|  |
| --- |
|  |
| **Adrien Taneko**  **R0780417**  **Groep1-**2 |
| Tweede Bachelor - Toegepaste Informatica  Choose an item. |
|  |
|  |
| **Softaware testing**  Frank Serneels |
|  |
| Academiejaar 2021-2022 |

|  |  |
| --- | --- |
| Documentatie testomgeving | |
| REST API testing met Postman |

Inhoudstafel

[Inleiding 3](#_Toc84811460)

[Labo 1 - Inleiding tot API-verzoeken 4](#_Toc84811461)

[Labo 2 – Meer verzoeken en testen 5](#_Toc84811462)

[Labo 3 - Dynamische verzoeken met gebruik te maken van variabelen. 10](#_Toc84811463)

[Labo 4- 12](#_Toc84811464)

# Inleiding

Voor de vak software testing zijn we gevraagd om een ​​testomgeving naar keuze te onderzoeken en te documenteren.

In dit document zal ik documenteren

# Labo 1 - Inleiding tot API-verzoeken

*Open a new tab in Postman and create a GET request with this URL:*

*https://api.nytimes.com/svc/books/v3/lists.json*

*Submit the request.*

**What is the status code that you get back?**

{

"fault": {

"faultstring": "Failed to resolve API Key variable request.queryparam.api-key",

"detail": {

"errorcode": "steps.oauth.v2.FailedToResolveAPIKey"

}

}

}

*According to the error message you have received, the API is expecting your API key as a query parameter. The error that you see in the response body also indicates that is wrong.  
  
Go ahead and create a new query parameter called****api-key****and put the API key you have received in the Value field.*

*Now resubmit the request.*

**What status code do you get now?**

{

    "status": "ERROR",

    "copyright": "Copyright (c) 2021 The New York Times Company.  All Rights Reserved.",

    "errors": [

        "No Parameters specified\n",

        "Bad Request"

    ],

    "results": []

}

Try adding a GET query parameter called **list** with any value you want.

Resubmit the request.

**What status code do you see now?**

{

    "status": "OK",

    "copyright": "Copyright (c) 2021 The New York Times Company.  All Rights Reserved.",

    "num\_results": 15,

    "last\_modified": "2021-10-06T22:12:45-04:00",

    "results": [

}

# Labo 2 - Meer verzoeken en testen

*Using trello's online API documentation (https://developer.atlassian.com/cloud/trello/rest/)****build another request which creates a new list called TODO****inside your already created board.*

***Write tests to check that the request performed as expected.****As a minimum, please test the following:*

*- status is 200  
- the name of the list is TODO  
- should not be closed*

Eerst een bord aanmaken, en gebruik zijn id om de TODO-lijst erin te kunnen invoegen

(POST)

pm.test("Status code is 200", *function* () {

    pm.response.to.have.status(200);

});

*const* jsonData = pm.response.json();

pm.test("Check board name", *function* () {

    pm.expect(jsonData.name).to.eql(`Trello board`);

});

*let* id = jsonData.id;

pm.global.set("board\_id", id);

Dan TODO lijst aanmaken, (POST)

pm pm.test("Status code is 200", function () {

pm.response.to.have.status(200);

});

pm.test("Check list name", function () {

var jsonData = pm.response.json();

pm.expect(jsonData.name).to.eql("TODO");

});

pm.test("Check list is not closed", function () {

var jsonData = pm.response.json();

pm.expect(jsonData.closed).to.eql(false);

});

let id = pm.response.json().id;

pm.globals.set("list\_TODO\_id", id);

*Additionally****create a new list****called****DONE****inside your already created board (in a similar way you have created the previous one).*

***Write tests to check that the request performed as expected.****As a minimum, please test the following:*

*- status is 200  
- the name of the list is DONE  
- should not be closed*

(POST)

pm.test("Status code is 200", function () {

    pm.response.to.have.status(200);

});

pm.test("Check list name", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.name).to.eql("DONE");

});

pm.test("Check list is not closed", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.closed).to.eql(false);

});

pm.test("List is created in API\_test board", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.idBoard).to.eql

    (pm.globals.get("board\_id"));

});

let id = pm.response.json().id;

pm.globals.set("list\_DONE\_id", id);

*Inside the TODO list,****create a new card****(which is like a task) with the name****Learn Postman.***

*Write tests to check that the request performed as expected.*

*As a****minimum****, please test the following:*

*- status is 200  
- the name of the card is Learn Postman  
- should not be closed*

***Optional****:*

*- test that the card was created in the desired TODO list*

(POST)

pm.test("Status code is 200", function () {

    pm.response.to.have.status(200);

});

pm.test("Check card name", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.name).to.eql("Learn Postman");

});

pm.test("Check card is not closed", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.closed).to.eql(false);

});

pm.test("card is created in TODO list", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.idList).to.eql

    (pm.globals.get("list\_TODO\_id"));

});

let id = pm.response.json().id;

pm.globals.set("card\_learnpostman\_id", id);

***Move the card****from the TODO list to the DONE list.*

*Write tests to check that the request performed as expected.*

*Please test the following:*

*- status is 200   
- the name of the card is still Learn Postman  
- test that the card was moved to the desired DONE list*

(PUT)

pm.test("Status code is 200", function () {

    pm.response.to.have.status(200);

});

pm.test("Check card name", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.name).to.eql("Learn Postman");

});

pm.test("Check card is not closed", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.closed).to.eql(false);

});

pm.test("card is in DONE list", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.idList).to.eql

    (pm.globals.get("list\_DONE\_id"));

});

let id = pm.response.json().id;

pm.globals.set("card\_learnpostman\_id", id);

*It is time to clean things up. As you do not need the task, the lists and the board, make sure you****delete the board****, so that your Trello account does not get filled with useless boards.*

*Please test the following:*

*- status is 200*

**Hint:**You will be using the **DELETE** method in order to **delete the board**.

pm.test("Status code is 200", function () {

    pm.response.to.have.status(200);

});

Alle testen werken !

Graphical user interface

Description automatically generated with medium confidence

# Labo 3 - Dynamische verzoeken met gebruik te maken van variabelen.

*When creating a new board, instead of My Board, try generating a unique board name by incrementing a number (using a pre-request script).*

*For example the first board you create will be:*

*My Board 1*

*The second board:*

*My Board 2*

*The third board:*

*My Board 3*

*and so on.*

***Hint #1****: Use an environment variable to save the last used number.*

***Hint #2****: Take into account that when retrieving a value for a variable that has not been defined, you will get the value null.*

Pre- request script (uitgevoerd vóór het verzoek)

//Get lastest value of last used number

var number = pm.environment.get("LastUsedNumber");

//increment last used number

pm.environment.set("LastUsedNumber",number+1);

*Use environment variables for any dynamic parts in your requests (boardId, listId) and use them in requests and tests. Replace all usages of global variables in your requests.*

*How do does your request and tests look like now?*

Aanmaken van een nieuwe omgeving met naam Trello waar ik alle variabelen ga bewaren in plaats van in de globale omgeving

A screenshot of a computer

Description automatically generated with medium confidence

*Inspect your environment variables after you reached the last request. Clear any variables that you do not need as soon as you do not use them anymore.*

*What were the lines of code needed to clear the variables?*

pm.environment.unset("board\_id");

pm.environment.unset("list\_TODO\_id");

pm.environment.unset("list\_DONE\_id");

pm.environment.unset("card\_learnpostman\_id");

pm.environment.unset("LastUsedNumber");

Alle testen werken weer !

Graphical user interface

Description automatically generated with medium confidence

# Labo 4-