

Projet Maintenance APP - DEVE 427

Questions :

1) A partir des documents, établissez et prioriser votre backlog de tâches. Pour chaque tâche, catégoriser le type de maintenance.

2) Estimer le coût de chaque élément du backlog, accompagné d'une explication sur pourquoi cela coûte (ressources utilisées + effort à fournir)

3) Commencer la rédaction d'un plan de maintenance, incluant :

- Le périmètre de la maintenance (qui doit correspondre aux demandes du client)
- Les prérequis pour la maintenance (et donc les prérequis à la reprise du projet par votre entreprise)
- Définir un processus de Modification Request (MRs) ou Problem Reports
- Proposer des SLO (Service Level Objectives), en justifiant ces choix

4) Vous recevez un appel indiquant qu'un magasin Leroy Merlin n'arrive pas à accéder à l'outil de configuration d'aménagement. Le support de niveau 0 n'a pas de solution et suspecte un problème technique. Que faites-vous ?

5) Estimer le coût d'un bug critique détecté en dehors des heures de travail (ressource / effort / pourquoi ?) 1 personne 2 heures

6) Créer un référentiel de code, puis commencer le développement des user story suivantes :

- L'utilisateur peut consulter une liste de produits.
- L'utilisateur peut ajouter un produit au panier.
- L'utilisateur peut payer son panier en indiquant son adresse de livraison. (juste donner son adresse et un bouton payé envoyé de mail peut-être).
- L'administrateur doit pouvoir consulter les dernières commandes.

L'objectif principal est de garantir un code maintenable pour cette nouvelle application. Chaque PR doit également déclencher l'exécution de tests de non-régression.

Par ailleurs, sur ce projet, vous avez décidé d'interdire l'utilisation de `() => {...}` si le code peut être remplacé par `() => ...` (cf. la règle arrow-body-style d'eslint).

7) Bonnes Pratiques : Intégré ou non dans un plan de maintenance, un ingénieur logiciel aura à sa charge d'assurer la maintenabilité du système qu'il développe. Cela passe par une bonne culture logiciel et l'application de bonnes pratiques.

Des suggestions pour améliorer la maintenabilité comprennent :

- Assurer la lisibilité du code ;
- Adopter un code structuré ;
- Réduire la complexité du code ;
- Fournir des commentaires de code utile et précis ;
- Utiliser des techniques facilitant le suivi des erreurs ;
- Assurer la traçabilité du code source à la conception ;
- Suivre des normes de codage ;
- Réduire la complexité des décisions et du flux de contrôle ;
- Effectuer des inspections du code et des cas de test ;
- Maintenir la documentation pendant le cycle de développement ;

3) Plan de maintenance :

- Périmètre de la maintenance : Maintenance **corrective**, correction des bugs rencontrés sur le site. Maintenance **évolutive**, pour toute ajout de nouvelle fonctionnalité prendre rendez-vous avec le commercial. Maintenance **perfective**, modification de la documentation. Il n'y a pas d'adaptative car on ne change pas de logiciel ou n'en crée pas un. On ne nous a pas demandé de faire du préventif donc il n'apparaît pas.
- Les prérequis à la maintenance : Le code de l'ancienne application, la documentation (même partielle) qui a été transférée, l'accès aux outils de développement et de test, ainsi qu'à la base de données. Il faut également disposer des accès aux comptes administrateurs, aux API, ainsi qu'aux outils Google ou Microsoft en lien avec le développement de l'application. Test unitaire, déploiement continu.

Un processus de Modification Request (MRs) ou Problem Reports :

Lorsqu'un problème est rencontré par un client, il est important de vérifier en premier si notre équipe est déjà en train de le traiter. Si le problème est en cours de résolution, il n'est pas nécessaire de nous contacter.

Dans le cas où le problème n'a pas encore été pris en charge, vous pouvez nous contacter par notre plate-forme de ticketing (en astreinte seulement) ou via le support téléphonique au **07 85 96 74 25** (seulement du lundi au vendredi de **8H00 à 12H00** et de **13H30 à 18H00**). Le client ne doit soumettre qu'une seule demande par problème.

Si le support ne répond pas ou ne peut traiter votre demande, vous allez être redirigé directement vers le développeur JavaScript qui analysera la situation et tentera d'apporter une solution. Si ce développeur ne peut pas répondre ou ne sont pas dans ses compétences, la demande sera transférée au développeur Python afin qu'il puisse résoudre le problème.

- SLO :

Problèmes rencontrés	Temps de réponse (jours Ouvrés)	Temps de réponse (Astreintes)	Temps de correction
Demande de modification	15 minutes	XXXXXX	2 à 3 jours ouvrés
Problème sans impacte direct sur la disponibilité	15 minutes	30 minutes	2 jours ouvré
Problème urgent	15 minutes	1 Heure	5 Heures

- Temps de réponse (jours Ouvrés) : 15 minutes car le contact de support peut répondre lorsqu'il est au bureau
- Temps de réponse (Astreintes) : XXXXX car une demande de modification n'est pas notifiée au support.
- Temps de réponse (Astreintes) : 30 minutes car une notification est envoyée ce qui déclenche une astreinte.
- Temps de correction : 1 à 2 jours ouvrés pour la demande de modification car le problème n'est pas urgent et que les développeurs feront la modification au pire des cas le lundi matin. 2 jours ouvrés car cela n'impact pas le chiffre d'affaires du client. 5 Heures avec 1 Heure incluse pour mettre en place les équipes pour pouvoir régler le problème qui affecte surement le chiffre d'affaires du client.

4) Le support de niveau 0 redirige l'appel vers le manager, qui explique que la version actuelle étant en phase alpha et le projet étant en attente, nos équipes ne sont pas en mesure d'intervenir pour résoudre le problème. Il recommande d'utiliser l'outil précédent pour le magasin concerné et de le recontacter en cas de nouvel incident.

5) Lors de l'erreur critique détectée, le support 0 étant d'astreinte, il remonte le bug à un développeur JavaScript qui réglera le bug en 2 heures (comme dit en cours 😊). Alors cela nous fait 100€ pour le support 0 et 200€ pour le développeur JavaScript donc un total de 300 € pour cette erreur.