

Locany - Localization and Analytics

Fernando Tavares Nominato
Graduando em Sistemas de Informação - Uni-FACEF
fernandotnominato@gmail.com

Higor Canavez Jardini
Graduando em Engenharia de Software - Uni-FACEF
higorjardini07@gmail.com

Orientador Carlos Alberto Lucas
Mestre em Educação - Uni-FACEF
projetos@profcarloslucas.com.br

Resumo

Este projeto traz como prioridade o desenvolvimento de uma ferramenta que auxilie o usuário a encontrar um CEP (Código de Endereçamento Postal) ou endereço específico por meio do preenchimento parcial da informação, Para isso, o sistema fornece as opções de cep ou endereços disponíveis de acordo com as informações que estão sendo inseridos. O software trará uma interface simples, disponibilizando um campo para inserir dígitos de um cep ou então um endereço específico na qual retornará as opções de cep/endereço referentes às informações inseridas pelo usuário. O projeto pretende adicionar funcionalidades extras no decorrer do desenvolvimento, como o fornecimento do endereço de estabelecimentos comerciais presentes em uma determinada faixa de CEP.

Palavras-chave: Endereço, CEP.

Abstract

This project presents the development of an engine that helps the user to find a zip code or a specific address through the partial information filling. For that, the system provides the zip codes or addresses options according to the information that is being inserted. This software has a simple interface, which provides an input field so the user can insert digits or a single address and then returns the options related to the searched information. The project pretends to add extra functionalities during the development, as the provision of any commercial establishments available around a specific zip code area.

Keywords: Address, Zip code.

1 Introdução

Qual a dificuldade encontrada pelo usuário ao utilizar sistemas que solicitam a informação referente à um determinado CEP em serviços online como, cadastro e compras em sites de *e-commerce*, busca de endereços comerciais, consulta em plataformas governamentais?

O objetivo deste projeto é desenvolver uma ferramenta que auxilie o usuário a encontrar um endereço específico por meio do preenchimento de poucas

informações. O sistema deve fornecer as opções de endereço disponíveis a partir dos primeiros dígitos obtidos, indicando a localidade que cada faixa de CEP representa.

Foi identificada uma possível dificuldade do usuário em saber o próprio CEP ao realizar o preenchimento de cadastros em plataformas de vendas online, ou então o uso de serviços online que solicitam a informação referente à um CEP. Também, a falta de conhecimento de um determinado CEP referente à uma localização/região que ainda não possui familiaridade.

As seguintes ferramentas foram utilizadas para a realização do trabalho:

- Utilização da linguagem backend Golang;
- Utilização de sistemas relacionados à banco de dados como MySQL e
- RabbitMQ;
- Spring Boot;
- Docker e Docker Compose.

2 Referencial Teórico

O referencial teórico foi elaborado por meio de pesquisa e conhecimento obtido durante a leitura do referencial, com o objetivo de adquirir informações sobre o tema deste projeto. A partir de reuniões, foram definidos os principais pontos a serem buscados e abordados nos tópicos a seguir.

2.1 Conhecendo a informação

O código de endereçamento postal (CEP) é um conjunto de números constituídos de 8 dígitos os quais têm a função de orientar e acelerar os processos de tratamento e distribuição de objetos de correspondência por meio de sua atribuição a localidade, logradouros, serviços etc.

2.1.1 História

Criado em 1971, pela Empresa Brasileira de Correios e Telégrafos, o Código de Endereçamento Postal (CEP), tem como principal objetivo a orientação e facilitação do tratamento e distribuição dos itens geridos pelos Correios. O código auxilia neste processo pois é composto de um resumo das informações referentes à lógica político-administrativas brasileiras, identificando o estado, cidade e região, incluindo neste último o bairro e a rua quando se trata de uma grande cidade. Em um primeiro momento, este código foi desenvolvido para substituir as Zonas Postais ou Zonas de Correio, criadas pelo Departamento de Correios e Telégrafo (DCT) vinculado ao Ministério de Viação e Obras Públicas, que auxiliou na viabilização do projeto.

O CEP foi divulgado ao público no mesmo ano de sua criação e a partir daí passou a ser utilizado em todo o território nacional como requisito obrigatório para a utilização de serviços prestados pelos Correios. Além disso, atualmente, a maioria de serviços e sistemas digitais que solicitam um cadastro, exige a informação referente ao CEP, independentemente se irá utilizar algum sistema de envio/entrega ou não. Sendo assim, é de suma importância preencher corretamente a informação, visto que a própria empresa destaca a relevância em seu website, para que seja feito o uso adequado para a execução de sua própria atividade:

“O uso adequado do CEP é imprescindível para que os Correios possam tratar com rapidez os objetos que lhe são confiados, racionalizando os métodos de separação ao permitir o tratamento mecanizado (triagem, encaminhamento e distribuição) com a utilização de equipamentos

eletrônicos de triagem. Com o CEP inadequado ou errado, as correspondências e encomendas demandam mais tempo para serem devidamente separadas e entregues [...]” (CORREIOS, 2022).

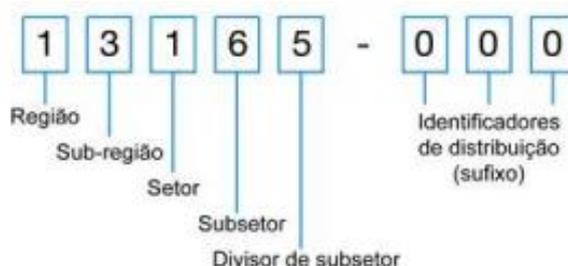
2.1.2 Composição

Inicialmente, o código de endereçamento postal foi criado apenas com 5 dígitos e esta primeira configuração durou até o ano de 1992, quando foram inseridos mais 3 dígitos à formação original, deixando-os separados por um dígito “-“. Portanto, em sua formação final, ficou definido da seguinte forma: 00000-000, que os 5 primeiros dígitos representam: a Região, Sub-região, Setor, Subsetor e Divisor de Sub Setor; e os três últimos representam os indicadores de distribuição, chamados de sufixos, como demonstrado na Figura 1. Estes que são designados para a identificação individual de localidades, logradouros, códigos especiais e unidades dos Correios.

Também vale ressaltar que existem cidades que não atingem os requisitos mínimos para se obter mais de um código postal, como o número mínimo de habitantes, que é de 50 mil. Portanto, a cidade inteira é identificada por um único CEP, que é válido para todos os bairros e logradouros e por isso, exige um extenso e detalhado mapeamento por parte da prefeitura da cidade.

O CEP único ainda pode ser encontrado em muitas cidades no interior do país. Para que uma cidade passe a utilizar a identificação de logradouros, é necessário que o Censo Populacional identifique nesta área o número mínimo de habitantes e então a prefeitura informa aos Correios todos os endereços da cidade e cada localidade passa a ser identificada por um CEP único.

Figura 1 – Setores.



Fonte: Site dos Correios, 2002.

Para que os códigos fossem distribuídos, o país foi dividido em 10 regiões postais, utilizando como parâmetro o desenvolvimento socioeconômico e fatores de desenvolvimento demográfico de cada Unidade Federativa ou conjunto delas. Após a formatação final, os códigos foram distribuídos de forma anti-horária, começando pelo estado de São Paulo, a partir do primeiro algarismo, conforme apresenta a Figura 2.

Figura 2 – Mapeamento.



Fonte: Site dos Correios, 2022.

- Número 0: Cidade de São Paulo e Região Metropolitana;
- Número 1: Interior do estado de São Paulo;
- Número 2: Estados do Rio de Janeiro e Espírito Santo;
- Número 3: Estado de Minas Gerais;
- Número 4: Estados da Bahia e Sergipe;
- Número 5: Estados de Alagoas, Paraíba, Pernambuco e Rio Grande do Norte;
- Número 6: Estados do Acre, Amapá, Amazonas, Ceará, Maranhão, Pará e Roraima;
- Número 7: Distrito Federal, Goiás, Mato Grosso, Mato Grosso do Sul, Rondônia e Tocantins;
- Número 8: Paraná e Santa Catarina;
- Número 9: Rio Grande do Sul.

2.2 Interface de Programação de Aplicação

Mais conhecido como API, o termo *Application Programming Interface* diz respeito a um conjunto de rotinas e padrões que facilitam a comunicação e a troca de informações entre sistemas.

O termo API é utilizado para atribuir uma interface de comunicação que um sistema oferece para que outros sistemas possam acessar as suas funções, dados e recursos sem que este software necessite de desenvolver a funcionalidade desejada do zero. Geralmente uma API é criada quando uma empresa da área da tecnologia detecta a necessidade de que outros criadores de software desenvolvam produtos associados a seu serviço.

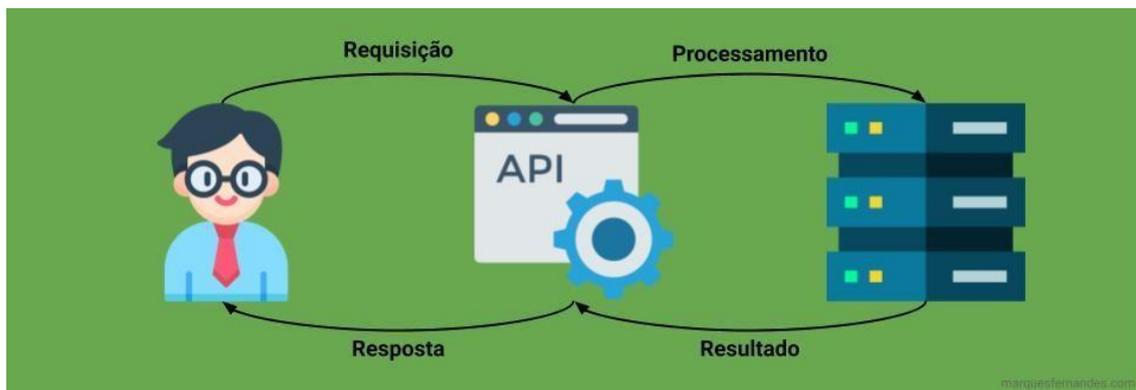
2.2.1 Como funciona

Como um exemplo, páginas de reservas de Hotéis oferecem um mapa para melhor identificação do local do estabelecimento, este mapa é utilizado por meio de uma API, os desenvolvedores do site do hotel utilizam o código de sistemas como o *Google Maps*.

Por meio das APIs, os softwares e aplicativos podem se comunicar sem a intervenção direta dos usuários, já que a interação é feita diretamente pelos códigos dos sistemas em questão, que definem comportamentos e respostas específicas dos objetos destacados em uma interface. Portanto a API, conecta as diversas funções em um site de maneira que possam ser utilizadas em outros sistemas. É muito comum a utilização de APIs em *plug-ins*, que servem para complementar a funcionalidade de um certo programa, ou seja, os desenvolvedores de um programa criam uma API específica e fornecem para outros desenvolvedores, que criam *plug-ins* para aumentar as funcionalidades relacionadas ao programa.

Quando uma API é bem construída e documentada corretamente, a manutenção é deveras facilitada e provém uma maior segurança às regras de negócio dos usuários. Outra vantagem da utilização de APIs é a promoção de inovação em produtos e serviços já existentes, gerando parcerias entre empresas e possibilitando o desenvolvimento de soluções personalizadas a quem deseja utilizar este sistema fornecedor. A Figura 3 mostra o ciclo de funcionamento de uma API.

Figura 3 – Como funciona a API.



Fonte: Google Imagens, 2022.

2.2.2 Construção

O desenvolvimento de uma API vai depender da necessidade do desenvolvedor e empresa que está responsável, podendo ser desenvolvida a partir de qualquer linguagem de programação e geralmente com um formato pré-definido de dados para o compartilhamento de informações entre os sistemas, como XML ou JSON.

Para APIs Web, existe um padrão adicional chamado REST (*Representational State Transfer*), que significa “Transferência Representacional de Estado” ou outro termo conhecido como *API Restful*. Este termo representa uma abstração da arquitetura Web composta por um conjunto de regras e padrões, que, quando bem definidas, possibilitam o desenvolvimento de projetos com interfaces bem definidas. Dentre os possíveis requisitos, geralmente as solicitações são gerenciadas pelo protocolo HTTP (*Hyper Text Transfer Protocol*).

3 Engenharia de Software

Nesta seção são mostrados os artifícios de Engenharia de Software utilizados como base para confecção deste trabalho.

3.1 EAP

Conforme Figura 4, a Estrutura Analítica do Projeto foi baseada em cinco pilares. No primeiro momento, o planejamento inicial definiu os primeiros documentos a serem elaborados para que ficasse claro quais as definições e especificações do projeto. Para isso foi elaborado o Termo de Abertura de Projeto, onde foi especificado qual seria o problema encontrado e as ferramentas para solucionar o mesmo, justificativa e objetivos.

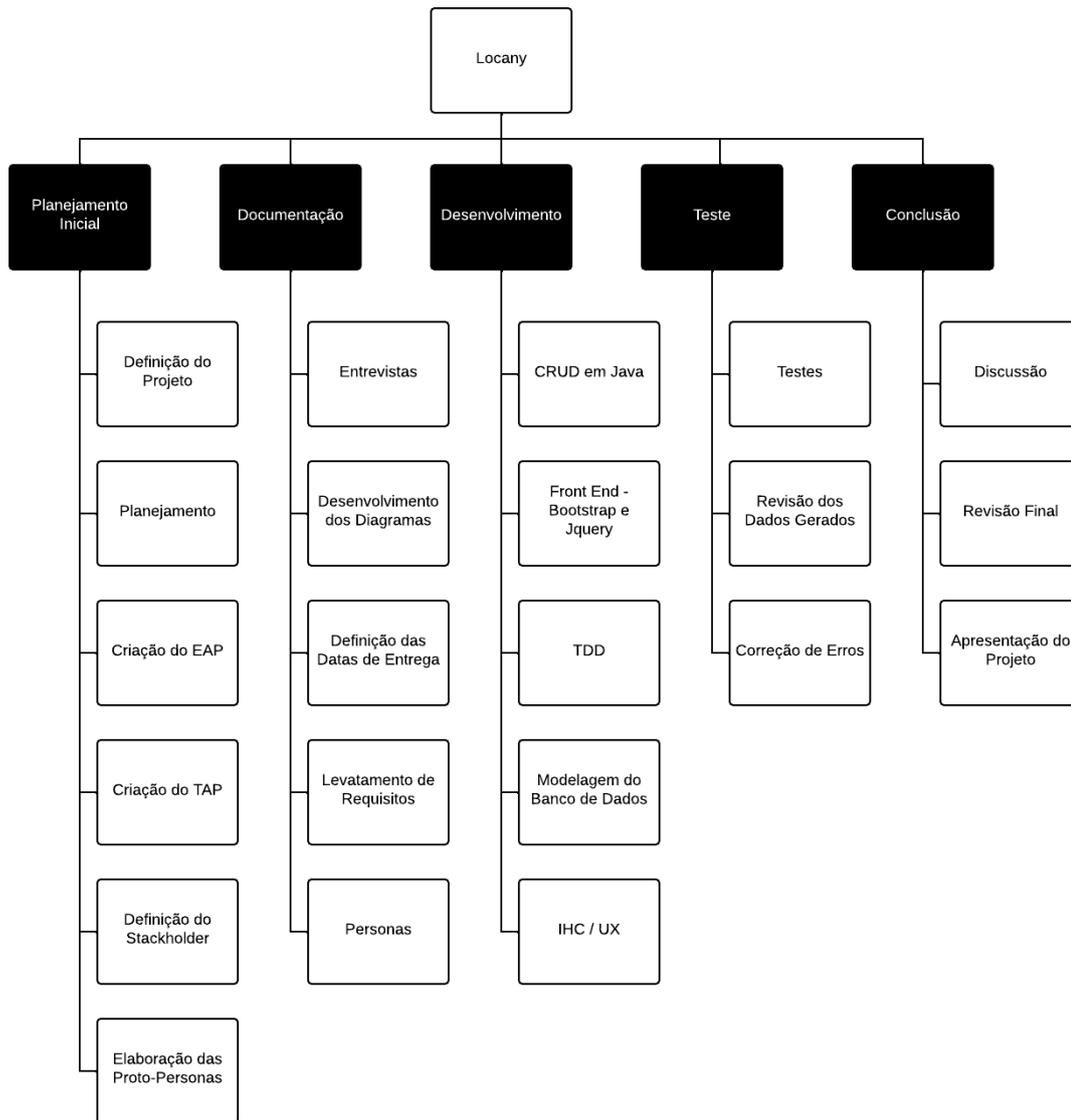
A partir disso, foi criado todo o cronograma de atividades a serem realizadas, que detalhou o tempo destinado para a criação de cada artefato. Com isso, foi possível visualizar o projeto como um todo e qual o tempo disponível para a realização do mesmo.

O próximo passo seria definir quais as ferramentas específicas seriam utilizadas para o desenvolvimento do sistema, para que facilitasse o estudo e pesquisa sobre o funcionamento das mesmas. Então, como prioridade foi escolhido RabbitMQ para geração de filas para processamento de requisições, para o desenvolvimento backend foi utilizado o Spring Boot, que se trata de um framework de Java e também Golang para processamento final da requisição. Para banco de dados, foi escolhido o MySQL, além da utilização de Docker e Docker Compose, para controle de *build* dos contêineres.

Também foi definida a etapa de testes, para garantir que o funcionamento do sistema estaria de acordo com o programado e para que fosse possível avaliar todos os dados obtidos e caso necessário, aplicar a correção de erros.

E como etapa final, foi definida a discussão de resultados, revisão final do projeto e apresentação do projeto. Após apresentação, aplicar todas as correções e orientação obtidas a partir das avaliações da banca presente. Estas correções deverão ser aplicadas antes da entrega final do projeto, para que todas as questões fiquem alinhadas e sejam entregues conforme o desejado.

Figura 4 - EAP.



Fonte: Acervo Próprio.

3.2 Personas

Para que fosse possível identificar o público alvo deste projeto, foi necessário elaborar o documento de *personas*, que detalha as características e descritivos dos usuários que o projeto deseja atingir. Os principais padrões de usuários buscados são aqueles que não possuem conhecimento de localizações específicas e precisam buscar muitos endereços de locais diferentes, seja para trabalho e lazer, quanto para quaisquer outras necessidades pessoais. A seguir, exemplos destas *personas*:

Figura 5 - Persona Mulher.



Fonte: Google Imagens, 2022

- **Nome**
 - Eliane Santos
- **Apelido**
 - Eli
- **Idade**
 - 31
- **História (Descrição)**
 - Eli tem 31 anos, é professora em uma escola pública. Atualmente vive em São Paulo e reside em um pequeno apartamento na zona leste da capital. Solteira e sem filhos, tem um gato de 4 anos chamado "Meow".

Eli leciona matemática para crianças e adolescentes em uma pequena escola localizada a 2 ruas de sua casa, na qual consegue uma renda mensal de R\$2.557,74.

Eli desenvolveu um problema de consumismo devido ao período de pandemia, no qual foi necessário ficar em casa. Adora conhecer locais novos na cidade, pois por conta do trabalho, é necessário que se mude de residência uma vez por ano.

Eli adora praticar esportes, caminhar na parte da manhã e assistir filmes junto ao seu gato "meow" experimentando novas culturas da gastronomia.

- **Objetivos**
 - Fazer compras online.
 - Pedir comida por meio de um aplicativo de delivery.
 - Localização para praticar esportes.
- **Problemas**
 - Se muda com muita frequência.
 - Trabalha em vários lugares durante o ano.
- **Solução**
 - Localização por meio de informações parciais.
 - Saber apenas o nome da rua.
 - Sugestão de lugares.

Figura 6 - Persona Homem.



Fonte: Google Imagens, 2022

- **Nome**
 - José Inácio
- **Apelido**
 - Cantagalo
- **Idade**
 - 61
- **História (Descrição)**
 - José, ou mais conhecido entre os amigos de longa data como o famoso “Cantagalo”. Homem simples, viveu a vida toda na fazenda com sua família, a qual acabou herdando de seu falecido pai.
José está com problemas administrativos devido a uma plantação de café que acabou saindo do controle de produção e fez com que ele viesse constantemente à cidade em busca de pessoas para ajudar no carregamento da plantação.
Nessas vindas para a cidade acabou se hospedando em um pequeno hotel ao sul da cidade Santa Rita do Sapucaí onde seguiria viagem para a capital de seu estado, Belo Horizonte - MG.
Porém, ao parar no hotel para descansar se viu em um dilema, precisava fazer um cadastro em um site para aluguel de caminhões com o objetivo de fazer carregamentos de toda a plantação cujo cadastro era solicitado a ele informações de localização para descarga de toda a carga após a torra do café.
- **Objetivos**
 - Localização de barracões para despacho de grandes quantidades de carga.
 - Informações sobre rotas para viagem da plantação.
- **Problemas**
 - Não reside na cidade onde irá fazer a entrega da carga de café.
- **Solução**
 - Localização em meios comerciais.
 - Falta de informação devido a falta de conhecimento do lugar.
 - Não sabe mexer em sistemas complexos de localização.

4 Ferramentas de Desenvolvimento

Nesta seção são mostradas as ferramentas utilizadas como base para desenvolvimento deste software.

4.1 Spring Boot

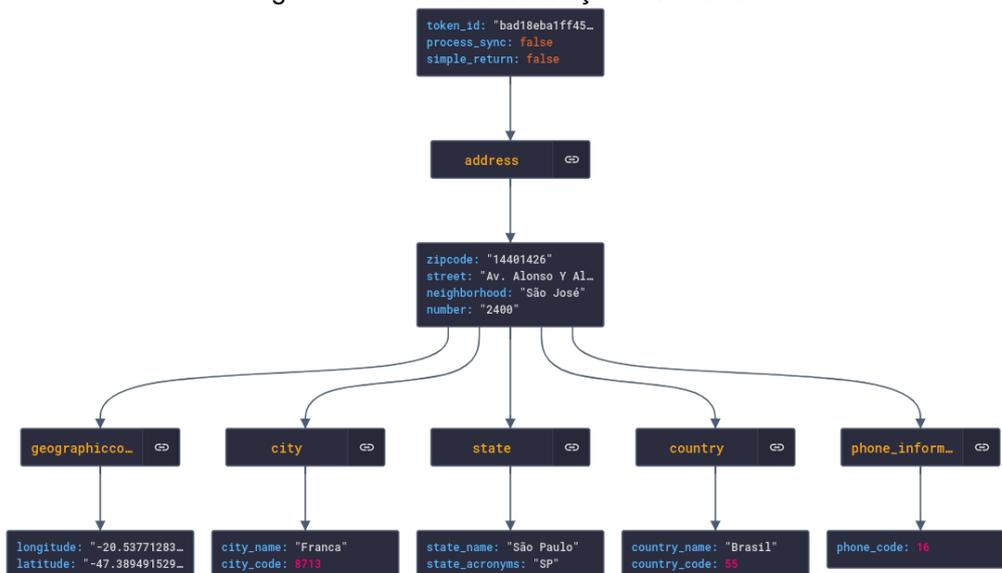
O *spring boot* é um framework Java *open source* que tem como objetivo a facilitação de processos e aplicações em Java, trazendo mais agilidade para o desenvolvimento, uma vez que não é necessário gastar muito tempo com configurações iniciais. Com ele, é possível facilitar a configuração de ambientes como servidores, bibliotecas, métricas, dependências, entre outros.

Esta ferramenta trabalha em um modelo de convenção sobre configuração, ou seja, a mesma decide qual a melhor forma de algo ser feito dentro daquilo que é desejado, aplicando configurações padrões para facilitar o trabalho. Além disso, é um ferramenta flexível, que permite a personalização destas 'configurações *default*', para que se adeque às preferências do usuário.

O *spring boot* permite o agrupamento de suas dependências com outras, que possuem um mesmo propósito, como por exemplo a construção de soluções de mensageria, baseada em outra ferramenta utilizada neste projeto, o *RabbitMQ*.

Neste projeto, o *Spring Boot* foi utilizado como principal meio de comunicação entre os serviços gerais do projeto, por meio de requisições com o auxílio de *endpoints* (Um tipo de nó de comunicação) da API. O *Spring Boot* recebe um *JSON*, contendo informações gerais sobre um endereço buscado pelo usuário e um *Token* de autenticação e validação de serviço, que tem o objetivo de garantir uma certa rastreabilidade e confiabilidade para que seja possível identificar de onde vem a solicitação e os possíveis *endpoints* de retorno. Estas informações são enviadas para o *RabbitMQ*, para que o processo continue. Como mostra a Figura 7, as informações que o usuário preenche são recebidas pelo sistema por meio de um *JSON* (*Javascript Object Notation*).

Figura 7 - JSON de Informações Recebidas.



Fonte: Acervo Próprio.

Após o tratamento das informações por meio do processo citado no decorrer deste texto, o sistema também retorna um *JSON* com as informações que devem ser fornecidas ao usuário, conforme demonstrado na Figura 8.

Figura 8 - JSON de Informações Retornadas.



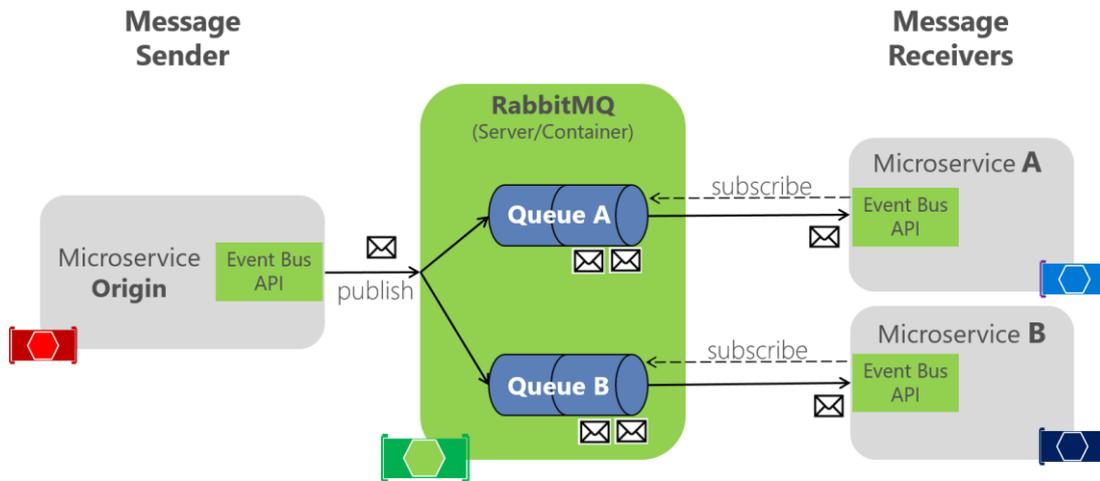
Fonte: Acervo Próprio.

4.2 RabbitMQ

O *RabbitMQ* é uma ferramenta *open source* conhecida como “*message broker*”, que permite a comunicação entre sistemas de forma assíncrona, ou seja, ele age como um intermediário que processa as “mensagens” entre produtores e consumidores, além de gerenciar “filas”, que organizam as informações e possuem diversas opções de encaminhamento destas mensagens. Na Figura 9 está demonstrado de forma resumida, como é o funcionamento desta ferramenta.

Neste projeto, o uso da mesma foi imprescindível para otimizar a troca de informações, durante o recebimento e fornecimento delas. Quando o sistema recebe as informações que foram preenchidas pelo usuário, o intuito é otimizar e agilizar ao máximo o retorno destas informações para que o usuário receba de fato as “respostas” as quais ele busca.

Figura 9 - RabbitMQ.



Fonte: Google Imagens, 2022.

Neste projeto, o *RabbitMQ* recebe o JSON contendo as informações de endereço, armazenando as mesmas em filas para que sejam enviadas de forma gerenciada e ordenada para *Golang*. Após isso, recebe novamente as informações, desta vez do sistema, como forma de “resposta”, para que sejam novamente ordenadas em filas e enviadas de volta para o *spring boot*, respondendo então, a requisição feita ao sistema.

4.3 Golang

É uma linguagem de programação backend de código aberto criada em 2007 por colaboradores do *Google*, que estavam insatisfeitos com linguagens que não ofereciam os recursos necessários para alguns projetos da empresa. Em um primeiro momento, foi criada para ser utilizada apenas internamente, mas seu potencial de usabilidade permitiu a abertura de capital, que em 2009 foi disponibilizado para o público em geral.

Neste projeto, o *Golang* é utilizado para realizar todo o processamento de informações, análise das regras de negócio e tratativa dos dados, por exemplo, garantir que um CEP, caso seja preenchido, siga o padrão da quantidade e formato dos números (com base no padrão citado anteriormente neste trabalho), ou então que o endereço realmente exista, se a cidade é compatível com o estado, caso contrário emite uma mensagem de erro, solicitando o preenchimento com novas informações, entre outros.

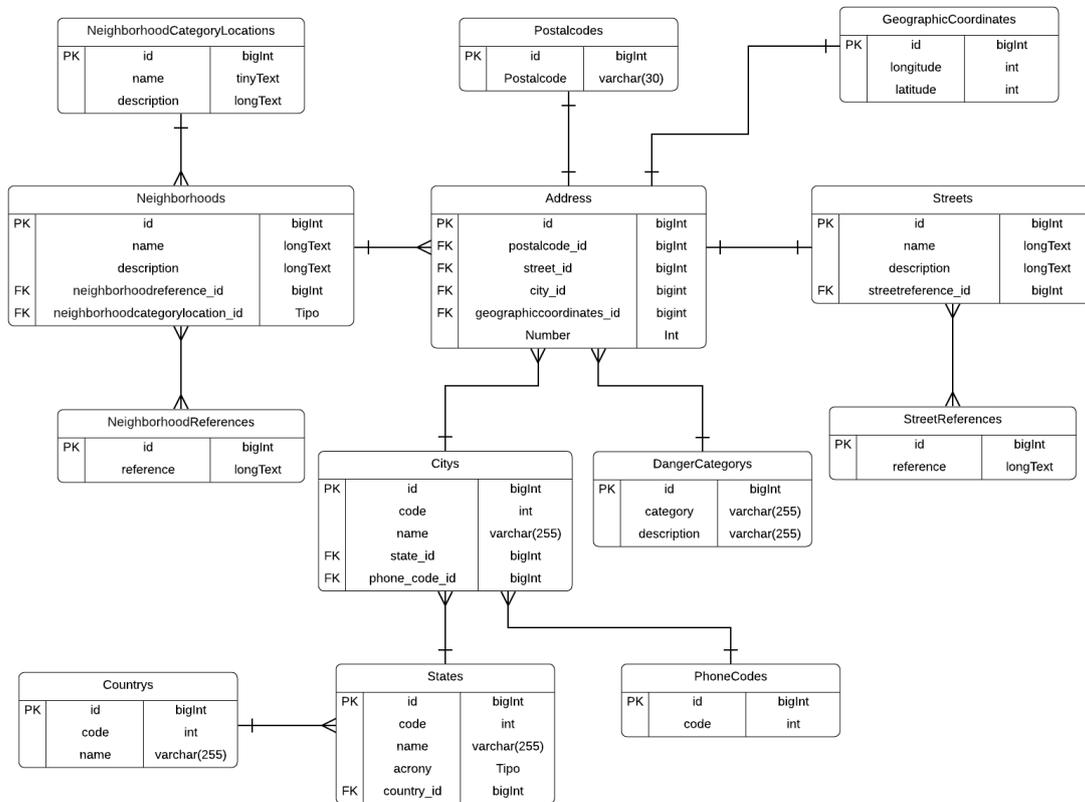
Após a aplicação das regras de negócio, estas informações são consultadas e cruzadas no banco de dados, para que seja possível obter um retorno compatível com a requisição realizada.

4.4 MySQL

Trata-se de um SGBD (sistema de gerenciamento de banco de dados) relacional de código aberto, que funciona com uma modelagem baseada no padrão cliente-servidor. Este sistema permite que sejam realizadas ações como, pesquisa de dados (buscar informações específicas), manipulação de dados (adicionar, apagar, alterar, modificar etc), identidade de dados (delimitar tipos) e controle de acesso de dados (proteção de dados através de acesso).

Para este projeto, o banco de dados irá armazenar toda e qualquer informação a respeito de endereços como, rua, bairro, número, cep, cidade, estado, país e seus respectivos atributos, conforme imagens abaixo.

Figura 10 - DER (Diagrama Entidade Relacionamento).



Fonte: Acervo Próprio.

4.5 Docker e Docker Compose

Docker é uma ferramenta open source que possibilita o desenvolvimento, envio e execução de aplicativos. Em resumo, o *Docker* possibilita a virtualização de uma máquina e criação de um ambiente isolado que agiliza o processo de entrega de um software, pois permite que as aplicações da infraestrutura sejam distribuídas em "contêineres" sem que interfira no restante do sistema. Estes, realizam um

conjunto de processos que são executados a partir de uma imagem, na qual fornece todos os arquivos necessários. Já o *Docker Compose*, é uma ferramenta de administração de contêineres que determina como os mesmos irão se comportar entre si, dentro de um sistema.

5 Pesquisa

Como forma de garantir a melhor identificação do público alvo, foi elaborada uma pesquisa quantitativa destacando perguntas específicas, com o intuito de identificar a real necessidade de um usuário em obter informações relacionadas ao CEP e qual a dificuldade em encontrar tais informações.

Esta pesquisa foi fornecida massivamente para usuários de serviços de transportadoras, identificados a partir de listas de *leads* obtidos por meio de contato pessoal. Também foi divulgada em fóruns online, em tópicos específicos de *e-commerce* e comunidades de vendedores online.

Com isso, foi possível a visualização, em percentual, de usuários que necessitam dessas informações para utilização em atividades pessoais e profissionais.

5.1 Resultados

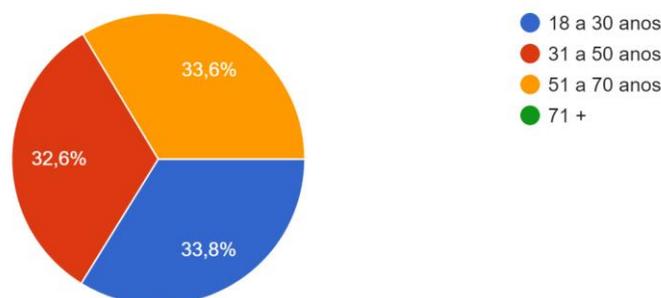
Nesta seção são mostrados os resultados obtidos por meio da pesquisa quantitativa.

5.1.1 Faixa de Idade

Como primeiro ponto, foi necessário entender a média de idade dos usuários que tiveram acesso à pesquisa. A partir de uma pergunta simples, destacando faixas de idade pré-definidas, obtivemos o seguinte resultado.

Figura 11 - Faixa de Idade.

Qual a sua idade?
506 respostas



Fonte: Acervo próprio / Resultado de Pesquisa.

5.1.2 Costume de Compra

Como aponta a Figura 12, a necessidade era entender se os usuários estão ligados às compras online. A partir disso, foi possível concluir que a grande maioria possui o costume de realizar compras na internet no geral. Levando em consideração e-commerce, marketplaces, vendas privadas etc.

Figura 12 - Costume de compra.

Você tem o costume de comprar pela internet?

506 respostas



Fonte: Acervo próprio / Resultado de Pesquisa.

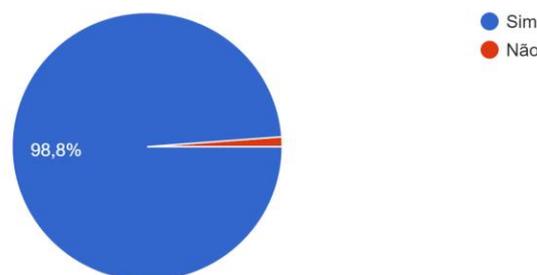
5.1.3 Compra para terceiros

Como sequência da pergunta anterior, obtivemos o resultado que detalha a porcentagem de usuários que têm o costume de realizar compras destinadas a terceiros. Com isso, é possível supor que existe a dificuldade em ter a informação sobre o CEP específico necessário. Aliado a isso, tentamos entender se realmente existe tal dificuldade na hora de preencher um CEP que não seja o próprio, conforme imagens abaixo.

Figura 13 - Compras a terceiros.

Você costuma fazer compras destinadas à terceiros na internet?

506 respostas



Fonte: Acervo próprio / Resultado de Pesquisa.

Figura 14 - Dificuldades

Você já teve dificuldades em algum processo relacionado aos serviços de localização em sites ou softwares de sua empresa?

506 respostas



Fonte: Acervo próprio / Resultado de Pesquisa.

A pesquisa conta também com outras questões relacionadas ao uso de informação de endereço destinados a tarefas profissionais, portanto como resguardo, foi decidido mantê-las privadas até o presente momento.

6 Planos e Expectativas

Como destacado durante o decorrer deste projeto, o principal intuito do mesmo era fornecer uma solução específica, tanto para usuários intermediários, como para usuários finais, ou seja, para aqueles que procuram uma ferramenta de busca de CEP diretamente ou então para aqueles que oferecem essa opção através de uma plataforma própria, como um e-commerce, para que os usuários finais utilizem.

Porém, como forma de buscar evolução e novas funcionalidades, foram definidas novas opções que poderiam ser implementadas, conforme listadas abaixo.

6.1 Estrutura de Informações

Desenvolver uma nova estrutura de recebimento e repasse de informações sobre setores diferentes, para facilitar o entendimento e distribuição dos dados recebidos. Como exemplo, informações sobre as transações financeiras, são repassadas de forma separada para não ter conflito com informações específicas de endereço etc.

6.2 Pontos de Interesse

A partir do endereço/CEP buscado, a ferramenta traria informações sobre o que existe de comércio, hotéis, restaurantes, recreação, praças públicas em geral,

nos arredores do mesmo, para auxiliar pessoas que buscam novos locais de moradia e buscam entender de forma geral o bairro que estão procurando.

6.3 Pontos de Coleta

Buscando auxiliar usuários que utilizam a informação como forma de trabalho, a ferramenta traria resultados que indicariam os pontos de coleta/ entrega mais próximos do endereço/CEP buscado.

7 Conclusão

A partir do planejamento detalhado deste projeto, foi possível construir uma ferramenta que cumprisse com o objetivo proposto. Para isso ocorrer da melhor forma, foram utilizados os conhecimentos obtidos durante o curso para a construção de artefatos baseados nos conceitos de Engenharia de Software. Estes, foram necessários para que a estrutura do planejamento estivesse a mais detalhada e clara possível. A partir de um planejamento bem estruturado, foi possível prosseguir com o projeto e busca de informações sobre o tema, para que assim, tivéssemos o conhecimento necessário para construir a ferramenta desejada.

Com base na busca de conhecimento sobre o tema e resultados da pesquisa elaborada, foi possível entender com mais detalhes a real necessidade de uma ferramenta como essa no mercado e qual o público-alvo que deveríamos atender. Portanto, a conclusão deste projeto implica na importância de entender a necessidade das informações citadas durante todo o projeto e o quanto pode ser necessário no mercado as funcionalidades das novas implementações propostas para o futuro.

Referências

BORGES, Dayane. Código de endereçamento postal – O que significa e para que serve o CEP? Disponível em: <https://conhecimentocientifico.com/codigo-de-enderecamento-postal-cep/>. Acesso em: 07/04/2022.

CORREIOS. Tudo sobre o CEP. Disponível em [https://www.correios.com.br/enviar/precisa-de-ajuda/tudo-sobre-cep#:~:text=O%20CEP%20\(C%3%B3digo%20de%20Endere%C3%A7ament%20o,%C3%B3rg%C3%A3os%20p%C3%ABlicos%2C%20empresas%20e%20edi%C3%ADcios](https://www.correios.com.br/enviar/precisa-de-ajuda/tudo-sobre-cep#:~:text=O%20CEP%20(C%3%B3digo%20de%20Endere%C3%A7ament%20o,%C3%B3rg%C3%A3os%20p%C3%ABlicos%2C%20empresas%20e%20edi%C3%ADcios). Acesso em: 07/04/2022. Acesso em 07/04/2022.

Docker Overview. Disponível em: <https://docs.docker.com/>. Acesso em 19/07/2022.

Documentation. Table of Contents. Disponível em: <https://www.rabbitmq.com/documentation.html>. Acesso em 19/07/2022.

GO Documentation. Disponível em: <https://go.dev/doc/>. Acesso em 21/07/2022.

NASCIMENTO, Anderson. O que é API? 02/07/2014. Disponível em <https://canaltech.com.br/software/o-que-e-api/>. Acesso em 08/04/2022.

WEBB ,Phillip. Spring Boot Reference Documentation. Disponível em:<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>. Acesso em 21/07/2022.

Anexos

Em anexo, o link para os repositórios contendo todos os artefatos de engenharia de software, documentações e o código fonte do projeto.

<https://github.com/HigorJardini/Locany>