PYTHON API REST FLASK (USER GUIDE)

Software engineer: *Rafael Costa Teixeira*

Version: *1.0 (16/11/2020)*
Project on my GitHub:
https://github.com/rafaelct/python-api-flask

Python Api Rest Flask

Index

Introduction

This document describes the functions the API Rest (requests and responses with examples).
To start using, it will be necessary to register a user in the system (see 1.1).
All other api requests require a valid token, to obtain a valid token, it will be necessary to authenticate with the system (see 1.2).
The token is automatically invalidated after 5 minutes without calling any request, after that period you will have to authenticate again in the system to obtain a new valid token (see 1.2).
Note the first request may take approximately 20 seconds, due to a feature of the free Heroku account where the application is hosted.

Enjoy :-)

1. Document API Rest (requests and responses with examples)

1.1 Register a new user in the system.

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/registry		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
login	login user	Yes	String
password	password user	Yes	Number
fullname	fullname of the user	Yes	Number

Example

Request (JSON)

```
{
   "loginname":"teste",
   "password":"teste",
   "fullname":"Teste o tal"
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [],
   "msgReturn": "Success"
}
```

1.2 Authenticate a user in the system

<table>
<tr><th colspan="4">Request (http headers)</th></tr>
<tr><td><b>Content-type</b></td><td colspan="3">application/json</td></tr>
<tr><td><b>Method</b></td><td colspan="3">POST</td></tr>
<tr><td><b>URL</b></td><td colspan="3">https://calm-hamlet-60163.herokuapp.com/auth</td></tr>
<tr><th colspan="4">Request (http data body)</th></tr>
<tr><td><b>Fields</b></td><td><b>Describe</b></td><td><b>Required</b></td><td><b>Format</b></td></tr>
<tr><td>login</td><td>login user</td><td>Yes</td><td>String</td></tr>
<tr><td>password</td><td>password user</td><td>Yes</td><td>Number</td></tr>
<tr><th colspan="4">Response (http data body)</th></tr>
<tr><td><b>Fields</b></td><td><b>Describe</b></td><td><b>Field father</b></td><td><b>Format</b></td></tr>
<tr><td>codReturn</td><td>return status code</td><td>-</td><td>Number (0 = OK
1 = Erro)</td></tr>
<tr><td>msgReturn</td><td>Message error or Success</td><td>-</td><td>String</td></tr>
<tr><td>data</td><td>Array in the fields</td><td>-</td><td>List</td></tr>
<tr><td>token</td><td>token session valid</td><td>data</td><td>String</td></tr>
</table>

Example

Request (JSON)

```
{
   "loginname":"teste",
   "password":"teste"
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": {
      "token": "16054443937095"
   },
   "msgReturn": "Success"
}
```

1.3 Return one or more customers

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/customers/get		
Request (http data body)			
Field	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	ID customer	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List
name	Name customer	data	String
fullname	Fullname customer	data	String
birth	Birth customer	data	MM/DD/YYYY
document	CPF customer	data	Number

gender	Sex	data	M = Male
			F = Female

Example

Request (JSON)

```
{
  "token":"16052984452697"
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [
    {
      "birth": "06/05/1982",
      "document": "12345678909",
      "fullname": "Rafael Costa Teixeira",
      "gender": "M",
      "id": 1,
      "name": "Rafael"
    }
  ],
  "msgReturn": "Success"
}
```

1.4 Register a customer

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/customers		
Request (http data body)			
Field	**Describe**	**Required**	**Format**
token	Token session	Yes	String
name	Name customer	Yes	String
fullname	Fullname customer	Yes	String
birth	Birth customer	Yes	MM/DD/YYYY
document	CPF customer	Yes	Number

gender	Sex	Yes	M = Male
			F = Female

Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
  "token":"16054442065000",
  "birth": "06/05/1982",
  "document": "12345678909",
  "fullname": "Rafael Costa Teixeira",
  "gender": "M",
  "name": "Rafael"
   }
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [],
   "msgReturn": "Success"
}
```

1.5 Update a customer

Request (http headers)			
Content-type	application/json		
Method	PUT		
URL	https://calm-hamlet-60163.herokuapp.com/customers		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id customer	Yes	Number
name	Name customer	Yes	String
fullname	Fullname customer	Yes	String
birth	Birth customer	Yes	MM/DD/YYYY
document	CPF customer	Yes	Numérico

gender	Sex	Yes	M = Male
			F = Female

Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
   "token":"16054442065000",
   "name":"Rafaela",
   "id":"10",
   "fullname":"Rafaela Novaes",
   "birth":"12/31/1959",
   "document":"12398745678",
   "gender":"F"
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [],
  "msgReturn": "Success"
}
```

1.6 Remove a customer

Request (http headers)			
Content-type	application/json		
Method	DELETE		
URL	https://calm-hamlet-60163.herokuapp.com/customers		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id customer	Yes	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
  "token":"16054442065000",
  "id":"10"
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [],
  "msgReturn": "Success"
}
```

1.7 Return a one or more product

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/products/get		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id product	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List
token	Token session	data	String
id	Id product	data	Number
name	Name of the product	data	String
brand	Name brand	data	String
codProduct	Cód. of the product	data	Number
qtd	Quantity in stock	data	Number
linkImg	Url Image	data	String
price	Price of product	data	String (ex: 100.00)

Example

Request (JSON)

```
{
  "token":"16054442065000",
  "id":"2"
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [
    {
      "brand": "Continental",
      "codProduct": 2,
      "id": 2,
      "linkImg": "/cod_1.png",
      "name": "Fogão",
      "qtd": 1,
      "registration": "11/13/2020",
      "price": "$500.00"
    }
  ],
  "msgReturn": "Success"
}
```

1.8 Register a product

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/products		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
name	Name product	Yes	String
brand	Name brand	Yes	String
codProduct	Cód. product	Yes	Number
qtd	Quantity in stock	Yes	Number
linkImg	Url image	Yes	String
price	Price of product	Yes	String (ex: 100.00)
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
    {
      "token":"16054442065000",
      "brand": "Brastemp",
      "codProduct": "2",
      "linkImg": "/cod_1.png",
      "name": "Fogão",
      "qtd": "1",
      "price": "500.00"
    }
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [],
   "msgReturn": "Success"
}
```

1.9 Update a product

Request (http headers)			
Content-type	application/json		
Method	PUT		
URL	https://calm-hamlet-60163.herokuapp.com/products		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id product	Yes	Number
name	Name of the product	No	String
brand	Name brand	No	String
codProduct	Cód. of the product	No	Number
qtd	Quantity in stock	Yes	Number
linkImg	Url Image	No	String
price	Price of product	Yes	String (ex: 100.00)
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
   "token":"16054442065000",
   "brand": "Continental",
   "codProduct": "2",
   "id": "2",
   "linkImg": "/cod_1.png",
   "name": "Fogão",
   "qtd": "1",
   "price": "500.00"
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [],
   "msgReturn": "Success"
}
```

1.10 Add quantity of the product

Request (http headers)			
Content-type	application/json		
Method	DELETE		
URL	https://calm-hamlet-60163.herokuapp.com/products		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id do Produto	Yes	Number
qtd	Quantity in stock	Yes	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
   "token":"16054442065000",
   "id": "2",
   "qtd": "2",
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [],
   "msgReturn": "Success"
}
```

1.11 Substract quantity of the product

Request (http headers)			
Content-type	application/json		
Method	DELETE		
URL	https://calm-hamlet-60163.herokuapp.com/products		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id do Produto	Yes	Number
qtd	Quantity in stock	Yes	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
  "token":"16054442065000",
  "id": "2",
  "qtd": "2",
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [],
  "msgReturn": "Success"
}
```

1.12 Return a one or more Orders

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/orders/get		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
id	Id order	No	Number
customerId	Id do cliente	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List
token	Token session	data	String
id	Id order	data	Number
customerId	Id do cliente	data	Number
productId	productId of the product	data	Number

Example

Request (JSON)

```
{
   "token":"16055288961839"
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [
     {
        "customerId": 2,
        "id": 1,
        "orderStatus": "Aguardando Pagamento",
        "productId": 2,
        "registration": "11/15/2020"
     },
     {
        "customerId": 2,
        "id": 2,
        "orderStatus": "Aguardando Pagamento",
      "productId": 1,
      "registration": "11/15/2020"
    }
  ],
  "msgReturn": "Success"
}
```

1.13 Register a order

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/orders		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
Token	Token session	No	String
Id	Id order	No	Number
customerId	Id customer	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
  "token":"16055288961839",
  "customerId": "2"
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [],
  "msgReturn": "Success"
}
```

1.14 Return one or more carts

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/carts/get		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
Id	Id cart	No	Number
customerId	Id customer	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

Request (JSON)

```
{
  "token":"16055288961839",
  "customerId": "2"
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [
    {
      "customerId": 2,
      "id": 1,
      "productId": 2,
      "registration": "11/15/2020"
    },
    {
      "customerId": 2,
      "id": 2,
      "productId": 1,
      "registration": "11/15/2020"
    }
  ],
  "msgReturn": "Success"
}
```

1.15 Register a cart

Request (http headers)			
Content-type	application/json		
Method	POST		
URL	https://calm-hamlet-60163.herokuapp.com/carts		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
Id	Id cart	No	Number
customerId	Id customer	No	Number
productId	Id product	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example
Request (JSON)

```
{
  "token":"16055288961839",
  "customerId": "2",
  "productId": "2",
}
```

Response (JSON)

```
{
  "codReturn": 0,
  "data": [],
  "msgReturn": "Success"
}
```

1.16 Remove a cart

Request (http headers)			
Content-type	application/json		
Method	DELETE		
URL	https://calm-hamlet-60163.herokuapp.com/carts		
Request (http data body)			
Fields	**Describe**	**Required**	**Format**
token	Token session	Yes	String
Id	Id cart	No	Number
customerId	Id customer	No	Number
productId	Id product	No	Number
Response (http data body)			
Fields	**Describe**	**Field father**	**Format**
codReturn	return status code	-	Number (0 = OK 1 = Erro)
msgReturn	Message error or Success	-	String
data	Array in the fields	-	List

Example

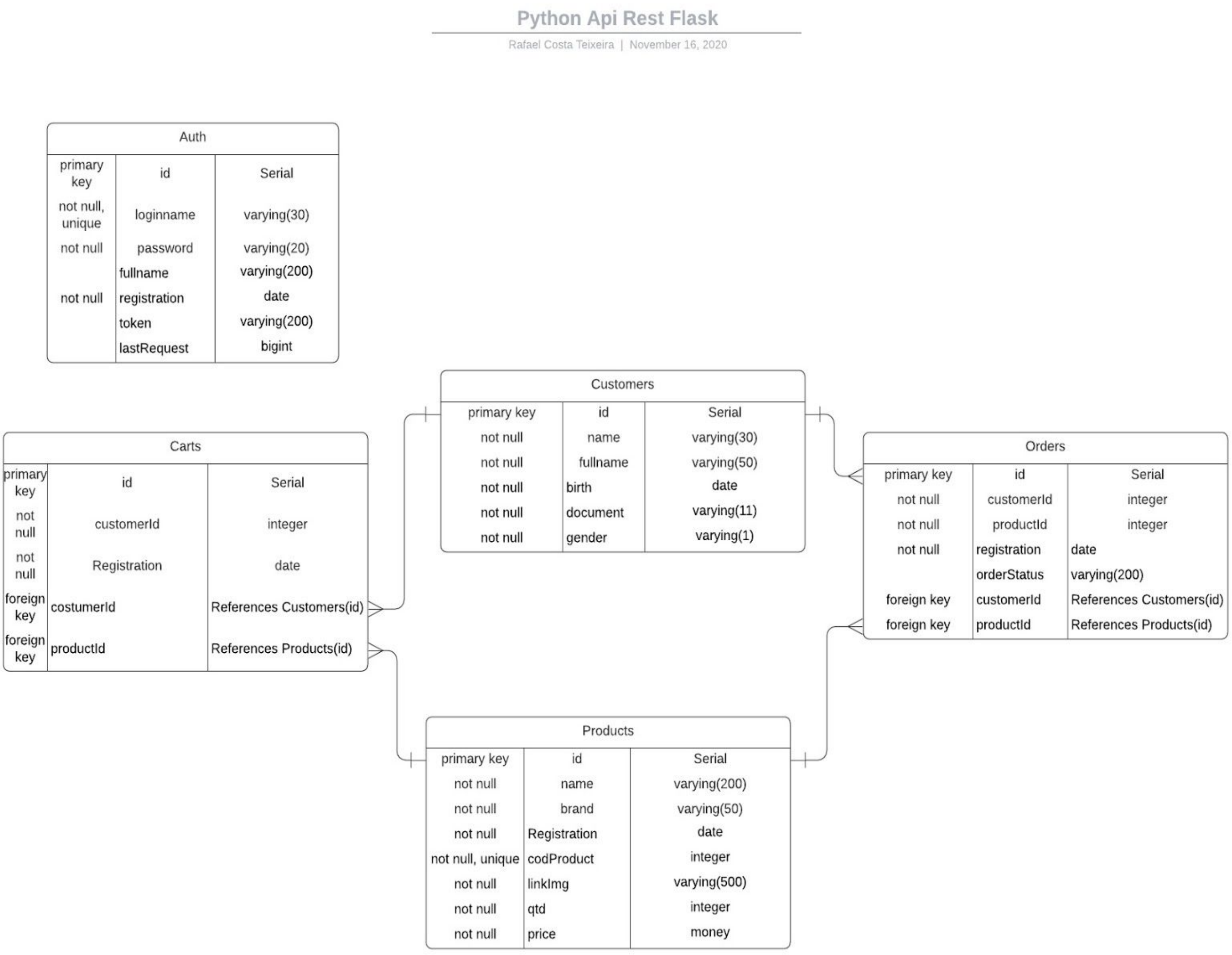
Request (JSON)

```
{
   "token":"16055288961839",
   "customerId": "2"
}
```

Response (JSON)

```
{
   "codReturn": 0,
   "data": [],
   "msgReturn": "Success"
}
```

2. Postgresql Data Base Models



Python Api Rest Flask

Rafael Costa Teixeira | November 16, 2020

Auth		
primary key	id	Serial
not null, unique	loginname	varying(30)
not null	password	varying(20)
	fullname	varying(200)
not null	registration	date
	token	varying(200)
	lastRequest	bigint

Customers		
primary key	id	Serial
not null	name	varying(30)
not null	fullname	varying(50)
not null	birth	date
not null	document	varying(11)
not null	gender	varying(1)

Carts		
primary key	id	Serial
not null	customerId	integer
not null	Registration	date
foreign key	costumerId	References Customers(id)
foreign key	productId	References Products(id)

Orders		
primary key	id	Serial
not null	customerId	integer
not null	productId	integer
not null	registration	date
	orderStatus	varying(200)
foreign key	customerId	References Customers(id)
foreign key	productId	References Products(id)

Products		
primary key	id	Serial
not null	name	varying(200)
not null	brand	varying(50)
not null	Registration	date
not null, unique	codProduct	integer
not null	linkImg	varying(500)
not null	qtd	integer
not null	price	money

3. Running project in your local machine

To run this project on your local machine, follow the instructions below:

Download the project to your local machine with the command:

```
git clone
https://github.com/rafaelct/python-api-flask.git
```

Enter the project folder:

```
cd python-api-flask
```

Go to the Python page and download version 3.x:

https://www.python.org/downloads/release/python-386/

On the python installation screen, check the option to include in PATH.

After the installation is complete, close the command prompt and open it again to consider the PATH environment variable updated.
Enter the project folder again.

Install the necessary python libraries in the project:

```
pip install flask
pip install psycopg2-binary
pip install -U pytest
```

Download and install the Postgresql database from the link below:

https://www.enterprisedb.com/postgresql-tutorial-resources-training?cid=48

After installed and configured, enter the Windows settings under *Systems-> Edit environment variables* and create a variable named DATABASE_URL, and in the content put the database connection string.

Access pgAdmin in query tools, run the contents of the scripts.sql file, to create the database structure.

I started the application with the command below:

```
python web.py
```

I started the application with the command below:

```
http://localhost:5000/<desired service request url>
```

4. Running automated tests on your local machine

Enter the project folder (downloaded in the previous step, see 3):

Clean the database by running the scripts.sql file in pgtools:

To test on localhost use:

```
pytest teste_web_urls.py
```

To test the model classes use:

```
pytest teste_web.py
```