

## **PARTIE 1**

### **### Travaux Pratiques : Back-end pour une Boutique de Chaussures**

#### **\*\*Introduction :\*\***

L'objectif de ces travaux pratiques est de créer un serveur back-end en utilisant Express.js pour une boutique de chaussures en ligne. Vous allez mettre en place une API RESTful qui permettra aux utilisateurs de gérer les articles de la boutique, y compris l'enregistrement, la récupération, la modification, la suppression, le "like" et le tri par "like". Chaque article sera caractérisé par des données telles que le nom, le prix, la description, la photo et le nombre de likes.

Vous pouvez utiliser la structure mvc du chapitre 04-MVC

#### **1. \*\*Configuration du Serveur Express :\*\***

- Créez un serveur Express.js qui écoutera sur un port de votre choix (par exemple, 3000) ou utiliser le serveur du dernier cours;

#### **2. \*\*Création d'Articles de Chaussures :\*\***

- Mettez en place une route permettant de créer de nouveaux articles de chaussures. Chaque article doit contenir les données suivantes :

- Nom (name)
- Prix (price)
- Description (description)
- Image (picture)
- Nombre de likes (like)

#### **3. \*\*Récupération de Tous les Articles :\*\***

- Configurez une route pour récupérer la liste de tous les articles de chaussures disponibles dans la boutique.

#### **4. \*\*Récupération d'un Article Unique :\*\***

- Mettez en place une route pour récupérer les détails d'un article de chaussures spécifique en utilisant son identifiant unique.

## **5. \*\*Modification d'un Article :\*\***

- Configurez une route qui permettra de mettre à jour les détails d'un article de chaussures en utilisant son identifiant unique.

## **6. \*\*Suppression d'un Article :\*\***

- Mettez en place une route pour supprimer un article de chaussures en utilisant son identifiant unique.

## **7. \*\*Ajout de Likes :\*\***

- Créez une route qui permettra d'ajouter un like à un article de chaussures en utilisant son identifiant unique.

## **8. \*\*Tri par Likes :\*\***

- Configurez une route pour trier les articles de la boutique par nombre de likes, en affichant les articles les plus populaires en premier.

**\*\*Données Attendues pour un Article de Chaussures :\*\***

- Nom (name) : Le nom de la chaussure (chaîne de caractères).
- Prix (price) : Le prix de la chaussure (nombre).
- Description (description) : Une description de la chaussure (chaîne de caractères).
- Image (picture) : URL de l'image de la chaussure (chaîne de caractères).
- Nombre de likes (like) : Le nombre de likes de la chaussure (nombre entier).

**\*\*Consignes Générales :\*\***

- Assurez-vous que votre code est propre, bien commenté et respecte les bonnes pratiques de développement.
- Testez chaque fonctionnalité de votre API à l'aide d'un outil tel que Postman

## **PARTIE 2**

Développement d'une application mobile native

Fonctionnalités requises

### **Écran d'accueil**

Sur cet écran, vous devez afficher la liste complète des produits disponibles dans votre boutique, avec les images de chaque produit visibles.

### **Détails**

Un écran de détails doit être mis en place. Lorsque l'utilisateur clique sur un produit, il doit être redirigé vers une page affichant toutes les informations détaillées de ce produit.

### **Ajout d'un produit**

Il doit être possible d'ajouter un nouveau produit à la boutique.

### **Suppression**

Les utilisateurs doivent avoir la possibilité de supprimer un produit de la boutique.

### **Mise à jour**

La mise à jour des informations d'un produit doit être autorisée, permettant ainsi la modification de ces informations.

### **J'aime (Like)**

Les utilisateurs doivent pouvoir "aimer" un produit et voir le nombre total de "likes" accumulés pour ce produit.

Aidez-vous, travailler en équipe, un développeur ne travaille jamais seul, bon courage.