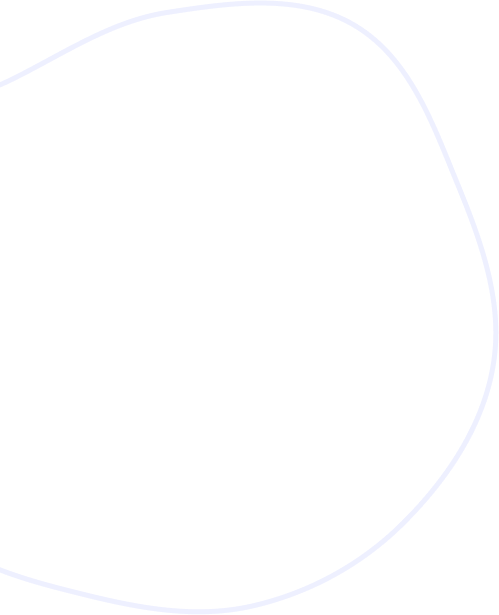
**DOCUMENTAÇÃO FINAL VAPM – VAI ALI PRA MIM**

****

****

**Integrantes:**

Pedro Abbate

Wellington Macena

Kívia Araújo

Martha Ferreira

Rubens Nascimento

Paulo Felipe

**1ª SPRINT**

**1.1 Documento de Contexto de Negócio**

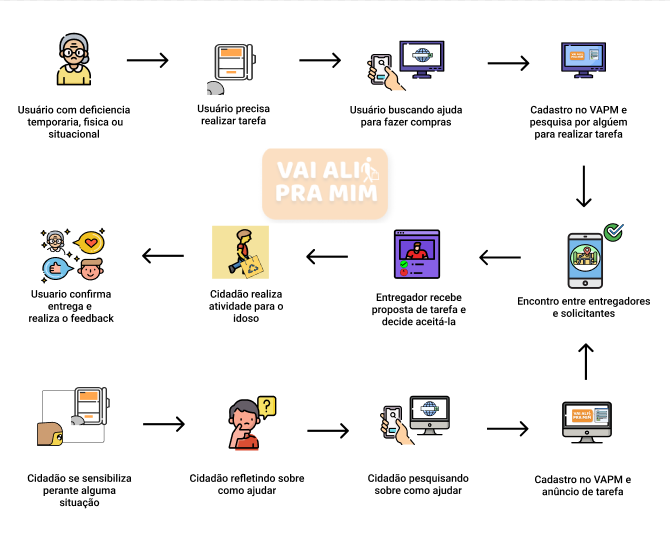
Documento de contextualização e justificativa do projeto foram feitos e adicionados ao Trello, a partir dele fizemos todas as contextualizações das apresentações de sprints

**Contextualização contida no documento:**

O mercado global de API’s vem crescendo cada vez mais. Analistas esperam que mercado global de APIs as a service atinja 965 milhões de dólares até 2020 com taxa de crescimento anual acima de 30%, segundo estudo realizado pela Technavio.

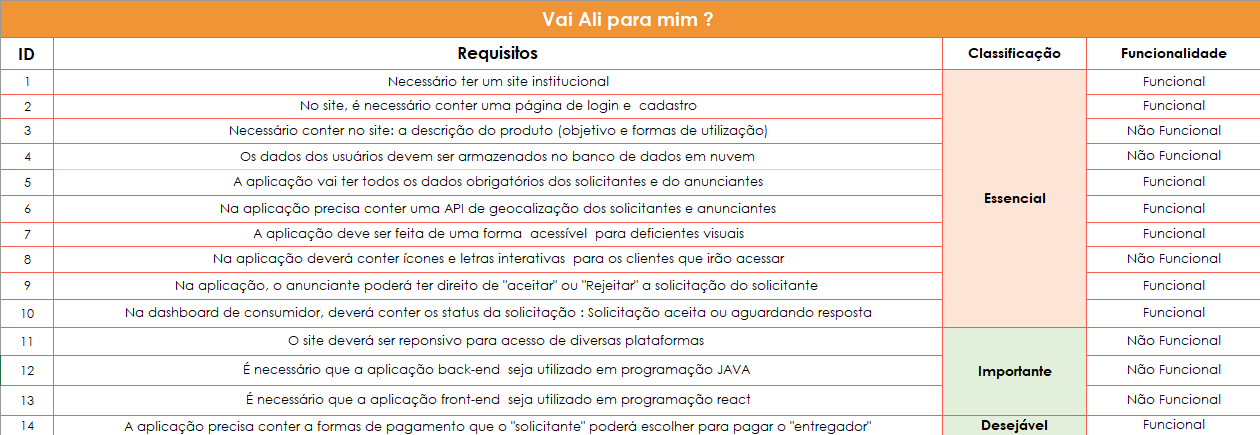
Com a ascensão da transformação digital, muitas empresas estão adentrando o campo de Internet of Things. E com isso, API acabou se tornando um widget de tecnologia avançada que reúne aplicações, dispositivos, dados e cloud, estabelecendo a comunicação entre diversos sensores por meio de múltiplos dispositivos, além de também viabilizar análise de dados em cloud. Lembrando também que, segundo o Gartner, até 2020 serão mais de 21 bilhões de dispositivos conectados à Internet, o que abre muitas oportunidades para o mercado de APIs, além é claro de sua necessidade.

Dado o cenário atual do pais, temos como projeto uma solução API para compras e venda de serviços para moradores locais em grupo de risco, visando diminuir os riscos de contágio pelo COVID-19, aumentar a conexão entre pessoas do mesmo ciclo para que seja possível ajudarem uns aos outros mantendo o distanciamento social.

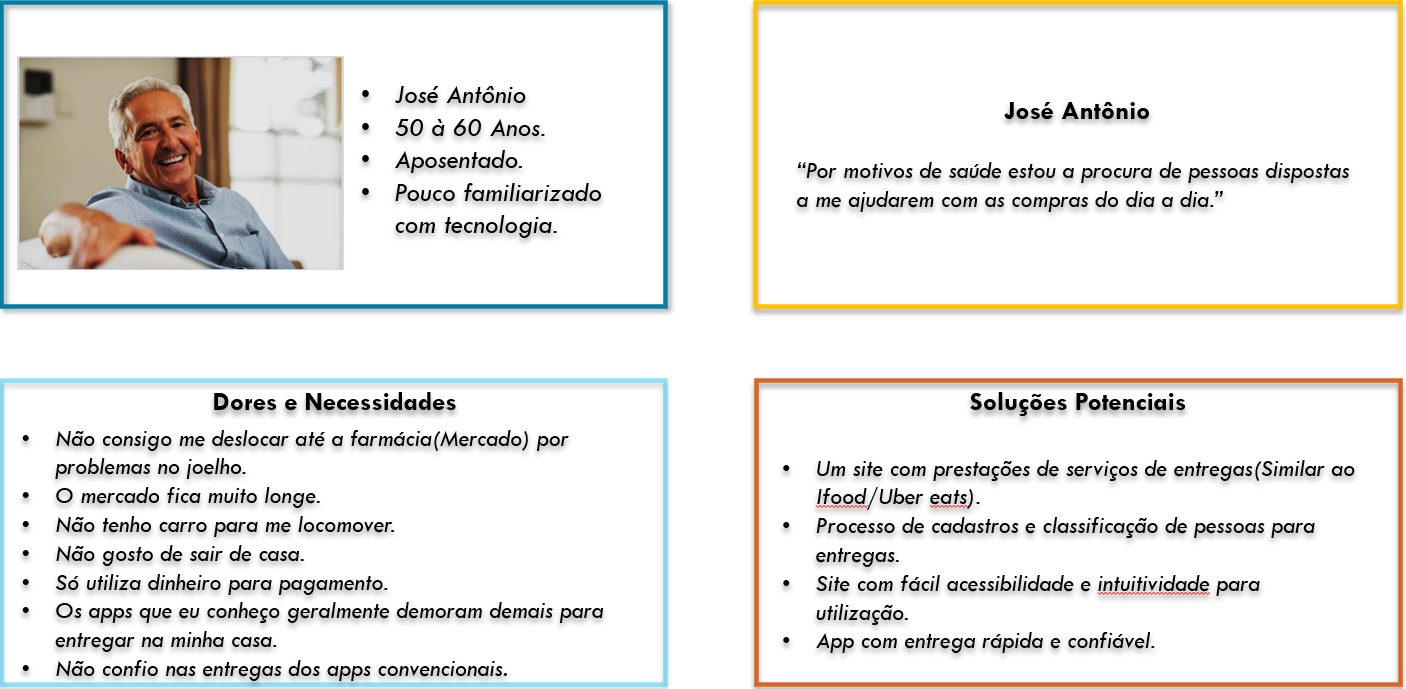
**1.2 Desenho de Solução**

**1.3 Definição das personas e lista de requisitos**

Lista de requisitos VAI ALI PRA MIM



Persona VAI ALI PRA MIM



**1.4 Criação de conta no Azure**

Conta criada com login: [192-3a-grupo2@bandtec.com.br](mailto:192-3a-grupo2@bandtec.com.br)

Com usuário editor: localadmin

**1.5 Projeto criado e configurado GitHub**

Projeto criado e configurado no github com a aprovação dos professores:

<https://github.com/BandTec/vai-ali-pra-mim.git>

**1.6 Equipe criada no planner ou ferramenta aprovada pelo professor**

Inicialmente utilizamos o Trello como ferramenta, porém atualmente estamos utilizamos a ferramenta Bitrix.

Trello: <https://trello.com/b/eLKvCWEY/vai-ali-pra-mim>

Bitrix: <https://vapm.bitrix24.com.br/stream/?current_fieldset=SOCSERV>

**2.1 Estrutura do projeto funcionando (protótipo SpringBoot)**

De primeiro momento fizemos as controllers do projeto, sendo elas as ControllerUsuario, ControllerPedido, ControllerCartao, ControllerPedido e ControllerPost.

**2.2 Serviços de Login e Logoff usando Postman**

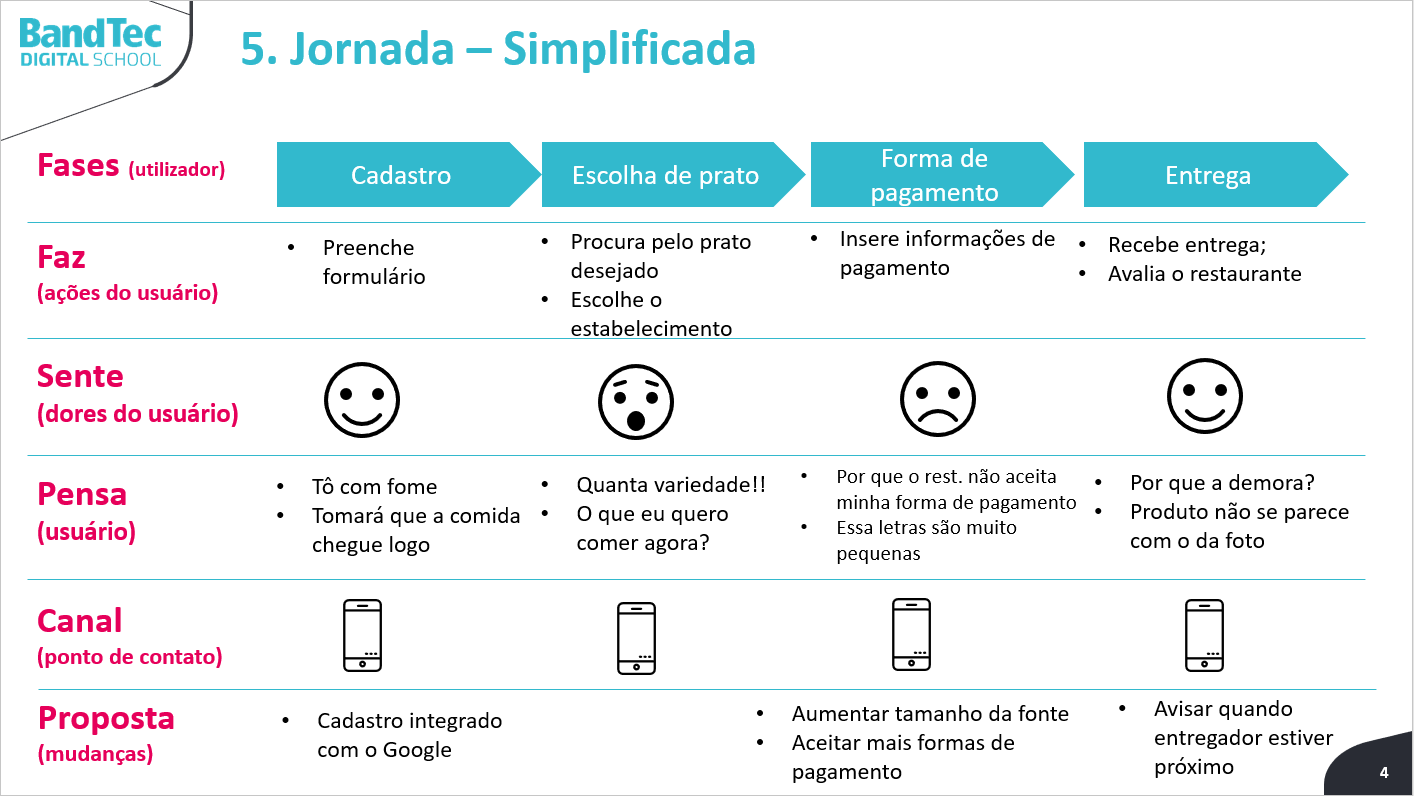
Para esse item fizemos os métodos de verificação para login e logoff, e utilizamos o postaman para fazer os testes, uma vez que o Front-end ainda não estava em produção.

**3.1 Pesquisa de campo - jornada do usuário**

Utilizamos o template disponibilizado pelo professor para fazer a jornada do usuário:



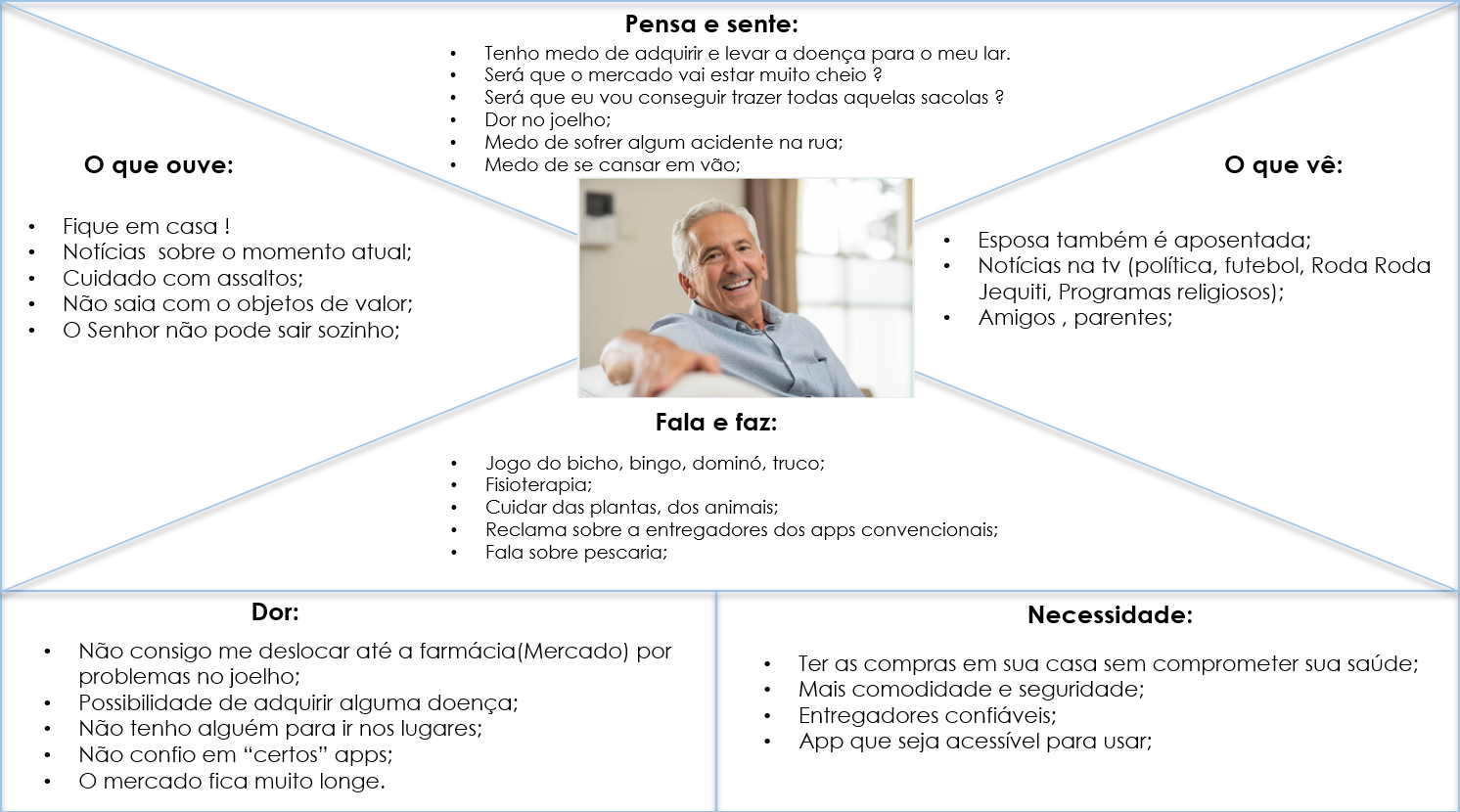




**4.1 Classe abstrata ou Interface do projeto**

Implementamos uma classe abstrata para adicionar informações de cadastro, tanto pro solicitante quanto para o entregador.

**5.1 Mapa de empatia**



**6.1 Protótipo (Wireframe telas do Projeto) – Figma**

Telas feitas no figma:

<https://www.figma.com/file/SB34nXwn8bt4OUaFACIUH3/Telas-vai-ali-pra-mim?node-id=0%3A1>

**2ª SPRINT**

**1.1 Gestão do projeto**

Gestão do projeto foi feita a partir do Trello:

<https://trello.com/b/eLKvCWEY/vai-ali-pra-mim>

**1.2 Github atualizado**

Continuação do projeto no mesmo repositório criado na primeira sprint:

<https://github.com/BandTec/vai-ali-pra-mim.git>

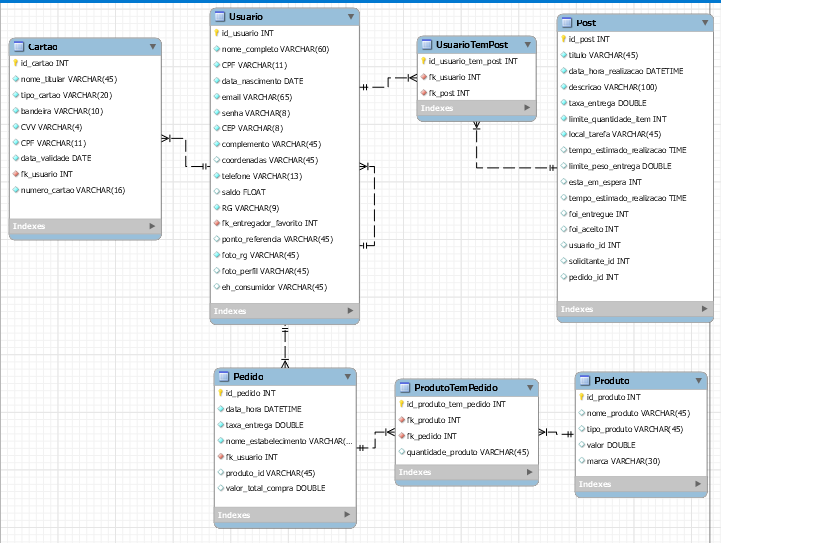
**1.3 Requisitos e tasks atualizadas**

Requisitos e tasks continuaram a serem atualizadas no Trello:

<https://trello.com/b/eLKvCWEY/vai-ali-pra-mim>

**1.4 Modelagem de Dados**

Modelagem feita no MySQL



**2.1 Desenho de arquitetura da aplicaçãoDiagram

Description automatically generated**

**3.1 CRUDS (Sprint boot + ORM)**

Nesse entregável foi implementado o banco de dados do Azure para armazenar e consultar dados, também foi implementado um método de download de dados da classe usuário tanto em .txt como em .csv e foi melhorado as controllers com a implementação de funcionalidades aprendidas no período.

**Implementação de banco, métodos de download, melhora controllers**

**3.2 Login/Logoff com ORM**

Foram implementadas classes que fazem a verificação de cadastro para realizar o login e uma funcionalidade para realizar o logoff

**4.1 Padrão de Projeto (Design Partners)**

Escolhemos para utilizar o padrão de projeto Adapter uma vez que de forma exemplificável por adaptadores de cabos, o padrão Adapter converte a interface de uma classe para outra interface que o cliente espera encontrar, "traduzindo" solicitações do formato requerido pelo usuário para o formato compatível com o a classe *adapter* e as redirecionando. Dessa forma, o Adaptador permite que classes com interfaces incompatíveis trabalhem juntas.

**4.2 Criar documento de layout (entrada e saída) para carregamento de dados**

Documento criado e adicionado ao repositório do git com os dados:

1. **Header**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Número do campo | Nome do campo | Tamanho | Posição | Formato | Descrição |
| 1 | Tipo de registro | 002 | 001-002 | A | Registro header: 01 |
| 2 | Tipo de arquivo | 003 | 003-005 | A | Extensão do arquivo |
| 3 | Tipo de  registo | 005 | 006-007 | A | Registro  header |

**Tamanho de dados úteis: 031**

1. **Corpo**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Número do campo | Nome do campo | Tamanho | Posição | Formato | Descrição |
| 1 | Tipo de registro | 002 | 001-002 | A | Registro de dados: 02 |
| 3 | Nome completo | 050 | 003-062 | A | Nome completo do usuário do sistema |
| 4 | CPF | 011 | 063-074 | A | Nº de CPF do usuário |
| 5 | Data de nascimento | 008 | 075-085 | N | Data de nascimento do usuário |
| 6 | E-mail | 050 | 086-151 | A | E-mail usuário |
| 7 | Telefone | 013 | 152-162 | N | Telefone do usuário |
| 8 | CEP | 008 | 163-174 | N | Nº do CEP do usuário |
| 9 | Complemento | 020 | 175-196 | A | Complemento da residência do usuário |
| 10 | Saldo | 007 | 197-204 |  | Salvo em conta do usuário |
| 11 | RG | 009 | 205-214 |  | RG do usuário |
| 12 | Ponto de referência | 060 | 215-275 |  | Ponto de referência de endereço do usuário |
| 13 | Senha | 015 | 256-271 |  | Senha do usuário |
| 14 | Coordenadas | 030 | 272-302 |  | Coordenadas do usuário |
| 15 | Foto RG | 004 | 303-307 | N | Foto do RG do usuário |
| 16 | Foto Perfil | 004 | 308-312 | N | Foto de perfil do usuário |
| 17 | Consumidor ou não | 002 | 313-315 |  | Se é consumidor ou não |
| 18 | ID Cartão | 002 | 316-318 | N | Nº de cartões cadastrados |

**Tamanho de dados úteis: 207**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | Tipo de registro | 002 | 001-002 | A | Registro trailer: 01 |
| 2 | Quantidade de registros de dados | 006 | 003-008 | N | Número de ocorrências gravadas no arquivo |

1. **Trailer**

**4.2 Funcionalidades de Importação/Exportação (Header, Detail, Trailer) sem usar componentes e usando pilha e fila.**

Funcionalidades foram implementadas a partir do documento citado a cima, assim gerando a possibilidade de download da classe de Usuários, em txt ou csv.

**3ª SPRINT**

**1.1 Dois microserviços**

Como microserviços temos o matching entre solicitantes e entregadores, e também temos a parte de login e cadastro tanto dos entregadores quando dos solicitantes.

**2.1 Planilha de testes UAT**

Nessa planilha detalhamos o funcionamento de todas as nossas páginas do site. Ela está localizada em nosso repositório do github:

<https://github.com/BandTec/vai-ali-pra-mim.git>

**2.2 Projeto WEB aderente aos requisitos de UI + UX**

Para esse tópico implementamos as telas WEB a partir das telas descritas no figma(<https://www.figma.com/file/SB34nXwn8bt4OUaFACIUH3/Telas-vai-ali-pra-mim?node-id=0%3A1>), pois estão seguindo os devidos padrões solicitados.

**2.3 Diagrama de solução de software**

Nosso diagrama de solução de software contem todas as tecnologias e integrações que utilizamos no projeto, com ele é possível visualizar o front-end, back-end e banco de dados da solução, está localizado também em nosso repositório do github:

<https://github.com/BandTec/vai-ali-pra-mim.git>

**3.1 Implementação de pilha e fila no projeto**

No projeto foram implementadas

**3.2 Funcionalidade de importação**

As funcionalidades de importação foram implementadas para aceitar dados enviados para preenchimento dos atributos da classe usuários

Disponível em nosso repositório do github:

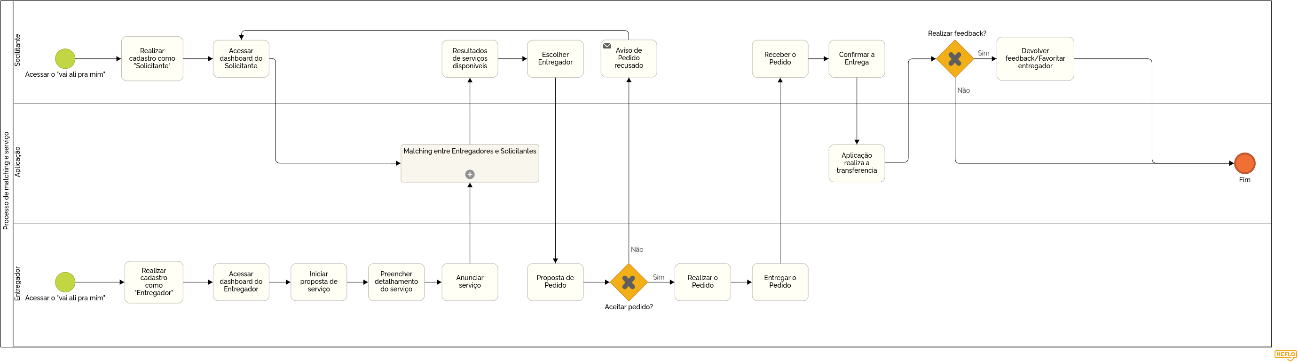
<https://github.com/BandTec/vai-ali-pra-mim.git>

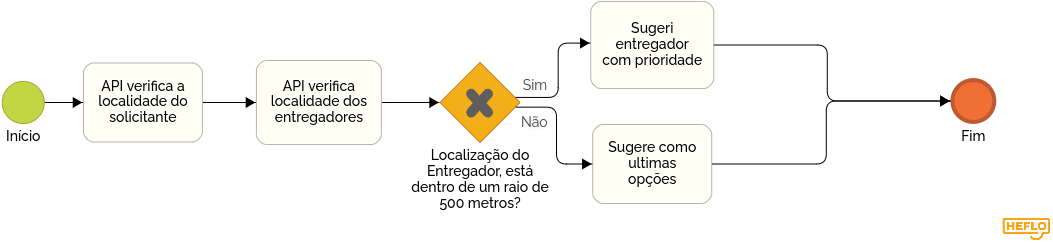
**4.1 Bootamp**

As páginas de Dashboard do consumidor, Dashboard do entregador, login e página de cadastro, além dos pop-ups nas dashboards

**Documentos Adicionais**

**BPMN**

****

****

**Diagrama de Classes**

