

DÉMARCHE PROJET 8 - APPLICATION PURBEURRE

Avec ce document je vais expliquer ma démarche pour la réalisation d'une application web basée sur les données d'Open Food Facts.

- [Repository GitHub](#)
- [Tableau Trello](#)

1. Étude du cours Django associé

Avant de commencer avec la pratique j'ai étudié le cours Django grâce auquel j'ai eu la possibilité de comprendre de manière généraliste la gestion d'un projet sous ce framework. J'ai compris le modèle MVT et les fonctions des différents composants, ainsi que la différenciation entre projet et application.

2. Organisation du travail

J'ai donc initialisé un tableau Trello en décomposant le projet à réaliser en sprints dans le cadre d'une méthodologie de travail Agile. Ici on peut le voir

3. Démarrage projet Purbeurre

J'ai d'abord créé un environnement virtuel et téléchargé les différentes librairies permettant de travailler sur ce projet, par exemple `psycopg2` pour travailler avec PostgreSQL.

J'ai donc effectué les changements nécessaires sur le fichier de configuration `settings.py`.

4. Démarrage application Purbeurre

Je suis passé ensuite au démarrage de l'application web en soi. Les premières choses pour moi étaient le parcours utilisateur, donc définir les routes dans le fichier `urls.py`, la gestion du compte utilisateur etc, et la création de la base de données. Pour la création de la base, j'ai travaillé sur le fichier `models.py` en créant les tables avec leurs propres données.

5. Site web – corps principal

À ce point je me suis intéressé à la partie front-end de l'application. J'ai commencé par coder l'header et ensuite le vrai corps de la page principale, avec l'importation de Bootstrap, des fonts et autres éléments graphiques indiqués sur le cahier des charges, les formulaires de recherche des produits et d'authentification des clients et enfin le footer avec les mentions légales et les contacts.

6. Formulaires

Pour un correct fonctionnement de l'inscription ou login des utilisateurs, il fallait maintenant coder la partie back-end liée à la gestion du compte client et plus précisément l'envoi de requêtes sécurisées et respectant certains critères (ex. mot de passe assez longue etc.).

J'ai codé la situation où l'utilisateur n'a pas un compte et il doit le créer et celle où il doit effectuer le login avec ses données.

7. Espace personnel client

Il était temps de coder l'interface de l'espace client en respectant les consignes du cahier des charges. J'ai créé donc le message de bienvenue, l'icône de carotte pour l'accès aux aliments sauvegardés et le bouton de déconnexion pour terminer la session courante et revenir à la homepage.

8. Recherche produits

À travers l'API d'Open Food Facts j'ai rempli la base de données et défini la recherche des produits en faisant apparaître les résultats en ordre de propriétés nutritionnelles grâce à leur indice *nutriscore* allant de A jusqu'à E.

Ensuite j'ai codé la sauvegarde d'un produit dans la base de données personnelle de l'utilisateur connecté et la page des détails d'un produit spécifique avec ses propriétés alimentaires et le lien vers la fiche détaillée sur Open Food Facts.

9. Peaufinage site web

En dernier je me suis intéressé aux détails: j'ai terminé le codage des pages de recherche produits et leur affichage, des éléments graphiques et de la mise en forme selon les media queries sélectionnés.

10. Tests

- Test des fonctionnalités: recherche des produits, inscription ou connexion à son propre espace utilisateur, sauvegarde d'un produit et connexion à sa base de données personnelle, déconnexion et retour à la homepage, parcours utilisateur, urls des pages web.
- Cross browser testing: j'ai testé le site web sur différents navigateurs web, notamment Chrome, Opera et Firefox pour m'assurer du correct fonctionnement de l'application.
- Test responsive: j'ai simulé la consultation de l'application sur différents écrans et dispositifs grâce aux outils à disposition sur les browsers, outre que grâce au magnifique site web <http://www.responsinator.com/>
- Test des composants avec Pytest: dans le dossier 'tests' j'ai écrit des tests concernant la création des tables dans la base des données, la requête http à travers l'API d'Open Food Facts et l'affichage des différentes vues de l'application (remplissage formulaires, déconnexion utilisateur etc.).

11. Déploiement

J'ai enfin déployé mon application sur Django en suivant la partie du cours dédiée à ce sujet.

Difficultés rencontrées

J'ai eu des difficultés liées à la compréhension du modèle MVT dans un projet Django et il m'a fallu du temps aussi pour assimiler le concept de l'effectuation des migrations pour l'enregistrement des changements.

La documentation Django, très bien écrite, m'a beaucoup aidé à comprendre ces aspects et plus en général dans la résolution des problèmes en phase de réalisation projet.