées de la manière suivante :
    a. **Benoist** : Initialisation du serveur + gestion de sockets de communication.
    b. **Jan** : initialisation de l'implémentation de la vue et du contrôleur du module de gestion de salle (création d'une nouvelle salle).
    c. **Mélanie** : initialisation de l'implémentation de la vue et du contrôleur du module utilisateur (connexion).
    d. **Miguel** : création de quelques modèles « exemples » de données, qui seront utilisés comme bases + initialisation de l'implémentation de la vue et du contrôleur du module chat (synchronisation des salles, utilisateurs et messages) + gestion de l'avancement des travaux.
    e. **Bastien** : création d'un modèle de données de base, qui sera utilisé dans l'application.

3.8 SEMAINE 8 – 21.04.2015

1. Mise en commun : un bon travail a été effectué durant la semaine 7, ce qui nous a permis de combler le retard :
    a. **Benoist** : initialisation terminée, en phase de gestion des sockets.
    b. **Jan** : l'ajout d'une salle est terminé ; il ne manque plus que la communication avec le serveur.
    c. **Mélanie** : un peu de retard a été pris de ce côté-là ; nous nous verrons durant la semaine pour rattraper cela.
    d. **Miguel** : la synchronisation avec les données de tests est implémentée correctement, envoi des messages en cours d'implémentation.
    e. **Bastien :** DB ok et fonctionnelle avec le serveur de Benoist.
2. Les vues, ainsi qu'une grande partie des contrôleurs ont été implémentés (pour la phase 1 du projet) ; le travail de la semaine sera de réaliser les communications serveur-clients correctement, afin de pouvoir faire une présentation convaincante des résultats.

3. Les modules ont été distribués de manière plus ou moins semblables à la semaine passée, si ce n'est que :
    a. Jan prête main-forte à Benoist quant au serveur.
    b. Bastien se joint à Mélanie pour l'implémentation du module User.

    Idéalement, nous devrions être apte à pouvoir exécuter une application dans laquelle les modules sont liés et communiquent entre eux (parfois de manière très basique), lors de la présentation de la semaine prochaine.

Nous nous sommes vus vendredi et samedi et lundi, afin de pouvoir travailler correctement ; un bon travail a été fait du côté du module User. Les modules communiquent désormais entre eux, et avec le serveur.

3.9 Semaine 9 – 28.04.2015

1. Présentation intermédiaire.
2. Bilan : la présentation s'est bien déroulée en général ; nous avons cependant mal introduit la partie de comparaison des planifications initiales et réelles, en n'allant pas directement au but (retard ou pas ?).
   Nous sommes dans les temps ; il reste quelques détails à finaliser pour terminer l'implémentation de base. Ceux-ci seront réalisés durant la semaine, en plus de l'initialisation de la phase d'implémentation des fonctionnalités avancées.
3. Rafraîchissement des rôles, afin que tout le monde soit d'accord :
    a. **Benoist** : finalisation des méthodes d'inscription utilisateur, ainsi que celles permettant de modifier/supprimer un message, du côté serveur, et initialisation de la phase de gestion de sécurité.
    b. **Jan** : finalisation des créations de salles, puis rédaction de documentation sur les implémentations de la sécurité, dans un second temps.
    c. **Mélanie** : mise à jour de la documentation en général et centralisation ; création d'une ébauche de GUI pour l'édition d'un compte utilisateur.
    d. **Miguel** : gestion des notifications de nouveaux messages ; implémentation de l'édition/suppression de messages.
    e. **Bastien** : finalisation de la fonctionnalité d'inscription utilisateur ; édition utilisateur avec Mélanie.

3.10 Semaine 10 – 05.05.2015

1. Mise en commun : nous avons pris du retard durant cette semaine, à cause du travail auxiliaire que nous avions dans les autres cours (rendus de laboratoires, tests, …). Voici ce qui a tout de même été fait :
    a. **Benoist** : rien.
    b. **Jan** : il reste un bug mineur dans la création de salles ; la rédaction de la documentation a été faite en partie.
    c. **Mélanie** : rien.
    d. **Miguel** : gestion des notifications ok ; rien n'a été fait quant à l'édition/suppression de messages.
    e. **Bastien** : pour l'inscription, il reste à implémenter la communication avec le serveur ; rien du côté de l'édition de compte.

2. Il faudra donc rattraper rapidement ce retard, afin d'éviter qu'il s'incrémente de manière critique ; cette semaine risquant d'être aussi chargée, il nous faudra rattraper tout ça durant les prochaines.
3. A faire : rattraper le retard, et avancer sur les différents points respectivement attribués.

3.11 Semaine 11 – 12.05.2015

1. Mise en commun :
    a. **Benoist & Miguel** :
        i. Edition des messages ; il reste un petit bug à résoudre.
        ii. Gestion des connexions/déconnexions d'un utilisateur (broadcast aux personnes présentes dans les salles auxquelles il appartient, gestion de l'affichage en conséquence, …).
    b. **Jan** : recherches sécuritaires, création de prototypes à utiliser, et initialisation de l'implémentation de ceux-ci ; réalisation de modes d'emploi pour les collègues.
    c. **Mélanie** : mise à jour, restructuration, et centralisation de toute la documentation.
    d. **Miguel** : implémentations diverses (interface, bugs mineures, fonctionnalités mineurs du style « Fichier > Quitter », …).
    e. **Bastien** : implémentation de l'inscription ; il reste encore un peu de travail.
2. Discussions à propos de la procédure d'inscription, mise en commun des idées et définition détaillée de la fonctionnalité.
3. Mise à jour de la base de données (ajout d'un champ + étendue du rôle d'un autre champ).
4. Distribution des tâches pour la semaine prochaine :
    a. **Benoist** (tout côté serveur) :
        i. Mise à jour de l'inscription selon ce qui a été discuté.
        ii. Édition du compte de l'utilisateur connecté.
        iii. Édition/Suppression de salles.
        iv. Adhérer/Quitter une salle.
        v. Suppression d'un message.
        vi. Finalisation de l'édition d'une salle.
        vii. Gestion des demandes d'inscriptions à une salle privée.
        viii. Gérer la limitation du nombre de message maximal.
    b. **Jan** :
        i. Finalisation de l'édition d'une salle, côté client.
        ii. Finalisation de l'implémentation de « l'API » sécuritaire qui sera utilisée par les autres. Idéalement, tout cela devrait être terminé à la fin de la semaine, afin que nous puissions développer les fonctionnalités sécuritaires de l'application dès mardi prochain.
    c. **Mélanie** :
        i. Mise à jour de la documentation.
        ii. Edition du compte.
        iii. Adhésion à une salle.
    d. **Miguel** (côté client) :
        i. Finalisation de la modification des messages.
        ii. Suppression de messages.
        iii. Suppression de salles.

iv. Quitter une salle.
v. Gestion des demandes d'inscriptions à une salle privée.
vi. Redimensionnement dynamique du module Chat.
vii. Optimisation des chargements des messages, car ceux-ci sont beaucoup trop lents lorsque la salle sélectionnée a changé.

e. **Bastien** :
i. Finalisation de l'inscription.
ii. Edition du compte.
iii. Adhésion à une salle.

3.12 SEMAINE 12 – 19.05.2015

1. Mise en commun.
   Un problème s'est rapidement fait sentir au niveau du temps restant ; après discussion et validation de la part de tous les membres, nous avons donc décidé de laisser tomber la fenêtre de re-génération de la paire de clés privées / publiques de l'utilisateur, parce cela s'avèrerait beaucoup trop lourd par rapport à ce que nous avions pensé au début. Il ne s'agit aussi pas d'une fonctionnalité prioritaire. S'en est suivi de même pour le changement de type d'une salle lors de son édition : nous bloquerons cette fonctionnalité.
   Nous avons travaillé sans relâche durant la semaine passée ; voici ce qui n'a pas pu être fait par rapport à la distribution de la semaine passée :
   a. **Benoist** (tout côté serveur) :
      i. Édition du compte de l'utilisateur connecté.
      ii. Édition d'une salle, pas terminé.
      iii. Ashésion à une salle, pas terminé.
      iv. Gestion des demandes d'inscriptions à une salle privée.
   b. **Jan** :
      i. Finalisation de l'édition d'une salle, côté client, il reste quelques bugs.
   c. **Mélanie** :
      i. Edition du compte.
      ii. Adhésion à une salle.
   d. **Miguel** (côté client) :
      i. Suppression de salles, il reste un bug.
      ii. Gestion des demandes d'inscriptions à une salle privée.
   e. **Bastien** :
      i. Edition du compte.
      ii. Adhésion à une salle.
2. A partir d'aujourd'hui, nous allons commencer à implémenter la sécurité au sein de notre application. Voici donc la répartition des nouvelles tâches à faire pour la semaine suivante :
   - **Bastien**
     - Connexion avec sécurité.
     - Inscription avec sécurité.
     - Edition de compte.
   - **Mélanie** :
     - Mise à jour des requêtes SQL, car celles-ci ne sont pas optimisées.

  - Rédaction de la procédure de test.
  - Rédaction du manuel d'utilisateur.
  - Documentation de la communication client-serveur.
- **Jan** :
  - Création de salle avec la sécurité.
  - Edition de salle.
  - Adhésion (+ sécurité).
- **Miguel** :
  - Correction des bugs.
  - Gestion des clés des salles (stocker la clé secrète de la salle dans le modèle).
  - Chiffrement/Déchiffrement des messages.
  - Gestion des notifications (stocker dans le modèle).
  - Bannir.
  - Afficher les images.
  - Réalisation de la fenêtre d'à-propos.
- **Benoist** :
  - Sécurité du serveur.
  - Rejoindre/Quitter une salle (notifications).
  - Sorties du serveur.

3.13 SEMAINE 13 – 26.05.2015

La fin du projet commence à se faire ressentir ; un grand travail a été fourni durant la semaine.

1. Mise en commun. Nous avons pu implémenter la majeure partie des tâches réparties la semaine passée, si ce n'est :
   a. **Bastien**
      i. Edition de compte.
   b. **Mélanie** :
      i. Documentation de la communication client-serveur.
   c. **Jan** :
      i. Edition de salle, il reste des bugs.
      ii. Adhésion, pas terminée.
   d. **Miguel** :
      i. Chiffrement/Déchiffrement des messages, n'a pas pu être testé.
      ii. Gestion des notifications (stocker dans le modèle).
      iii. Bannir.
   e. **Benoist** :
      i. Rejoindre/Quitter une salle (notifications), pas terminé.

   A noter que nous avons un peu de retard ; mais celui-ci est rattrapable durant la dernière semaine. Nous avons laissé tomber l'édition de compte (qui deviendra désormais une simple fenêtre d'affichage des détails du compte), ainsi que le bannissement d'un utilisateur, car il s'agit de fonctionnalités mineures que nous n'aurons pas le temps d'implémenter correctement.
2. Voici ce que nous devrons faire d'ici la semaine prochaine, pour le rendu :
   a. En priorité, terminer le développement.

   b. Mettre à jour nos documentations respectives.
   c. Tester l'application de fond en comble.
   d. Préparer le rendu.
3. Séance de dimanche 31.05.2015 : le développement est terminé, et la documentation arrive gentiment à son terme. Nous nous verrons demain pour mettre tout au clair et préparer le rendu.

3.14 SEMAINE 14 – 02.06.2015

Derniers contrôles et rendu du projet au terme de la séance, il faudra préparer la présentation pour la semaine prochaine.

4 RÉPARTITION DES HEURES DE TRAVAIL

SEMAINE	BASTIEN	BENOIST	JAN	MELANIE	MIGUEL	TOTAL
1	Néant	Néant	Néant	Néant	Néant	**Néant**
2	1	1	1	1	1	**5**
3	3	2	2	2	3	**12**
4	3.5	4	4	6	2.5	**20**
5	8	4	5	7	4.5	**28.5**
6	6	4	3	6	3	**22**
7	5	6.5	12	3	9.5	**36**
8	8	16	20.5	10	11	**65.5**
9	5	0.5	11	0.5	4	**21**
10	4	6.5	2	5	5.5	**23**
11	7.5	7	9.5	6	11.5	**41.5**
12	12	12.5	9	7	12.5	**53**
13	28	35	36	37	37.5	**173.5**
TOTAL	**91**	**99**	**115**	**90.5**	**105.5**	**501**

Nous pouvons observer qu'en règle générale, tout le monde a fourni un bon travail, les heures étant assez équilibrées ; nous avons fourni une moyenne totale d'environ 7.7 heures de travail par semaine et par personne. Certaines semaines ont été plus chargées que d'autres, notamment la semaine 13 qui s'est avérée assez sympathique ; nous pouvons aussi remarquer qu'il y a eu un pic de travail durant la semaine 8. Nous avions en effet un peu plus de temps à disposition que d'habitude. Les semaines 9 et 10 n'étaient pas des semaines très rentables du tout, car nous étions très chargés dans les autres cours (laboratoires et travaux écrits).

Voici un petit graphique illustrant ces données de manière un peu plus visuelle :

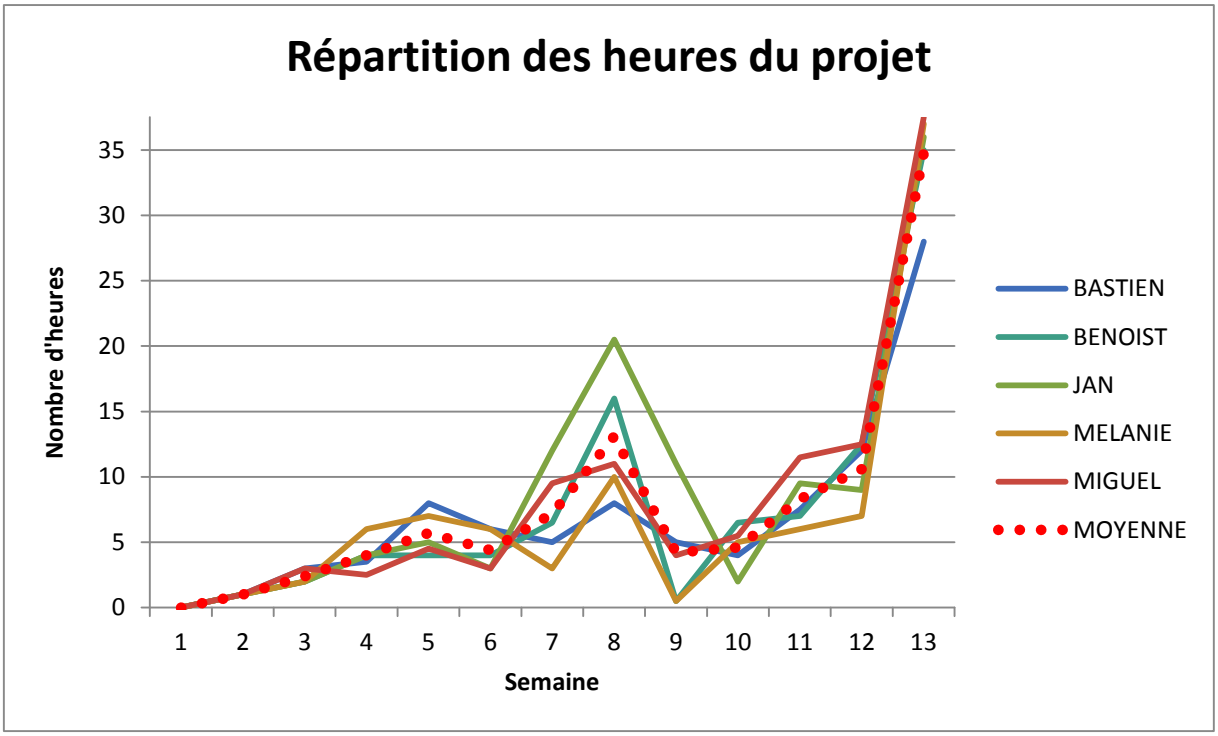


Figure 3 : Graphique de répartition des heures durant le projet.