

### **Demande initiale**

Votre client vous demande de lui livrer ces fonctionnalités pour le 3 octobre 2016, avant 9h30. La date de livraison n'est pas négociable.

L'application à développer est un logiciel qui effectuera la validation des déclarations d'activités de formation continue pour les membres d'un ordre professionnel.

Chaque ordre professionnel impose à ses membres d'effectuer un certain nombre d'heures d'activité de formation pour une période donnée. La période en question est appelée un cycle. Dans ce cas-ci, le cycle dure 2 ans et les membres de l'Ordre doivent effectuer un minimum de 40 heures de formation continue durant le cycle.

Le logiciel ne possèdera pas d'interface utilisateur car il est destiné à être invoqué à partir d'une application web. Le contrat ne consiste donc qu'au développement du moteur de validation de l'application.

## **Fonctionnalités**

Le fichier d'entrée, en format JSON, aura l'air de ceci :

```
{
  "numero_de_permis": "A0001",
  "cycle": "2014-2016",
  "heures_transferees_du_cycle_precedent": 2,
  "activites": [
    {
      "description": "Cours sur la déontologie",
      "categorie": "cours",
      "heures": 14,
      "date": "2015-03-20"
    },
    {
      "description": "Séminaire sur l'architecture contemporaine",
      "categorie": "séminaire",
      "heures": 10,
      "date": "2016-01-07"
    },
    {
      "description": "Rédaction pour le magazine Architecture moderne",
      "categorie": "rédaction professionnelle",
      "heures": 6,
      "date": "2014-10-22"
    },
    {
      "description": "Participation à un groupe de discussion sur le
        partage des projets architecturaux de plus de 20 ans",
      "categorie": "groupe de discussion",
      "heures": 6,
      "date": "2015-04-01"
    },
    {
      "description": "Visite d'établissements architecturaux",
      "categorie": "voyage",
      "heures": 2,
      "date": "2016-02-02"
    }
  ]
}
```

Le fichier de résultat généré par le logiciel devra ressembler à ceci :

```
{
  "complet": false,
  "erreurs": [
    "L'activité Visite d'établissements architecturaux est dans une
      catégorie non reconnue. Elle sera ignorée.",
    "Il manque 2 heures de formation pour compléter le cycle."
  ]
}
```

Le programme devra prendre ce fichier comme argument lors de l'exécution du logiciel dans une console. Le fichier où sera placé le résultat devra également être spécifié à la console. Exemple :

```
java -jar FormationContinue.jar declaration.json resultat.json
```

Voici la liste des catégories reconnues pour les activités de formation continue :

- cours
- atelier
- séminaire
- colloque
- conférence
- lecture dirigée
- présentation
- groupe de discussion
- projet de recherche
- rédaction professionnelle

Voici les règles d'affaires à valider :

- Dans un premier temps, uniquement le cycle "2014-2016" sera supporté. L'utilisation de tout autre cycle devra produire un message d'erreur dans le fichier de sortie.
- Toutes les activités déclarées pour le cycle 2014-2016 doivent avoir été complétées entre le 1er avril 2014 et le 1er avril 2016 inclusivement. Si une activité déclarée a été complétée à l'extérieur de cet intervalle, un message doit être produit dans le fichier de sortie et l'activité sera ignorée des calculs. Toutes les dates sont indiquées en format ISO 8601.
- Toutes les activités doivent appartenir à une des catégories reconnues mentionnées précédemment. Si une activité utilise une catégorie non reconnue, un message doit être produit dans le fichier de sortie et l'activité sera ignorée des calculs.
- Le champ "heures\_transferees\_du\_cycle\_precedent" contient des heures de formation en surplus qui ont été complétées lors du cycle précédent et qui peuvent être utilisées dans le cycle courant. Ce nombre ne peut pas être négatif et ne doit pas être supérieur à 7. S'il est supérieur à 7, un message doit être produit dans le fichier de sortie et uniquement 7 heures seront considérées lors des calculs.
- Un minimum de 40 heures de formation doivent être déclarées dans le cycle. Il n'y a pas de maximum. En dessous du 40 heures, un message doit être produit dans le fichier de sortie.
- Un minimum de 17 heures doivent être déclarées dans les catégories suivantes : cours, atelier, séminaire, colloque, conférence, lecture dirigée. Autrement dit, la somme des heures des activités appartenant à l'ensemble de ces catégories doit être supérieure ou égale à 17 heures. En dessous du 17 heures, un message doit être produit dans le fichier de sortie. Les heures transférées du cycle précédent sont comptabilisées dans cette somme.
- Un maximum de 23 heures peuvent être déclarées dans la catégorie présentation. Au-delà de 23 heures, les heures supplémentaires sont ignorées des calculs mais aucun message n'est produit dans le fichier de sortie.
- Un maximum de 17 heures peuvent être déclarées dans la catégorie groupe de discussion. Au-delà de 17 heures, les heures supplémentaires sont ignorées des calculs mais aucun message n'est produit dans le fichier de sortie.
- Un maximum de 23 heures peuvent être déclarées dans la catégorie projet de recherche. Au-delà de 23 heures, les heures supplémentaires sont ignorées des calculs mais aucun message n'est produit dans le fichier de sortie.
- Un maximum de 17 heures peuvent être déclarées dans la catégorie rédaction professionnelle. Au-delà de 17 heures, les heures supplémentaires sont ignorées des calculs mais aucun message n'est produit dans le fichier de sortie.
- Les heures d'une activité doivent être supérieures ou égales à 1 et elles sont des valeurs entières.

## **Contraintes technologiques**

Voici les contraintes que vous devez respecter :

- Le logiciel doit être développé avec le langage de programmation Java (JDK8).
- Il est impératif d'utiliser l'environnement de développement intégré NetBeans 8.1 ou plus.
- Les fichiers d'entrées et de sorties doivent être des documents JSON.
- Les sources doivent être entreposées dans un dépôt GIT sous github.