

Laboratório de Desenvolvimento de Aplicações Móveis e Distribuídas Sincronização *Off-line first*

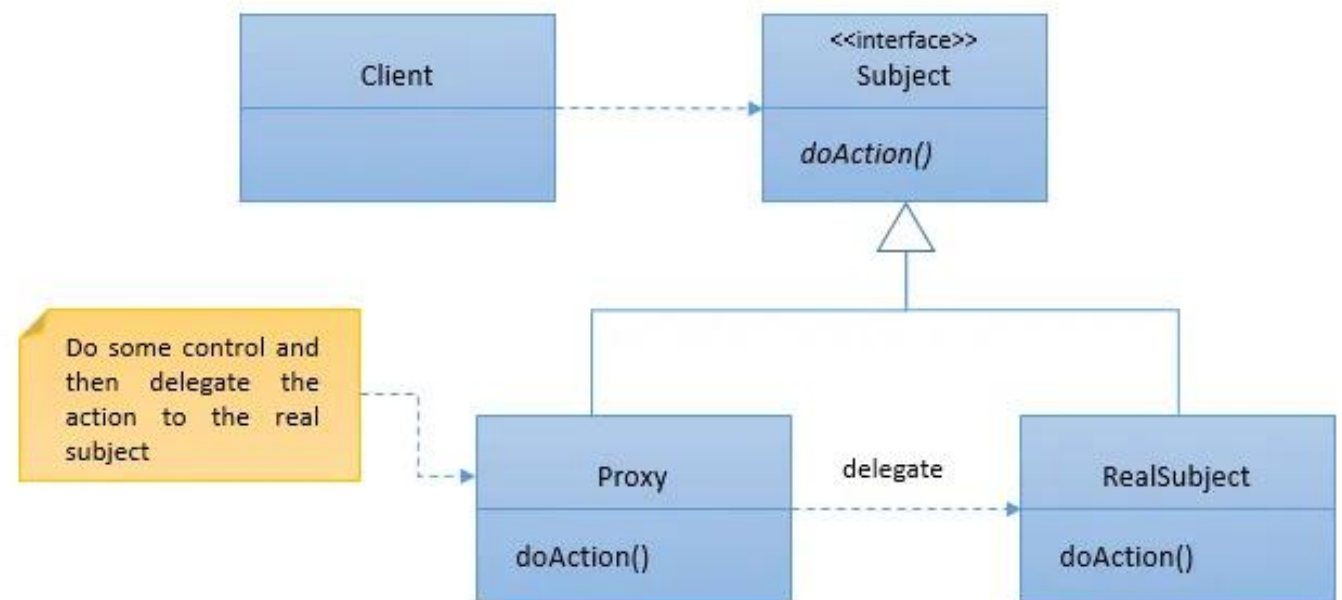
Prof. Hugo Bastos de Paula

Off-line first

- Abordagem de desenvolvimento onde o app funciona sem conexão com a internet ou com conexão intermitente (típico móvel, mas pode ser web).
- Usuário pode acessar e modificar dados críticos mesmo offline.
- Sincronização automática dos dados quando a conexão é restabelecida.
- Exemplo: WhatsApp, Google Docs, Trello, Twitter Lite.
- Melhora a experiência do usuário: menos latência, mais confiabilidade.
- Reduz dependência do backend e economiza banda.
- Permite produtividade contínua e acesso a funcionalidades essenciais.

Componentes

- Camada de Dados Local:
 - banco de dados local
 - SQLite,
 - Room,
 - Hive,
 - Isar



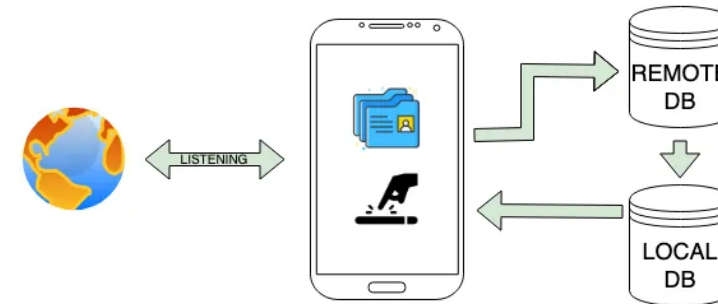
Componente

- Camada de Sincronização:
 - gerencia envio e recebimento de dados do servidor.

<https://islamoktay.medium.com/offline-first-and-syncing-local-with-remote-database-in-flutter-d64796379480>

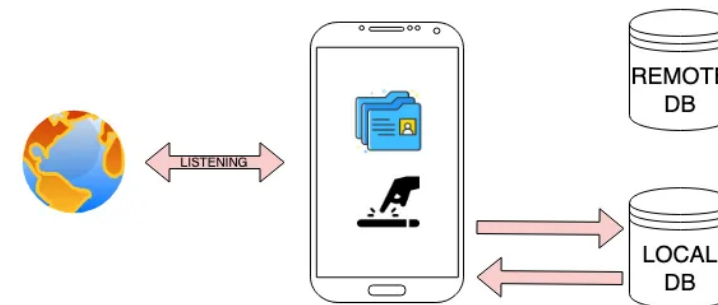
STATUS: Stable Network Connection

DESIRED ACTION: It should send request to remote db, and sync REMOTE DB with LOCAL DB on successful interaction.



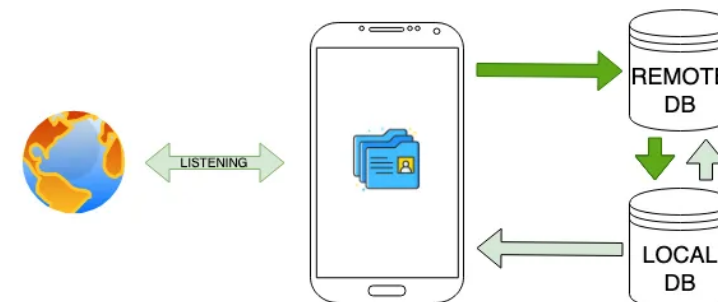
STATUS: No network Connection

DESIRED ACTION: It should save request to LOCAL DB, then update the state for saved requests and show the requests to user. Optional notification can be provided periodically for the waiting requests.



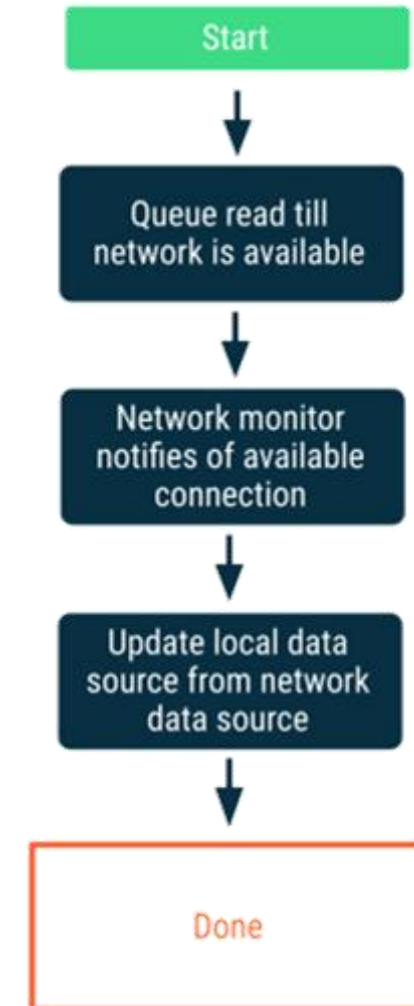
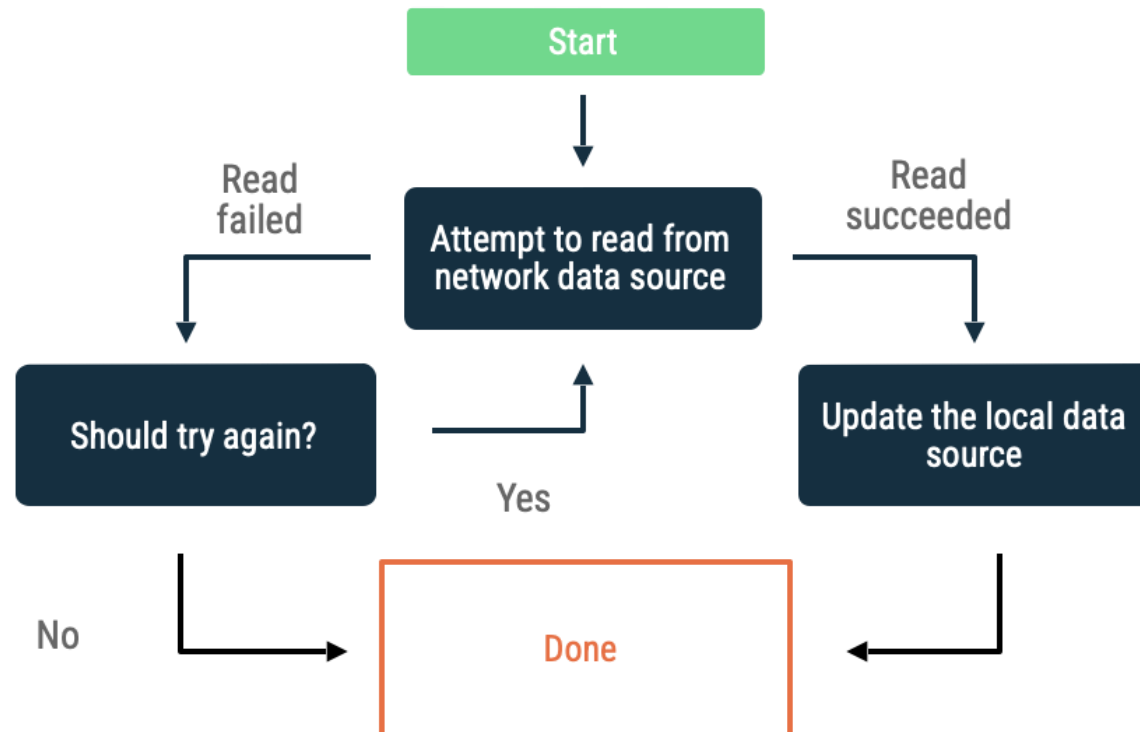
STATUS: Saved requests in LOCAL DB, and stable network maintained

DESIRED ACTION: It should listen network status. If stable network is maintained, it should send the waiting requests to REMOTE DB. Then it should update state for the waiting requests if they are successful or otherwise.



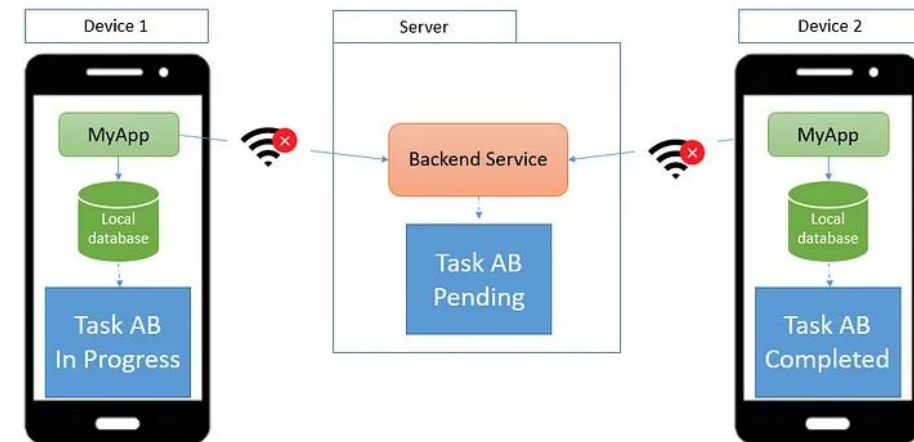
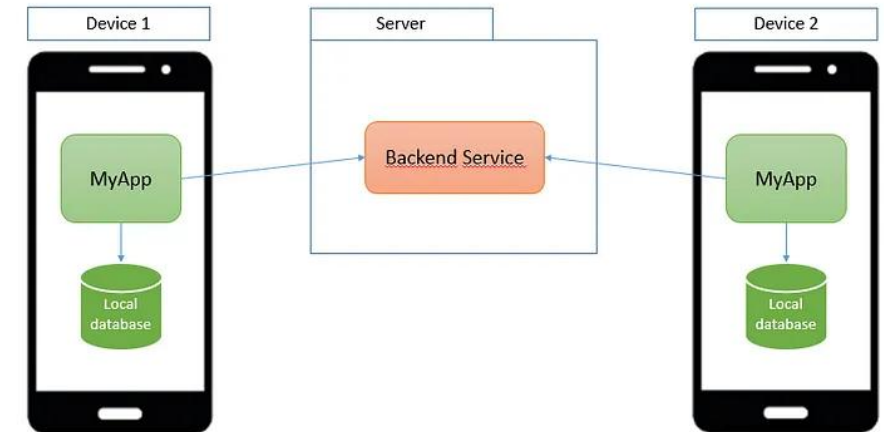
Componentes

- Camada de Conectividade:
 - detecta mudanças de estado da rede.



Componentes

- Camada de Resolução de Conflitos:
 - trata divergências entre dados locais e remotos

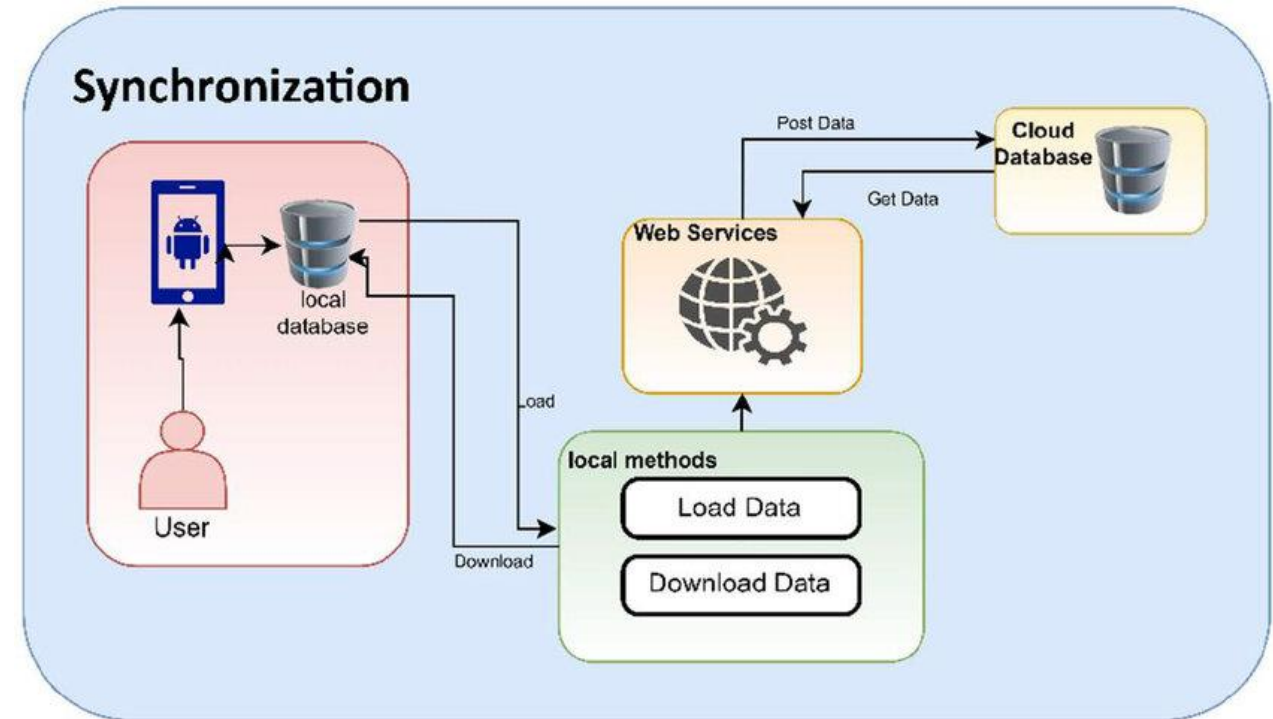


Estratégias de Sincronização

- Sincronização Otimista: assume que conflitos são raros; alterações locais são aplicadas imediatamente e sincronizadas depois.
- Sincronização Pessimista: assume que conflitos são comuns; bloqueia dados durante edição até sincronizar com o servidor.
- Incremental Sync: apenas dados alterados são sincronizados para otimizar tempo e banda.
- Background Sync: sincronização automática em segundo plano, respeitando bateria e conexão

Fluxo Básico de Sincronização

- 1.Usuário faz alterações offline.
- 2.Dados são salvos localmente.
- 3.App monitora conectividade.
- 4.Ao detectar conexão, inicia sincronização.
- 5.Resolve conflitos (ex: “última escrita vence” ou merge manual).
- 6.Atualiza dados locais e remotos.



https://www.researchgate.net/publication/361245836_Synchronization_procedure_for_data_collection_in_offline-online_sessions

Tecnologias

- Banco Local: SQLite, Room (Android), Hive (Flutter), Core Data (iOS)
- Sincronização: WorkManager (Android), Background Fetch (iOS), Service Workers (PWA)
- Controle de Versão: versionamento local e remoto para detectar conflitos
- Algoritmos: Rolling checksum, three-way merge para arquivos grandes
- Resolução de Conflitos:
 - Automática: “server wins”, “client wins”, merge por timestamp.
 - Manual: usuário escolhe qual versão manter.
 - Versionamento: cada alteração gera um novo número de versão para facilitar o merge
- <https://github.com/arn4v/offline-first>

Tecnologias

Recurso	Isar	sqflite	Floor/Drift	Realm	ObjectBox	Hive
Tipo	NoSQL	SQL	ORM (sobre sqflite)	NoSQL	NoSQL	NoSQL (usa JSON)
Facilidade de Uso	API simples, configuração mínima	API de baixo nível, requer mais código	Configuração mais complexa, mas código mais limpo	API simples	API simples	API muito simples
Desempenho	Excelente	Muito bom	Bom	Excelente	Excelente	Bom
Segurança de Tipos	Forte (tipos Dart)	Não	Forte (via código gerado)	Opcional	Opcional	Não
Funcionalidades	Rico (consultas flexíveis, JSON, criptografia, busca full-text)	Básico (consultas SQL)	Rico (baseado no sqflite)	Rico (similar ao Isar)	Rico (similar ao Isar)	Básico (armazenamento chave-valor)
Conformidade ACID	Sim	Sim (via sqflite)	Sim (via sqflite)	Sim	Sim	Não (consistência eventual)