TPTD FLASK - Novembre 2020

L’objectif de ce TPTD est d’explorer et mettre en œuvre quelques concepts pour ébaucher le développement d’une webapp.

**Le contexte** : l’objectif est de développer une webapp permettant de stocker et utiliser les données des publications des enseignants-chercheurs ESIEE. Parmi les questions qui peuvent se poser : afficher toutes les publis d’un enseignant-chercheur, d’un laboratoire, télécharger les listes correspondantes, afficher tracer l’évolution du nombre de publis au cours du temps (pour un enseignant-chercheur ou un labo), des données croisées, etc etc

Les routes pourraient également être accessibles en GET avec des paramètres

**Les données et documents** :

* Le cours “Quelques éléments autour de Flask”
* Liste des laboratoires : LIGM, ESYCOM, LISIS, IRG, ESIEE
* Liste des enseignants-chercheurs, avec affectation labo - listeBiblio.xls
* Plusieurs fichiers excel de liste de publis

1 - Consulter le cours, ou internet, et créer un squelette d’application, sous la forme “package” python avec un fichier config.py, un \_\_init\_\_.py, un fichier views.py, un forms.py, un models.py, un fichier run.py qui sera à lancer via “python run.py”

1. votre premier jet permettra d’afficher un “hello world” (run, \_\_init\_\_, views, config)

2 - votre second jet permettra d’afficher un écran de login et utilisera flask\_login (on ajoute forms.py, éventuellement une template)

3 - Base de données simple

1. dans un troisième temps, vous intégrerez une base de données simple, avec une classe Auteur comportant les champs nom, prenom, labo, debut, fin (d’activité). Cela vous conduira à définir la structure de la base de données, intégrer les paramètres dans le fichier de configuration. Il vous faudra créer la base de données physique par ailleurs.
2. vous écrirez un petit programme dbcreate.py pour créer la base, et dbinit.py permettant d’initialiser la base de données à partir du fichier listeBiblio.xls
3. vous ajouterez une route /affichEC/<EC> qui vous permettra d’afficher les infos sur l’auteur en question

4 - Menu déroulant

1. créer une route /selectEC permettant d’afficher un menu déroulant permettant de sélectionner un nom et redirigeant vers la page /affichEC/<EC> affiche les données sur la personne

5 - Discuter de la structure d’une base de données relationnelles qui mettra en correspondance laboratoire, auteurs, publications

1. créer la base de données correspondante (au moins 3 tables). Consulter les fichiers de publis pour lister les champs à retenir.
2. vous compléterez dbinit.py pour initialiser la base à partir de listeBibio.xls et des fichiers de publis.

6 - mettre à jour la page de redirection de /selectEC pour afficher des infos sur les publis de la personne sélectionnée. Prévoir cette page de manière à ce qu’elle puisse être consultée directement, avec passage des paramètres via l’URL.

6’ - permettre de télécharger les résulats (via un bouton ou un paramètre dans l’URL)

7 - créer une route /selectLabo permettant d’afficher un menu déroulant permettant de sélectionner un labo et redirigeant vers une page qui affiche les publis du labo correspondant

8 - intégrer une route permettant de tracer l’évolution du nombre de publis d’un labo au cours du temps (avec un choix, liste ou tracé sur la page selectLabo)

etc, etc