

# Rapport de Projet Final

Informatique et Gestion

Polytech Montpellier

## POLYMESSAGES

**Présenté par Marvin BONTEMPS, Eri AGNESE et Lucas NOUGUIER  
le 01 Juin 2022**

**Sous la direction de Thomas GODEL**

**Devant le jury composé de**

**Thomas GODEL**



**UNIVERSITÉ  
DE MONTPELLIER**

# Contents

<b>1</b>	<b>Protocole de communication</b>	<b>2</b>
<b>2</b>	<b>Architecture</b>	<b>4</b>
<b>3</b>	<b>Répartition du travail</b>	<b>4</b>
<b>4</b>	<b>Exécution du code</b>	<b>6</b>
<b>5</b>	<b>Difficultés</b>	<b>6</b>

# 1 Protocole de communication

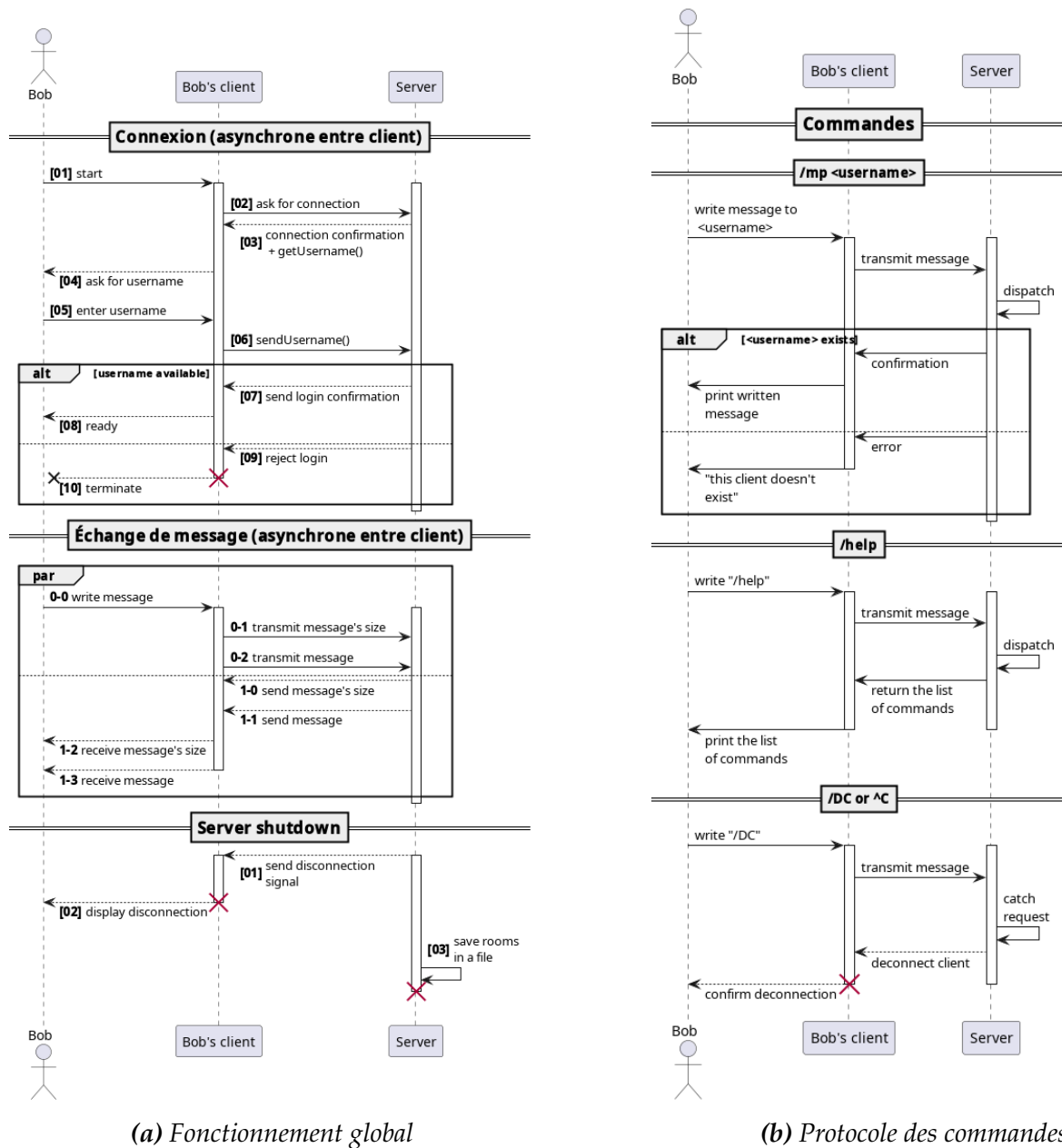
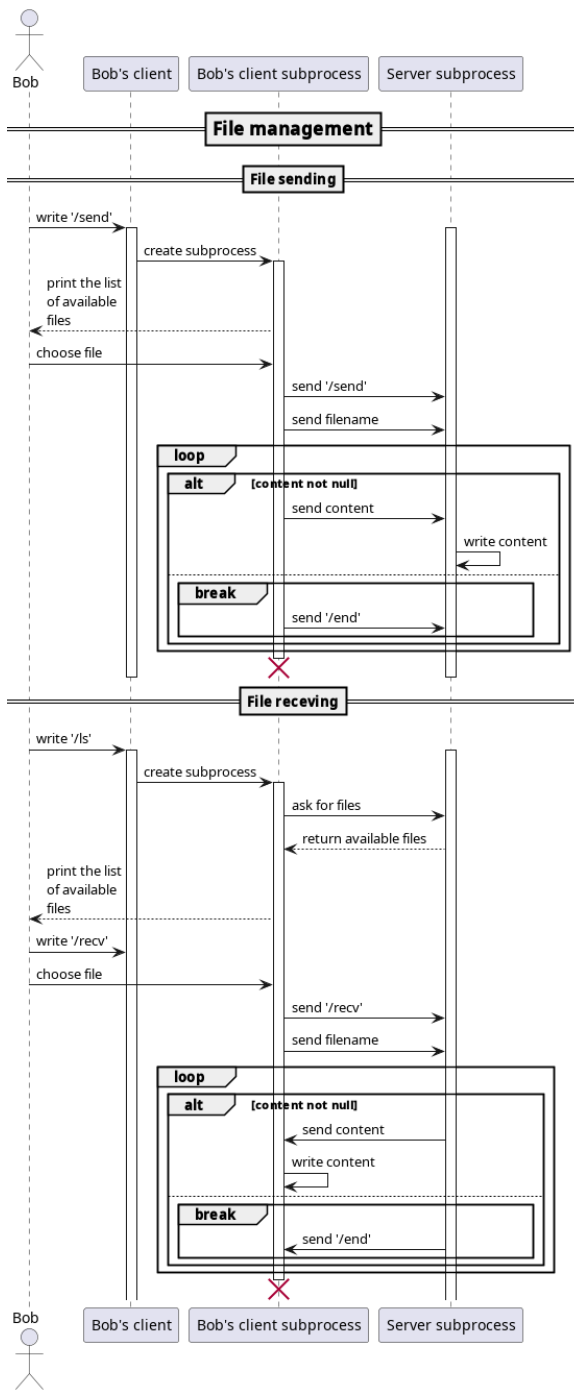
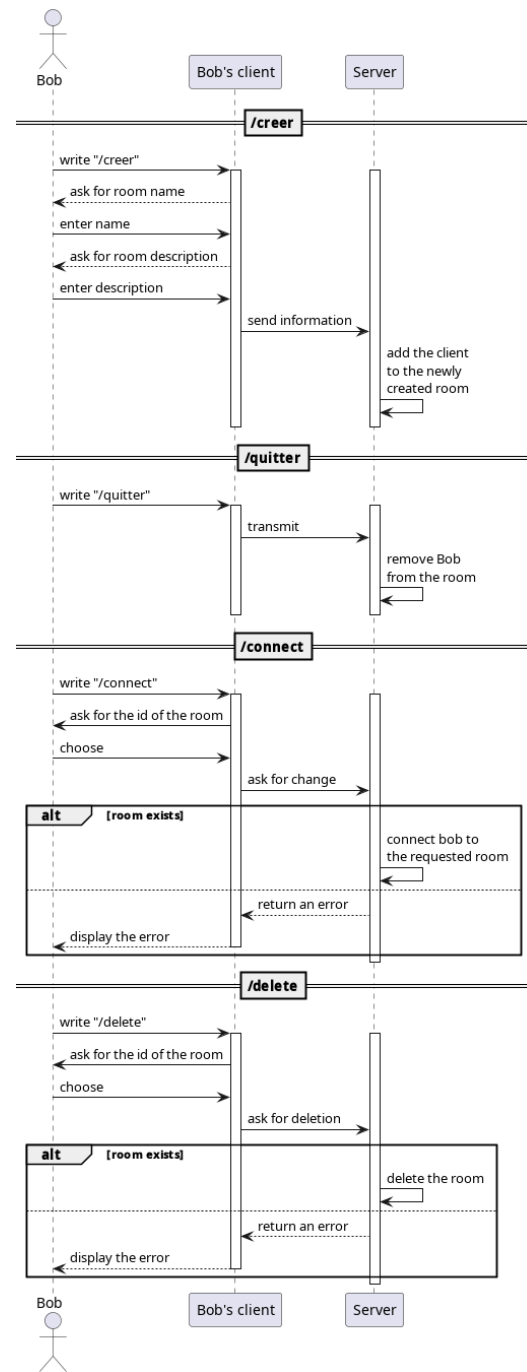


Figure 1: Protocoles



(c) Protocole d'échange de fichiers



(d) Protocole des salons

Figure 1: Protocoles

## 2 Architecture

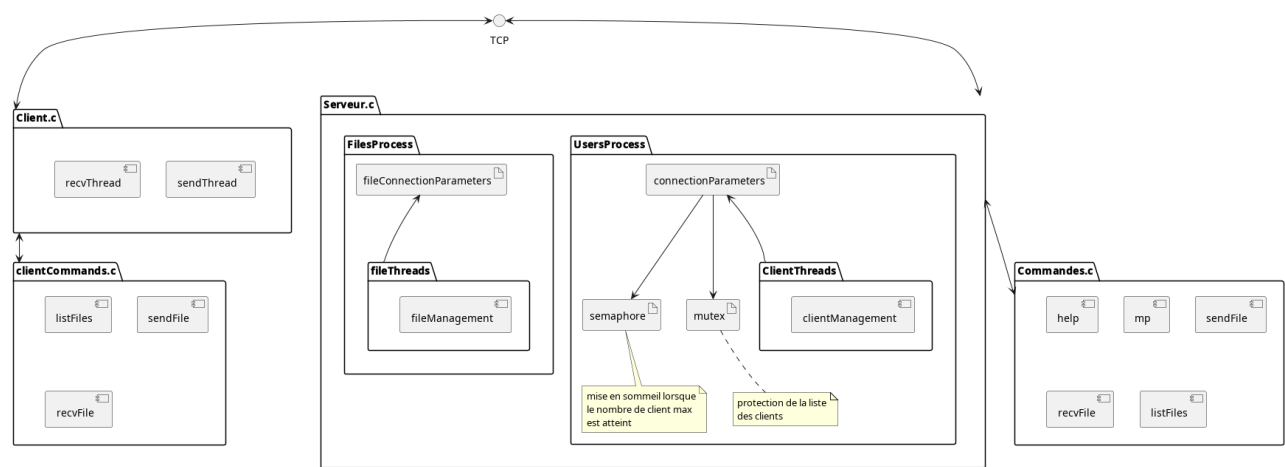


Figure 2: Architecture de la messagerie

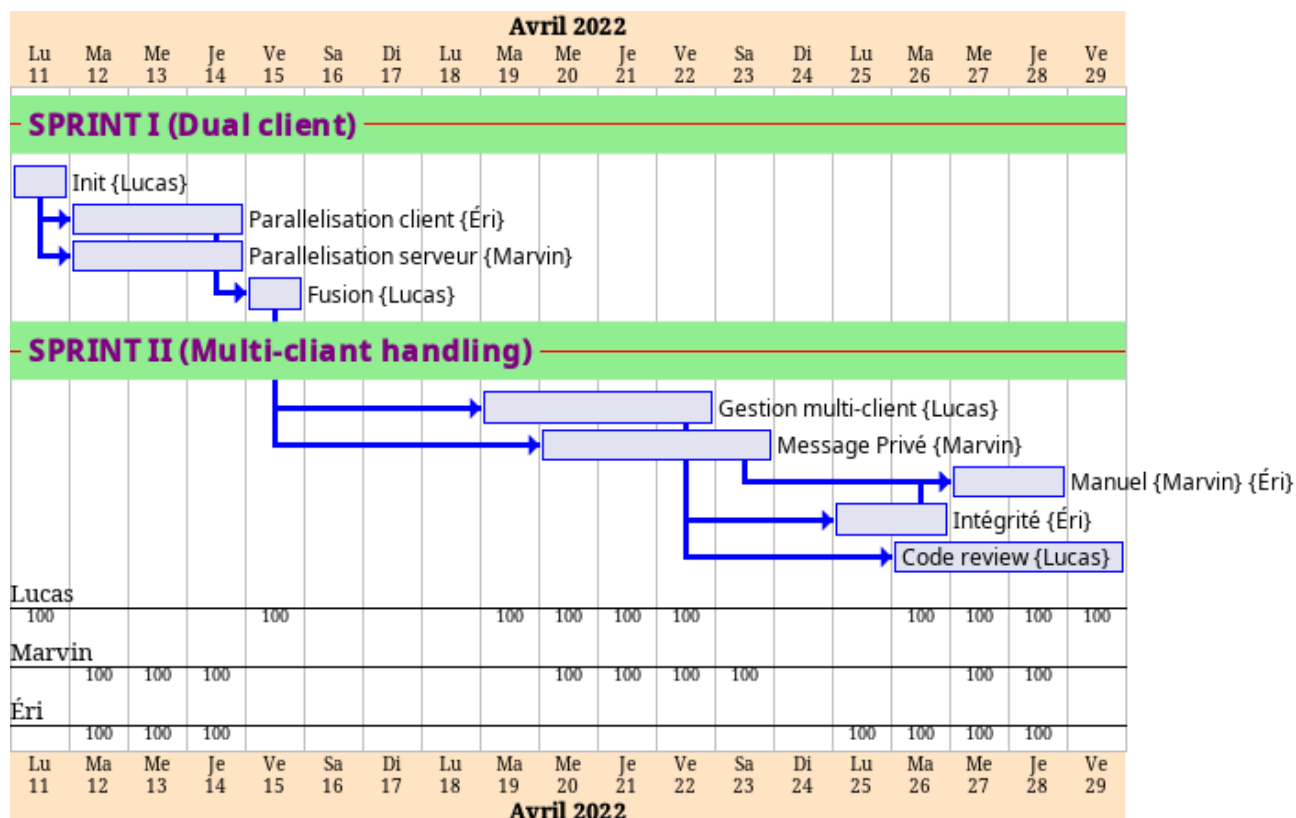
## 3 Répartition du travail

La 4<sup>ème</sup> version de la messagerie consistait à rajouter en priorité une fonctionnalité de salons textuels permettant aux clients de discuter de manière plus privée ailleurs que sur le salon général. Pour cela, nous avons d’abord implémenté une commande permettant de créer des salons et de leur donner un nom et une description, ensuite nous avons codé la fonction permettant d’énumérer les différents salons présents sur le serveur ainsi que la fonctionnalité pour se connecter à un salon et celle permettant de s’en déconnecter. Pour finir, nous avons rajouté la fonction permettant la suppression d’un salon.

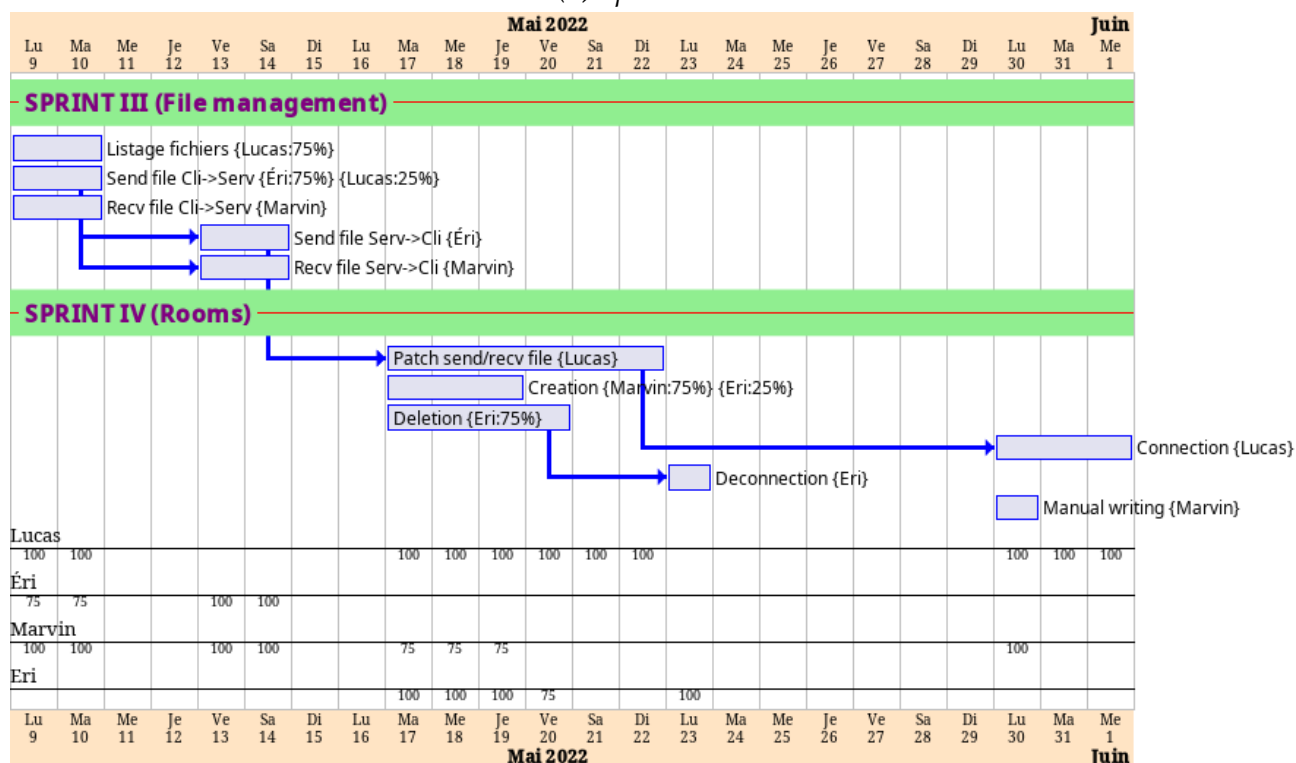
Nous n’avons malheureusement ni réussi à ajouter la possibilité de modifier les noms et descriptions des salons, ni réussi à implémenter la fonctionnalité permettant de restaurer les salons à l’allumage du serveur et nous n’avons pas eu le temps de rajouter des fonctionnalités supplémentaires.

Tâches	Étudiants		
	Lucas	Éri	Marvin
Finalisation fichiers	X		
Création des salons		X	X
Destruction des salons		X	
Connexion aux salons	X		
Deconnexion des salons		X	
Rédaction du guide			X

Table 1: Tâche effectuée par étudiant



(a) Sprint I &amp; II



(b) Sprint III &amp; IV

Figure 3: Diagramme de Gantt sur la réalisation du projet

## 4 Exécution du code

Pour lancer la messagerie, il faut commencer par compiler et lancer le serveur en spécifiant les 2 ports (1: port d'écoute pour les messages, 2: port d'écoute pour les fichiers)

```
[lucas@xps-lucas ~]$ gcc -o server serveur.c commandes.c  
[lucas@xps-lucas ~]$ ./server <PORT1> <PORT2>
```

On peut alors lancer les clients en spécifiant les mêmes ports que pour le serveur

```
[lucas@xps-lucas ~]$ gcc -o client client.c clientCommands.c  
[lucas@xps-lucas ~]$ ./client <IP> <PORT1> <PORT2>
```

## 5 Difficultés

- Les fonctionnalités du sprint précédent n'étaient pas opérationnelles, il y a donc eu une prise de retard sur les fonctions à implémenter lors de ce sprint puisqu'il a fallu terminer les fonctions concernant l'envoi des fichiers
- A la fin, lors de la fusion des travaux, une erreur d'allocation mémoire s'est invitée dans la commande permettant de lister les salons et son origine n'a pas pu être trouvée.
- Pour la création des salons, nous avons eu au départ un problème car on ne pouvait pas mettre d'espace dans le nom ni dans la description du salon, on a finalement réglé ce problème en utilisant un fgets à la place d'un scanf