

Groupe 10

Rapport du livrable 2

Numéro de commit : 51f67063

Numéro de build : #85

Benbouchta Younes

Duong Quoc An

Nguyen Ngoc

Queguineur Tanguy

Yi Nghi Ke Man

9/03/2023

Table des matières

1. Rapport de métriques	1
2. Fonctionnalités non remplies	2
3. Diagramme de classes	3
4. Diagramme de séquences	5
5. Complément du livrable 1	8
6. Rapport d'activités	10

1. Rapport de métriques

Violations :

Nombre de violations de PMD	0
Nombre de violations de checkstyle	0
Nombre de violations de CPD	0

Couverture de code des tests unitaires pour la couche business:

UserFactoryImpl	100%
UserImpl	23%
UserUCCImpl	100%

Lien vers le job: <https://jenkins.vinci.be/job/projet-ae-groupe10/85/>

Merge-requests :

	Livrable 1	Livrable 2
Nombre total de merge-requests	0	8

2. Fonctionnalités non remplies

La fonctionnalité “Rafraîchissement” n’a pu qu’être partiellement développée. Pour l’instant, l’application garde bien l’utilisateur connecté lors d’un rafraîchissement mais il n’y a pas de vérification du token dans le backend à chaque rafraîchissement. Il y a eu des problèmes lors de la récupération de la réponse lors de la requête au niveau frontend qui nous a empêché de réussir la mise en place complète de cette fonctionnalité.

Une branche feat-refresh-token comprend la tentative de développement de cette fonctionnalité au niveau backend (création d’une route pour récupérer un user en vérifiant le token).

3. Diagramme de classes

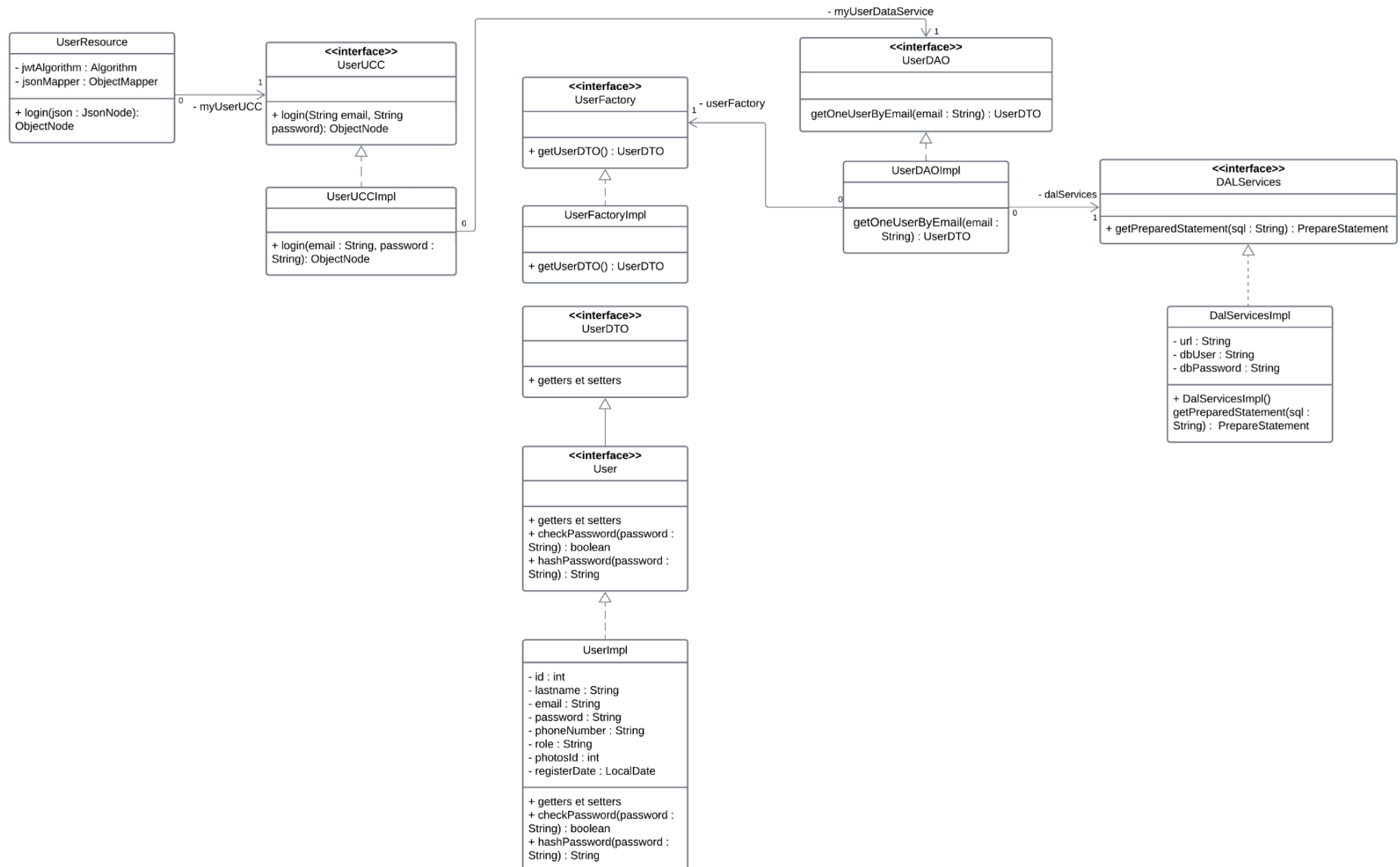
Le diagramme de classe ci-dessous représente l'état actuel de notre architecture en backend.

Détails des différentes classes :

- **UserResource** : Permet de communiquer avec le frontend. Elle contient les méthodes nécessaires à l'envoi ou/et à la réception de données d'un utilisateur.
- **UserUCCImpl** : Possède tous les cas d'utilisations d'un utilisateur. Elle contient l'attribut suivant : userDAO. Le cas d'utilisation "S'identifier" est représenté par la méthode login(String email, String password). (Elle implémente l'interface **UserUCC**)
- **UserFactoryImpl** : permet de fournir un objet de type "UserDTO". (Elle implémente l'interface **UserFactory**)
- **UserDTO** : Contient les getter et setter de la classe "UserImpl". Elle permet de filtrer les opérations importantes.
- **UserImpl** : Permet de représenter un utilisateur. (Elle implémente l'interface **User**)
- **UserDAOImpl** : Possède toutes les opérations à effectuer d'un utilisateur vers la base de données. (Elle implémente l'interface **UserDAO**)
- **DalServicesImpl** : Permet de communiquer avec la base de données. (Elle implémente l'interface **DalServices**)

Activer la compatibilité du lecteur d'écran

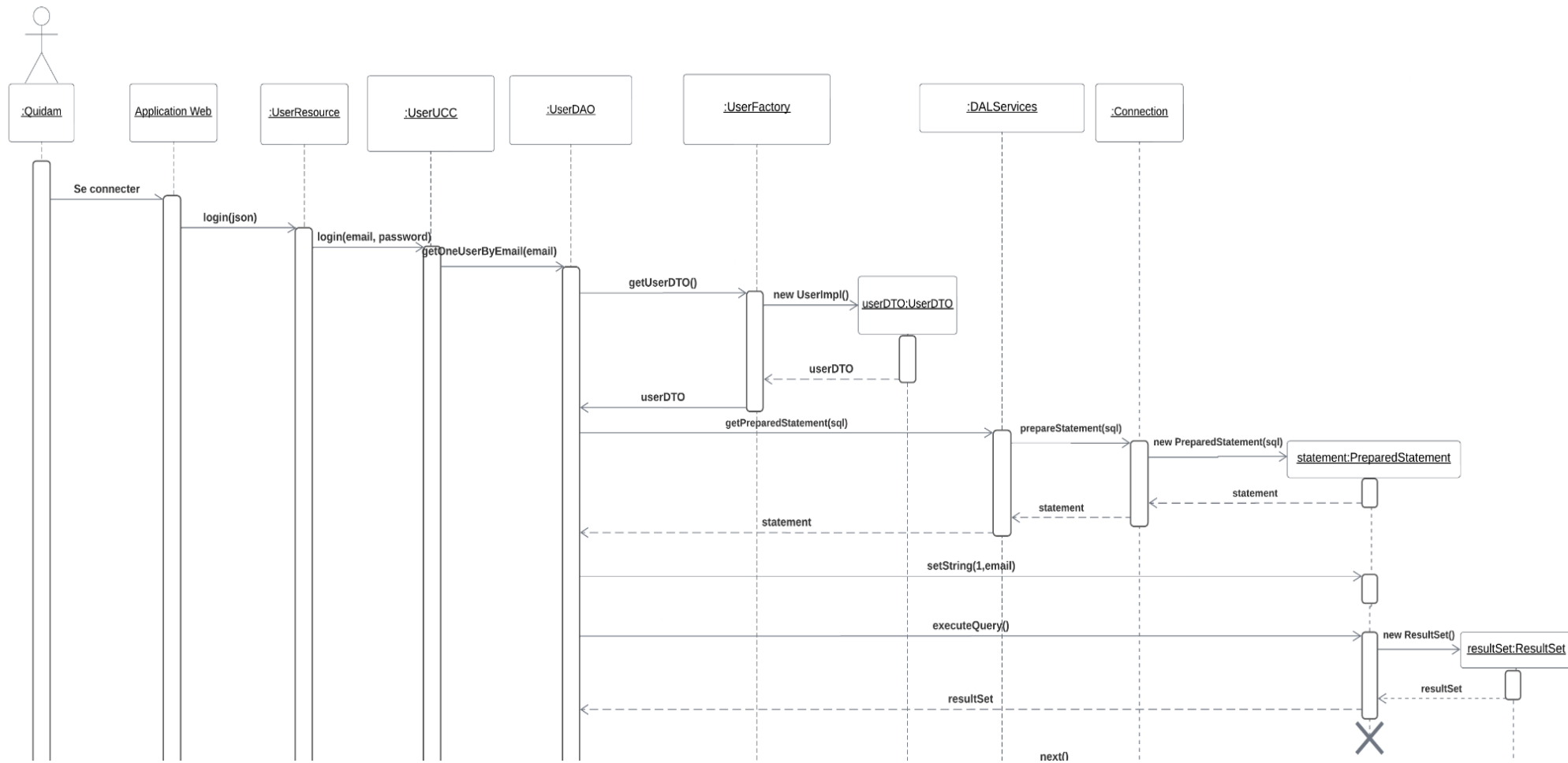
Pour activer la compatibilité avec le lecteur d'écran, appuyez sur Ctrl+Alt+Z Pour connaître les raccourcis clavier, appuyez sur Ctrl+barre oblique



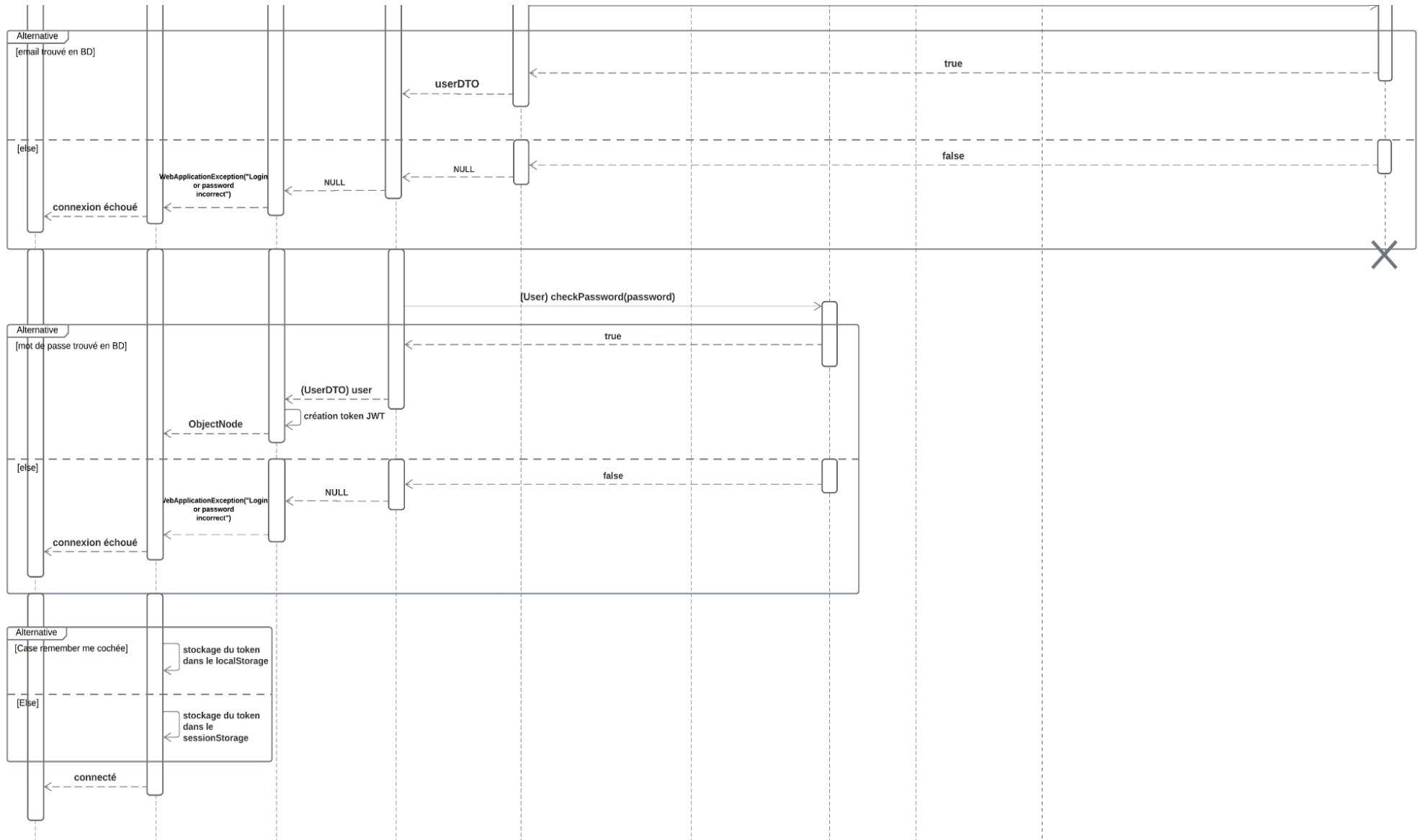
4. Diagramme de séquences

Le diagramme de séquences représenté à la page suivante explique le cas d'utilisation "S'identifier" pour un quidam, avec tout le flux depuis le front-end jusqu'à la DB.

Pour activer la compatibilité avec le lecteur d'écran, appuyez sur Ctrl+Alt+Z Pour connaître les raccourcis clavier, appuyez sur Ctrl+barre oblique



Pour activer la compatibilité avec le lecteur d'écran, appuyez sur Ctrl+Alt+Z Pour connaître les raccourcis clavier, appuyez sur Ctrl+barre oblique

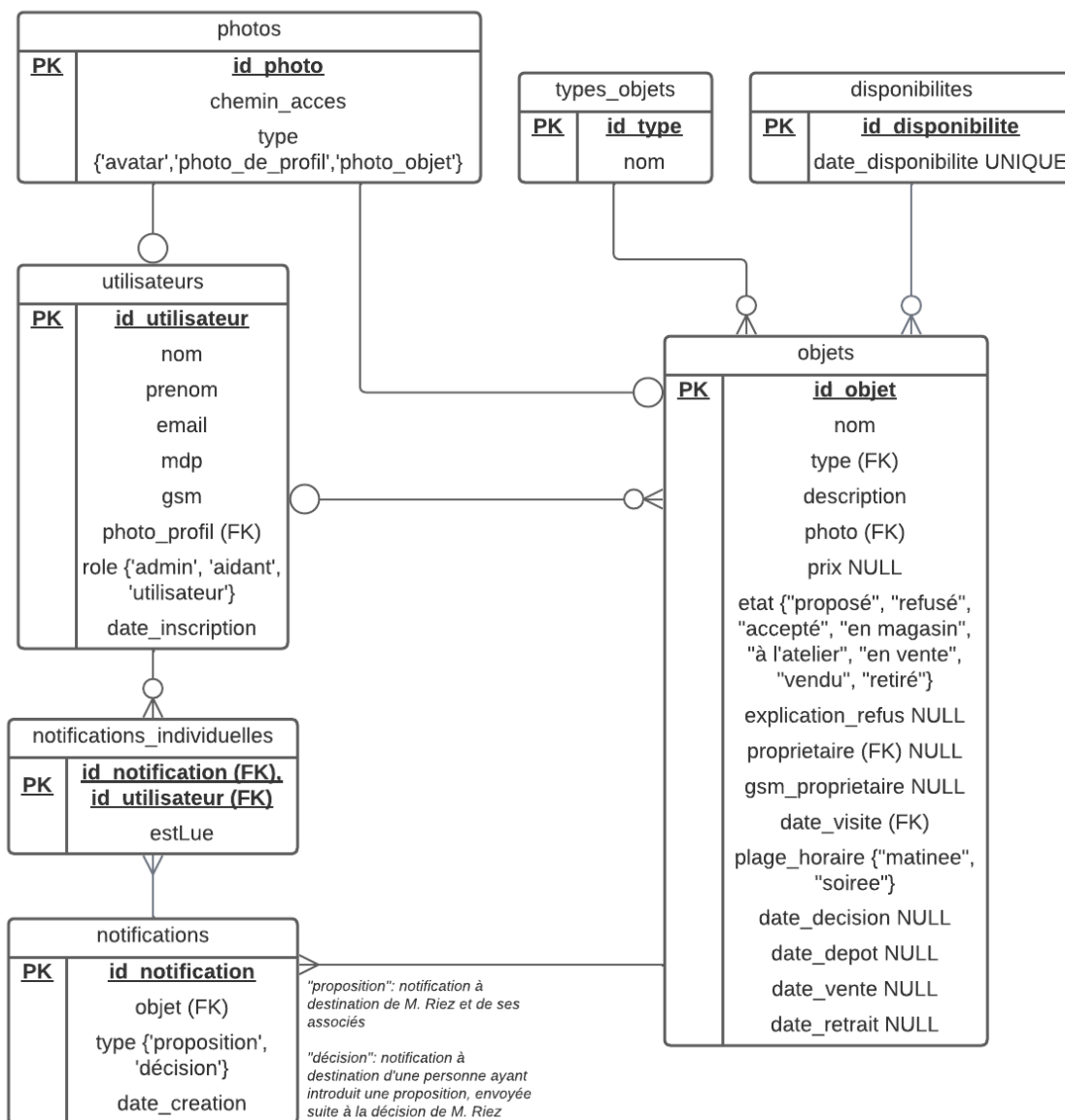


5. Complément du livrable 1

Nous avons apporté de multiples changements au DSD (Diagramme de Structure de Données) afin de faciliter le développement et la maintenance de notre base de données et d'optimiser la communication entre l'application Turbotron 2000 et celle-ci. Voici les changements opérés sur notre DSD :

Pour activer la compatibilité avec le lecteur d'écran, appuyez sur Ctrl+Alt+Z Pour connaître les raccourcis clavier, appuyez sur Ctrl+barre oblique

Tous les champs sont considérés NN,
sauf si le contraire est explicité (NULL)



Pour activer la compatibilité avec le lecteur d'écran, appuyez sur Ctrl+Alt+Z Pour connaître les raccourcis clavier, appuyez sur Ctrl+barre oblique

Changements généraux :

- modification de la relation entre la table "*photos*" et "*utilisateurs*"
- ajout de la table "*notifications_individuelles*"

Changements au niveau des tables :

- Table *utilisateurs* :
 - champs "*estAidant*" et "*estResponsable*" remplacés par le champ "*role*"
- Table *objets* :
 - champ "*explication_refus*" ajouté, transféré de la table "*notifications*"
- Table *notifications* :
 - champ "*utilisateur*" supprimé
 - champ "*explication_refus*" transféré à la table "*objets*"
 - champ "*estLue*" transféré à la table "*notifications_individuelles*"

- Table *notifications_individuelles* :

Cette table permet l'envoi d'une même notification à plusieurs personnes et permet d'enregistrer le fait qu'une personne ait lu la notification ou non. Nous utilisons pour cela, en clé primaire, les identifiants de la notification en question et de la personne concernée (champs "*id_notification*" et "*id_utilisateur*"). Nous gardons aussi en mémoire le fait que la notification ait été lue ou non par l'utilisateur (champ "*estLue*").

6. Rapport d'activités

Pour la réalisation de ce deuxième livrable, notre groupe s'est réparti les tâches de la façon suivante:

- La création du script SQL permettant la création de la base de données a été faite par **Younes**.
- Au niveau du frontend, ce sont **Ke Man** et **Quoc An** qui se sont occupés de créer la Single Page Application contenant la page de connexion et la barre de navigation.
- L'architecture du backend et les tests unitaires ont été réalisés par **Ngoc** et **Tanguy**.
- Une fois l'architecture de l'application complétée, les diagrammes de classes et de séquences ont pu être réalisés et ce sont **Younes**, **Quoc An** et **Ke Man** qui se sont occupés de cela.
- Pour finir, le groupe s'est réuni pour rédiger ce rapport.

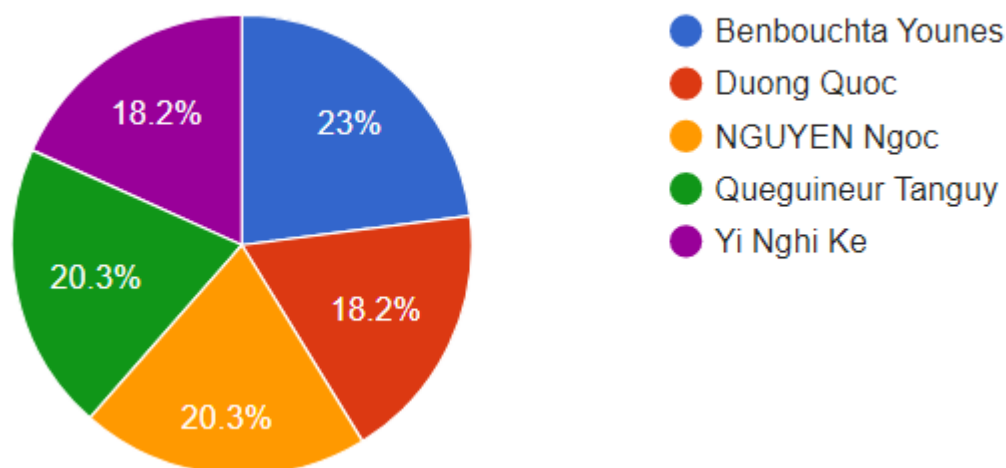
Dans les choses positives, le groupe fonctionne bien et les tâches sont réparties équitablement. Par contre, il faudra faire attention dans le futur à ce que tout le monde travaille sur les différentes parties du projet car ce n'est pas le cas actuellement.

Quant aux nombres d'heures prestées par chacun, les voici :

- **Younes** : 16h50
- **Ngoc** : 13h
- **Ke Man** : 15H30
- **Quoc An** : 16h40
- **Tanguy** : 10h

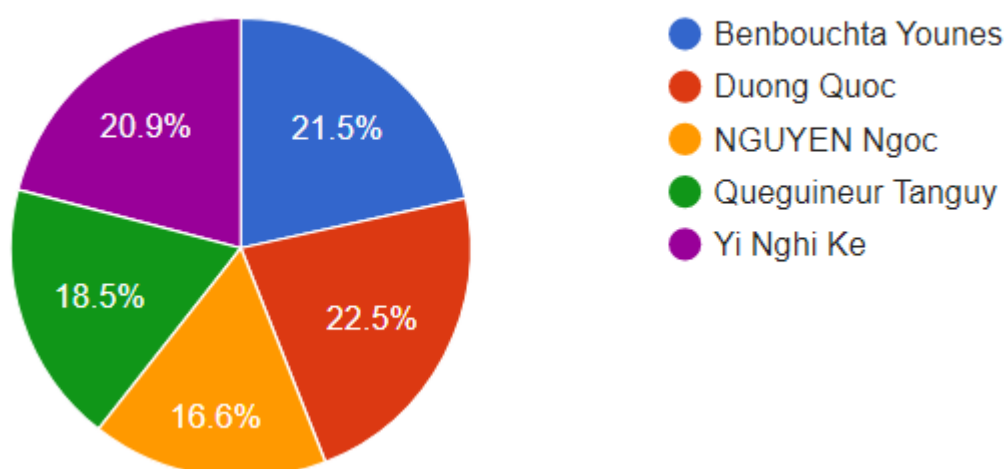
Temps pour le cours 3 :

Temps de cette itération de cours (en heures)

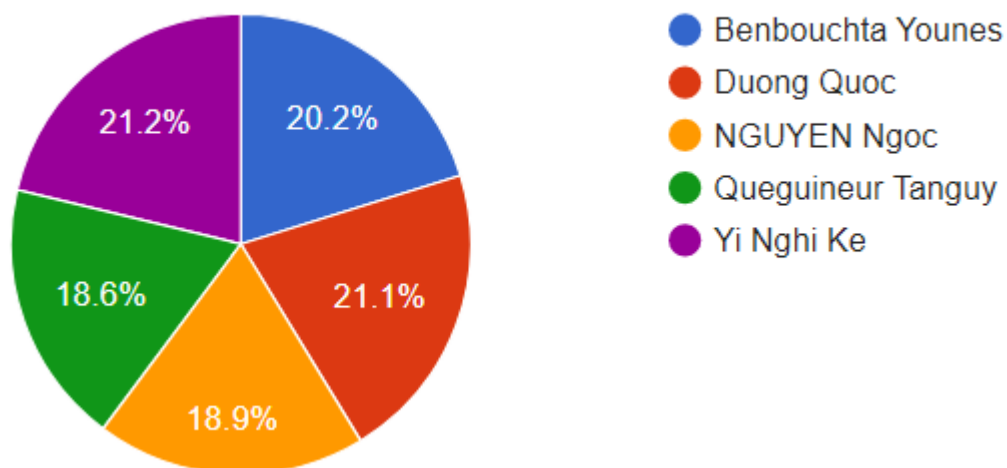


Temps pour le cours 4 :

Temps de cette itération de cours (en heures)



Temps total complet (en heures)



[Activer la compatibilité du lecteur d'écran](#)

Pour activer la compatibilité avec le lecteur d'écran, appuyez sur Ctrl+Alt+Z Pour connaître les raccourcis clavier, appuyez sur Ctrl+barre oblique

Ceci est la dernière page du rapport