**PROJETO IFSTAR – REQUISITOS FUNCIONAMENTO**

* **Aplicativos Instalados:**
  + **Visual Studio 2019 e/ou Visual Studio Code**: Para configuração da conexão com o Banco de Dados e Testes.
  + **Banco de Dados SQL Server e Microsoft SQL Server Management Studio:** Instalar versões mais recentes. Será o Banco de Dados utilizado para armazenar as informações.
  + **Postman:** Para realizar testes das funcionalidades, através de requisições HTTP.
* **Configurações:**
  + **Banco de Dados:** Após ser instalado o Banco de Dados SQL Server na máquina, deve ser rodado os seguintes scripts:

CREATE TABLE tbUsuario

(

idUsuario INT PRIMARY KEY IDENTITY(1,1),

nome VARCHAR(60) NOT NULL,

cpf VARCHAR(11) NULL,

email VARCHAR(60) NOT NULL UNIQUE,

senha VARCHAR(100) NOT NULL,

cdPrivilegio INT NOT NULL DEFAULT(1)

)

CREATE TABLE tbVotacao

(

idVotacao INT PRIMARY KEY IDENTITY(1,1),

tema VARCHAR(40) NOT NULL,

edicao INT NOT NULL,

ano INT NOT NULL,

dtVotacao DATE NOT NULL,

horaInicio VARCHAR(5) NOT NULL,

horaFim VARCHAR(5) NOT NULL,

flAberto BIT NOT NULL DEFAULT(0),

flVotacaoEncerrada BIT NOT NULL DEFAULT(0)

)

CREATE TABLE tbParticipante

(

idParticipante VARCHAR(8) PRIMARY KEY,

nome VARCHAR(60) NOT NULL,

musica VARCHAR(50) NOT NULL,

instEnsino VARCHAR(40) NOT NULL,

telefone VARCHAR(14) NULL

)

CREATE TABLE tbPublicoVotacao

(

idUsuario INT NOT NULL,

idParticipante VARCHAR(8) NOT NULL,

dtVoto DATETIME NOT NULL,

nota INT NULL

)

ALTER TABLE tbPublicoVotacao

ADD CONSTRAINT FK\_IdUsuario FOREIGN KEY (idUsuario) REFERENCES tbUsuario (idUsuario)

ALTER TABLE tbPublicoVotacao

ADD CONSTRAINT FK\_IdParticipante FOREIGN KEY (idParticipante) REFERENCES tbParticipante (idParticipante)

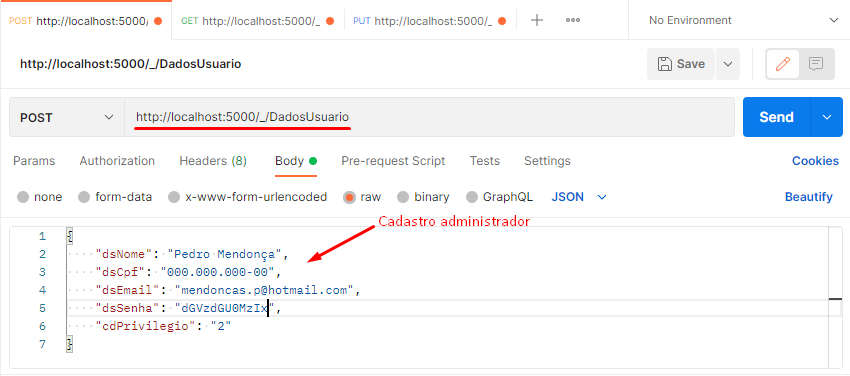
ALTER TABLE tbPublicoVotacao

ADD CONSTRAINT PK\_ControleVotacao

PRIMARY KEY (idUsuario, idParticipante)

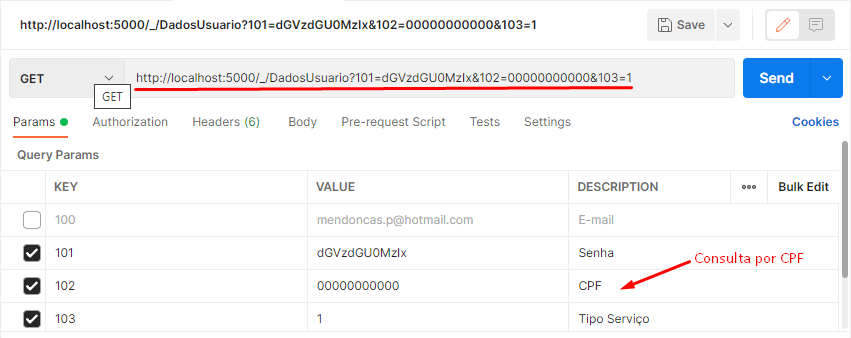
* **Modelo Requisições:**
  + As funcionalidades de cada categoria (Usuário, Participante, Votação) será distribuída nos métodos do Gateway de acordo com método HTTP requisitado.
  + **OBS**: Para as funcionalidades Cadastrar Usuário, Consultar Usuário e Alterar Usuário, deve ser enviado sempre a senha já codificada em Base64. A aplicação fará a descodificação e processará os dados.
* Cadastrar Usuário (AddUsuario) – um método Post que terá informações diferentes no body de acordo com o usuário cadastrado. Essa requisição não receberá retorno com dados, apenas o retorno HTTP com código 201 (Created):

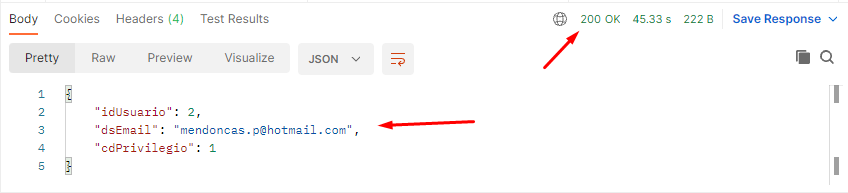




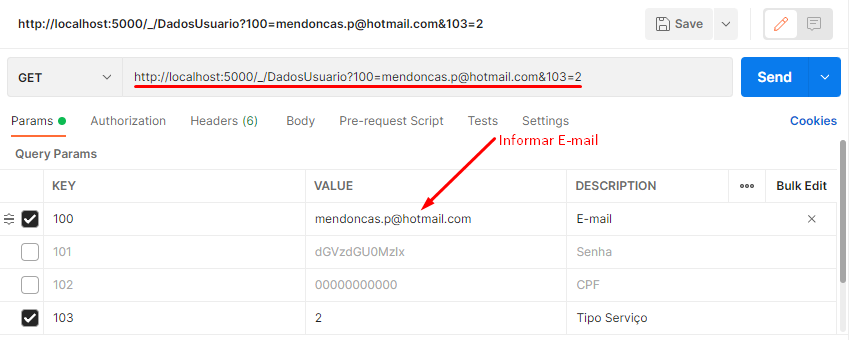
* Consultar Usuário (GetUsuario) – um método Get que será utilizado ao informar o Tipo de Serviço 1. A consulta pode ser realizada através do e-mail ou CPF, de acordo com o tipo de usuário. O campo que representa cada número também se encontra na API. Essa requisição receberá um retorno com alguns dados do usuário, que serão importantes para determinar quais acessos o usuário terá, como também para registrar seu voto e outras ações no sistema:

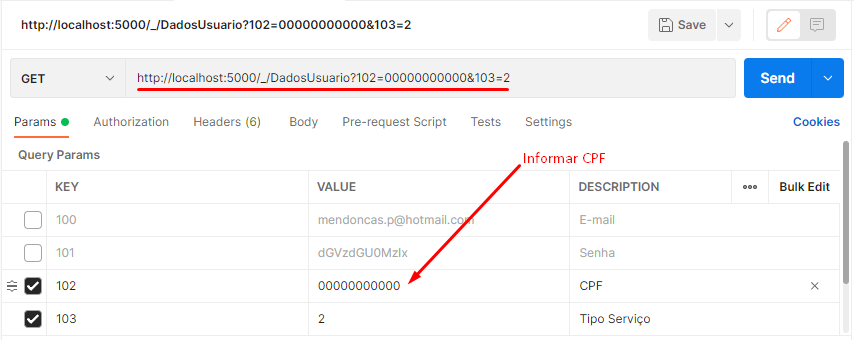


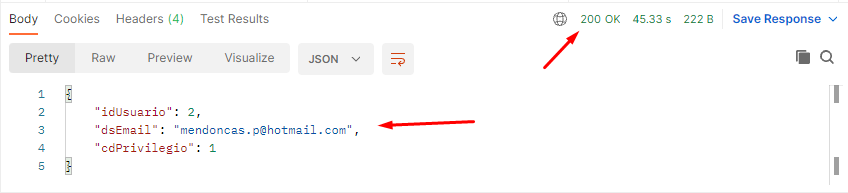




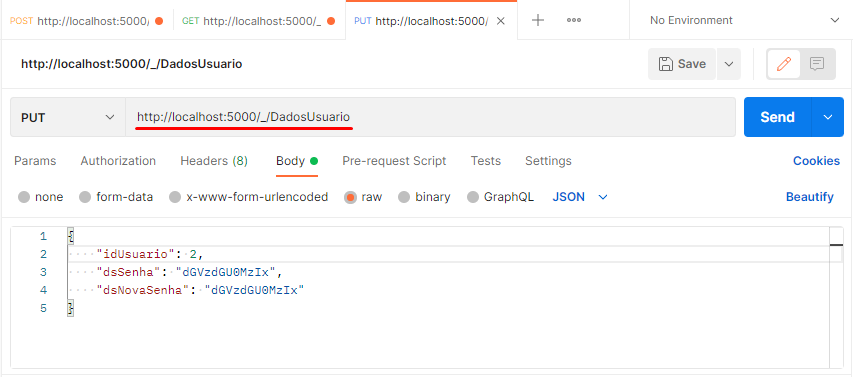
* Recuperar Senha (RecuperarSenha) - um método Get que será utilizado ao informar o Tipo de Serviço 2. Deve ser informado o e-mail ou CPF do usuário, para que seu cadastro seja encontrado no sistema e o e-mail com a nova senha seja enviado. Essa requisição receberá um retorno com alguns dados do usuário, que não serão tão importantes porque o usuário ainda não fez Login no sistema. O objetivo é caso o front-end deseje exibir alguma informação sobre o usuário:



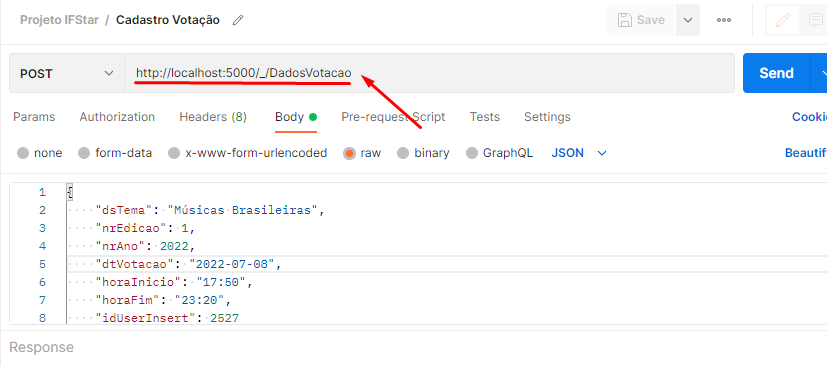




* Alterar Usuário (AlterUsuario) – um método PUT que terá informações sobre o idUsuário, senha atual e nova senha. Esse método será usado apenas para alteração de senha. Essa requisição não receberá retorno com dados, apenas o retorno HTTP com código 200 (OK):



* Cadastrar Votação (AddVotacao) - um método Post que terá informações da votação. Essa requisição não receberá retorno com dados, apenas o retorno HTTP com código 201 (Created):



* Alterar Votação (AlterVotacao) - um método PUT que terá informações sobre o idVotacao e poderá alterar o tema, data da votação, hora de início e hora de fim. Essa requisição não receberá retorno com dados, apenas o retorno HTTP com código 200 (OK):

