

Demande de changement #3

La date de livraison est pour le 17 avril 2014 avant 17h30.

Le contenu de ce document s'ajoute à la demande initiale du client et à toutes les demandes de changement précédentes.

Exigences fonctionnelles

Depuis la dernière demande de changement, une nouvelle réalité est apparue dans l'entreprise.

- Certains employés peuvent maintenant charger du temps de transport dans leur feuille de temps. Ce cas s'applique lorsqu'un employé doit se déplacer ailleurs qu'au bureau dans le cadre de ses fonctions. Voici les détails :
 - Uniquement les membres de la direction et les employés de l'administration peuvent charger du temps de transport. Les autres types d'employés n'ont pas ce droit.
 - Le temps de transport correspond au code de projet 777.
 - Le temps de transport est considéré comme du temps de présence au bureau pour les employés de l'administration. Cependant, il est considéré comme du temps de télétravail pour les membres de la direction.
 - Il n'est pas permis de charger plus de 5 heures dans une semaine en temps de transport.
 - Le président de la compagnie est le seul à pouvoir charger plus de 5 heures de temps de transport par semaine. Également, son temps de transport est considéré comme du temps de présence au bureau. Le numéro d'employé du président est 6000.

Nous ajoutons aussi quelques clarifications sur des exigences antérieures.

- Le temps de présence minimal au bureau durant les jours ouvrables pour les membres de la direction est de 8 heures.
- Les membres de la direction n'ont pas de maximum d'heures travaillées par semaine. Ils peuvent travailler autant qu'ils le veulent.

Exigences non fonctionnelles

Voici quelques contraintes à respecter qui touchent votre code :

- Tout le nouveau code écrit pour cette DDC doit être rédigé selon la méthodologie Test-Driven Development.
- Vous devez utiliser Maven dans votre projet pour gérer la construction du logiciel et pour gérer les dépendances de votre projet.
- Vous devez toujours respecter le «Single Responsibility Principle» au niveau des méthodes de classe.
- Chacune de vos classes doit également respecter le «Single Responsibility Principle» tel qu'indiqué dans le chapitre 10 du livre «Coder proprement.»
- Vous devez rédiger suffisamment de «bons» tests unitaires afin d'acquérir une couverture de tests d'au moins 80% de tout votre projet. Vos tests doivent suivre les règles vues en classe et le code de vos tests doit également être propre.