

# TP 3 — MVVM, ViewModel et Navigation Compose

Pr. Lamia ZIAD

EST Essaouira

## Objectifs

- Comprendre l'architecture MVVM en développement mobile ;
- Manipuler un ViewModel ;
- Utiliser Navigation Compose ;
- Mettre en place un écran liste + un écran détails.

## Exercice 1 — Création d'un ViewModel

Créer un **ViewModel** contenant une liste d'utilisateurs.

**Code à écrire :**

```
class UserViewModel : ViewModel() {  
    val users = listOf("Amine", "Laila", "Yassine", "Salma")  
}
```

## Exercice 2 — Mise en place de la Navigation

Créer deux écrans :

- **ScreenListe** : affiche la liste des utilisateurs ;
- **ScreenDetails** : affiche “Profil de : [nom]”.

Créer un **NavHost** avec les routes suivantes :

- "liste"
- "details/{nom}"

Exemple de navigation (à écrire dans votre code) :

```
navController.navigate("details/" + nom)
```

## Exercice 3 — Interaction Liste → Détails

Lorsque l'utilisateur clique sur un nom dans la liste, naviguer vers l'écran détails.

Exemple de composable liste (à écrire vous-même) :

- une colonne ou `LazyColumn` ;
- un `Text` cliquable pour chaque utilisateur ;
- un appel à `navController.navigate(...)`.

## Exercice 4 — Écran de profil

Créer un composable :

```
@Composable fun ScreenDetails(nom: String)
```

Affichage attendu :

```
"Profil de : " + nom
```

Vous pouvez utiliser :

— Column

— Text

## Travail à Rendre

- Code complet du ViewModel ;
- Code du NavHost ;
- Écran liste et écran détails ;
- Capture d'écran montrant le passage d'un écran à l'autre ;
- Brève explication (5 lignes) du fonctionnement de Navigation Compose.