

Trabalho 3 – ToDo List com Autenticação Firebase

1. Introdução

Este trabalho consiste na reescrita do aplicativo ToDo List desenvolvido anteriormente, agora integrado com autenticação de usuário utilizando Firebase Authentication. O objetivo principal foi implementar autenticação em nuvem, gerenciamento de estado avançado com Jetpack Compose e organização do código seguindo o padrão arquitetural MVVM.

2. Modelo de Dados

A aplicação utiliza o Room Database para persistência local dos dados. Cada tarefa é representada pela entidade `TodoEntity`, contendo os seguintes atributos:

- `id`: identificador único da tarefa.
- `title`: título da tarefa.
- `description`: descrição opcional.
- `isCompleted`: indica se a tarefa foi concluída.
- `userId`: identificador do usuário autenticado (UID do Firebase).

O campo `userId` garante que cada usuário visualize apenas suas próprias tarefas, evitando compartilhamento de dados entre contas diferentes.

3. Implementação da Persistência

A persistência foi implementada utilizando Room Database. As consultas SQL filtram as tarefas com base no `userId` do usuário autenticado. O Firebase Authentication mantém a sessão ativa e fornece o UID do usuário.

O gerenciamento de estado ocorre através dos ViewModels, que expõem os dados como Flow. A interface observa esses dados utilizando `collectAsState()`, garantindo atualização automática da UI.

4. Melhorias Futuras

- Integração com Firebase Firestore para sincronização em nuvem.

- Implementação de backup automático.
- Melhorias visuais na interface do usuário.
- Implementação de notificações para lembretes.
- Suporte a modo offline com sincronização posterior.

5. Conclusão

O aplicativo atende aos requisitos propostos, implementando autenticação de usuários, persistência de dados separada por conta e gerenciamento de estado utilizando Jetpack Compose. A arquitetura adotada facilita manutenção, escalabilidade e futuras integrações com serviços em nuvem.