#Desafio Sensedia

Utilizando a API do Trello (<https://developers.trello.com/docs/api-introduction>) fazer levantamento dos cenários para a seguinte regra de negócio:

1. O usuário deve realizar a autenticação no Trello APIs (obter token de acesso).

2. O usuário deve criar um card e editar esse card.

3. O usuário deve excluir esse card e também as sujeiras geradas por esta automação.

Para a criação dos cenários de testes devem ser considerados casos de sucesso e fluxos alternativos sobre as requisições da API do Trello para cada tópico apresentado acima na especificação de requisito.

Como você vai trabalhar diariamente com a Automação, **é muito importante que esses cenários sejam automatizados utilizando uma ferramenta de sua preferência.**

Sugestões de ferramentas para automação da API: SuperTest (Node.js), HttParty (Ruby), Karate / Rest assured (Java) e afins. Se você manjar de BDD, usar ferramentas como o Cucumber seria bem legal :)

Referência: <https://developers.trello.com/docs/api-introduction> #Ferramenta Utilizada Java, AssuredRest, bitbucket, postman, cucumeber# Especificação de Requisito

Montar ambiente:

Instalar o Cucumber, para rodar os testes automatizados deverá ser aberto a classe “Runner.java” dentro do packege test.runners e executar Botão direito “Run as” >> “1 Junit Test”.

**Features e seus casos de Testes**

**Feature 1**

Como usuário quero criar um board no trello.

**Cenário 1**

**Descrição do cenário :** Criar um board

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com.

**Resultado Esperado:** Board criado com sucesso

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

<https://api.trello.com/1/boards/?key=50322bf66cfbc3865f0a5398cf430724>

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": "NameBoard",

"desc": "DescBoard",

"token": "87f80f85ee45e5e9d5d06e79982e349sb99c30bad4d6f41bb8c9f5d6e8aa59c09"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um array com id eos parâmetros passados no body da requisição, com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um aquivo o retorno da requisição.

**Cenário 2**

**Descrição do cenário :** Tentar Inserir um board com um token invalido.

**Pré Requisitos :** Estar com key valido e token “invalido” da API.

**Resultado Esperado:** Api retornar a mensagem para o usuário “invalid token”.

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/boards/?key=50322bf66cfbc3865f0a5398cf430724

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": "NameBoard",

"desc": "DescBoard",

"token": "invalidToken"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem “invalid token” com statusCode = 401 Unauthorized, com o tempo de resposta e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo de retorno da requisição, anexar junto dos testes.

**Cenário 3**

**Descrição do cenário :** Tentar Inserir um board com um key invalido.

**Pré Requisitos :** Estar com key invalido e token valido da API.

**Resultado Esperado:** Api retornar a mensagem para o usuário “invalid key”.

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/boards/?key=invalidkey

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": "NameBoard",

"desc": "DescBoard",

"token": "87f80f85ee45e5e9d5d06e79982e349sb99c30bad4d6f41bb8c9f5d6e8aa59c09"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem “invalid key” com statusCode = 401 Unauthorized, com o tempo de resposta e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo de retorno da requisição, anexar junto dos testes.

------------------------------------------------------------------------------------------------------------------------------------------

**Feature 2**

Como usuário eu quero criar um List no trello.

**Cenário 1**

**Descrição do cenário :** Criar um List

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1.

**Resultado Esperado:** Board criado com sucesso

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

<https://api.trello.com/1/list/?key=50322bf66cfbc3865f0a5398cf430724>

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": " 649845ce0a37634632afaafd9d62ce9b",

"pos": " 8dc2c485f3108c7fb2012f756ca3f435",

"closed": " false",

"idBoard": "{idBoard}",

"token": " 8dc2c485f3108c7fb2012f756ca3f435"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto list criado com id eos parâmetros , com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno da requisição.

------------------------------------------------------------------------------------------------------------------------------------------

**Feature 3**

Como usuário eu quero procurar um board no trello.

**Cenário 1**

**Descrição do cenário :** Procurar um board

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1.

**Resultado Esperado:** Board encontrado com sucesso

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

<https://api.trello.com/1/list/?key=50322bf66cfbc3865f0a5398cf430724>

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": " 649845ce0a37634632afaafd9d62ce9b",

"pos": " 8dc2c485f3108c7fb2012f756ca3f435",

"closed": " false",

"idBoard": "{idBoard}",

"token": " 8dc2c485f3108c7fb2012f756ca3f435"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto list criado com id eos parâmetros , com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno da requisição.

**Cenário 2**

**Descrição do cenário :** tentar procurar um board com um boardid invalido

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1.

**Resultado Esperado:** Mensagem de erro exibida “id board invalido”

**Passo 1**

Configurar a url da requisição HTTP GET conforme ex:

<https://api.trello.com/1/boards/5dac9b2f09c5ed5caf525487?key=50322bf66cfbc3865f0a5398cf430724&token=fb2253fc938002037c4aa86c4897be02dde7b964555112c25dddd332585a1600185>

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": " 649845ce0a37634632afaafd9d62ce9b",

"pos": " 8dc2c485f3108c7fb2012f756ca3f435",

"closed": " false",

"idBoard": "{idBoardInvalido}",

"token": " 8dc2c485f3108c7fb2012f756ca3f435"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “GET”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um array com os dados do borad com id eos parâmetros , com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno da requisição.

**Cenário 3**

**Descrição do cenário :** tentar procurar um board com key invalido

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1.

**Resultado Esperado:** Mensagem de erro exibida “invalid key”

**Passo 1**

Configurar a url da requisição HTTP GET conforme ex:

<https://api.trello.com/1/boards/5dac9b2f09c5ed5caf525487?key=invalidkey&token=fb2253fc938002037c4aa86c4897be02dde7b964555112c25dddd332585a1600185>

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": " 649845ce0a37634632afaafd9d62ce9b",

"pos": " 8dc2c485f3108c7fb2012f756ca3f435",

"closed": " false",

"idBoard": "{idBoard}",

"token": " 8dc2c485f3108c7fb2012f756ca3f435"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “GET”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem ‘invalid key”, com status code 401, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno da requisição.

**Cenário 4**

**Descrição do cenário :** tentar procurar um board com token invalido

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1.

**Resultado Esperado:** Mensagem de erro exibida “invalid token”

**Passo 1**

Configurar a url da requisição HTTP GET conforme ex:

https://api.trello.com/1/boards/5dac9b2f09c5ed5caf525487?key=50322bf66cfbc3865f0a5398cf430724

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem alterados ex. abaixo:

{

"name": " 649845ce0a37634632afaafd9d62ce9b",

"pos": " 8dc2c485f3108c7fb2012f756ca3f435",

"closed": " false",

"idBoard": "{idBoard}",

"token": " invalidtoken"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “GET”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem ‘invalid token”, com status code 401, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno da requisição.

------------------------------------------------------------------------------------------------------------------------------------------

**Feature 4**

Como usuário eu quero criar um card no trello.

**Cenário 1**

**Descrição do cenário :** Criar um Card

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Card encontrado com sucesso

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/cards?idList=5d938ee091f58f6044729885&keepFromSource=all&key=80dd902d459c0fec323b0c3ae4375234&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "teste123",

"pos": 131073,

"closed": false,

"idLabels": [],

"idMembers": [],

"dateLastActivity": 1569951721591,

"idBoard": "5d938de2adb18a0273993b18",

"idList": "5d938ea70fd3b63787e681e7",

"token": "87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto list criado com id eos parâmetros , com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 2**

**Descrição do cenário :** Criar um Card com um idlist inexistente

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Menssagem enviada de erro.

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/cards?idList=Idvinalidp&keepFromSource=all&key=80dd902d459c0fec323b0c3ae4375234&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "teste123",

"pos": 131073,

"closed": false,

"idLabels": [],

"idMembers": [],

"dateLastActivity": 1569951721591,

"idBoard": "5d938de2adb18a0273993b18",

"idList": "invalidid",

"token": "87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto list criado com id eos parâmetros , com status code 400, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 3**

**Descrição do cenário :** Criar um Card com um invalid key

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Menssagem enviada de erro.

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/cards?idList=Idvinalidp&keepFromSource=all&key=invalid&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "teste123",

"pos": 131073,

"closed": false,

"idLabels": [],

"idMembers": [],

"dateLastActivity": 1569951721591,

"idBoard": "5d938de2adb18a0273993b18",

"idList": "invalidid",

"token": "87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma menssagem, com status code 401, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 4**

**Descrição do cenário :** Criar um Card com um token

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Menssagem enviada de erro.

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/cards?idList=Idvinalidp&keepFromSource=all&key=80dd902d459c0fec323b0c3ae4375234&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "teste123",

"pos": 131073,

"closed": false,

"idLabels": [],

"idMembers": [],

"dateLastActivity": 1569951721591,

"idBoard": "5d938de2adb18a0273993b18",

"idList": "invalidid",

"token": "invalid token"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto list criado com id eos parâmetros , com status code 401, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 5**

**Descrição do cenário :** Tentar criar um card sem passar o id list o body da resquisição

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello sem passar um id liste o body da requisição.

**Resultado Esperado:** Menssagem enviada de erro.

**Passo 1**

Configurar a url da requisição HTTP POST conforme ex:

https://api.trello.com/1/cards?idList=5d938ee091f58f6044729885&keepFromSource=all&key=50322bf66cfbc3865f0a5398cf430724&token=fb2253fc938002037c4aa86c4897be02dde7b964555112c25d332585a1600185

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “POST”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem “could not find the board that the card belongs to”, com status code 400, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

------------------------------------------------------------------------------------------------------------------------------------------

**Feature 5**

Como usuário eu quero atualizar um card no trello.

**Cenário 1**

**Descrição do cenário :** Atualizada um Card

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Card atualizado com sucesso.

**Passo 1**

Configurar a url da requisição HTTP PUT conforme ex:

https://api.trello.com/1/cards/{IdCard}?key=80dd902d459c0fec323b0c3ae4375234&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "UpdateCardName","desc": "UpdateCardDesc"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “PUT”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto list atualizado com os parâmetros passados com id , com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 2**

**Descrição do cenário :** Tentar atualizada um Card com um card id invalido

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Menssagem de erro retornada.

**Passo 1**

Configurar a url da requisição HTTP PUT conforme ex:

https://api.trello.com/1/cards/{IdCarinvalido}?key=80dd902d459c0fec323b0c3ae4375234&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "UpdateCardName","desc": "UpdateCardDesc"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “PUT”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma messagem “invalid id”, com status code 400 Bad Request, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 3**

**Descrição do cenário :** Tentar atualizar um Card com uma key invalida

**Pré Requisitos :** Estar com key invalida e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Mensagem de erro retornada.

**Passo 1**

Configurar a url da requisição HTTP PUT conforme ex:

https://api.trello.com/1/cards/{IdCard}?key={Keyinvalid}&token=87f80f85ee45e5e9d5d06e79982e349b99c30bad4d6f41bb8c9f5d6e8aa59c09

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "UpdateCardName","desc": "UpdateCardDesc"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “PUT”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem “invalid key” , com status code 401 Unauthorized, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 4**

**Descrição do cenário :** Tentar atualizar um Card com uma token invalida

**Pré Requisitos :** Estar com key e token invalido da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Mensagem de erro retornada.

**Passo 1**

Configurar a url da requisição HTTP PUT conforme ex:

https://api.trello.com/1/cards/{IdCard}?key={Key}&token={tokenInvalid}

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

"name": "UpdateCardName","desc": "UpdateCardDesc"

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “PUT”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir uma mensagem “invalid token” , com status code 401 Unauthorized, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno

**Cenário 5**

**Descrição do cenário :** Atualizar um card sem alterar nada.

**Pré Requisitos :** Estar com key e token da API já configuradas no site da Trello.com e um id de um board criado na feature 1 e um list feature 2.

**Resultado Esperado:** Mensagem de erro retornada.

**Passo 1**

Configurar a url da requisição HTTP PUT conforme ex:

https://api.trello.com/1/cards/{IdCard}?key={Key}&token={token}

**Resultado Esperado**

Url da Requisição configurada.

**Passo 2**

Configurar o body da requisição passando os parâmetros a serem pesquisados ex. abaixo:

{

}

**Resultado Esperado**

Body da requisição configurada.

**Passo 3**

Selecionar o tipo da Requisição a ser executada “PUT”.

**Resultado Esperado**

Tipo da Requisição selecionada.

**Passo 4**

Clicar na opção “Send” para fazer a requisição.

**Resultado Esperado**

No response body deverá vir um objeto do card sem alterações , com status code 200 OK, tempo de execução em “ms” e o tamanho.

**Passo 5**

Clicar na opção “Save Response” para salvar em um arquivo o retorno