

# BASES DE DADOS - Installation Manual - Projeto 2023/24 Hospital Management System

Diogo de Oliveira Mendes Dias Saldanha - 2022232761

João António Faustino Vaz -2022231087

João Maria Moreira Dias - 2022225061



1 2



9 0

FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE DE  
COIMBRA

## **Introdução**

Para este projeto é necessário obter algumas ferramentas e portanto elaboramos este documento para permitir uma melhor compreensão.

## **Linguagens de Programação necessárias**

Python  
SQL e pgSQL

## **Sistema de Gerenciamento da base de dados**

pgAdmin 4

## **Bibliotecas utilizadas em Python**

- flask
- logging
- psycopg2
- hashlib
- jwt
- datetime

## **Outros ferramentas utilizadas**

- Onda
- Postman

## Instalação das ferramentas e dos softwares utilizados

Antes de começar a elaborar o projeto é necessário verificar se temos todas as bibliotecas e, se não tivermos, é preciso instalar. Logo, encontra-se abaixo um guia com comandos para verificar o que é preciso.

- pip install flask
- pip install psycopg2
- pip install jwt

Relativamente ao SQL é necessário instalar, se possível, a última versão da **PostgreSQL** ou então atualizar caso já o tenha instalado. Para utilizar o **Postman** também é necessário instalar, pode ser feito a partir de uma página web (<https://www.postman.com/downloads/>).

Para o acesso à base de dados, pode-se usar a **psql** ou **pgadmin4**, onde é necessário configurar o acesso através de um username e password, a porta e o host já estão definidos.

## Mais informação

<https://www.python.org/>

<https://www.postgresql.org/>

<https://www.postman.com/>

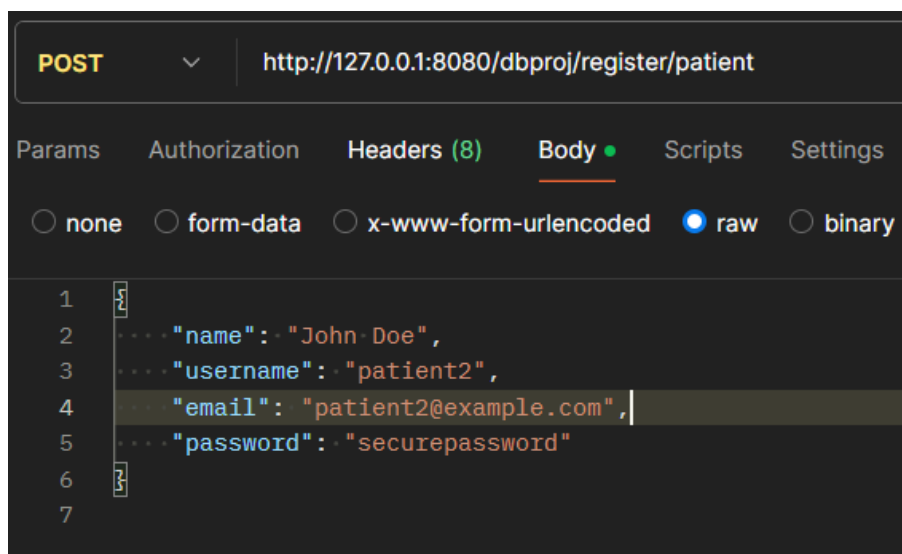
# USER MANUAL

## Registo de utilizadores

Descrição: Serve para registar tanto doentes como empregados.

URL: <http://127.0.0.1:8080/dbproj/register/> <user type>

Método: POST



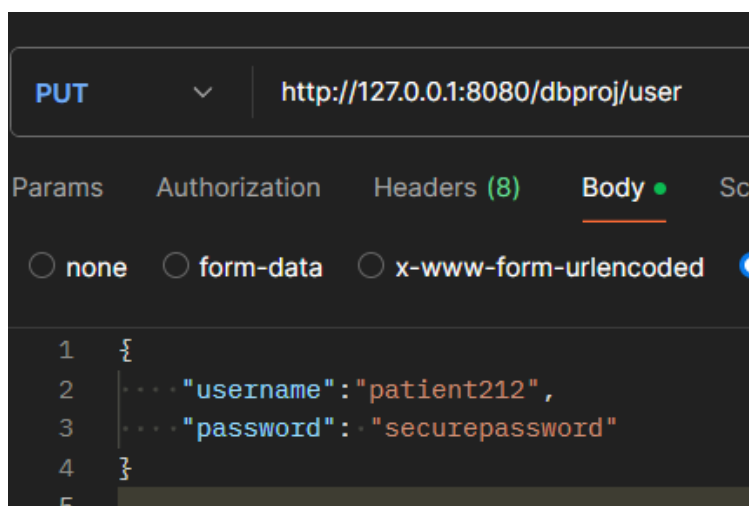
```
{
  "errors": null,
  "results": 7,
  "status": 200
}
```

## Login dos utilizadores

Descrição: Serve para fazer login, retornando um token que será usado noutras funções.

URL: <http://127.0.0.1:8080/dbproj/user>

Método: PUT



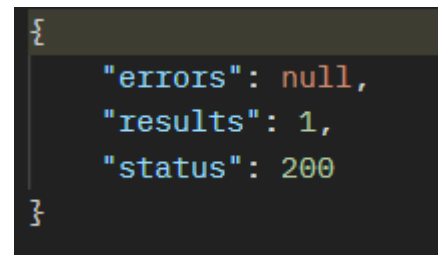
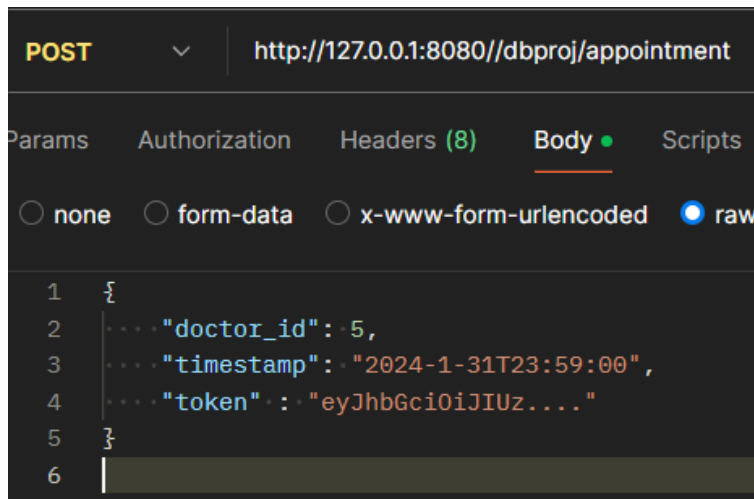
```
{
  "errors": null,
  "results": "eyJhbGciOiJIUzI1
  "status": 200
}
```

### Marcar consulta

Descrição: Marca uma consulta, apenas os pacientes podem marcar.

URL: <http://127.0.0.1:8080/dbproj/appointment>

Método: POST

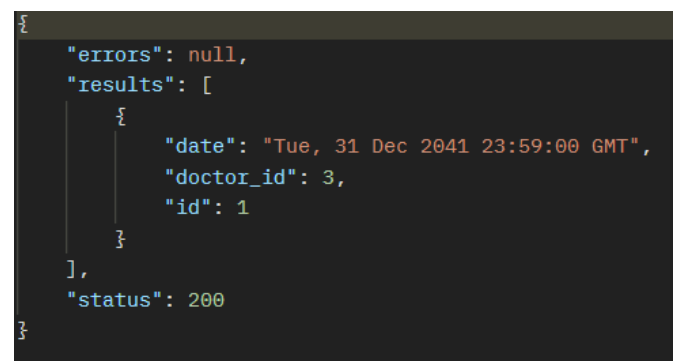
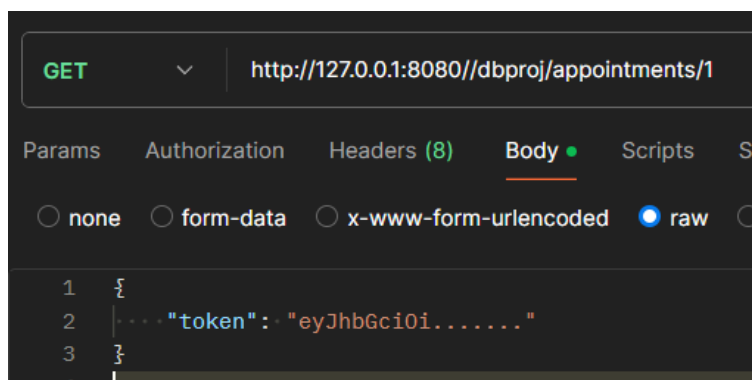


### Listar consultas

Descrição: Lista as consultas de um certo paciente, apenas as assistentes e o paciente em questão podem usar esta função.

URL: [http://127.0.0.1:8080/dbproj/appointments/<patient\\_user\\_id>](http://127.0.0.1:8080/dbproj/appointments/<patient_user_id>)

Método: GET

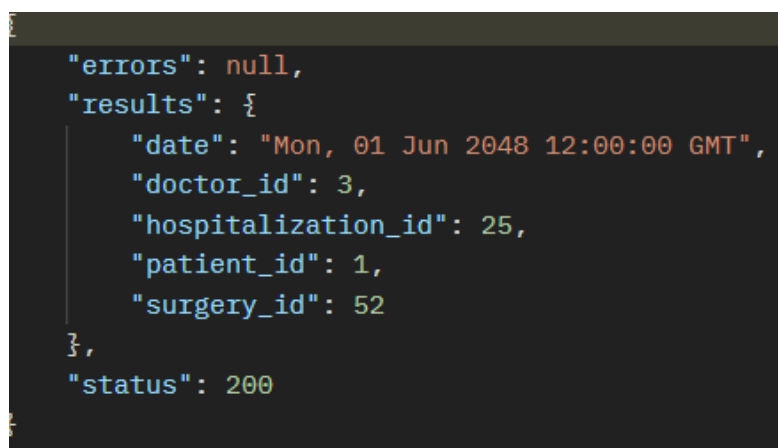
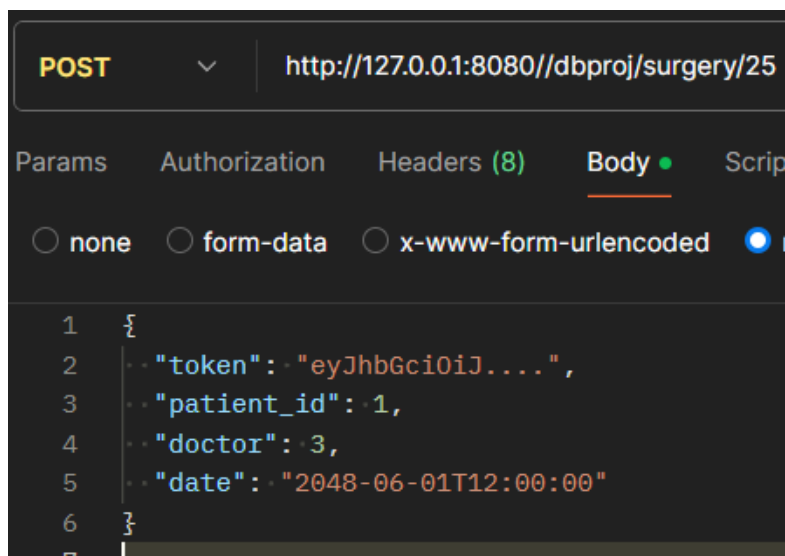
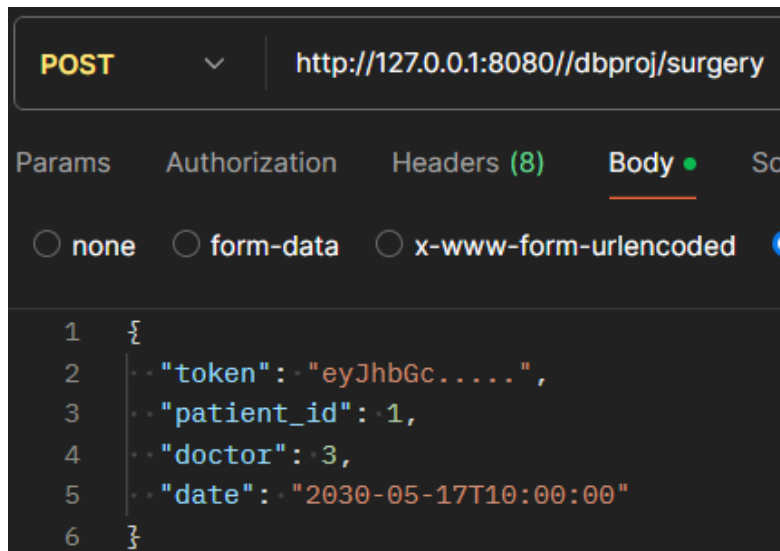


## Marcar cirurgias

Descrição: Marcar uma cirurgia podendo estar associada já a uma hospitalização ou não, caso não esteja é criada uma nova hospitalização, apenas as assistentes podem usar esta função.

URL: `http://127.0.0.1:8080/dbproj/surgery/(<hospitalization_id>)`

Método: POST



### Criar um prescrição médica

Descrição: Criar uma prescrição, podendo vir de uma consulta ou de uma hospitalização, apenas os médicos podem usar esta função.

URL: <http://127.0.0.1:8080/dbproj/prescription>

Método: POST

```
POST http://127.0.0.1:8080/dbproj/prescription

Body (raw)

{
  "token": "eyJhbGciOiJI...",
  "type": "appointment",
  "event_id": 1,
  "validity": "2046-06-01T10:00:00",
  "medicines": [
    {
      "medicine": "medicine1_name",
      "posology_dose": 10,
      "posology_frequency": "daily"
    },
    {
      "medicine": "medicine2_name",
      "posology_dose": 5,
      "posology_frequency": "twice a day"
    }
  ]
}
```

```
{
  "errors": null,
  "results": 2,
  "status": 200
}
```

### Ver prescrições

Descrição: Ver as prescrições de um certo doente, todos os empregados e o doente em questão podem usar esta função.

URL: [http://127.0.0.1:8080/dbproj/prescriptions/<person\\_id>](http://127.0.0.1:8080/dbproj/prescriptions/<person_id>)

Método: GET

```
GET http://127.0.0.1:8080/dbproj/prescriptions/1

Body (raw)

{
  "token": "eyJhL8DKxCXeUf0Rh6s....."
}
```

```
{
  "errors": null,
  "results": [
    {
      "id": 1,
      "posology": [
        {
          "dose": 5,
          "frequency": "twice a day",
          "medicine": "medicine2_name"
        },
        {
          "dose": 10,
          "frequency": "daily",
          "medicine": "medicine1_name"
        }
      ]
    },
    {
      "type": "hospitalization",
      "validity": "Fri, 01 Jun 2046 10:00:00 GMT"
    }
  ]
}
```

Diogo Saldanha - 2022232761

João Vaz - 2022231087

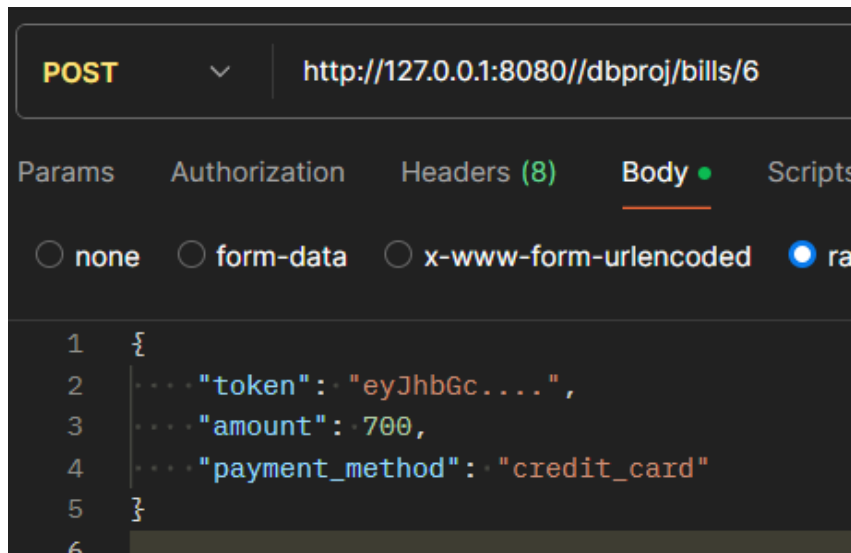
João Dias - 2022225061

### Pagar uma fatura

Descrição: Pagar um valor indicado pelo utilizador, apenas o utilizador pode utilizar esta função.

URL: `http://127.0.0.1:8080/dbproj/bills/<bill_id>`

Método: POST



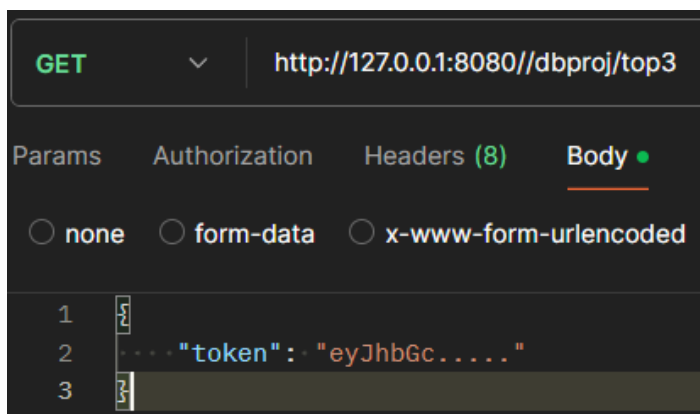
```
{
  "errors": null,
  "results": {
    "remaining_value": 0
  },
  "status": 200
}
```

### Apresentar o top 3

Descrição: Apresentar o top 3 doentes com mais dinheiro gasto no hospital, apenas os assistentes podem usar esta função.

URL: `http://127.0.0.1:8080/dbproj/top3`

Método: GET



```
{
  "errors": null,
  "results": [
    {
      "amount_spent": "2100",
      "patient_name": "John Doe",
      "patient_username": "patient1",
      "procedures": [
        {
          "date": "2024-05-12T10:00:00",
          "id": 1
        },
        {
          "date": "2024-05-13T10:00:00",
          "id": 2
        },
        {
          "date": "2024-05-14T10:00:00",
          "id": 3
        }
      ]
    },
    {
      "amount_spent": "1400",
      "patient_name": "John Doe",
      "patient_username": "patient2",
      "procedures": [
        {
          "date": "2024-05-15T10:00:00",
          "id": 4
        }
      ]
    }
  ]
}
```

8

Diogo Saldanha - 2022232761

João Vaz - 2022231087

João Dias - 202225061

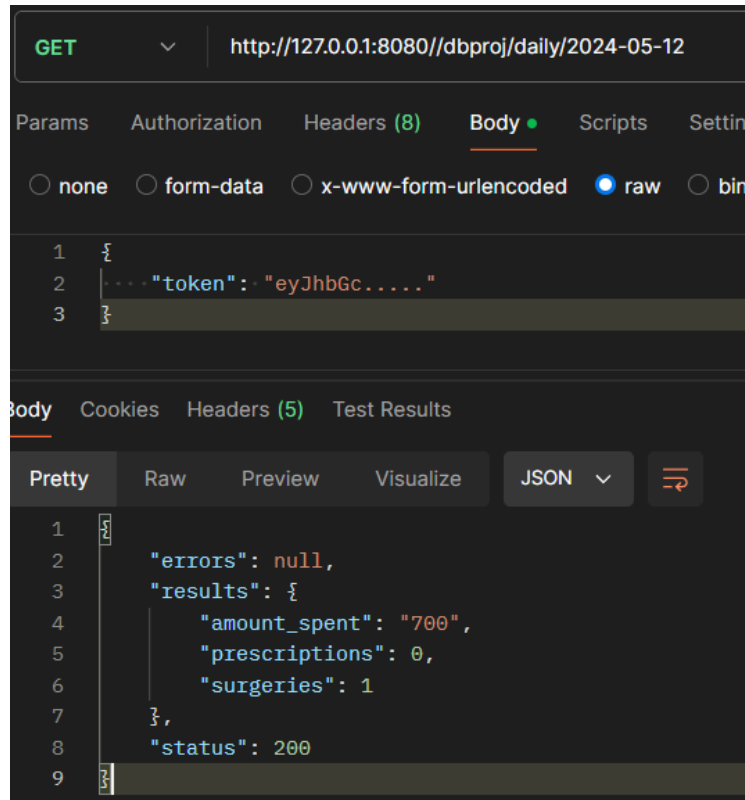


### Informações sobre um certo dia

Descrição: Listar as hospitalizações num certo dia, apresentando as cirurgias, os pagamentos e as prescrições associadas, apenas os assistentes podem usar esta função.

URL: `http://127.0.0.1:8080/dbproj/daily/<ano-mes-dia>`

Método: GET



```
GET http://127.0.0.1:8080/dbproj/daily/2024-05-12

{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTJ5In0:"
}

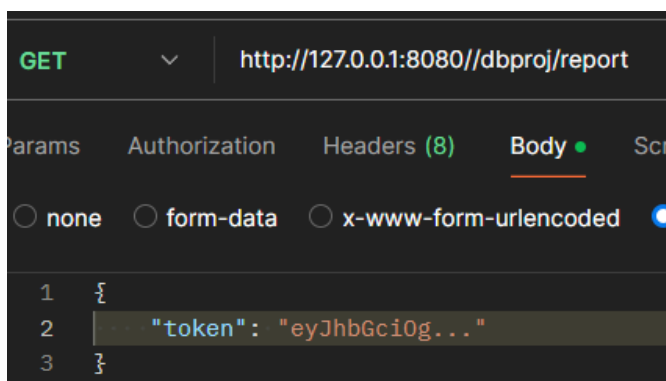
{
  "errors": null,
  "results": {
    "amount_spent": "700",
    "prescriptions": 0,
    "surgeries": 1
  },
  "status": 200
}
```

### Gerar um relatório

Descrição: Gera um relatório dos últimos 12 meses indicando o médico mais cirurgias em cada mês , apenas os assistentes podem usar esta função.

URL: `http://127.0.0.1:8080/dbproj/report`

Método: GET



```
GET http://127.0.0.1:8080/dbproj/report

{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTJ5In0:"
}
```



```
{
  "errors": null,
  "results": [
    {
      "doctor": "Dr. Jane Smith",
      "month": "2024-05",
      "surgeries": 6
    }
  ],
  "status": 200
}
```

## Informações adicionais

### Development plan

No desenvolvimento deste projeto trabalhamos todos em conjunto mas no início começamos por distribuir tarefas e podemos dizer que a maior parte foi cumprida. Abaixo a tabela inicial da distribuição das tarefas!

Diogo Saldanha	João Vaz	João Dias
Registo dos utilizadores e login	Cirurgia com hospitalização	Diagrama final
Prescrições	Consultas	Sumário diário
Cirurgia	top3	Report por mês
relatório e powerpoint	relatório e powerpoint	relatório e powerpoint

### Triggers

Uma das funcionalidades mais importantes do nosso trabalho é a implementação de triggers.

No total implementamos dois triggers. Um relativamente às consultas e outro relativamente às cirurgias. O que acontece é que sempre que marcamos uma consulta o trigger é ativado e é adicionada uma nova fatura ( bill ). O mesmo acontece em relação às cirurgias, caso se use o primeiro endpoint é sempre criada uma nova fatura, caso se use o segundo endpoint ,ou seja, estamos a associar a uma hospitalização já existente, a fatura já existente é atualizada.

## Diagrama Entidade-Relacionamento e Modelo-Relacional

