**Foodflow**

### Fonctionnalité qui sera prise en main:

**Alexis:**

**Fonctionnalité réalisée:**

J’ai créé la stratégie de connexion en me basant sur la librairie passport.js. Elle utilise les JSON Web Tokens (JWT) pour sécuriser l'accès à notre application en vérifiant l’identité des utilisateurs lors de leur connexion.

**Plus grand défi:**

Le plus grand défi a été d’assurer la sécurité des mots de passe en utilisant des hash avant de stocker les mots de passe de manière sécuritaire dans la base de données. De plus, le processus de signature des jwt token a rendu l’implémentation plus difficile et a nécessité plus de temps pour être complètement fonctionnel. Au départ, j’avais seulement ajouter un jwt token qui contenait les informations de l’utilisateur, mais en cherchant d’avantage, j’ai réalisé que la meilleure pratique pour utiliser cette technologie était d’ajouter un second token qui permet de rafraîchir le premier.

**Thomas:**

**Fonctionnalité réalisée (ajouter produit au site):**

J’ai créé un ProductController et un ProductDao qui s’assure de faire les bonnes requêtes sur la base de donnée pour ajouter dans les tables prix, catégories et produits les bonne informations pour pouvoir avoir un produit complet créer dans la base de donnée. J’ai ensuite fait la bonne requête pour pouvoir aller chercher les bonnes informations qui sont affichées sur le site.

**Plus grand défi:**

La plus grande difficulté à été de faire la requête pour créer et retourner chercher les catégories pour les produits vu la présence d’une table de liaison.

Ce code se trouve dans le dossier api/src/1/Controller et le dossier api/src/1/DAO

### Fonctionnalité qui sera prise en main:

**Alexis:**

**Prochaine fonctionnalité:**

La prochaine fonctionnalité que je vais implémenter sera le walletController. Il sera nécessaire pour afficher le solde de l’utilisateur, lui permettre d’ajouter des fonds à son compte et lui permettre d’en retirer. Ce contrôleur nécessite un DAO qui se nomme walletDAO et il communique avec la base de données en utilisant des requêtes SQL.

**Défi:**

Le plus gros défi de cette tâche sera d'après moi de relier le DAO avec le contrôleur et relier les routes d’une manière sécuritaire en utilisant l'authentification JWT pour sécuriser les soldes. Cela inclut qu’un utilisateur peut seulement ajouter ou enlever des fonds sur son wallet.

**Thomas:**

**Prochaine fonctionnalité:**

La prochaine fonctionnalité que je compte implanter est de faire le Controller et le DAO pour les transactions afin que l’on puisse pouvoir faire des achats sur le site. Je devrais bien sûr faire les bonnes requêtes pour ajouter une transaction, aller chercher les infos d’une transaction déjà existante et supprimer une transaction si jamais celle-ci est annulée par l’administrateur par exemple. Je devrais aussi m’assurer que les routes avec Express seront bien linker avec le RootController.

**Défi:**

Je crois que le plus gros défi sera de linker les transactions au wallet puisque nous devons nous assurer que le client qui fait la transaction à suffisamment de fonds pour pouvoir faire la transaction et que si la transaction s’effectue, il faut enlever les fonds dans le Wallet du client. Nous n'avons pas vraiment encore été confronté à ce genre de situation à date dans le projet donc ce sera clairement un beau défi.

**Progression:**

**Nous pensons avoir 65% du projet de compléter au moment de la publication de ce document.**

**À peu près 75% de l’api est compléter et à peu près 55% du front-end en React est complété.**