TP Flutter N°03

SmartShop : Navigation et Interaction

Objectif général

Mettre en œuvre la navigation entre plusieurs interfaces Flutter et gérer la communication entre les écrans, en ajoutant de l'interactivité (boutons, états, passage de paramètres).

Ce TP poursuit le développement de l'application SmartShop, réalisée lors du TP2.

Rappel de la structure du projet

```
lib/
 ├── main.dart
 ├── pages/
 │     ├── home_page.dart
 │     ├── product_page.dart
 │     ├── profile_page.dart
 └── widgets/
       └── product_card.dart
```

Étape 1 – Navigation entre les pages

Objectif :

Permettre à l'utilisateur de passer :

- de la page d'accueil (HomePage) → à la page de détails produit (ProductPage)
- de la page d'accueil → à la page profil (ProfilePage)
- puis de revenir avec la flèche de retour ← .

Indices :

- Utilisez :

```
Navigator.push(
  context,
  MaterialPageRoute(builder: (context) => const ProfilePage()),
);
```

- Pour revenir à la page précédente :

```
Navigator.pop(context);
```

- Ajoutez une icône "Profil" dans la AppBar pour accéder au profil.

Étape 2 – Passage de données entre les pages

Objectif :

Lorsqu'on clique sur un produit de la page d'accueil, envoyer ses informations à la page de détails.

Indices :

Modifiez le constructeur de ProductPage pour accepter :

```
final String name;
```

TP Flutter N°03

SmartShop : Navigation et Interaction

```
final String price;
final String description;
final String imagePath;
Dans HomePage, au moment de l'appel Navigator.push, transmettez les valeurs :
ProductPage(
  name: "Smartphone Android Pro",
  price: "2999 DH",
  description: "128 Go, 8 Go RAM, Caméra 108MP",
  imagePath: "assets/products/phone.png",
)
```

Dans ProductPage, affichez ces données via widget.name, widget.price, etc.

Étape 3 – Gestion d'état simple (setState)

Objectif :

Simuler l'ajout au panier et le changement d'état d'un bouton.

Indices :

Dans ProductPage, créez une variable booléenne :

```
bool added = false;
Dans le bouton "Ajouter au panier", faites :
onPressed: () {
  setState(() {
    added = !added;
  });
},
```

Changez le texte du bouton selon l'état :

```
label: Text(added ? "Retirer du panier" : "Ajouter au panier")
```

Affichez un SnackBar pour informer l'utilisateur :

```
ScaffoldMessenger.of(context).showSnackBar(
  SnackBar(content: Text("Produit ajouté au panier !"))
);
```

Étape 4 – Navigation par barre inférieure (BottomNavigationBar)

Objectif :

Créer une navigation globale entre les trois pages via une barre fixe en bas.

Indices :

Utilisez un StatefulWidget nommé MainScreen.

TP Flutter N°03

SmartShop : Navigation et Interaction

Créez une liste :

```
final List<Widget> pages = [HomePage(), ProductPage(...), ProfilePage()];
int index = 0;
Dans le Scaffold, ajoutez :
bottomNavigationBar: BottomNavigationBar(
  currentIndex: index,
  onTap: (i) => setState(() => index = i),
  items: const [
    BottomNavigationBarItem(icon: Icon(Icons.home), label: "Accueil"),
    BottomNavigationBarItem(icon: Icon(Icons.shopping_cart), label: "Produits"),
    BottomNavigationBarItem(icon: Icon(Icons.person), label: "Profil"),
  ],
),
body: pages[index],
```

Cela permet de changer de page sans Navigator.

Étape 5 – Interaction et feedback utilisateur

Objectif :

Améliorer l'expérience utilisateur avec des messages et icônes interactives.

Indices :

- Affichez un message de confirmation quand un produit est ajouté au panier (SnackBar).
- Changez l'icône du bouton selon l'état (Icons.check ou Icons.add_shopping_cart).
- Utilisez un FloatingActionButton pour afficher le nombre d'articles dans le panier (même simulé).
- Essayez de garder le même thème visuel (Colors.teal, Poppins).

Étape 6 – Extension

1. Page "Panier"
   - Créez une page CartPage affichant la liste des produits ajoutés.
   - Simulez un total à payer.
2. Navigation Drawer
   - Ajoutez un Drawer latéral dans la page d'accueil :
     - "Accueil"
     - "Profil"
     - "Paramètres"
3. Thème sombre

- Ajoutez un Switch dans la page "Profil" pour activer un mode sombre :
- themeMode: isDark ? ThemeMode.dark : ThemeMode.light