

## API INDICAÇÃO REALIZADOS VIA POSTMAN

### # TAREFAS

IMPLEMENTAÇÃO DA API COM FRAMEWORK (LARAVEL)  
CRIAÇÃO DA ESTRUTURA DO BANCO DE DADOS (POSTGRESQL)  
VALIDAÇÃO DE CPF E EMAIL  
CPF NÃO PODE SER INDICADO MAIS DE UMA VEZ

### # ESTRUTURA DO BANCO

INDICAÇÕES: ID, NOME, CPF, TELEFONE, EMAIL, STATUS\_ID  
STATUS DAS INDICAÇÕES: ID, DESCRIÇÃO [1. INICIADA / 2. FINALIZADA / 3. EM PROCESSO]

### # FUNCIONALIDADES

LISTAR  
INCLUIR (STATUS INICIAL = INICIADA)  
EVOLUIR STATUS (precisa seguir o fluxo Iniciada -> Em processo -> Finalizada)

---

### # Ferramentas usadas

- Aptana Studio
- Bitnami (integrados apache 2.0, php versão 7.3.10, postgresql versão 9.5, laravel versão 10.15.0)
- GitHub
- Git
- Postman (para os testes)

---

### # Estrutura de banco de dados

```
CREATE TABLE public.indicacao (  
  id bigint NOT NULL DEFAULT nextval('indicacao_id_seq'::regclass),  
  nome character varying(80) COLLATE pg_catalog."default" NOT NULL,  
  cpf character varying COLLATE pg_catalog."default" NOT NULL,  
  telefone character varying COLLATE pg_catalog."default",  
  email character varying(80) COLLATE pg_catalog."default",  
  status_id integer,  
  CONSTRAINT indicacao_pkey PRIMARY KEY (id),  
  CONSTRAINT indicacao_status_id_fkey FOREIGN KEY (status_id)  
    REFERENCES public.indicacaostatus (id) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION  
)
```

```
CREATE TABLE public.indicacaostatus (  
  id integer NOT NULL DEFAULT nextval('indicacaostatus_id_seq'::regclass),  
  descricao character varying(20) COLLATE pg_catalog."default" NOT NULL,  
  CONSTRAINT indicacaostatus_pkey PRIMARY KEY (id)  
)
```

## # Nome da API

api-indicacao

## # Classes da API (pasta api-indicacao\app\http\Controllers)

IndicacaoController - Manter dados da indicação

ValidationController – Métodos de validação de dados da indicação

## # Rotas

```
Route::get('/indicacao', 'IndicacaoController@indicacao');
```

```
Route::post('/indicacao', 'IndicacaoController@salvar');
```

```
Route::put('/indicacao', 'IndicacaoController@status');
```

```
Route::get('/showindicacao/{id}', 'IndicacaoController@showindicacao');
```

```
Route::get('/showindicacaostatus/{id}', 'IndicacaoController@showindicacaostatus');
```

## # Testes realizados no POSTMAN

### # Teste listando dados

The screenshot shows the Postman interface with a GET request to `http://localhost:8000/api/indicacao`. The response is a JSON array of two objects, each representing an indication with fields like id, nome, cpf, telefone, email, and status\_id.

**Request Details:**

- Method: GET
- URL: `http://localhost:8000/api/indicacao`

**Response Details:**

- Status: 200 OK
- Time: 144 ms
- Size: 1.03 KB

**Response Body (JSON):**

```
[{"id": 1, "nome": "CAMILLE CASAS", "cpf": "95011013715", "telefone": "21999233829", "email": "camillecacasas@gmail.com", "status_id": 1}, {"id": 2, "nome": "Rosane", "cpf": "95011013715", "telefone": "111", "email": "rosane@gmail.com", "status_id": 2}]
```

## # Teste incluindo dados

OverviewPUT Get daPOST Post cGET selectGET Get daGET GET dzGET GET sh+No Environment

RESTful API basics: CRUD, test & variable / Post dataSave

POSThttp://localhost:8000/api/indicacaoSend

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettingsCookies

noneform-datax-www-form-urlencodedorawbinaryGraphQL

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	nome	Fabiano Moreira			
<input checked="" type="checkbox"/>	cpf	17859232006			
<input checked="" type="checkbox"/>	email	fabiano@gmail.com			
<input checked="" type="checkbox"/>	telefone	(21)98988585			
	Key	Value	Description		

BodyCookiesHeaders (9)Test Results (1/1)Status: 200 OKTime: 279 msSize: 311 BSave as Example

PrettyRawPreviewVisualizeJSON

```
1
2  "success": "Indicacao cadastrada."
3
```

## # Teste incluindo dados - Validação de CPF válido

OverviewPUT Get daPOST Post cGET selectGET Get daGET GET dzGET GET sh+No Environment

RESTful API basics: CRUD, test & variable / Post dataSave

POSThttp://localhost:8000/api/indicacaoSend

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettingsCookies

noneform-datax-www-form-urlencodedorawbinaryGraphQL

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	nome	Renata Vieira			
<input checked="" type="checkbox"/>	cpf	78898787			
<input checked="" type="checkbox"/>	email	renata@gmail.com			
<input checked="" type="checkbox"/>	telefone	(21)98987777			
	Key	Value	Description		

BodyCookiesHeaders (9)Test Results (1/1)Status: 200 OKTime: 1137 msSize: 318 BSave as Example

PrettyRawPreviewVisualizeJSON

```
1
2  "msg": "Erro: CPF invalido.",
3  "error": true
4
```

# Teste incluindo dados - Validação de E-MAIL válido

OverviewPUT Get daPOST Post cGET selectGET Get daGET GET dGET GET sh+...No Environment

RESTful API basics: CRUD, test & variable / Post data

Save

POSThttp://localhost:8000/api/indicacao

Send

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettingsCookies

noneform-datax-www-form-urlencodedrawbinaryGraphQL

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	nome	Renata Vieira			
<input checked="" type="checkbox"/>	cpf	38116848097			
<input checked="" type="checkbox"/>	email	renata@gmail			
<input checked="" type="checkbox"/>	telefone	(21)98987777			
	Key	Value	Description		

BodyCookiesHeaders (9)Test Results (1/1)Status: 200 OKTime: 290 msSize: 320 BSave as Example...

PrettyRawPreviewVisualizeJSON

```
1
2  "msg": "Erro: Email invalido.",
3  "error": true
4
```

# Teste incluindo dados - Validação de CPF existente (CPF NÃO PODE SER INDICADO MAIS DE UMA VEZ)

OverviewPUT Get daPOST Post cGET selectGET Get daGET GET dGET GET sh+...No Environment

RESTful API basics: CRUD, test & variable / Post data

Save

POSThttp://localhost:8000/api/indicacao

Send

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettingsCookies

noneform-datax-www-form-urlencodedrawbinaryGraphQL

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	nome	Aline Casas			
<input checked="" type="checkbox"/>	cpf	76515240716			
<input checked="" type="checkbox"/>	email	alinecasas@gmail.com			
<input checked="" type="checkbox"/>	telefone	(21)99998888			
	Key	Value	Description		

BodyCookiesHeaders (9)Test Results (1/1)Status: 200 OKTime: 145 msSize: 320 BSave as Example...

PrettyRawPreviewVisualizeJSON

```
1
2  "msg": "Erro: CPF cadastrado.",
3  "error": true
4
```

# Teste de put do status, validando o fluxo respeita a evolução

OverviewPUT Get daPOST Post cGET selectGET Get daGET GET dGET GET sh+No Environment

RESTful API basics: CRUD, test & variable / Get data

Save

PUT

http://localhost:8000/api/indicacao

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

<input type="checkbox"/>	cpr	95011013/13	
<input type="checkbox"/>	email	rosanecasas@gmail.com	
<input type="checkbox"/>	telefone	111	
<input checked="" type="checkbox"/>	status_id	2	
<input checked="" type="checkbox"/>	id	7	
	Key	Value	Description

Body

Cookies

Headers (9)

Test Results (1/1)

Status: 200 OK

Time: 159 ms

Size: 308 B

Save as Example

Pretty

Raw

Preview

Visualize

JSON

1

2

3

"success": "Status Atualizado."

OverviewPUT Get daPOST Post cGET selectGET Get daGET GET dGET GET sh+No Environment

RESTful API basics: CRUD, test & variable / Get data

Save

PUT

http://localhost:8000/api/indicacao

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

<input type="checkbox"/>	cpr	95011013/13	
<input type="checkbox"/>	email	rosanecasas@gmail.com	
<input type="checkbox"/>	telefone	111	
<input checked="" type="checkbox"/>	status_id	3	
<input checked="" type="checkbox"/>	id	6	
	Key	Value	Description

Body

Cookies

Headers (9)

Test Results (1/1)

Status: 200 OK

Time: 363 ms

Size: 336 B

Save as Example

Pretty

Raw

Preview

Visualize

JSON

1

2

3

"msg": "Erro: Status inválido não segue o fluxo."