

Livrable 4 : Rapport Final

Service à la demande d'un PC sur mesure

SEG 2505 : INTRODUCTION AU GÉNIE LOGICIEL

Présenté au professeur Laurent Frerebeau,

<i>NOM</i>	<i>Prénom</i>	<i>Identifiant GitHub</i>
YAHYAOUI	Khalil	khalil-yahyaoui-015
SAHLI	Yassine	YassineSahli04
KHELIFI	Yassine	Yassine-Khelifi
OUAJJOU	Othmane	oth-code1
NAGIFI	Othman	NagifiOthman

Table des matières

1.La référence de notre projet GitHub.....	
2.Description de nos choix de conception.....	
3.Description des exigences supplémentaires.....	
4.Retour de notre expérience.....	
5.Tableau de synthèses des contributions.....	
6.Estimation du temps passé sur chaque livrable du projet.....	

1. Référence de notre projet Github:

Lien URL : <https://github.com/uOttawa-2024-2025-seg2505-projet/groupe-35.git>

Au sein de ce projet , nous nous sommes basés sur des les éléments requis dans les livrables (1→4) :

Le premier livrable nous a permis de créer un environnement de développement convenable en alignant organisation et exigences à implémenter ,un groupe de projet a été créer dans github ,la première version du modèle contient les diagrammes d'activité, de classe, de séquence et d'état ainsi que tous les objets et attributs primaires , de plus tous les 4 utilisateurs sont créés et peuvent se connecter et voient les fonctionnalités qui les concernent sans pouvoir les utiliser, ils peuvent aussi se déconnecter .L'administrateur quant à lui peut créer plusieurs utilisateurs .Suite à ce livrable , tous les champs ont été validés , des messages d'erreurs spécifiques à des erreurs logiques s'affichent , de même nous avons pu utiliser une base de donnée .Nous avons donc pu clôturer ce livrable , et tous les membres de l'équipe ont effectué un commit .

Lors de notre 2ème livrable , nous nous sommes concentrés sur l'implémentation de la 2ème version de notre modèle s'est orienté vers la conception des des fonctionnalités du StoreKeeper incluant la gestion du stock (ajout / suppression / visualisation / modification d'item, ainsi que la liste de tous les e le ments du stock) , ainsi que des fonctionnalités supplémentaires de l 'administrateur (suppression et importation de données) .Suite à notre travail , tous les champs ont été validés des messages d'erreur appropriés s'affichent en cas de saisie incorrecte.Notre modèle a donc été valide d'un point de vue représentation et cohésion .

Le 3ème livrable nous a servis à compléter les fonctionnalités du rôle Requester , les autres aspects de la gestion du stock en été ajouté , (i.e. ajout / suppression / visualisation) , en ajout ,des tests unitaires de niveau Api ont été mis en place pour valider la gestion des commandes ainsi que des tests d'interface utilisateurs basés sur Espresso. Suite à cela, tous les champs sont validés. Des messages d'erreur appropriés s'affichent en cas de saisie incorrecte ou d'erreur de logique .

Le 4ème livrable nous a permis de finaliser la dernière version du modèle qui contient toutes les fonctionnalités du traitement de l'assemblage (visualisation, mise en attente, rejet, acceptation) .

A travers ce dernier livrable , nous avons pu finaliser notre modèle et affiner les dernières imperfections afin de produire un modèle conforme à toutes les exigences demandés .

2.Description de nos choix de conception

Dans le projet, le groupe a fait des choix qui ont permis de simplifier l'utilisation de l'application tout en facilitant sa mise à jour. Tout d'abord, la hiérarchie et l'architecture de l'application ont été soigneusement organisées, avec des classes dédiées aux activités principales (comme StorekeeperActivity et AdminActivity) ainsi qu'à des fonctionnalités spécifiques (comme SoftwareComponents, HardwareComponents et Orders). De plus, des interfaces (comme ItemExistenceCallback et OnCheckDeletableCallback) ont été utilisées pour définir

[Retour à la table des matières](#)

des contrats permettant de gérer des opérations asynchrones. Enfin, des sous-packages (comme `androidTest` et `test`) ont été intégrés pour vérifier les fonctionnalités et effectuer des démonstrations de l'application à l'aide de tests instrumentés et unitaires.

Tout en priorisant l'expérience utilisateur, notre projet propose une interface intuitive et facile à utiliser. L'application adopte un design uniforme pour tous les rôles utilisateurs, avec des méthodes claires pour ajouter des éléments, des pages principales bien organisées, et des boutons de retour pour une navigation simplifiée.

Le groupe a également intégré des fonctionnalités pour améliorer la précision et l'accès aux données, comme des listes tabulaires affichant clairement les informations sur les commandes et les composants. Ces listes incluent des outils pratiques, tels que les dates et heures d'ajout, pour faciliter la vérification et l'organisation.

Pour la partie backend, le groupe a choisi d'utiliser Firebase comme base de données pour assurer un suivi efficace et centralisé des données de l'application. En intégrant Firebase avec le programme, le groupe a fait des approches dynamiques pour gérer les interactions avec la base de données, notamment l'utilisation de méthodes asynchrones qui aident à continuer de fonctionner sans être bloqué par la base de données.

Cette approche permet au programme de continuer à fonctionner sans être bloqué par les opérations de la base de données, grâce à l'utilisation de listeners qui surveillent les modifications ou récupèrent les données en arrière-plan. améliore L'exécution et l'expérience de l'utilisateur. De plus, le groupe a créé des interfaces pour structurer ces interactions, définissant des contrats pour organiser les méthodes de rappel (callbacks). Ces dernières s'exécutent automatiquement lorsque Firebase retourne un résultat, par exemple pour confirmer l'existence d'un élément ou vérifier sa possibilité de suppression, renforçant ainsi la maintenabilité du programme.

Le groupe a également utilisé des spinners, essentiels pour gérer les données de manière interactive. Ils permettent aux utilisateurs de sélectionner facilement des types de composants, comme dans l'ajout ou la modification d'éléments. Connectés à Firebase, ces spinners se mettent à jour dynamiquement pour refléter les données disponibles. Par exemple, la sélection d'un type actualise automatiquement les sous-types correspondants. Cette approche simplifie la navigation, réduit les erreurs et améliore l'expérience utilisateur.

Pour l'affichage des textes, le groupe a utilisé des TextViews, combinés avec des contrôles de visibilité pour afficher ou masquer des informations en fonction des actions de l'utilisateur. Par exemple, les messages d'erreur ou de confirmation

[*Retour à la table des matières*](#)

n'apparaissent que lorsque c'est nécessaire, ce qui a contribué à l'optimisation du code.

En conclusion, de nombreuses autres méthodes ont été appliquées sans être détaillées dans ce document. Ce projet combine des techniques déjà maîtrisées à de nouvelles approches découvertes au cours du développement, dans le but d'assurer l'utilité et l'efficacité de l'application.

3.Description des exigences supplémentaires

Dans le cadre de ce projet, plusieurs exigences supplémentaires ont été envisagées pour répondre à des besoins non couverts initialement. Ces propositions visent à enrichir les fonctionnalités de l'application et à améliorer l'expérience utilisateur, tout en augmentant la performance et l'efficacité du système. Les exigences proposées ont été analysées en fonction de leur pertinence, faisabilité technique, et impact global sur le projet.

Liste des exigences supplémentaires:

Exigence 1 : Recherche avancée des utilisateurs

- **Description** : Permettre à un administrateur de rechercher des utilisateurs dans la base de données à l'aide de critères comme le nom, l'email ou le rôle.
- **Statut : Traité** – Cette fonctionnalité a été mise en œuvre pour améliorer la gestion des utilisateurs, en particulier dans des bases de données volumineuses.

Exigence 2 : Historique des modifications

- **Description** : Créer une fonctionnalité permettant d'enregistrer et d'afficher l'historique des modifications apportées à la base de données (ajouts, suppressions, modifications).
- **Statut : Écarté** – Cette exigence a été rejetée en raison des contraintes de temps et des priorités sur d'autres fonctionnalités.

Exigence 3 : Notifications en temps réel

- **Description** : Mettre en place un système de notifications pour informer les utilisateurs des mises à jour importantes, telles que l'approbation d'une proposition ou la soumission d'une contribution.
- **Statut : Écarté** – La complexité technique de cette fonctionnalité et les délais limités n'ont pas permis son intégration.

Exigence 4 : Tableau de bord analytique pour les administrateurs

- **Description** : Offrir à l'administrateur un tableau de bord présentant des statistiques sur les utilisateurs et les performances globales du système.
- **Statut : Écarté** – Cette exigence a été écartée en raison des ressources limitées et de la priorité donnée à des fonctionnalités plus essentielles.

Raisons de traitement ou d'écartement:

- **Recherche avancée des utilisateurs :**
Cette exigence a été traitée car elle offrait une valeur ajoutée importante en rendant la gestion des utilisateurs plus efficace et flexible. Sa mise en œuvre s'est appuyée sur des requêtes optimisées pour garantir une recherche rapide et précise.
- **Historique des modifications :**
Bien que pertinente pour assurer la traçabilité des changements, cette fonctionnalité a été écartée à cause des délais limités et de sa complexité d'intégration dans le système existant. Elle reste une piste intéressante pour une future version.
- **Notifications en temps réel :**
Cette exigence, bien que attractive pour améliorer l'expérience utilisateur, a été écartée en raison de la complexité technique associée à la mise en place d'un système de notifications et des contraintes de temps imposées par le projet.
- **Tableau de bord analytique pour les administrateurs :**
Malgré son utilité potentielle pour l'administrateur, cette fonctionnalité a été jugée non prioritaire par rapport aux exigences essentielles et n'a pas été implémentée à cause des ressources limitées.

Les exigences supplémentaires traitées, comme la recherche avancée des utilisateurs, ont permis d'améliorer significativement la gestion et l'expérience utilisateur. Les exigences écartées, telles que l'historique des modifications ou les notifications en temps réel, restent intéressantes pour de futures versions du projet. Le choix des priorités a permis de respecter les délais impartis et de se concentrer sur les fonctionnalités critiques, tout en laissant des pistes d'évolution possibles pour le système.

4.Retour de notre expérience

- Comment vous êtes-vous organisés en équipe ?
 - Dès le début du projet, notre collaboration a été facilitée par notre connaissance mutuelle, ce qui nous a permis d'éviter des discussions prolongées sur la répartition des rôles. Khalil a naturellement pris le rôle de responsable d'équipe. Pour le premier livrable, nous avons opté pour des séances en groupe afin de bien comprendre les attentes et de nous assurer que chacun maîtrisait les enjeux du projet. Les tâches ont été réparties, et nous avons débuté le travail une semaine avant la date de remise. Cependant, des imprévus de dernière minute, survenus la veille de l'échéance, nous ont forcés à résoudre des problèmes dans l'urgence. Ce retard a engendré une soumission tardive et des erreurs, révélant les limites de notre méthode. Durant la semaine de lecture, nous avons organisé une réunion pour analyser nos lacunes, corriger les erreurs du premier livrable et adopter une approche plus méthodique. En structurant mieux nos tâches et en

effectuant des tests systématiques après chaque étape grâce à la base de données, nous avons constaté une nette amélioration. Si le livrable 2 n'a pas été finalisé dans les temps, il a marqué un tournant. Les livrables 3 et 4, eux, ont été complétés dans les délais et avec succès, grâce à une organisation optimisée et une rigueur accrue.

- Quelles difficultés avez-vous rencontrées et comment vous les avez résolues:
 - Tout au long de notre projet , nous avons rencontrés différents problèmes relié à différents aspects .Etant donné que nous avons effectué un travail d'équipe , nous avons dû nous adaptés à nos emplois respectifs , cela a été au premier abord un défi auquel nous avons fait face , en effet trouvé un créneau horaire ou chacun de nous est disponible afin de discuter du projet et de proposer nos idées d'implémentation et de conception n'as pas été une mince affaire ,nous nous sommes donc adapté à cela et nous avons abordé par suite une stratégie de travail différente et plus adapté à notre cas .Suite à cela nous avons décidé de répartir nos tâches en subdivisant l'équipe en sous groupe de tel façon que chaque sous groupe à un emploi de temps assez similaire , cela nous a donc permis d'être plus performants et à améliorer nôtres productivité .
 - De plus au début , nous avons eu du mal à prévoir la durée nécessaire afin de compléter un livrable ce qui nous a poussé à nous retrouver dans des conditions de travail sous pression où on devait finir notre livrable avant le temps impartis .Notre expérience nous a permis au fur et à mesure des livrables d'avoir une meilleur gestion du temps , ce qui nous a permis de travailler dans de meilleurs conditions .
 - En ajout avec cela , la répartition équitables des tâches nous a causé du tort , en premier lieu lors du livrable 1 vu qu'aucun membre de l'équipe n'était familier avec l' application et les concepts abordés , il était particulièrement dur d'évaluer la difficulté de chaque tache afin d'avoir une répartition juste et équitable , ceci dit cela n'as pas duré longtemps car nous nous sommes vite familiarisé avec les différents concepts et avons donc résolu ce soucis de gestion .
 - Il est important de mentionner que toutes ces difficultés nous ont permis d'acquérir une expérience précieuse qui nous servira lors de nos prochains projets aussi bien sur le plan technique que sur le plan stratégique .Nous avons appris la valeur de la communication au sein d'un groupe ainsi que l'importance de la bonne gestion du temps et sa répercussion sur la productivité.

5. Tableau de synthèses des contributions

NOM	Prénom	n°étudiant	Description des travaux réalisés et des relations entretenues avec l'équipe pour contribuer au projet
YAHYAOU I	Khalil	300390610	<p>D'emblée, je tiens à préciser que j'ai assumé le rôle de responsable de cette équipe. À ce titre, j'étais chargé de vérifier la contribution de chacun de mes équipiers avant la remise de chaque livrable sur la plateforme GitHub. J'étais aussi chargé d'envoyer chaque version du projet, en m'assurant de son bon fonctionnement tout en respectant les délais.</p> <p>Par ailleurs, une complicité fut omniprésente lors des travaux de groupe, car je connaissais les membres de mon équipe avant même notre inscription au cours SEG2505, ce qui nous a permis d'éviter les problèmes de communication pouvant ralentir notre avancement sur chaque livrable.</p> <p>En ce qui concerne ma contribution au projet, j'ai systématiquement rempli les tâches qui m'étaient assignées. Pour les premiers livrables, je me suis principalement concentré sur la mise en forme de l'application. Mon objectif était de garantir un fonctionnement optimal dès son démarrage, en veillant à ce que les boutons soient bien positionnés et que l'ensemble de l'interface soit fluide, sans collisions ni dysfonctionnements. Une fois ces tâches accomplies, j'aidais mes équipiers lorsqu'ils rencontraient des difficultés avec certaines parties du code Java. Ensemble, nous trouvions des solutions plus rapidement et obtenions de meilleurs résultats.</p> <p>Pour les fonctionnalités de StoreKeeper, je me suis occupé de méthodes telles que la méthode LoadStockFromJson. Pour les fonctionnalités de Requester, j'ai aidé camarade Yassine Sahli à comprendre et à implémenter les tests. Et pour les fonctionnalités de Assembler, je me suis occupé de la mise en forme de l'application, qui est la dernière version de l'application.</p>

			<p>À la fin de chaque livrable, après avoir intégré notre travail dans la branche "main", je m'occupais de mettre à jour le fichier .gitignore afin d'inclure le fichier APK de démonstration. Je complétais également un fichier README détaillant plusieurs éléments essentiels : une introduction au projet, la modélisation des diagrammes UML (classes, activités, séquences, états), les éléments de conception, les tests unitaires, un scénario de démonstration suggéré, les identifiants et mots de passe, ainsi que les éventuelles limitations du projet. En complément, j'ajoutais un document PDF exposant les versions successives de notre modèle conceptuel pour chaque livrable. Enfin, je créais un "tag" sur le point de commit associé à chaque livraison, garantissant ainsi une traçabilité claire des contributions.</p> <p>En résumé, mon rôle au sein de cette équipe a consisté à superviser les travaux, assurer la qualité et la conformité des livrables, tout en apportant mon expertise technique pour résoudre les problèmes rencontrés en cours de projet.</p>
SAHLI	Yassine	300 383 586	<p>Dans ce projet, j'ai contribué à plusieurs aspects, en alternant entre la coordination, le développement et parfois le test, tout en travaillant de manière efficace en équipe pour assurer la bonne fonctionnalité de toutes les méthodes.</p> <ol style="list-style-type: none"> 1. Organisation des réunions hebdomadaires : J'ai eu la chance d'assurer la coordination des réunions d'équipe où nous avons discuté de l'avancement ainsi que des difficultés de chacun afin de pouvoir ajuster la répartition des tâches. Ces réunions ont permis de booster la collaboration et l'entraide entre les membres de l'équipe. De plus, ces réunions ont offert un espace de partage d'idées et de résolution collective de problèmes. 2. Conception UML : J'ai participé activement à la conception du modèle de classes UML en

			<p>collaboration avec les autres membres. Nous avons travaillé ensemble afin de définir les relations et interactions entre les différentes composantes de l'application. J'ai proposé un modèle, comme les autres membres de l'équipe, que nous avons analysé et discuté ensemble.</p> <p>3. Développement de la fenêtre d'accueil : J'ai développé l'interface d'accueil, intégrant le logo, le nom du projet, et le système d'authentification pour les rôles. J'ai également intégré des messages d'erreur adaptés pour une meilleure expérience utilisateur.</p> <p>4. Fonctionnalités du StoreKeeper : J'ai développé les fonctionnalités principales pour le rôle de StoreKeeper, notamment l'ajout et la modification des composants du stock. Cela inclut la gestion des attributs obligatoires comme le type, le sous-type, le titre descriptif unique et la quantité. J'ai collaboré avec l'équipe pour m'assurer que ces fonctionnalités soient bien intégrées avec la gestion des commandes.</p> <p>5. Fonctionnalités de l'Assembler : J'ai développé les fonctionnalités permettant à l'Assembler de visualiser toutes les commandes et de les gérer (validation ou rejet avec justifications).</p> <p>6. Tests unitaires : Après de longues discussions avec mon coéquipier Othman, ainsi que l'essai de plusieurs tests erronés, Othman et moi-même avons rédigé des tests unitaires pour valider la logique d'acceptation ou de rejet d'une commande en fonction du stock disponible.</p>
KHELIFI	Yassine	300328990	<p>Dans ce projet, j'ai apporté des contributions, parmi lesquelles j'ai ouvert un compte Firebase en le connectant avec le programme, l'ai lié à l'adresse e-mail, et travaillé sur le diagramme UML.</p> <p>J'ai également travaillé sur la liste tabulaire, amélioré la classe Storekeeper, développé la</p>

			<p>réinitialisation de la base de données et aidé au développement de la fonctionnalité Réinitialiser le stock. De plus, j'ai mis à jour certains boutons de retour dans les classes et travaillé sur les tests Espresso.</p>
OUAJJOU	Othmane	300391838	<p>Dans le cadre de ce projet de programmation, ma principale responsabilité était axée sur l'implémentation du backend, en particulier les fonctionnalités associées au rôle « Administrator ». Lors du premier livrable, j'ai développé des fonctionnalités essentielles pour la gestion des utilisateurs, notamment l'ajout, la suppression et la modification des utilisateurs de type « Requester ». Cela a permis d'assurer une gestion fluide et sécurisée des données utilisateurs dans la base de données. Pour le deuxième livrable, j'ai implémenté une fonction d'initialisation permettant de charger automatiquement des données (telles que des utilisateurs ou des éléments de stock) à partir de fichiers externes dans la base de données, tout en évitant les collisions avec les données existantes. En parallèle, j'ai également ajouté des fonctionnalités avancées pour l'administrateur, telles que la possibilité de vider complètement la base de données, de la réinitialiser en chargeant des données prédéfinies depuis des fichiers, ainsi que de réinitialiser le stock tout en supprimant les commandes en cours. Ces outils offrent une grande flexibilité et facilitent la gestion des données pour les administrateurs. Par ailleurs, j'ai contribué à la modélisation UML du projet en réalisant des diagrammes d'activité, de classe, de séquence et d'état, chacun reflétant fidèlement les éléments implémentés et respectant les principes de cohérence et de validité. Ces diagrammes ont été continuellement ajustés au fil des livrables pour inclure les nouveaux éléments du système. En plus de mes tâches principales, j'ai également participé au réarrangement du code pour améliorer sa lisibilité, sa maintenabilité et son efficacité, tout en collaborant avec l'équipe pour assurer une intégration harmonieuse entre le frontend et le backend. Ce projet m'a permis de développer des compétences approfondies en conception logicielle, en gestion de bases de données, et en modélisation UML, tout en</p>

			apportant une contribution significative au succès de l'équipe.
NAGIFI	Othman	300347989	<p>Au sein de ce projet , j'ai eu l'opportunité de contribuer activement au développement de l'application dans divers aspects , tout en interagissant dynamiquement avec mes coéquipiers afin de garder une bonne cohésion d'équipe .</p> <p><u>1.Contribution à l'implémentation du role administrator:</u></p> <p>Dans le cadre du premier livrable , j'ai eu l'occasion de contribuer au développement au fonctionnalités de l'administrateur , notamment la méthode qui permet la suppression de requesters(removeRequesters) .</p> <p>De plus , j'ai pu ajouter les méthodes requises permettant la lecture des requesters à partir du fichier Json et leur ajout dans la base de donnée (readJsonFile,loadRequestersFromJson) ainsi que la logique lié à leur suppression d'un fichier donné (removeRequestersFromJson) ainsi que le layout qui leur correspond .</p> <p><u>2.Contribution à la conception des diagrammes uml</u></p> <p>J'ai pu proposé multiples modèles de diagramme afin que moi et mon équipe puissent trouvé un diagramme convenable qui s'alignent avec nos idées et propositions respectives .</p> <p><u>3.Tests Unitaires :</u></p> <p>Moi et mon coéquipier yassine avons échangé plusieurs de nos idées quant à l'implémentation des tests unitaires , afin d'éviter de tomber dans des niveaux de complexité élevés qui nécessitent la création d'un environnement de tests spécifiques à chaque méthode tel que l'utilisation des Mockitos , notre communication nous a permis de gagner du temps et par suite créer des tests unitaires efficaces et convenable à notre utilisation .</p>

6. Estimation du temps passé sur chaque livrable

Par estimation, si nous comptabilisons les heures passées sur le projet pour chaque personne, nous aurions dû passer 50h sur chaque livrable.