

Desafio Umbler: Refatoração & Arquitetura

O Cenário Original

Mapeamento e identificação dos gargalos críticos encontrados
na estrutura do código legado e na infraestrutura.

Principais Dores Encontradas



**Alto
Acoplamento**
O código era tecnicamente
acoplado ao sistema de
bibliotecas externas
(LookupClient/WhoisClient)
diretamente, inviabilizando
qualquer teste isolado.



**Bloqueio de
Infra**
Tentativas de acesso ao MySQL
da Umbler resultavam em
Timeout constante devido a
bloqueios de rede na porta
3306.



**Vazamento de
Dados**
A API retornava a Entidade de
domínio inteira diretamente
para o client, expondo chaves
primárias e tempos de cache
(TTL).



**UX
Precária**
A interface do usuário exibia o
retorno da consulta Whois como
um bloco de texto JSON cru e
desestruturado na tela.

Soluções Arquiteturais

Aplicando princípios S.O.L.I.D, Clean Code e
Padrões de Projeto robustos na camada de Back-end.

O Desafio da Infraestrutura

A transição pragmática de banco de dados para destravar o ambiente de desenvolvimento local.

- O Problema:** O firewall local impedia o acesso ao banco remoto, causando quebras na execução do Entity Framework.
- A Decisão:** Substituição da dependência do MySQL pelo provedor Microsoft.EntityFrameworkCore.SqlServer.
- O Resultado:** O projeto rodou de forma autossuficiente e isolada. Essa mudança simples atestou o excelente poder de abstração do ORM sobre o domínio.

Nome do banco	Usuário	Servidor	Espaço
desafiodevjc	caetanopo	mysql246.umbler.com	2,00GB

Até criei o DB mas não utilizei.

```
0 referências
public void ConfigureServices(IServiceCollection services)
{
    var connectionString = Configuration.GetConnectionString("DefaultConnection");
    services.AddSwaggerGen();
    services.AddScoped<IWhoisService, WhoisService>();
    services.AddScoped<IDnsService, DnsService>();

    services.AddDbContext<DatabaseContext>(options =>
        options.UseSqlServer(connectionString)
            .LogTo(Console.WriteLine, LogLevel.Information)
            .EnableSensitiveDataLogging()
            .EnableDetailedErrors()
    );
}

services.AddControllersWithViews();
```

Injeção de Dependência

Quebrando as amarras do Controller e descentralizando as responsabilidades através de interfaces.

Extraímos as regras rígidas de comunicação externa para duas novas abstrações: **IDnsService** e **IWhoisService**.

Agora, o DomainController não precisa saber "como" buscar na internet, apenas "o que" solicitar. As instâncias são entregues limpas e prontas pelo Container de DI nativo do .NET 6 no Startup.cs.

```
using Desafio.Umbler.DTOs;
using Desafio.Umbler.Interfaces;
using Desafio.Umbler.Models;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using System;
using System.Threading.Tasks;

namespace Desafio.Umbler.Controllers
{
    [Route("api")]
    3 referências
    public class DomainController : Controller
    {
        private readonly DatabaseContext _db;
        private readonly IWhoisService _whoisService;
        private readonly IDnsService _dnsService;

        2 referências | 2/2 passing
        public DomainController(DatabaseContext db, IWhoisService whoisService, IDnsService dnsService)
        {
            _db = db;
            _whoisService = whoisService;
            _dnsService = dnsService;
        }

        [HttpGet, Route("domain/{domainName}")]
        2 referências | 2/2 passing
        public async Task<IActionResult> Get(string domainName)
        {
            if (string.IsNullOrWhiteSpace(domainName))
                return BadRequest("O nome do domínio é obrigatório.");

            var domain = await _db.Domains.FirstOrDefaultAsync(d => d.Name == domainName);

            if (domain == null || DateTime.Now.Subtract(domain.UpdatedAt).TotalMinutes > domain.Ttl)
            {
                domain ??= new Domain();

                var (ip, ttl) = await _dnsService.GetIpAndTtlAsync(domainName);
                var whoisRaw = await _whoisService.GetWhoisRawDataAsync(domainName);
                var hostedAt = await _whoisService.GetHostedAtAsync(ip);
            }
        }
    }
}
```

Proteção e Padrão DTO

- **Riscos no Legado:** O retorno original Ok(domain) enviava o modelo de dados integral para o navegador. Campos vitais de infraestrutura (Id do registro, Ttl de cache) viajavam expostos na rede.
- **Solução:** Implementação do Padrão Data Transfer Object (DTO).
- **Ganhos Imediatos:** A classe DomainResponseDTO foi criada para mapear apenas o Nome, IP, Whois e Hospedagem. O Payload de rede ficou muito mais leve e seguro, blindando a entidade central da aplicação.

```
namespace Desafio.Umbler.DTOs
{
    3 referências
    public class DomainResponseDTO
    {
        3 referências | 2/2 passing
        public string Name { get; set; }
        3 referências | 2/2 passing
        public string Ip { get; set; }
        2 referências | 1/1 passing
        public string HostedAt { get; set; }
        1 referência
        public string WhoIs { get; set; }
    }
}
```

100%

Testabilidade Restaurada

Suite de Testes Confiável

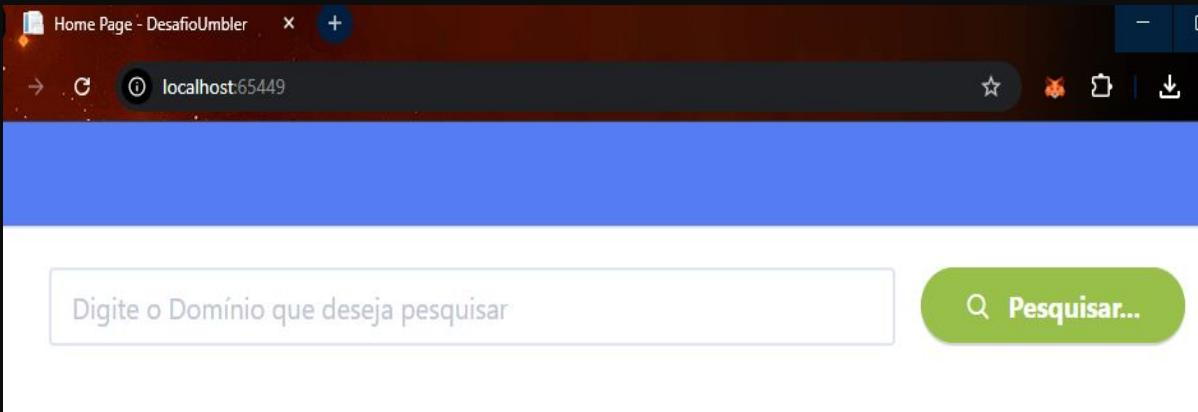
Os testes originais falhavam sem acesso à internet. Com a nova arquitetura desacoplada baseada em interfaces, reescrevemos o cenário no ControllersTests.cs.

Utilizando a biblioteca **Moq** e o Banco em Memória, simulamos cenários de sucesso e falha instantaneamente, sem gerar requisições reais ou depender de terceiros estarem no ar.

Governança com Swagger

Foi adicionado o pacote Swashbuckle.AspNetCore no Startup para gerar uma documentação automática (OpenAPI).

Isso permite que times de front-end testem os endpoints, visualizem o formato exato do DomainResponseDTO e compreendam os códigos de erro, tudo por meio de uma interface interativa.



A screenshot of the Swagger UI interface. The title bar says "Swagger UI" and the address bar shows "localhost:65449/swagger/index.html". The top navigation bar includes "Select a definition" and "Umbler Challenge API v1". The main content area shows the API documentation for "Desafio.Umbler 1.0 OAS3". A specific endpoint is highlighted: "GET /api/domain/{domainName} Parameters domainName * required string (path) Description domainName Responses Code 200 Description Success Links No links". There is also a "Try it out" button next to the parameters table.

O Antes e o Depois.

The screenshot shows a WHOIS search result for the domain 'umbler.com'. The results are presented in a structured text format. At the top, there is a search bar with the text 'umbler.com' and a green 'Pesquisar...' button. Below the search bar, the WHOIS data is displayed:

{ "name": "umbler.com", "ip": "187.84.237.146", "hostid": "RedeHost Internet Ltda.", "whois": "Domain Name: UMBLER.COM [registry Domain ID: 1805958813_DOMAIN_COM_VRSN] Registrar WHOIS Server: whois.publicdomainregistry.com/vRegistrar URL: www.publicdomainregistry.com/vUpdated Date: 2024-11-26T00:05:03Z [Creation Date: 2013-06-03T20:42:15Z] [Registration Expiration Date: 2027-06-03T20:42:15Z] Registrar: PDR Ltd. d/b/a PublicDomainRegistry.com/vRegistrar IANA ID: 303 [Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibitedInRegistry Registration ID: Not Available From Registry] Registrant Name: [REDACTED] Cardoso [Organization: REDACTED FOR PRIVACY] [Registra Street: ROD SC 401, 4150] [Registrant City: Florianopolis] [State/Province: Santa Catarina] [Registrant Postal Code: 88032 000] [Registrant County: null] [Registrant Phone: +55 51 4062 0000] [Registrant Fax: null] [Registrant Email: flavio@umbler.com.br] [Registrant Admin Email: flavio@umbler.com.br] Admin ID: Not Available From Registry [Admin Name: Flavio Eduardo Cardoso] [Admin Organization: Umbler provedor de internet LTDA] [Admin Street: ROD SC 401, 4150] [Admin City: Florianopolis] [Admin State/Province: Santa Catarina] [Admin Postal Code: 88032-000] [Admin Country: BR] [Admin Phone: +55 51 4062 0000] [Admin Fax Ext:] [Admin Fax Ext:] [Admin Email: flavio@umbler.com.br] [Tech ID: Not Available From Registry] [Tech Name: Flavio Eduardo Cardoso] [Tech Organization: Umbler provedor de internet LTDA] [Tech Street: ROD SC 401, 4150] [Tech City: Florianopolis] [Tech State/Province: Santa Catarina] [Tech Postal Code: 88032 000] [Tech Country: BR] [Tech Phone: +55 51 4062 0000] [Tech Fax Ext:] [Tech Fax Ext:] [Tech Email: flavio@umbler.com.br] [Name Server: ns1.umbler.com.br] [Name Server: ns2.umbler.com.br] [Name Server: ns3.umbler.com.br] [Name Server: ns4.umbler.com.br] [DNSSEC: Unsigned] [Registrar Abuse Contact Email: abuse.contact@publicdomainregistry.com] [Registrar Abuse Contact Phone: +55 11 3779-2188] [of the ICANN WHOIS Data Problem Reporting System: http://whois.icann.org/whois/lastupdateofWHOISdatabase:2024-02-26T12:28:32Z << click for more information on Whois status codes. please visit https://icann.org/applyforRegistrationService Provided By: RIPE NCC Python The data in this whois database is provided to you for information purposes only. That is, to assist you in obtaining information about or related to a domain name registration record. We make this information available 'as is', and do not guarantee its accuracy. By submitting a whois query, you agree that you will use this data only for lawful purposes and that, under no circumstances will you use this data to: (i) enable, high volume, automated, electronic processes that stress or load 'our' whois database system providing you this information; or (ii) allow, enable, or otherwise support the transmission of mass unsolicited, commercial advertising or solicitations via direct mail, electronic mail, or by telephone. In the compilation, reworking, dissemination or other use of this data is expressly prohibited without prior written consent from us. The Registrar of this record is PDR Ltd. d/b/a PublicDomainRegistry.com. We reserve the right to modify these terms at any time. By submitting this query, you agree to abide by these terms.' }

The screenshot shows a simplified WHOIS search result for the domain 'umbler.com'. The results are presented in a clean, modern interface. At the top, there is a search bar with the text 'umbler.com' and a green 'Pesquisar...' button. Below the search bar, the WHOIS data is displayed:

Resultados para: umbler.com

ENDERECO IP
187.84.237.146

EMPRESA DE HOSPEDAGEM
RedeHost Internet Ltda.

Dados Brutos do Whois

Domain Name: UMBLER.COM
Registry Domain ID: 1805958813_DOMAIN_COM_VRSN
Registrar WHOIS Server: whois.publicdomainregistry.com
Registrar URL: www.publicdomainregistry.com
Updated Date: 2024-11-26T00:05:03Z
Creation Date: 2013-06-03T20:42:15Z
Registrar Registration Expiration Date: 2027-06-03T20:42:15Z
Registrar: PDR Ltd. d/b/a PublicDomainRegistry.com
Registrar IANA ID: 303
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Registry Registrant ID: Not Available From Registry
Registrant Name: Flavio Eduardo Cardoso
Registrant Organization: REDACTED FOR PRIVACY
Registrant Street: ROD SC 401, 4150
Registrant City: Florianopolis
Registrant State/Province: Santa Catarina

A estrutura do Projeto.

The screenshot shows a file explorer window with the title "Solução 'Desafio.Umbler' (2 de 2 projetos)". The structure is as follows:

- Fontes externas
- src
 - Desafio.Umbler
 - Connected Services
 - Dependências
 - Properties
 - wwwroot
 - Controllers
 - DomainController.cs
 - HomeController.cs
 - DTOs
 - DomainResponseDTO.cs
 - Interfaces
 - IDnsService.cs
 - IWhoisService.cs
 - Migrations
 - 20260226031106_InitialCreate.cs
 - DatabaseContextModelSnapshot.cs
 - Models
 - Services
 - DnsService.cs
 - WhoisService.cs
 - src
 - Views
 - appsettings.json
 - bower.json
 - bundleconfig.json
 - package.json
 - Program.cs
 - Startup.cs
 - webpack.config.js
- Desafio.Umbler.Test

Pronto para Produção.

A arquitetura foi estabilizada, refatorada e o débito técnico zerado.

Muito obrigado pela oportunidade!