

API REST – BRASILPREV

Esse projeto foi desenvolvido utilizando as seguintes tecnologias: **PYTHON, DJANGO, DJANGO REST FRAMEWORK, SQLITE, DOCKER e DOCKER COMPOSE.**

O objetivo do projeto é uma **API REST** capaz de executar as seguintes ações:

- **Cadastro de cliente**
- **Cadastro de produto**
- **Contratação de plano**
- **Resgate de plano**
- **Aporte extra**

Seguindo as regras impostas pelo desafio da brasilprev.

EXECUÇÃO DO PROJETO

DOCKER:

Após descompactar o projeto, presumindo que o usuário já tenha o **Docker** e o **DOCKER COMPOSE** instalados em sua máquina, dentro do diretório raiz do projeto onde se encontra o arquivo “**docker-compose.yml**” digite os seguintes comandos:

1. `docker-compose up -d`
2. após a finalização da execução do docker compose, acessar a aplicação no seu navegador (**GOOGLE, BRAVE, FIREFOX, EDGE**) pelo seu endereço IP local, exemplo: “**192.168.1.10**”.

obs: para descobrir o seu endereço IP, no prompt de comando ou terminal, você dará o seguinte comando no seu terminal:

- **WINDOWS – ipconfig**
- **LINUX – ifconfig**

EXECUÇÃO MANUAL:

Caso o usuário não tenha o docker e o docker compose instalados em sua máquina, siga os passos abaixo para uma execução do projeto alternativa.

Dentro do diretório raiz do projeto, onde se encontra o arquivo “manage.py” faça:

1. CRIE UMA VENV

- No terminal digite: **python -m venv .venv**

2. ATIVAR VENV

- WINDOWS - **.venv\Scripts\activate.bat**
- LINUX – **source .venv/bin/activate**

3. INTALANDO OS PACOTES DO PROJETO DENTRO DA VENV

- No terminal digite: **pip install -r requirements.txt**

4. ACESSANDO A APLICAÇÃO

- No terminal digite: **python manage.py runserver**
- No seu navegador acessa a aplicação pelo endereço IP:
“http://127.0.0.1:8000/”

CADASTRO CLIENTE:

[POST, GET, PUT, DELETE]

"http://IP/cadastro_cliente/"

EXEMPLE REQUEST:

POST

```
{  
  "cpf": "45645645600",  
  "name": "Jose da Silva",  
  "email": "jose@cliente.com",  
  "dataDeNascimento": "2010-01-01",  
  "sexo": "M",  
  "rendaMensal": 3000  
}
```

EXEMPLE RESPONSE:

```
{  
  "id": 1,  
  "cpf": "45645645600",  
  "name": "Jose da Silva",  
  "email": "jose@cliente.com",  
  "dataDeNascimento": "2000-01-01",  
  "sexo": "M",  
  "rendaMensal": 3000.0  
}
```

CADASTRO PRODUTO:

[POST, GET, PUT, DELETE]

"http://IP/cadastro_produto/"

EXEMPLE REQUEST:

POST

```
{
  "name": "Brasilprev Longo Prazo",
  "susep": "15414.900840/2018-17",
  "expiracaoDeVenda": "2024-01-01",
  "valorMinimoAporteInicial": 1000,
  "valorMinimoAporteExtra": 100,
  "idadeDeEntrada": 18,
  "idadeDeSaida": 60,
  "carenciaInicialDeResgate": 60,
  "carenciaEntreResgates": 30
}
```

EXEMPLE RESPONSE:

```
{
  "id": 1,
  "name": "Brasilprev Longo Prazo",
  "susep": "15414.900840/2018-17",
  "expiracaoDeVenda": "2021-01-01",
  "valorMinimoAporteInicial": 1000.0,
  "valorMinimoAporteExtra": 100.0,
  "idadeDeEntrada": 18,
  "idadeDeSaida": 60,
  "carenciaInicialDeResgate": 60,
  "carenciaEntreResgates": 30
}
```

CONTRATAÇÃO PLANO:

[POST, GET, PUT, DELETE]

"http://IP/ contratacao_plano/"

EXEMPLE REQUEST:

POST

```
{  
  "aporte": 1000,  
  "dataDaContratacao": "2023-01-16",  
  "idCliente": 1,  
  "idProduto": 1  
}
```

EXEMPLE RESPONSE:

```
{  
  "id": 1,  
  "aporte": 1000.0,  
  "dataDaContratacao": "2023-01-16",  
  "idCliente": 1,  
  "idProduto": 1  
}
```

APOORTE EXTRA:

[POST, GET, PUT, DELETE]

"http://IP/ aporte_extra/"

EXEMPLE REQUEST:

POST

```
{  
  "valorAporte": 200,  
  "idCliente": 1,  
  "idPlano": 1  
}
```

EXEMPLE RESPONSE:

```
{  
  "id": 1,  
  "valorAporte": 200.0,  
  "idCliente": 1,  
  "idPlano": 1  
}
```

RESGATE:

[POST, GET, PUT, DELETE]

"http://IP/ resgate/"

EXEMPLE REQUEST:

POST

```
{  
  "valorResgate": 1000,  
  "idPlano": 1  
}
```

EXEMPLE RESPONSE:

```
{  
  "id": 1,  
  "valorResgate": 1000.0,  
  "idPlano": 1  
}
```