## Pre release (28/02/2020) :

* Choix du groupe
* Team : Adrien, Matthieu, Maxime, Pauline
* Serveur : Pauline, Adrien
* Client : Maxime, Matthieu
* Identification des user stories (backlog)

## Backlog :

1. L’utilisateur souhaite s’identifier à l’aide d’un login et password au démarrage sur le chat
2. L’utilisateur souhaite engager un échange avec un collègue sur le chat en direct.
3. L’utilisateur souhaite créer un groupe sur le chat avec plusieurs collègues.
4. L’utilisateur souhaite échanger avec ses collègues sur des groupes du chat.
5. L’utilisateur souhaite effectuer une recherche pour trouver un collègue.
6. L’utilisateur souhaite connaitre le statut d’un collègue.
7. L’administrateur souhaite s’identifier à l’aide d’un login et password au démarrage sur le chat
8. L’administrateur souhaite déclarer une liste d’utilisateurs à partir d’une base de données externe.
9. L’administrateur souhaite identifier les utilisateurs en ligne sur le chat.

## A chaque sprint :

* Mettre à jour le diagramme de classe UML et la JavaDoc.
* Sélectionner quelques user stories
* Mettre à jour les cartes CRC
* Tests
* Obtenir une version exécutable du code à déposer sur moodle

Saut de page

**Sprint 1**

**Objectif**: user story 1 et 7

* Comparer avec les TD précédent (banque) et exploiter les analogies.
* Cartes CRC
* Diagramme de classe
* Tests ?
* Adrien, Pauline : protocole coté serveur
* Maxime, Matthieu : protocole coté client

**Objectifs atteints :**

* Une image contenant capture d’écran, avant, moniteur, debout

  Description générée automatiquementDiagramme de classe

Diagramme de classe de fin de sprint 1

**Erreur ! Nom du fichier non spécifié.**

**Sprint 2**

**Objectifs** : user story 2 et 3

**Objectifs atteints :**

A présent, plusieurs utilisateurs peuvent se connecter au serveur.   
Ils communiquent sur la socket, le serveur reçoit le message, et le renvoie sur la socket. Dans un premier temps il renvoie le message, notifiant alors le client que son message est parvenu.

Nous avons un début d’interface graphique

**Diagramme de Classe :**

Généré automatiquement avec IntelliJ.

Une image contenant équipement électronique, ordinateur, circuit

Description générée automatiquement

Exemple de carte CRC :

|  |  |
| --- | --- |
| **Classe :** Protocole Connexion | |
| **Responsabilités :**  Enclenche la connexion d’un client au serveur.  Différencie les cas (User ou Admin) | **Collaborateurs :**  ProtocoleConnexionUsers  ProtocoleConnexionAdmin  ProtocoleChat |