

## Feuille de temps

Vous devez concevoir un logiciel de feuille de temps, en version simplifiée. Ce logiciel permet à des employés de spécifier les heures travaillées et produire la facturation liée au temps travaillé.

### **La base de données**

Un script de création de base de données pour SQLite vous sera fourni par courriel. La base de données n'a qu'une seule table et permet de contenir l'information sur les entrées de temps d'un employé. Voici les champs et leur utilité :

- `matricule` : Une chaîne de caractères représentant le matricule d'un employé.
- `code_de_projet` : Une chaîne de caractères représentant un code de projet. Ce code permet d'identifier sur quel projet le travail a été effectué.
- `date` : Une date correspondant à la date où le temps a été travaillé. La date est un champ TEXT en format ISO8601 (voir la documentation de SQLite).
- `duree` : Le nombre de minutes travaillées par cet employé durant la journée sur ce code de projet.

### **Les routes**

Plusieurs routes doivent être supportées par votre application web. Voici le détail des exigences pour chaque route. Vous avez l'entière liberté de disposer les éléments demandés à votre guise. L'esthétique du site web est importante. À cet effet, vous êtes autorisés à utiliser des gabarits CSS provenant du web, dans la mesure où vous respectez leur licence d'utilisation.

En plus de celles spécifiées dans l'énoncé, vous pouvez ajouter toutes les routes que vous jugez nécessaire à la réalisation du travail.

#### **GET /**

Cette route correspond à la page d'accueil du site. La page doit contenir un champ texte permettant à l'utilisateur d'y spécifier son numéro de matricule. Le numéro de matricule doit avoir le format suivant : XXX-99. En cas d'erreur, un message doit être affiché à l'utilisateur. En cas de succès, on redirige l'utilisateur à la route `GET /<matricule>/<date_du_jour>`

Ex : `GET /BER-98/2018-02-09`

#### **GET /<matricule>/<date\_du\_jour>**

Cette route correspond à la page permettant d'ajouter du temps sur des codes de projet pour la journée spécifiée dans l'URL. La date du jour doit être en format ISO8601 et le matricule doit avoir le format spécifié précédemment.

S'il existe des données dans la base données pour ce matricule et cette date, les données doivent être affichées à l'utilisateur. L'utilisateur devrait pouvoir modifier les données précédemment fournies au logiciel (modifier et supprimer).

Lors d'un ajout ou d'une modification de données, tous les champs doivent être validés par l'application.

Sur la page, il doit y avoir des boutons pour naviguer au jour suivant et au jour précédent (attention

aux années bissextiles).

## **GET /<matricule>/overview/<mois>**

Cette page affiche un résumé des heures travaillées pour un mois en particulier. Le format du mois est AAAA-MM (ex. 2018-02). La page doit afficher un calendrier mensuel et indiquer le total des heures travaillées dans chaque journée du calendrier mensuel.

Sur la page, il doit y avoir des boutons pour naviguer au mois suivant et au mois précédent.

Chaque total d'heures quotidien doit être un lien vers la journée en question.

## **GET /<matricule>**

Cette page doit afficher la liste de tous les mois où ce matricule a déclaré des heures travaillées. Chaque élément de la liste doit être un lien vers le sommaire du mois en question.

## ***Technologies***

Dans le front-end, vous **devez** utiliser les technologies suivantes :

- HTML 5
- CSS 3

Toujours dans le front-end, vous **pouvez** utiliser les technologies suivantes au besoin :

- Javascript
- JQuery
- Underscore.js
- Bootstrap

Dans le back-end, vous ne devez utiliser que les technologies vues en classe, ce qui inclut :

- Python 3
- Flask 0.12
- SQLite 3
- Jinja2

Le code Python doit respecter PEP8. Tous les fichiers sources doivent être encodés en UTF8.

Il est nécessaire d'utiliser le patron POST-REDIRECT-GET partout où c'est pertinent.

## ***Remise***

Le travail peut être fait seul ou en équipe de 2 personnes. Le répertoire de travail contenant les fichiers doit être archivé dans un fichier zip et nommé selon les codes permanents des auteurs. L'archive doit être remise par Moodle avant le 16 mars 2018 à 9h30. Aucun retard ne sera accepté et une pénalité sera appliquée pour une archive non conforme sans les codes permanents.

Si vous voulez travailler en équipe, vous devez mettre votre logiciel sous gestion de sources dans un dépôt privé git (ex. github, bitbucket) et m'en donner un accès en lecture. Cette mesure sert à m'assurer que les deux coéquipiers ont partagé l'effort équitablement. Un coéquipier n'ayant pas offert une participation significative n'aura pas les points pour le travail.

## ***Pondération***

Fonctionnalités : 55%

Respect des exigences et standard du web : 25%

Respect de PEP8 et qualité du code (taille des fonctions, qualité de la nomenclature, etc.) : 10%

Esthétique du site web : 10%