

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS
GERAIS**

INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA

**UNIDADE EDUCACIONAL SÃO GABRIEL
Bacharelado em Sistema de Informação**

RELATÓRIO FINAL

**Alunos: Isabella Bosco Clemente
Leonardo Ribeiro Andrade
Lucas Henrique Simplicio Monteiro
Luísa Ferreira Braz Machado**

**Belo Horizonte
Ano 2023**

TEMPERO DE CASA

ISABELLA BOSCO; LEONARDO RIBEIRO; LUCAS HENRIQUE; LUÍSA BRAZ

Resumo

Nosso site, Tempero de Casa, tem como objetivo ajudar pequenos produtores de alimentos artesanais a divulgarem, venderem e expandirem seus negócios. Além de facilitar para as pessoas de Belo Horizonte que procuram por alimentos mais saudáveis, caseiro, ou mesmo produtos de categoria mais específicas (de comidas veganas ou um bolo personalizado), a acharem tais produtos. Durante esse relatório será descrito como foi feito o modelo de negócios para tornar nosso site viável, a partir disso, será explicado como foi feito o banco de dados, entre outras atividades importantes para o desenvolvimento desse projeto. Além disso, o final desse relatório conterá o resultado obtido junto com alguns testes realizados.

1-Introdução

Para pequenos empreendedores, o mercado alimentício, ainda que muito grande, é uma casca dura de se adentrar. Nos aplicativos consolidados, pequenos empreendedores de comidas artesanais se vêm competindo contra aqueles de maior porte, e dificilmente conseguem conquistar novos clientes

Além disso, os consumidores de produtos orgânicos/não industrializados passam aperto ao procurar por produtores de confiança, os quais ele possa entrar em contato e que mostrem os ingredientes que usam em seus produtos.

O nosso principal objetivo é desenvolver uma solução de software que atenda aos pequenos produtores de alimentos artesanais, auxiliando-os a crescer ou até mesmo a criar seu próprio pequeno negócio. De maneira conjunta, procuramos também auxiliar aos consumidores que procuram um estilo de alimentação onde a produção seja mais orgânica e feita a sua maneira. Seja para sanduíches naturais, seja para bolos de cenoura com chocolate.

Assim como vários talentos da alimentação artesanal são ignorados, vários clientes têm dificuldades em encontrar alimentos orgânicos e artesanais à sua maneira, contentando-se então com aqueles alimentos encontrados em padarias ou lanchonetes que talvez não atendam a seus gostos.

Vale também lembrar que várias pessoas de idade avançada possuem grandes habilidades e prazer com confeitaria, mas tem dificuldades em começar um negócio próprio.

Produtores de alimentos orgânicos/caseiros que buscam novos clientes ou que desejam começar um negócio, mas não possuem grandes conhecimentos tecnológicos para divulgar tais serviços de maneira efetiva.

Consumidores de alimentos orgânicos e/ou caseiros que desejam encontrar bons produtores, seja para consumo pessoal, para eventos, etc, a preços acessíveis; assim como aqueles que desejam alimentos com ingredientes/aparência escolhidos a dedo.

Apesar do software ser aberto para todas as faixas etárias, o desenvolvimento será voltado à população idosa (50 anos acima), visando melhorar a usabilidade e tornar o ambiente mais intuitivo.

2-Trabalho relacionados e referencial teórico

Tivemos como base o sistema que algumas empresas que atuam como ponte entre dois públicos fornecendo uma plataforma como intermediário como a Uber e, posteriormente ao afunilar o princípio da ideia, delimitando ainda mais como referência os princípios de funcionamento do lfood e Horta 31. Apesar de sensível pelo modo de produção artesanal e orgânico ser de pouco interesse a grandes empresas - e logo de pouco interesse pelo público -, a produção independente e orgânica e/ou artesanal é promissora desde que tratada de uma forma mais humana e com uma comunicação e apresentação ao cliente moderna e bem trabalhada tal como propomos.

Ambas as empresas são grandes atuantes no mercado alimentício, onde o lfood oferta uma grande variedade de produtos com um foco no funcionamento dinâmico e apelativo, e a Horta 31 possui uma temática mais jovem, princípios interessantes ao bem estar que procuramos alcançar com nossa proposta. Então, fundamos os diferenciais na dinâmica de interação com o usuário do primeiro, o princípio de melhora social a partir de uma culinária mais saudável de outro e ainda o foco a produtores independentes que seriam sufocados em meio a plataformas que priorizam empresas de grande porte.

Para o funcionamento técnico de nossa plataforma, também visamos a facilidade e conforto de venda por uma busca em região que é feita diretamente a partir do endereço que o usuário fornece para o aplicativo, da qual procede-se uma filtragem que prioriza produtores mais próximos do endereço na lista de opções para consumo. Para além do conforto do cliente, o produtor, igualmente, não é necessário CNPJ - o que promove mais fluidez e menos pressão à produção independente que é abraçada pela plataforma.

3-Metodologia

3.1-Modelo de Negócio (AS IS e TO BE)

AS IS

Nosso modelo de negócio começou como uma plataforma que apresentava para os usuários produtores de comida artesanais, sem fazer o contato direto entre eles.

Em nossa plataforma o consumidor consegue procurar produtores por regiões ou mesmo por categoria de alimentos. Em cada perfil de produtor existe um cardápio com seus produtos ofertados, onde se encontram a descrição desse produto, sua categoria, seu nome, avaliação e valor. Além disso, a princípio a intenção era de colocar como contactar esse produto, ou seja, colocar em sua página seus contatos. Mais para frente, ao partirmos para o Modelo de Negócio TO BE, isso sofre modificações, mas isso será descrito mais para frente desse documento.

Ainda no AS IS, o produtor que desejar cadastrar seu cardápio em nosso site primeiramente tem que fazer seu cadastro. Durante essa parte ele poderá escolher entre uma conta grátis ou uma assinatura premium, que a partir de um pagamento mensal garante ao produtor algumas melhorias em sua página assim como em sua divulgação.



Imagem 01 - Assinatura Premium

TO BE

Ao iniciar nosso modelo TO BE, muitas coisas dos AS IS se mantiveram, porém algumas sofreram modificações e/ou acréscimos. Inicialmente nosso negócio não fazia o contato entre o produtor e consumidor diretamente. Além disso, foram adicionadas algumas funções que antes não existiam, como a confirmação de criação de conta por email.

Com o desenvolvimento TO BE do nosso modelo de negócios verificamos algumas falhas, não só na forma de funcionamento do software, mas no negócio em si. Para tornar nosso negócio lucrativo para a empresa, optamos por transformar o software na forma de contato entre produtor e consumidor e também o local onde irá ocorrer as transações financeiras. Assim, o lucro do software não dependerá exclusivamente de assinaturas prêmios, pois agora iremos ter uma taxa para cada pedido. Dessa forma, acrescentamos ao software formas de pagamento. Caso ocorra algum erro, acrescentamos uma função para perguntar ao consumidor se ele gostaria de tentar novamente ou de cancelar seu pedido. A mesma mecânica foi utilizada para o pagamento da assinatura do prêmio para os produtores. Com essa modificação, agora o consumidor passa a ter que se cadastrar no site assim como o produtor, no entanto a assinatura premium funciona apenas para o produtor.

Acrescentamos também algumas automatizações no processo. Como já citado anteriormente, foi criada a confirmação de criação de conta por email. Além dessa, também utilizamos a automatização na assinatura de prêmios, na finalização de pagamento e também na finalização de pedidos. Nesse último, um e-mail deve ser enviado para o produtor com o pedido feito.

3.2-Arquitetura do Negócio

Na arquitetura do negócio, fizemos uma dinâmica em que o produtor e o cliente se encontram de uma forma direta. Para isso, a forma mais direta, comum e acessível para nosso público seria uma página web simples com a interface gráfica e o fluxo de dados do cliente com a plataforma, do produtor com a plataforma e entre ambos para a conclusão de vendas. Destacamos, também, que todos os públicos são bem vindos, apesar de não ter uma interação mais apelativa - o que é comumente visto em negócios mais tradicionais da área alimentícia.

De uma forma geral, o produtor se junta à plataforma com seus produtos enquanto a plataforma retém uma rede de clientes. Tanto os produtores quanto clientes possuem uma conta onde os dados são disponibilizados, onde, apesar do tratamento ser o mesmo em questão de linguagem, por questões de formalidade e confiança para o processo de vendas ser feito, os produtores terão acesso obrigatório aos processos financeiros sustentados pela plataforma e manejo de controle de demandas referentes ao próprio negócio feitas com cada cliente.

As informações reduzem à dados pessoais (como endereço, nome, email e senha, por exemplo), dados financeiros (como preços e transações) e dados de conta que são disponibilizados exclusivamente para o produtor, nos casos de parceria entre o produtor e a empresa a partir dos planos oficiais de fidelidade à empresa.

Buscamos principalmente fomentar o incentivo relacionado a uma boa alimentação a partir desse processo mais simplificado e fluido de comunicação para ambas as partes de produtor/cliente e uma experiência leve e atraente ao cliente pela adesão ao site e interação com as possibilidades em formato digital. O esforço de operação da plataforma como um todo é direcionada às necessidades do cliente, sendo sustentada a partir das parcerias que permitem maior foco aos produtores que a assinam, sem forçar a conexão e protagonismo dos produtores em relação ao

próprio sistema de negócio e no contato com o cliente de forma pessoal e humanizada.

3.3-Banco de Dados

A primeira parte do banco de dados foi fazer o diagrama de entidade-relacionamento. Decidimos fazer 4 entidades: CLIENTE , PRODUTOR, PRODUTO e PEDIDO.A partir dessas entidades foi definida a relação que cada uma delas tem entre si. Após isso, foi criado quais atributos essas entidades precisam ter, incluindo sua entidade principal. A figura abaixo mostra o resultado final desse diagrama.

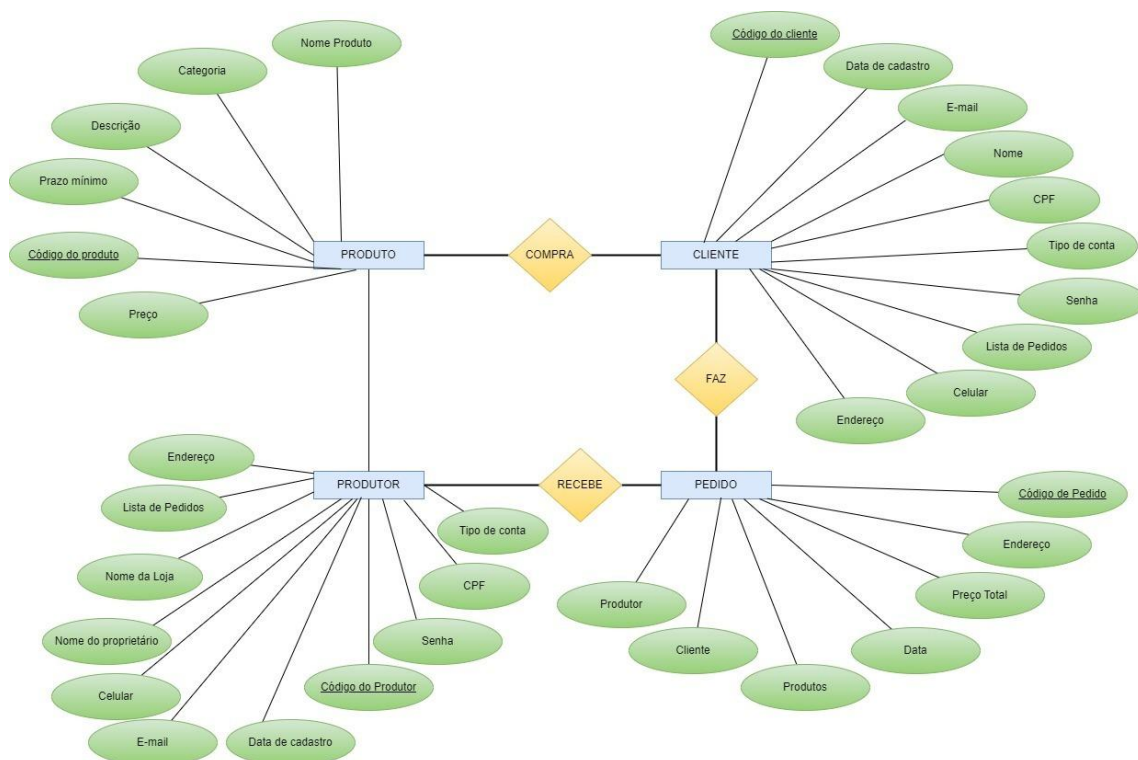


Imagem 02 - Diagrama de Entidade-Relacionamento

Com a finalização do diagrama foi possível criar as tabelas de modelo relacional mais facilmente e assim, logo depois monta-las no SQL, programa no qual fizemos a parte do banco de dados para aplicar ao site.

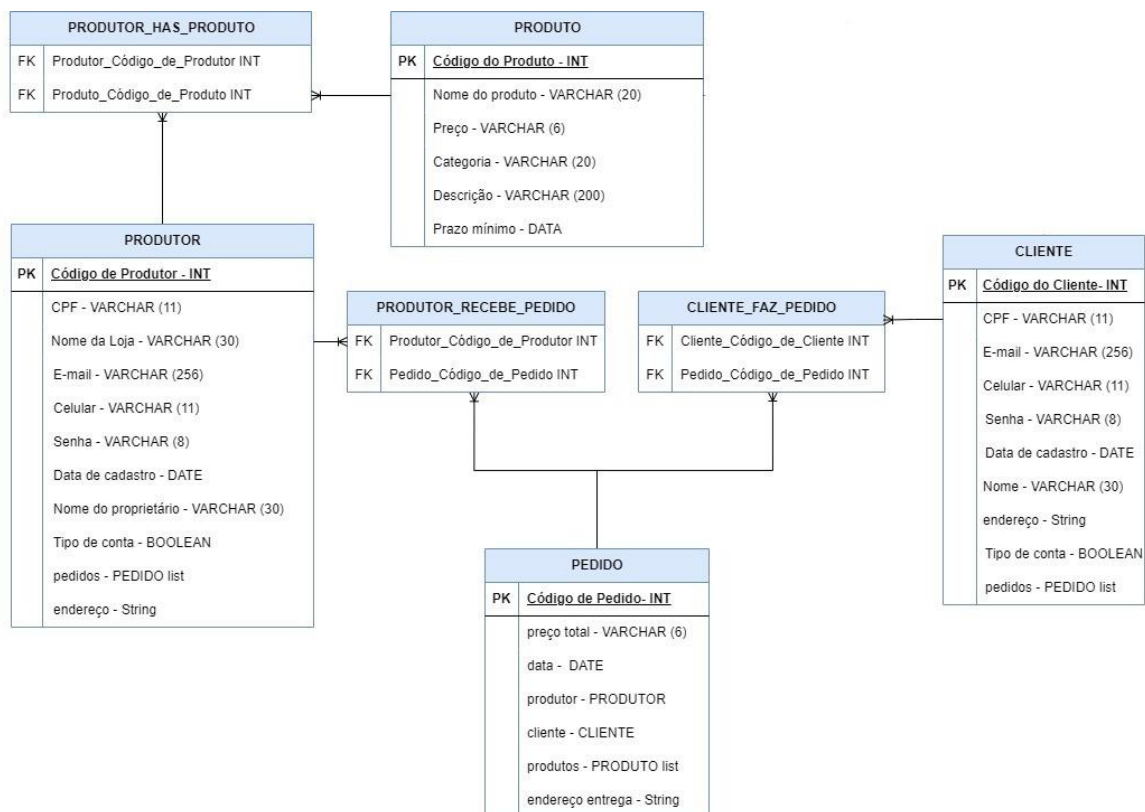


Imagem 03 - Modelo Relacional

	Nome da Coluna	Tipo de Dados	Permitir Nulos
?	Codigo_produto	int	<input type="checkbox"/>
	Nome	varchar(50)	<input type="checkbox"/>
	Preco	float	<input type="checkbox"/>
	Categoria	varchar(200)	<input type="checkbox"/>
	Descricao	varchar(MAX)	<input type="checkbox"/>
	Prazo_min	varchar(50)	<input type="checkbox"/>
	produtor	int	<input type="checkbox"/>
	Imagem	varchar(MAX)	<input type="checkbox"/>

Tabela 01 - Tabela SQL Produto

	Nome da Coluna	Tipo de Dados	Permitir Nulos
🔑	Codigo_produto	int	<input type="checkbox"/>
	Nome_loja	varchar(50)	<input type="checkbox"/>
	Email	varchar(256)	<input type="checkbox"/>
	Celular	varchar(11)	<input type="checkbox"/>
	Senha	varchar(8)	<input type="checkbox"/>
	Data_cadastro	varchar(10)	<input type="checkbox"/>
	Nome_proprietario	varchar(50)	<input type="checkbox"/>
	Tipo_conta	bit	<input type="checkbox"/>
	Endereco	varchar(MAX)	<input type="checkbox"/>
	CPF	varchar(11)	<input type="checkbox"/>
	Imagem	varchar(MAX)	<input type="checkbox"/>

Tabela 02 - Tabela SQL Produtor

	Nome da Coluna	Tipo de Dados	Permitir Nulos
🔑	Codigo_pedido	int	<input type="checkbox"/>
	Preco_total	money	<input type="checkbox"/>
	Data	varchar(10)	<input type="checkbox"/>
	Endereco_entrega	varchar(MAX)	<input type="checkbox"/>
	Cliente	int	<input type="checkbox"/>
	Produtor	int	<input type="checkbox"/>

Tabela 03 - Tabela SQL Pedido

	Nome da Coluna	Tipo de Dados	Permitir Nulos
🔑	Codigo_cliente	int	<input type="checkbox"/>
	Email	varchar(256)	<input type="checkbox"/>
	Celular	varchar(11)	<input type="checkbox"/>
	Senha	varchar(8)	<input type="checkbox"/>
	Data_cadastro	varchar(10)	<input type="checkbox"/>
	Nome	varchar(50)	<input type="checkbox"/>
	Endereco	varchar(MAX)	<input type="checkbox"/>
	Tipo_conta	bit	<input type="checkbox"/>
	CPF	varchar(11)	<input type="checkbox"/>

Tabela 04 - Tabela SQL Cliente

4-Resultados

4.1-Relatório de Melhoria

Desde o começo do projeto decidimos utilizar o SQL Server Management Studio da Microsoft para monitorar o banco de dados, por ser uma ferramenta que utilizamos nas aulas de Banco de Dados e por ter sido indicação do professor da matéria do trabalho. Porém, como em todos projetos nós tivemos problemas. Tanto problemas mais simples como erros de digitação, como mais complexos como o fato de um dos nossos integrantes estar com sistema Linux no seu computador, logo ele não pode baixar o SSMS e fazer os testes do banco de dados em seu computador. Mas enfrentamos esses problemas, mesmo sendo tarde, nós fomos atrás e tentamos o nosso melhor.

4.2-Comando associados ao SQL

Utilizamos comandos associados ao SQL para podermos inserir, atualizar, deletar e buscar dados em nosso banco de dados. Com JavaScript, Node.js e Express nós fizemos as rotas, comandos e ligações com o banco de dados para que certas funções do nosso site funcionassem.

```
async function getprodutos(){
  try{
    let pool = await sql.connect(config);
    let TemperodeCasa = await pool.request().query("SELECT * from [dbo].[Produtos]");
    return TemperodeCasa.recordsets;
  }
  catch (erro){
    console.log(erro);
  }
}
```

Imagem 04 - Comando getprodutos para pegar todos os produtos do banco

```
async function deletepedido(pedidoCodigo_pedido){
  try{
    let pool = await sql.connect(config);
    let TemperodeCasa = await pool.request()
      .input('input_parameter', sql.Int, pedidoCodigo_pedido)
      .query("DELETE from [dbo].[Pedidos] where Codigo_pedido = @input_parameter");
    return TemperodeCasa.recordsets;
  }
  catch (error) {
    console.log(error);
  }
}
```

Imagem 05 - Comando delete pedido para deletar um pedido específico

```

async function getproduto(produtor){
  try{
    let pool = await sql.connect(config);
    let TemperodeCasa = await pool.request()
      .input('input_parameter', sql.Int, produtor)
      .query("SELECT * from [dbo].[Produtos] where produtor = @input_parameter");
    return TemperodeCasa.recordsets;
  }
  catch (erro){
    console.log(erro);
  }
}

```

Imagem 06 - Comando getproduto com parâmetro para pegar todos os produtos de um certo produtor

```

async function updtCliente(cliente){
  try{
    let pool = await sql.connect(config);
    let TemperodeCasa = await pool.request()
      .input('input_parameter', sql.Int, cliente.Codigo_cliente)
      .query(`UPDATE [dbo].[Clientes]
        SET
          [Email] = '${cliente.Email}'
          ,[Celular] = '${cliente.Celular}'
          ,[Senha] = '${cliente.Senha}'
          ,[Data_cadastro] = '${cliente.Data_cadastro}'
          ,[Nome] = '${cliente.Nome}'
          ,[Endereco] = '${cliente.Endereco}'
          ,[CPF] = '${cliente.CPF}'
        WHERE Codigo_cliente = @input_parameter`);
    return TemperodeCasa.recordsets;
  }
  catch (error) {
    console.log(error);
  }
}

```

Imagem 07 - Comando updtCliente para atualizar um cliente específico

```

async function addprodutor(produtor){
  try{
    let pool = await sql.connect(config);
    let TemperodeCasa = await pool.request()
    .query(`INSERT INTO [dbo].[Produtores](
      [Codigo_produtor]
      ,[CPF]
      ,[Email]
      ,[Celular]
      ,[Senha]
      ,[Data_cadastro]
      ,[Nome_proprietario]
      ,[Endereco]
      ,[Tipo_conta]
      ,[Nome_loja]
      ,[Imagem]
    ) VALUES (
      '${produtor.Codigo_produtor}',
      '${produtor.CPF}',
      '${produtor.Email}',
      '${produtor.Celular}',
      '${produtor.Senha}',
      '${produtor.Data_cadastro}',
      '${produtor.Nome_proprietario}',
      '${produtor.Endereco}',
      '${produtor.Tipo_conta}',
      '${produtor.Nome_loja}',
      '${produtor.Imagem}'
    )`);
    return TemperodeCasa.recordsets;
  }
  catch (error){
    console.log(error);
  }
}

```

Imagem 08 - Comando addprodutor para adicionar um produtor

4.3-Testes realizados

Foram realizados testes para ver se o banco de dados recebia os resultados dos comandos associados ao SQL mostrados anteriormente. Nós utilizamos a ferramenta Postman para verificar se as rotas e os comandos estavam funcionando.

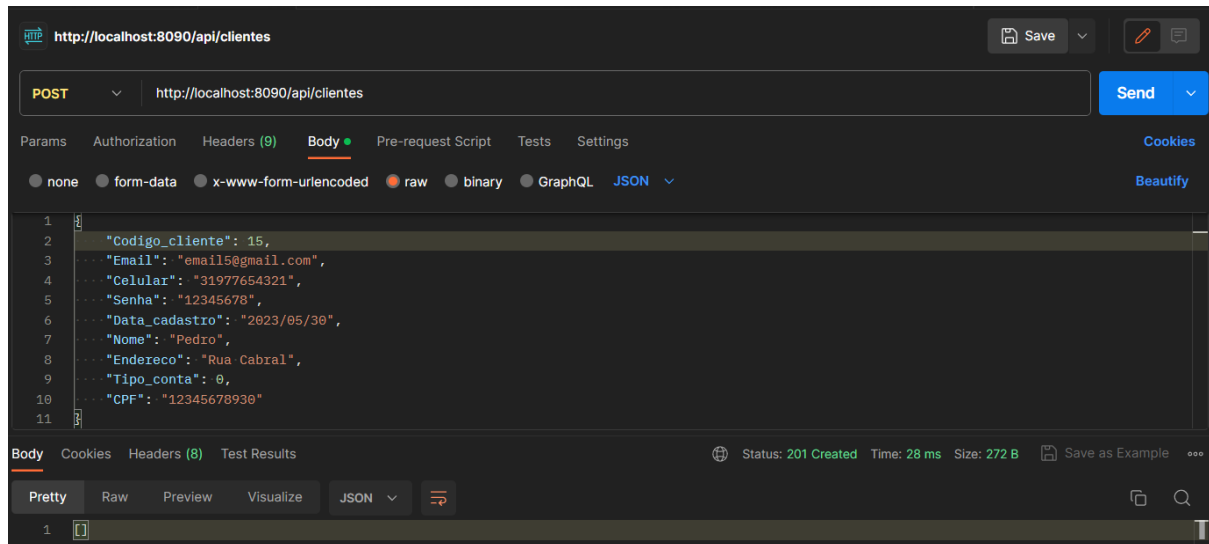


Imagem 09 - Teste de post (addcliente) no postman, dando 201 create

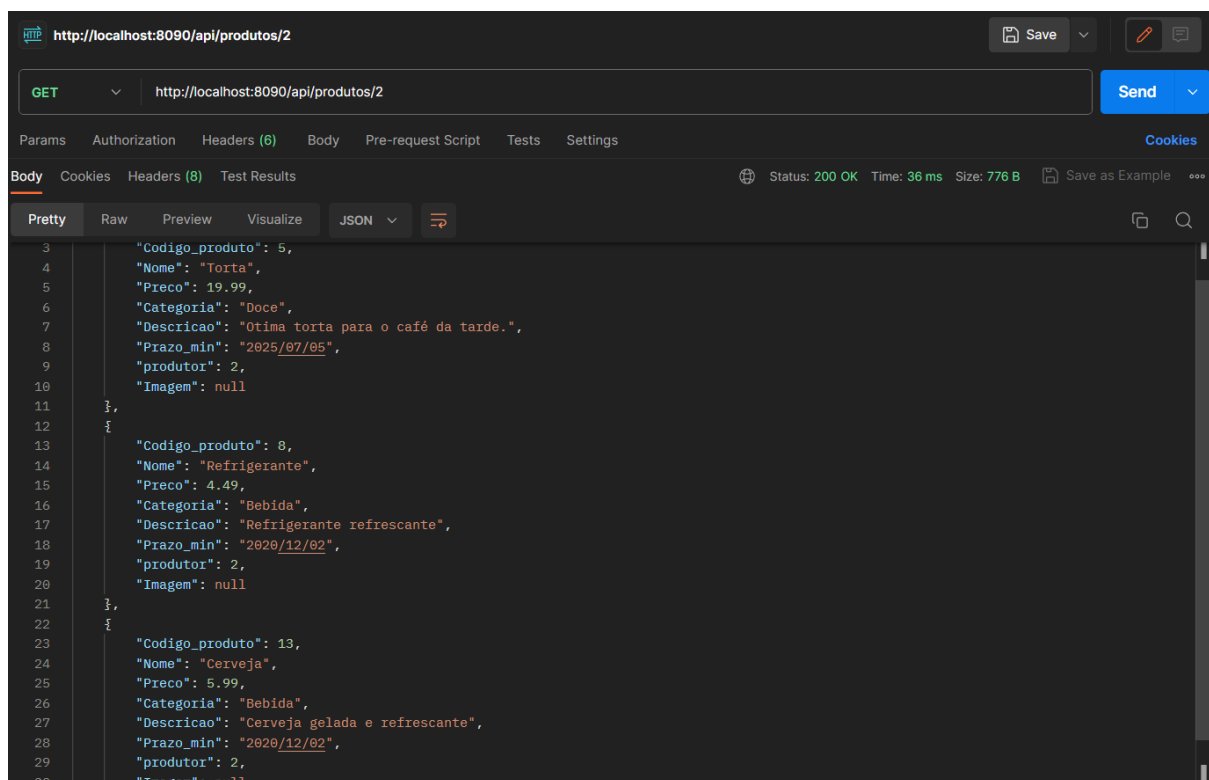


Imagem 10 - Teste de get dos produtos do produtor de código 2

5-Conclusão

Por fim, a plataforma que define a solução definida primariamente atende ao que problema delimitado ao início no projeto apresenta, realmente, apesar que ainda num estágio não absolutamente elaborado.

Para próximos passos, definimos, principalmente, a elaboração de normas mais estritas com questões legais, tanto para o produtor, quanto pela empresa e o cliente em segurança mantendo a proposta de contato entre todas as entidades que entrem sob responsabilidade da empresa. Além disso, algumas melhorias técnicas pontuais para melhor desempenho, leveza e eficiência bruta da forma web a partir da forma que se mostra atualmente e outras possíveis formas de contato e adesão do cliente à empresa.

Em suma, apresentamos uma solução simples que une a interação por página web e banco de dados para a melhor integração da plataforma que desenvolvemos antes conceitualmente - sendo feita pela disponibilização dos produtos e conta de produtor pela página principal, recomendações e diretamente pela página do produtor. E apesar das dificuldades e constante trabalho nos erros na integração entre interface e banco de dados ou na estruturação prática da solução em forma alinhada ao propósito do negócio, se finalizou notavelmente íntegra e estável.