**TRAVAIL PRATIQUE 2 : REALISATION D’UNE APPLICATIONWEB SIMPLE AVEC PYTHON**

1. **ROLE DE L’APPLICATION WEB**

Application web qui permet d’enregistrer et de mettre a jour les produits dans un entrepôts de ventes, de production, de stockage ou de distribution de produits ménagers, électroménagers et autres.

1. **Principes de fonctionnement**

Pour utiliser l’application web on procède par des opérations simples qui peuvent être énumérées comme suit :

1. Lancer l’application avec la commande python suivante : python flask\_produit.py. Mais avant vérifier que l’on soit dans le répertoire de travail :

templates=getcwd() # recuper le repertoire de travail

app = Flask(\_\_name\_\_, *template\_folder*=templates+"\\templates")

1. Ouvrir l’application en tapant dans un navigateur web l’adresse : http://127.0.0.1:5001/
2. Une fois l’application ouverte, elle affiche la liste des produits en stock avec les champs :

* *ID\_Code* : C’est la clé primaire pour la table sqlite3
* *Type*: à choisir parmi une liste entre les valeurs (Appareils, Outils, Vestimentaire, Meubles, Autres)
* *Catégorie*: Choisir parmi une liste (Téléphone, Ordinateur, Tablette, Objet de Cuisine, Meuble, Autre)
* *Nom*: Le nom du produit
* *Marque*: La quantité du produit
* *La Quantité :*  La quantité peut être positive dans le cas ou on ajoute un produit dans le stock, ou négative si on retire du stock.

1. L’utilisateur est invite à compléter les autres champs pour renseigner tous les éléments du produit.
2. Une fois tous les champ remplis, on appuis sur le bouton enregistrer et la liste est automatiquement mise à jour. En image ci-dessus nous avons une apparence de l’interface graphique de notre application. Sur la page d’accueil on peut clairement remarquer que l’interface est repartie en deux grands blocs : La liste des produits dans la base de données au-dessus et en dessous le formulaire pour enregistrer ou supprimer un produit de la base de données.
3. Le serveur de l’application web écoute continuellement le port 5001 tant que l’application Flaks continue de tourner.
4. Pour arrêter le serveur Flask a la fin des opérations d’interaction avec la base de données, on peut taper dans l’invite de commande la commande : ctrl + C

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

***Figure : Interface de l’application web***

1. **Les scenarios d’utilisation** :

* Pour ajouter un produit on saisit une quantité positive.
* Pour retirer un produit on saisit une quantité négative, ceci si le produit existe déjà dans la table, mais a condition que la valeur absolue de la quantité saisie soit inferieure a la quantité effective dans la table.
* Si on entre une quantité positive, le produit est enregistré comme nouveau produit dans la base de données s’il n’existe pas déjà et la page est mise à jour.
* Si un produit existe déjà et que la quantité est positive, la table est mise à jour en additionnant a la quantité déjà existante.
* Si un produit existe déjà et que la quantité saisie est négative (cas ou on veut retirer une certaine quantité), la table est mise à jour en soustrayant a la quantité déjà existante.
* Si la quantité saisie est négative et si le produit n’existe pas déjà dans la base de donnes : Le produit n’est pas enregistré dans la base de données.
* Si la quantité saisie est négative et si le produit n’existe pas déjà dans la base de donnes : Le produit n’est enregistré que si la valeur absolue de la quantité saisie est inferieure a la valeur dans la table du produit en question.
* Supprimer un article de la base de données.

1. **Développement de l’application**

Le développement de l’application prend en compte les langages ***python, HTML, JavaScript***

1. Nous commençons par notre environnement de développement a l’intérieur du répertoire de travail.
2. A l’intérieur du répertoire de travail on crée un répertoire ‘‘templates’’.
3. Dans le dossier ‘’templates’’ on crée le fichier HTML ‘’ produits.html ‘’.
4. Dans le répertoire courant nous avons les fichiers :

* flask\_produit.py : contient le Code python pour le déploiement du server flask
* Stock.db : qui contient la base de données des produits

1. **Explication de mon code**
2. **Le Code python : flask\_produits.py**

Le code utilise les modules :

* Flask : pour déployer le serveur flask.
* Sqlite3 : pour la gestion de la base de données.
* Jinja2 : pour la programmation des templates HTML
* Os : pour les repertoires

1. **Le code HTML : *produits.html***

* Style CSS : pour le rendu de la page
* Le JavaScript pour la programmation de la page : on va programmer les éléments de la page afin qu’ils puissent mettre a jour les quantités négatives, les débordements de quantités.

1. **Remédiations et améliorations futur**

Pour les futures améliorations de mon application, je pourrais travailler a :

1. Permettre à l’utilisateur de créer lui-même, d’ajouter et de modifier les champs dans la table de la liste des objets.
2. Permettre d’utiliser plusieurs sources d’entrer d’information en dehors du clavier, comme les lecteurs de code-bar, les codes QR pour capter les informations à stocker dans la base de données.
3. Travailler sur l’interface graphique de façon à ajouter des boutons pour sélectionner directement une ligne de la table et la modifier.
4. Travailler sur la gestion des messages d’erreurs.
5. Ajouter des fonctionnalités d’imprimer la liste en en fichier csv ou pdf.