

WAD

WEB APPLICATION DOCUMENT

PONTE

Autores: Bruno Wasserstein,

Celine Pereira de Souza,

Henrique Godoy,

Marcos Vinicyus,

Raab lane,

Yago Araújo

Data de criação: 10/10/2022

Controle do Documento

Histórico de revisões

Data	Autor	Versão	Resumo da atividade
11/10/2022	Raab	1.1	Atualização da capa do documento e seção 3.1
11/10/2022	Yago	1.2	Atualização dos pontos 1, 1.1, 1.2,1.3 e inserção do Value Proposition Canvas
17/10/2022	Yago	1.3	Inserção da Matriz de Riscos
17/10/2022	Celine	1.4	Formatação dos textos e conteúdo escrito e Matriz SWOT
19/10/2022	Celine	1.5	Modelos de negócio
19/10/2022	Yago	1.6	Organizei os tópicos e inseri as Forças de Porter
20/10/2022	Celine	1.7	Corrigi as fontes
20/10/2022	Raab	1.8	Preenchimento da seção 1.3.1
20/10/2022	Yago	1.9	Correção dos objetivos gerais
27/10/2022	Celine	2.0	Arquitetura da Solução - seções 4.1 e 4.2
02/11/2022	Yago	2.1	Modelagem Relacional e Lógica- seção 4.4
03/11/2022	Yago	2.2	Inserção do link das documentações no apêndice
12/11/2022	Henrique Godoy	2.3	Correção de erros apontados nos artefatos
15/11/2022	Yago	2.4	Inserção do guia de estilos do sistema
16/11/2022	Raab	2.5	Correção de erros apontados nos artefatos
16/11/2022	Yago	2.6	Inserção dos prints dos endpoints no apêndice I

Sumário

Visão Geral do Projeto

Parceiro de Negócios

O Problema

Objetivos

Objetivos gerais

Objetivos específicos

Descriitivo da Solução

Partes Interessadas

Análise do Problema

Análise da Indústria

Análise do cenário: Matriz SWOT

Proposta de Valor: Value Proposition Canvas

Matriz de Risco

Requisitos do Sistema

Persona

Histórias dos usuários (user stories)

Arquitetura do Sistema

Módulos do Sistema e Visão Geral (Big Picture)

Tecnologias Utilizadas

UX e UI Design

Wireframe

Design de Interface - Guia de Estilos

Projeto de Banco de Dados

Modelo Conceitual

Modelo Lógico

Testes de Software

Teste de Usabilidade

Referências

Apêndice

1. Visão Geral do Projeto

1.1 Parceiro de Negócios

A empresa parceira é a MRV, a maior construtora da América Latina. A companhia atua no setor imobiliário e teve sua criação em 1979. O nome MRV advém dos sócios: Rubens Menin Teixeira de Souza, Mário Lúcio Menin e Vega Engenharia Ltda. Como objetivo principal, a empresa possui foco na construção civil e já atua em 160 cidades do Brasil. Dentro dessa lógica, a MRV construiu mais de 4000 mil casas e apartamentos lançados em 22 estados brasileiros e no Distrito Federal, o que gera mais de 6 mil postos de trabalho por ano em seus locais de construção. Além disso, sua principal missão é possibilitar o sonho da casa própria para a classe média brasileira.

1.2 O Problema

Conforme o parceiro, a empresa está com dificuldades de encontrar empreiteiros para suas obras. Conforme informações disponibilizadas, a contratação dessa classe de candidatos é lenta e gera muitos gastos, já que não há um padrão seletivo de fato com tal grupo. Isso acontece porque o processo de contratação dessas pessoas é muito estratificado e pouco direto, gerando muitos custos para a MRV. Com base nisso, o grupo utilizará de um parâmetro de seleção no formato web para escalar o negócio, ou seja, aumentar o valor gerado e reduzir as despesas.

1.3 Objetivos

1.3.1 Objetivos gerais

Otimizar o contato entre as empreiteiras do setor de construção civil com a empresa MRV, por meio de um site focado na contratação dessas pequenas/médias empresas.

1.3.2 Objetivos específicos

- Facilitar o procedimento de contratação;
- Ajudar no processo de escalabilidade da empresa;
- Aumentar a qualidade dos funcionários na MRV;
- Criar uma plataforma de contratação com linguagem simples;
- Desenvolver uma aplicação web eficiente;

1.4 Descritivo da Solução

A solução é construir uma aplicação web acessível para os dois públicos alvos analisados pela equipe: empreiteiros e time de recrutamento da MRV. Nesse projeto, o foco será produzir uma plataforma que seja o mais fácil possível de entender e realizar os objetivos dentro da aplicação. Nesse contexto, as principais metas traçadas na jornada do usuário são: conseguir se cadastrar na plataforma, navegar por nela assim que for cadastrado, conseguir se candidatar aos empregos que apareceram e possibilitar o envio de arquivos para efetivação do possível candidato. Do lado do administrador, ele poderá gerenciar as vagas que poderão ser disponibilizadas, receber arquivos para análise com foco em contratação e metrificar os candidatos. Por fim, o projeto possuirá também botões e interfaces que levarão os dois tipos de usuários a cumprir essas metas dentro do seu espaço de forma rápida e simples.

1.5 Partes interessadas

Primariamente, há dois segmentos interessados: a equipe de contratação da MRV e os empreiteiros desacostumados com as interfaces web ou mobile. Nesse sentido, o primeiro grupo está focado em diminuir os gastos com captura de possíveis colaboradores e o segundo se refere a um público de pequenas empresas que não conseguem fechar contratos devido à sua falta de conhecimento acerca do processo de fechamento nas plataformas modernas. Posteriormente, o objetivo é abrir a plataforma para o restante do mercado, conectando outras construtoras aos empreiteiros, realizar parcerias com lojas, e realizar a venda de cursos. Isso acontecerá com o intuito de amenizar a dor citada acima: falta de conexão entre empreiteiros e grandes construtoras.

2.0 Análise do Problema

2.1 Análise da Indústria

2.1.1 5 Forças de Porter em Relação ao Cenário da Empresa MRV

Rivalidade Entre os Concorrentes: Há empresas rivais do mesmo porte, devido ao seu alto impacto em todo Brasil. Embora a empresa seja focada em apartamentos para pessoas de baixa renda, suas obras são efetivadas em todos os estados do país, o que a coloca em pé de igualdade com outros players do mercado imobiliário.

Poder de Barganha de Clientes: Possui uma probabilidade praticamente nula de acontecer, já que para barganhar a compra do produto oferecido (neste caso, um imóvel) é preciso possuir um aporte alto, destoando da capacidade monetária do público alvo da MRV.

Poder de Barganha de Fornecedores: Muito possivelmente acontecerá, já que o mercado está em alta demanda no âmbito de infraestrutura e tende a crescer mais alta ainda conforme a Urbanização se fortalece.

Ameaças de Produtos Substitutos: Há muita chance de acontecer. O produto que mais compete seria o da Housing Pact - empresa com foco em junção de startups para solucionar o problema de habitação para pessoas de baixa renda.

Ameaças de Novos Entrantes: Deverá acontecer, por meio de empresas multinacionais com poder aquisitivo maior e focados no investimento de longo prazo na infraestrutura do futuro no país.

2.1.2 Principais competidores

O público-alvo da MRV varia entre classe baixa, classe média (possivelmente majoritária) e chega até as camadas mais baixas do que pode ser considerado classe alta, conforme o critério de classe alta sendo a renda familiar acima de 5 salários mínimos. Por ter seu foco em apartamentos, consegue ter um mercado muito maior do que na construção de casas, além disso, constrói por 22 estados do Brasil, mostrando que compete por quase todo o país. Dessa forma seus principais competidores são: Direcional Engenharia, Cyrela e Tenda.

2.1.3 Modelo de negócios

A MRV é uma das empresas mais tradicionais do ramo imobiliário no Brasil. Está no mercado há 40 anos, já construiu imóveis para mais de 1,5 milhão de brasileiro e colaborou para o crescimento do programa Minha Casa, Minha Vida no país (MCMV). Nos últimos anos, a construtora buscou novos modelos de negócio, visando reduzir a exposição ao programa MCMV e aumentar seu

leque de oportunidades. Após a aquisição da startup Luggo, a MRV passou a atuar no mercado de locação de imóveis. Além disso, a Luggo é uma plataforma para testar modelos de negócio no mercado imobiliário. Por meio dela, a construtora já testou lojas de conveniência dentro de condomínios, onde o acesso é permitido através de um QR Code, e até um modelo de aluguel de carro compartilhado. Dado o exposto, a MRV demonstra interesse em novos modelos de negócio e pretende minimizar as ameaças externas como, por exemplo, uma crise econômica no país.

2.1.4 Tendências de mercado

A MRV é uma incorporadora e construtora, ou seja, além de construir imóveis, a MRV também cuida do processo de comercialização de um imóvel. Pensando na parte de construção civil, as empresas do ramo pretendem investir em:

- Adaptação de recursos: A tendência é pensar cada vez mais na proteção do meio ambiente, para isso estão sendo desenvolvidas novas tecnologias para serem incorporadas nas construções como, por exemplo, tijolos ecológicos;
- Tecnologia: O investimento em tecnologia está no centro das futuras soluções para este mercado, pois facilitam a comunicação entre as partes envolvidas e agregam mais qualidade no resultado da obra. Alguns tipos de tecnologias já são realidade no setor como, por exemplo, drones de construção que facilitam o mapeamento de grandes áreas e longas distâncias de forma rápida.
- Robotização do processo: A robotização torna o processo mais rápido e diminui a erros.

Sobre a parte de comercialização de imóveis, a tendência é colocar o cliente no centro das decisões, pois os novos clientes são mais jovens, ou seja, esperam uma experiência de compra mais rápida, fluida e digital. Além de pensar na forma como vão vender os imóveis, o futuro também é pensar em como essa nova geração espera de um imóvel. Dado o exposto, o futuro do setor imobiliário e de construção civil está focado na sustentabilidade e tecnologia.

2.2 Análise do cenário: Matriz SWOT

Para compreender as forças, fraquezas, oportunidades e ameaças envolvendo a MRV, realizamos uma análise utilizando uma Matriz SWOT. Indicamos a seguir (Figura 01)

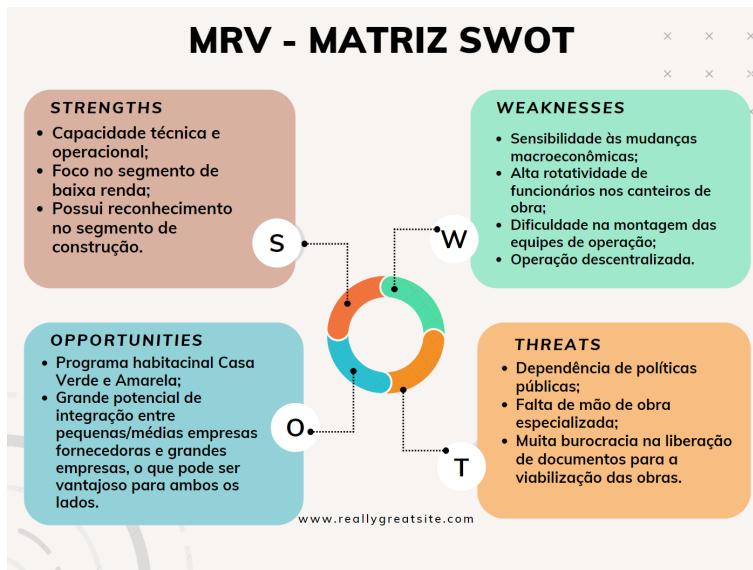


Figura 01 : Elaboração dos autores

2.3 Proposta de Valor: Value Proposition Canvas

Uma estrutura que ajuda a garantir que o nosso site esteja de acordo com o que o cliente valoriza e precisa, é descrita a seguir (Figura 2)



Figura 02 : Elaboração dos autores

2.4 Matriz de Risco

A seguir (Figura 3), uma análise da Matriz de Risco que auxilia na visualização da probabilidade de um determinado cenário ocorrer em nossa parceria com a MRV e o impacto do mesmo.

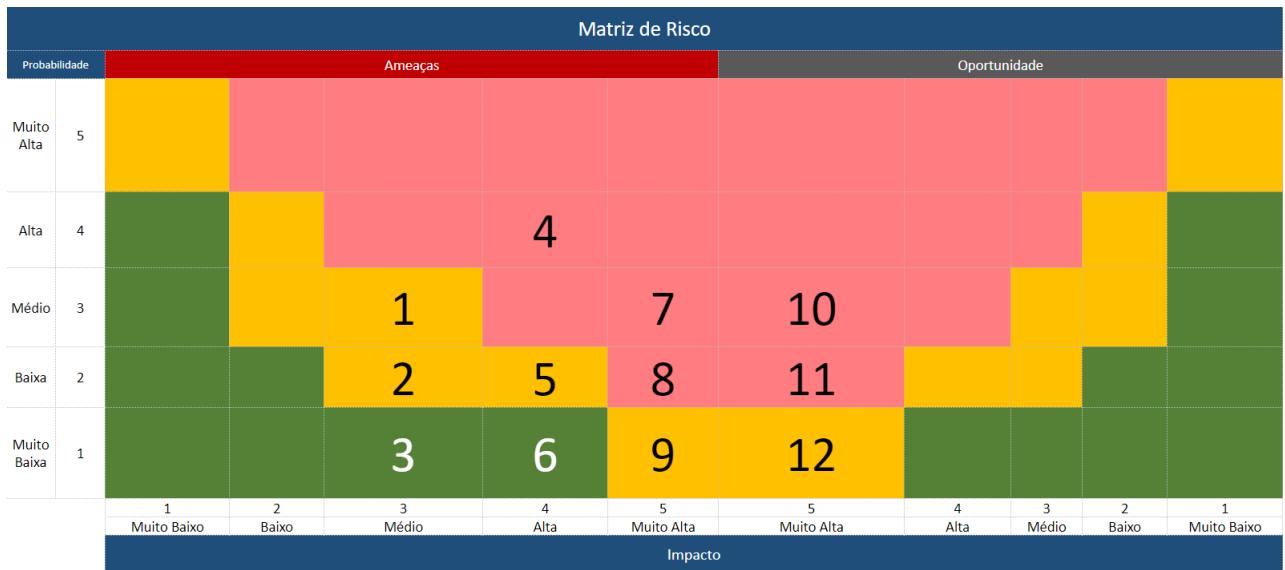


Figura 03 : Elaboração dos autores

Legenda

- 1 - Atrito entre os membros do grupo
- 2 - Não compreender o que foi pedido
- 3 - Membros do grupo não entregarem o trabalho
- 4 - Discordância de ideias
- 5 - MRV não fornecer os dados necessários
- 6 - Membro sair do grupo
- 7 - Entregar menos do que foi planejado
- 8 - Erros no código não serem corrigidos
- 9 - Falta de comunicação causaram atrasos nas entregas
- 10 - Grupo entregar um MVP impressionante
- 11 - Entregar funcionalidades além das esperadas
- 12 - MRV querer dar continuidade ao site

Obs: Cada número com fonte de tamanho maior representa respectivamente uma observação relatada na matriz de risco. Foi usada a legenda para ser possível ler a matriz de risco com mais facilidade.

3.0 Requisitos do Sistema

3.1 Personas

Personagens fictícios criados para representar os usuários do nosso site.



Manoel, 55 anos, de Belo Horizonte, Minas Gerais. Casado, e possui apenas um filho e uma filha, é extrovertido e gosta de conversar com os amigos no whatsapp, usando como uma opção de entretenimento e trabalho, se considera organizado e prática. Possui uma empresa que presta serviços de gesso há 20 anos, de modo que começou trabalhando sozinho, e ao decorrer do tempo agregou funcionários para sua empresa.



Mônica, 36 anos, de São Paulo, São Paulo. Solteira, gosta de ler livros de romance e jogar sudoku. Possui deficiência auditiva. Ama o meio-ambiente e se preocupa com sua preservação. Trabalha na área de gerenciamento da obra na MRV, de modo que sua responsabilidade é de procurar e gerenciar as empreiteiras que prestam ou prestaram serviços para MRV nas obras.

3.2 Histórias dos usuários (user stories)

Eu (Manoel), enquanto dono de uma empresa que realiza a parte de gesso em obras de construção civil, quero ter acesso a um site de fácil usabilidade para que eu possa cadastrar minha empresa e logo conseguir maior constância na contratação de obras para MRV.

Eu (Mônica), enquanto membro da equipe de gerenciamento de obras na MRV, quero ter acesso ao site e as empreiteiras nele cadastradas, para ter mais otimização de tempo e qualidade em achar empresas prestadoras de serviço que farão um serviço terceirizado em nossas obras e conseguir ter um portfólio maior de possíveis parceiras (empreiteiras).

4.0 Arquitetura do Sistema

4.1 Módulos do Sistema e Visão Geral (Big Picture)

O diagrama a seguir (Figura 4) apresenta a Arquitetura da Solução, conjunto de diagramas que documentam vários níveis da arquitetura.

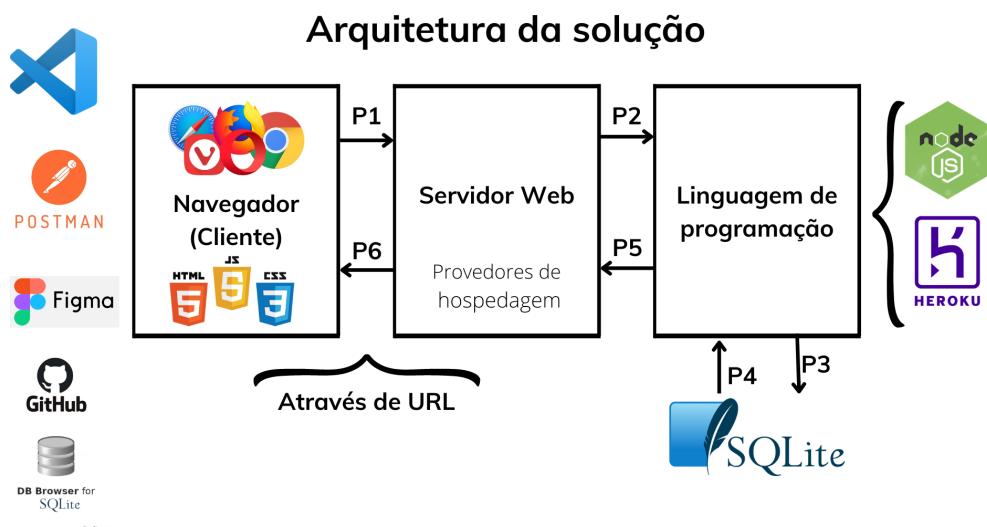


Figura 04: Elaboração dos autores

Passo 1 - Requisição HTTP;

Passo 2 - O servidor acessa o banco de dados através de uma linguagem de programação;

Passo 3 - Consulta o banco de dados;

Passo 4 - Banco de dados envia a resposta;

Passo 5 - Envio dos dados para o servidor;

Passo 6 - Resposta HTTP.

- **Exemplo de funcionamento da Web**

A seguir (Figura 5), diagrama que apresenta como é o funcionamento da arquitetura da solução, na prática, como a Web funciona.

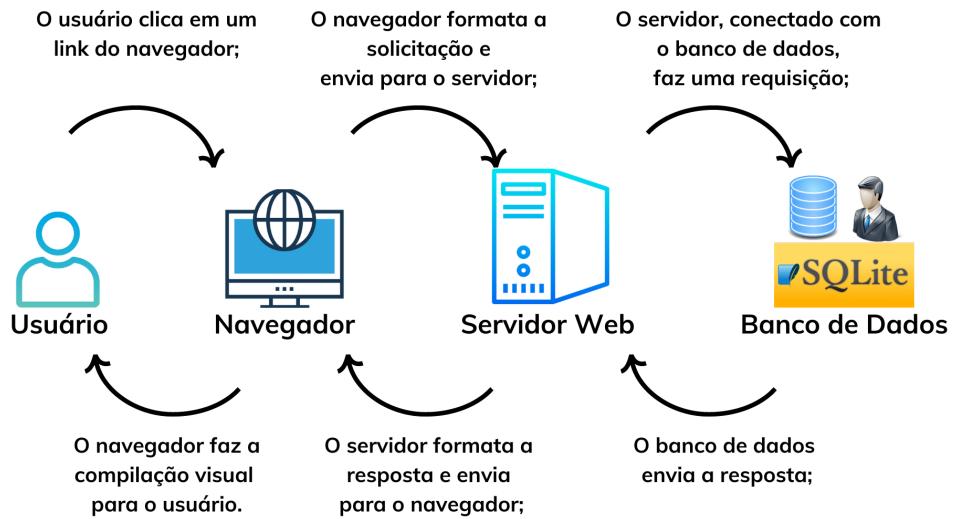


Figura 05: Elaboração dos autores

4.2 Tecnologias Utilizadas

Tecnologia	Função	Versão
Visual Studio Code	O Visual Studio Code é um editor de código-fonte. Dentro do projeto, utilizamos para depuração, controle de versionamento, Git incorporado, realce de sintaxe, complementação inteligente de código e refatoração de código	1.72
Postman	Postman é uma plataforma de API para desenvolvedores. Usamos para projetar, construir, testar e iterar suas APIs	9.31.0
Figma	Figma é um editor gráfico de vetor e prototipagem de projetos de design baseado principalmente no navegador web	88.1.0
Github	GitHub é uma plataforma que nos permite hospedar	3.7.0

	código-fonte e arquivos com controle de versão usando o Git	
Node.js	Node.js é um software de código aberto, multiplataforma, baseado no interpretador V8 do Google e que permite a execução de códigos JavaScript fora de um navegador web	16.2.0
Heroku	Heroku é uma plataforma de nuvem como serviço que suporta várias linguagens de programação. Permite-nos hospedar código e não se preocupar muito com a disponibilidade, escala e infraestrutura da aplicação	-
db Browser	Database Browser é um aplicativo que nos permite visualizar, editar e executar scripts SQL em banco de dados Oracle, ODBC e SQL Server. Com ele é possível navegar ou alterar dados, executar scripts, exportar e imprimir dados e muito mais	3.12.2
SQLite	SQLite é uma biblioteca em linguagem C que implementa um banco de dados SQL embutido. Utilizamos para criar banco de dados	3.39.4
HTML	Utilizamos na construção de páginas na Web	5.3
CSS	Utilizamos para adicionar estilo a um documento web	3
JavaScript	JavaScript é a linguagem de programação que nos permite implementar itens complexos em páginas web	ECMAScript 2022

4.3 UX e UI Design

Projeto das telas do sistema.

4.3.1 Wireframe

O Wireframe desenvolvido encontra-se neste link: [WIREFRAME - PONTE](#)

4.3.2 Design de Interface - Guia de Estilos

O guia de estilos se trata de um documento, o qual guarda todos os dados referentes à estilização do sistema criado. Esse arquivo, desenvolvido pela a equipe CONSTRUDEVS, encontra-se neste link:

[W Guia de Estilos - CONSTRUDEVS .docx](#)

4.4 Projeto de Banco de Dados

Na imagem abaixo, está esquematizado como funcionará o Banco de Dados do projeto PONTE.

4.4.1 Modelo Conceitual

A seguir (Figura 6), o Modelo Conceitual trata da descrição do banco de dados em um alto nível de abstração.

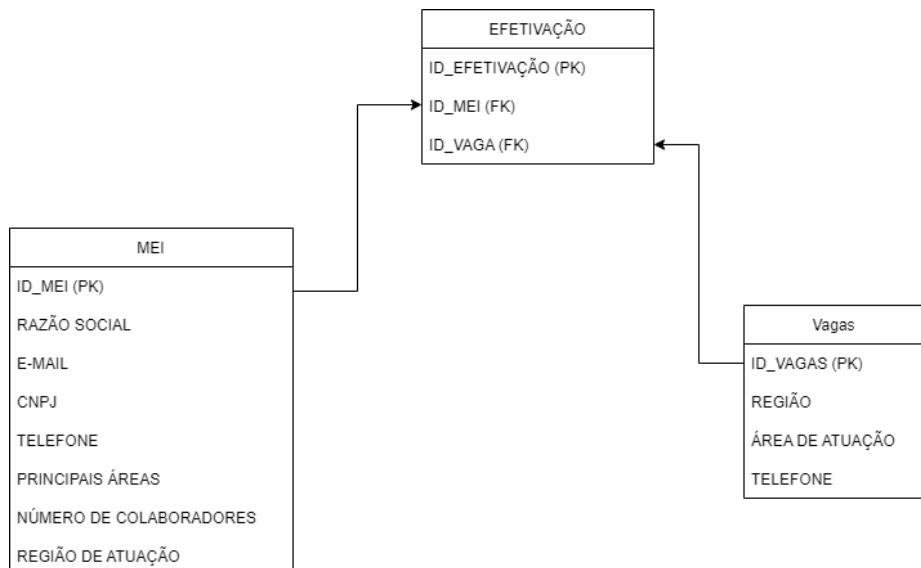


Figura 06: Elaboração dos autores

4.4.2 Modelo Lógico

Nesta sessão, cada atributo será especificado com sua respectiva funcionalidade e tipo de valor com o qual trabalha.

TABELA MEI :

- 1 - **ID_MEI (PK)**: Primary Key da tabela, refere-se à chave primária, ou seja, serve como identificador, trabalha com valores numéricos únicos e auto incrementáveis;
- 2 - **RAZÃO SOCIAL**: Termo registrado sob o qual o MEI se individualiza exerce suas atividades, trabalha com valores do tipo texto;
- 3 - **E-MAIL**: Conjunto de caracteres responsável pela identificação dentro da empresa, funciona com dados do tipo texto;
- 4 - **CNPJ**: Sequência numérica para identificação da empresa. trabalha somente com valores numéricos;
- 5 - **TELEFONE**: Conjunto de números para contato, funciona unicamente com valores numéricos;
- 6 - **PRINCIPAIS ÁREAS**: Agrupamento das principais áreas nas quais a empresa trabalha, esse campo só trabalha com dados do tipo texto;
- 7 - **NÚMERO DE COLABORADORES**: Número que se refere à quantidade de funcionários naquela empresa, esse atributo só funciona com valores numéricos;
- 8 - **REGIÃO DE ATUAÇÃO**: Local onde a empresa trabalha, esse campo recebe valores somente do tipo texto;

TABELA VAGAS:

- 1 - **ID_VAGAS (PK)**: Primary Key da tabela, ou seja, é uma chave primária que serve como identificador dela. Nesse caso, ela recebe valores numéricos auto incrementáveis;
- 2 - **REGIÃO**: Refere-se ao local daquela vaga, esse atributo funciona com dados do tipo texto;
- 3 - **ÁREA DE ATUAÇÃO**: Principais áreas requeridas pela MRV para o trabalho, esse campo trabalha com valores do tipo texto;
- 4 - **TELEFONE**: Sequência numérica que estabelece contato com a MRV, tal atributo trabalha com conjunto de dados do tipo número;

TABELA EFETIVAÇÃO:

1 - ID_EFETIVAÇÃO (PK): Chave principal da tabela, funciona como identificador desse campo. Tal atributo trabalha com dados do tipo numérico e é auto incrementável;

2 - ID_VAGAS (FK): Chave estrangeira (Foreign Key) da tabela Vagas, possui a mesma funcionalidade e aceita o mesmo conjunto de dados;

3 - ID_MEI (FK): Chave estrangeira (Foreign Key) da tabela MEI, possui a mesma funcionalidade e aceita o mesmo conjunto de dados;

5.0 Testes de Software

5.1 Teste de Usabilidade

WIREFRAME (SPRINT 2): [figma com o wireframe](#)

5.2 Guia de Estilos (Style Guide)

Nessa sessão, é fornecido o link para o documento que se refere ao guia de estilos do sistema. Nele, há todas as determinações dos seguintes elementos: cores, tipografia e ícones usados no site.

Documento: [Guia de Estilos - PONTE](#)

Referências

- 6 Construtoras que têm imóveis do Minha Casa Minha Vida. Konkero, 2015. Disponível em: <<https://konkero.com.br/posts/5-construtoras-que-tem-imoveis-do-minha-casa-minha-vida>>. Acesso em: 17 de out. de 2022
- As 100 maiores construtoras do Brasil em 2022. Dica de obra, 2022. Disponível em: <<https://dicadeobra.com/as-100-maiores-construtoras-do-brasil-em-2022/>>. Acesso em: 17 de out. de 2022
- Maiores Construtoras do Brasil: Conheça as Principais!. Total construção, 2020. Disponível em: <<https://www.totalconstrucao.com.br/maiores-construtoras-do-brasil/>>. Acesso em: 17 de out. de 2022
- RANKING INTEC DAS 100 MAIORES CONSTRUTORAS DO BRASIL - 2022. 100 maiores construtoras, 2022 . Disponível em: <<https://100maioresconstrutoras.com.br/construtoras-2022/>>. Acesso em: 17 de out. de 2022
- RAMOS, Martha. 50 das classificadas no Ranking INTEC das 100 maiores construtoras do Brasil 2022 são clientes Sienge ou CV. Sienge, 2022. Disponível em: <<https://www.sienge.com.br/blog/100-maiores-construtoras-do-brasil/>>. Acesso em: 17 de out. de 2022
- As maiores construtoras do Brasil segundo a INTEC. Noventa, 2022. Disponível em: <<https://noventa.com.br/new/as-10-maiores-construtoras-do-brasil-confira-o-ranking-completo/>>. Acesso em: 17 de out. de 2022
- MONTEIRO, João. MRV reproduz modelo de big techs para entregar melhor experiência a clientes. iP News, 2020. <<https://ipnews.com.br/mrv-reproduz-modelo-de-big-techs-para-entregar-melhor-experiencia-a-clientes/>> . Acesso em: 19 de out. de 2022
- MRVE3 – O que analisar antes de investir na incorporadora MRV?. URBE.LAB, 2019. <<https://urbe.me/lab/mrv-vale-a-pena-investir/>>. Acesso em: 19 de out. de 2022
- MRV. MRV&CO, 2022. Página inicial. Disponível em <<https://ri.mrv.com.br/>>. Acesso em: 19 de out. de 2022
- MRV, 2022. Página inicial. Disponível em <<https://www.mrv.com.br/>>. Acesso em: 19 de out. de 2022
- Ellison, N. B., Lampe, C., & Steinfield, C. (2009). Feature social network sites and society: current trends and future possibilities. interactions, 16(1), 6-9.

Apêndice I

Pontes API

Com o objetivo de fazer possível as operações do CRUD na aplicação web criando a oportunidade de utilização de criação, consulta, atualização e eliminação de dados por meio do código, permitindo a manipulação destes dados junto a automatização no objetivo de trazer funcionalidades ao site, desde a parte da construtora quanto na parte da empreiteira. Com o Javascript se faz possível a conexão do front-end com o back-end, e com o NodeJs se faz possível a conexão do Javascript com o Banco de Dados. No momento inicial permitindo a criação de conta criando os dados no banco de dados, na aplicação fazendo a leitura das variáveis citadas, e ao decorrer do tempo permitindo a atualização dos dados, e se necessário deletando os dados, de modo que a aplicação possa respeitar a LGPD.

Funções dentro da API

Utilizando de tabelas contendo dados que podem ser necessários para funções dentro da aplicação, foram feitas funções para a manipulação, tais como leitura, incorporação, atualização e eliminação dos dados dentro das tabelas. Na leitura como na função selectVaga onde recebe-se como resposta da função o id_vaga, região, área de atuação e telefone da empreiteira candidata, na incorporação com a função createVagas onde se cria uma vaga pela construtora, incorporando os dados de região, área de atuação, telefone incorporando isto no banco de dados, na atualização com o updateEfetivacao onde recebe-se o id_efetivacao, id_vaga, e o id_mei atualizando o status de efetivação da vaga postada pela construtora, e eliminação na utilização da função deleteEfetivacao onde recebe-se apenas o id_efetivacao permitindo a eliminação da vaga pela construtora.

CONCLUSÃO

Com o uso de API, se torna possível a realização de várias tarefas de forma automatizada, permitindo a conexão entre o banco de dados com a aplicação

Funções select:

selectVagas

selectMei

selectAllMei

selectEfetivacao

selectAllEfetivacao

Funções insert:

insertVaga

insertMei

insertEfetivacao

Funções delete:

deleteVaga

deleteMei

deleteEfetivacao

Funções update:

updateMei

updateEfetivacao

- Link do Postman: [postman-construdevs](#)]

- Nessa parte, estão os testes de cada requisição do banco de dados.

● MÉTODOS GET

Postman screenshot showing a GET request to `http://localhost:3000/allMeiVagas`. The Body tab contains the following JSON:

```

1
2   "id_mei": 1,
3   "id_vaga": 1

```

The response body shows a single object with the following properties:

```

28
29   {
30     "id_mei": 3,
31     "razaoSocial": "Yago Pães",
32     "cnpj": 1619386108,
33     "telefone": 10693267,
34     "email": "suporte@yago.com",
35     "regiao_empresa": "Internacional",
36     "principaisAreas": "Pães e Alegria",
37     "numeroColaboradores": 5000,
38     "id_vagas": 3,
39     "regiao": "Sudeste",
        "area_de_atuacao": "engataxe"

```

Postman screenshot showing a GET request to `http://localhost:3000/vagas`. The Body tab shows the following JSON response:

```

1
2   [
3     {
4       "id_vagas": 1,
5       "regiao": "Nordeste",
6       "area_de_atuacao": "ajudante",
7       "telefone": 16246245
8     },
9     {
10       "id_vagas": 2,
11       "regiao": "Sudeste",
12     }
]

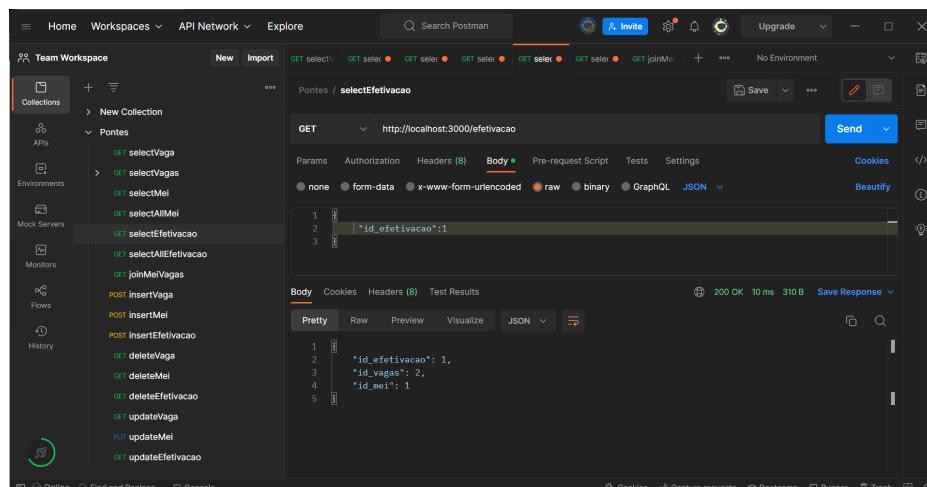
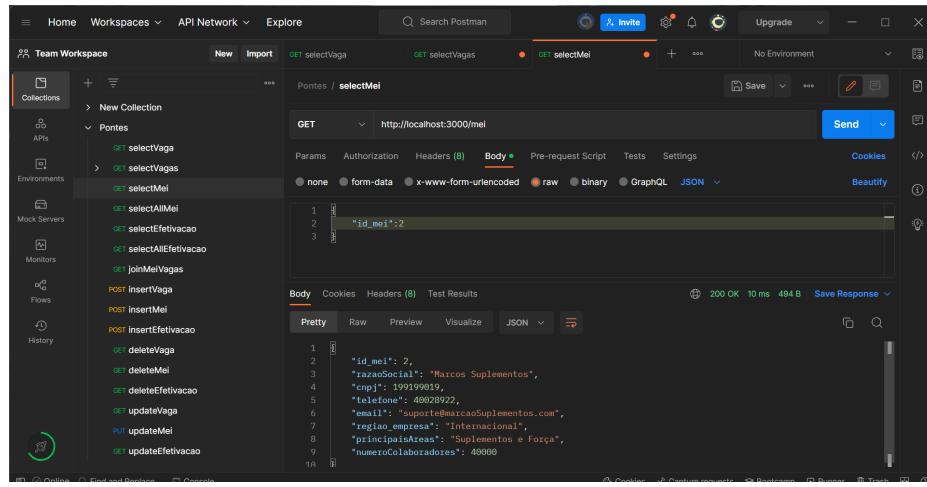
```

The screenshot shows the Postman interface with a collection named "Pontes". A GET request is defined for the endpoint `http://localhost:3000/vaga`. The "Body" tab is selected, showing a JSON object with a single key-value pair: `"id_vaga": 1`. The response status is 200 OK with a response size of 350 B.

```
GET http://localhost:3000/vaga
{
  "id_vaga": 1
}
```

The screenshot shows the Postman interface with a collection named "Pontes". A GET request is defined for the endpoint `http://localhost:3000/allmei`. The "Body" tab is selected, showing a JSON object with multiple fields: `"id_mei": 4, "ruaSocial": "GOD PLAN", "cep": 1619361008, "telefone": 126309620, "email": "super@GOD.com", "regiao_empresa": "GLOBAL", "principaisAreas": "Pies e Alegria", "numeroColaboradores": 5000}`. The response status is 200 OK with a response size of 1.08 KB.

```
GET http://localhost:3000/allmei
{
  "id_mei": 4,
  "ruaSocial": "GOD PLAN",
  "cep": 1619361008,
  "telefone": 126309620,
  "email": "super@GOD.com",
  "regiao_empresa": "GLOBAL",
  "principaisAreas": "Pies e Alegria",
  "numeroColaboradores": 5000
}
```



Postman Screenshot showing a GET request to `http://localhost:3000/efetivacao`. The response body is a JSON array:

```

[{"id_efetivacao": 1, "id_vaga": 3, "id_mei": 2}, {"id_efetivacao": 2, "id_vaga": 1, "id_mei": 3}, {"id_efetivacao": 3, "id_vaga": 5, "id_mei": null}
]
  
```

● MÉTODOS INSERT

Postman Screenshot showing a POST request to `http://localhost:3000/vaga`. The response body is a JSON object:

```

{
  "statusCode": 200
}
  
```

Postman screenshot showing a POST request to `http://localhost:3000/newmei`. The request body is a JSON object:

```

1 "razaoSocial": "Thunders Tech",
2 "cnpj": "1637624902",
3 "telefone": "865329359,
4 "email": "superuser@thundersTech.com",
5 "regiao_empresa": "Norte de Constantinopla",
6 "principaisAreas": "Cloud e DevOps",
7 "numeroColaboradores": 40000
8
9

```

The response status is 200 OK with a `statusCode: 200`.

Postman screenshot showing a POST request to `http://localhost:3000/newefetivacao`. The request body is a JSON object:

```

1 ...
2 ... "id_efetivacao": 5,
3 ... "id_vagas": 6,
4 ... "id_mei": 7
5 ...

```

The response status is 200 OK with a `statusCode: 200`.

● MÉTODOS DELETE

Postman screenshot showing a DELETE request to `http://localhost:3000/vaga`. The request body is a JSON object:

```

1 ...
2 ... "id_vagas": 7
3 ...

```

The response status is 200 OK with a `statusCode: 200`.

Postman / deleteMei

DELETE <http://localhost:3000/mei>

Params Authorization Headers (8) Body **Body** Pre-request Script Tests Settings Cookies Beautify

```

1
2
3
  "id_mei":5

```

Body Cookies Headers (8) Test Results 200 Ok 10 ms 285 B Save Response

Pretty Raw Preview Visualize JSON

1
2
3
 "statusCode": 200

Postman / deleteEletivacao

DELETE <http://localhost:3000/eletivacao>

Params Authorization Headers (8) Body **Body** Pre-request Script Tests Settings Cookies Beautify

```

1
2
3
  "id_eletivacao":5

```

Body Cookies Headers (8) Test Results 200 Ok 15 ms 285 B Save Response

Pretty Raw Preview Visualize JSON

1
2
3
 "statusCode": 200

● MÉTODOS UPDATE

Postman / updateVaga

PUT <http://localhost:3000/vaga>

Params Authorization Headers (8) Body **Body** Pre-request Script Tests Settings Cookies Beautify

```

1
2
3
4
5
6
7
  "id_vaga":6,
  "regiao": "Sul",
  "area_de_atuacao": "Projetista",
  "telefone": "239976543"

```

Body Cookies Headers (8) Test Results 200 Ok 51 ms 265 B Save Response

Pretty Raw Preview Visualize JSON

1
2
3
 "statusCode": 200

The screenshot shows the Postman application interface. On the left, the sidebar displays 'Team Workspace' with a collection named 'Pontes'. Under this collection, several API endpoints are listed: GET selectVaga, GET selectVagas, GET selectMei, GET selectAllMei, GET selectEfetivacao, GET selectAllEfetivacao, POST insertVaga, POST insertMei, POST insertEfetivacao, DELETE deleteVaga, DELETE deleteMei, DELETE deleteEfetivacao, PUT updateVaga, PUT updateMei, and PUT updateEfetivacao. The 'PUT updateMei' endpoint is currently selected. The main workspace shows a 'PUT' request to 'http://localhost:3000/mei'. The 'Body' tab is selected, showing a JSON payload:

```
1 ... "id_mei": 4,
2 ... "razaoSocial": "AMAZING CLOUD",
3 ... "emp": 9876543,
4 ... "telefone": 23458997654,
5 ... "email": "suporte@cloud.com",
6 ... "regiao_empresa": "maciana",
7 ... "principaisAreas": "cloud, devops e segurança",
8 ... "numeroColaboradores": 240000
```

The response status is 200 OK with a duration of 9 ms and a size of 285 B. The response body is: "statusCode": 200.

This screenshot shows the same Postman interface as the previous one, but with a different selected endpoint. The 'PUT updateEfetivacao' endpoint under the 'Pontes' collection is now selected. The main workspace shows a 'PUT' request to 'http://localhost:3000/efetivacao'. The 'Body' tab is selected, showing a JSON payload:

```
1 ... "id_efetivacao": 4,
2 ... "id_vagas": 5,
3 ... "id_mei": 7
```

The response status is 200 OK with a duration of 17 ms and a size of 285 B. The response body is: "statusCode": 200.