

Application Android - ToDo List

Introduction

Dans le cadre de l'électif PMR, il nous a été demandé de concevoir et développer une application Android permettant la gestion de listes de tâches personnalisées. L'objectif principal était de mettre en œuvre des concepts d'UI (ListView/RecyclerView, Adapters), la persistance de données (SharedPreferences, fichiers JSON via la bibliothèque GSON), et la gestion des préférences utilisateur. Cet exercice a permis de renforcer notre maîtrise du langage Kotlin et des bonnes pratiques de développement mobile.

Analyse et Conception

Analyse des besoins

L'application devait permettre à un utilisateur de :

- Saisir un pseudo (avec suggestions issues de l'historique),
- Gérer plusieurs listes de tâches associées à ce pseudo,
- Ajouter, cocher/décocher, et visualiser les items de chaque liste,
- Accéder à un menu de préférences pour modifier le pseudo ou vider l'historique,
- Persister toutes les données localement (SharedPreferences et fichiers JSON).

Conception technique

Le projet s'appuie sur un diagramme de classes structurant les données autour de trois entités principales :

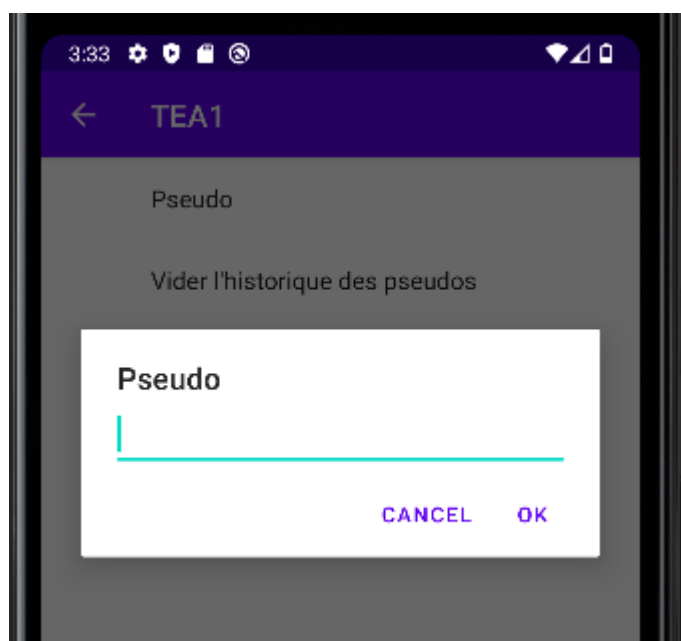
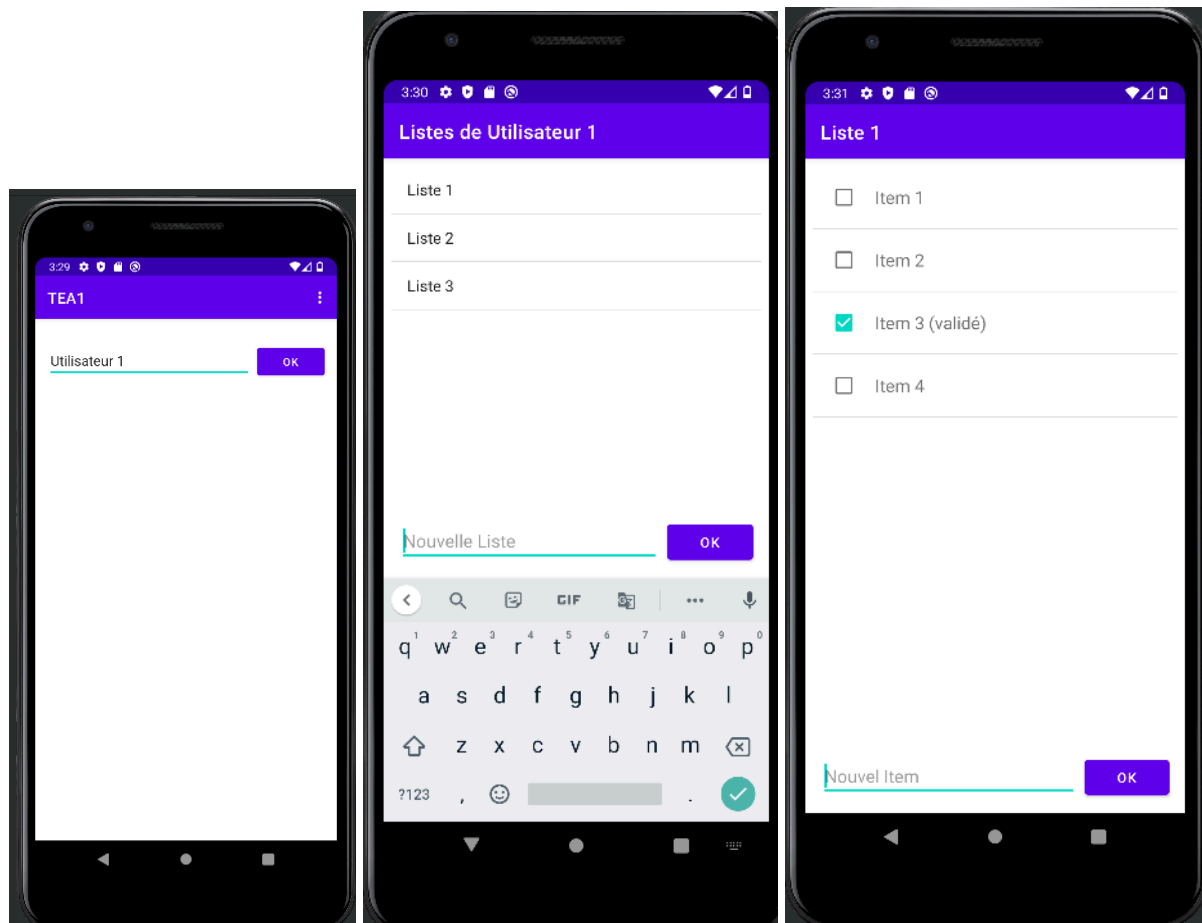
- ItemToDo : représente une tâche avec description et état (fait/non fait)
- ListeToDo : regroupe un ensemble d'items et possède un titre
- ProfilListeToDo : associe un pseudo utilisateur à ses listes de tâches

L'architecture de l'application repose sur plusieurs activités :

- MainActivity : saisie du pseudo, suggestions, accès aux préférences
- SettingsActivity : gestion des préférences et de l'historique
- ChoixListActivity : affichage et création de listes
- ShowListActivity : gestion des items d'une liste

Une activité générique (GenericActivity) a été faite pour factoriser la gestion du menu et des utilitaires communs.

Réalisation



Développement

- UI/UX : Les interfaces ont été réalisées en XML avec ConstraintLayout pour garantir une bonne adaptabilité
- Persistance : Les préférences (pseudo courant, historique) sont stockées via SharedPreferences
- Navigation : L'application utilise des Intent pour passer d'une activité à l'autre, en transmettant les informations nécessaires (pseudo, index de liste...)
- Robustesse : Des contrôles d'erreur et des messages utilisateurs ont été ajoutés

Difficultés rencontrées

- Problèmes de synchronisation Gradle et de dépendances externes (GSON),
- Gestion des conversions de vues (EditText vs AutoCompleteTextView),
- Manipulation des fichiers et gestion des erreurs ainsi que le stockage des pseudos.

Conclusion

Ce projet nous a permis de consolider nos compétences en développement Android natif, en particulier sur la gestion de la persistance locale et la conception d'interfaces. L'application répond à l'ensemble des exigences fonctionnelles et propose une expérience utilisateur correcte.

Perspectives d'amélioration

Plusieurs axes d'amélioration :

- Internationalisation de l'application (plusieurs langues)
- Utilisation d'outils plus adaptés pour une meilleure performance dans le cas de grandes listes
- Ajout de notifications pour les tâches à faire
- Synchronisation cloud pour la sauvegarde multi-appareils
- Amélioration graphique

Bibliographie / Références

- Documentation officielle Android
- Documentation Kotlin : <https://kotlinlang.org/docs/home.html>
- Documentation GSON : <https://github.com/google/gson>
- Stack Overflow ou IA pour la résolution de problèmes spécifiques
- Cours de PMR