

20 de setembro de 2024

NoSQL: Um pequeno passo na modelagem, um grande salto na eficiência



```
[ApiController]
[Route("api/[controller]")]
public class LeadSoftController : ControllerBase
{
    [HttpGet("missao")]
    public async Task<ActionResult<string>> GetMissao()
    {
        var missao = await leadsRepository.GetMissao(2024);
        if (missao != null)
            return Ok(@criamos soluções tecnológicas
                      para negócios visionários.");
        else
            return BadRequest("Have another coffee and try again.");
    }
}
```

Leadsoft[®]
SOLUÇÕES WEB

RavenDB

Apertem os cintos, estamos prestes a decolar...

Alguns elementos tradicionais

- λ DER – Diagramas de Entidade Relacionamento
- λ Tabelas, tabelas de relacionamento
- λ Connection strings
- λ Views
- λ Procedures e Functions
- λ Triggers
- λ Chaves primárias e estrangeiras (PK, FK, Unique)
- λ Null, Not Null...



SQL pode ser visto
como poderoso. E é.

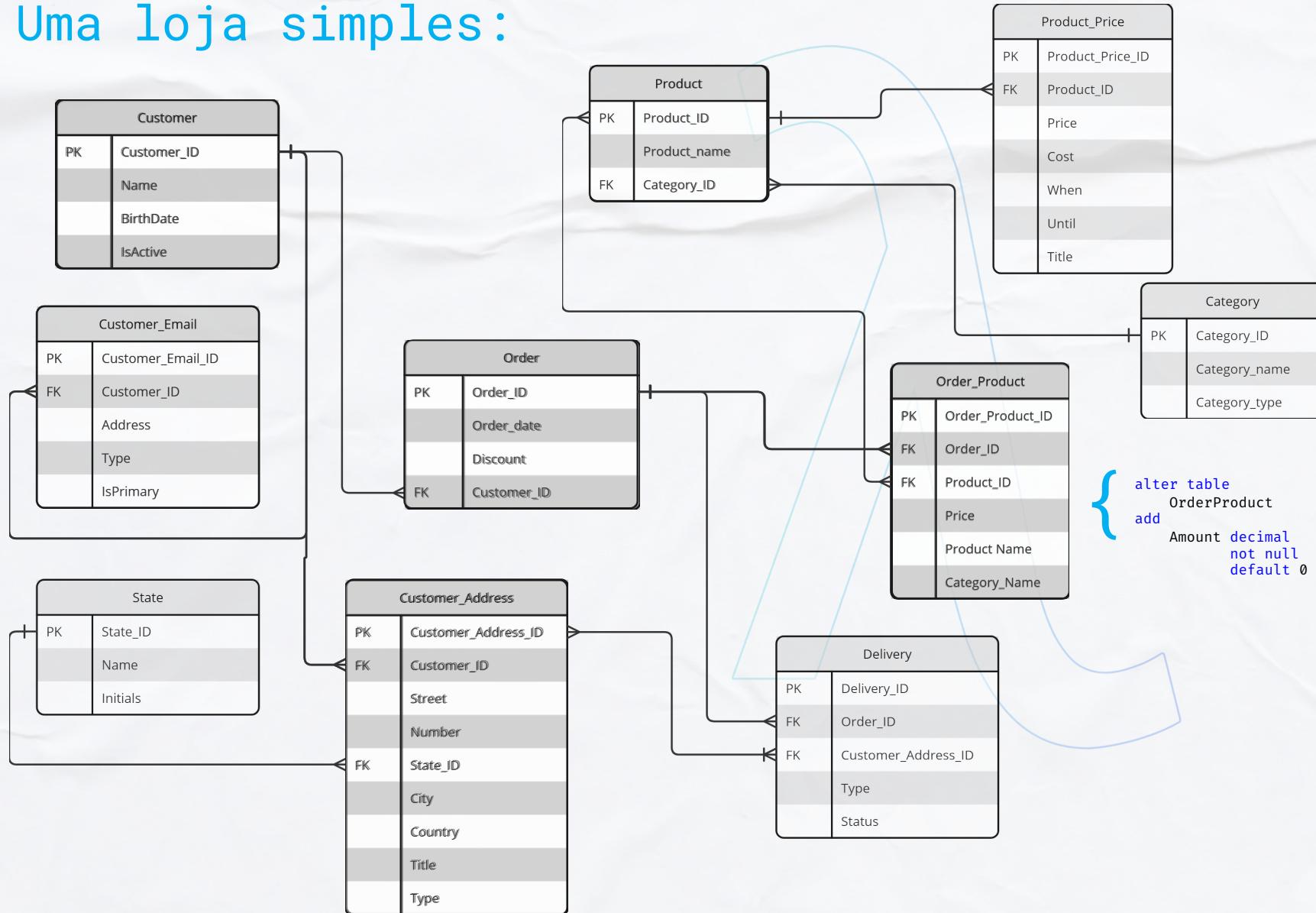
Mas a que custo e
esforço?
Vamos rever alguns
pontos juntos...

Alguns comandos conhecidos

- λ Select
- λ Inner, left, outer e right join
- λ Update
- λ Delete (não esqueça do where!!)
- λ Create, alter, drop...
- λ Cursor... ☹
- λ Commit, rollback, begin tran...

A visualização tabelar é bem familiar...

Uma loja simples:



```

SELECT
    ord.Order_ID AS 'NumeroPedido',
    ord.OrderDate AS 'DiaDaCompra',
    (
        select
            count(op.Order_Product_ID)
        from
            OrderProduct op
        where
            op.Order_ID = ord.Order_ID
    ) AS 'QtdeItens',
    (
        select
            sum(op.Price)
        from
            OrderProduct op
        where
            op.Order_ID = ord.Order_ID
    ) AS 'TotalItens',
    ord.Discount AS 'Desconto',
    (
        select
            sum(op.Price)
        from
            OrderProduct op
        where
            op.Order_ID =
                - ord.Discount) AS 'TotalDaVenda',
    cus.Name AS 'EntregaPara',
    del.Type AS 'TipoEntrega',
    del.Status AS 'StatusEntrega',
    sta.Name AS 'EntregaEm'
FROM
    Order ord
    INNER JOIN Customer cus ON cus.IsActive = 1
    and cus.Customer_ID = ord.CUSTOMER_ID
    LEFT JOIN Delivery del ON del.Order_ID = ord.Order_ID
    LEFT JOIN Customer_Address cua ON cua.Customer_Address_ID =
        del.Customer_Address_ID
    and (
        cua.State_ID IN (
            select
                s.STATE_ID
            from
                State s
            where
                s.Initials in ('SP', 'MG', 'RJ', 'ES')
        )
    )
    LEFT JOIN State sta ON sta.State_ID = cua.State_ID
WHERE
    ord.Discount > 0
ORDER BY
    Order_date DESC
    
```

sum(op.Price * ap.Amount)

sum(op.Price * ap.Amount)

alter table OrderProduct add Amount decimal not null default 0



Em um contexto de banco de dados SNI

Em sistema
assustador.

... Depois
de relações.



código
abelas.

as

ns com
mação.

desenvolver. Mais caro para manter.

Mas então, como simplificar?

TODO:

- λ Foco aos conceitos de algoritmos e estruturas de dados
- λ Esquecer a necessidade de fazer joins.
- λ Não pense no banco de dados.
- λ Use sua energia na construção das suas classes:
 - λ O domínio de dados deve ser o núcleo da sua aplicação.
 - λ Não crie classes anêmicas, concentre-se nas regras de negócio e métodos de validação dentro das classes.
 - λ Organize os dados conforme a regra de negócio necessita.
 - λ Use e abuse de data annotations nas propriedades;

```
/// <summary>
/// Desconto aplicado ao pedido, que deve ser entre 0% e 80%.
/// O desconto é representado como um valor decimal (0 a 0.8), onde 0 equivale a 0% e 0.8 equivale a 80%.
/// </summary>
[Range(0, 80, ErrorMessage = "Des
5 references | 1/1 passing
public decimal Discount { get; pr
    /// <summary>
    /// Define o desconto do pedido, garantindo que o valor esteja entre 0% e 80%.
    /// </summary>
    /// <param name="discount">Desconto a ser aplicado (em porcentagem).</param>
    /// <returns>Retorna a própria instância do pedido com o desconto aplicado.</return
3 references
public Order SetDiscount(decimal discount)
{
    if (discount < 0)
        Discount = 0;
    else
        Discount = discount > 80 ? 0.80m : discount / 100;

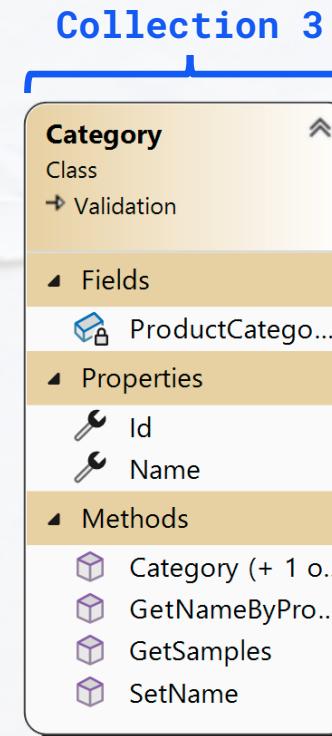
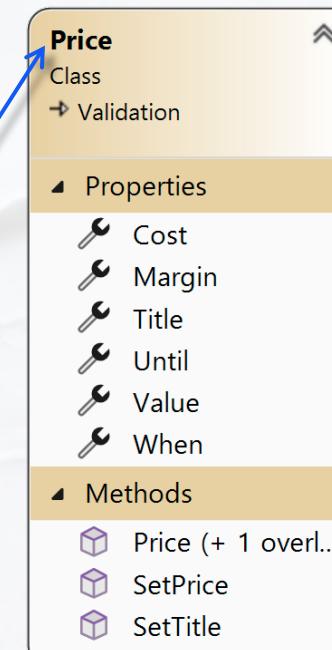
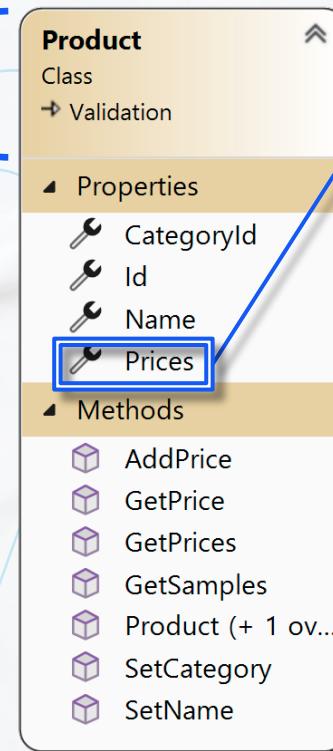
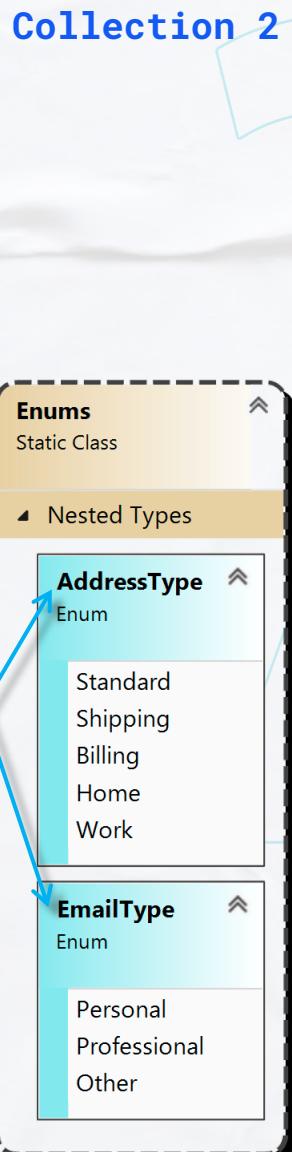
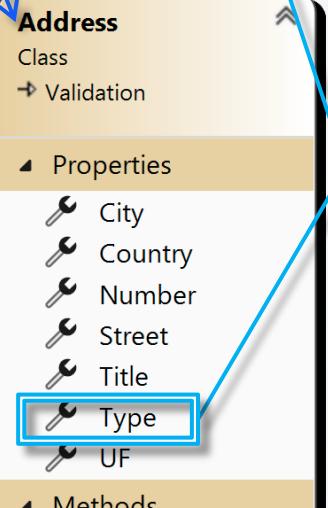
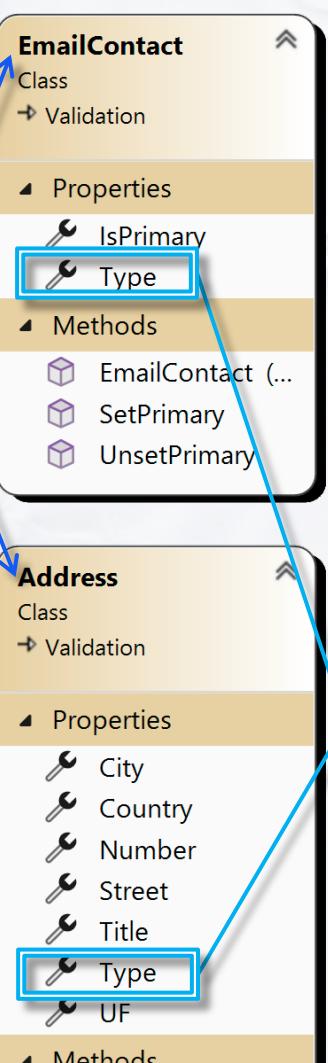
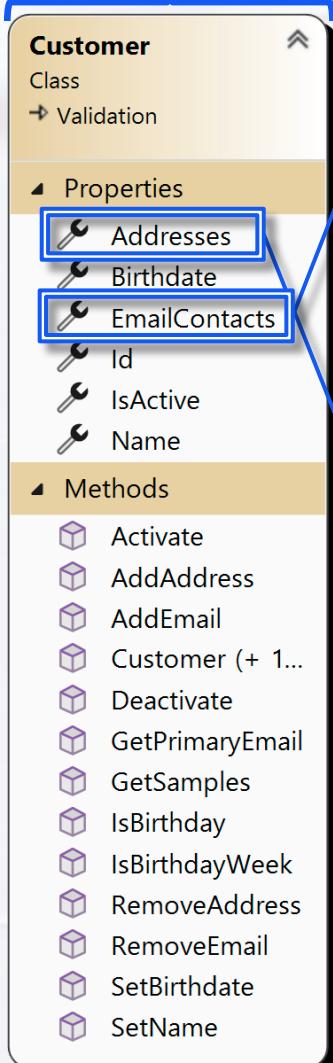
    return this;
}
```



A vida é uma aventura!

- Seu banco de dados não deveria ser...

Collection 1



As classes que não
são collections, são
entidades de valor.



A vida é uma aventura!

- Seu banco de dados não deveria ser...



A simplicidade de consultas usando LINQ

Query

Syntax ?

Query em RQL dentro do RavenDB Studio

```
1 from Orders where Shipping != null and
2             Shipping.Type != "PickUp" and
3             Shipping.Address != null and
4             Shipping.Address.UF in ("RJ", "MG", "SP", "ES") and
5             Discount > 0.00
6 order by When desc
7 select Number,
8       When,
9       Items.Count,
10      TotalItemsCurrency,
11      DiscountPercent,
12      TotalCurrency,
13      Consumer.Name,
14      Shipping.Type,
15      Shipping.Status,
16      Shipping.Address.UF
```

Results ↗

Index 'Auto/Orders/ByDiscountAndShippingAndShipping.AddressAndShipping.Address.UFAndShipping.TypeAndWhen' was used to get the results in 0 ms ⓘ

Delete documents

Preview	@id	Number	When	Items.Count	TotalItemsCurrency	DiscountPercent
	orders/50-A	20240.050	2024-09-15T21:26:23....	1	R\$ 150.000,00	5,00%
	orders/42-A	20240.042	2024-09-15T21:26:23....	1	R\$ 720.000,00	10,00%

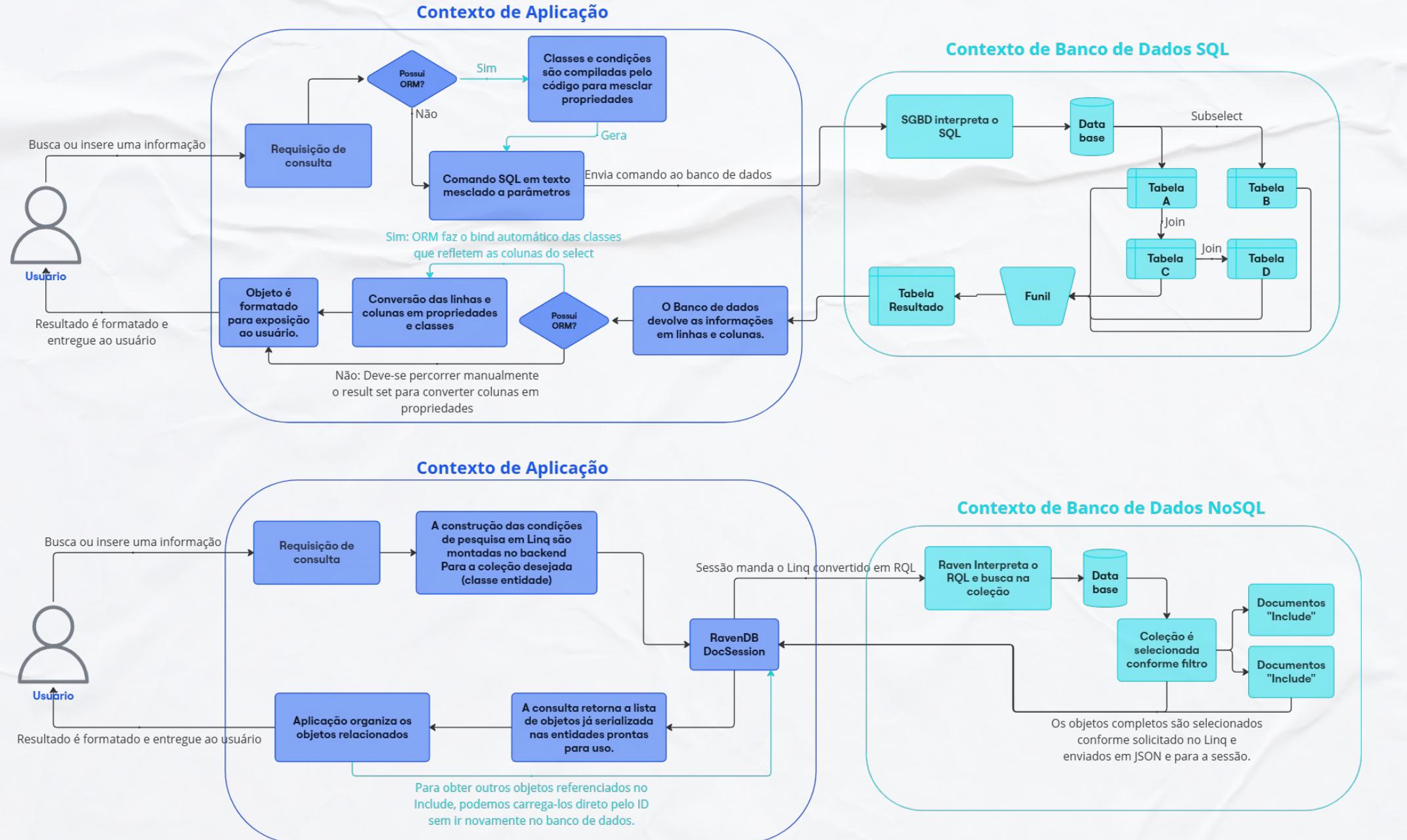
```
SELECT
    ord.Order_ID AS 'NumeroPedido',
    ord.OrderDate AS 'DiaDaCompra',
    (
        select
            count(op.Order_Product_ID)
        from
            OrderProduct op
        where
            op.Order_ID = ord.Order_ID
    ) AS 'QtdeItens',
    (
        select
            sum(op.Price)
        from
            OrderProduct op
        where
            op.Order_ID = ord.Order_ID
    ) AS 'TotalItens',
    ord.Discount AS 'Desconto',
    (
        select
            sum(op.Price)
        from
            OrderProduct op
        where
            op.Order_ID =
                ) - ord.Discount) AS 'TotalDaVenda',
    cus.Name AS 'EntregaPara',
    del.Type AS 'TipoEntrega',
    del.Status AS 'StatusEntrega',
    sta.Name AS 'EntregaEm'
FROM
    Order ord
INNER JOIN Customer cus ON cus.IsActive = 1
and cus.Customer_ID = ord.CUSTOMER_ID
LEFT JOIN Delivery del ON del.Order_ID = ord.Order_ID
LEFT JOIN Customer_Address cua ON cua.Customer_Address_ID =
    del.Customer_Address_ID
and (
    cua.State_ID IN (
        select
            s.STATE_ID
        from
            State s
        where
            s.Initials in ('SP', 'MG', 'RJ', 'ES')
    )
)
LEFT JOIN State sta ON sta.State_ID = cua.State_ID
WHERE
    ord.Discount > 0
ORDER BY
    Order_date DESC
```

Não precisa de Join?

- λ O Include, na query, vai carregar os Ids dos documentos contidos nas propriedades. Assim, ele vai processar tudo uma vez só e trazer na sessão.
 - λ Ao utilizar Load destes objetos na sessão, os documentos serão carregados da memória, sem ir no servidor.
- λ E então você carrega estes documentos e os utiliza como precisar. Sem Joins.

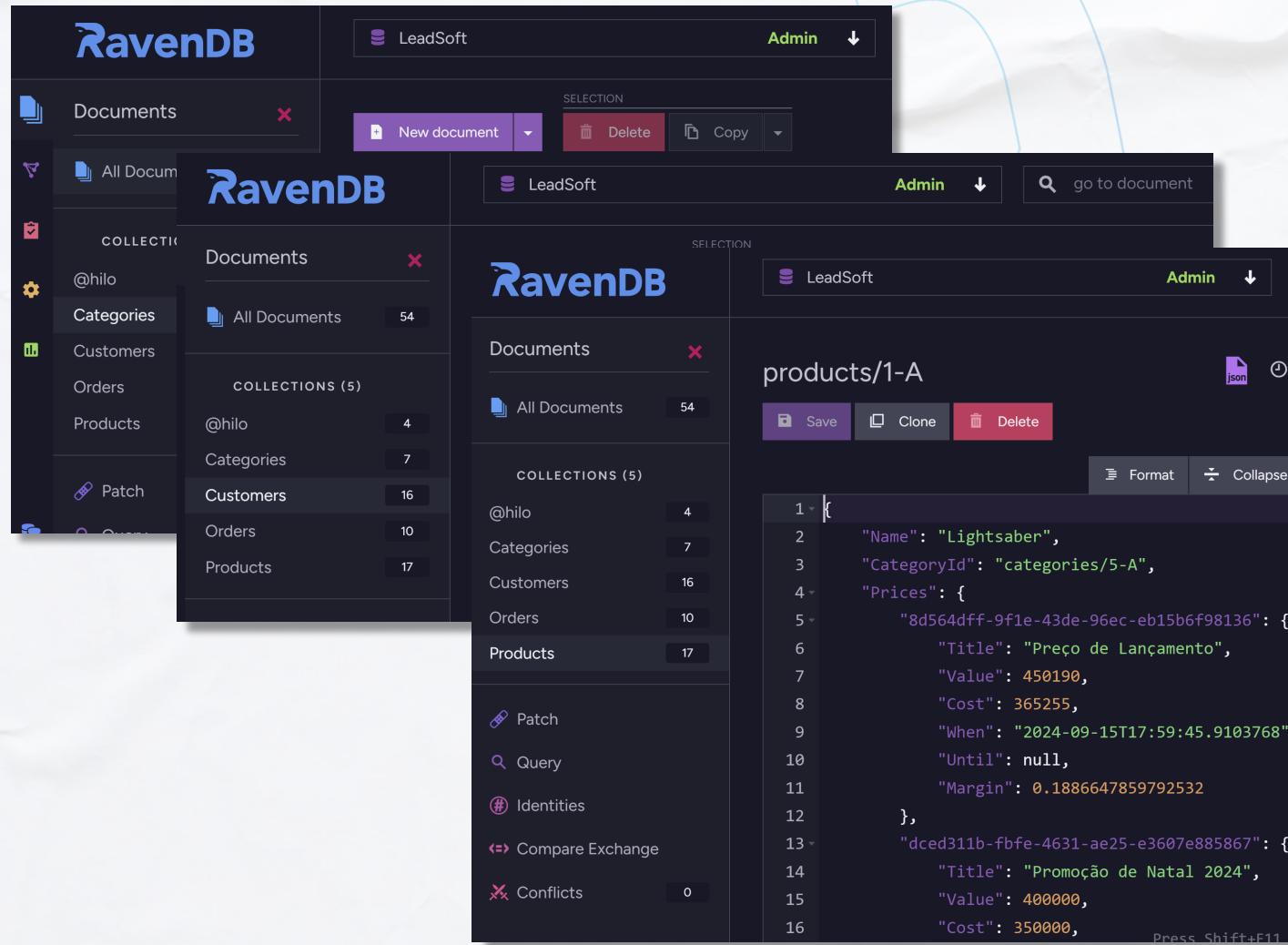
```
.Query<Order>()
    .Include(o => o.Consumer.Id)
    .Include(o => o.Items.Select(i => i.Product.Id))
    .Include(o => o.Items.Select(i => i.Product.CategoryId))
    .Where(o => o.Shipping != null &&
               o.Shipping.Type != OrderShippingType.PickUp &&
```

Entendo as rotas em cada caso



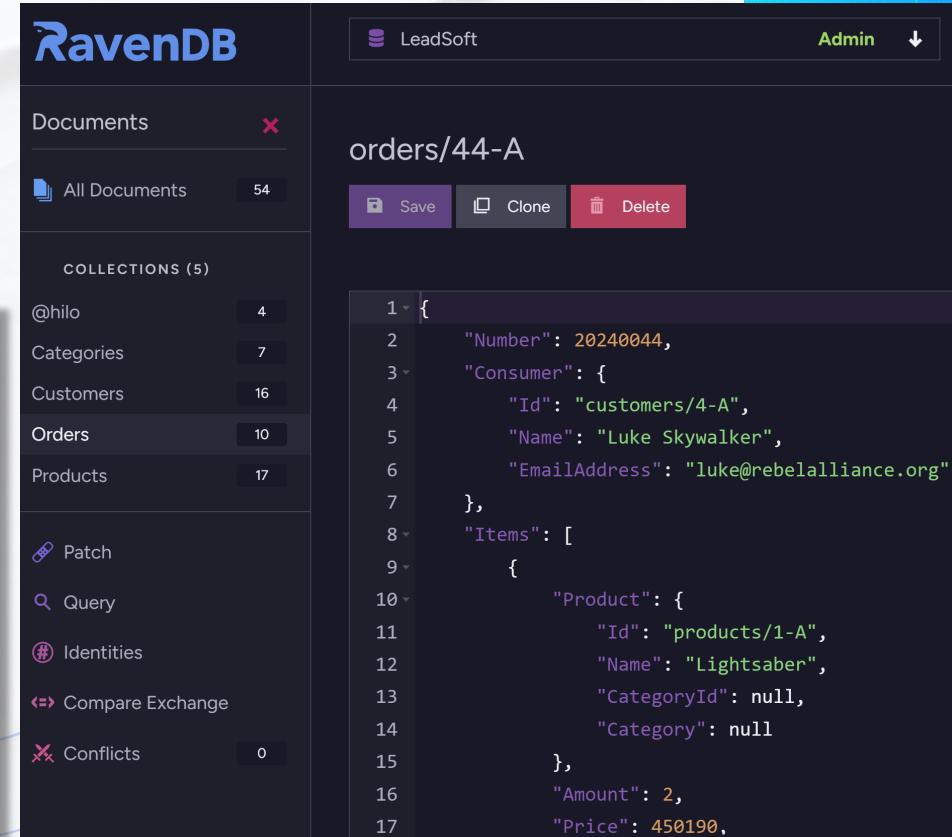
E meus dados, como ficam?

- Simples, em documentos formato JSON. Você pode acessar pelo RavenDB Studio Web. Simples e prático de se utilizar, sem precisar instalar. O RavenDB é uma solução completa!



The screenshot shows the RavenDB Studio Web interface. On the left, there's a sidebar with navigation links: Documents, All Documents, Categories, Customers, Orders, Products, Patch, Query, Identities, Compare Exchange, and Conflicts. The main area displays a hierarchical document structure under the collection 'products/1-A'. At the top, there are buttons for New document, Delete, Copy, and go to document. Below that, there's a list of documents with their names and counts: @hilo (4), Categories (7), Customers (16), Orders (10), and Products (17). The bottom part of the interface shows the JSON content of a selected document, which is a Lightsaber.

```
1 {
2   "Name": "Lightsaber",
3   "CategoryId": "categories/5-A",
4   "Prices": {
5     "8d564dff-9f1e-43de-96ec-eb15b6f98136": {
6       "Title": "Preço de Lançamento",
7       "Value": 450190,
8       "Cost": 365255,
9       "When": "2024-09-15T17:59:45.9103768",
10      "Until": null,
11      "Margin": 0.1886647859792532
12    },
13    "dced311b-fbfe-4631-ae25-e3607e885867": {
14      "Title": "Promoção de Natal 2024",
15      "Value": 400000,
16      "Cost": 350000,
```

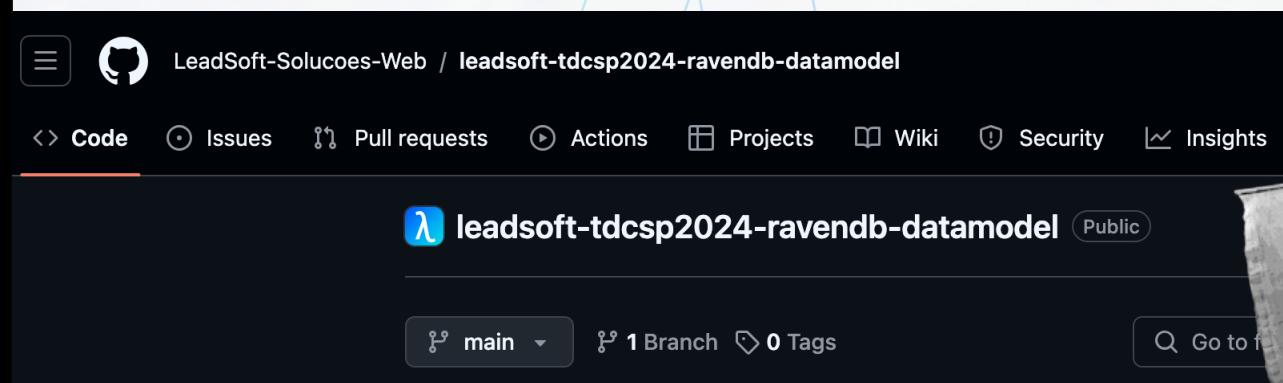


This screenshot shows the RavenDB Studio Web interface with a document content editor. The top bar includes 'LeadSoft' and 'Admin' buttons. The main area is titled 'orders/44-A' and contains buttons for Save, Clone, and Delete. To the right, there's a list of collections: @hilo (4), Categories (7), Customers (16), Orders (10), and Products (17). Below that are links for Patch, Query, Identities, Compare Exchange, and Conflicts. The bottom half of the screen shows the JSON content of the document 'orders/44-A', which includes a list of items, each with a product ID, name, category ID, and price.

```
1 {
2   "Number": 20240044,
3   "Consumer": {
4     "Id": "customers/4-A",
5     "Name": "Luke Skywalker",
6     "EmailAddress": "luke@rebelalliance.org"
7   },
8   "Items": [
9     {
10       "Product": {
11         "Id": "products/1-A",
12         "Name": "Lightsaber",
13         "CategoryId": null,
14         "Category": null
15       },
16       "Amount": 2,
17       "Price": 450190,
```

Quer saber mais como usar RavenDB?

- λ Anteriormente à esta apresentação, no workshop **RavenDB: Simples, ágil e eficiente**, Fred falou sobre os comandos básicos para começar a utilizar o RavenDB.
- λ Baixe o repositório exemplo desta apresentação e faça seus testes.
 - λ Comentem no repositório e abram Issues e que tiraremos suas dúvidas por lá.



The screenshot shows a GitHub repository page. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, and Insights. Below the navigation bar, the repository name 'leadsoft-tdcsp2024-ravendb-datamodel' is displayed, along with a 'Public' badge. Underneath the repository name, there are buttons for 'main' (with a dropdown arrow), '1 Branch', and '0 Tags'. A search bar with the placeholder 'Go to file...' is also visible. To the left of the screenshot is a large QR code.

Clique para abrir o link.



Sobre o palestrante



LUCAS R. TAVARES

Founder & CEO

- λ Natural de Belo Horizonte/MG, 33 anos
 - λ Bacharel em Sistemas de Informação – Faculdade Cotemig MG



- λ Possui uma ampla experiência no desenvolvimento de soluções tecnológicas na área do software. Fundou a LeadSoft em 2018 em Curitiba, com o propósito de otimizar resultados e impactar a vida das pessoas com software de alta qualidade em negócios visionários.
- λ Desenvolvedor Delphi e asp.net C# desde 2008.
- λ Arquiteto em .Net desde 2017 (Core 2.1)
- λ Pacotes NuGet .net 8 da LeadSoft
- λ Aficionado pelo RavenDB desde 2018
- λ Especialista em SQL Server por 10 anos
 - λ Nunca mais optou por desenhar aplicações usando SQL desde que começou a utilizar RavenDB



The Developer Week
San Francisco 2024

```
[ApiController]
[Route("api/[controller]")]
public class LeadSoftController : ControllerBase
{
    [HttpGet("missao")]
    public async Task<ActionResult<string>> GetMissao()
    {
        var missao = await LeadSoftRepository.GetMissao(2024);
        if (missao != null)
            return Ok(@$"Criamos soluções tecnológicas
                    para negócios visionários");
        else
            return BadRequest();
    }
}
```

A LeadSoft

return Ok(@\$"Criamos soluções tecnológicas
 para negócios visionários");

**Criamos soluções tecnológicas para negócios visionários,
cativando pessoas por toda a galáxia.**



Cientes que decolaram

Leadsoft
SOLUÇÕES WEB



Manifesto

A LeadSoft® é mais do que uma empresa de soluções tecnológicas: somos os propulsores que fazem ideias decolarem e negócios explorarem além dos limites conhecidos, vislumbrando horizontes nunca antes alcançados.

Com lealdade e honestidade como nossos guias, trabalhamos olho-no-olho com cada parceiro e cliente. Valorizamos a colaboração e proatividade, pois entendemos que para olhar para além dos cosmos, precisamos fazer juntos, e com total sinergia.

Nosso compromisso vai além de simples soluções: é com a humildade de aprender e empatia de ouvir as ideias das outras pessoas, que fazemos com que o conhecimento seja o combustível que alimenta nossos propulsores, e a comunicação clara e eficaz o que nos mantém alinhados na mesma trajetória.

Não apenas sonhamos com o futuro, mas trabalhamos arduamente para construí-lo, experimentando coisas novas para manter um constante progresso, e isso é o que torna a LeadSoft® um laboratório de ideias, e nós, os cientistas à frente da inovação!

Embarque em nossa nave e venha explorar possibilidades infinitas entre as estrelas.

Juntos, DECOLAREMOS!



```
[ApiController]  
[Route("api/[controller]")]  
public class LeadSoftController : ControllerBase  
{  
    [HttpGet("missao")]  
    public async Task<ActionResult<string>> GetMissao()  
    {  
        var missao = await _repository.Missao.FindAsync(1);  
        if (missao != null)  
            return Ok(@$"Criamos soluções tecnológicas  
para negócios visionários");  
        else  
            return BadRequest();  
    }  
}
```

Mais sobre nós



Saiba mais
sobre nossos
serviços e
soluções



Faça parte
da
Tripulação
LeadSoft



Sobre nossa
parceria
com o
RavenDB





RavenDB

```
[ApiController]
[Route("api/[controller]")]
public class LeadSoftController : ControllerBase
{
    [HttpGet("missao")]
    public async Task<ActionResult> GetMission()
    {
        var missao = await leadSoftRepository.GetMission(2024);
        if (missao != null)
            return Ok($"Conheça nossas soluções tecnológicas para negócios visionários");
        else
            return BadRequest();
    }
}
```

Leadsoft[®]

SOLUÇÕES WEB



LEADSOFT.INFO



LinkedIn

Siga-nos nas redes sociais
Juntos, decolaremos!