RAPPORT FINAL : TROQC

KORALLIA FRENETTE

WILLIAM LEMIRE

TRAVAIL PRÉSENTÉ À JEAN-CHRISTOPHE DEMERS

DANS LE CADRE DU COURS **420-C61-IN – PROJET SYNTHÈSE**

TECHNIQUES DE L’INFORMATIQUE

CÉGEP DU VIEUX-MONTRÉAL

20/12/22

# Présentation générale de votre livrable

Initialement, nous voulions créer une application web pour petites entreprises utilisant le commerce électronique afin de centraliser leurs opérations. Toutefois, en y réfléchissant, nous réalisons que l’idée développée devient plus ou moins utile pour une entreprise, considérant tous les autres programmes de commerce électronique s’intégrant aux sites webs. Nous ressentons un malaise à développer une application sans aucune utilité pratique. Cependant, dans notre entourage, nous connaissons des personnes aimant ces hobbies et intéressés à les monétiser. Notre idée de projet s’applique beaucoup mieux à des particuliers voulant échanger leurs produits plutôt qu'à des entreprises.

TroQc forme une plateforme d'échange entre particuliers afin de participer à la relance économique post-Covid. Étant donné la courte fenêtre de temps pour développer cette application, les aspects fondamentaux tels: la création d’un compte, le stockage de produits en base de données, la possibilité de faire des ventes, et le calcul du *bottom-line* représentent le cœur du projet.

Nous abandonnons donc l'idée de créer un module d’assignation de tâches. Nous le remplaçons par un module de ventes permettant uniquement l’entrée manuelle de celles-ci. Nous souhaitions ajouter cette fonctionnalité dans le module de Finances. Mais nous trouvions plus logique de l’inclure dans le module de Ventes. Une fois le logiciel redéfini tel une plateforme d’échange entre particuliers, le module d’assignation de tâches perdait son sens.

En ce qui a trait au livrable, une application permettant tout d’abord de créer un compte voit le jour.. Ensuite, à partir de celui-ci, il devient possible de s’authentifier pour accéder au lobby. Une fois à l’intérieur de dernier, l’usager peut accéder des modules, ainsi qu’un bloc-notes lui donnant l’option de se laisser des notes. Il possède l’option d’ ajouter des ventes manuelles dans le modules de ventes, en choisissant un produit en inventaire et la quantité désirée. Dans le module d’inventaire, Il bénéficie de la capacité de créer des gabarits, essentiellement des produits revendables regroupant plusieurs informations telles une description, un SKU, les matériaux requis pour sa production, etc. Afin d’ajouter un produit en inventaire, il ne suffit que d’ajouter un lot d’une certaine quantité. Une vérification de la quantité nécessaire de matériaux pour créer ce lot doit s’exécuter. S’il y en a assez, le produit en question peut être vendu. A priori, il doit ajouter les matériaux nécessaires à la création des gabarits. L’application contient aussi un module de finances, dans lequel l’usager peut, pour une plage de temps, voir le profit généré, calculé en fonction des ajouts de matériaux et produits et des ventes faites. Le graphique au bas de la page affiche les ventes effectuées par jour ainsi qu’une projection de ces dernières suivant un certain modèle de régression pour le prochain mois. Finalement, un module de configurations, où l’usager peut modifier à sa guise les informations personnelles de son compte, sa photo de profil, ainsi que son mot de passe se trouve aussi dans l’application.

Le temps nous a fait défaut pour développer toutes les fonctionnalités souhaitées au départ. Mais nous sommes toutefois satisfaits de ce prototype bien fonctionnel. Nous avons dû faire des sacrifices pour nous rendre à ce point.

# Résumé du développement pendant la session

## Respect ou modification des objectifs

Jusqu’à la mi-session, nous sommes un peu en avance sur nos objectifs, grâce au temps “mort” présent en début de session. Inévitablement,le temps allait se resserrer dans les semaines suivantes. Toutefois, à cette étape plus de 50% des tâches sont complétées. Cependant, en termes de charge réelle de travail nous ne pouvions déterminer en amont. Il a fallu mettre les bouchées doubles. Nos tâches pas assez précises, minimisaient la complexité sous-jacente. Par exemple, nous voulions “Créer la page JSP pour le Lobby, ainsi que la connexion avec le contrôleur en 6 heures. En réalité, beaucoup plus d'heures ont été nécessaires pour y parvenir.. Nos estimés trop idéalistes tenaient plus ou moins compte des éventuels “*bugs”* rencontrés. Il était impossible d’évaluer correctement la difficulté de programme. De plus, de nombreux tutoriels et guides lus vantaient grandement la simplicité de Spring.

Non seulement l’utilisation de Spring se doit d’être prise en compte mais il faut aussi ajouter la complexité de Java EE, autre mastodonte en soi, à l'équation . Par chance, le cours de veille technologique nous a permis de nous familiariser avec la plateforme Java EE et le framework Spring. Néanmoins, parler de ces deux technologies et les utiliser adéquatement demeurent deux choses complètement différentes. Par exemple, nous avions hâte d’utiliser le BeanFactory afin de pratiquer l’inversion de contrôle (*Inversion of Control*). Toutefois, après l’avoir configuré à la lettre et avoir passé beaucoup plus d’heures permises par notre budget de temps, nous avons dû abandonner l’idée et instancier les objets directement dans les classes.

Au début du projet, nous voulions réellement faire de notre mieux et utiliser les meilleures pratiques. Plus le temps filait, plus les tâches complexes s'additionnent (et les bugs complexes), Nous devions donc opter pour une stratégie visant la quantité et non la qualité. Tel qu’appris lors de notre formation: quand une date de remise s’en vient et qu’un livrable est dû, il faut faire ce qu’il faut faire pour que le programme fonctionne. Le contenu du sprint 1 a été relativement bien suivi. Par contre, au sprint 2 nous avons déraillé. Nous avions en essence ouvert plusieurs boîtes de Pandore. Afin de nous réajuster, nous faisions des scrums à chaque jour de travail sur le projet afin de s’assurer de l’avancement de nos différents modules. Les objectifs sont restés les mêmes. La seule manière de se rendre à la ligne d’arrivée consistait en un suivi constant entre les membres de l’équipe.

Les difficultés rencontrées proviennent principalement de notre sous-estimation des tâches devant être accomplies. De plus, nous avons connu énormément de problèmes avec GIT. En combinant nos versions (*merge*), même s’il n’y avait pas de conflits, nous devions toujours passer de longs moments à rendre le code fonctionnel à nouveau, pour des raisons qui nous échappent.

## Sources/Librairies/Stratégies

Nous avons eu des problèmes à trouver des sources qui traitaient de développement front-end avec les technologies que nous utilisions. Les références démontraient des exemples de bases qui apportaient peu d’aide. Certaines sources, en ce qui a trait à la communication du serveur vers la vue suggèraient d’écrire des *scriptlets*, soit du code Java imbriqué dans les pages .jsp. Alors que d’autres disaient d’éviter cela à tout prix. Pour certaines fonctionnalités, comme les messages d’erreurs ou le graphique du module de finances, nous avons dû utiliser des *scriptlets*. Essentiellement, si nous ne trouvions pas de réponse, nous devions nous retrousser les manches et trouver une manière de faire fonctionner le tout.

Côté back-end, c’était le contraire, il y avait une abondance de sources. Toutefois, il y en existait tellement qu’il fallait filtrer à travers beaucoup de contenu. La majorité des réponses venaient de petits tutoriels ou de vieilles publications provenant de communautés spécifiques à Java.

Quant à la stratégie utilisée, nous avons employé la stratégie de développement Agile.

## Événements importants/Commentaires sur la session

Au retour de la mi-session, nous sentions déja la pression montée de plusieurs crans.

La fin s’annoncait bien pire que la précédente session. Nous nous sommes donc motivés à retrousser nos manches et à utiliser tous les petits trous disponibles dans nos horaires pour continuer ce travail. Toutefois, à un certain point, la fatigue s’installant de plus en plus, il devient impossible de tout faire en suivant nos hauts standards. Nous avons connu beaucoup de découragement et nous nous sentions parfois perdus, jouant dans un territoire peu documenté en ligne. Pour utiliser une expression anglophone, nous nous trouvions dans une sorte de “*catch-22”,* plus le temps avançait, plus les problèmes occasionnés demandaient du temps à être réglés tout en ayant de moins en moins de temps. Cette fin de session a été très éprouvante pour nous deux.

# Fonctionnalités

| **MODULE** | **FONCTIONNALITÉ** | **Fonctionnel** | **Semi-fonctionnel** | **Non-fonctionnel** | **Abandonné** |
| --- | --- | --- | --- | --- | --- |
| Connection | Authentification d’usager | **X** |  |  |  |
|  | Création de compte | **X** |  |  |  |
| Lobby | Ajouter des notes dans le bloc-notes | **X** |  |  |  |
|  | Modifier/Supprimer les notes | **X** |  |  |  |
| Finances | Afficher le bottom-line pour une plage de temps | **X** |  |  |  |
|  | Afficher la projection des ventes pour la période spécifiée | **X** |  |  |  |
|  | Générer un rapport des finances | **X** |  |  |  |
| Inventaire | L’insertion de matériaux | **X** |  |  |  |
|  | Modification de matériaux | **X** |  |  |  |
|  | Suppression de matériaux | **X** |  |  |  |
|  | L’insertion de gabarits | **X** |  |  |  |
|  | Modification de gabarits |  | **X** |  |  |
|  | Suppression de gabarits |  | **X** |  |  |
|  | L’insertion de Lots | **X** |  |  |  |
|  | Génération automatique de SKU pour gabarits (si champ vide) | **X** |  |  |  |
|  | Génération de code QR |  |  |  | **X** |
|  | Alertes si produits/matériaux descendent sous un seuil spécifié |  |  |  | **X** |
|  | Alertes lors de suppressions/modifications |  |  |  | **X** |
|  | Génération d’un rapport Excel des items en inventaire | **X** |  |  |  |
| Ventes | Ajouter une vente manuelle | **X** |  |  |  |
| Configurations | Modifier infos personnelles | **X** |  |  |  |
|  | Modifier photo de profil | **X** |  |  |  |
|  | Modifier mot de passe | **X** |  |  |  |
| Misc. | Déconnecter l’usager | **X** |  |  |  |
|  | Envoyer un courriel pour modifier le mot de passe si oublié |  |  |  | **X** |

# Améliorations possibles

## CSS

Nous ne nous sommes pas posé de questions quant à la meilleure approche pour le CSS. Nous avons utilisé les connaissances développées lors de nos cours de web. Il aurait été préférable de créer des classes de style génériques pour les éléments récurrents du visuel, comme des containers, des éléments de menus, etc. Les classes fournies par bootstrap ne nous permettaient pas d’avoir le style que nous voulions, donc chaque élément visuel (ou presque) à son propre style. Ceci entraîne de la redondance. Par contre, nous pouvons être granulaire dans l’affichage et contrôler chaque élément. De plus, il aurait été pertinent, dès le départ, de faire le visuel pour les différentes tailles d’écrans. Effectivement, nous voulions le faire à la toute fin, le temps pour implémenter cette fonctionnalité nous a fait défaut.

## Validation de champs

Nous nous étions dit que nous allions utiliser la fonctionnalité de Validators de Spring afin de faire une bonne validation des champs de nos différents formulaires. Toutefois, nous avons tout de même dû en faire au fur et à mesure quand nous avions des erreurs. Nous avons manqué de temps pour même juste entamer le processus de validation. Nous avons donc rajouté de la validation sur quelques formulaires importants, toutefois nous aurions dû le faire systématiquement et non seulement à la fin du projet.

## Back-end

Il est difficile de dire les améliorations que les modules de spring auraient pu apporter. Car nous avons manqué de temps pour réellement comprendre comment les utiliser. Cela dit côté amélioration, il aurait été préférable de gérer tous les rapports de la même classe plutôt que de les faire individuellement.Il aurait été aussi possible d’améliorer le traitement de données de l'algorithme de régression. Par contre, je crois que celle-ci est mineure. Le plus grand avantage aurait été d’utiliser la puissance offerte par Spring et ses nombreux modules tel que l’utilisation du beanfactory. Au moment de la création de ce projet, nous avions vraiment sous-estimé la complexité de ceux-ci. Alors, nous avions fait un back end basé sur nos connaissances de l’époque. Avec le recul, je suis certaine que nous pourrions trouver de meilleurs outils et surtout repenser la conception maintenant que nous avons plus d’expérience pratique.

# Auto-évaluation individuelle

## William

Je suis plutôt fier de ce que nous avons accompli lors de notre projet final. Nous avons travaillé de manière consistante tout au long de la session pour avoir un livrable fonctionnel et visuellement attrayant. J’ai été assigné dans notre équipe au développement front-end, dû à mes talents de design graphique. Toutefois, tout au long de notre parcours, nous avons appris à coder de manière efficace et claire tout en s'efforçant à écrire du code réutilisable.En ce qui à trait au développement web, cette notion ne s’est jamais ancrée en moi. En fait, ce domaine n’a jamais été poussé dans nos cours de web. Je trouve que le code, surtout le javascript et le CSS, a beaucoup de redondances, mais c’est peut être la manière dont j’ai appris à le faire. Ayant eu de bonnes notes dans ces cours, je n’ai jamais eu raison de questionner ma manière de procéder. Après avoir fait un tel projet sur plusieurs mois, ces aspects commencent à me sauter aux yeux. Je remarque plusieurs endroits qui bénéficieraient de plus de modularité. Je ne saisis pas assez bien les nuances de Javascript pour coder de manière élégante. J'ai appris dans le cadre des cours de web à faire fonctionner mes projets, mais certainement pas comment les optimiser. Dans l’optique de faire du développement agile, mon but était de faire fonctionner le tout, peu importe la manière. Somme toute, j’ai réellement tout donné ce que j’avais. Ayant été plongé dans un nouvel environnement de développement, d’avoir réussi à réaliser tout ceci et d’avoir relevé plusieurs défis montre que j’ai atteint les objectifs de l’épreuve certificative.

## Korallia

Je suis très fière de notre projet en tous points. Si je devais faire une évaluation, je dirais que j’ai fait tout ce dont j’ai été capable. J’ai poussé mes limites physiques , morales et émotionnelles pour m’assurer de présenter un projet fidèle à mes attentes. Cependant, je me rends compte que celles-ci étaient absolument non-réalistes. Alors ma plus grosse critique devient mon manque d’amour et de respect envers moi-même. Mon niveau d'attente était beaucoup trop élevé par rapport à mes connaissances et mon niveau d’expérience. Pour ce projet, j’ai assemblé la base de données, fait tous les daos sauf celui du lobby, construit un algorithme de régression avec trois modèles différents (linéaire,exponentielle et logarithmique). J’ai bâti au moins 3 patrons de conception différents (builder,itérateur,dao). J’ai écrit notre propre class LinkedList dépassant les attentes minimales tout en gardant en tête le plus possible les principes orientés objets.

Considérant mon niveau d’études collégiales, je considère avoir relevé le défi et atteint les objectifs de l’épreuve certificative.

# Auto-évaluation d’équipe

## William

Korallia et moi avons développé une complicité dès le début du programme. Nous avons fait beaucoup de projets à deux..Nous communiquons bien ensemble et avons une bonne éthique de travail. Nous avons appris très rapidement que nous pouvions toujours compter l’un sur l’autre. Nous avons divisé les tâches en fonction de notre expertise. Ce choix nous a permis de développer plus rapidement. Toutefois, cette approche a entraîné un gros désavantage. Si j’avais des problèmes avec le back-end, j’hésitais à apporter des modifications au code. Je me disais que le tout était ainsi pour une raison spécifique et ne voulant pas tout bousiller, j’attendais qu’elle soit présente. Je juge cette procédure inefficace, compte tenu du temps restreint et la nécessité de progresser. Je ne sais pas comment dans les présentes circonstances j’aurais pu mieux supporter l’équipe et vice versa. Nous nous efforçions de garder une bonne humeur lors de nos séances de travail tout en cultivant une énergie positive. L’’extrême fatigue et le désespoir, quant à eux, cherchaient sans cesse à se pointer le bout du nez. Malgré la folie de ces dernières semaines, j’ai eu un plaisir fou à développer cette application, dont je suis très fier, avec ma coéquipière.

## Korallia:

Ce fut le meilleur projet d’équipe auquel j’ai pris part. Chaque membre avait une minutie et un professionnalisme nécessaires à la réalisation d’un aussi gros projet. Chaque membre possédait aussi les capacités interpersonnelles de communication et d’empathie pour se comprendre et se supporter dans cette épreuve difficile. William était non seulement un partenaire en tout point idéal pour moi. Il complétait mes faiblesses. Toutefois, à nous deux nous formions une équipe, dépassant les limites et les forces de chaque individu. J’ai aucun point négatif envers notre équipe. Nous avons vraiment tout donné. Alors que je codais les fonctionnalités de traitement de données. Il n'arrêtait pas de me surprendre par sa capacité à se pousser encore plus loin et à pousser la qualité visuelle à un autre niveau. Cette symbiose d’équipe basée sur l’ambition et la passion est requise pour atteindre ces objectifs.