

Documento de Teste - Teste API da aplicação Mub

Segue abaixo uma lista como as funcionalidades que serão testadas na API da aplicação MUB. Onde as funcionalidades citadas abaixo estão divididas em **casos de testes positivo e negativo**, ou seja, os casos de testes positivos reference as funcionalidades que seguem o fluxo principal de forma correta. No entanto, os casos de testes negativos reference a uma exceção ou falha no percurso do fluxo principal.

- **Manter Usuário :**

- Realizar Cadastro - Positivo

O Realizar Cadastro se deu da seguinte maneira, o usuário preenche os campos com seus dados:

Email: ["ruanrodriguesgmail.com"]

Telefone: ["(88) 9.8819-6854"]

CPF: ["079.276.153-78"]

Nome: ["Ruan Rodrigues Lira"]

Senha: ["10203040a"]

DataDeNascimento: ["18/08/1998"]

E o sistema retorna seu dados confirmando que foi inserido, juntamente com o ID

Segue prints do testlink

caso de teste realizar cadastro positivo				
<u>Pré-condições:</u>				
ter cpf válido				
#:	Ações do Passo:	Resultados Esperados::	Notas da Execução:	Status da Execução:
1	O usuário clica em "Cadastre-se"	O sistema retorna para a tela de cadastro		Passou
Anexos da Execução				
• Api 1.PNG				
<pre>{ "email": "ruanrodriguesgmail.com", "telefone": "(88) 9.8819-6854", "cpf": "079.276.153-78", "nome": "Ruan Rodrigues Lira", "senha": "10203040a", "dataDeNascimento": "18/08/1998" }</pre>				

2	<p>O usuário preenche os campos</p> <p>Email: [ruanrodriguesgmail.com] Telefone: [(88) 9.8819-6854] CPF: [079.276.153-78] Nome: [Ruan Rodrigues Lira] Senha: [10203040a] DataDeNascimento: [18/08/1998]</p>	<p>O sistema valida se todos os campos estão preenchidos corretamente e retorna:</p> <pre> { "id": 3, "cpf": "079.276.153-78", "nome": "Ruan Rodrigues Lira", "senha": "10203040a", "contatoUsuario": { "id": 3, "email": "ruanrodriguesgmail.com", "telefone": "(88) 9.8819-6854" }, "statusUsuario": { "id": 1, "nome": "Ativo" } } </pre>	Passou
---	--	--	--------

Anexos da Execução

• Api 2.PNG

```

1  {
2    "id": 3,
3    "cpf": "079.276.153-78",
4    "nome": "Ruan Rodrigues Lira",
5    "senha": "10203040a",
6    "contatoUsuario": {
7      "id": 3,
8      "email": "ruanrodriguesgmail.com",
9      "telefone": "(88) 9.8819-6854"
10   },
11   "statusUsuario": {
12     "id": 1,
13     "nome": "Ativo"
14   }
15 }

```

Tipo de Execução:	Manual
Estimativa da duração da Execução (min):	10.00
Prioridade:	Alto
Requisitos	RF-001 - Cadastrar usuário: cadastro de usuário [Versão: 1]
Palavras-chave:	Nenhum
Detalhes da Execução	
Baseline	Baseline Cadastro de usuario - API
Testador	admin
Resultado da Execução:	Passou
Modo de Execução:	Manual
Duração da Execução (min):	10.00

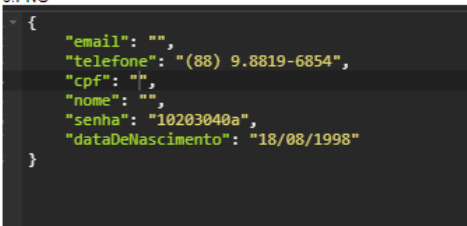
Opções


	usuario_id	usuario_cpf	usuario_nome	usuario_senha	usuario_contato_id	usuario_status_id
<input type="checkbox"/> Editar Copiar Apagar	1	NULL	Hiago M. Fernandes	1020304050	1	1
<input type="checkbox"/> Editar Copiar Apagar	2	064.939.544-10	Hiago M. Fernandes	1020304050	2	1
<input type="checkbox"/> Editar Copiar Apagar	3	079.276.153-78	Kelvin	1020304050	3	1
<input type="checkbox"/> Editar Copiar Apagar	4	079.276.153-78	Ruan Rodrigues Lira	10203040a	4	1
<input type="checkbox"/> Editar Copiar Apagar	5	079.276.153-78	Ruan Rodrigues Lira	10203040a	5	1

Realizar Cadastro - Negativo

O usuário digita alguns de seus dados para cadastro com o intuito de dar erro
 Email:[" "]

Telefone: ["(88) 9.8819-6854"]
CPF: [" "]
Nome: [" "]
Senha: ["10203040a"]
DataDeNascimento: ["18/08/1998"]

Caso de Teste MUB-2: CT-002 - Realizar Cadastro - Negativo [Versão : 1]				
Autor:		admin		
Objetivo do Teste::				
caso de teste realizar cadastro negativo				
Pré-condições:				
ter cpf válido				
#:	Ações do Passo:	Resultados Esperados::	Notas da Execução:	Status da Execução:
1	O usuário clica no botão "Cadasta-se"	O sistema redireciona para a tela de cadastro		Com Falha
Anexos da Execução				
• 3.PNG				
				
2	O usuário preenche os campos Email: [""] Telefone: ["(88) 9.8819-6854"] CPF: [""] Nome: [""] Senha: ["10203040a"] DataDeNascimento: ["18/08/1998"]	O sistema retorna: "Falta preencher os campos" e o mantém na tela de cadastro		Com Falha

Anexos da Execução	
• 4.PNG	
	
Tipo de Execução:	Manual
Estimação da duração da Execução (min):	10.00
Prioridade:	Alto
Requisitos	RF-001 - Cadastrar usuário: cadastro de usuário [Versão: 1]
Palavras-chave:	Nenhum
Detalhes da Execução	
Baseline	Baseline Cadastro de usuario - API
Testador	admin
Resultado da Execução:	Com Falha
Modo de Execução:	Manual
Duração da Execução (min):	

O teste foi concluído Com Falha, porque o sistema mesmo com dados faltando ele efetuou o cadastro;

Opções		usuario_id	usuario_cpf	usuario_nome	usuario_senha	usuario_contato_id	usuario_status_id
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	1	NULL	Hiago M. Fernandes	1020304050	1	1
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	2	064.939.544-10	Hiago M. Fernandes	1020304050	2	1
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	3	079.276.153-78	Kelvin	1020304050	3	1
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	4	079.276.153-78	Ruan Rodrigues Lira	10203040a	4	1
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	5	NULL		10203040a	5	1

- Alterar Usuário - Positivo

O usuário seleciona o ID do cadastro que deseja ser alterado, no caso estou alterando o cadastro de ID = 2

```
{
  "email": "hiagomfernandes@gmail.com",
  "telefone": "(88) 9.9953-1410",
  "nome": "Hiago Fernandes",
  "cpf": "064.939.544-10",
  "senha": "teste2030",
  "dataDeNascimento": "10/10/2000"
}
```

Opções		usuario_id	usuario_cpf	usuario_nome	usuario_senha	usuario_contato_id	usuario_status_id
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	1	NULL	Hiago M. Fernandes	1020304050	1	1
<input type="checkbox"/>	✎ Editar ✎ Copiar ✖ Apagar	2	064.939.544-10	Hiago Fernandes	teste2030	2	1

- Alterar Usuário - Negativo

O usuário busca um ID inexistente e preenche os campos:
O ID em questão é o ID - 10

Alterar Usuário

PUT

http://localhost:8080/ApiMub/user/alterar/10

Params Authorization Headers (9) Body Pre-request Script

none form-data x-www-form-urlencoded raw bin

```
1 {
2   "email": "hiagomfernandes@gmail.com",
3   "telefone": "(88) 9.9953-1410",
4   "nome": "Hiago Fernandes",
5   "cpf": "064.939.544-10",
6   "senha": "teste2030",
7   "dataDeNascimento": "10/10/2000"
8 }
```

O sistema retorna:

```
{
  "timestamp": "2019-10-15T02:10:01.215+0000",
  "status": 500,
  "error": "Internal Server Error",
  "message": "Objeto não encontrado! Id: 10, Tipo: com.voxelsbrasil.apimub.domains.Usuario",
  "path": "/ApiMub/user/alterar/10"
}
```

- Buscar Usuário - Positivo
O usuário

O usuário busca por um ID existente, no caso o ID em questão é o 1

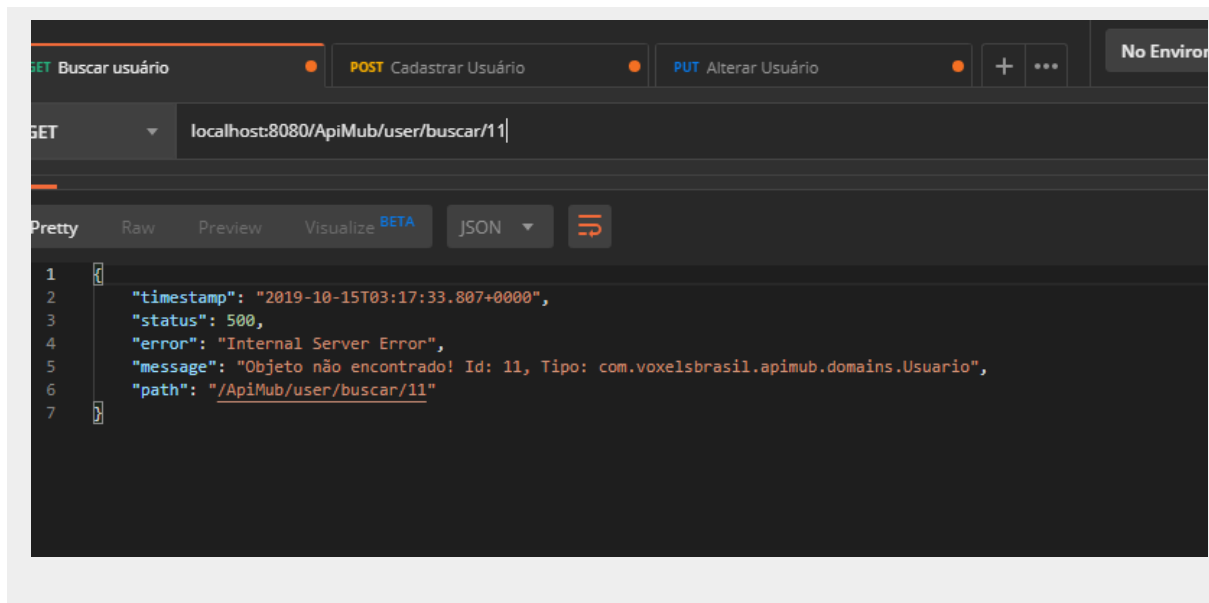
GET localhost:8080/ApiMub/user/buscar/1

Pretty Raw Preview Visualize BETA JSON

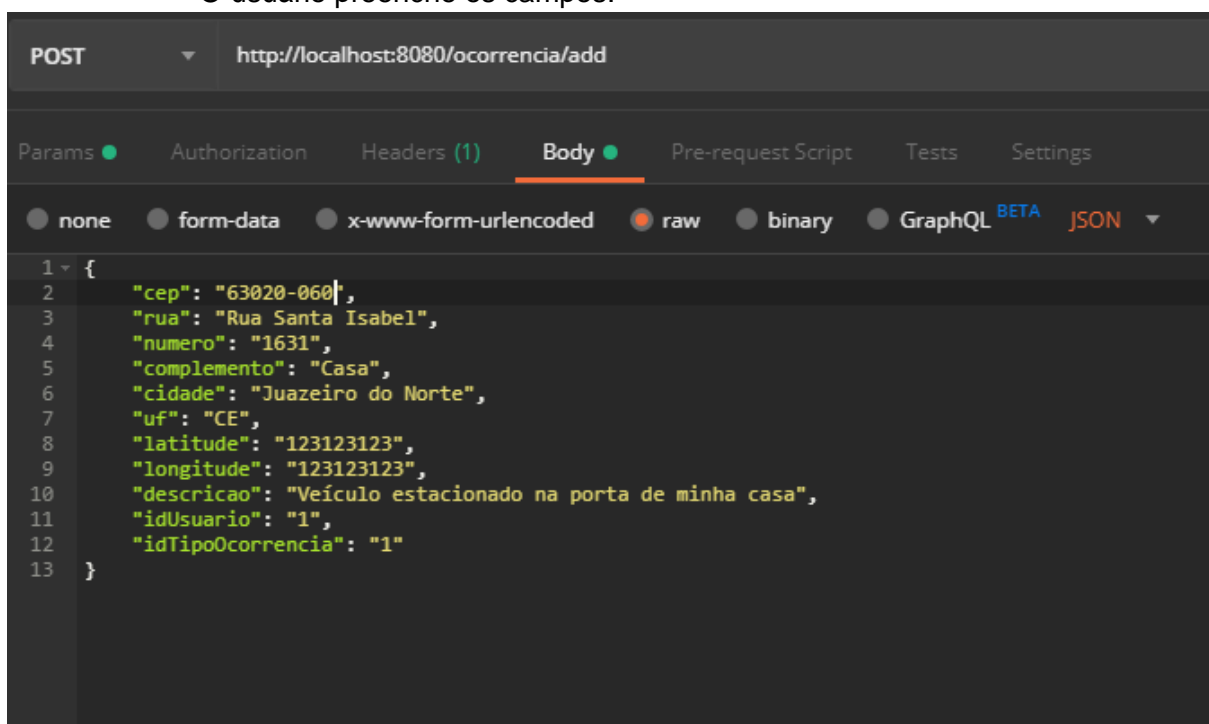
```
1 {
2   "id": 1,
3   "cpf": null,
4   "nome": "Hiago M. Fernandes",
5   "senha": "1020304050",
6   "contatoUsuario": {
7     "id": 1,
8     "email": "hiagomfernandes@gmail.com",
9     "telefone": "(88) 9.9953-1410"
10  },
11  "statusUsuario": {
12    "id": 1,
13    "nome": "Ativo"
14  }
15 }
```

- Buscar Usuário - Negativo

O usuário busca por um ID inexistente que no caso é o ID = 11



- Desativar Usuário - Positivo
O usuário 1 foi desativo da base
- Desativar Usuário - Negativo
O usuário 11 não foi desativado da base, pois o cadastro 11 não existe.
 - Ocorrências
 - Inserir Ocorrência - Positivo
O usuário preenche os campos:



SELECT * FROM `ocorrencia`

Perfil [Editar em linha] [Edita] [Explicar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de registros: 25 | Filtrar registros: Pesquisar esta tabela | Ordenar por chave: Nenhum

Opções

	ocorrencia_id	ocorrencia_descricao	ocorrencia_protocolo	ocorrencia_atividade	ocorrencia_endereco_id	ocorrencia_localizac
<input type="checkbox"/> Editar Copiar Apagar	2	Veículo estacionado na porta de minha casa	JUAZEIRO DO NORTETUE OCT 15 18:55:07 BRT 2019	1		1

☐ Marcar todos
 Com os seleccionados: Editar Copiar Apagar Exportar

- Alterar Ocorrência - Positivo

O usuário seleciona o ID da ocorrência que deseja ser alterada e preenche os campos de acordo com a mudança:

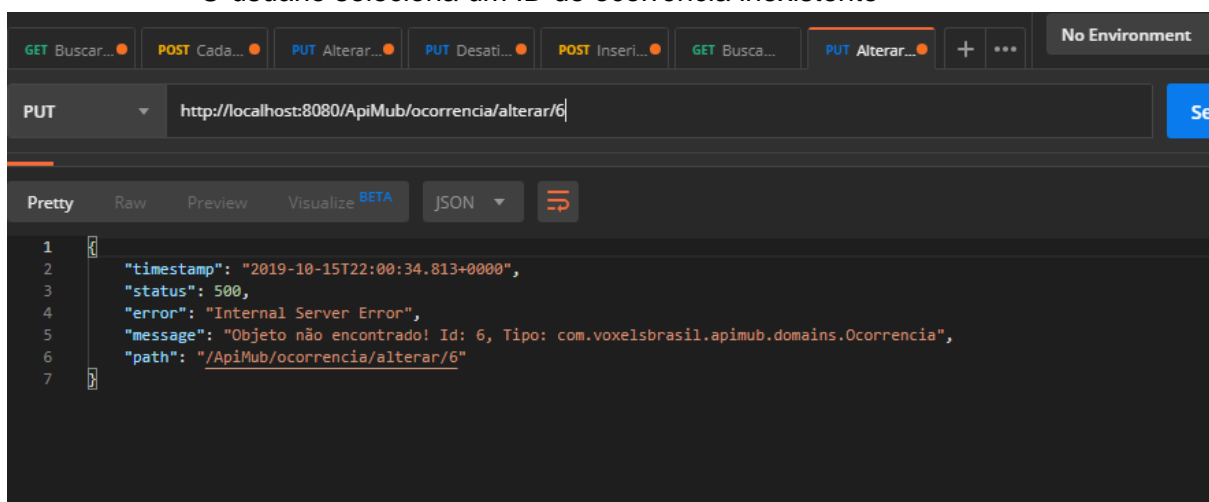
```
{
  "id": 2,
  "descricao": "Alteração OK",
  "protocolo": "JUAZEIRO DO NORTETUE OCT 15 18:55:07 BRT 2019",
  "statusDeAtividade": true,
  "localizacaoOcorrencia": {
    "id": 1,
    "latitude": "00000000",
    "longitude": "00000000"
  },
  "tipoOcorrencia": {
    "id": 2,
    "nome": "Acidente"
  },
  "enderecoOcorrencia": {
    "id": 1,
    "cep": "63020-060",
    "rua": "testeeeeeeeeee",
    "numero": "2424",
    "complemento": "teste",
    "cidade": "RIO BRANCO",
    "uf": "AC"
  },
  "usuario": {
    "id": 1,
    "cpf": null,
    "nome": "Teste",
    "email": "teste@teste.com",
    "password": "12345678"
  }
}
```

```

},
"enderecoOcorrencia": {
  "id": 1,
  "cep": "63020-060",
  "rua": "testeeeeeeeeee",
  "numero": "2424",
  "complemento": "teste",
  "cidade": "RIO BRANCO",
  "uf": "AC"
},
"usuario": {
  "id": 1,
  "cpf": null,
  "nome": "Hiago M. Fernandes",
  "senha": "1020304050",
  "contatoUsuario": {
    "id": 1,
    "email": "hiagomfernandes@gmail.com",
    "telefone": "(88) 9.9953-1410"
  },
  "statusUsuario": {
    "id": 1,
    "nome": "Ativo"
  }
}
}

```

- Alterar Ocorrência - Negativo
O usuário seleciona um ID de ocorrência inexistente



- Buscar Ocorrência - Positivo
O usuário busca o ID da ocorrência que deseja visualizar ex ID = 2;


```
GET http://localhost:8080/ApiMub/ocorrencia/buscar/2

Pretty Raw Preview Visualize BETA JSON

{
  "id": 2,
  "descricao": "Alteração OK",
  "protocolo": "JUAZEIRO DO NORTE TUE OCT 15 18:55:07 BRT 2019",
  "statusDeAtividade": true,
  "localizacaoOcorrencia": {
    "id": 1,
    "latitude": "00000000",
    "longitude": "00000000"
  },
  "tipoOcorrencia": {
    "id": 2,
    "nome": "Acidente"
  },
  "enderecoOcorrencia": {
    "id": 1,
    "cep": "63020-060",
    "rua": "testeeeeeeeeee",
    "numero": "2424",
    "complemento": "teste",
    "cidade": "RIO BRANCO",
    "uf": "AC"
  },
  "usuario": {
    "id": 1,
    "cpf": null,
    "nome": "..."
  }
}
```

- Buscar Ocorrência - Negativo
O usuário busca por um ID de ocorrência inexistente, ex ID = 10;

```
GET http://localhost:8080/ApiMub/ocorrencia/buscar/10

Pretty Raw Preview Visualize BETA JSON

{
  "timestamp": "2019-10-15T22:07:08.335+0000",
  "status": 500,
  "error": "Internal Server Error",
  "message": "Objeto não encontrado! Id: 10, Tipo: com.voxelsbrasil.apimub.domains.Ocorrencia",
  "path": "/ApiMub/ocorrencia/buscar/10"
}
```

- Desativar Ocorrência - Positivo
O usuário busca a ocorrência desejada em desativar pelo ID, no caso do exemplo Id = 2

```
1 {
2   "id": 2,
3   "descricao": "Alteração OK",
4   "protocolo": "JUAZEIRO DO NORTE TUE OCT 15 18:55:07 BRT 2019",
5   "statusDeAtividade": false,
6   "localizacaoOcorrencia": {
7     "id": 1,
8     "latitude": "00000000",
9     "longitude": "00000000"
10  },
11  "tipoOcorrencia": {
12    "id": 2,
13    "nome": "Acidente"
14  },
15  "endereçoOcorrencia": {
16    "id": 1,
17    "cep": "63020-060",
18    "rua": "testeeeeeeeeee",
19    "numero": "2424",
20    "complemento": "teste",
21    "cidade": "RIO BRANCO",
22    "uf": "AC"
23  },
24  "usuario": {
25    "id": 1,
26    "cpf": null,
```

- Desativar Ocorrência - Negativo
O usuário busca por um ID inexistente de ocorrência para desativar. ex:ID-10

```
1 {
2   "timestamp": "2019-10-15T22:10:14.262+0000",
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "Objeto não encontrado! Id: 10, Tipo: com.voxelsbrasil.apimub.domains.Ocorrencia",
6   "path": "/ApiMub/ocorrencia/desativar/10"
7 }
```

- Departamento
- Inserir Departamento - Positivo
O usuário do órgão municipal de trânsito preenche os dados conforme solicitado;

```

1 {
2   "idUsuarioResponsavel" : "1",
3   "email" : "demutran@juanorte.com",
4   "telefone" : "8888.8888",
5   "cnpj" : "0001",
6   "nomeEmpresarial" : "Departamento de Transito Jua Norte",
7   "nomeFantasia" : "Demutran Juazeiro do Norte",
8   "efr" : "Não lembro",
9   "status": true,
10  "cep": "63020-000",
11  "rua": "Av. Castelo Branco",
12  "numero": "1000",
13  "complemento": "Prédio Comercial"
14 }

```

```

1 {
2   "id": 1,
3   "endereco": {
4     "id": 1,
5     "cep": "63020-000",
6     "rua": "Av. Castelo Branco",
7     "numero": "1000",
8     "complemento": "Prédio Comercial",
9     "cidade": null,
10    "uf": null
11  },
12  "contato": {
13    "id": 1,
14    "email": "demutran@juanorte.com",
15    "telefone": "8888.8888"
16  },
17  "usuarioResponsavel": {
18    "id": 1,
19    "cpf": null,
20    "nome": "Hiago M. Fernandes",
21    "senha": "1020304050",
22    "contatoUsuario": {
23      "id": 1,
24      "email": "hiagomfernandes@gmail.com",
25      "telefone": "(88) 9.9953-1410"
26    }
27  }
28 }

```

- Buscar departamento - Positivo
O órgão municipal de trânsito busca o departamento de acordo com seu ID, exemplo de ID = 1

```
GET localhost:8080/ApiMub/departamento/buscar-informacoes/1

Pretty Raw Preview Visualize BETA JSON

{
  "id": 1,
  "endereco": {
    "id": 1,
    "cep": "63020-000",
    "rua": "Av. Castelo Branco",
    "numero": "1000",
    "complemento": "Prédio Comercial",
    "cidade": null,
    "uf": null
  },
  "contato": {
    "id": 1,
    "email": "demutran@juanorte.com",
    "telefone": "8888.8888"
  },
  "usuarioResponsavel": {
    "id": 1,
    "cpf": null,
    "nome": "Hiago M. Fernandes",
    "senha": "1020304050",
    "contatoUsuario": {
      "id": 1,
      "email": "hiagomfernandes@gmail.com",
      "telefone": "(88) 9.9953-1410"
    }
  }
}
```

- Buscar departamento - Negativo
O usuário do órgão municipal de trânsito busca por um ID de departamento inexistente, exemplo id = 6

```
GET localhost:8080/ApiMub/departamento/buscar-informacoes/6 Send

Pretty Raw Preview Visualize BETA JSON

{
  "timestamp": "2019-10-15T22:21:22.521+0000",
  "status": 500,
  "error": "Internal Server Error",
  "message": "Objeto não encontrado! Id: 6, Tipo: com.voxelsbrasil.apimub.domains.DeptoTransito",
  "path": "/ApiMub/departamento/buscar-informacoes/6"
}
```

- Agente de trânsito
- Inserir Agente - Positivo
O usuário preenche os campos do agente de trânsito:

```

{
  "id": 1,
  "nvAcesso": {
    "id": 1,
    "nome": "Moderador de Trânsito"
  },
  "departamento": {
    "id": 1,
    "endereco": {
      "id": 1,
      "cep": "63020-000",
      "rua": "Av. Castelo Branco",
      "numero": "1000",
      "complemento": "Prédio Comercial",
      "cidade": null,
      "uf": null
    },
    "contato": {
      "id": 1,
      "email": "demutran@juanorte.com",
      "telefone": "8888.8888"
    },
    "usuarioResponsavel": {
      "id": 1,
      "cpf": null,
      "nome": "Hiago M. Fernandes",
      "email": "hiago@juanorte.com"
    }
  }
}

```

- Inserir Agente - Negativo

O usuário não preenche os campos, identificando assim que o usuário não é agente de trânsito

The screenshot shows a REST client interface with a POST request to `http://localhost:8080/ApiMub/agente/add`. The request body is a JSON object with empty string values for `idUsuario`, `idDepartamento`, `idNivelAcesso`, and `statusAgente`.

The response status is **500 Internal Server Error**. The response body, shown in JSON format, contains the following error details:

```

{
  "timestamp": "2019-10-15T22:29:04.252+0000",
  "status": 500,
  "error": "Internal Server Error",
  "message": "The given id must not be null; nested exception is java.lang.IllegalArgumentException: The given id must not be null!",
  "path": "/ApiMub/agente/add"
}

```

- Buscar Agente - Positivo

O órgão busca o agente de trânsito pelo ID do mesmo, por exemplo ID - 1

```
GET http://localhost:8080/ApiMub/agente/buscar-agente/1 Send Save

Pretty Raw Preview Visualize BETA JSON

1 {
2   "id": 1,
3   "nvAcesso": {
4     "id": 1,
5     "nome": "Moderador de Trânsito"
6   },
7   "departamento": {
8     "id": 1,
9     "endereco": {
10      "id": 1,
11      "cep": "63020-000",
12      "rua": "Av. Castelo Branco",
13      "numero": "1000",
14      "complemento": "Prédio Comercial",
15      "cidade": null,
16      "uf": null
17    },
18    "contato": {
19      "id": 1,
20      "email": "demutran@juvanorte.com",
21      "telefone": "8888.8888"
22    },
23    "usuarioResponsavel": {
24      "id": 1,
25      "cpf": null,
26      "nome": "Hiago M. Fernandes",
27      "email": "hiago.m.fernandes@juvanorte.com"
28    }
29  }
30 }
```

- Buscar Agente - Negativo

O usuário do órgão municipal busca um agente com ID inexistente, ID - 7

```
GET http://localhost:8080/ApiMub/agente/buscar-agente/7 Send

Pretty Raw Preview Visualize BETA JSON

1 {
2   "timestamp": "2019-10-15T22:36:40.216+0000",
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "Objeto não encontrado! Id: 7, Tipo: com.voxelsbrasil.apimub.domains.AgenteTransito",
6   "path": "/ApiMub/agente/buscar-agente/7"
7 }
```

- Ativar Agente - Positivo

O órgão municipal de trânsito busca o ID que deseja ativar, no exemplo ID - 1

```
PUT http://localhost:8080/ApiMub/agente/reactivar-agente/1 Send Save

Pretty Raw Preview Visualize BETA JSON

1 {
2   "id": 1,
3   "nvAcesso": {
4     "id": 1,
5     "nome": "Moderador de Trânsito"
6   },
7   "departamento": {
8     "id": 1,
9     "endereco": {
10      "id": 1,
11      "cep": "63020-000",
12      "rua": "Av. Castelo Branco",
13      "numero": "1000",
14      "complemento": "Prédio Comercial",
15      "cidade": null,
16      "uf": null
17    },
18    "contato": {
19      "id": 1,
20      "email": "demutran@juanorte.com",
21      "telefone": "8888.8888"
22    },
23    "usuarioResponsavel": {
24      "id": 1,
25      "cpf": null,
26      "nome": "Hiago M. Fernandes",
27    }
28  }
29 }
```

- Ativar Agente - Negativo

O órgão municipal de trânsito ativa o usuário de ID - 9, o sistema deverá retornar erro, pois o mesmo não existe.

```
PUT http://localhost:8080/ApiMub/agente/reactivar-agente/9 Send

Pretty Raw Preview Visualize BETA JSON

1 {
2   "timestamp": "2019-10-15T22:38:45.072+0000",
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "Objeto não encontrado! Id: 9, Tipo: com.voxelsbrasil.apimub.domains.AgenteTransito",
6   "path": "/ApiMub/agente/reactivar-agente/9"
7 }
```

- Desativar Agente - Positivo

O órgão busca o agente que deseja desativar pelo ID, no caso ele busca o ID - 1

```
PUT http://localhost:8080/ApiMub/agente/desativar-agente/1
Send Save

Pretty Raw Preview Visualize BETA JSON

1 {
2   "id": 1,
3   "nvAcesso": {
4     "id": 1,
5     "nome": "Moderador de Trânsito"
6   },
7   "departamento": {
8     "id": 1,
9     "endereco": {
10      "id": 1,
11      "cep": "63020-000",
12      "rua": "Av. Castelo Branco",
13      "numero": "1000",
14      "complemento": "Prédio Comercial",
15      "cidade": null,
16      "uf": null
17    },
18    "contato": {
19      "id": 1,
20      "email": "demutran@juanorte.com",
21      "telefone": "8888.8888"
22    },
23    "usuarioResponsavel": {
24      "id": 1,
25      "cpf": null,
26      "nome": "Hiago M. Fernandes",
```

- Desativar Agente - Negativo

O órgão municipal de trânsito busca por um ID inexistente, no caso o ID - 15 para desativar

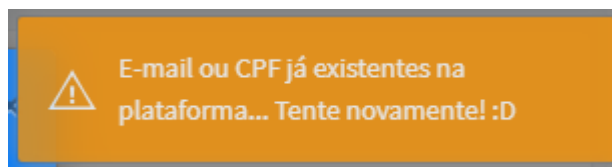
```
PUT http://localhost:8080/ApiMub/agente/desativar-agente/15
Send Save

Pretty Raw Preview Visualize BETA JSON

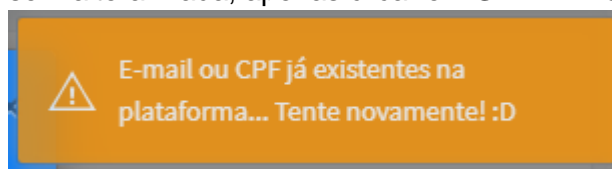
1 {
2   "timestamp": "2019-10-15T22:40:41.873+0000",
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "Objeto não encontrado! Id: 15, Tipo: com.voxelsbrasil.apimub.domains.AgenteTransito",
6   "path": "/ApiMub/agente/desativar-agente/15"
7 }
```

Relatório dos erros MUB - Plataforma Web

Após cadastrar o usuário com o e-mail ruan@gmail.com, como usuário **COMUM** e-mail inexistente nos cadastros o sistema retornou



Após selecionar alterar dados do usuário que acabou de ser cadastrado e sem alterar nada, apenas clicar em **SALVAR** o sistema apresentou



O teste acima foi feito com e-mail hiago@gmail.com, como usuário **COMUM** ele inseriu sem erro. MAS quando cliquei para alterar e não alterei nada e salvei, apresentou esse erro

localhost/AdminMub/usuarios

Dados do usuário

E-mail ou CPF já existentes na plataforma... Tente novamente! :D

Nome do usuário: Hiago Teste

CPF usuário: 716.683.767-50

Senha: 321

Data de nascimento: 31/12/1990

E-mail: hiago@gmail.com

Nível de acesso: Comum

Telefone: (99) 99999-9999

Status do usuário: Ativo

Fechar Salvar

Showing 1 to 6 of 6 entries

Previous 1 Next

Após atualizar a página e refazer o passo anterior ele salva sem erro nenhum

✓ Usuário atualizado com sucesso!

Search:

✓ Usuário sincronizado com sucesso!

STATUS DE ATIVIDADE

MAIS...

Foi inserido um novo registro com dados que não foram usados anteriormente e retornou erro, foram realizadas 7 tentativas, todas apresentaram esse erro e não inseriu.

Dados do usuário

⚠ E-mail ou CPF já existentes na plataforma... Tente novamente! :D

Nome do usuário:

Geovanny Teste

CPF usuário:

136.425.351-86



✓ Parabéns, o CPF é válido!

Senha:

321

Data de nascimento:

25/12/1992

E-mail:

ggeo@gmail.com

Nível de acesso

Administrador

Telefone:

(33) 33333-3333

Status do usuário:

Inativo

Fechar

Salvar

Search:

MAIS...

Opções

Opções

Opções

Opções

Opções

Opções