**Les multivers se rencontrent, et nous embauchons!**

La Commission de la fonction publique souhaite embaucher les meilleurs et les plus brillants employés, non seulement dans ces univers-cités, mais dans d’autres également. Pour cela, elle cherche à mettre en œuvre une nouvelle solution de recrutement qui permettra aux chercheurs d’emploi talentueux de postuler, peu importe où ils se trouvent. C’est un défi colossal : nous devons engager des gens de tous horizons, nous tenir au fait des compétences spéciales dont les nouveaux employés ont besoin pour fournir des services gouvernementaux dans des univers parallèles, et nous assurer que tout le monde peut postuler, même dans les endroits où la connexion Internet laisse à désirer. Nous n’avons plus beaucoup de temps – nous aurions dû commencer hier, et notre machine à voyager dans le temps n’est pas encore tout à fait au point.

On vous demande de créer une application Web qui intégrera les fonctions énumérées dans le tableau ci-dessous.

**Technologies**: Vous pouvez utiliser les technologies de votre choix du moment que l’application soit accessible en ligne et qu’elle soit compatible avec la plus récente version de Chrome ou de Firefox. L’application Web servira à valider la mise en œuvre de chaque fonction. Le code source doit aussi être accessible en ligne (p.ex. Github).

Vous devez fournir un diagramme qui indique les technologies que vous avez utilisées pour bâtir votre application (langage, base de données, rédaction de scripts, etc..).

Pour vous aider, la CFP a établi une structure de données JSON.

**Liste des fonctions et éléments associés**

|  |  |  |
| --- | --- | --- |
| **Fonctions évaluées que vous devez intégrer dans votre application** | **Points  (1,2,3,5,8,13)** | **Intégrée  (oui/non)** |
| Afficher, ajouter, retirer et mettre à jour le profil du chercheur d’emploi (voir JSON) | 13 |  |
| Afficher, ajouter, retirer et mettre à jour la description d’emploi (voir JSON) | 13 |  |
| Afficher les possibilités d’emploi sur une carte selon l’adresse des lieux de travail. | 8 |  |
| Afficher les chercheurs d’emploi sur une carte, selon leur adresse. | 8 |  |
| Repérer les emplois les plus proches d’un chercheur d’emploi donné. | 5 |  |
| Gestion du compte d’un chercheur d’emploi (capacité de s’identifier, d’ajouter et de modifier) :   * Information du chercheur d’emploi * Compétences du chercheur d’emploi | 8 |  |
| Possibilité pour le chercheur d’emploi de télécharger son curriculum vitae dans son profil. | 5 |  |
| Fonctions de recherche   * Chercheur d’emploi * Profils * Description d’emploi | 8 |  |
| Parcourir  Emplois, curriculum vitae, chercheurs d’emploi | 8 |  |
| Les listes d’emplois doivent permettre de trier et de filtrer par date de l’affichage, salaire, etc. | 5 |  |
| Accessibilité : l’application doit être compatible avec WCAG 2.0 niveau A | 8 |  |
| Jumelage d’emploi   * Repérer des chercheurs d’emploi pouvant être jumelés à chaque offre d’emploi. * Repérer les possibilités d’emploi pour chaque chercheur d’emploi. | 13 |  |
| Possibilité de partager les descriptions d’emploi sur les médias sociaux. | 5 |  |
| Authentification des utilisateurs du système (chercheurs d’emploi et gestionnaire d’embauche) : nom d’utilisateur et mot de passe | 8 |  |
| L’authentification doit contenir une fonction de type Captcha | 5 |  |
| Fonction de récupération de mot de passe autonome (libre‑service) | 5 |  |
| L’interface utilisateur doit etre disponible dans deux langues officielles (français/anglais) | 8 |  |
| Diagrame des technologies utilisées. | 8 |  |
| Qualité du code : Jusqu’a 8 points en fonction des commentaires, de la documentation, des noms des variables, la facilité de lecture | 8 |  |
| Vidéo récapitulative de 5 min sur l’application disponible en ligne | 5 |  |

Structure des données :

Candidate :

[

{

**"id"**:Integer, //Identifiant unique

**"first\_name "**:"String", //Prénom du candidat

**"last\_name"** :"String", //nom de famille du candidat

"**universe\_permits**":Tableau Integer de taille variable, //Tableau qui démontre les Univers que ce candidat est permis d’appliquer.

**"willing\_to\_do\_overtime"**:Bool, //Le candidat veut faire des heures supplémentaires

**"willing\_to\_relocate\_to\_other\_universe"**:Bool, //Le candidat veut relocaliser

**"home\_address"**:"String", //Adresse postale en String non formaté

**"education\_level"** :Integer, //Identifiant unique du niveau de scolarité

**"skills"** : "Tableau de String" //tout les expertise du candidat

}

]

Emploi :

[

{

**"id"**:Integer, //Identifiant unique

**"description\_en"**:"String", //Description en anglais

**"description\_fr"**:"String", //Description en français

**"salary"**:Interger, //Salaire de l’emploi

**"work\_location"**:"String", //Adresse postale en String non formaté

**"work\_universe"**: Integer, //Identifiant de l’Univers

**"required\_skill"**: "Tableau de String", //Liste de tous les expertises requises pour l’emploi

**"date\_posted"**:"Date",//Date

**"education\_level"**:Integer //Niveau minimale d’éducation demandé pour le poste

}

]

Univers

[

{

**"id"**:Integer, // Identifiant Unique

**"Universe"**:"String" // Nom de l’Univers

}

]

Éducation

[

{

**"id"**:Integer, // Identifiant Unique

**"Education\_Level "**:"String" // Nom du niveau d’éducationl

}

]

Résultats attendus

* Application URL accessible en ligne
* Code source de l’application
* Création d’un diagramme démontrant les technologies utilisées.
* Liste des fonctionnalités terminées
* Vidéo récapitulative de 5 min sur l’application