

MyStuff— Controle de Inventário

Aplicativo Android para o acompanhamento de bens pessoais.

. . .

Organização é essencial em tudo que nos propomos a fazer, e isso começa com algo que muitas vezes é ignorado: nosso conjunto de pertences (“nossas coisas”). Algo que certamente já aconteceu pelo menos uma vez a qualquer pessoa foi ter de procurar por algo em casa na última hora antes de sair para um compromisso.

Para organizar nossos pertences, primeiro é necessário que saibamos o que temos e a localização de cada item. Essa é a função do MyStuff, criar um catálogo dos itens pessoais do usuário e registrar sua localização dentro da sua casa, provendo uma maneira eficiente de encontrar o que se precisa, quando se precisa.

Isso proporciona melhor gerenciamento do tempo e maior eficiência na hora de realizar as atividades cotidianas, visto que os itens necessários são mais facilmente localizados, além de outros benefícios.

. . .

Definição do Problema

Com a quantidade de coisas que se adquire ao longo da vida, pode se tornar difícil controlar e localizar cada uma delas, o que acaba levando, quase inevitavelmente, à desorganização. Com o passar do tempo, aquilo que se achava estar em um lugar já não é mais encontrado lá, causando perda de tempo e, em alguns casos (dependendo da urgência em se localizar determinado item), até estresse.

Uma maneira de solucionar este problema é manter um controle dos itens pessoais e de sua localização, sendo uma das formas mais comuns anotar itens em papel (como cadernos ou blocos de anotações, por exemplo), o que pode ser útil, mas está sujeito às seguintes dificuldades:

- Legibilidade: A letra de quem faz o registro deve ser legível, dificultando a leitura caso não o seja;
- Durabilidade: O papel está sujeito à degradação pelo tempo, bem como fatores climáticos (umidade, temperatura) e acidentes (líquidos, fogo, etc.);

Para amenizar os problemas inerentes à anotação em papel, a solução lógica (em tempos atuais) é registrar os itens em meios eletrônicos, como documentos de texto ou planilhas. O problema, neste caso, vem do fato que, muito possivelmente, não se está usando uma ferramenta específica para esse fim, o que pode dificultar o registro, dependendo do nível de afinidade tecnológica do usuário.

Neste sentido, já existem alguns aplicativos que se propõem a realizar o controle de inventário, como demonstrado abaixo:

App	Desenvolvedor	Monetização	Público-Alvo	Observações
Stock e Inventário Simples	Chester Software	Grátis (compras no app)	Comércio/Lojas	
Inventário Móvel	Bino Solutions	Freemium	Comércio/Lojas	
Sistema de Inventário Inteligente	NonZeroApps	Grátis (compras no app)	Comércios/Lojas Usuários em Geral	Aparenta ser destinado principalmente a lojistas/comerciantes, já que registra variações nas quantidades de itens, bem como quantidades mínimas.
Inventário móvel PRO	FotonApps	Grátis (compras no app)	Comércios/Lojas Usuários em Geral	
MyStuff	Demétrius Jr.	Grátis	Usuários em Geral	

Comparação entre o MyStuff e alguns sistemas correlatos (autoria própria)

Como se vê, a maioria destes sistemas se destina a controle de movimentação de estoque, através da leitura de códigos de barras e afins, direcionado principalmente a lojistas e comerciantes em geral.

O MyStuff difere destes sistemas correlatos justamente no público-alvo, que, neste caso, engloba o usuário comum, no seu dia a dia. Este busca resolver seus problemas com o menor número possível de cliques (ou toques na tela, neste caso) e com terminologia acessível ao seu caso de uso, de maneira a evitar o atrito inerente ao aprendizado de uma ferramenta que não foi pensada para o seu problema.

Para entender melhor como (e se) o público-alvo do MyStuff controla seu inventário, foi realizada uma pesquisa que contou com 15 respostas. A metodologia escolhida foi um formulário do Google, disponível [aqui](#).

Para você, qual a importância de manter o controle sobre os bens (imóveis, móveis, utensílios, ferramentas, decoração, etc.) que possui?

15 respostas

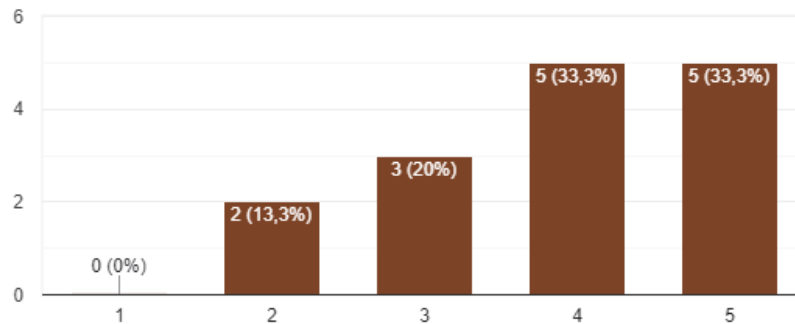


Gráfico 1: Nível de Importância atribuído à organização dos bens pessoais

Com que frequência você precisa procurar pelas coisas em casa?

15 respostas

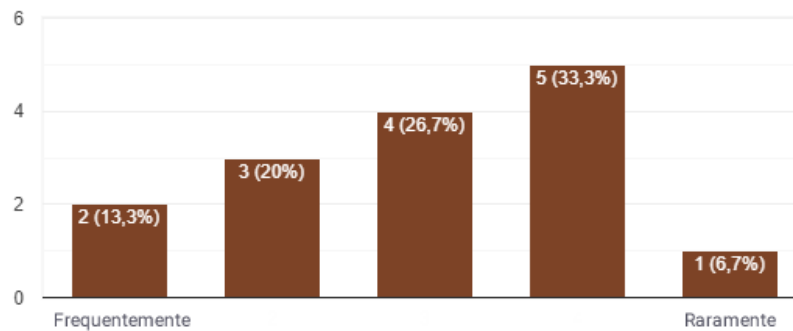


Gráfico 2: Frequência da necessidade de se procurar algum item no espaço doméstico

Você mantém um controle daquilo que possui?

15 respostas

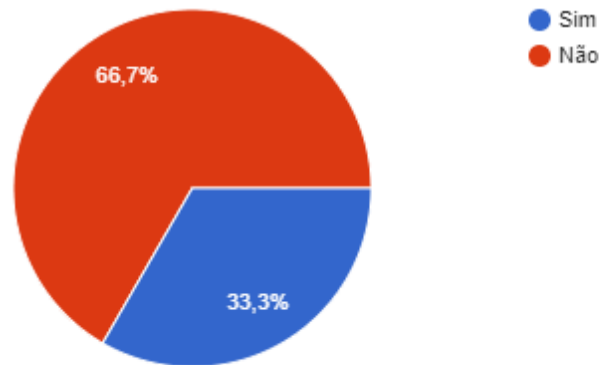


Gráfico 3: Realização de controle dos bens pessoais

O Gráfico 1 demonstra que 86,6% dos usuários que responderam à pesquisa atribuem algum nível de importância (de regular a elevado) à organização de seus bens. 60% dizem precisar procurar por itens em sua casa com certa frequência (de regular a elevada), segundo o Gráfico 2. Por fim, no Gráfico 3 podemos ver que 66,7% afirmam que não realizam nenhum tipo de controle sobre seus bens pessoais.

Destes resultados, fica clara a possibilidade de suprir a necessidade de controle sobre o inventário pessoal através de um aplicativo específico para este fim.

. . .

Objetivos

A ideia é que o MyStuff seja um aplicativo nativo para o sistema Android, que forneça uma ferramenta simples para que o usuário crie um catálogo daquilo que possui, sendo possível gerenciar seus itens de maneira intuitiva.

O objetivo principal é prover uma maneira confiável e duradoura para que o usuário armazene os inventários de seus itens pessoais, possibilitando a consulta e modificação sempre que necessário, em uma interface que apresente ao usuário exatamente a informação que ele precisa sobre aquilo que possui.

Para tanto, busca-se maneiras de incluir, categorizar e organizar os itens sob sua posse de acordo com sua localização, de modo que o usuário possa ter uma visão detalhada de tudo o que tem, segundo os critérios que ele entender mais apropriados.

. . .

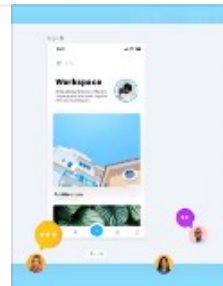
Stack Tecnológico

Para o desenvolvimento do projeto, foram escolhidas as seguintes tecnologias/ferramentas:

Marvel

Marvel - The design platform for digital products. Get started for free.

The collaborative design platform. Wireframe, prototype, user test, design and inspect designs i...
marvelapp.com



Marvel é uma ferramenta para modelagem de aplicações que permite criar protótipos interativos das telas do sistema.

Possui modelos para a maioria dos componentes da interface Android, possibilitando a criação de wireframes de alta fidelidade e facilitando a criação das telas do aplicativo no momento do desenvolvimento, uma vez que já há uma ideia clara do que se pretende alcançar.

Todos os protótipos da interface do MyStuff foram construídos no designer de telas do Marvel.

GitHub

GitHub - 15demi08/PosSenac_TCC: Trabalho de Conclusão de Curso para a Pós-Graduação ...

Trabalho de Conclusão de Curso para a Pós-Graduação em Desenvolvimento de Aplicações ...
github.com



Ferramenta de controle de versão, hospedagem e sincronização do código e outros arquivos do projeto.

Facilita o versionamento dos arquivos, bem como possibilita o “rollback” se houver algum problema.

Android Studio

<https://developer.android.com/studio>

IDE oficial e recomendada pelo Google para o desenvolvimento de aplicações nativas para o sistema operacional Android.


Possibilita o gerenciamento de dependências e metadados do projeto, a escrita do código, o gerenciamento de recursos multimídia (imagens, sons, vídeos, etc.) utilizados pelo app, além de recursos de exportação e assinatura do app, implantação e testes em dispositivos físicos e virtuais, entre outras funcionalidades.

Kotlin

Kotlin Programming Language

Easy to pick up, so you can create powerful applications immediately. Concise data class ...

kotlinlang.org

The image shows a promotional card for the Kotlin Programming Language. On the left, there is a white box containing the text 'Kotlin Programming Language', a short description 'Easy to pick up, so you can create powerful applications immediately. Concise data class ...', and the website 'kotlinlang.org'. To the right of this box is a colorful graphic with purple and pink geometric shapes and the word 'Kotlin' in white text.

Linguagem de programação recomendada pelo Google para o desenvolvimento Android.

Em relação ao Java, possui uma escrita mais sucinta e legível, além de prover construções de linguagem extremamente úteis (como o “*let {}*” e o “*apply {}*”, por exemplo).

Room (SQLite)

<https://developer.android.com/training/data-storage/room?hl=pt-br#kotlin>

Biblioteca de abstração para persistência de dados baseada em SQLite, recomendada pelo Google.

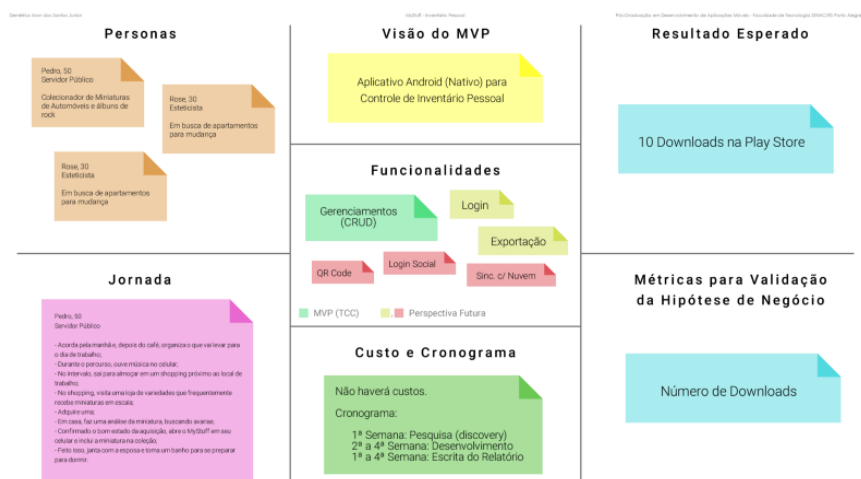
Possibilita a criação do Banco de Dados da aplicação através da definição de Entidades representadas por Classes de Dados e Objetos de Acesso (DAOs), além de substituir muito do código boilerplate que seria escrito para a utilização da biblioteca SQLite nativa do Android.

. . .

Descrição da Solução

Busca-se desenvolver uma ferramenta que possibilite ao usuário criar um catálogo dos itens sob sua posse, organizá-los de acordo com seus próprios critérios, e consultar de maneira simples sempre que for necessário.

O detalhamento do problema e a solução proposta constam no quadro Canvas MVP:



Canvas MVP (autoria própria) — disponível aqui.

A fim de flexibilizar a definição destes critérios, ao passo que as nomenclaturas finais dos dados incluídos no MyStuff ficarão a cargo do usuário, foram estabelecidos alguns termos que servem de “guia” para a organização dos itens:

- **Inventário:** Conjunto de itens que possuem alguma característica geral em comum. Por exemplo: “Itens de Casa” (todos os itens presentes na residência do usuário), “Coleção de DVDs”, etc.
- **Local:** A localização física de um item. Exemplo: Dentro do Inventário “Itens de Casa”, o usuário pode definir locais de

acordo com os cômodos de sua residência—“Cozinha”, “Quarto”, etc.

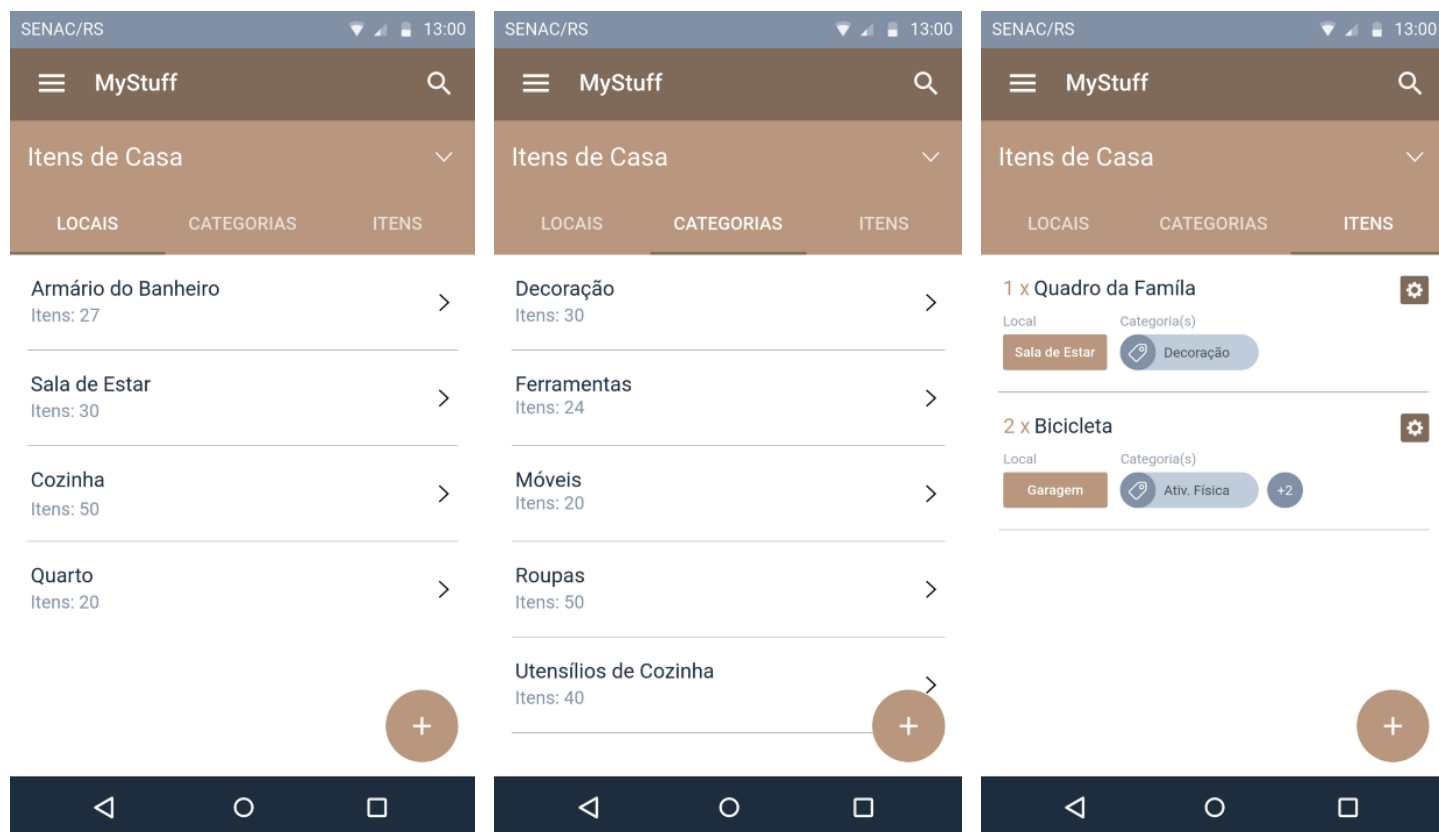
- **Categoria:** Uma característica específica inerente a um ou mais itens, mesmo em locais diferentes. Exemplo, para os “Itens de Casa”, podem ser criadas categorias como “Ferramentas”, “Decoração”, “Eletrodomésticos”, etc.
- **Item:** Cada objeto individual sob posse do usuário, posicionado em um determinado Local e pertencente a uma ou mais Categorias. Seguindo o exemplo dos “Itens de Casa”, poderia haver um item nomeado “Geladeira”, localizado na “Cozinha”, categorizado como “Eletrodomésticos”.

O uso do MyStuff começa pela criação de um ou mais Inventários:



Lista de Inventários—Protótipo criado na ferramenta Marvel

Escolhido um inventário, há listas específicas para visualizar e incluir Locais, Categorias e Itens:



Listas de Locais, Categorias e Itens—Protótipos criados na ferramenta Marvel

Locais e Categorias podem ser criados de maneira avulsa. Como melhoria futura, planeja-se possibilitar a criação também no momento da inclusão de um item:

SENAC/RS13:00

←

Criar/Editar Item

🔍

Descrição

Nome do Item

Quantidade

-

1

+

Local

Selecione

▼

+

Categoria(s)

Selecione

▼

+

VOLTAR

SALVAR

EXCLUIR

◀

○

□

Criação/Edição de Item—Protótipo criado na ferramenta Marvel

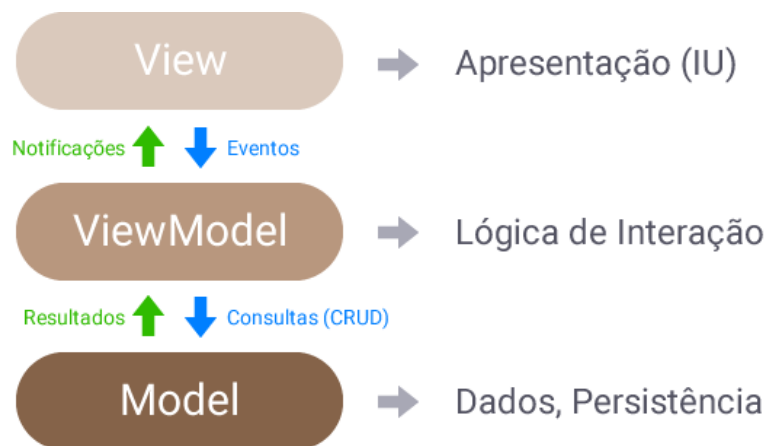
Na versão inicial, os dados do aplicativo são salvos em um Banco de Dados local, armazenado no dispositivo do usuário. Se o aplicativo tiver seus dados limpos ou for desinstalado, ou se o dispositivo for formatado, perdido, roubado, etc., os dados serão perdidos.

A fim de resolver este problema, como melhoria futura, planeja-se implementar as funcionalidades de Exportação dos dados (para formatos como PDF e CSV, por exemplo), além de Login Social e Sincronização em Nuvem (que poderá fazer parte da conta utilizada para o login, se o usuário desejar).

. . .

Arquitetura

O MyStuff segue o Padrão de Design conhecido como MVVM (*Model-View-ViewModel*), que separa o software em camadas responsáveis pelos dados e pela lógica de negócio (*Model*—Modelo), pela apresentação (*View*—Visão) e pela lógica de interação (*ViewModel*—Modelo de Visão):



Visualização do Padrão MVVM (autoria própria)

Para o Modelo, os dados que compõem o software, ao menos inicialmente, estão descritos no diagrama a seguir:

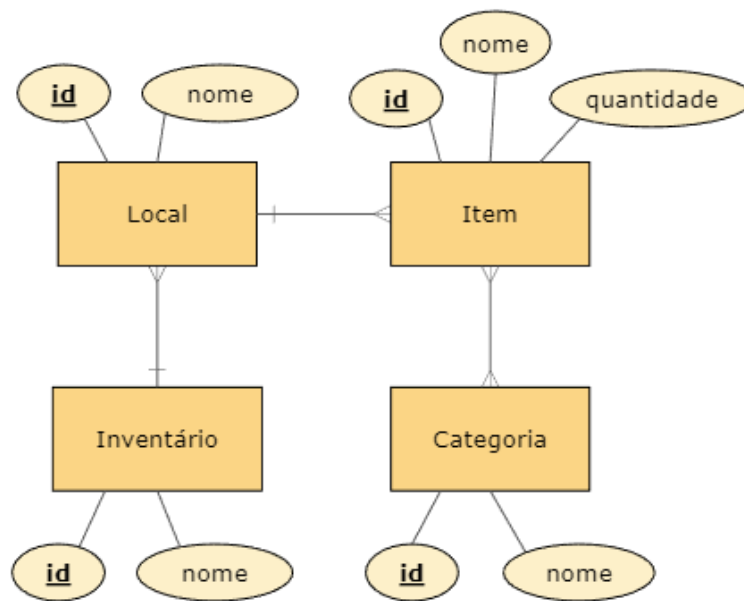
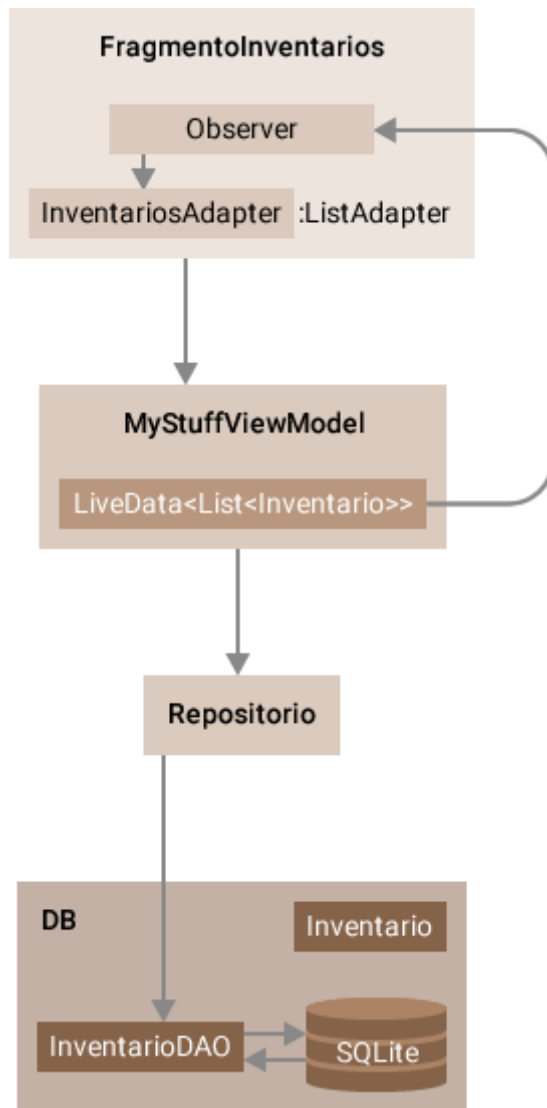


Diagrama Entidade-Relacionamento (autoria própria)

O banco de dados é local, criado na tecnologia SQLite, permitindo ao usuário acessar o aplicativo e manipular seu inventário mesmo na ausência de uma conexão à internet. No MyStuff, a implementação concreta do Banco de Dados fica a cargo da biblioteca Room, do Google.

Seguindo o padrão MVVM, a arquitetura do aplicativo está exemplificada no diagrama a seguir:



Parte da Arquitetura do MyStuff (Autoria própria, adaptado daqui).

Na Camada de Dados, a classe **DB**, que herda de **RoomDatabase**, é responsável por prover a instancia do banco de dados e os Objetos de Acesso a Dados (*Data Access Object*— **DAO**, em inglês) que realizarão as consultas.

A classe **Repositorio** fica em uma camada intermediária entre as Camadas de Dados e de Lógica de Interação, ficando responsável por isolar a implementação da Camada de Dados do resto da aplicação.

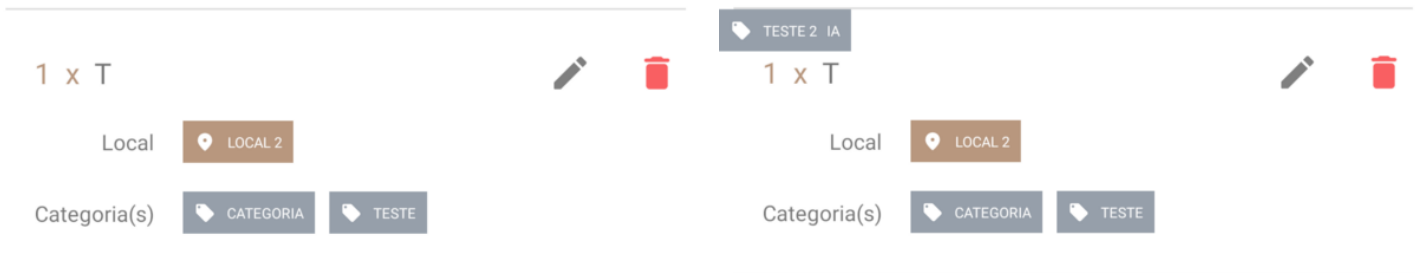
O **MyStuffViewModel** (em outras palavras, a Camada de Lógica de Interação) é responsável por conter os dados necessários à Camada de Apresentação e acionar os respectivos **DAOs** sempre que um evento de IU for disparado, bem como notificar a IU sobre mudanças nos dados—o que acontece através do uso de objetos **LiveData** e seus respectivos **Observers**.

Conclusões

Do que foi exposto, conclui-se pela possibilidade de que o MyStuff venha a alcançar o objetivo ao qual este projeto se propôs—prover uma ferramenta nativa para o sistema Android que habilite o usuário a elaborar um inventário de seus itens pessoais de maneira rápida e descomplicada, trazendo mais organização e removendo empecilhos à realização de outras tarefas do dia a dia.

Limitações do Projeto e Perspectivas Futuras

O desenvolvimento do MVP foi concluído, mas não foi possível corrigir um bug visual, na Lista de Itens. Ao se editar um item, se houver mudança na lista de Categorias associadas a eles, a atualização do item na lista pode criar categorias órfãs dentro dele:



Esquerda: Item original. Direita: O mesmo item, após adição e remoção de uma terceira categoria

Infelizmente a documentação e guias disponibilizados oficialmente pelo Google deixam a desejar em vários pontos, e buscas na internet (como no [Stack Overflow](#), por exemplo) nem sempre dão os frutos esperados.

Devido ao tempo reduzido para a concepção e realização do projeto, bem como a interferência de fatores externos à cadeira do TCC, não foi realizada a etapa de validação, a qual será realizada posteriormente, através de pesquisa via Formulário do Google, de maneira análoga à etapa de descobrimento, citada em **Definição do Problema**.

Na seção central do Canvas MVP (em **Descrição da Solução**), constam outras funcionalidades idealizadas para o aplicativo que não puderam ser desenvolvidas dentro do tempo disponível. Estas também serão implementadas mais tarde, após a validação do MVP.

Referências Bibliográficas

- Guia do Desenvolvedor Android, Google. Disponível em <<https://developer.android.com/guide>>
- Guia de IU do Desenvolvedor Android, Google. Disponível em <<https://developer.android.com/develop/ui>>