#NOME DO PROJETO - DESAFIOCPF

Esta API destina-se ao cadastro de CPFs, proporcionando a capacidade de cadastrar múltiplos CPFs, realizar pesquisas específicas ou abrangentes, e excluir registros específicos. O desenvolvimento da API foi realizado no ambiente .NET 6, seguindo o padrão arquitetural Clean Architecture. Adotando princípios sólidos (SOLID) e boas práticas, como Clean Code, o produto final será uma API robusta, com testes unitários e pronta para ser containerizada para fins de deploy.

## INSTALAÇÃO

Para que essa API funcione localmente será necessária à instalação de alguns aplicativos (Lembrando que o mesmo foi desenvolvido todo no Sistema Operacional Windows 10 e para instalação ira seguir o padrão do mesmo) dentre eles são:

1 -> Visual Studio 2022

1.1 -> Baixar: Acesse o site oficial do Visual Studio 2022 [https://visualstudio.microsoft.com/pt-br/downloads/] e baixe a versão mais recente do Visual Studio 2022.

1.2 -> Instalar:

Execute o instalador baixado.

Selecione "Desenvolvimento para área de trabalho com .NET" durante a instalação.

Siga as instruções na tela.

### INSTALAÇÃO – DEPENDENCIAS

## - Modo de instalação indo em dependências do projeto com o botão direito e manage nuget packager essas mesmas estão detalhadas em uma imagem no raiz do projeto com suas respectivas versões.

2 -> PostgreSQL 12.0

2.1 -> Baixar: Acesse a página de download do PostgreSQL e selecione a versão 12.0. [https://www.enterprisedb.com/downloads/postgres-postgresql-downloads] não esquescendo de instalar na porta 5433

Ao escolher o sistema operacional lembrando que o padrão a ser utilizado inicialmente nessa API foi o Windows 10.

2.2 -> Instalar:

Execute o instalador baixado.

Siga as instruções do assistente de instalação.

Durante a instalação, lembre-se de fornecer a senha para o usuário postgres.

3 -> PgAdmin4

3.1 -> Baixar: Acesse a página de download do PgAdmin4 e baixe a versão mais recente. [https://www.pgadmin.org/download/pgadmin-4-windows/]

Selecione a versão adequada para o seu sistema operacional.

3.2 -> Instalar:

Execute o instalador baixado.

Siga as instruções do assistente de instalação.

4 -> Docker

4.1 -> Baixar: Acesse o site oficial do Docker e baixe a versão mais recente do Docker Desktop. [https://www.docker.com/get-started/]

Certifique-se de que a virtualização esteja ativada em seu sistema.

4.2 -> Instalar:

Execute o instalador baixado.

Siga as instruções do assistente de instalação.

Ao finalizar a instalação do Docker abra o visual Studio 2022 na pasta do projeto va ate a opção tools – comand line – PowerShell e execute esses comando para fim de configuração do docker e funcionamento da aplicação pelo mesmo.

criar uma rede para que os containers possam se comunicar atraves do comando

docker network create --driver bridge my-network

execução do comando para subir o postgres na minha network

docker run --name my-postgres --network=my-network -p 5433:5432 -e POSTGRES\_PASSWORD=123456 -d postgres

comando para subir o pgadmin4 nessa mesma rede

docker run --name my-pgadmin --network=my-network -p 15432:80 -e PGADMIN\_DEFAULT\_EMAIL=daniel@gmail.com -e PGADMIN\_DEFAULT\_PASSWORD=123456 -d dpage/pgadmin4

comando para buildar a aplicação executar na pasta da aplicação

docker build -t my-desafiocpf -f Cpf/Dockerfile .

comando para subir a aplicação

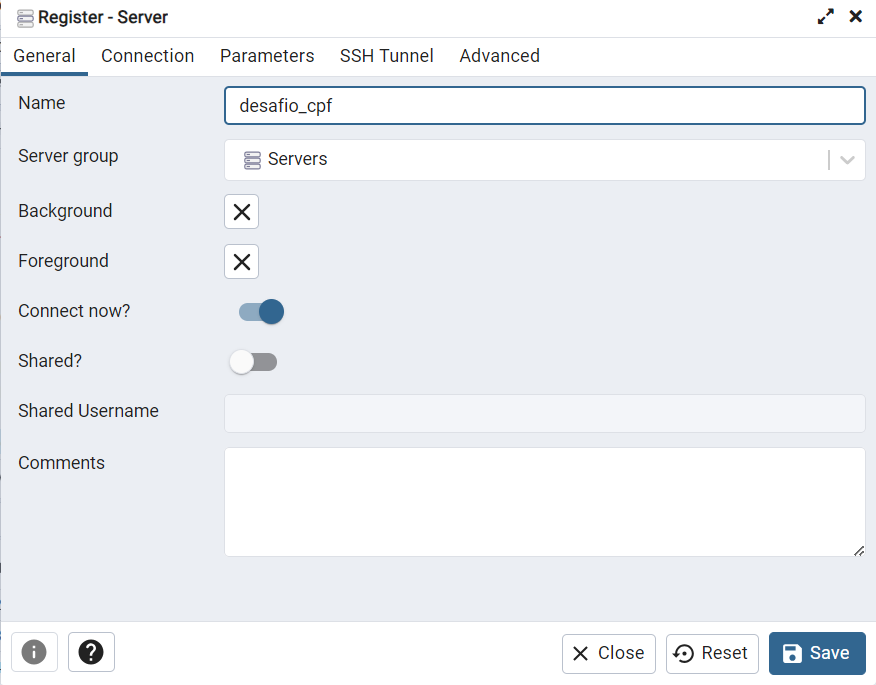
docker run --name my-desafiocpf --network=my-network -d -p 5000:80 my-desafiocpf

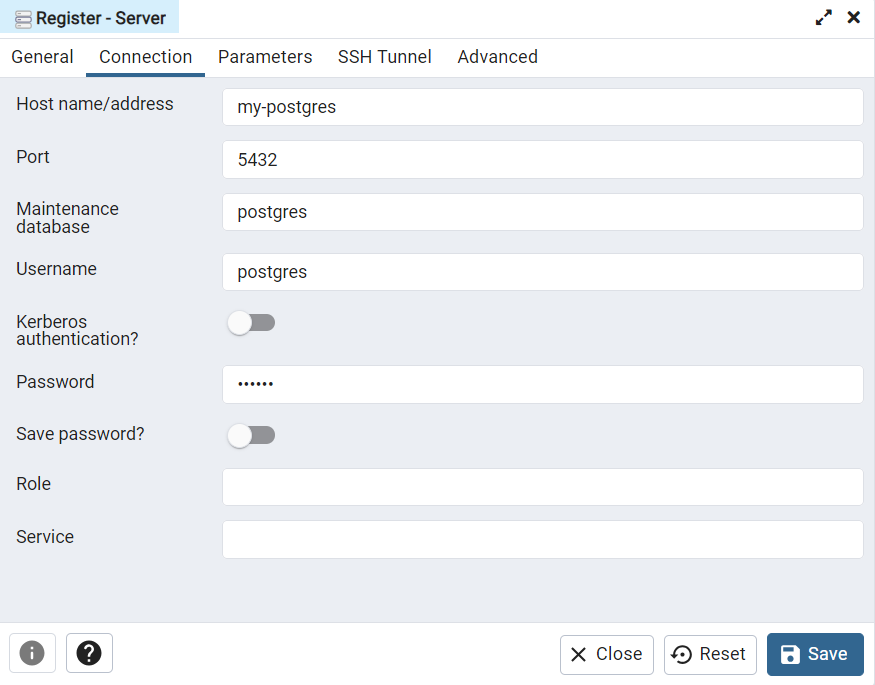
Ao finalizar todos esses comandos e configurações o primeiro passo antes de executar a aplicação você vai ate o seu browser de maior preferencia e digite a seguinte url [http://localhost:15432] ao pedir login e senha você vai digitar os seguintes

Login: [daniel@gmail.com](mailto:daniel@gmail.com)

Senha: 123456

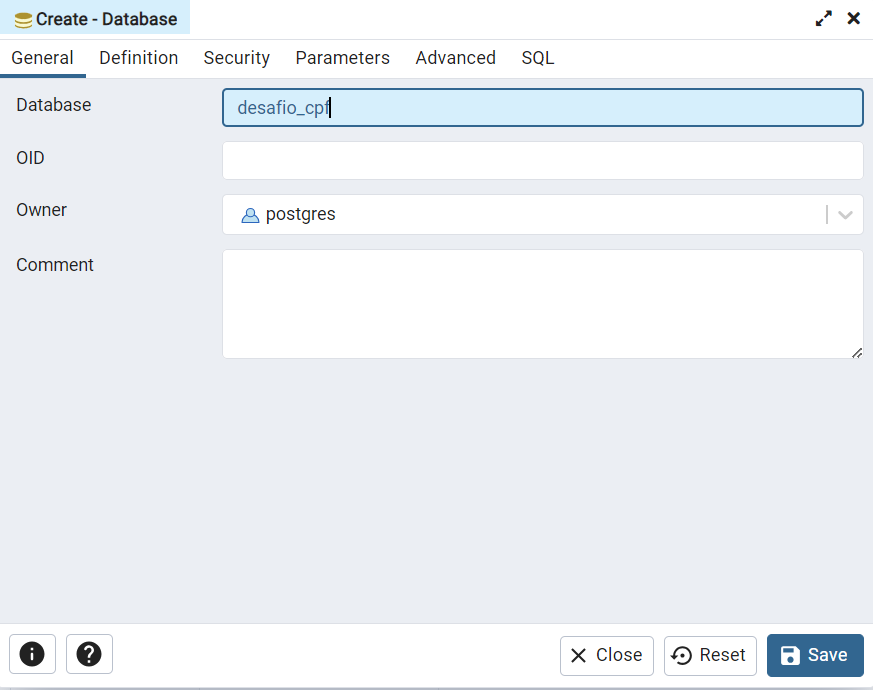
Ao iniciar no server você vai clicar com o botão direito e depois na opção server vai seguir as seguintes configurações.





Com o Password: 123456

Após salvar essa parte você vai fazer o seguinte indo na opção Databases vai clicar com o botão direito e criar a database desafio\_cpf



Apenas dessa maneira e salvar

Após isso na database criada você vai com o botão direito do mouse e a opção Query Tool

E vai executar esses comandos

-- SEQUENCE: public.controle\_\_id\_seq

-- DROP SEQUENCE IF EXISTS public.controle\_\_id\_seq;

CREATE SEQUENCE IF NOT EXISTS public.controle\_\_id\_seq

INCREMENT 1

START 1

MINVALUE 1

MAXVALUE 4294967296

CACHE 1;

ALTER SEQUENCE public.controle\_\_id\_seq

OWNER TO postgres;

-- Table: public.controle

-- DROP TABLE IF EXISTS public.controle;

CREATE TABLE IF NOT EXISTS public.controle

(

\_id integer DEFAULT nextval('controle\_\_id\_seq'::regclass),

cpf\_numero character varying(11) COLLATE pg\_catalog."default" NOT NULL,

created\_at timestamp without time zone NOT NULL,

CONSTRAINT controle\_cpf\_numero\_key UNIQUE (cpf\_numero)

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public.controle

OWNER to postgres;

Após executar esses comando e finalizar essa parte sim você pode executar a api em uma nova guia do browser com o link [http://localhost:5000/swagger/index.html]

E pode começar a brincar e se divertir com ela KKK.

## Uso

Essa aplicação demonstra sua utilidade ao oferecer funcionalidades essenciais, como o cadastro e validação de CPFs. Além disso, proporciona a flexibilidade de excluir registros do banco de dados quando necessário, bem como a capacidade de realizar pesquisas específicas ou abranger todos os CPFs cadastrados.

## Motivação

A principal motivação para a concepção desta API, empregando uma gama de tecnologias, reside na minha profunda experiência ao longo do tempo com essas plataformas, bem como na minha facilidade intrínseca de desenvolvimento por meio delas. Embora tenha adquirido conhecimentos adicionais para enfrentar os desafios específicos desta empreitada, a verdadeira motivação provém da oportunidade singular de contribuir para uma empresa de notável renome. O privilégio de participar deste projeto é um estímulo significativo e reforça meu compromisso com o desenvolvimento de soluções robustas e eficazes.

### OBS

Caso precise de alguma variável de ambiente a do docker por exemplo você pode inserir ela dessa maneira:

Para adicionar variáveis de ambiente no Windows, incluindo aquelas relacionadas ao Docker, você pode seguir essas etapas:

1. \*\*Pressione as teclas Win + S para abrir a barra de pesquisa. Digite "Variáveis de Ambiente" e selecione "Editar as variáveis de ambiente do sistema".\*\*

2. \*\*Na janela Propriedades do Sistema, clique no botão "Variáveis de Ambiente" na parte inferior.\*\*

3. \*\*Na seção "Variáveis do Sistema", clique em "Novo" para adicionar uma nova variável.\*\*

4. \*\*Para adicionar uma variável relacionada ao Docker, você precisará fornecer o nome e o valor. Por exemplo:\*\*

- \*\*Nome:\*\* `DOCKER\_HOST`

- \*\*Valor:\*\* `tcp://localhost:2375`

(O valor pode variar dependendo da configuração do seu ambiente Docker.)

5. \*\*Clique em "OK" para fechar cada janela e salvar as alterações.\*\*

Lembre-se de que as alterações nas variáveis de ambiente geralmente só afetarão os novos processos. Se você já tiver um prompt de comando aberto, será necessário fechá-lo e abrir um novo para que ele veja as novas variáveis de ambiente.

Se você estiver usando Docker Toolbox, o processo pode ser um pouco diferente. Nesse caso, você pode criar um arquivo chamado `.bashrc` no seu diretório de usuário (por exemplo, `C:\Users\SeuNomeDeUsuário`) e adicionar as configurações do Docker lá. Isso garantirá que as configurações sejam aplicadas sempre que você abrir um novo terminal.