1. Softwares e bibliotecas utilizadas

Para a construção do desafio foram utilizados os seguintes recursos:

* Microsoft Visual Studio Professional 2022, versão 17.7.5
* Microsoft .NET Framework, versão 4.8.09032
* ASP.NET and Web Tools, versão 17.7.273.65229
* Ferramentas C#, versão 4.7.0-3.23469.2+a3b9f0c274649870015cfe508cacde8fdc15df55 – Componentes de C# usados no IDE. Dependendo do seu tipo de projeto e configurações, uma versão diferente do compilador poderá ser usada.
* Ferramentas do Serviço de Aplicativo do Azure, versão v3.0.0 17.7.273.65229
* Ferramentas do TypeScript, versão 17.0.20829.2001
* Ferramentas do Visual Basic, versão 4.7.0-3.23469.2+a3b9f0c274649870015cfe508cacde8fdc15df55 – Componentes do Visual Basic usados no IDE. Dependendo do seu tipo de projeto e configurações, uma versão diferente do compilador poderá ser usada.
* Gerenciador de Pacotes do NuGet 6.7.0 – Gerenciador de Pacotes do NuGet no Visual Studio. Para obter mais informações sobre o NuGet, acesse https://docs.nuget.org/
* Microsoft JVM Debugger, versão 1.0
* Razor (ASP.NET Core), versão 17.7.3.2333001+0ab18affdf2a37647768d0e25f5f021bee6257a1
* SQL Server Data Tools, versão 17.7.10.1
* Visual F# Tools, versão 17.7.0-beta.23314.10+e612cf93b989503c89e3a5830090062b7ab5e143
* SQL Server Management Studi0, versão 19.1.56.0
* SQL Server Management Objects (SMO), versão 16.200.48044.0+eeb184ee48a91ebc6a27a5d192c0d67bdfaae8b6
* Microsoft T-SQL Parser, versão 17.0.8.0+3c5555b8bd579d12add8f155f1dbc871e3e734c4
* Microsoft Analysis Services Client Tools, versão 16.0.20010.0
* Microsoft Data Access Components (MDAC), versão 10.0.22621.2428

As bibliotecas que foram adicionadas ao projeto foram:

* Microsoft.EntityFrameworkCore, versão 6.0.0
* Microsoft.EntityFrameworkCore.SqlServe, versão 6.0.0
* Microsoft.EntityFrameworkCore.Tools, versão 6.0.0
* Swashbuckle.AspNetCore, versão 6.2.3
* Microsoft.NET.Test.Sdk, versão 17.5.0
* AutoFixture, versão 4.18.0
* Moq, versão 4.20.69
* Xunit, versão 2.4.2

1. Estrutura do código

A aplicação foi construída e dividida em pastas chamadas: Controllers, Data, Migrations, Models, Repositories.

Controllers -> Classe responsável por realizar as chamadas HTTP do CRUD das API’s;

Data -> Classe responsável por realizar o mapeamento e a conexão com Banco de Dados;

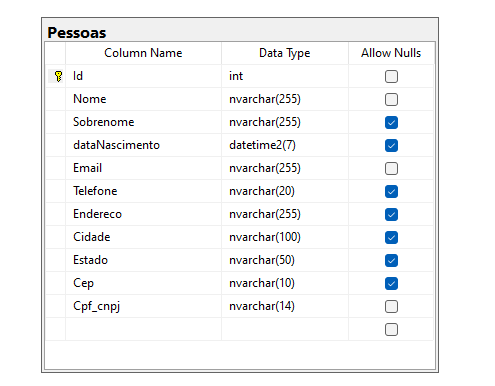
Migrations -> Classe responsável por realizar a criação do Banco de Dados e suas tabelas, com Base no mapeamento realizado na classe Data;

Models -> Classe que recebe o Modelo da Entidade, onde foi inserido todos os campos necessários para a Entidade;

Repositories -> Classe responsável por realizar os métodos do CRUD, e salvar os dados no banco de dados, quando estiverem corretos de acordo com as regras de negócio, caso não esteja, retornar mensagem de erro na API.

1. Modelo Entidade Relacionamento

Esse projeto inicialmente contempla somente uma tabela no banco de dados, segue a composição da mesma:



1. Testes da API (Conforme a regra de negócio)

Ao informar um CPF ou CNPJ inválido, a API retorna a mensagem de erro “Não foi informado um CPF ou CNPJ válido: {cpf\_cnpj}”, segue dados do teste:

**curl -X 'POST' \**

**'https://localhost:7039/api/Pessoa' \**

**-H 'accept: text/plain' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lf.homero@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "12837124671"**

**}'**

Response:

**System.Exception: Não foi informado um CPF ou CNPJ válido: 12837124671**

**at DesafioFirst.Repositories.PessoaRepository.AdicionarPessoa(PessoaModel pessoa) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Repositories\PessoaRepository.cs:line 37**

**at DesafioFirst.Controllers.PessoaController.Adicionar(PessoaModel pessoaModel) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Controllers\PessoaController.cs:line 36**

**at lambda\_method5(Closure , Object )**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ActionMethodExecutor.AwaitableObjectResultExecutor.Execute(IActionResultTypeMapper mapper, ObjectMethodExecutor executor, Object controller, Object[] arguments)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeActionMethodAsync>g\_\_Awaited|12\_0(ControllerActionInvoker invoker, ValueTask`1 actionResultValueTask)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeNextActionFilterAsync>g\_\_Awaited|10\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Rethrow(ActionExecutedContextSealed context)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Next(State& next, Scope& scope, Object& state, Boolean& isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.InvokeInnerFilterAsync()**

**--- End of stack trace from previous location ---**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeFilterPipelineAsync>g\_\_Awaited|20\_0(ResourceInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Routing.EndpointMiddleware.<Invoke>g\_\_AwaitRequestTask|6\_0(Endpoint endpoint, Task requestTask, ILogger logger)**

**at Microsoft.AspNetCore.Authorization.AuthorizationMiddleware.Invoke(HttpContext context)**

**at Swashbuckle.AspNetCore.SwaggerUI.SwaggerUIMiddleware.Invoke(HttpContext httpContext)**

**at Swashbuckle.AspNetCore.Swagger.SwaggerMiddleware.Invoke(HttpContext httpContext, ISwaggerProvider swaggerProvider)**

**at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddleware.Invoke(HttpContext context)**

**HEADERS**

**=======**

**Accept: text/plain**

**Host: localhost:7039**

**User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36**

**:method: POST**

**Accept-Encoding: gzip, deflate, br**

**Accept-Language: en-US,en;q=0.9,pt-BR;q=0.8,pt;q=0.7,es;q=0.6**

**Content-Type: application/json**

**Origin: https://localhost:7039**

**Referer: https://localhost:7039/swagger/index.html**

**Content-Length: 300**

**sec-ch-ua: "Chromium";v="118", "Google Chrome";v="118", "Not=A?Brand";v="99"**

**sec-ch-ua-mobile: ?0**

**sec-ch-ua-platform: "Windows"**

**sec-fetch-site: same-origin**

**sec-fetch-mode: cors**

**sec-fetch-dest: empty**

Não é possível cadastrar informando o id, pelo fato de o id ser autoincremento, segue o response do teste:

**curl -X 'POST' \**

**'https://localhost:7039/api/Pessoa' \**

**-H 'accept: text/plain' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"id": 15,**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lf.homero@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "05266413123"**

**}'**

Response:

**Microsoft.EntityFrameworkCore.DbUpdateException: An error occurred while saving the entity changes. See the inner exception for details.**

**---> Microsoft.Data.SqlClient.SqlException (0x80131904): Não é possível inserir um valor explícito para a coluna de identidade na tabela 'Pessoas' quando IDENTITY\_INSERT está definido como OFF.**

**at Microsoft.Data.SqlClient.SqlCommand.<>c.<ExecuteDbDataReaderAsync>b\_\_188\_0(Task`1 result)**

**at System.Threading.Tasks.ContinuationResultTaskFromResultTask`2.InnerInvoke()**

**at System.Threading.Tasks.Task.<>c.<.cctor>b\_\_272\_0(Object obj)**

**at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext, ContextCallback callback, Object state)**

**--- End of stack trace from previous location ---**

**at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext, ContextCallback callback, Object state)**

**at System.Threading.Tasks.Task.ExecuteWithThreadLocal(Task& currentTaskSlot, Thread threadPoolThread)**

**--- End of stack trace from previous location ---**

**at Microsoft.EntityFrameworkCore.Storage.RelationalCommand.ExecuteReaderAsync(RelationalCommandParameterObject parameterObject, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Storage.RelationalCommand.ExecuteReaderAsync(RelationalCommandParameterObject parameterObject, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.ReaderModificationCommandBatch.ExecuteAsync(IRelationalConnection connection, CancellationToken cancellationToken)**

**ClientConnectionId:0b6ee35b-3301-42b1-b066-90b04505e0fc**

**Error Number:544,State:1,Class:16**

**--- End of inner exception stack trace ---**

**at Microsoft.EntityFrameworkCore.Update.ReaderModificationCommandBatch.ExecuteAsync(IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable`1 commandBatches, IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable`1 commandBatches, IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable`1 commandBatches, IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.ChangeTracking.Internal.StateManager.SaveChangesAsync(IList`1 entriesToSave, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.ChangeTracking.Internal.StateManager.SaveChangesAsync(StateManager stateManager, Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.SqlServer.Storage.Internal.SqlServerExecutionStrategy.ExecuteAsync[TState,TResult](TState state, Func`4 operation, Func`4 verifySucceeded, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.DbContext.SaveChangesAsync(Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.DbContext.SaveChangesAsync(Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)**

**at DesafioFirst.Repositories.PessoaRepository.AdicionarPessoa(PessoaModel pessoa) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Repositories\PessoaRepository.cs:line 29**

**at DesafioFirst.Controllers.PessoaController.Adicionar(PessoaModel pessoaModel) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Controllers\PessoaController.cs:line 36**

**at lambda\_method5(Closure , Object )**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ActionMethodExecutor.AwaitableObjectResultExecutor.Execute(IActionResultTypeMapper mapper, ObjectMethodExecutor executor, Object controller, Object[] arguments)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeActionMethodAsync>g\_\_Awaited|12\_0(ControllerActionInvoker invoker, ValueTask`1 actionResultValueTask)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeNextActionFilterAsync>g\_\_Awaited|10\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Rethrow(ActionExecutedContextSealed context)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Next(State& next, Scope& scope, Object& state, Boolean& isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeInnerFilterAsync>g\_\_Awaited|13\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeFilterPipelineAsync>g\_\_Awaited|20\_0(ResourceInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Routing.EndpointMiddleware.<Invoke>g\_\_AwaitRequestTask|6\_0(Endpoint endpoint, Task requestTask, ILogger logger)**

**at Microsoft.AspNetCore.Authorization.AuthorizationMiddleware.Invoke(HttpContext context)**

**at Swashbuckle.AspNetCore.SwaggerUI.SwaggerUIMiddleware.Invoke(HttpContext httpContext)**

**at Swashbuckle.AspNetCore.Swagger.SwaggerMiddleware.Invoke(HttpContext httpContext, ISwaggerProvider swaggerProvider)**

**at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddleware.Invoke(HttpContext context)**

**HEADERS**

**=======**

**Accept: text/plain**

**Host: localhost:7039**

**User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36**

**:method: POST**

**Accept-Encoding: gzip, deflate, br**

**Accept-Language: en-US,en;q=0.9,pt-BR;q=0.8,pt;q=0.7,es;q=0.6**

**Content-Type: application/json**

**Origin: https://localhost:7039**

**Referer: https://localhost:7039/swagger/index.html**

**Content-Length: 312**

**sec-ch-ua: "Chromium";v="118", "Google Chrome";v="118", "Not=A?Brand";v="99"**

**sec-ch-ua-mobile: ?0**

**sec-ch-ua-platform: "Windows"**

**sec-fetch-site: same-origin**

**sec-fetch-mode: cors**

**sec-fetch-dest: empty**

Ao tentar cadastrar uma pessoa com o mesmo email, é retornado o erro de que o email está duplicado, segue dados do teste:

**curl -X 'POST' \**

**'https://localhost:7039/api/Pessoa' \**

**-H 'accept: text/plain' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lf.homero@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "05266413123"**

**}'**

Response:

**Microsoft.EntityFrameworkCore.DbUpdateException: An error occurred while saving the entity changes. See the inner exception for details.**

**---> Microsoft.Data.SqlClient.SqlException (0x80131904): Não é possível inserir uma linha de chave duplicada no objeto 'dbo.Pessoas' com índice exclusivo 'IX\_Pessoas\_Email'. O valor de chave duplicada é (lf.homero@gmail.com).**

**A instrução foi finalizada.**

**at Microsoft.Data.SqlClient.SqlCommand.<>c.<ExecuteDbDataReaderAsync>b\_\_188\_0(Task`1 result)**

**at System.Threading.Tasks.ContinuationResultTaskFromResultTask`2.InnerInvoke()**

**at System.Threading.Tasks.Task.<>c.<.cctor>b\_\_272\_0(Object obj)**

**at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext, ContextCallback callback, Object state)**

**--- End of stack trace from previous location ---**

**at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext, ContextCallback callback, Object state)**

**at System.Threading.Tasks.Task.ExecuteWithThreadLocal(Task& currentTaskSlot, Thread threadPoolThread)**

**--- End of stack trace from previous location ---**

**at Microsoft.EntityFrameworkCore.Storage.RelationalCommand.ExecuteReaderAsync(RelationalCommandParameterObject parameterObject, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Storage.RelationalCommand.ExecuteReaderAsync(RelationalCommandParameterObject parameterObject, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.ReaderModificationCommandBatch.ExecuteAsync(IRelationalConnection connection, CancellationToken cancellationToken)**

**ClientConnectionId:0b6ee35b-3301-42b1-b066-90b04505e0fc**

**Error Number:2601,State:1,Class:14**

**--- End of inner exception stack trace ---**

**at Microsoft.EntityFrameworkCore.Update.ReaderModificationCommandBatch.ExecuteAsync(IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable`1 commandBatches, IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable`1 commandBatches, IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable`1 commandBatches, IRelationalConnection connection, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.ChangeTracking.Internal.StateManager.SaveChangesAsync(IList`1 entriesToSave, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.ChangeTracking.Internal.StateManager.SaveChangesAsync(StateManager stateManager, Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.SqlServer.Storage.Internal.SqlServerExecutionStrategy.ExecuteAsync[TState,TResult](TState state, Func`4 operation, Func`4 verifySucceeded, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.DbContext.SaveChangesAsync(Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)**

**at Microsoft.EntityFrameworkCore.DbContext.SaveChangesAsync(Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)**

**at DesafioFirst.Repositories.PessoaRepository.AdicionarPessoa(PessoaModel pessoa) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Repositories\PessoaRepository.cs:line 29**

**at DesafioFirst.Controllers.PessoaController.Adicionar(PessoaModel pessoaModel) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Controllers\PessoaController.cs:line 36**

**at lambda\_method5(Closure , Object )**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ActionMethodExecutor.AwaitableObjectResultExecutor.Execute(IActionResultTypeMapper mapper, ObjectMethodExecutor executor, Object controller, Object[] arguments)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeActionMethodAsync>g\_\_Awaited|12\_0(ControllerActionInvoker invoker, ValueTask`1 actionResultValueTask)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeNextActionFilterAsync>g\_\_Awaited|10\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Rethrow(ActionExecutedContextSealed context)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Next(State& next, Scope& scope, Object& state, Boolean& isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeInnerFilterAsync>g\_\_Awaited|13\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeFilterPipelineAsync>g\_\_Awaited|20\_0(ResourceInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Routing.EndpointMiddleware.<Invoke>g\_\_AwaitRequestTask|6\_0(Endpoint endpoint, Task requestTask, ILogger logger)**

**at Microsoft.AspNetCore.Authorization.AuthorizationMiddleware.Invoke(HttpContext context)**

**at Swashbuckle.AspNetCore.SwaggerUI.SwaggerUIMiddleware.Invoke(HttpContext httpContext)**

**at Swashbuckle.AspNetCore.Swagger.SwaggerMiddleware.Invoke(HttpContext httpContext, ISwaggerProvider swaggerProvider)**

**at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddleware.Invoke(HttpContext context)**

**HEADERS**

**=======**

**Accept: text/plain**

**Host: localhost:7039**

**User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36**

**:method: POST**

**Accept-Encoding: gzip, deflate, br**

**Accept-Language: en-US,en;q=0.9,pt-BR;q=0.8,pt;q=0.7,es;q=0.6**

**Content-Type: application/json**

**Origin: https://localhost:7039**

**Referer: https://localhost:7039/swagger/index.html**

**Content-Length: 300**

**sec-ch-ua: "Chromium";v="118", "Google Chrome";v="118", "Not=A?Brand";v="99"**

**sec-ch-ua-mobile: ?0**

**sec-ch-ua-platform: "Windows"**

**sec-fetch-site: same-origin**

**sec-fetch-mode: cors**

**sec-fetch-dest: empty**

Ao passar por todas as validações citadas a cima, é possível cadastrar ou alterar os dados referente a pessoa, segue:

**curl -X 'POST' \**

**'https://localhost:7039/api/Pessoa' \**

**-H 'accept: text/plain' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lfdwhomero@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "05266413123"**

**}'**

Response:

**{**

**"id": 12,**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lfdwhomero@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "05266413123"**

**}**

**curl -X 'PUT' \**

**'https://localhost:7039/api/Pessoa/12' \**

**-H 'accept: text/plain' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lfdwhmr@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "30335476000144"**

**}'**

**{**

**"id": 12,**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204Z",**

**"email": "lfdwhmr@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "30335476000144"**

**}**

Os métodos que devem informar o id, caso não exista no banco de dados, é retornada mensagem de erro, segue o teste:

**curl -X 'GET' \**

**'https://localhost:7039/api/Pessoa/20' \**

**-H 'accept: text/plain'**

Response:

**System.Exception: Pessoa para o ID: 20 não foi encontrada no banco de dados.**

**at DesafioFirst.Controllers.PessoaController.BuscarPessoaPorId(Int32 id) in C:\Users\lf\_ho\Downloads\desafioFirst\DesafioFirst\Controllers\PessoaController.cs:line 32**

**at lambda\_method5(Closure , Object )**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ActionMethodExecutor.AwaitableObjectResultExecutor.Execute(IActionResultTypeMapper mapper, ObjectMethodExecutor executor, Object controller, Object[] arguments)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeActionMethodAsync>g\_\_Awaited|12\_0(ControllerActionInvoker invoker, ValueTask`1 actionResultValueTask)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeNextActionFilterAsync>g\_\_Awaited|10\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Rethrow(ActionExecutedContextSealed context)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.Next(State& next, Scope& scope, Object& state, Boolean& isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker.<InvokeInnerFilterAsync>g\_\_Awaited|13\_0(ControllerActionInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeFilterPipelineAsync>g\_\_Awaited|20\_0(ResourceInvoker invoker, Task lastTask, State next, Scope scope, Object state, Boolean isCompleted)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Mvc.Infrastructure.ResourceInvoker.<InvokeAsync>g\_\_Awaited|17\_0(ResourceInvoker invoker, Task task, IDisposable scope)**

**at Microsoft.AspNetCore.Routing.EndpointMiddleware.<Invoke>g\_\_AwaitRequestTask|6\_0(Endpoint endpoint, Task requestTask, ILogger logger)**

**at Microsoft.AspNetCore.Authorization.AuthorizationMiddleware.Invoke(HttpContext context)**

**at Swashbuckle.AspNetCore.SwaggerUI.SwaggerUIMiddleware.Invoke(HttpContext httpContext)**

**at Swashbuckle.AspNetCore.Swagger.SwaggerMiddleware.Invoke(HttpContext httpContext, ISwaggerProvider swaggerProvider)**

**at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddleware.Invoke(HttpContext context)**

**HEADERS**

**=======**

**Accept: text/plain**

**Host: localhost:7039**

**User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36**

**:method: GET**

**Accept-Encoding: gzip, deflate, br**

**Accept-Language: en-US,en;q=0.9,pt-BR;q=0.8,pt;q=0.7,es;q=0.6**

**Referer: https://localhost:7039/swagger/index.html**

**sec-ch-ua: "Chromium";v="118", "Google Chrome";v="118", "Not=A?Brand";v="99"**

**sec-ch-ua-mobile: ?0**

**sec-ch-ua-platform: "Windows"**

**sec-fetch-site: same-origin**

**sec-fetch-mode: cors**

**sec-fetch-dest: empty**

Quando o id existe, é retornado o objeto, segue o teste:

**curl -X 'GET' \**

**'https://localhost:7039/api/Pessoa/12' \**

**-H 'accept: text/plain'**

**{**

**"id": 12,**

**"nome": "Marcos Antonio",**

**"sobrenome": "Homero",**

**"dataNascimento": "1994-01-06T13:35:47.204",**

**"email": "lfdwhmr@gmail.com",**

**"telefone": "61983260602",**

**"endereco": "qn 27 conjunto 3 casa 7",**

**"cidade": "Brasília",**

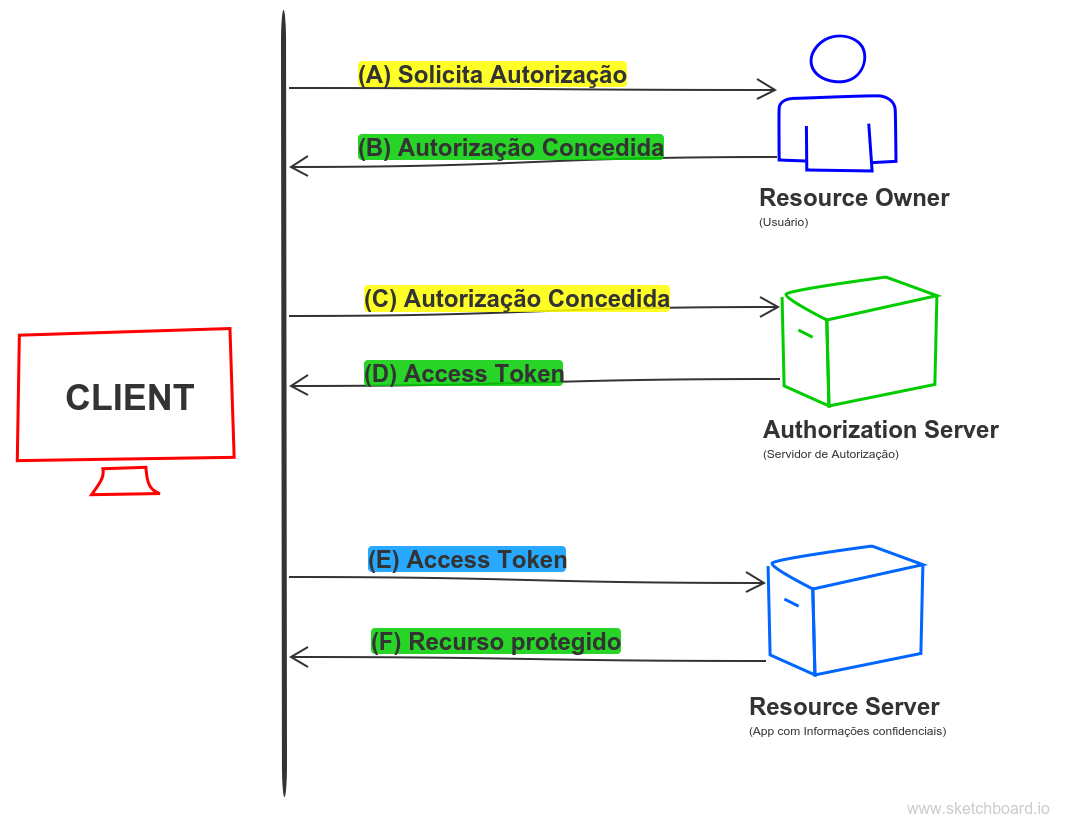
**"estado": "DF",**

**"cep": "71880647",**

**"cpf\_cnpj": "30335476000144"**

**}**

1. Autenticação

Para a autenticação o protocolo recomendado seria oAuth2 por conta da segurança que tal protocolo de autenticação oferece, funcionária da seguinte forma:

(A) O Usuário acessa um client. Para ter acesso ao conteúdo protegido da api (Resource Server) o client solicita autorização, um login com as credenciais do usuário previamente já cadastrado.

(B) A Autorização é Concedida pelo usuário, realizando o login.

(C) O client solicita um token de acesso ao Authorization Server através da autenticação de sua própria identidade.

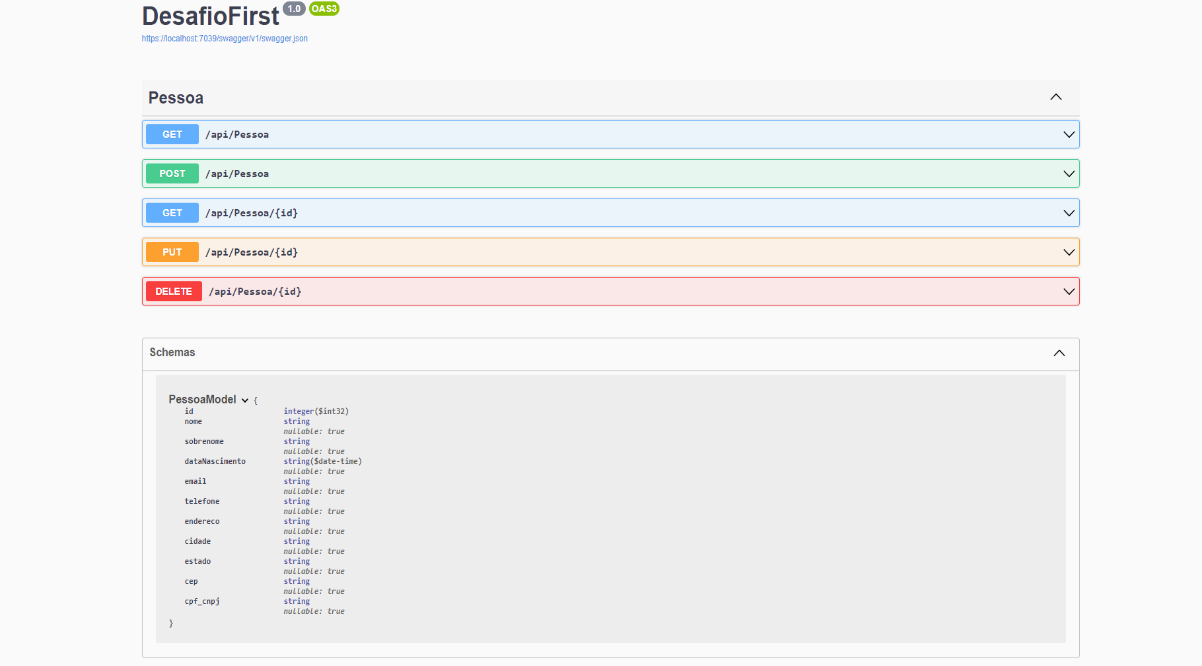
(D) O Usuário (Resource Owner) confirma sua identidade através do seu usuário e senha. Se tudo ocorrer bem um Access Token será criado e devolvido para o client gerenciar.

(E) Por fim o client informa o Access Token ao Resource Server.

(F) O Resource Server faz validação e retorna o Conteúdo Protegido.

1. Documentação da API (Swagger)

Além da documentação por escrita, é importante abastecer o Swagger com o máximo de informações possível para que o desenvolvedor consiga consumir a API da melhor maneira possível.

1. Segurança
   1. Como você implementaria medidas de segurança, como proteção contra-ataques de injeção de SQL ou XSS;

Primeiro passo é a restrição de permissão ao banco de dados, instancia e sistema operacional, para restringir o impacto caso aconteça o ataque.

Posteriormente tentar evitar esses tipos de ataque colocando tratamento a nível FrontEnd no HTML, em seguida intervindo a nível BackEnd, tratando as consultas dinâmicas realizadas no banco de dados (Uma alternativa é o uso do sp\_executesql) e o uso de “stored procedures” para validação dos dados a nível banco de dados.

* 1. Como você implementaria um tratamento de erro apropriado para lidar com falhas na comunicação com o banco de dados.

Faria um tratamento com Try() e Catch() de acordo com o código de erro retornado entre o servidor de banco de dados e a instancia, existe uma tabela com os erros retornados que é possível levar ao usuário uma mensagem de forma mais amigável, presente na seguinte documentação: https://learn.microsoft.com/pt-br/azure/azure-sql/database/troubleshoot-common-errors-issues?view=azuresql