

# DÉMARCHE DE CRÉATION

## **Présentation**

L'objectif de ce projet est de récupérer des données mises à disposition par l'API [OpenFoodFacts](#) afin de développer une solution web fullstack avec le framework backend Django.

Le client est PurBeurre comme au projet 5.

## **Préparation**

Avant de commencer j'ai pris connaissance des consignes via :

- 1) la description du projet par Openclassrooms
- 2) la lecture du cahier des charges
- 3) des questions à mon mentor pour bien m'assurer d'avoir compris ce qui était attendu
- 4) une relecture du cahier des charges et des documents annexes

J'ai ensuite créé le trello, téléchargé les images et polices de caractères.  
J'ai après cela suivi un tutoriel et lu le début de la documentation officielle de Django.

## **Structuration**

J'ai créé le projet dans un environnement virtuel avec une application pour la 'navigation' et une application pour gestion des propriétés de l'utilisateur, installé Postgresql et configuré l'accès à la base de données dans Django. J'ai ensuite importé dans Postgresql ma base de données de P5. Je n'ai gardé que 3 tables puisque Django gère les utilisateurs.

L'étape suivante a été de remplir la base de données. J'ai donc utilisé deux scripts (un script de requêtage de l'API OpenFoodFacts et un script d'insertion) pour la remplir.

## **Mise en page**

J'ai créé les urls, les vues et les templates Django de chaque page en incluant les images via le langage de template.

Une fois qu'une vue retournait une page il a été possible de tester la réponse HTTP.

J'ai ensuite ajouté les formulaire HTML autour des balises d'input.

## **Exploitation de la base de données**

J'ai créé mes classes de formulaire, enrichi mes urls et mes vues en fonction des besoin en faisant appel aux méthodes statiques d'une classe manipulation des données.

Le fait de retourner un contexte aux templates a permis de rendre le site interactif.

Pour tout ce qui concerne le comportement de l'utilisateur (connexion, déconnexion, inscription, like, unlike) j'ai utilisé la base de données du système d'administration de Django.

## **Tests**

J'ai complété mes tests avec Selenium, Coverage (81 %) et Flake8.

## **Déploiement**

J'avais créé un compte Heroku au P7 et le déploiement a été plus facile. Je me suis appuyé sur le cours d'introduction à Django d'Openclassrooms ainsi que sur la documentation d'Heroku.

J'ai corrigé les erreurs dues aux chemins manquants vers les fichiers statiques et lancé les tests Selenium.

## **Difficultés rencontrées**

- 1) Se familiariser avec Django
- 2) Les forms à la place d' Ajax
- 3) La nécessité d'adapter certains tests (pas d'objet request parfois)
- 4) La compréhension du langage de templates
- 5) La configuration de settings.py

Le projet est suivi sur [Trello](https://trello.com/) et le produit final est accessible à l'adresse <https://beurrepur.herokuapp.com/>

Lien Github : <https://github.com/nojovent/Projet8>