

# DOCUMENTAÇÃO DE REST APIs HOTEL

## Rotas existentes:

GET: http://127.0.0.1:5000/hoteis .....	2
GET: http://127.0.0.1:5000/hoteis?_FILTROS_AVANCADOS_ .....	3
GET: http://127.0.0.1:5000/hoteis/id .....	4
POST: http://127.0.0.1:5000/hoteis/id .....	4
PUT: http://127.0.0.1:5000/hoteis/id.....	6
DELETE: http://127.0.0.1:5000/hotel/id.....	8
POST: http://127.0.0.1:5000/cadastro_usuario.....	9
POST: http://127.0.0.1:5000/login_usuario.....	10
POST: http://127.0.0.1:5000/logout_usuario.....	12
GET: http://127.0.0.1:5000/usuarios/id .....	13
GET: http://127.0.0.1:5000/senha_usuarios.....	13
DELETE: http://127.0.0.1:5000/usuarios/id .....	15
GET: http://127.0.0.1:5000/sites .....	16
GET: http://127.0.0.1:5000/sites/url .....	17
POST: http://127.0.0.1:5000/sintes/url .....	18
DELETE: http://127.0.0.1:5000/sites/url.....	20
GET: http://127.0.0.1:5000/ativacao_usuario/id .....	21

## Requisição

**GET:** <http://127.0.0.1:5000/hoteis>

Obtenha dados de uma lista de hotéis armazenada no Banco de Dados.

Method	URL
GET	http://127.0.0.1:5000/hoteis

Status	Response Body
200 OK	<pre>{   "hotéis": [     {       "hotel_id": "agadir",       "nome": "Agadir Hotel",       "estrelas": 3.5,       "diaria": 250.0,       "cidade": "Rio de Janeiro",       "site_id": 1     },     {       "hotel_id": "dsa",       "nome": "DSA Hotel",       "estrelas": 3.5,       "diaria": 250.0,       "cidade": "São Paulo",       "site_id": 2     }   ],   "quantidade de itens": 2,   "quantidade de paginas": 1,   "pagina atual": 1 }</pre>

Status	Response Body
400 BAD REQUEST	<pre>{   "message": "Hotel não existe." }</pre>

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**GET:** [http://127.0.0.1:5000/hoteis?\\_FILTROS\\_AVANÇADOS\\_](http://127.0.0.1:5000/hoteis?_FILTROS_AVANÇADOS_)

Filtro avançado retorna pesquisa por filtros pré-definidos conforme consultas enviadas por URL.

Exp: [http://127.0.0.1:5000/hoteis?\\_FILTROS\\_AVANÇADOS\\_](http://127.0.0.1:5000/hoteis?_FILTROS_AVANÇADOS_)

### \_FILTROS\_AVANÇADOS\_:

- **cidade** ⇒ Filtrar hotéis pela cidade escolhida. Padrão: Nulo
- **estrelas\_min** ⇒ Avaliações mínimas de hotéis de 0 a 5: Padrão: 0
- **estrelas\_max** ⇒ Avaliações máximas de hotéis de 0 a 5: Padrão: 5
- **diaria\_min** ⇒ Valor mínimo da diária do hotel de R\$: Padrão: 0
- **diaria\_max** ⇒ Valor máximo da diária do hotel de R\$: Padrão: 0
- **site** ⇒ Filtra endereço de site por número do id. Padrão: Nulo
- **itens** ⇒ Quantidade máxima de elementos exibidos por página: Padrão: 3
- **pagina** ⇒ Navega pelas paginas conforme quantidade de itens exibido pelas paginas

Method	URL
GET	<a href="http://127.0.0.1:5000/hoteishoteis?cidade=Rio de Janeiro&amp;estrelas_max=4&amp;diaria_max=300">http://127.0.0.1:5000/hoteishoteis?cidade=Rio de Janeiro&amp;estrelas_max=4&amp;diaria_max=300</a>

Status	Response Body
200 OK	<pre>{   "hotéis": [     {       "hotel_id": "agadir",       "nome": "Agadir Hotel",       "estrelas": 3.5,       "diaria": 250.0,       "cidade": "Rio de Janeiro"     }   ],   "quantidade de itens": 1,   "quantidade de paginas": 1,   "pagina atual": 1 }</pre>

Status	Response Body
400 BAD REQUEST	{

	<pre> "message": "Hotel não existe." } </pre>
--	---

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**GET:** <http://127.0.0.1:5000/hoteis/id>

Obtenha dados de um hotel especificado por "id" enviado por URL Exp:

[http://127.0.0.1:5000/hoteis/hotel\\_id](http://127.0.0.1:5000/hoteis/hotel_id).

Method	URL
GET	<a href="http://127.0.0.1:5000/hoteis/id:str">http://127.0.0.1:5000/hoteis/id:str</a>

Status	Response Body
200 OK	<pre> {     "hotel_id": "dsa",     "nome": "DSA Hotel",     "estrelas": 3.5,     "diaria": 250.0,     "cidade": "São Paulo" } </pre>

Status	Response Body
400 BAD REQUEST	<pre> {     "message": "Hotel não existe." } </pre>

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**POST:** <http://127.0.0.1:5000/hoteis/id>

Obtenha cadastrando de um novo hotel apontado por novo “id” enviado por URL. Exp:

**http://127.0.0.1:5000/hoteis/hotel\_id.**

Para criação de cadastros é necessário preenchimento de todos os dados, possuir um login de usuário ativo para obtenção de token de acesso e **vincular** o hotel em um **site** pré-cadastrado no Banco de Dados.

Method	URL
<b>POST</b>	<b>http://127.0.0.1:5000/hoteis/<span style="color: green;">id:str</span></b>

Header	
Content-Type	application/json
Authorization	Bearer <b>SUA_SENHA</b>

Body	Request Body
JOSN	<pre>{     "nome" : "iii",     "estrelas" : 3.8,     "diaria" : 102.57,     "cidade" : "Minas Gerais",     "site_id": 1 }</pre>

Status	Response Body
<b>201 CREATED</b>	<pre>{     "hotel_id": "iii",     "nome": "iii",     "estrelas": 3.8,     "diaria": 102.57,     "cidade": "Minas Gerais",     "site_id": 1 }</pre>

Status	Response Body
<b>400 BAD REQUEST</b>	<pre>{     "message": "Hotel já existe." }</pre>

Status	Response Body
--------	---------------

404 NOT FOUND	Not Found
---------------	-----------

Status	Response Body
401 UNAUTHORIZED	<pre>{   "msg": "Missing Authorization Header" }  OU  {   "msg": "Token has expired" }  OU  {   "msg": "Missing 'Bearer' type in 'Authorization' header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>

Status	Response Body
422 UNPROCESSABLE ENTITY	<pre>{   "msg": "Bad Authorization header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }  OU  {   "msg": "Signature verification failed" }</pre>

## Requisição

**PUT:** <http://127.0.0.1:5000/hoteis/id>

Obtenha dados cadastrais do hotel **alterado** ou cadastro de um **novο hotel** apontado por “id” enviado por URL. Exp: **http://127.0.0.1:5000/hoteis/hotel\_id**

- **hotel\_id existe:** **Atualizar** dados **hotel**.
- **hotel\_id não existe:** Cria um **novο hotel**.

Para criação de cadastros ou atualização é necessário preenchimento de todos os dados, possuir um login de usuário ativo para obtenção de token de acesso e vincular o hotel em um site pré-cadastrado no Banco de Dados.

Method	URL
PUT	<b>http://127.0.0.1:5000/usuarios/id:str</b>

Header	
Content-Type	application/json
Authorization	Bearer <b>SUA_SENHA</b>

Body	Request Body
JOSN	{ "nome" : "???", "estrelas" : 5, "diaria" : 100, "cidade" : "???", "site_id": 5 }

Status	Response Body
200 OK 201 CREATED	{ "hotel_id": "iii", "nome": "???", "estrelas": 5.0, "diaria": 100.0, "cidade": "???", "site_id": 5 }

Status	Response Body
404 NOT FOUND	Not Found
Status	Response Body
401 UNAUTHORIZED	{ "msg": "Missing Authorization Header" } <b>OU</b> { "msg": "Token has expired" } <b>OU</b> { "msg": "Missing 'Bearer' type in 'Authorization' header. Expected 'Authorization: Bearer <JWT>'" }

Status	Response Body
422 UNPROCESSABLE ENTITY	{ "msg": "Bad Authorization header. Expected 'Authorization: Bearer <JWT>'" }

## Requisição

**DELETE:** <http://127.0.0.1:5000/hotel/id>

Excluir hotel apontado por “id” enviado por URL. Exp: **http://127.0.0.1:5000/hoteis/hotel\_id**

Para exclusão de hotel cadastrado no Banco de Dados é necessário possuir um login de usuário ativo para obtenção de token de acesso.

Method	URL
DELETE	http://127.0.0.1:5000/usuarios/id:str

Header	
Authorization	Bearer SUA_SENHA

Status	Response Body
200 OK	{ "message": "Hotel deletado." }

Status	Response Body
400 BAD REQUEST	{ "message": "Hotel já existe." }

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
401 UNAUTHORIZED	{ "msg": "Missing Authorization Header" }



	<p><b>OU</b></p> <pre>{   "msg": "Token has expired" }</pre> <p><b>OU</b></p> <pre>{   "msg": "Missing 'Bearer' type in 'Authorization' header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>
--	---

Status	Response Body
422 UNPROCESSABLE ENTITY	<pre>{   "msg": "Bad Authorization header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>

## Requisição

**POST:** [http://127.0.0.1:5000/cadastro\\_usuario](http://127.0.0.1:5000/cadastro_usuario)

Obtenha cadastrando de um novo usuário onde o **novo** "id" é gerado automaticamente por um número inteiro sequencial. Para criação de cadastros é necessário preenchimento de todos os dados. Após cadastramento será enviado uma mensagem para o e-mail cadastrado para ativação do cadastro.

Method	URL
POST	<a href="http://127.0.0.1:5000/cadastro">http://127.0.0.1:5000/cadastro</a>

Header	
Content-Type	application/json

Body	Request Body
JOSN	<pre>{   "login" : "xxx",   "senha" : "xxx",   "email" : "xxx@xxx.com" }</pre>

Status	Response Body
201 CREATED	<pre>{   "mensagem": "Usuário criado com sucesso!!!" }</pre>

Status	Response Body
400 BAD REQUEST	<pre>{   "mensagem": "Login 'qqq' já existe!!!" }</pre> <p style="text-align: center;"><b>OU</b></p> <pre>{   "message": "Failed to decode JSON object: Expecting value: line 1 column 1 (char 0)" }</pre>

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
415 UNSUPPORTED MEDIA TYPE	<pre>{   "message": "Did not attempt to load JSON data because the request Content-Type was not 'application/json'." }</pre>

## Requisição

**POST:** [http://127.0.0.1:5000/login\\_usuario](http://127.0.0.1:5000/login_usuario)

Obtenha token de acesso. Para obtenção de token de acesso é necessário preenchimento de dados de um usuário previamente cadastrado e com status ativo.

Method	URL
POST	<a href="http://127.0.0.1:5000/login">http://127.0.0.1:5000/login</a>

Header	
Content-Type	application/json

Body	Request Body
JOSN	<pre>{   "login" : "xxx",   "senha" : "xxx" }</pre>

Status	Response Body
200 OK	<pre>{   "token de acesso":   "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJmcmVzaCI6ZmFsc2UsImIhdC6MTczOTYwMDI1OCwianRpIjoib3ZTY1ZjU0MTVhMC00ODNkLTk4MmUtNjQwNTImZGJhYTQ5IiwidHlwZSI6ImFjY2VzcyIsbnN1Yil6IjEiLCJuYmYiOiE3Mzk2MDAyNTgsImNzcmYiOiJkNjAzNjgwYS03ZjE4LTRiZmUtODVhMi01YzFhOGE5ZWU1ZjYiLCJleHAiOiE3Mzk2MDE5NTh9.EhuW7v_OiMSh_yNKAneF-WCtd4fOnrNAC76-uTmSM" }</pre>

Status	Response Body
400 BAD REQUEST	<pre>{   "mensagem": "Usuário ou senha errado." }</pre> <p><b>OU</b></p> <pre>{   "message": "Failed to decode JSON object: Expecting value: line 1 column 1 (char 0)" }</pre>

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
415 UNSUPPORTED MEDIA TYPE	<pre>{   "message": "Did not attempt to load JSON data because the request Content-Type was not 'application/json'." }</pre>

## Requisição

**POST:** [http://127.0.0.1:5000/logout\\_usuario](http://127.0.0.1:5000/logout_usuario)

Finalize token de acesso do usuário e envie para uma lista negra (BLACKLIST), para o mesmo não ser usado novamente.

Method	URL
POST	http://127.0.0.1:5000/logou

Header	
Authorization	Bearer SUA_SENHA

Status	Response Body
200 OK	{ "mensagem": "Saiu do login com sucesso!!!" }

Status	Response Body
401 UNAUTHORIZED	{ "msg": "Missing Authorization Header" }  OU  { "msg": "Token has expired" }

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
422 UNPROCESSABLE ENTITY	{ "msg": "Not enough segments" }

## Requisição

**GET:** <http://127.0.0.1:5000/usuarios/id>

Obtenha dados de um usuário cadastrado apontado por "id" enviado por URL. Exp:

[http://127.0.0.1:5000/usuarios/usuario\\_id](http://127.0.0.1:5000/usuarios/usuario_id)

Method	URL
GET	<a href="http://127.0.0.1:5000/usuarios/id:int">http://127.0.0.1:5000/usuarios/id:int</a>

Status	Response Body
200 OK	<pre>{   "usuario_id": 5,   "login": "xxx",   "email": "xxx@xxx.com",   "ativado": false }</pre>

Status	Response Body
400 BAD REQUEST	<pre>{   "message": "Hotel não existe." }</pre>

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**GET:** [http://127.0.0.1:5000/senha\\_usuarios](http://127.0.0.1:5000/senha_usuarios)

Obtenha todos os dados de usuários cadastrados, **este recurso é apenas para fins didáticos**, devido à exposição de dados sensível armazenados no Banco de Dados.

Method	URL
GET	<a href="http://127.0.0.1:5000/senha_usuarios">http://127.0.0.1:5000/senha_usuarios</a>

Status	Response Body
200 OK	<pre>{   "usuario": [     {       "usuario_id": 3,       "login": "qqq",       "senha": "qqq",       "email": "qqq@qqq.com",       "ativado": true     },     {       "usuario_id": 4,       "login": "aaa",       "senha": "aaa",       "email": "decioagu@gmail.com",       "ativado": true     },     {       "usuario_id": 5,       "login": "xxx",       "senha": "xxx",       "email": "xxx@xxx.com",       "ativado": false     }   ] }</pre>

Status	Response Body
400 BAD REQUEST	<pre>{   "message": "Hotel não existe." }</pre>

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**DELETE:** <http://127.0.0.1:5000/usuarios/id>

Exclua do Banco de Dados cadastro de usuários apontado por “id” enviado por URL. Exp:

**http://127.0.0.1:5000/usuarios/usuario\_id**

Para exclusão de usuários cadastrado no Banco de Dados é necessário possuir um login de usuário ativo para obtenção de token de acesso.

Method	URL
DELETE	http://127.0.0.1:5000/usuarios/id:int

Header	
Authorization	Bearer SUA_SENHA

Status	Response Body
200 OK	{ "mensagem ": "Usuário deletado." }

Status	Response Body
400 BAD REQUEST	{ “mensagem ": "Usuario já existe." }

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
401 UNAUTHORIZED	{ "msg": "Missing Authorization Header" } OU { "msg": "Token has expired" } OU

	<pre>{   "msg": "Missing 'Bearer' type in 'Authorization' header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>
--	---

Status	Response Body
422 UNPROCESSABLE ENTITY	<pre>{   "msg": "Bad Authorization header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>

Requisição

GET: <http://127.0.0.1:5000/sites>

Obtenha lista de sites cadastrados no Banco de Dados.

Method	URL
GET	http://127.0.0.1:5000/sites

Status	Response Body
200 OK	<pre>{   "sites": [     {       "site_id": 1,       "url": "www.meu_sites_01.com",       "hoteis": [         {           "hotel_id": "bbb",           "nome": "zzz",           "estrelas": 4.8,           "diaria": 102.57,           "cidade": "Minas Gerais",           "site_id": 1         }       ]     }   ] }</pre>



Status	Response Body
400 BAD REQUEST	<pre>{   "mensagem": "Site não existe" }</pre>

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**GET:** <http://127.0.0.1:5000/sites/url>

Obtenha dados de sites específicos com endereço de “url” cadastrados enviados por URL. Exp:  
<http://127.0.0.1:5000/sites/url>

Method	URL
GET	<a href="http://127.0.0.1:5000/sites/url:str">http://127.0.0.1:5000/sites/url:str</a>

Status	Response Body
200 OK	<pre>{   "sites": [     {       "site_id": 1,       "url": "www.meu_sites_01.com",       "hoteis": [         {           "hotel_id": "bbb",           "nome": "zzz",           "estrelas": 4.8,           "diaria": 102.57,           "cidade": "Minas Gerais",           "site_id": 1         }       ]     }   ] }</pre>

Status	Response Body
400 BAD REQUEST	{ "mensagem": "Site não existe" }

Status	Response Body
404 NOT FOUND	Not Found

## Requisição

**POST:** <http://127.0.0.1:5000/sintes/url>

Obtenha cadastrando de um novo site apontado por “url” envio por URL. Exp: **http://127.0.0.1:5000/hoteis/url**  
Para cadastramento de usuários no Banco de Dados é necessário possuir um login de usuário ativo para obtenção de token de acesso.

Method	URL
POST	http://127.0.0.1:5000/hoteis/url:str

Header	
Authorization	Bearer SUA_SENHA

Status	Response Body
201 CREATED	{ "site_id": 1, "url": "www.meu_sites_01.com", "hoteis": [ ] }

Status	Response Body
400 BAD REQUEST	{ "mensagem": "Site já existe" }

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
401 UNAUTHORIZED	<pre>{   "msg": "Missing Authorization Header" } OU {   "msg": "Token has expired" } OU {   "msg": "Missing 'Bearer' type in 'Authorization' header. Expected 'Authorization: Bearer &lt;JWT&gt;" }</pre>

Status	Response Body
422 UNPROCESSABLE ENTITY	<pre>{   "msg": "Invalid header string: 'utf-8' codec can't decode byte 0x88 in position 6: invalid start byte" }</pre>

Status	Response Body
500 SERVER ERROR	<pre>{ 'mensagem': 'Ocorreu um erro interno' }</pre>

## Requisição

**DELETE:** <http://127.0.0.1:5000/sites/url>

Exclua do Banco de Dados cadastro de site apontado por “url” enviado por URL. Exp:

**http://127.0.0.1:5000/usuarios/url**

Para exclusão de usuários cadastrado no Banco de Dados é necessário possuir um login de usuário ativo para obtenção de token de acesso.

**OBS:** Hoteis vinculado ao site **excluído serão excluídos permanentemente** do Banco de dados.

Method	URL
DELETE	http://127.0.0.1:5000/usuarios/ <b>url:str</b>

Header	
Authorization	Bearer <b>SUA_SENHA</b>

Status	Response Body
200 OK	{ "mensagem": "Site deletado" }

Status	Response Body
400 BAD REQUEST	{ "mensagem": "Site não existe" }

Status	Response Body
404 NOT FOUND	Not Found

Status	Response Body
401 UNAUTHORIZED	{ "msg": "Missing Authorization Header" }  <b>OU</b>  { "msg": "Token has expired" }  <b>OU</b>

	<pre>{   "msg": "Missing 'Bearer' type in 'Authorization' header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>
--	---

Status	Response Body
422 UNPROCESSABLE ENTITY	<pre>{   "msg": "Bad Authorization header. Expected 'Authorization: Bearer &lt;JWT&gt;'" }</pre>

Status	Response Body
500 SERVER ERROR	<pre>{ 'mensagem': 'Ocorreu um erro interno ao deletar o site' }</pre>

## Requisição

GET: [http://127.0.0.1:5000/ativacao\\_usuario/id](http://127.0.0.1:5000/ativacao_usuario/id)

Faça ativação forçada em e-mail de usuários cadastrado no Banco de Dados por “id” enviado por URL. Exp:  
[http://127.0.0.1:5000/ativacao\\_usuario/id](http://127.0.0.1:5000/ativacao_usuario/id)

Method	URL
GET	<a href="http://127.0.0.1:5000/sites/id:int">http://127.0.0.1:5000/sites/id:int</a>

Status	Response Body
200 OK	<p>Obrigado!</p> <p><b><i>O email "xxx@xxx.com" foi confirmado com sucesso! Seu usuário "xxx" já está ativo.</i></b></p>

Status	Response Body
400 BAD REQUEST	<pre>{   "mensagem": "Usuário com id:9 não foi encontrado" }</pre>

Status	Response Body
404 NOT FOUND	Not Found