



UNIVERSIDADE  
FEDERAL DO CEARÁ  
CAMPUS DE QUIXADÁ

**CURSO DE ENGENHARIA DE SOFTWARE**

**RELATÓRIO – PROJETO INTEGRADO III**  
**Doe**

**Nome:** Jeferson Gonçalves Noronha Soriano

**Matrícula:** 471110

**Professor:** Camilo Camilo Almendra

**QUIXADÁ**

**Outubro, 2021**

## SUMÁRIO

- 1 INTRODUÇÃO
- 2 DESCRIÇÃO DO SISTEMA
- 3 FEATURES
- 4 REQUISITOS
  - 5.1 Requisitos Funcionais
  - 5.2 Requisitos Não Funcionais
- 5 ESTÓRIA DE USUÁRIO
- 6 PROTÓTIPO
- 7 LINK PARA REPOSITÓRIO E NOTION
  - 7.1 GitHub
  - 7.2 notion

## 1 INTRODUÇÃO

Um tema que sempre está em alta, principalmente no Brasil, é a doação, não só pelo altruísmo que o ato representa, mas pela necessidade, visto que no país existem muitas pessoas em situação desfavorável, que estão precisando de alguma coisa, (seja isso um bem ou serviço) e que por falta de dinheiro não conseguem adquirir/usufruir. Levando em consideração o contexto tecnológico, já existem aplicativos feitos especificamente para doação monetária, pensando nisso, comecei a analisar algumas coisas a minha volta, como por exemplo: minha tia estava com uma banheira de criança que não tinha mais serventia alguma para ela, no entanto, ela simplesmente não sabia para quem doar porque não conhecia ninguém que a quisesse e cogitou jogá-la fora, sem contar em coisas que julgamos que ninguém irá querer, seja pela simplicidade ou pelo desconhecimento da necessidade, assim, deixamos de ajudar alguém e acabamos jogando fora, dada essa análise, pensei em transformar a ideia em um aplicativo, onde haveria uma comunidade onde todos podem se ajudar, com um fluxo de dois tipos de usuário postando no aplicativo: os que possuem tal produto e que querem doar e os interessados que estão precisando, assim, poderá ser feito o contato entre as partes e uma boa ação será consolidada.

## 2 DESCRIÇÃO DO SISTEMA

O sistema proposto é denominado **Doe** e tem como solução principal facilitar a comunicação entre o usuário doador(provedor) com o usuário que recebe a doação(beneficiado) e é pensado para a plataforma mobile.

O aplicativo tem essa divisão simples entre em doador e beneficiado(o mesmo usuário pode ser os 2 tipos de usuário), com um fluxo diferente para cada um, o beneficiado não precisa realizar login, assim que entrar no app se depara com um mapa com pin que representa os “postes” dos usuários provedor, esse pin será mostrado dentro de um raio que o usuário beneficiado poderá alterar, clicando nesse pin terá uma funcionalidade que mostra um rota até a pessoa que vai doar, e também terá informações sobre o produto/serviço e uma forma de entrar em contato com o provedor para combinar uma hora para receber o produto

Quando o usuário quer ser um provedor o que ele precisa fazer é se logar no sistema ou criar um conta caso não tenha, entrado no perfil ele poderá ver seus post já feitos, onde

poderá apagar, editar e criar, esses são os “postes” que será mostrado em forma de pin na tela principal

### 3 FEATURES

- **CRUD simples do usuário (Cadastro, Alteração e Remoção)**

- Cadastro do usuário na plataforma.

- Login do usuário na plataforma.

- Edição das informações do usuário na plataforma.

- Exclusão da conta do usuário na plataforma.

- **CRUD simples do objetos/bens(Cadastro, Listagem, Remoção e Alteração).**

- Definição de hábitos do usuário, na plataforma.

- Listagem dos hábitos do usuário, cadastrados na plataforma.

- Edição dos hábitos do usuário, cadastrados na plataforma..

- **Criação de categorias.**

- roupas

- objetos

- brinquedo

- comida

- e mais

- **Filtragem dos “postes” a partir das categorias.**

- **Criação dos pin**

- diferenciação com cores seguindo as categorias

- **Utilização da geolocalização**

- para mostrar os “postes” próximo a pessoa

- modificação do raio de proximidade

- calcular um rota até o usuário provedor

## 4 REQUISITOS

### 4.1 Requisitos Funcionais

#### USUÁRIO ENQUANTO DOADOR

##### [RF01] - O usuário poderá realizar seu cadastro no sistema

<b>Identificador</b>	[RF01] - O usuário poderá realizar seu cadastro no sistema
<b>Descrição</b>	A partir dos dados pessoais (Nome, sobrenome, email, senha) requisitados no formulário de cadastro, o sistema deve enviar um email de confirmação para o endereço de email fornecido.
<b>Prioridade</b>	Essencial.

Tabela 1 – Tabela do Requisito Funcional 01.

##### [RF02] -O usuário poderá efetuar login no sistema.

<b>Identificador</b>	[RF02] - O usuário poderá efetuar login no sistema.
<b>Descrição</b>	O usuário deve fornecer o email e senha que usou no cadastro, no formulário de login.
<b>Prioridade</b>	Essencial.

Tabela 2 – Tabela do Requisito Funcional 02.

##### [RF03] - O usuário poderá editar suas informações no sistema

<b>Identificador</b>	[RF03] - O usuário poderá editar suas informações no sistema.
<b>Descrição</b>	A partir do momento que o usuário possui sua conta confirmada e está logado na aplicação, ele deve poder editar as informações do seu perfil.
<b>Prioridade</b>	Essencial.

Tabela 3 – Tabela do Requisito Funcional 03.

**[RF04] - O usuário poderá excluir sua conta a qualquer momento.**

<b>Identificador</b>	[RF04] - O usuário poderá excluir qualquer momento
<b>Descrição</b>	A partir do momento que está logado na aplicação, ele deve ser capaz de apagar sua conta, feito isso, o sistema deve remover suas informações do banco de dados.
<b>Prioridade</b>	Essencial

Tabela 4 – Tabela do Requisito Funcional 04.

**[RF05] - O usuário deverá ser capaz de publicar um item a ser doado.**

<b>Identificador</b>	[RF05] - O usuário deverá ser capaz de cadastrar os seus hábitos definidos, na plataforma.
<b>Descrição</b>	O usuário deve poder postar um item a ser doado, preenchendo as informações (nome, descrição, categoria, foto).
<b>Prioridade</b>	Essencial

Tabela 5 – Tabela do Requisito Funcional 05.

**[RF06] - O usuário deverá ser capaz de visualizar a listagem de todas suas doações.**

<b>Identificador</b>	[RF06] - O usuário deverá ser capaz de visualizar a listagem de todas suas doações.
<b>Descrição</b>	O usuário deverá ser capaz de verificar todas suas doações cadastradas na plataforma, como também as informações referentes a elas.
<b>Prioridade</b>	Essencial

Tabela 6 – Tabela do Requisito Funcional 06.

**[RF07] - O usuário deverá ser capaz de editar suas doações.**

<b>Identificador</b>	[RF07] - O usuário deverá ser capaz de editar suas doações.
<b>Descrição</b>	O usuário deverá ser capaz de editar suas doações cadastradas na plataforma, alterando as informações referentes a elas.
<b>Prioridade</b>	Essencial

Tabela 7 – Tabela do Requisito Funcional 07.

**[RF08] - O usuário poderá excluir suas doações.**

<b>Identificador</b>	[RF08] - O usuário poderá excluir suas doações.
<b>Descrição</b>	O usuário deverá ser capaz de excluir uma doação cadastrada no sistema a partir do momento que ele já não tenha mais o objeto de doação, feito isso, o sistema deve remover as informações do banco de dados. .
<b>Prioridade</b>	Importante

Tabela 8 – Tabela do Requisito Funcional 08

## **USUÁRIO ENQUANTO BENEFICIADO**

**[RF09] - O usuário poderá visualizar os “posters” dentro de um raio em um mapa em forma de pin**

<b>Identificador</b>	[RF09] - O usuário poderá visualizar os posters dentro de um raio em um mapa em forma de pin
<b>Descrição</b>	Serão mostrados os posters próximo a pessoa com um cálculo de geolocalização.
<b>Prioridade</b>	Essencial.

Tabela 9 – Tabela do Requisito Funcional 09.

**[RF10] - O usuário poderá ver de forma detalhada os dados do produto entrando no post.**

<b>Identificador</b>	[RF10] - O usuário poderá ver de forma detalhada os dados do produto entrando no post.
<b>Descrição</b>	O usuário deve fornecer o email e senha que usou no cadastro, no formulário de logon.
<b>Prioridade</b>	Essencial.

Tabela 10 – Tabela do Requisito Funcional 10.

**[RF11] - O usuário beneficiado poderá entrar em contato com o doador.**

<b>Identificador</b>	[RF11] - O usuário beneficiado poderá entrar em contato com o doador.
<b>Descrição</b>	A partir do momento que o usuário está no post do que ele deseja receber ele poderá entrar em contato com o doador para conversar com ele .
<b>Prioridade</b>	Essencial.

Tabela 11 – Tabela do Requisito Funcional 11.

**[RF12] - O usuário poderá filtrar quais objetos/serviço irão aparecer de acordo com uma categoria.**

<b>Identificador</b>	[RF12] - O usuário poderá filtrar quais objetos/serviço irão aparecer de acordo com uma categoria.
<b>Descrição</b>	O usuário poderá filtrar o que aparecer para ele de acordo com um categoria de interesse, disponibilizada pelo sistema, exemplo: livro.
<b>Prioridade</b>	Essencial

Tabela 12 – Tabela do Requisito Funcional 12.



**[RF13] - O usuário deverá ser capaz de receber uma rota até o doador.**

<b>Identificador</b>	[RF13] - O usuário deverá ser capaz de receber uma rota até o doador.
<b>Descrição</b>	O usuário deverá ser capaz de ver uma rota até o doador o sistema deve mostrar na tela
<b>Prioridade</b>	Desejável

Tabela 13 – Tabela do Requisito Funcional 13.

**[RF14] - O usuário deverá ser capaz de alterar o raio que mostra os posters.**

<b>Identificador</b>	[RF14] - O usuário deverá ser capaz de alterar o raio que mostra os posters.
<b>Descrição</b>	O usuário deverá ser capaz de alterar o raio até no máximo 60 km que mostrar os posters nesse raio, e tem raio mínimo de 1 km
<b>Prioridade</b>	Importante

Tabela 14 – Tabela do Requisito Funcional 14.

**[RF15] - O sistema deve criar categorias.**

<b>Identificador</b>	[RF15] - O sistema deve criar categorias em forma de tag.
<b>Descrição</b>	O sistema deverá ter um sistema de categoria que servirá para facilitar a filtragem de posters algumas categorias: Eletrodoméstico roupas objetos brinquedo comida Outros
<b>Prioridade</b>	Essencial

Tabela 15 – Tabela do Requisito Funcional 15.

## 4.2 Requisitos Não Funcionais.

**[RNF01] - A interface deve possuir um design familiar e intuitivo**

<b>Identificador</b>	[RNF01] - A interface deve possuir um design familiar e intuitivo
<b>Descrição</b>	A interface deve ser familiar para os usuários, então usaremos o máximo de elementos de interface já consolidado e usado no mercado, seguindo também o material design da google
<b>Prioridade</b>	Essencial

Tabela 16 – Tabela do Requisito não Funcional 01.

**[RNF 02] - Os usuários não precisam de treinamento prévio para entender o fluxo base de execução.**

<b>Identificador</b>	[RNF 02] - Os usuários não precisam de treinamento prévio para entender o fluxo base de execução.
<b>Descrição</b>	Interface sendo o mais simples e descritivo possível, servindo para usuários que não tem conhecimento aprofundado de tecnologia, entendendo o sistema apenas seguindo o fluxo de execução base
<b>Prioridade</b>	Essencial

Tabela 17 – Tabela do Requisito não Funcional 02.

**[RNF 03] - Pins que aparecem no mapa devem ser divididos por cores para cada categoria.**

<b>Identificador</b>	[RNF 03] - Pins que aparecem no mapa devem ser divididos por cores para cada categoria.
<b>Descrição</b>	Para que não haja poluição visual nem confusão de informações para o usuário, o pin que aparece no mapa terá uma divisão de cor, por exemplo: roupa com o pin vermelho.
<b>Prioridade</b>	Importante

Tabela 18 – Tabela do Requisito não Funcional 03.

## Requisitos de Confiabilidade

**[RNF 04] - O sistema deve estar disponível 97% do tempo, 24 horas por dia , 7 dias por semana**

<b>Identificador</b>	[RNF 04] - O sistema deve estar disponível 97% do tempo, 24 horas por dia , 7 dias por semana.
<b>Descrição</b>	O sistema deve ser capaz de ficar no ar em média 97% do tempo de um calendário mensal, fora os períodos de manutenção e melhorias, isso são 24 horas por dia o dia que ficar no ar e 7 dias as semanas que fica no ar o sistema, ou seja, o quanto ele para resulta no percentual de 3%.
<b>Prioridade</b>	Importante

Tabela 19 – Tabela do Requisito não Funcional 04.

## Requisitos de Implementação

**[RNF 05] - O aplicativo deve funcionar em todo e qualquer android a partir do android 4.4 KitKat.**

<b>Identificador</b>	[RNF 05] - O aplicativo deve funcionar em todo e qualquer android a partir do android 4.4 KitKat.
<b>Descrição</b>	Os aplicativo deve rodar sem problemas em celulares a partir do android 4.4 sem problemas, a escolha desse android é que comportar cerca de 98% dos celulares do mundo
<b>Prioridade</b>	Importante

Tabela 20 – Tabela do Requisito não Funcional 05.

## Requisitos de Ética

**[RNF 06] - Não será divulgado nem negociado quaisquer dados relacionados ao usuário.**

<b>Identificador</b>	[RNF 06] - Não será divulgado nem negociado quaisquer dados relacionados ao usuário.
<b>Descrição</b>	Visando a integridade ética e a segurança dos nossos usuários é essencial que não iremos divulgar/vender, nenhum dado do usuário ou relacionado a ele.
<b>Prioridade</b>	Essencial

Tabela 21 – Tabela do Requisito não Funcional 06.

## 5 ESTÓRIA DE USUÁRIO

**EU 01.** como doador, eu gostaria de me cadastrar no sistema, pois eu quero ser um usuário doador do sistema.

**EU 02.** como usuário doador, eu gostaria de realizar login, pois eu quero ter acesso aos dados relacionados a mim.

**EU 03.** como doador, eu gostaria de mudar meu email, whatsapp e senha, pois eu quero manter meus dados atualizados.

**EU 04.** como doador, eu gostaria de deletar minha conta, pois eu não quero mais ter a conta.

**EU 05.** como doador, eu gostaria de cadastrar uma doação, pois eu quero doar.

**EU 06.** como doador, eu gostaria de visualizar todas minhas doações, pois eu quero visualizar tudo que estou doando.

**EU 07.** como doador, eu gostaria de editar informações sobre a doação, pois eu quero corrigir algum erro ou informação.

**EU 08.** como doador, eu gostaria de excluir o post da doação, pois o item já foi doado para alguém.

**EU 09.** como beneficiado, eu gostaria de visualizar doações próximas, pois eu quero encontrar de forma mais rápida possíveis doações que eu possa receber.

**EU 10.** como beneficiado, eu gostaria de ver de forma detalhada informações da doação, pois eu quero saber mais sobre o item.

**EU 11.** como beneficiado, eu gostaria de entrar em contato com o doador, pois eu quero combinar como vou receber a doação.

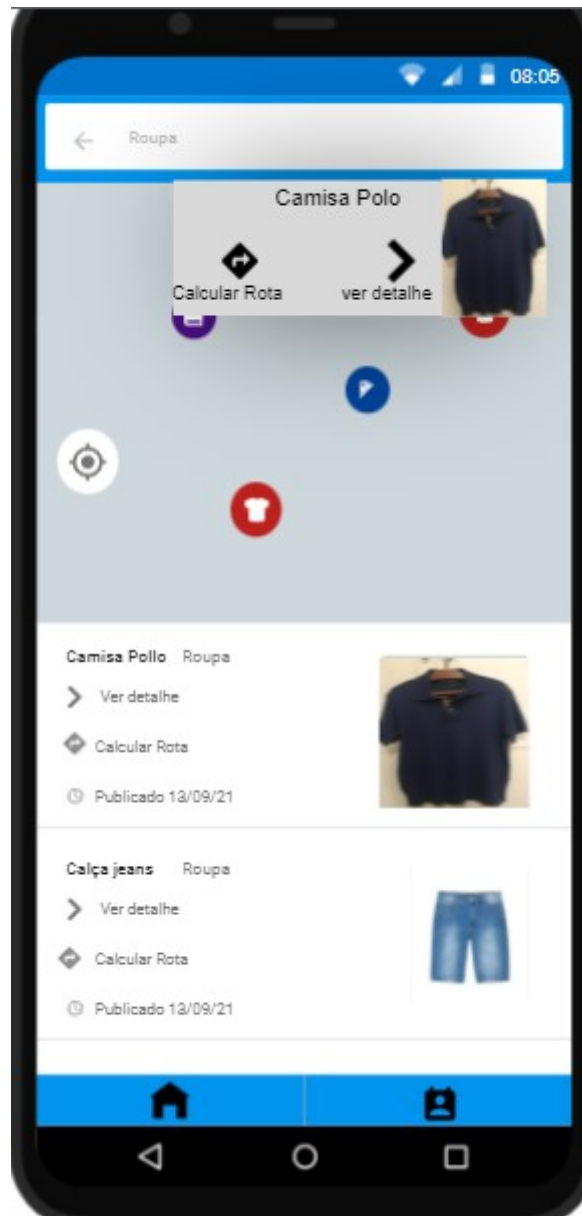
**EU 12.** como beneficiado, eu gostaria de pesquisar uma categoria , pois eu quero filtrar os posters que irão aparecer.

**EU 13.** como beneficiado, eu gostaria de ver uma rota até o doador, pois eu quero saber como chegar até o doador.

**EU 14.** como beneficiado, eu gostaria de alterar o raio que as posters aparecem, pois eu quero escolher a distância que quero pegar alguma doação.

## 6 PROTÓTIPO

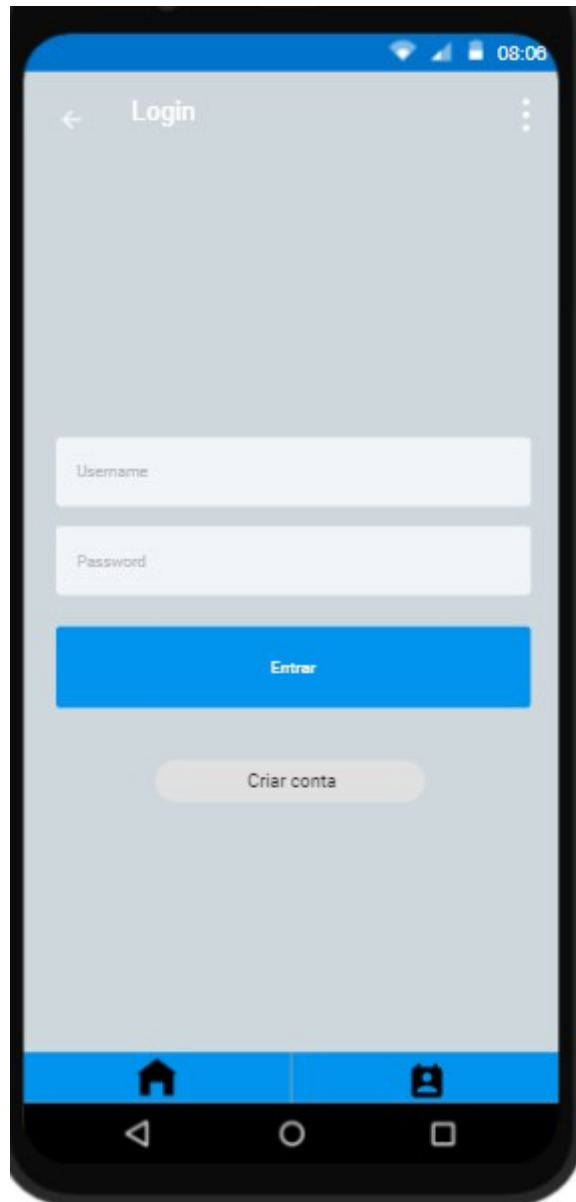
3 telas que representa mais ou menos o fluxo do aplicativo, a primeira tela seria a home que tem as informações dos postes cadastrados pelos doadores onde as pessoas podem mais detalhes sobre os itens, depois se o usuário quiser fazer uma doação, aí sim terá que fazer um login e entrar em uma aba que pode cadastrar uma doação



1- tela home



**2- tela de detalhamento**

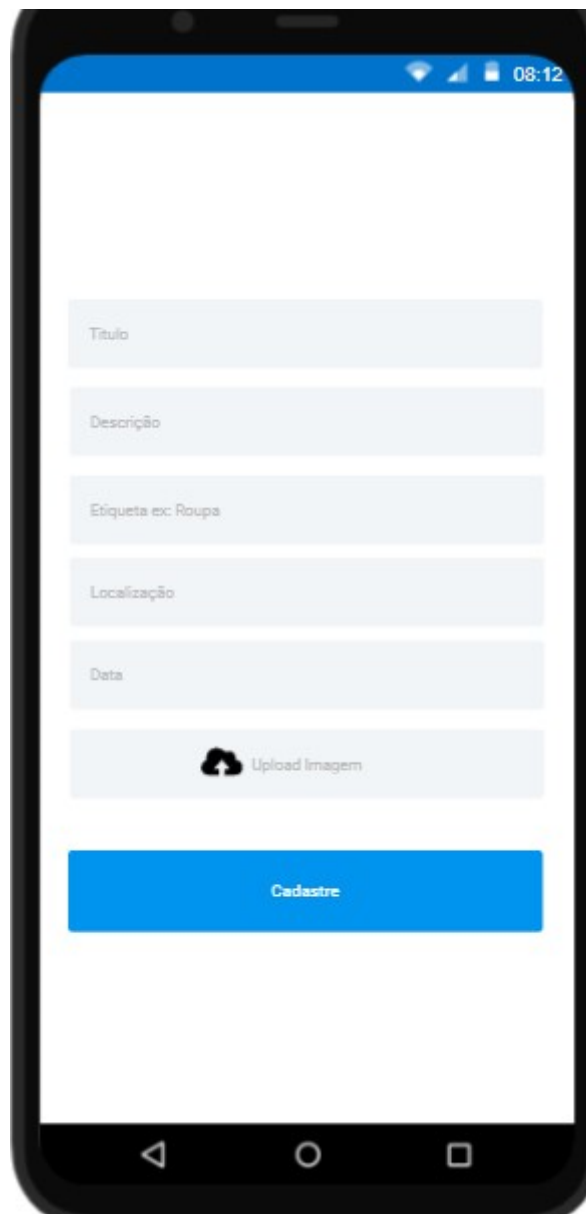


**3- tela login**





5- tela dashboard



**1- tela cadastro da doação**

## **7 LINK PARA REPOSITÓRIO E NOTION**

### **7.1 GitHub**

<https://github.com/jefo3/Projeto-Integrado-03>

## 7.2 Notion

<https://pointed-blanket-54c.notion.site/Projeto-Integrado-3147a494a74d47b6a6e2fb7a97b2bdf1>

## 8 DECISÕES DE ARQUITETURA

Para o projeto a linguagem escolhida será o java por alguns motivos, um dos principais motivos é que o projeto será todo em android nativo, e o que não falta é conteúdo na internet caso tenha alguma dúvida, com vários fóruns com uma comunidade ativa.

O firebase no sentido de servidor, fornece aos desenvolvedores uma variedade de ferramentas e serviços para ajudá-los a desenvolver aplicativos de qualidade, aumentar sua base de usuários e ser mais lucrativo. Toda sua base é construída na infraestrutura do Google, sendo categorizado como um programa de banco de dados NoSQL, que armazena dados em documentos do tipo JSON.

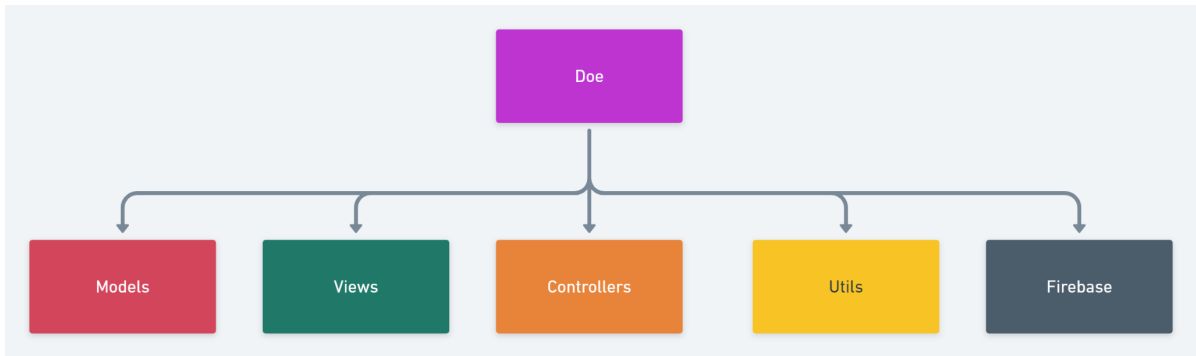
Em questão da arquitetura explicarei as decisões do back-end e mobile

**Arquitetura do Back-end:** E back-end que o mobile irá consumir será construído usando o firebase que é um Backend-as-a-Service (Baas)., essa decisão vem pela facilidade que o firebase trás, com sistema de autenticação, um banco de dados próprio, então no mobile irá consumir os dados do firebase como se fosse um api externa, que pode ser modificada dentro dele mesmo, assim não precisando construir uma API do zero com node por exemplo.

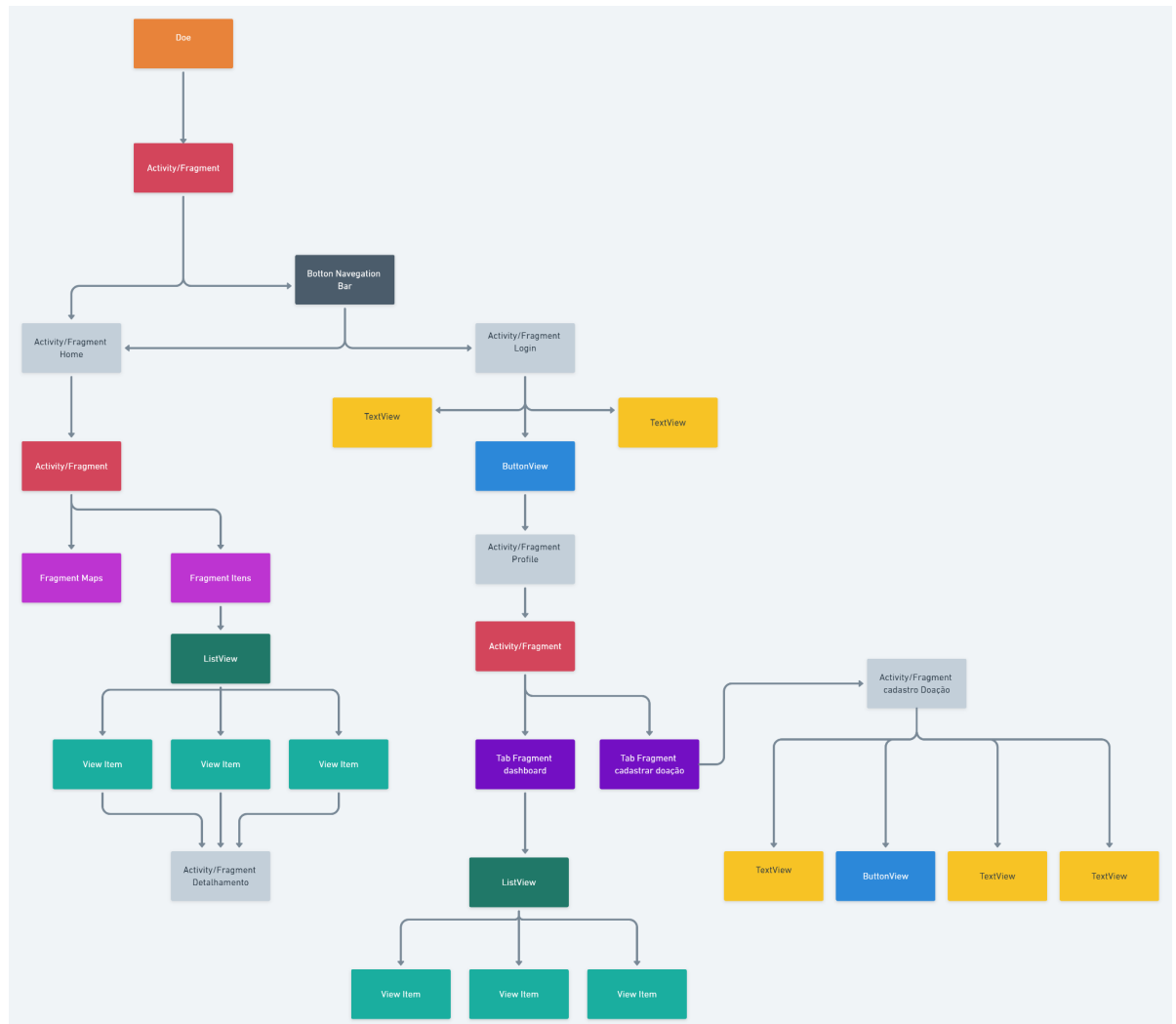
**Arquitetura Mobile:** E o front-end mobile será construído com o xml que será responsável por construir a parte gráfica, e com Java para aplicar a lógica de comunicação e a criação das interações, fiz um diagrama que explica como vai funcionar a arquitetura do projeto, para idealizar esse diagrama utilizamos um modelo baseado camadas. Segui o modelo de camadas baseado em controllers, services, models e views.

A divisão apresentada no diagrama a seguir foi feito pensando em garantir a manutenibilidade, coesão, e legibilidade do código

ARQUITETURA



## DIAGRAMA DE WIDGETS



## 9 BANCO DE DADOS

Criar corretamente um banco de dados estruturado exige planejamento. O mais importante é planejar como os dados serão salvos e depois recuperados para tornar esse processo o mais fácil possível, pensando nisso que resolvi utilizar o banco de dados disponibilizado pelo firebase, buscando toda essa facilidade que ele traz, e extraindo o máximo da ferramenta, essa ferramenta é o **Firestore Realtime Database** um banco nosql que diferente de um banco sql onde os banco são representados por tabelas e essas tabelas têm relacionamentos entre si, onde tem chave primária e chave estrangeira, o banco do firebase funciona da seguinte maneira, os dados são armazenados como objetos JSON. Pense no banco de dados como uma árvore JSON hospedada na nuvem, quando você adiciona dados à árvore JSON, eles se tornam um nó na estrutura JSON com uma chave

associada. É possível fornecer suas próprias chaves, como códigos de usuário e nomes semânticos

Esse foi um dos principais motivos pela escolha do banco NoSql do firebase, pela facilidade de implementação, por ser um projeto pequeno que pode vir a crescer muito, é bem mais fácil de trabalhar com NoSql, por toda flexibilidade que traz, e escalabilidade ao armazenar e acessar dados.

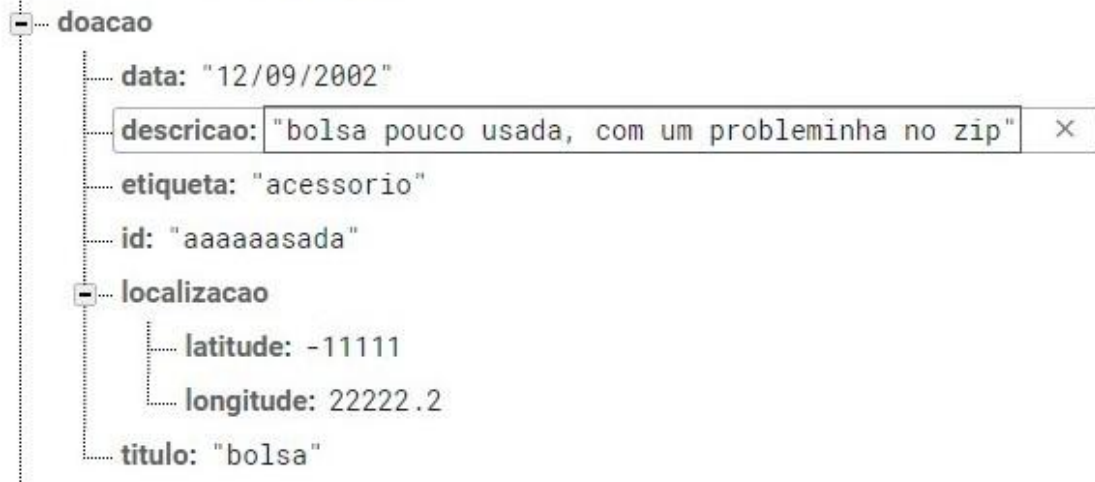
## ESQUEMA VISUAL DO DADOS

Para a modelagem dos dados no Firebase Realtime Database, faz-se necessário uma abordagem diferente da convencional, pois não temos relacionamentos diretos (chaves primárias ou estrangeiras) entre as coleções.

Apesar de não ter um relacionamento direto, podemos ter valores fazendo referências para outros objetos em outras coleções, como pode ser visto na figura, onde, cada registro da doação o usuário possui uma lista de doações que cadastrou. Essa lista, por sua vez, possui apenas o id da doação, que faz referência para os valores da coleção da doação.



projeto-doe-9f714-default-rtdb



## PADRÃO DE ACESSO AO DADO

Em questão de acesso ao dado o firebase tem funções próprias de acesso ao dado e envio, o que posso fazer é uma classe que será o padrão do dados a serem enviados, que seria o model do banco de dados, então todos dados enviados tem que ser naquele modelo, assim mantendo um padrão de persistência dos dados