

Soluções Técnicas para Backup de Exames - App + SAÚDE

Considerando a arquitetura proposta (Flutter + Firebase) e as necessidades do público idoso (simplicidade, segurança, acessibilidade), detalhamos as soluções técnicas para a funcionalidade de backup/armazenamento de exames médicos:

1. Armazenamento Seguro na Nuvem

- **Tecnologia: Firebase Storage.** Integrado ao ecossistema Firebase, oferece armazenamento escalável e seguro para arquivos como PDFs e imagens de exames.
- **Estrutura de Pastas:** Organizar os arquivos por usuário para garantir isolamento e facilitar o gerenciamento de permissões. Exemplo de caminho: `exams/{userId}/{examId}.{extensao}` (onde `{userId}` é o ID único do usuário no Firebase Authentication).
- **Segurança de Acesso:** Configurar **Firebase Storage Security Rules** rigorosas. A regra básica deve permitir que apenas o usuário autenticado leia e escreva em sua própria pasta (`allow read, write: if request.auth != null && request.auth.uid == userId;`). Regras mais granulares podem ser implementadas para compartilhamento.
- **Criptografia:** O Firebase Storage criptografa os dados em repouso automaticamente. A comunicação (upload/download) ocorre via HTTPS, garantindo criptografia em trânsito.

2. Processo de Upload Simplificado

- **Interface no App (Flutter):**
 - Botão grande e claro na seção "Agenda" ou em uma seção dedicada "Meus Exames": "Adicionar Exame".
 - Opções claras após clicar no botão:
 - "Tirar Foto do Exame": Abre a câmera do dispositivo diretamente. Interface da câmera deve ser simples.
 - "Escolher Arquivo do Exame": Abre o seletor de arquivos/galeria do dispositivo.
 - Pré-visualização da imagem/arquivo selecionado antes do upload.
 - Barra de progresso visual clara durante o upload.

- **Metadados Essenciais (Opcionais, mas recomendados):**
 - Após selecionar/tirar foto, apresentar um formulário simples:
 - "Tipo de Exame" (Campo de texto simples ou lista pré-definida: Sangue, Imagem, Coração, Outro).
 - "Data do Exame" (Seletor de data simplificado).
 - "Descrição Curta" (Campo de texto opcional para identificação).
 - Esses metadados seriam salvos no **Firestore** ou **Realtime Database**, associados ao ID do arquivo no Storage, permitindo busca e organização sem precisar baixar os arquivos.

3. Visualização e Organização no App

- **Listagem:** Tela "Meus Exames" exibindo os exames em uma lista vertical clara.
 - Cada item da lista deve mostrar: Tipo de Exame, Data, Descrição Curta.
 - Usar ícones representativos (ex: gota de sangue, raio-x) ao lado do tipo para facilitar a identificação visual.
 - Fonte grande e bom espaçamento.
- **Visualização do Exame:**
 - Ao tocar em um item da lista, abrir o exame.
 - Usar pacotes Flutter para visualização in-app:
 - PDFs: `flutter_pdfview` ou similar.
 - Imagens: Widget `Image` padrão, dentro de um `InteractiveViewer` para permitir zoom com gestos simples (pinça).
 - Controles de visualização (zoom, paginação para PDF) devem ser simples e acessíveis.
- **Organização/Busca (Opcional Avançado):** Uma busca simples baseada nos metadados (Tipo, Data, Descrição) pode ser implementada.

4. Segurança e Privacidade (Reforço)

- **Autenticação:** Acesso à funcionalidade de exames só é permitido após login seguro via Firebase Authentication.
- **Regras de Acesso:** As regras do Firebase Storage e Firestore/Realtime Database são a principal linha de defesa para garantir que apenas o usuário acesse seus dados.
- **Conformidade LGPD:** O aplicativo atua como um cofre digital para os exames enviados pelo usuário. É crucial informar claramente na política de privacidade como os dados são armazenados e protegidos. O controle sobre o que é armazenado é do usuário. Não realizar processamento ou análise dos conteúdos dos exames sem consentimento explícito.

5. Compartilhamento Seguro (Funcionalidade Avançada Opcional)

Se a funcionalidade de compartilhar exames com familiares/médicos for implementada:

- **Modelo de Permissão:**

- O idoso deve explicitamente conceder permissão para um familiar (previamente cadastrado na seção "Familiares") visualizar exames específicos ou todos os exames.
- Gerenciar essas permissões no Firestore/Realtime Database (ex: uma coleção `examPermissions` que mapeia `userId_familiar -> userId_idoso -> [examId1, examId2]` ou `'all'`).
- As Security Rules do Storage/Firestore seriam adaptadas para verificar essas permissões antes de liberar o acesso.

- **Interface de Compartilhamento:**

- Opção "Compartilhar" na tela de visualização do exame.
- Mostrar lista de familiares autorizados com checkboxes/botões grandes para selecionar com quem compartilhar.
- Confirmação clara antes de efetivar o compartilhamento.

- **Alternativa (Link Seguro):** Gerar um link seguro e temporário para um exame específico usando Firebase Cloud Functions. Isso é mais complexo de implementar e gerenciar, mas evita a necessidade de cadastro do destinatário no app. O modelo de permissão interna é geralmente mais seguro e controlável para este público.

Esta abordagem técnica visa equilibrar funcionalidade, segurança e a necessária simplicidade para o público idoso, utilizando as ferramentas do ecossistema Firebase de forma eficaz.

Referências: * Documentação Firebase (Storage, Firestore, Authentication, Security Rules) * Documento: Modelo-Proposta de Projeto de Extensão-1.pdf * Sugestões Arquitetura/Funcionalidades: `/home/ubuntu/arquitetura_funcionalidades_sugeridas.md`