Christopher-William Archambault-Bouffard

Programmation dans un environnement transactionnel

420-445-AL, gr :01

**Travail personnel 1**

Travail présenté à

François Lacoursière

Département – Informatique

Cégep André-Laurendeau

02/21/2022

J’atteste que j’ai personnellement produit ce travail et que j’ai indiqué tous les emprunts de textes ou d’idées (provenant de documents écrits, oraux ou visuels).

Signature de l’étudiant : Christopher-William Archambault-Bouffard

1. Introduction au projet

**Votre entreprise**

Librairie au sein de la ville Javatown. Nous fessons le prêt de documents pour les clients et ceux-ci ont un délai pour rapporter ce bien. Si le délai est dépassé, nous émettons des amendes de 25 cents par jour au client. Pour emprunter de nouveau les amendes doivent être payées.

**Typologie de produit**

Un programme java qui dépose et récupère les données stocker sur une base de données H2. Ce programme fait l’emprunt de documents auprès du client et le préposé émet des amendes sur ce programme aux clients qui ne rapporte pas le ou les documents à temps.

**Objectifs du produit**

- Refaire un système désuet

- Gérer l’emprunt des livres et émettre des amendes aux retardataires

- Refont d’un produit déjà existant

- Gérer l’emprunt des livres, les amendes et bloquer les usagers qui n’ont pas payés leur due.

**La valeur ajoutée du produit**

- Rendre le système complétement informatisé et automatique

- La facilité a repéré les documents emprunter, les amendes aux clients et bloquer les usagers qui n’ont pas payés

**Le modèle économique**

- L’emprunt des documents et les retards ?

- Le revenu vient des amendes qui sont les retards accumulés pour un documents qui est 25 cents par jours.

- Les financeur majeur sera la ville car c’est un service public mais pour continuer le financement il faut tenir les amendes contres les retardataires et ne pas les laisser emprunter les personnes qui ont des amendes.

<https://www.google.com/imgres?imgurl=https%3A%2F%2Fimage.slidesharecdn.com%2Fnorse-mythology-130205172128-phpapp02%2F85%2Fnorse-mythology-5-320.jpg%3Fcb%3D1360085565&imgrefurl=https%3A%2F%2Fwww.slideshare.net%2Fmichaelheumann90%2Fnorse-mythology-16370210&tbnid=cZs1vifPQg4g5M&vet=10CLIBEDMotwFqFwoTCIiR6O7TjPYCFQAAAAAdAAAAABAC..i&docid=mWON_OWYldtl3M&w=320&h=240&q=nordic%20mythology%20animals&hl=fr&ved=0CLIBEDMotwFqFwoTCIiR6O7TjPYCFQAAAAAdAAAAABAC>

**Pour qui ?**

- Les personne de Javatown qui veut emprunter un livre, le préposé(admin) pour émettre des amendes et entrer des livres

Conseil : il est possible que vous ayez plusieurs types d’utilisateurs aussi appelés acteurs. Décrivez-les avec précision et n’oubliez pas de segmenter votre cible en fonction du rôle et des attentes de vos utilisateurs. PS : si un administrateur chez vous va utiliser un back-office pour paramétrer votre produit, il est considéré comme un utilisateur aussi. Il vous faudra donc l’inclure dans la description de vos utilisateurs.

**Perspectives de développement**

- Il sera mesuré par le fait que les amendes sont émises aux bonnes personnes et les livres qui sont emprunter dans la librairie sera seulement non-empruntable mais encore visible.

- 2

- Implémenter un site web user/admin?

Conseil : définissez les indicateurs clés de performance, ou KPI (Key Performance Indicator), de votre produit afin de mettre l’accent sur les priorités du projet. Avoir une idée des performances attendues permet aussi d’assurer la scalabilité future de votre produit. En effet, vos KPIs serviront de guide à votre prestataire lorsqu’il devra effectuer des choix techniques.

2. Tableau fonctionnel

Ce tableau résume vos besoins fonctionnels sous forme de ‘user story’. Il ne s’agit pas de remplir des solutions techniques, mais bien des besoins. Voici les champs que vous aurez à remplir :

* **Story : votre besoin ;**
* **Critères d’appréciation : les critères par lesquels la bonne réalisation du besoin sera mesurée ;**
* **Niveau de performance : le niveau visé pour chaque critère listé.**

Priorités

Quelles sont les fonctionnalités prioritaires ?

Nous vous invitons à hiérarchiser par ordre d’importance et d’urgence les fonctions que vous voulez que votre produit remplisse.

Un ‘story map’ peut être utilisé pour mieux définir les priorités. (Tel que vu au cours 420-535)

**Comment prioriser avec la méthode MoSCoW ?**

Ce Framework qui fait partie des méthodologies agiles permet de vous faciliter le travail de priorisation et de hiérarchisation de vos besoins.

* **Must : fonctionnalités essentielles et non négociables.**
* **Should : fonctionnalités à implémenter dans la mesure du possible.**
* **Nice-to-have: fonctionnalités qui contribuent à la satisfaction de l’utilisateur mais qui ne sont pas essentielles.**

Liste des ‘User stories’

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Priorité | En tant que… | Je veux… | Afin de… |
| 3 | Must | Client | Voir les documents disponibles | Choisir un document à emprunter |
| 4 | Must | Client | Sélectionner un document | L’emprunter |
| 5 | Must | Client | Désélectionner un document | Le rendre |
| 7 | Must | Préposer | Emmètre des amendes | Suivre le compte des retards et bloquer leur accès à l’emprunt |
| 1 | Must | Préposer | Créer des documents | Rendre disponible de nouveau documents |
| 2 | Must | Préposer | Supprimer des documents | Rendre indisponible certains documents |
| 6 | Must | Préposer | Voir la liste de tous les retards pour un client | Pouvoir charge la somme totale au client |
| 8 | Must | Préposer | Voir les retards cumulés pour un document | Pour charger un client pour les retards d’un document |
| 9 | Should | Préposer | Avoir une liste des personnes qui veulent emprunter un même document | Le prêter aux personnes dans l’ordre du premier arrivé |
| 11 | NTH | Client | Réemprunter un document | Renouveler l’emprunt sans frais |
| 10 | Should | Client | Voir les amendes à son insu | Pour voir combien on doit ou simplement pour payer le montant dû |

3. Contraintes graphiques

**Valeurs et sources d’inspiration**

Les documents, les non-disponibles car emprunter et les amendes en tant que client ou préposé

- Amazon, Archambault, leslibrairies.ca

4. Contraintes techniques

**Architecture**

- Ne pas laisser un client qui a une dette emprunter de nouveau

- JDBC?

- Sur un serveur H2 ?

- Où sera hébergée la solution développée ?

- On implémenterons le CI/CD

**Spécificités Techniques**

- On utilise la technologie de base de données H2 car celle-ci est gratuit la relié à un site internet pour avoir une interface graphique pour le client et le préposé(admin).

- Nous visons tous les systèmes d’exploitation car nous voulons le faire en site web pour que tous puissent y avoir accès

- Les versions Windows 10, Android 12 et Ios 15.3.1 ?

- Tout les navigateurs web sont ciblé

- Les ordinateurs et les smartphones ?

5. Encadrement du projet

**Typologie de la gestion de projet**

- Le type de gestion choisi sera Kanban pour se concentrer sur les tâches principales

- Le model sera conçue de zéro, a part pour quelle sont les nécessités.

- Un peu des deux pour faire un pont entre les membres qui font le site web et eux qui font le java avec la base de données et pour aussi que les membres de la même équipe collabore ensemble et pense à des solutions pour des problèmes non résolues pas d’autres ?

- Le travail est réparti de façon à ce que l’équipe se divise pour avoir des personnes sur le java qui va être relié à la base de données pour stocker les données et faire en sorte que les personnes qui non pas payés peuvent pas emprunter et aussi d’émettre des amendes aux clients retardataires. L’autre moitié travaillerais sur le site web pour afficher les documents disponible les retards et les montants de ces retards et faire deux sections une pour préposé(admin) et une pour les clients qui peuvent seulement consulter les documents et voir leurs amendes et montant dû

Conseil : ne négligez pas cette partie. Soyez précis sur chaque détail des prestations attendues et définissez avec précision les responsabilités de chacun. Ajoutez toute autre question qui vous semblerait pertinente.

ANNEXE. Diagrammes UML et schémas de base de donnée.

Diagram

Description automatically generated