

Rapport phase 1 :

Samuel Nicod, lead développeur dans une équipe de deux personnes.

Le projet consiste à créer une application web de messagerie pour une entreprise avec une gestion de ses utilisateurs. Le cahier des charges précise que les technologies utilisées doivent être Spring Boot et React, le choix de la base de données est laissé au prestataire. L'envoi de message doit être sécurisé par la technologie au choix du prestataire mais doit être justifié.

Rôles et tâches :

-Samuel :

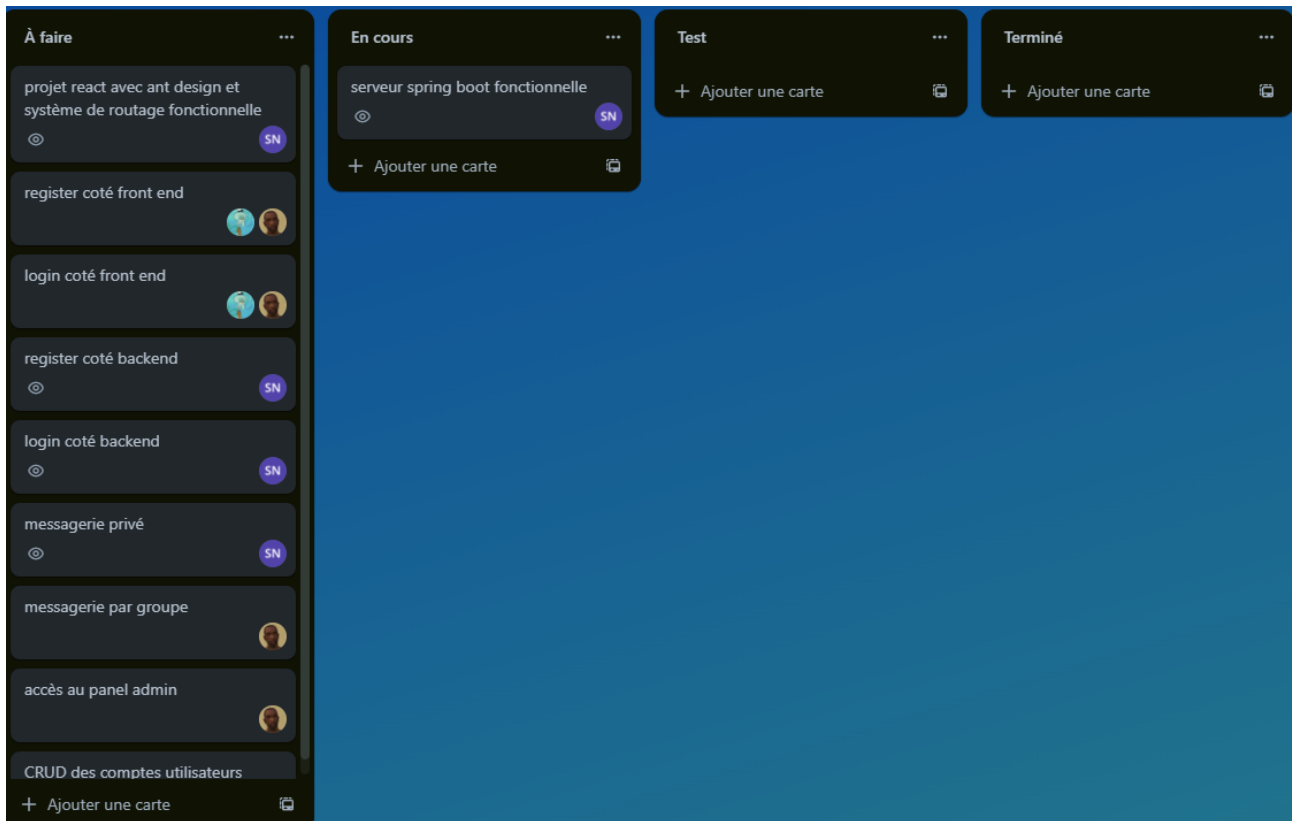
- ::lead développeur
- ::schématisation
- ::installation du projet de départ
- ::développement de l'API REST(Spring Boot).

-Ilyes :

- ::développeur
- ::développeur du CRUD utilisateur.
- ::développeur du front-end(react).

-Mathis :

- ::développeur
- ::développeur de la connexion de la base de données dans l'API REST
- ::développeur du front-end(react).



Les tâches basiques comme l'installation de la base de données n'ont pas été notifié car elle ne prend le temps que d'une commande linux (30 secondes maximum). Le projet doit être cependant bien encadré, configuration de la base de donnée dans le code, installation des routes principales sur le back et le front et création des fonctions de sécurisation grâce au jwt(json web token). C la phase d'installation du projet cité ci-dessus.

Avancement actuel :

Front-End avec react pas encore commencer (pour l'installation basique : 1h),
 Back-End avec spring boot en cours(temps passer dessus : 4h), connexion avec la base de donnée presque fini (temps restant avant de pouvoir commencer à coder l'appli web : 2h).
 DB est déjà prête à recevoir des données.

Schématisations :

stack technique :

- Db = rethinkdb – doc api java : <https://rethinkdb.com/api/java/> (base de données sous format json, plus facile d'utilisation plus rapide à mettre en place et tout aussi efficace que les format de db sql comme mariadb et mongodb)
- Back-End = Spring Boot
- Front-End = react - ant design
- sécurisation = https – (pour la sécurisation d'envoi de message le https est suffisant à lui seul)

DB mld :

Role(

-id : int

-name : text(30 max)

)

User(

-id : uuid

-name : text(30 max)

-firstname : text(30 max)

-phone : text(10 max)

-mail : text(100 max)

-password : longtext(256 max)

-role : idRole

)

Conversation(

-id : int

-idUser : longtext(format : {'uuid', 'uuid', ...etc},)

admin : idUser

)

Message(

-id : int

-message : longtext(300 max)

-conv : idConv

-sender : idUser

-sendingDate : datetime(m/h/d/y)

)