

PROJET 8

Lien GitHub: <https://github.com/Tony380/Projet8>

Lien Trello: <https://trello.com/b/RhX1abT3/projet-8>

Lien Heroku: <https://purbeurre2020.herokuapp.com>

CADRE DE DEPART:

La startup Pur Beurre, souhaite développer une plateforme web à destination de ses clients qui permettra à quiconque de trouver un substitut sain à un aliment.

Après avoir créé un tableau Trello afin de définir les objectifs du projet et les étapes à suivre suivant une méthodologie agile, j'ai créé un repo Git, un environnement de travail pour le développement de l'application et une base de données PostgreSQL.

DJANGO:

La première étape a été la création d'un projet Django nommé « purbeurre » et la modification de certains réglages par défaut comme par exemple la configuration relative à la base de données utilisée pour le projet, PostgreSQL.

LA PAGE D'ACCUEIL :

Pour cette étape, et selon le cahier des charges du client, j'ai intégré au projet les assets fournis par le client ainsi que le thème Bootstrap Creative. Ce thème comprend une page HTML au design Bootstrap Creative avec son fichier CSS et un fichier JQuery. Une fois les fichiers mis en place, j'ai adapté le contenu des fichiers HTML et CSS afin de rendre la page d'accueil conforme aux exigences du client. A noter que nous aurons un fichier « base.html », qui contiendra un entête et un pied de page, et qui sera donc la base de tous les templates de l'application.

L'ESPACE UTILISATEUR :

Cette étape a consisté en la création d'une nouvelle application Django nommée « users ». Cette application contient toutes les fonctionnalités relative à l'utilisateur comme, la création d'un compte, la connexion et la déconnexion de ce compte ainsi que sa page de profil. Les vues, les urls, les templates et les formulaires relatifs ont donc été créés. Il est à noter que Django possède des formulaires et des vues prêts à l'emploi qu'il est possibles d'utiliser ainsi qu'un model User.

LA RECHERCHE DE PRODUITS :

Après la création d'une seconde application nommée « core » et qui comprend toutes les fonctionnalités relatives aux produits et la base de données, j'ai créé les models correspondants à nos tables puis effectué les migrations nécessaires.

J'ai personnalisé une commande Django afin de remplir la base de données en ligne de commande avec manage.py. Pour ce faire, j'ai créé une classe « Command » avec une fonction « handle() »,

qui contient la requête à l'API d'OpenFoodFacts. Après le lancement de cette commande, la base de données est désormais remplie. Enfin, j'ai programmé l'algorithme qui permet de trouver le substitut plus sain d'un produit, ainsi que les templates nécessaires.

L'ENREGISTREMENT DES FAVORIS :

Création de la vue et du template relatifs, permettant à l'utilisateur de sauvegarder des favoris.

L'utilisateur a aussi la possibilité de les supprimer. Cette fonctionnalité fait partie de l'application « core ». Un nouveau model a été créé à cet effet. Nos tables étant reliées entre elles, elles permettent d'assigner un produit et son substitut à un utilisateur.

FINITIONS :

J'ai créé la page mentions légales, indiquant les sources des assets utilisés, ainsi que les fichiers requirements.txt, .gitignore, et le fichier Readme.md. J'ai aussi intégré une gestion des erreurs 404 et 500 avec leurs templates respectifs.

LES TESTS :

Création de tests utilisant Unittest et Selenium et utilisation de Coverage pour avoir un rapport de la couverture des tests, d'un minimum de 80%. Chaque application contient donc un fichier tests.py avec les tests relatifs à ses vues, urls, models, et formulaires.

MISE EN LIGNE SUR HEROKU :

Enfin, pour cet dernière étape, j'ai créé un nouveau projet sur Heroku puis configuré les settings de Django de manière à pouvoir utiliser l'application selon qu'elle se trouve en développement ou en production. Cela inclut entre autres, la modification du chemin vers les fichiers statiques et la mise en place d'un fichier Procfile.

DIFFICULTES RENCONTREES :

Trois difficultés mineures ont été rencontrées au cours du développement de cette application. La prise en main de Django, la sauvegarde des favoris et l'écriture des tests.

Ces difficultés ont été surmontés grâce au cours d'Openclassrooms et de tutoriel sur internet en complément.

CONCLUSION :

La combinaison de Django et Bootstrap permet l'élaboration de sites web sophistiqués, fonctionnels et assez sécurisés.

Une amélioration possible pour ce projet serait par exemple la possibilité pour un utilisateur de personnaliser son compte mais aussi de le supprimer.

Pour la plus grande partie de ce projet, l'application s'en tient au cahier des charges.