

PokeShop

<https://github.com/gorilla/mux#examples>

<https://pokeapi.co>

Sujet

Projet 1 :Thème Vente de produits Vous allez devoir développer une application mobile de vente de produits. Ce sera un projet à thème libre, càd que tant que vous restez dans votre thème et respectez les besoin de l'application, il y aura pas de pénalité (attention à ne pas vendre des produits borderline qui pour le coup sera pénalisé). L'application aura besoin: Une api REST pour les données (langage non imposé). Une base de données ou des fichiers JSON (ici également non imposés). Une page d'accueil, qui présentera l'application Un Système de listing des produits Une fiche détaillée des produits Une interface de Login qui nous permettra d'ajouter et/ou supprimer des produits. Créer/supprimer des utilisateurs. Bien sur ce projet est évolutif, vous pouvez ajouter de nouvelles features.

Interprétation

L'idée principale est de faire un shop permettant d'acheter des pokémon avec back-end en go se basant sur la PokeAPI et un front en Flutter. Les pokémon seront achetables en pokédollars, il y aura aussi une notion de stock, d'inventaire.

L'api contient les particularités suivantes :

- Informations détaillées des pokémons (basé sur la PokeAPI)
- route de type et de counters
- type
- attaque
- points de vie
- move signature

Et d'ajouter des informations tel que:

- le prix d'un pokémon
- son nombre
- son type

Technologies

Le Front est en Flutter (dart + flutter) comme imposé par le sujet.

Nous avons décidé pour le backend de partir sur le langage de google go (ou golang). Ce choix est motivé par les raisons suivantes :

- Volonté d'apprendre le go
- facilité de la mise en place d'api avec go
- tendance du langage
- nombreux frameworks d'api et micro api existants

Pour la base de données, pour l'instant nous n'en utilisons pas. Cependant si le projet évolue suffisamment il y a 2 options :

- soit on stocke les données des prix et de l'inventaire dans un JSON.
- Soit on crée une bdd dans une base en sql ou document (mongoDB ou Sql-like)

Itérations de développement

Les différentes versions :

- 1) Afficher tous les Pokémons avec le stock restant
 - a) front flutter :
 - i) image des pokémons + nom
 - ii) détails des pokémons + nombre en stock
 - b) back end go :
 - i) mise en place du stock
 - ii) envoyer que les informations des pokémons en stock (limiter le nombre d'id possible de pokémon à un nombre genre 200)
- 2) Acheter des Pokémons et faire diminuer un stock
 - a) front flutter :
 - i) interface pour acheter un pokémon
 - b) back end go :
 - i) prix du pokémon par id
 - ii) Diminution du stock en fonction des achats
- 3) Création des comptes utilisateurs
 - a) front flutter :
 - i) interface de connection
 - ii) page d'inscription
 - iii) page de paramètre (dédiée à l'utilisateur)
 - iv) stock et panier user
 - v) solde en pokédollars de l'user
 - b) back end go :
 - i) ajout d'une bdd pour stocker les user
 - ii) gestion inscription/connection des users
 - iii) route pour modification des paramètres des users
 - iv) Stock et panier user
 - v) gestion du solde en pokédollars de l'user
- 4) Pannel admin depuis l'application pour manager les stocks et l'argent des comptes
 - a) front flutter :
 - i) pannel admin
 - ii) gestion des users

- iii) gestion des soldes des users
- iv) gestion des stock des users
- b) back end go :
 - i) compte admin afin d'autoriser ces manipulations
 - ii) Ajout d'une connexion à l'API par JWT ? (very optional)