

1. Decisão pela Arquitetura API + Frontend:

- Decidi criar uma estrutura de API e Frontend para garantir um sistema flexível e escalável. A API será responsável por fornecer os dados necessários para o Frontend.

2. Modelagem Inicial:

- Modelei as Models de acordo com o funcionamento esperado do sistema.

3. Criação da Solution:

- Criei a Solution "ultracar_backend".

4. Projeto ASP.NET Core Web API:

- Adicionei um projeto ASP.NET Core Web API utilizando o .NET 7.0.

5. Modelagem e Estruturação:

- Desenvolvi os Models de acordo com a modelagem inicial. Alguns campos foram mantidos como strings temporariamente para evitar atrasos devido ao tempo necessário para tratar os dados corretamente.

6. Conexão com o Banco de Dados:

- Implementei a parte de conexão com o banco de dados e rodei migrations para criar minhas tabelas de acordo com as models.

7. Controllers e Funções:

- Criei as Controllers e suas funções correspondentes.

8. Testes com Swagger:

- Testei a API continuamente usando o Swagger após cada etapa de desenvolvimento.

Fase 2: Desenvolvimento do Frontend em React

9. Projeto Ultracar_Frontend em React:

Fiz a criação do projeto "ultracar_frontend" em React para o Frontend, iniciando a integração com a API.

IMPORTANTE! Interrupção devido a Urgência com Cliente:

Devido a uma urgência com um cliente do freelancer que estou fazendo, não pude completar todas as etapas do teste. Comecei o teste tarde e tive que interrompê-lo devido à emergência, impactando o andamento do projeto, fiz até a parte de salvar os dados da Ordem de Serviço, passando pela fase de identificação dos dados por qr code, listagem dos dados do cliente e formulário para iniciação do serviço.

Capturas de tela do projeto:

- Tela inicial que identifica o cliente por meio da inserção do código no campo ou por meio da leitura do qr code.



- A tela após a verificação do cliente por código encaminha para a tela que lista informações do mesmo e mostra seus veículos atrelados, decidi colocar a opção de mais de um veículo porque o cliente pode ter mais de um.

Detalhes do Cliente

Código: 1

Nome: Ruan Igor

CPF: 591.148.035-26

Email: ruan_drumond@yogoothies.com.br

Telefone: (82) 2918-1304

Endereço: Condomínio Costa Norte II, São Jorge - Maceió - AL

Veículos

Nome	Placa	Ano	Selecionar
Fiat - UNO ECONOMY Celeb. 1.4 EVO F. Flex 8V 2p	MWY-5634	2012	<button>Selecionar</button>

SalvarVoltar

Após a seleção do veículo, a tela apresenta campos para inserir informações sobre o mecânico responsável pelo serviço. Esses campos incluem uma lista suspensa (dropdown) com os mecânicos cadastrados, um espaço para descrever detalhes específicos sobre o serviço e uma tabela que lista os produtos a serem utilizados durante o processo.

Na tabela de produtos, é possível especificar a quantidade de cada item a ser utilizado. Automaticamente, são realizados cálculos com base nas quantidades inseridas, apresentando o custo total dos produtos e o valor total da ordem de serviço. Além disso, os dados de início e término do serviço, como data e hora, são registrados.

Uma vez preenchidas todas as informações necessárias, o usuário pode salvar os dados. Quanto à tela que deveria listar as ordens em andamento, não foi concluída no prazo estipulado.

Detalhes do Cliente

Código: 1

Nome: Ruan Igor

CPF: 591.148.035-26

Email: ruan_drumond@yogoothies.com.br

Telefone: (82) 2918-1304

Endereço: Condomínio Costa Norte II, São Jorge - Maceió - AL

Veículos

Nome	Placa	Ano	Selecionar
Fiat - UNO ECONOMY Celeb. 1.4 EVO F. Flex 8V 2p	MWY-5634	2012	<input type="button" value="Selecionado"/>

Mecânico:

Descrição do Serviço:

Valor Total Ordem Produção:
R\$ 2000.00

Produtos Disponíveis

Nome	Qtd Estoque	Preço Unitário	Quantidade	Valor Total	Selecionar
Filtro de óleo	20	R\$ 2000	<input type="text" value="1"/>	R\$ 2000.00	<input type="button" value="Selecionar"/>
Pastilhas de Freio	40	R\$ 6000	<input type="text" value="0"/>	R\$ 0.00	<input type="button" value="Selecionar"/>
Bateria para Carro	50	R\$ 12000	<input type="text" value="0"/>	R\$ 0.00	<input type="button" value="Selecionar"/>

Data e Hora Início Serviço: 06/12/2023, 09:27:17

Data e Hora Fim Serviço:

Ao executar o projeto frontend, é essencial ajustar a configuração de Cors no backend, no arquivo Program.cs. Isso permite definir a política de acesso à API, autorizando a URL específica onde o frontend está hospedado. Isso garante uma comunicação segura entre o frontend e o backend, sem restrições de segurança.

```
Program.cs
ultracar_backend
23
24
25
26 builder.Services.AddCors(options =>
27 {
28     options.AddPolicy("AllowSpecificOrigin", builder =>
29     {
30         builder.WithOrigins("http://localhost:3000") // Adicione aqui a origem do seu frontend
31         .AllowAnyMethod()
32         .AllowAnyHeader();
33     });
34 }
```

Após executar o script para criar o banco de dados com suas tabelas, se o seu usuário do banco não for local, será preciso configurar a string de conexão no arquivo Contexto.cs contido no backEnd.

```
0 referências
protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
{
    if (!optionsBuilder.IsConfigured)
    {
        optionsBuilder.UseNpgsql("Server=localhost;Port=5432;Database=ultracar;User Id=postgres;Password=suasenha;");
    }
    base.OnConfiguring(optionsBuilder);
}
```