



Application DeliveryFit

Proposition commerciale

1. Contexte

Alisa's Closet souhaite créer l'application **DeliveryFit** dont le rôle sera de s'insérer sur le site actuel de l'entreprise au moment de validation de panier et optimiser le lieu d'expédition des livraisons en calculant le temps d'expédition le plus rapide.

Le résultat de ce calcul définira d'où part le colis en choisissant entre :

- Le centre de livraison
- Le magasin le plus proche du lieu de livraison.

1.1 Contexte global

Les buts ultimes de l'application seront d'optimiser les couts, les temps de livraison et l'impact sur l'environnement d'Alisa's Closet.

1.2 Fonctionnalités métier

Trouver tous les magasins ayant les produits achetés en stock

- Trouver le lieu le proche du lieu de livraison en comparant :
 - Le centre de livraison principal de Alisa's Closet
 - Le magasin le plus proche

2. Réalisation du projet

2.1 Définition des tâches techniques

Certaines tâches peuvent être réalisées en **parallèle**. Les autres tâches peuvent rencontrer une contrainte de dépendance : elles dépendent de tâches précédentes. Le planning estimé prend ce fait en compte pour optimiser les temps de production.

Liste de tâches fournie par le client

- Créer l'application Back-end en utilisant Spring Boot
- Créer l'application Front-end en utilisant Angular
- Comparer les distances entre les magasins et/ou le centre de livraison et le domicile
- Intercepter l'URL du site de vente en ligne et rediriger vers l'URL DeliveryFit
- Transmettre les informations de livraison
- Trouver le point de livraison le plus proche
- Envoyer une notification au magasin expéditeur
- IHM - Créer la page de validation du panier avec les informations d'expédition
- Créer la redirection de DeliveryFit vers le site de vente en ligne

Observations

- Nous supposons que le travail technique de l'architecture aura lieu en premier.
- Les applications Back-end et Front-end peuvent être créées simultanément.
- Certaines fonctionnalités dépendent de précédentes.
- D'autres fonctionnalités sont moins importantes que les autres.
- Des fonctionnalités primaires.

2.2 Gestion des points de complexité

Je compterais la complexité avec une note sur 20. 20/20 étant considéré comme étant extrêmement complexe.

Tâche technique	Complexité (/20)	Estimation temps (heures)	Marge (heures)
Documentation architecture technique	5	8	0
Schématisation de la base de donnée.	3	4	4
Créer l'application Backend	2	4	2
Créer l'application Frontend	2	4	2
Fonctionnalité de comparaison entre les magasins et/ou le centre de livraison et le domicile.	12	16	8
Fonctionnalité d'interception de l'url du site de vente en ligne et rediriger vers l'url DeliveryFit	8	8	2
Transmission des information de livraison	5	4	2
Trouver le point de livraison le plus proche	6	6	2
Création de la redirection de DeliveryFit vers le site de vente en ligne	4	4	2
Notification au magasin expéditeur	6	4	2
Créer la vue de la page de validation du panier avec les informations d'expédition	7	6	2
Mise en production	4	4	4

2.3 Coûts

L'équipe est composée de 3 développeurs :

- 1 développeur front-end
- 1 développeur back-end
- 1 architecte / développeur

Les tarifs journaliers moyens (TJM) de ces développeurs sont de 800€ pour l'architecte et 700€ pour les deux autres. Les tarifs étant des taux journaliers, nous pouvons utiliser un calcul simple de produit en croix en partant sur une base de 8h par jour.

Temps / 8 * TJM

Tâche	Temps (heures)	Membre (développeur)	TJM	Total (euros)
Documentation architecture technique	8 (0)	Architecte	800	800
Schématisation de la base de donnée.	4 (4)	Architecte	800	400
Créer l'application Backend	4 (2)	Back-end	700	350
Créer l'application Frontend	4 (2)	Front-end	700	350
Fonctionnalité de comparaison entre les magasins et/ou le centre de livraison et le domicile.	16 (8)	Back-end	700	1400
Fonctionnalité d'interception de l'url du site de vente en ligne et rediriger vers l'url DeliveryFit	8 (2)	Front-end	700	700
Transmission des information de livraison	4 (2)	Front-end	700	350
Trouver le point de livraison le plus proche	6 (2)	Back-end	700	525
Création de la redirection de DeliveryFit vers le site de vente en ligne	4 (2)	Front-end	700	350
Notification au magasin expéditeur	4 (2)	Back-end	700	350
Créer la vue de la page de validation du panier avec les informations d'expédition	6 (2)	Front-end	700	525
Mise en production	4 (4)	Architecte	800	350

Total: 6450 €

2.4 Risques identifiés

Conséquences / Probabilité	Peu probable [1]	Probable [2]	Très probable [3]
Conséquences catastrophiques [3]	Absence (maladie, décès, autre...) d'un développeur	/	/
Conséquences modérées [2]	La fonctionnalité principale ne fonctionne qu'à moitié : le point de livraison trouvé n'est pas exactement le plus proche.	Ne pas finir dans le temps imparti	L'intégration "transparente" de l'application dans le processus existant n'est pas totalement optimisée : interception et redirection
Conséquences négligeables [1]	/	Les algorithmes pour trouver la liste des points de livraisons et trouver le plus proche peuvent ralentir le processus de paiement des utilisateurs.	Le suivi en temps réel des colis n'est pas totalement implémenté ou performant.

Le tableau ci-dessus tente de classer un certain nombre de risques identifiés, qui pourrait porter atteinte à la bonne finalisation et livraison du projet.

Les risques sont classés en ordre d'importance pour apporter une vigilance adéquate pendant la production et pour permettre de prévoir, en amont, une série de solutions possible en fonction du taux de risque final.

Risque = Conséquence * Probabilité

Taux de risque

Réactions possibles

1 - 3	Repousser la date de livraison le temps de prendre des mesures adéquate ou présenter quand même l'application en proposant au client un deuxième sprint à prix réduit pour finaliser l'application et améliorer les aspects ayant des lacunes.
3 - 6	Ne pas présenter l'application et imposer un deuxième sprint à prix réduit pour finaliser l'application
6 - 9	Imposer un deuxième sprint à une date ultérieure, le temps de prendre des mesures importante (engager un autre développeur, remplacer un développeur, former un développeur, revoir le backlog à la baisse, refaire une phase d'architecture)

2.5 Définition des objectifs de performance

Il est possible, en amont, d'identifier deux fonctionnalités qui pourraient être la source principale d'un ralentissement de l'application, et donc, de l'expérience utilisateur à son passage au paiement.

Cela dit, globalement, l'application n'a pas suffisamment de fonctionnalités et fonctions déclenchées aux événements pour que l'application soit ralentie de manière significative.

Voici les trois suspects principaux :

- **Requête SQL**
 - Nous voulons récupérer la liste de tous les magasins à proximité de la destination du colis.
- **Algorithmie**
 - Nous calculons les distances pour trouver le point de livraison le plus proche.
- **Geo fencing**
 - Avec *PostGreSql*, par exemple, le traitement de géo-repérage est intégrée dans les fonctionnalités de la base de données.

Nous chercherons à minimiser le plus possible le temps que prendraient ces deux opérations, mais elles ne devraient pas prendre plus une seconde.

Dans le cas où le client serait non satisfait de l'expérience utilisateur présentée, il sera possible de proposer un deuxième sprint pour améliorer le site.

3. Synthèse

Cette proposition prend en considération les contraintes et le budget du client.

Le prix annoncé de **6450 €** est un prix annoncé dans le cas où aucun obstacle ne ralentirait le temps de développement.

Dans le cas où des obstacles se dresseraient, sont prévues des marges de temps pour chaque fonctionnalité qui pourraient augmenter le prix jusqu'à **9400 €**.

Autre proposition

Il est aussi possible d'aller plus loin dans l'application, si vous le souhaitez, pour améliorer certains aspects de l'application et aller au-delà d'un *Proof of Concept*.

Fonctionnalité	Description
	Il serait sans doute judicieux pour une meilleure satisfaction des clients de pouvoir modifier les détails d'une livraison alors qu'elle a déjà été acceptée .
Modifier la livraison après prise en charge.	<p>Ex : Instructions de livraison, date de livraison... etc</p> <p>Il serait possible de donner cette possibilité de manière limitée. La limite étant le status de la livraison : si le colis est déjà pris en charge par le transporteur, alors l'utilisateur ne peut plus modifier sa livraison, il est trop tard. Par contre, si le colis n'est pas encore mouvement, alors il peut modifier les détails de la livraison.</p>
Click & Collect	<p>Puisque nous avons déjà la liste des magasins les plus proches de la destination du colis, pourquoi ne pas donner la possibilité à l'utilisateur d'aller chercher son colis en main propre dans le magasin en question ?</p> <p>Une telle fonctionnalité aurait des bienfaits environnementaux et financiers pour l'entreprise. La "livraison" à partir du magasin le plus proche devient plutôt une "réservation" du produit avec une date de mise à disposition pour l'utilisateur.</p>

Prix

Pour rajouter les fonctionnalités décrites ci-dessus, il faudrait sans doute rajouter 1 à 3 jours de développement, ce qui augmenterait le prix de 1400€ à 2100€. Pour un total maximum de 8550 € à 11500 €.