

APLICAÇÃO PARA CONSULTA DE CONTRATOS PÚBLICOS FEDERAIS

Rayan M. P. Cordeiro¹, Vinícius Soares Jovito², Gabriel Bezerra dos Santos³, Felipe Augusto de Souza⁴.

Ciência da Computação – Instituto de Ensino Superior de Brasília (IESB)
70.200-730 – Brasília – DF – Brasil.

rayanmpcordeiro@gmail.com, gabrielbezerra38@hotmail.com,
viniciusjovito@gmail.com, felipe-augusto09@hotmail.com.

Abstract. *To present the Transparency of federal public agencies and entities to the population through an android application, providing information about the number of contracts, expiration dates, values, categorization, and suppliers.*

Resumo. *Apresentação da Transparência de órgãos e entidades públicas federais à população através de uma aplicação android, disponibilizando informações sobre quantidade de contratos, vencimentos, valores, categorização e fornecedores.*

1. Introdução

O ComprasNet 4.0 é um aplicativo mobile, tanto para funcionários públicos, quanto cidadãos comuns para fiscalização e consulta dos contratos públicos ativos, permitindo acesso aos dados, como valores, gestores e vencimento de todos os contratos ativos do governo federal de uma forma mais simplificada.

Com isso, o aplicativo possui uma interface amigável que auxilia usuários menos familiarizados a realizarem consultas em contratos do governo, filtrar por sua data de vencimento e consultar a localização dos órgãos responsáveis pelos mesmos.

2. Contextualização

O *Comprasnet Contratos* é uma ferramenta do governo federal que automatiza os processos de gestão contratual e conecta servidores públicos, foi criada para reduzir problemas com rotinas de trabalho e ter controle total sobre as informações, sendo uma plataforma sem custos para órgãos e entidades públicas. Enquanto o aplicativo tem como objetivo tornar essa ferramenta mais acessível, por meio de uma interface familiar a de apps conhecidos, tornando a utilização mais simples e direta.

Através do módulo de importação de dados do SIASG (Sistema Integrado de Administração de Serviços Gerais) o usuário terá acesso automático a todos os contratos ativos da sua unidade (UASG-Unidades de Administração de Serviços Gerais), permitindo que o usuário seja capaz de fiscalizar as compras públicas.

3. Problema

Ao consultar os dados no site *Comprasnet Contrato*, o usuário que não está familiarizado com assuntos relacionados a essa área, se depara com as informações dispostas de forma complexa e uma interface pouco intuitiva. À vista disso, notou-se a

necessidade de apresentar esses dados de modo que a experiência do usuário seja priorizada.

4. Objetivo Geral

Desenvolver um aplicativo que viabilize a transparência de contratos dos órgãos e entidades públicas federais à usuários da plataforma *Android*, tendo em vista maior acessibilidade entre os cidadãos brasileiros, para aumentar o alcance da população a informações de grande importância como quantitativo de contratos, vencimentos, contratos acabados, valores, categorização e fornecedores que estão envolvidos no fornecimento de materiais e serviços para o Governo Federal.

4.1 Objetivo Específicos

Visando atender ao objetivo geral têm-se os seguintes objetivos específicos:

- Cadastrar usuário;
- Consultar processos de compras;
- Consultar localização dos órgãos responsáveis pelos contratos em aberto.

5. Referencial Teórico

Esta seção foi estruturada em três tópicos, são eles: sistema integrado de administração e serviços gerais, tipos de dados fornecidos pelo SIASG, tecnologias usadas para construção do aplicativo.

5.1 Comprasnet Contratos

O Comprasnet Contratos é uma ferramenta disponibilizada por meio de um site do governo federal que automatiza a fiscalização de contratos, tornando informações disponíveis a qualquer momento. Seu público alvo são órgãos e entidades da administração pública federal.

5.2 Sistema Integrado de Administração e Serviços Gerais

Instituído no Decreto nº 1.094/1994, no art. 7º, o SIASG tem como objetivo ser um instrumento de apoio nas atividades do SISG (Sistema de Serviços Gerais), através da informatização e operacionalização de suas atividades. Sua operacionalização se dá através de diversos módulos que constituem o SIASG. (SIASG, 2020)

5.3 Tipos de dados fornecidos pelo SIASG

Os dados disponibilizados pelo SIASG, disponível no site API de Compras Governamentais, de 2021, são: dados dos fornecedores, materiais, catálogo de serviços, licitações, contratos, compras sem licitação, plano anual de contratações.

5.4 Tecnologias usadas para construção do aplicativo

Para o desenvolvimento do aplicativo ComprasNet, foi utilizado algumas tecnologias e ferramentas que auxiliaram em algumas funcionalidades. Para conhecimento das mesmas, esta seção será dividida em tópicos, são eles: linguagem, arquitetura, biblioteca e ferramentas.

5.4.1 Linguagem: *Kotlin*

De acordo com o site DevMedia, para a fase de desenvolvimento, foi escolhido o *Kotlin* como linguagem de programação, o *Kotlin* é uma linguagem que é executada na *JVM* (*Java Virtual Machine*), sendo assim, capaz de integrar em seu próprio código, a linguagem *Java*. O *Kotlin* também é uma linguagem do tipo estática, assim tornando mais seguro e prevenindo erros de compilação .

5.4.2 Arquitetura: *MVVMi*

A arquitetura *MVVMi* foi escolhida para o projeto, é importante destacar a importância da *ViewModel* e *LiveData* que são dois componentes de extrema importância dessa arquitetura, a *ViewModel* sobrevive ao ciclo de vida da atividade e as mudanças de configuração, a *LiveData* é quem armazena os dados gerados pelo ciclo de vida, ou seja, tudo que acontece no aplicativo em segundo plano fica armazenado na *LiveData*. É importante destacar também a camada *interactor*, ela é responsável por toda lógica de negócio. (Medium,2020)

5.4.3 Biblioteca e ferramentas: *Retrofit*, *Firebase*, *Firebase Realtime Database*, *Dialog Flow*, *API do Google Maps*

***Retrofit*:** Para as requisições foi utilizado a biblioteca *Retrofit*, essa biblioteca é capaz de criar requisições e utilizar o aplicativo com um *Client HTTP* (*HyperText Transfer Protocol*), porém é necessário que haja a deserialização, ou seja, transformar os dados obtidos da requisição em informação.

***Firebase*:** Para controle de acesso de usuário, criação de novo login, recuperação de senha, foi utilizado a biblioteca *Firebase*, essa biblioteca conecta a plataforma da Google que tem como objetivo auxiliar no desenvolvimento de aplicativos móveis, ajudando na criação, melhoria e expansão de aplicações.

***Firebase Realtime Database*:** Responsável por armazenar e sincronizar dados em um banco na nuvem *NoSQL* (*Not Only SQL - Structured Query Language*), sincronizando todos os clientes em tempo real e tendo a possibilidade de ser mantido enquanto o aplicativo está *offline*. Os dados são armazenados através do *JSON*, possibilitando o acesso a multi plataforma que compartilham uma instância do *Firebase Realtime Database*.

***Dialog Flow*:** Para criação de chatbot, foi utilizado o *Dialog Flow*, um plataforma de processamento de linguagem natural da *Google*, ele funciona com agentes (bots) e através de entradas e saídas de informações.

***Google Services*:** Na utilização de mapas, o *Google Services* foi escolhido para performar dentro do aplicativo *ComprasNet*, assim possibilitando que haja uma interação entre os órgãos e os contratos ativos.

6. Metodologia

O aplicativo foi desenvolvido na linguagem *Kotlin*, utilizando das ferramentas mencionadas na seção anterior, as telas são todas feitas em fragmentos com a navegação

entre eles ocorrendo por meio de uma barra de navegação e um *NavHostFragment* presentes na *MainActivity*.

Se ao iniciar o aplicativo o usuário ainda não tiver logado, ou seja, aberto o app pela primeira vez, será apresentado a três telas *Onboarding* que contam com uma apresentação das principais funcionalidades do app para que o usuário possa estar ciente do que é possível realizar. Após o término é levado a tela inicial do aplicativo.



Figura 1. Barra de Navegação.

No primeiro contato será vista a barra de navegação inspirada na barra de navegação do aplicativo *Google*, por ela será possível acessar rapidamente as principais funcionalidades disponíveis: navegação em contratos, autenticação, localização e acesso ao chatbot.

Para que ocorra a navegação pelos contratos é necessário que o usuário faça seu cadastro, criado e autenticado pelo *Firebase*, o fluxo aprofundaremos adiante. Após cadastro e *login* realizados, a funcionalidade de navegação é liberada e de possível utilização.

A tela inicial apresenta de forma direta as principais funcionalidades relacionadas a fiscalização dos contratos pelo usuário. Diretamente dela pode-se pesquisar contratos a partir de informações presentes nas descrições dos mesmos, mostrar todos os contratos sem filtro aplicado ou mostrar contratos a partir de filtros pré determinados baseados em seu vencimento.



Figura 2. Inspiração visual da tela inicial do *app* na página inicial do site *Comprasnet Contratos*.

As cores dos botões da tela inicial e a identidade visual de toda a interface são baseadas no site *Comprasnet Contratos* para que o público alvo da outra plataforma também possa encontrar facilidade em nosso aplicativo assim como o cidadão comum que é nosso público alvo.

Ao pesquisar ou clicar em algum dos botões da tela inicial é enviado um requerimento à *API* de *compras governamentais* por meio do *Retrofit* onde são passados os parâmetros específicos definidos pelo filtro escolhido. Após realizado este *GET* as informações são retornadas em um arquivo *JSON* que são lidas e passadas para *Interactor* onde são aplicadas as regras necessárias para que as informações sejam dispostas num *recyclerview*, para isto o *Adapter* pega as informações do *JSON* fornecido que agora foram organizadas em um *array* e os dispõem em seus respectivos *cards* dentro do *layout* de *recyclerview* similar a um *feed* de notícias. Ao clicar no card todos os detalhes que a *API* retorna são apresentados na tela.



Figura 3. Cards com informações dos contratos dispostas em um *recyclerview* similar a um feed de notícias

Sobre a estrutura visual do card, sua barra de título apresenta a *UASG* (código do órgão) do contrato e a sua cor é definida com base na data de vencimento do contrato, a cor segue o cor do botão de filtro na tela inicial, por exemplo, se o contrato irá vencer dentro de um período de 30 dias a cor do *card* será vermelha como o botão que filtra contratos que vencem um mês. Com isso o usuário consegue ter uma ideia da validade sem precisar clicar no contrato para ver os detalhes, relacionando assim o visual ao conteúdo.

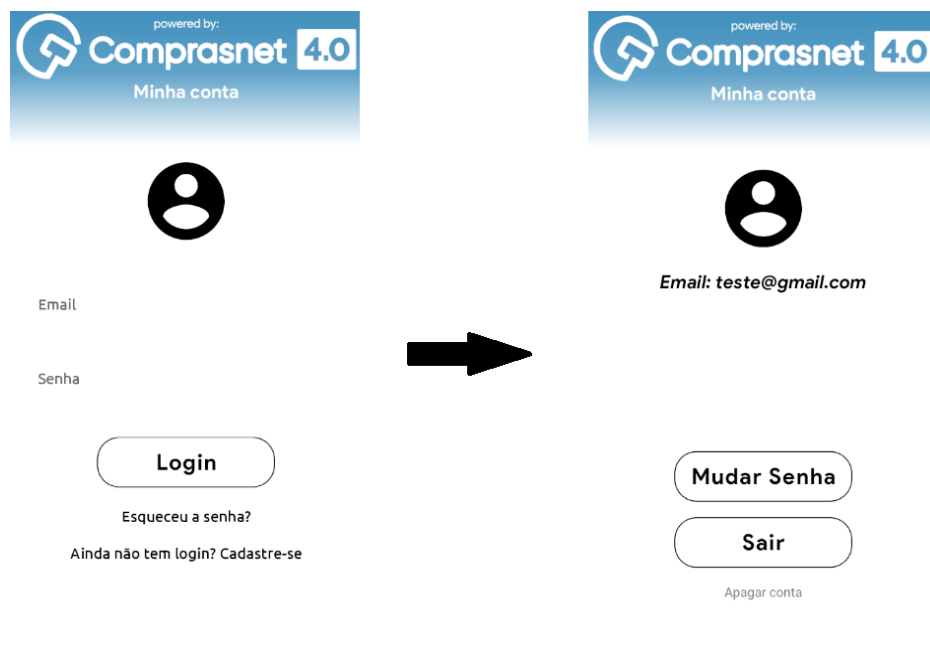


Figura 4. Fluxo de Login, tela de login a esquerda e tela com detalhes da conta a direita.

O segundo item da barra de navegação leva ao fluxo de login do app, por meio da implementação do *Firebase*, o usuário consegue sem maiores dificuldades se cadastrar com email e senha, realizar o login, alterar a sua senha ou apagar sua conta.

Utilizando o *auth().currentUser* disponibilizado pelo *Firebase*, o fluxo de *login* ao ser acessado pela barra consegue verificar se o usuário está logado, caso não, ele é direcionado a tela de *login* onde lhe é permitido se cadastrar ou logar, caso sim, ele é automaticamente redirecionado a tela de detalhes da conta onde pode alterar sua senha, apagar sua conta ou realizar o *logout*.



Figura 5. Fragmento de mapas.

O terceiro item da barra de navegação leva ao fragmento de mapas, nele é utilizado o *Google Play Services*, que solicita o mapa e seta a localização dos principais órgãos públicos gestores de contratos, após isso solicita a localização do usuário através da função *checkPermission()*, guarda-a no *locationManager* e chama o *setupLocation()* como última localização conhecida, até o próximo request da localização, tendo assim a localização em tempo real do usuário e possibilitando ao usuário traçar rotas até os pinos dos órgãos caso tenha o app do *Google Maps* instalado em seu celular.

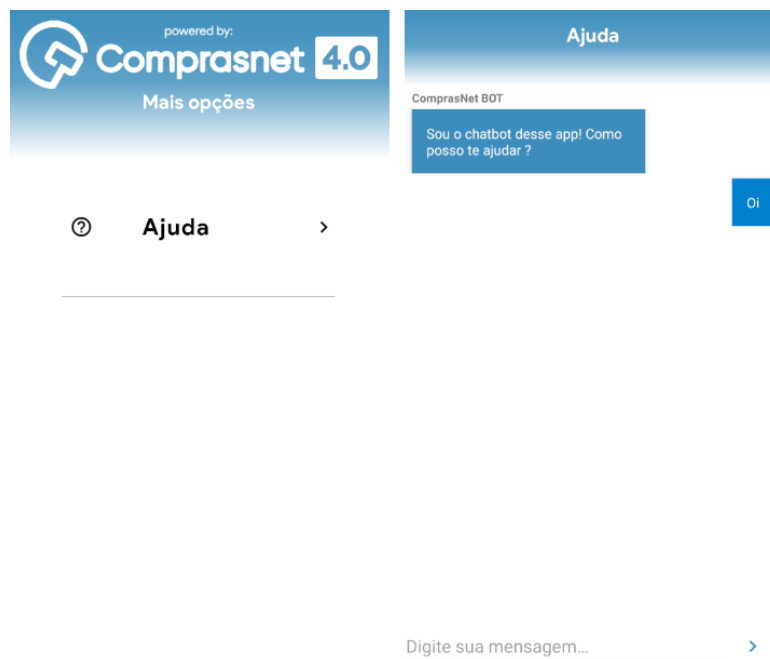


Figura 6. Fragmento Mais opções onde é possível acessar o chatbot.

O quarto e último item da barra de navegação leva ao fragmento Mais Opções onde é possível obter ajuda com o chatbot, por meio do *DialogFlow* implementado em um server no *Heroku*. É enviado a mensagem digitada pelo usuário ao servidor que retorna um *JSON* com a resposta do chatbot. Seguindo a mesma lógica *MVVMi* aplicada na coleta de dados da *API*, os dados recebidos são recebidos pela *repository*, passam pelo interactor e seus dados são dispostos em um adapter que formata as mensagens em um layout de chat.

7. Resultados Obtidos

Os resultados esperados durante o desenvolvimento do aplicativo, foram satisfatórios, em grande parte do projeto, funcionalidades previstas ao decorrer do desenvolvimento como requisitar a *API*, filtrar busca de contratos, manter fluxo de *login*, acessar a localização e informar os órgãos responsáveis, tiveram resultados positivos. Porém não teve o mesmo êxito com o assistente virtual.

8. Conclusão

A versão final do aplicativo manteve a identidade e quase todas as funcionalidades do protótipo. Fazendo a Requisição da *API* e filtrando os contratos. Cadastrando o usuário conforme o previsto e salvando-os na nuvem, além de coletar a localização do usuário em tempo real e informando os locais dos principais órgãos do governo. Já a função do chatbot foi implementada mas não se encontra operante por má administração do tempo.

9. Trabalhos Correlatos

Nesta seção será apresentado um trabalho que tem funcionalidades muito parecidas com a própria *API*, o *ContratosGov* que também tem o foco em gestão e fiscalização contratual.

9.1 ContratosGov

O *ContratosGov* é uma ferramenta gerencial que conecta de forma inteligente a atuação dos setores e agentes envolvidos na execução contratual, disponibilizando informações fundamentais e melhorando as condições operacionais da gestão e da fiscalização.

Sistema adquirido pela UFSCar (Universidade Federal de São Carlos) para gestão de contratos administrativos, onde são executadas algumas ações relativas à fiscalização, documentação e controle dos contratos UFSCar.

Fase 1

Após receber o contrato e realizar todos os procedimentos necessários, como reuniões preliminares e planejamento para uma fiscalização eficiente e eficaz. Onde o fiscal do

contrato deverá conferir e caso necessário alterar os registros. Permitindo controle sobre as informações financeiras do contrato

Fase 2

Durante a execução dos contratos são realizados os ateste de pagamentos. Após seguir os procedimentos de verificação dos serviços/entregas realizadas, o fiscal deverá registrar no sistema *ContratosGov* o ateste da Nota Fiscal.

10. Bibliografia

GOV. **Portal de Compras do Governo Federal**. Publicado em 07 de agosto de 2020. Disponível em: <www.gov.br/compras/pt-br/aceso-a-informacao/institucional/siasg> Acesso em: 28 de maio de 2021.

Api de Compras Governamentais. **SIASG**. Atualizado até janeiro de 2021. Disponível em: <<http://compras.dados.gov.br/docs/home.html>> Acesso em: 28 de maio de 2021.

DevMedia. **Guia completo de Kotlin**. Disponível em: <www.devmedia.com.br/guia/linguagem-kotlin/40739> Acesso em: 29 de maio de 2021.

Medium. **Arquitetura limpa com MVVMi, componentes de arquitetura e RxJava**. Publicado por: Luke Simpson. Disponível em: <medium.com/@thereallukesimpson/clean-architecture-with-mvvmi-architecture-components-rxjava-8c5093337b43> Acesso em: 29 de maio de 2021.

Medium. **Os princípios SOLID em imagens**. Publicado por: Ugonna Thelma. Disponível em: <medium.com/backticks-tildes/the-s-o-l-i-d-principles-in-pictures-b34ce2f1e898> Acesso em: 30 de maio de 2021.

Medium. **Consumindo API REST com Retrofit + Kotlin no Android**. Publicado por: Alifyz F. Pires. Disponível em: <medium.com/@alifyzfp/res/consumindo-api-rest-com-retrofit-kotlin-no-android-abba52820cc> Acesso em: 30 de maio de 2021.

Firebase. **Documentação**. Disponível em: <<https://firebase.google.com/docs/android/setup?hl=pt-br>> Acesso em: 30 de maio de 2021.

Medium. **Introdução - Room Database Android**. Publicado por: Gabriel Prado. Disponível em: <medium.com/android-dev-br/utilizando-room-database-no-android-fd76c2e6ccee> Acesso em: 30 de maio de 2021.

DialogFlow. **Documentação**. Disponível em: <<https://cloud.google.com/dialogflow/docs/?hl=pt-br>> Acesso em: 30 de maio de 2021.

Proad. **Fiscalização de contratos**. Disponível em:

<<https://www.proad.ufscar.br/pt-br/media/arquivos/manual-contratosgov.pdf>> Acesso em: 29 de maio de 2021

Proad. **Manual contratos Gov.** Disponível em:

<<https://www.proad.ufscar.br/pt-br/servicos/contratos/gestao-de-contratos/fiscalizacao-d-e-contratos>> Acesso em: 29 de maio de 2021

Contratos Gov. **Gestão e fiscalização de contratos.** Disponível em:

<<https://www.contratosgov.com.br/Home?ReturnUrl=%2f>> .Acesso em: 29 de maio de 2021

ComprasNet Contratos Disponível em:

<<https://contratos.comprasnet.gov.br/transparencia>>.Acesso em: 29 de maio de 2021