

CATALOG

Caso de Teste: Criar Estabelecimento

Para criar um estabelecimento, o usuário deverá primeiro responder se está ou não no estabelecimento. Caso não esteja, o usuário deverá informar o Nome do Estabelecimento, CEP, Rua, Cidade, Estado e número do estabelecimento (caso possua). Se o CEP for preenchido, o aplicativo preencherá automaticamente os campos de Rua, Cidade e Estado. Caso a resposta seja positiva, a latitude e longitude serão obtidas através das coordenadas do GPS, sendo preenchidos consequentemente o CEP e seus campos derivados automaticamente.

Variáveis de Entrada	Condições	Equivalência Válida	Equivalência Inválida
dtoEstabelecimento	Não pode ser nulo	dtoEstabelecimento!=null	dtoEstabelecimento=null
	Estar no formato JSON de dtoEstabelecimento	Sim	Não
dtoEstabelecimento.Nome	string com pelo menos 3 caracteres	t>=3	t<3
	string com primeiro caractere sendo uma letra	Sim	Não
	string com no máximo 100 caracteres	t<=100	t>100
dtoEstabelecimento.CEP	Inteiro de 8 dígitos	t==8	t!=0 && t!=8
	Inteiro de 0 dígitos	t==0	
dtoEndereçoEstabelecimento.Rua	string com pelo menos 3 caracteres	t>=3	t<3
	string com no máximo 100 caracteres	t<=100	t>100
dtoEndereçoEstabelecimento.Cidade	string com pelo menos 3 caracteres	t>=3	t<3

	string com no máximo 100 caracteres	$t \leq 100$	$t > 100$
dtoEndereçoEstabelecimento.Estado	string com 2 caracteres	$t == 2$	$t != 2$
	2 caracteres referentes a um estado Brasileiro	Sim	Nao
dtoEndereçoEstabelecimento.Numero	Número natural	$n \in \text{Naturais}$	$n \notin \text{Naturais}$

Índice

231.PV01: Criar um estabelecimento com valores mínimos válidos, com CEP válido preenchido	4
231.PV02: Criar um estabelecimento com valores máximos válidos, com CEP não preenchido	5
231.PV03: Criar estabelecimento 'filial'	6
231.PV04: Criar estabelecimento 'filial' já existente	7
231.PI01: DtoEstabelecimento nulo	8
231.PI02: DtoEstabelecimento em formato diferente de JSON DtoEstabelecimento	9
231.PI03: Estabelecimento com nome começando com caractere que não seja letra	10
231.PI04: Estabelecimento com nome de tamanho menor que o permitido	11
231.PI05: Estabelecimento com nome de tamanho maior que o permitido	12
231.PI06: Estabelecimento com número de caracteres de CEP próximo de 0	13
231.PI07: Estabelecimento com Estado contendo número de caracteres diferente do permitido	14
231.PI08: Estabelecimento com Estado inválido	15
231.PI09: Estabelecimento com Cidade contendo menor número de caracteres que o permitido	16
231.PI10: Estabelecimento com Cidade contendo maior número de caracteres que o permitido	17
231.PI11: Estabelecimento com Número não pertencente ao conjunto dos números naturais	18
231.PI12: Estabelecimento com Rua com número de caracteres menor que o permitido	19
231.PI13: Estabelecimento com Rua com número de caracteres maior que o permitido	20

TESTES DE CAIXA PRETA

Obs.: Os valores em cinza não são informados pelo usuário.

VÁLIDOS

231.PV01 - Criar um estabelecimento com valores mínimos válidos e CEP válido preenchido:

Descrição:

Criar um estabelecimento inédito contendo nome com tamanho mínimo, CEP de 8 caracteres válido, Rua com tamanho mínimo, Cidade com tamanho mínimo, Estado com tamanho de 2 caracteres, Número sendo um número pertencente ao conjunto de números Naturais.

Saída Esperada:

Dados: Um DtoEstabelecimento correspondente aos dados enviados, tanto de estabelecimento quanto de endereço, e um novo DtoChave.

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 8 caracteres e válido.

Rua: começando com letra e com tamanho de 3 caracteres.

Cidade: começando com letra e com tamanho de 3 caracteres.

Estado: Caracteres referentes a um estado Brasileiro.

Número: Número natural de 1 dígito.

Estabelecimento.id: irrelevante;

Estabelecimento.Nome: começando com letra e possuindo apenas 3 caracteres;

Pre-condições:

O usuário deverá estar na página de Criar Estabelecimento;

Pos-condições:

Chave atualizada devolvida ao usuário;

Novo estabelecimento cadastrado no servidor;

Novo endereço de estabelecimento cadastrado no servidor

231.PV02 - Criar um estabelecimento com valores máximos válidos, com CEP não preenchido:

Descrição:

Criar um estabelecimento inédito contendo nome com tamanho mínimo, CEP não preenchido, Rua com tamanho mínimo, Cidade com tamanho mínimo, Estado com tamanho de 2 caracteres, Número sendo um número pertencente ao conjunto de números Naturais.

Saída Esperada:

Dados: Um DtoEstabelecimento correspondentes aos dados enviados, tanto de estabelecimento quanto de endereço, e um novo DtoChave.

Variaveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

id: irrelevante;

Cep: 0 dígitos.

Rua: Com tamanho de 100 caracteres.

Cidade: Com tamanho de 100 caracteres.

Estado: Estado: Caracteres referentes a um estado Brasileiro..

Número: Número natural de 50 caracteres.

Estabelecimento.id: irrelevante;

Estabelecimento.Nome: começando com letra e possuindo 100 caracteres;

Pre-condições:

O usuário deverá estar na página de Criar Estabelecimento;

Pos-condições:

Chave atualizada devolvida ao usuário;

Novo estabelecimento cadastrado no servidor;

Novo endereço de estabelecimento cadastrado no servidor;

231.PV03 - Criar estabelecimento 'filial':

Descrição:

Criar um estabelecimento com nome já existente, porém com endereço contendo algum valor diferente de algum já existente.

Saída Esperada:

Dados: Um DtoEstabelecimento com os dados enviados, tanto de estabelecimento quanto de endereço, e um novo DtoChave.

Variaveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

[todos algum dos valores do endereço de estabelecimento deve ser diferente de um dos endereços de algum estabelecimento já cadastrado]

Estabelecimento.Nome: nome de estabelecimento já existente;

Pre-condições:

O usuário deverá tentar criar estabelecimento;

Pos-condições:

Chave atualizada devolvida ao usuário;

Novo endereço de estabelecimento cadastrado no servidor vinculado ao estabelecimento já existente;

231.PV04: Criar estabelecimento 'filial' já existente

Descrição:

Criar um estabelecimento com nome e endereço já existentes.

Saída Esperada:

Dados: Um DtoEstabelecimento com os dados enviados, tanto de estabelecimento quanto de endereço, e um novo DtoChave.

Variaveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

[todos algum dos valores, tanto o nome do estabelecimento quanto seu endereço, ja devem estar cadastrados no servidor]

Pre-condições:

O usuário deverá tentar criar estabelecimento;

Pos-condições:

Chave atualizada devolvida ao usuário;

Endereço e Estabelecimento existentes no servidor retornados para o usuário.

231.PI01 - DtoEnderecoEstabelecimento nulo:

Descrição:

Tentar cadastrar um Estabelecimento sem enviar o JSON de DtoEstabelecimento.

Saída Esperada:

Mensagem de erro de código "101 - Dados foram perdidos durante o processo!";

Variaveis de Entrada:

Chave:

token: >= 1000 e <= 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

null

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI02 - DtoEnderecoEstabelecimento em formato diferente de JSON DtoEstabelecimento:

Descrição:

Tentar cadastrar um estabelecimento sem passar um JSON de DtoEstabelecimento válido.

Saída Esperada:

Mensagem de erro de código "101": "Dados foram perdidos durante o processo!";

Variaveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EnderecoEstabelecimento:

{DtoEstabelecimentoEmUmFormatoNãoJSON}

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI03 -Estabelecimento com nome começando com caractere que não seja letra:

Descrição:

Usuário tenta cadastrar estabelecimento cujo nome começa com um caractere que não é uma letra.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nNome do Estabelecimento";

Variaveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 8 caracteres e válido.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Estabelecimento.id: irrelevante;

Estabelecimento.nome: começando com caractere que não seja letra e possuindo apenas 3 caracteres;

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI04: Estabelecimento com nome de tamanho menor que o permitido:

Descrição:

Usuário tenta cadastrar um estabelecimento cujo nome possui menos de 3 caracteres (o mínimo permitido).

Saída Esperada:

Mensagem de erro de código “104”: "Existem campos com valores inválidos:\nNome do Estabelecimento”;

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 8 caracteres e válido.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Estabelecimento.id: irrelevante;

Estabelecimento.nome: começando com caractere que seja letra e possuindo apenas 2 caracteres;

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI05: Estabelecimento com nome de tamanho maior que o permitido:

Descrição:

Usuário tenta cadastrar um estabelecimento com nome maior que o permitido, com mais de 100 caracteres.

Saída Esperada:

Mensagem de erro de código “104”: "Existem campos com valores inválidos:\nNome do Estabelecimento”;

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 8 caracteres e válido.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Estabelecimento.id: irrelevante;

Estabelecimento.nome: começando com letra e possuindo 101 caracteres (espaços consecutivos contam como 1);

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI06: Estabelecimento com número de caracteres de CEP maior que 0 e diferente de 8:

Descrição:

Usuário tenta cadastrar um estabelecimento com um CEP de tamanho maior que 0 e diferente de 8.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nCep";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 1 caractere.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Estabelecimento.id: irrelevante;

Estabelecimento.idEstabelecimento: maior que 0 e pertencente a algum usuário existente;

nome: Com tamanho de 3 caracteres.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI07: Estabelecimento com Estado contendo número de caracteres diferente do permitido

Descrição:

Usuário tenta cadastrar um estabelecimento com Estado com número de caracteres diferente do permitido, que são 2 caracteres.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nEstado";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 3 caracteres.

Número: Número natural de 1 caractere.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI08: Estabelecimento com Estado inválido

Descrição:

Usuário tenta cadastrar um estabelecimento com Estado inválido.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nEstado";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres, mas inválido.

Número: Número natural de 1 caractere.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI09: Estabelecimento com Cidade contendo menor número de caracteres que o permitido

Descrição:

Usuário tenta cadastrar um estabelecimento com Cidade contendo menor número de caracteres que o permitido.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nCidade";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 2 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI10: Estabelecimento com Cidade contendo maior número de caracteres que o permitido

Descrição:

Usuário tenta cadastrar um estabelecimento com Cidade contendo maior número de caracteres que o permitido.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nCidade";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 101 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI11: Estabelecimento com Número não pertencente ao conjunto dos números naturais

Descrição:

Usuário tenta cadastrar um estabelecimento com Numero não pertencente ao conjunto dos números naturais.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nNumero";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 3 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número Inteiro -1.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI12: Estabelecimento com Rua com número de caracteres menor que o permitido

Descrição:

Usuário tenta cadastrar um estabelecimento com Rua contendo número de caracteres menor que o permitido.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nRua";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 2 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;

231.PI13: Estabelecimento com Rua com número de caracteres maior que o permitido

Descrição:

Usuário tenta cadastrar um estabelecimento com Rua contendo número de caracteres maior que o permitido.

Saída Esperada:

Mensagem de erro de código "104": "Existem campos com valores inválidos:\nRua";

Variáveis de Entrada:

Chave:

token: ≥ 1000 e ≤ 9999 (4 caracteres);

ultimoAcesso: string no padrão "dd/mm/aaaa hh:mm:ss";

OBS.: a chave deve existir no banco de dados.

Estabelecimento:

id: irrelevante;

nome: Com tamanho de 3 caracteres.

EndereçoEstabelecimento:

id: irrelevante;

Cep: Com 0 caracteres.

Rua: Com tamanho de 101 caracteres.

Cidade: Com tamanho de 3 caracteres.

Estado: Com tamanho de 2 caracteres.

Número: Número natural de 1 caractere.

Pre-condições:

O usuário deverá estar tentando criar estabelecimento;

Pos-condições:

Mensagem de erro exibida na tela;

Estabelecimento não cadastrado;