



# API TESTING WITH POSTMAN

<b>SUMMARY</b>	<b>2</b>
<b>US001   Cards   Test create, update and delete a card from a trello board using Postman</b>	<b>2</b>
IN SCOPE	2
ACCEPTANCE CRITERIA	3
Scenario 1 (CREATE): User creates a card in the TO DO list with the name of a random city.	3
Scenario 2 (UPDATE): User updates the card description with the message "Updated!" and changes the color of the cover to purple.	3
Scenario 3 (UPDATE): User moves the card to the IN PROGRESS list and then to the DONE list.	3
Scenario 4 (DELETE): User deletes the card and checks if the card is removed.	3
<b>EXPECTED RESULTS</b>	<b>4</b>
<b>PRECONDITIONS</b>	<b>4</b>
<b>Test Cases</b>	<b>5</b>
SUMMARY: TC001  US001   Test create a card with a random city name.	5
SUMMARY: TC002  US001   Test update a card description and change the cover color.	5
SUMMARY: TC003  US001   Test move the card to the IN PROGRESS list.	6
SUMMARY: TC004  US001   Test move the card to the DONE list.	6
SUMMARY: TC005  US001   Test delete the card.	7
SUMMARY: TC006  US001   Check if the card is deleted.	7
<b>Test Execution</b>	<b>7</b>

## SUMMARY

In this project, I will validate the basic features (create, update and delete cards) of a Trello board, using the APIs and the public Trello API documentation with the tool Postman to make actions with POST, PUT, DELETE methods in an automated way and verify results in the UI.

This following documentation is written the same way it would appear in Jira XRAY.

- Software under test Documentation:  
<https://developer.atlassian.com/cloud/trello/guides/rest-api/api-introduction/>

---

### US001 | Cards | Test create, update and delete a card from a trello board using Postman

#### Description

*As a user with a registered trello account,  
I want to create, update and delete cards,  
So that I can organize myself.*

#### IN SCOPE

- Create a card
- Create lists
- Update the description of a card
- Change the color of a card's cover
- Move a card to other lists.
- Delete a card

## ✓ ACCEPTANCE CRITERIA

**Scenario 1 (CREATE): User creates a card in the TO DO list with the name of a random city.**

↳ 📁 Folder #1:

**POST** Create a card with a random city name.

**Scenario 2 (UPDATE): User updates the card description with the message "Updated!" and changes the color of the cover to purple.**

↳ 📁 Folder #2:

**PUT** Update card description and change the cover color.

**Scenario 3 (UPDATE): User moves the card to the IN PROGRESS list and then to the DONE list.**

↳ 📁 Folder #3:

**PUT** Move card to the IN PROGRESS list.

**PUT** Move card to the DONE list.

**Scenario 4 (DELETE): User deletes the card and checks if the card is removed.**

↳ 📁 Folder #4:

**DEL** Delete Card.

**GET** Check if the card is removed.

## ✓ EXPECTED RESULTS

- A new card is created with a random city name.
  - The description of the card is updated with the message "Updated!".
  - The color of the cover is changed to purple.
  - The card is moved to the TO DO list, then to IN PROGRESS and finally to DONE.
  - The card is deleted.
  - The status code is 200 for all requests.
- 

## ⚠ PRECONDITIONS

- Have a trello account and be logged in.
- Have a board created.
- Have 3 lists created (TO DO, IN PROGRESS, DONE)
- Have saved in the Environment (as variables) the URL (as "dom"), the token and the key that are found in the Trello documentation.
- Have the lists saved as variables (ListA, ListB, ListC) with their respective idList.
- Have set as a variable the ID of the card:

```
1. const body = pm.response.json();
2. const idC = body.id
3. pm.collectionVariables.set("cardId", idC)
```

- Have saved the Card Id as a path variable.

## Test Cases

**SUMMARY: TC001| US001 | Test create a card with a random city name.**

**! ADDITIONAL PRECONDITION:** Have saved in query params "name = \$randomcityname"

TC001			
Test Status: PASSED			
STEP	ACTION	DATA	EXPECTED RESULT
1	Click on Enter a Request URL and using the POST method, make a URL that creates a new card with a random city name in the TO DO list.	{{dom}}/cards?key={{key}}&token={{token}}&name={{\$randomCity}}&idList={{List A}}	
2	Click on SEND button		<ul style="list-style-type: none"><li>A new card should appear on the trello board with the name of a random city.</li><li>The status of the response should be 200 OK.</li></ul>

**SUMMARY: TC002| US001 | Test update a card description and change the cover color.**

**! ADDITIONAL PRECONDITION:** Have saved in query params "desc = Updated!"

TC002			
Test Status: PASSED			
STEP	ACTION	DATA	EXPECTED RESULT
1	Click on Enter a Request URL and using the PUT method, make a URL that modify the new card description.	{{dom}}/cards/:id?key={{key}}&token={{token}}&desc=Updated!	
2	Click on the "Body" tab, then on "Raw" and then select "JSON".	{"cover" : {"color":"purple"}}	

2	Click on SEND button		<ul style="list-style-type: none"> <li>The description should be updated.</li> <li>The cover color should be purple.</li> <li>The status response should be 200 OK</li> </ul>
---	----------------------	--	---

---

**SUMMARY: TC003| US001 | Test move the card to the IN PROGRESS list.**

TC003			
Test Status: PASSED			
STEP	ACTION	DATA	EXPECTED RESULT
1	Click on Enter a Request URL and using the PUT method, make a URL that moves the card to the next list.	{{dom}}/cards/:id?idList={{ListB}}&key={{key}}&token={{token}}	
2	Click on SEND button		<ul style="list-style-type: none"> <li>The card should be on the second list "IN PROGRESS"</li> <li>The status response should be 200 OK.</li> </ul>

---

**SUMMARY: TC004| US001 | Test move the card to the DONE list.**

TC004			
Test Status: PASSED			
STEP	ACTION	DATA	EXPECTED RESULT
1	Click on Enter a Request URL and using the PUT method, make a URL that moves the card to the next list.	{{dom}}/cards/:id?idList={{ListC}}&key={{key}}&token={{token}}	
2	Click on SEND button		<ul style="list-style-type: none"> <li>The card should be on the third list "DONE"</li> <li>The status response should be 200 OK.</li> </ul>

**SUMMARY: TC005| US001 | Test delete the card.**

TC005			
Test Status: PASSED			
STEP	ACTION	DATA	EXPECTED RESULT
1	Click on Enter a Request URL and using the DELETE method, make a URL that removes the card.	{{dom}}/cards/:id?key={{key}}&token={{token}}	
2	Click on the SEND button.		<ul style="list-style-type: none"><li>• The card should be deleted.</li><li>• The status code should be 200 OK.</li></ul>

**SUMMARY: TC006| US001 | Check if the card is deleted.**

TC006			
Test Status: PASSED			
STEP	ACTION	DATA	EXPECTED RESULT
1	Click on Enter a Request URL and using the GET method, make a URL that checks if the card is removed.	{{dom}}/cards/:id?key={{key}}&token={{token}}	
2	Click on SEND button		<ul style="list-style-type: none"><li>• The status code should be 200 OK.</li></ul>

 To see the Test Execution [CLICK HERE](#)