

# Dossier de Conception – MyNoteApp

compte github  GitHub - jdsgsdhghs/myNoteAp

## 1. Architecture générale

Le projet MyNoteApp est organisé selon une architecture modulaire avec une séparation claire des composants UI, de la logique métier et de la navigation. Le point d'entrée de l'application est le fichier App.js.

## 2. Arborescence du projet

```
mynoteapp/ | ├── App.js           # Configuration de la navigation principale |── index.js           # Point de
lancement de l'application |── app.json       # Configuration de l'application Expo |── package.json    #
Dépendances du projet |── assets/           # Icônes et images |   ├── icon.png |   ├── splash-icon.png |   └──
... |── Components/           # Composants React fonctionnels |   ├── Dashboard.js    # Vue d'accueil avec résumé
des notes |   ├── NoteList.js    # Affiche la liste des notes |   ├── FormList.js     # Formulaire de création ou
modification de note |   ├── ListeDetails.js # Vue détaillée d'une note |   ├── Header.js      # Composant d'en-
tête |   └── styles.css        # Styles CSS globaux
```

## 3. Composants principaux

- Dashboard.js :

Affiche la page principale avec les notes disponibles. Inclut des boutons vers la création de note.

- NoteList.js :

Gère l'affichage des notes sous forme de liste. Reçoit les données via les props.

- FormList.js :

Formulaire permettant d'ajouter ou de modifier une note. Utilise des inputs contrôlés via useState.

- ListeDetails.js :

Affiche les détails complets d'une note sélectionnée depuis NoteList ou Dashboard.

- Header.js :

En-tête personnalisé commun à toutes les vues.

## 4. Navigation

La navigation est gérée via le package @react-navigation/native-stack. Structure des routes :- Accueil : Dashboard- Liste des notes : NoteList- Détail d'une note : ListeDetails- Création / édition : FormList

## 5. États et gestion des données

Les données sont stockées localement en mémoire via des hooks React (useState, useEffect). Aucune base de données n'est utilisée. La gestion d'état est centralisée dans les composants principaux (Dashboard/FormList).

## 6. Logique d'affichage

- Condition d'affichage : affichage dynamique des composants en fonction des données présentes.- Mise à jour des listes : mise à jour immédiate suite à la création/modification d'une note.

