

Sumário

1 - Observações:	2
1.1 - Configurações para conseguir executar o projeto localmente	2
1.2 - Diagrama de classes:	2
1.3 - Software para enviar/receber as requisições	2
2 - Métodos	2
1.1 – POST - Cadastrar pessoa	2
1.2 – PUT – Atualizar Pessoa	3
1.3 – POST – Cadastrar doação	3
1.4 – POST – Cadastrar transfusão	3
1.5 – GET - Listagem de pessoas ordenada por qtde de doacoes	3
1.6 – GET - Listagem de pessoas ordenada por qtde de transfusoes.....	4
1.7 – GET - Listar doacoes por Id de pessoa	4
1.8 – GET Listar transfusoes por Id de pessoa	4

1 - Observações:

1.1 - Configurações para conseguir executar o projeto localmente

Para rodar o projeto, você deve ter instalado o Java 17, Spring Boot 3.2.4 e o banco de dados MySQL. A conexão com o banco de dados é feita no arquivo “**src/main/resources/application.properties**”, que possui o conteúdo a seguir:

```
spring.jpa.hibernate.ddl-auto=update

spring.datasource.url=jdbc:mysql://localhost:3306/bancodesangue

spring.datasource.username=root

spring.datasource.password=root


spring.jpa.show-sql=true

spring.jpa.properties.hibernate.format_sql=true
```

As propriedades “**spring.datasource.url**”, “**spring.datasource.username**” e “**spring.datasource.password**” devem ser alteradas de acordo com a sua instalação local do MySQL. O resto não deve ser alterado.

1.2 - Diagrama de classes:

Segue o link do diagrama de classes do projeto.

<https://online.visual-paradigm.com/share.jsp?id=333136323233352d31#diagram:workspace=vyeinsqz&proj=0&id=1>

1.3 - Software para enviar/receber as requisições

Para enviar/receber as requisições descritas a seguir, podem ser usados os softwares POSTMAN ou INSOMNIA, tanto faz.

2 - Métodos

1.1 – POST - Cadastrar pessoa

<http://localhost:8080/pessoas>

A primeira coisa a ser feita é utilizar esse POST para cadastrar novas pessoas no sistema. Segue abaixo dados fictícios de uma pessoa qualquer. Observação: o CPF não pode ser repetido.

```
{
  "cpf": "123456789-09",
  "nome": "Jose Josenildo",
  "celular": "41 45987-1591",
  "email": "josenildo@email.com",
  "tipoSanguineo": "TIPO_B_NEGATIVO",
  "dataDeNascimento": "2000-06-04",
  "regionalHemocentroResidencia": "PIRACICABA"
}
```

1.2 – PUT – Atualizar Pessoa

<http://localhost:8080/pessoas>

Esse PUT permite atualizar uma pessoa, cujo Id é especificado no body.

```
{
  "id": "1",
  "nome": "Gustavo Antunes",
  "celular": "11 93769-4868",
  "e-mail": "nome@email.com",
  "dataDeNascimento": "17/11/1996",
  "tiposanguineo": "TIPO_B_NEGATIVO",
  "regionalHemocentroResidencia": "GRANDE_SAO_PAULO"
}
```

1.3 – POST – Cadastrar doação

<http://localhost:8080/doacoes>

Esse post permite cadastrar numa nova doação.

```
{
  "cpfDoador": "123456789-02",
  "tipoSanguineo": "TIPO_A_POSITIVO",
  "quantidadeDoadada": "200",
  "regionalHemocentroDoacao": "GRANDE_SAO_PAULO"
}
```

1.4 – POST – Cadastrar transfusão

<http://localhost:8080/transfusoes>

Esse POST permite cadastrar uma nova transfusão.

```
{
  "cpf": "123456789-01",
  "nome": "Gustavo Antunes",
  "celular": "11 98765-4321",
  "email": "nome@email.com",
  "endereco": { "logradouro": "Rua 1", "numero": "1", "bairro": "Centro", "cep": "06726-458", "cidade": "Osasco", "uf": "SP", "pais": "Brasil" },
  "tiposanguineo": "TIPO_O_POSITIVO",
  "regionalHemocentroResidencia": "GRANDE_SAO_PAULO"
}
```

1.5 – GET - Listagem de pessoas ordenada por qtde de doacoes

<http://localhost:8080/pessoas/doacoes>

Esse GET permite obter a listagem de todas as pessoas cadastradas no sistema, ordenadas de acordo com a quantidade de doações realizadas.

1.6 – GET - Listagem de pessoas ordenada por qtde de transfusões

<http://localhost:8080/pessoas/transfusoes>

Esse GET permite obter a listagem de todas as pessoas cadastradas no sistema, ordenadas de acordo com a quantidade de transfusões realizadas.

1.7 – GET - Listar doacoes por Id de pessoa

<http://localhost:8080/pessoas/doacoes/4>

Esse GET permite obter todas as doações realizadas por uma pessoa com determinado Id. No exemplo acima, serão mostradas todas as doações realizadas pela pessoa de id "4". Basta mudar o número final da URL para mostrar as doações de outras pessoas.

1.8 – GET Listar transfusoes por Id de pessoa

<http://localhost:8080/pessoas/transfusoes/1>

Esse GET permite obter todas as transfusões realizadas por uma pessoa com determinado Id. No exemplo acima, serão mostradas todas as doações realizadas pela pessoa de id "1". Basta mudar o número final da URL para mostrar as doações de outras pessoas.