Printscreen explanations

1 - Creare bord manual _ Trello.png

Title: Create board manually

Description:
This screenshot shows the Trello interface where the user has manually created a board named **“1 - Creare bord manual”**. No lists have been added yet. On the left sidebar, we can see multiple boards, including those used for the Postman practice tasks.

Purpose:
Demonstrates that the user has completed the first step of the homework—manually creating a Trello board.

2 - ListaBoard - Tema 11 - Postman.png

Title: Board List

Description:
This Postman screenshot shows a **GET** request to the endpoint /members/me/boards in order to fetch all boards associated with the user.
In the response JSON (bottom section), we see details about the board "1 - Creare bord manual", including its **id**, **name**, and other properties.

Purpose:
Confirms that the user successfully retrieved the list of boards and extracted the **board ID**, which will be used in the following requests.

3 - Status - Tema 11 - Postman.png

Title: Status

Description:
This image shows a **POST** request used to create a new list on the previously created board, using the /1/lists endpoint. The URL parameters include name, idBoard, key, and token.
In the **Tests** tab, a script checks if the response status is 200. The result shows **PASS**, meaning the test passed successfully.

Purpose:
Confirms that the new list was created successfully and that the test to validate status code 200 is working correctly.

3.1 - InformatiiListaNoua - Tema 11 - Postman.png

Title: New List Information

Description:
This screenshot shows a **GET** request to retrieve the details of the list created earlier, using the list's id.
The JSON response includes relevant data such as the id, name (which is "Am creat o lista noua in bord" – "I created a new list in the board"), idBoard, and other attributes.

Purpose:
Confirms that the list was created with the correct name and that its details can be retrieved via a GET request.

3.1 - Verificare nume Lista Tema 11 - Postman.png

Title: List Name Verification

Description:
This image displays a test written in Postman's **Tests** tab that verifies whether the list name returned in the JSON response is exactly **"Am creat o lista noua in bord"**.
The script parses responseJson.name and checks if it equals the expected string. The result shows **PASS**, meaning the name matched successfully.

Purpose:
Verifies that the list name returned by the API matches the one used in the list creation request, as required in the task.

Image: "4 - Trello.png"

- **What it shows:** A manually created Trello board titled "1 - Creare bord manual" with two lists:
    - "Am creat o listă nouă în bord"
    - "Lista creată manual in board"
      Each list contains one card.

- **Conclusion:** ✔️ **Requirement 1 is fulfilled** – a manual board with lists and cards was successfully created.

Image: "4.1 - InformatiiCard - 2 - Tema 11 - Postman.png"

- **What it shows:** A GET request in Postman retrieving card information.
- **Response:** Status is 200 OK and the JSON shows a card named *"Am creat un card in lista"*.
- **Requirement 4.1 is fulfilled** – card information was retrieved successfully and the name matches the created card.

Image: "4.1 - InformatiiCard - Tema 11 - Postman.png"

- **What it shows:** Similar GET request for a card with full JSON response and status 200 OK.
- You correctly repeated the step to fetch card info – everything is consistent and correct.

Image: "4.1 - Verificare Nume - Tema 11 - Postman.png"

- **What it shows:** A test in Postman checking that the card's name is correct.

Test:

```
pm.test("Numele Cardului: Am creat un card in lista", () => {
    const responseJson = pm.response.json();
    pm.expect(responseJson.name).to.eql("Am creat un card in lista");
});
```

- **Test passes** – name verification is correctly implemented.

Image: "4 - CreateCardLista-ListaNoua - Tema 11 - Postman.png"

- **What it shows:** A POST request to create a card in a list with the correct parameters (idList, key, token, name).
- **Result:** Card with name *"Am creat un card in lista"* was created.
- **Requirement 4 is fulfilled** – card creation works correctly.

Image: "4 - Status - Tema 11 - Postman.png"

What it shows: Two Postman tests that check the response status code is 200:

Test:

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});

pm.test("Successful POST request", function () {
    pm.expect(pm.response.code).to.be.oneOf([200]);
});
```

- **Test result:** Both tests passed.
- Status test for card creation is implemented and validated.

Step 5 – Update Card Name

Screenshot: 5UpdateNumeCard (Postman)

- **Request Type**: PUT
- **Endpoint**: https://api.trello.com/1/cards/{{idAmCreatUnCard}}?key={{myKey}}&token={{myToken}}&name={{numeNouCard}}

Test:

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
```

- **Result**: The test passed — status 200 OK confirms the card name update was successful.

Screenshot: 5 - UpdateNumeCard - 2 (Postman)

- **Body response**: Shows that the new card name "NumeNouPentruCard" is correctly returned in the response.

JSON confirmation:

"name": "NumeNouPentruCard"

Screenshot: 5 - Update nume card - Trello

- **Visual Confirmation**: On the Trello board, the card now displays the updated name NumeNouPentruCard, confirming that the PUT request worked correctly.

Step 5.1 – Verify the Updated Card Name

Screenshot: 5.1 - Verificare Nume Card Actualizat

- **Request Type**: GET

Test:

```
pm.test("Verificare nume dupa update: NumeNouPentruCard", () => {
    const responseJson = pm.response.json();
    pm.expect(responseJson.name).to.eql("NumeNouPentruCard");
});

pm.test("Se verifica daca numele cardului este actualizat", function () {

pm.expect(pm.response.json().NumeNouPentruCard).to.eql(pm.environment.get("NumeNou
PentruCard"));
});
```

- **Result**: Both assertions passed — verifying that the name returned matches the new value both as string and via environment variable.

Step 6 – Delete the Card

Screenshot: 6DeleteCard (Postman)

- **Request Type**: DELETE
- **Endpoint**: https://api.trello.com/1/cards/{{idAmCreatUnCard}}?key={{myKey}}&token={{myToken}}

Test:

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
```

- **Result**: Test passed — status 200 confirms that the card was successfully deleted.

🔹 **Screenshot: 6 - delete _ Trello**

- **Visual Check**: The card with the updated name is no longer present in the Trello board, confirming successful deletion.
- **Card name update** via PUT – Confirmed via both API response and Trello board.
- **Verification of updated name** – Passed via test scripts in GET request.
- **Deletion of the card** – Executed and confirmed both by API and Trello interface.