

Apresentação Técnica: Integração Firebase no App Flutter

1. Objetivo

Apresentar um aplicativo Flutter simples (sem Firebase) e demonstrar como ele evolui com a integração do Firebase Authentication e Cloud Firestore, utilizando a documentação oficial do Firebase para Flutter como base.

2. Criação do Projeto no Firebase

- Acessar <https://console.firebase.google.com>
- Criar projeto: app_tarefas
- Ativar Authentication > Email/Senha
- Ativar Firestore Database (modo teste)
- Registrar app Android e baixar google-services.json
- Adicionar o JSON em android/app/

3. Configuração do Flutter com Firebase

- Adicionar dependências no pubspec.yaml:
`firebase_core, firebase_auth, cloud_firestore`
- Executar:
`flutter pub add firebase_core firebase_auth cloud_firestore`
- Inicializar Firebase no main.dart com `Firebase.initializeApp()`
- Gerar `firebase_options.dart` usando:
`flutterfire configure`

4. Implementações no Código

Apresentação Técnica: Integração Firebase no App Flutter

- login_page.dart: autenticação com FirebaseAuth
- register_page.dart: criação de usuário com FirebaseAuth
- home_page.dart:
 - leitura de tarefas em tempo real com Firestore
 - filtro por UID do usuário logado
 - adicionar, editar e excluir tarefas com Firestore

5. Estrutura de Dados no Firestore

Cada tarefa é um documento contendo:

- titulo (String)
- descricao (String)
- dataInicio (Timestamp)
- dataFim (Timestamp)
- uid (String) -> ID do usuário dono da tarefa

6. Comparação entre os Projetos

Recurso	app_tarefa	app_tarefaSE
-----	-----	-----
Firebase Auth	[X]	[OK]
Cloud Firestore	[X]	[OK]
google-services.json	[X]	[OK]
UID por tarefa	[X]	[OK]
CRUD em tempo real	[X]	[OK]

7. Conclusão

A integração com o Firebase permite transformar um app Flutter simples em uma aplicação escalável,

Apresentação Técnica: Integração Firebase no App Flutter

segura e com persistência em nuvem. Ideal para apps com múltiplos usuários, autenticação, e dados em tempo real.