|  |  |
| --- | --- |
| **Livrable** | **4** |
| **Sigle du cours** | **SEG2505** |
| **Professeur** | **Aziz Oukaira** |
| **Assistant (e)**  **Groupe** | **Gregory Price**  **G12** |

1. G12 :
2. William Mallette
3. Andy How Hok Hium
4. Chenliwei Zhou
5. Javisen Katherasa Pillay
6. Penghao Yuan

Table des matières:

[1](#_Toc152753066)

[Introduction: 3](#_Toc152753067)

[Diagrammes des classes UML: 4](#_Toc152753068)

[Leçons apprises: 5](#_Toc152753069)

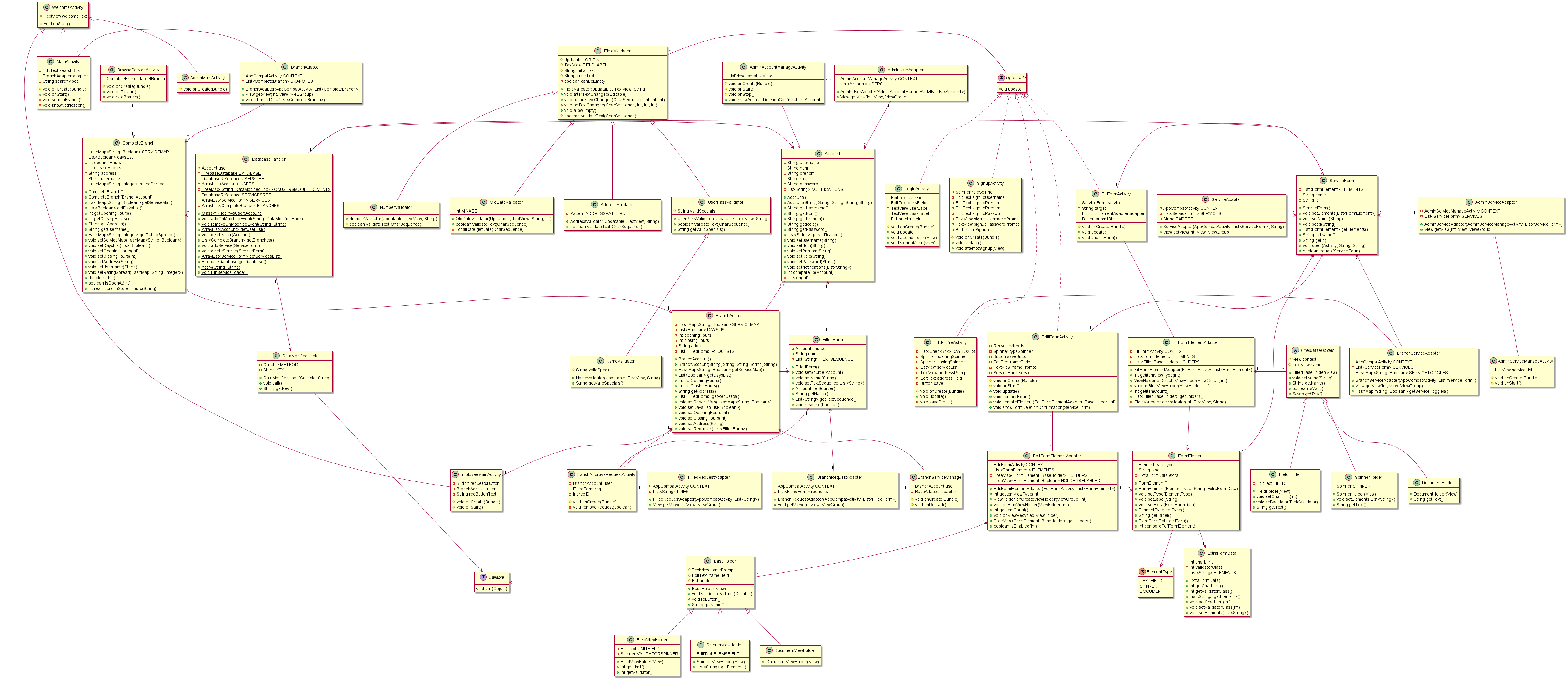
[Rôles de chacun des membres de l'équipe pour chaque livrable: 6](#_Toc152753070)

[Captures d'écrans de notre application: 7](#_Toc152753071)

# Introduction:

Nous avons développé une application appelé Service Novigrad qui permet à un utilisateur de créer un compte en tant que client ou employé d’une succursale. Les administrateurs peuvent ajouter, modifier ou enlever les services qui seront offert par les succursales de Service Novigrad et supprimer des comptes de client/succursale. De plus, les employés de la succursale peuvent modifier le profil de leur succursale (adresse, heures de travail) et de sélectionner quels services sont offerts par la succursale. Ils peuvent aussi visionner les demandes de service soumis par les clients, et soit approuver ou rejeter ces demandes. Finalement, nous avons implémenté les fonctionnalités relatives au client. C’est-à-dire, l'habilité du client de rechercher une succursale par rapport à l’adresse, les heures de travail, ou par les services offerts. Ensuite, ils peuvent visionner tous les services offerts par la succursale et faire une demande conformant aux exigences mises en place par l’administrateur. Lorsqu’une demande a été soumis, le client sera demandé d’évaluer leur expérience avec la succursale, et la prochaine fois qu’il se connecte au service, il sera notifié si leur demande a été acceptée ou rejetée

# Diagrammes des classes UML (un PDF de plus haute qualité est disponible avec ce document):



# Leçons apprises:

Nous avons appris plein de choses avec ce projet. Nous avons appris comme comment faire des diagrammes UML de façon efficace, ou faires des tests unitaires pour tester les fonctionnalités de nos méthodes. Mais le plus Important est que nous avons appris à utiliser Android Studio et nous nous sommes familiarisés avec des bases de données asynchrones tel que Firebase pendant ce semestre, qui sont tous les deux des outils qui pourrait être essentiel dans le futur dans nos carrières académiques et professionnelles.

# 

# Rôles de chacun des membres de l'équipe pour chaque livrable:

|  |  |  |
| --- | --- | --- |
|  | Livrable 1 | Livrable 2 |
| Nom | Contribution | Contribution |
| William Mallette | L’ajout du compatibilité Firebase Implémentation des activités et des systèmes de validation | Capacité de supprimer des comptes  Capacité de créer et modifier des services globaux |
| Andy How Hok Hium |  | Test unitaire |
| Chenliwei Zhou |  |  |
| Javisen Katherasa Pillay |  |  |
| Penghao Yuan | UML diagram, GitHub equipe, code implementation | UML diagram |

|  |  |  |
| --- | --- | --- |
|  | Livrable 3 | Livrable 4 |
| Nom | Contribution | Contribution |
| William Mallette | Capacité de modifier le profil de la succursale  Capacité d’approuver ou rejeter des demandes de service Introduction du PlantUML pour faciliter la création des diagrammes | Création et optimisation de CompleteBranch pour faciliter la recherche des succursales  Capacité de remplir les demandes de service Perfection des algorithmes de recherche Édition de la présentation et du rapport |
| Andy How Hok Hium |  | Redaction rapport |
| Chenliwei Zhou |  |  |
| Javisen Katherasa Pillay |  |  |
| Penghao Yuan | UML diagram | UML diagram, Capacité de remplir les informations et documents requis et soumettre la demande de service |







# Captures d'écrans de notre application:



