

**Api testing met postman**

|  |  |
| --- | --- |
| Software Testing | Yarne Valck |
| 05/10/2021 | 3 TI / 1-2 |
| *Toegepaste Informatica* | |

Inhoud

[Manuele Test 3](#_Toc89776744)

[Intro 3](#_Toc89776745)

[Nasa 5](#_Toc89776746)

[Get-Testing met een specifiek Id 5](#_Toc89776747)

[Spotify 7](#_Toc89776748)

[Post-testing 8](#_Toc89776749)

[Put-Testing 12](#_Toc89776750)

[Delete-testing 13](#_Toc89776751)

[Extra requests voor playlist aan te vullen 15](#_Toc89776752)

[Postman Collections voor spotify requests 16](#_Toc89776753)

[Slimme authenticatie token 16](#_Toc89776754)

[Diepgaande tests 20](#_Toc89776755)

[Test automation 23](#_Toc89776756)

[Environment 23](#_Toc89776757)

[Uitleg 23](#_Toc89776758)

[Confirmatie 24](#_Toc89776759)

[Collections 26](#_Toc89776760)

[Collection runner uitleg 26](#_Toc89776761)

[Collection runner uitvoering 28](#_Toc89776762)

# Manuele Test

## Intro

Wanneer we postman opstarten moeten we eerst een paar basis stappen voltooien voordat we door kunnen gaan tot requests en testing die we willen bekomen. Wanneer we een request toevoegen aan onze postman moeten we eerst een goede en duidelijke naam kiezen voor deze request zodat we later niet moeten zoeken voor wat deze nu juist dient.

Afbeelding met tekst

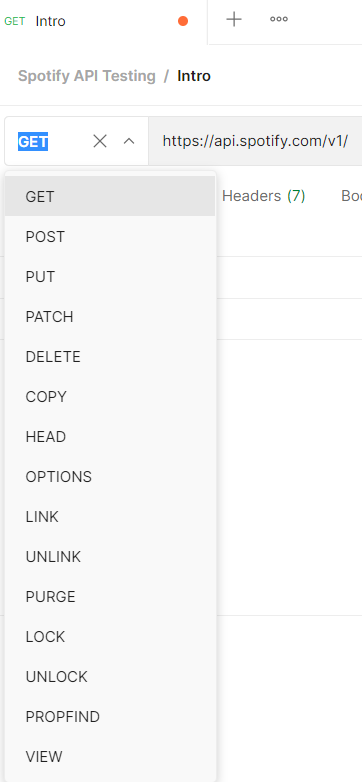
Automatisch gegenereerde beschrijving

Wanneer we dit hebben gedaan kunnen we naar de volgende stap, het zoeken van de basis URL voor de te gebruiken API. In mijn geval zal dit van Nasa en Spotify zijn. Deze URL verwisseld wanneer we andere soorten requests gaan gebruiken.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Hiernaast kunnen we ook kiezen wat voor een request we willen versturen (Get, Post, Put…). Zoals ik bij de vorige stap zei gaat vaak de URL ook veranderen als we een andere soort request nemen.



Bij sommige API’s is er ook een authenticatie nodig. Deze zijn meestal terug te vinden op de website zelf nadat je je hebt aangemeld. In het geval van de spotify API blijft de authenticatie code/token maar goed voor een aantal minuten en moet je nadien een nieuwe aanvragen.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

## Nasa

### Get-Testing met een specifiek Id

#### Correcte URL

We gebruiken het projectId: 17792

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

(Er staat zeer veel informatie in de body vandaar dat ik niet alles hierin heb gekopieerd)

Afbeelding met tekst

Automatisch gegenereerde beschrijving

#### Foutieve URL

Ik heb in plaats van projects, proje geschreven waardoor we een GET request doen op een url dat niet bestaat.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

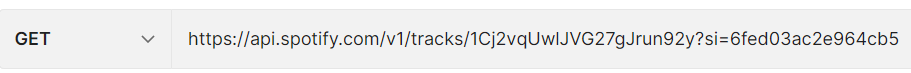
Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Op de eerste test gaf hij een FAIL terug omdat we geen code 200 hebben teruggekregen maar op de 2de test geeft hij natuurlijk wel een PASS omdat we gevraagd hebben of we een code 404 terugkrijgen en hier is dit waar.

## Spotify





Dus ik moest een Token aanvragen en toevoegen in postman:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Hierdoor kreeg ik wel het juiste nummer te zien.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

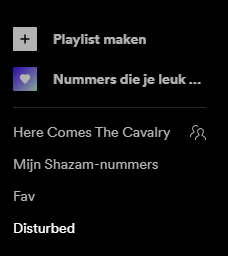
Voor verdere testing maak ik gebruik van spotify zijn playlist omdat zoals u op onderstaande afbeelding kan zien alle methodes aanwezig zijn:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

### Post-testing

In deze fase gaan we een playlist op spotify maken. Hieronder kan u zien welke playlists ik allemaal al op voorhand had.



Om een nieuwe playlist toe te voegen moeten we dus een POST uitvoeren waarbij we in de body meegeven wat de naam zal zijn en wat de beschrijving van deze playlist is. “Public”: false betekent dat alleen ik deze playlist ga kunnen bekijken en niemand anders.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

We gaan ook een test uitvoeren om te zien of de POST gelukt is of niet.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

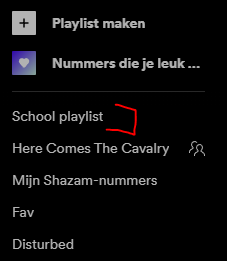
Nadat we dit hebben uitgevoerd zien we dat de POST met succes is beëindigd:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving



Nu dat de playlist aangemaakt is kunnen we doorgaan met het toevoegen van liedjes in de playlist. In dit geval ga ik The Sound of Silence van Disturbed toevoegen:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Als testen had ik gevraagd om een PASS terug te geven als de POST request gelukt was en ook als de reactie tijd minder dan 200ms was.

**Opmerking:**

Het toevoegen van een liedje ging minder vlot vooruit omdat de tutorial op udemy dat ik volgde niet werkte bij mij. Hierdoor verloor ik toch een paar uur aan het vinden van de fout. Dankzij een tutorial op YouTube die liedjes op een gelijkaardige maar toch verschillende manier toevoegde werkte het bij mij ook.

Bron: <https://www.youtube.com/watch?v=N34BM2CU_3g>

### Put-Testing

Voor put request te gaan testen heb ik adhv een put request de naam van mijn spotify playlist en de descriptie veranderd.

Afbeelding met tekst, monitor, schermafbeelding, scherm

Automatisch gegenereerde beschrijving

->

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Uit de testen blijkt dat de PUT request succesvol en onder de 200ms is gebeurd. Dit hebben we nadien ook gecheckt in spotify zelf.

Afbeelding met tekst, monitor, schermafbeelding, scherm

Automatisch gegenereerde beschrijving

### Delete-testing

Afbeelding met tekst, monitor, schermafbeelding, scherm

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Het door rood onderlijnde code is het id van The Sound of Silence. Hierdoor gaat hij dus alle liedjes met dit specifiek id verwijderen uit de playlist. Wat we zouden moeten bekomen is dus de playlist met enkel de liedjes ‘Demons Are a Girl’s Best Friend’ en ‘A Reason to Fight’.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst, schermafbeelding, scherm

Automatisch gegenereerde beschrijving

Hier ziet u dat ik het verwachte resultaat heb gekregen.

### Extra requests voor playlist aan te vullen

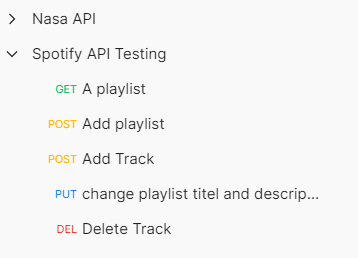
Voor dit gedeelte heb ik de requests en tests niet in mijn document gezet omdat deze ongeveer dezelfde zijn als bij de post-request die we een aantal pagina’s terug hebben gezien. Het enigste verschil met de vorige test is dat de uris (dit bevat de link/het id van de track) anders is omdat we ook steeds andere liedjes wouden toevoegen.

Afbeelding met tekst, monitor, zwart, scherm

Automatisch gegenereerde beschrijving

### Postman Collections voor spotify requests

Een collection is een soort mapje waaronder je alle request eenvoudig kan bijhouden. Hieronder ziet u de requests dat ik tot nu toe heb gemaakt en gebruikt.



### Slimme authenticatie token

Omdat al mijn requests in een collection staan is het veel handiger om mijn authenticatie vanuit mijn collection aan te vragen zodat ik dit niet voor elke request apart moet invullen.

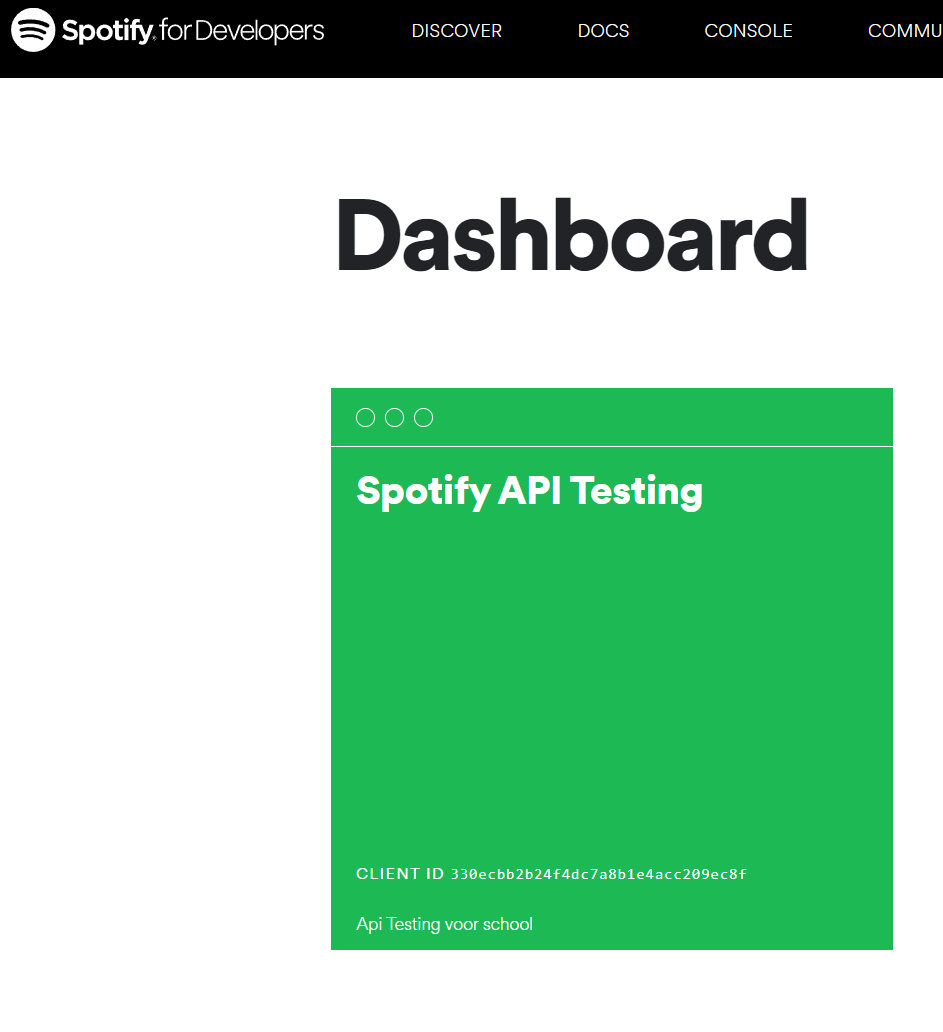
Voor deze authenticatie aan te vragen heb ik informatie nodig voor een aantal zaken zoals: Callback URL, Auth URL, Access Token URL, Client ID, Client Secret en Scope.



Afbeelding met tekst

Automatisch gegenereerde beschrijving

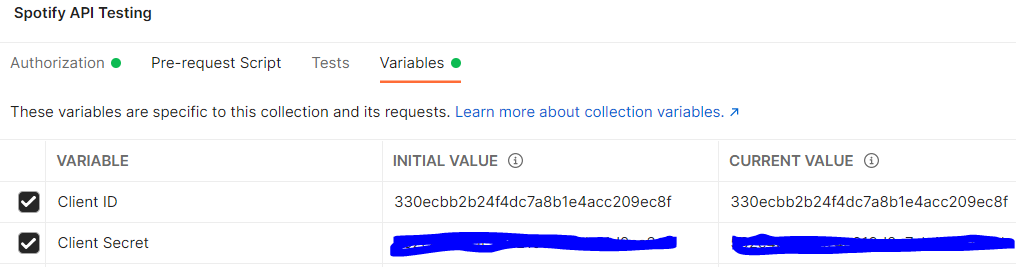
Voor mijn Client ID en Client Secret moest ik even rondzoeken op mijn spotify dashboard.



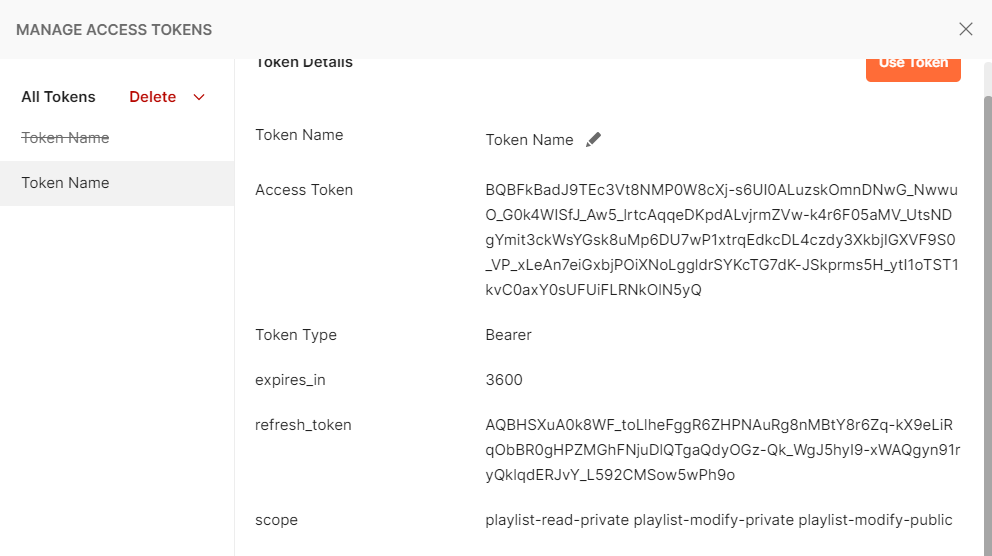
Moest ik Show client secret hebben opengedaan zou u mijn nummer daar ook zien maar dat heb ik voor veiligheidsredenen niet gedaan.

Afbeelding met tekst

Automatisch gegenereerde beschrijving



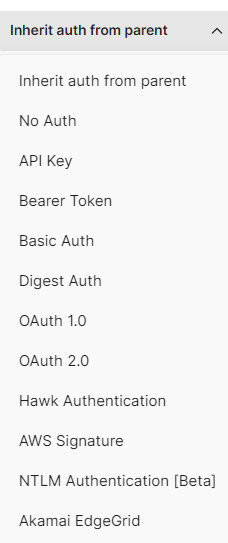
Nadat we al deze informatie hadden konden we op Get New Access Token drukken waarbij we de informatie kregen te zien van de token die we net hebben aangemaakt/gekregen. Nu drukken we op Use Token en dan kunnen we deze nieuwe token gebruiken voor 3600 seconden. Nadat deze vervallen is moeten we opnieuw een nieuwe token aanvragen.



Nu moeten we voor alle tests waar we OAuth 2.0 hadden gezet hadden, dit aanpassen naar Inherit auth from parent.

Afbeelding met tekst

Automatisch gegenereerde beschrijving



### Diepgaande tests

#### Get request

Voor de get request heb ik nu ook 2 extra tests toegevoegd. Dit zijn 2 tests die gaan kijken of de naam van de titel van de playlist dat ik opvraag correct is. Op lijn 9 maak ik een test die mij geen error zou mogen teruggeven omdat dit de juiste naam van de playlist is. Op lijn 14 zou dit wel een error/fout moeten geven want nu heb ik de foute naam gevraagd.

Postman zal dus gaan zoeken of de naam dat ik op lijn 11 en lijn 16 dezelfde is als in de response body.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Bij het uitvoeren kunnen we in de body zien dat de juiste naam: “School Playlist Api Testing” is.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Als we nu gaan kijken bij de Test results zien we dat ¾ tests gelukt zijn en 1tje gefaald is.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Dit komt door:

* De eerste test is gelukt omdat we de playlist hebben kunnen vinden op spotify.
* De tweede test is gelukt omdat de response tijd kleiner was dan 200ms.
* De derde test is gelukt omdat de naam dat ik vroeg dezelfde is als de naam van de playlist die ik opvroeg.
* De vierde test is mislukt omdat de naam “school” dat ik vroeg niet dezelfde is als de naam van de playlist die ik opvroeg.
* De vijfde test gaat kijken of “Demons are a girl’s best friend” in de body zit dat we ontvangen door onze get request te sturen. Demons are a girl’s best friend is een liedje dat normaal gezien in de playlist zit dus daarom dat hij een pass geeft.

Een laatste test die ik ga toevoegen is dezelfde als de vijfde test maar dan met een liedje dat niet in mijn playlist staat. Deze zou dus een fout moeten teruggeven. Bij deze laatste toevoeging heb ik ook alle test in een if else gezet. Deze if zegt dat wanneer we een response code van 200 hebben dan gaan we door met alle andere testen uit te voeren. Als dit niet zo is en er is dus geen playlist gevonden dan gaan we door naar de else. De else geeft dan in de console weer dat er geen playlist gevonden is.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving



Afbeelding met tekst

Automatisch gegenereerde beschrijving

Zoals verwacht hebben we 4 passes en 2 fails. De 2 fails komen omdat ik bij test 4 de foute naam opvraag en bij test 6 krijg ik een fail omdat het liedje “Christmas Truce” niet in mijn playlist staat. Hij geeft vervolgens bij de test results de hele body terug.

# Test automation

## Environment

### Uitleg

Voor automatische testen te maken gaan we beginnen met het maken van een nieuwe environment. Nadat ik deze environment heb gecreëerd moet ik hem ook als actief zetten.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

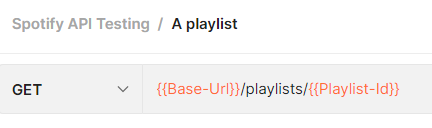
Automatisch gegenereerde beschrijving

In deze environment kunnen we verschillende variabelen steken die we vaak gaan gebruiken. In dit geval heb ik Base-Url en Playlist-Id erin gezet omdat ik deze elke keer ge gebruiken.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Nu dat ik deze variabelen heb gemaakt in mijn environment kan ik bij al mijn testen deze gebruiken.



Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

### Confirmatie

Voor te tonen dat dit ook werkt zal ik de GET request oproepen met deze variabelen:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

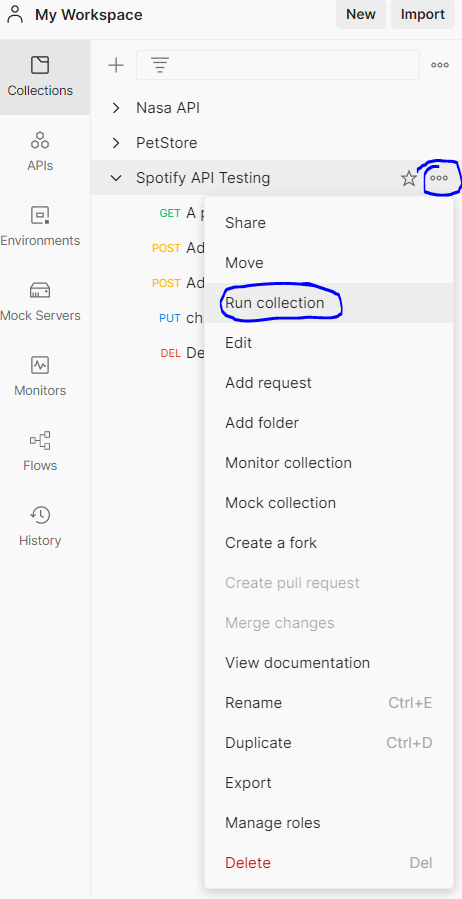
## Collections

Eerder heb ik al uitgelegd wat een collection is en hoe ik deze moet opzetten. Ook heb ik de authenticatie uitgelegd voor de collection.

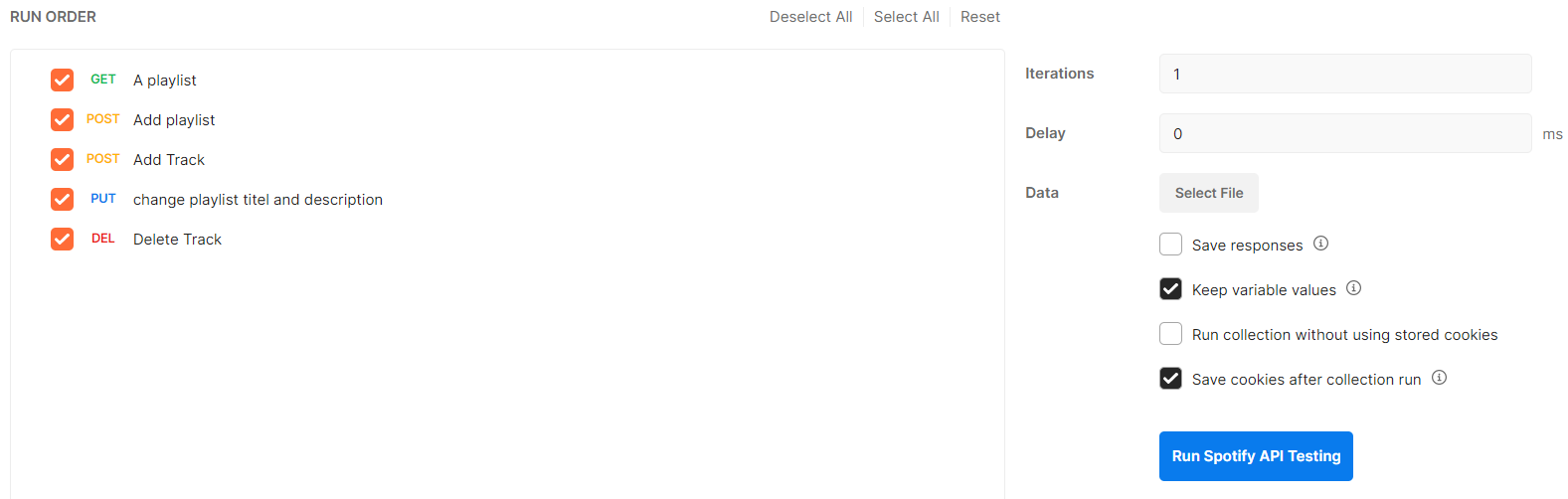
Nu ga ik door tot het runnen van deze collection waardoor al mijn tests in 1 automatische run worden uitgevoerd.

### Collection runner uitleg

Voor mijn ‘Spotify API Testing’-collection te runnen moest ik eerst op run collection drukken. Dit is te vinden door op de 3 bolletjes te drukken naast de collection naam.



Wanneer we dit doen zien we een scherm opengaan waar we al onze request kunnen zien.



Hierbij kan ik nog kiezen welke requests ik wil runnen en welke niet. Naast deze mogelijkheid heb ik ook de mogelijkheid om de volgorde aan te passen voor welke request er eerste komt en welke erna.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

In dit geval heb ik de PUT request naar boven geschoven en zal deze voor de ‘Add Track’ request uitgevoerd worden. Daarnaast heb ik ook de ‘Add playlist’ afgevinkt omdat ik niet nog een nieuwe playlist wil maken.

Aan de rechterkant kunnen we meer informatie vinden over hoe ik de collection run wil laten lopen.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Iterations geeft weer hoeveel keer ik de collection, met alle request erin, wil laten uitvoeren. In dit geval zal ik het 1 keer laten runnen.

Delay betekent hoeveel milliseconden er tussen elke request zal zitten.

Data dient voor het toevoegen van Data files dat eventueel nodig zijn tijdens de run.

Save response kan handig zijn als je de log wilt herbekijken maar dit kan inspelen op de snelheid van de run.

Keep variable values betekent dat als de variabelen van waarde zijn veranderd tijdens de run dan zullen deze ook veranderd blijven wanneer de run gedaan is.

‘Run collection without using stored cookies’ en ‘Save cookies after collection run’ spreken voor zich.

### Collection runner uitvoering

#### Get-request a playlist

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Afbeelding met tekst

Automatisch gegenereerde beschrijving

#### Put-request: change playlist title and description

Afbeelding met tekst

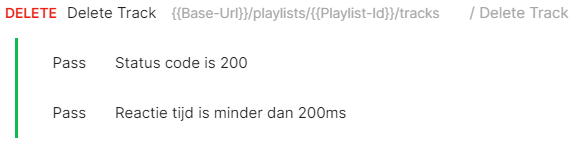
Automatisch gegenereerde beschrijving

#### Post-request: Add Track

Afbeelding met tekst

Automatisch gegenereerde beschrijving

#### Delete-request: Delete Track



Zoals we hierboven hebben kunnen zien is alles goed verlopen. Dit komt natuurlijk omdat we elke request al eerder 1 per 1 hadden getest om te zien of deze juist waren of niet. Het voordeel aan deze collection run is dat we niet voor elke request aparte runs gaan moeten uitvoeren maar nu kunnen we simpelweg vanuit 1 plaats op 1 knop drukken om alle requests dat ik wil uit te gaan voeren.

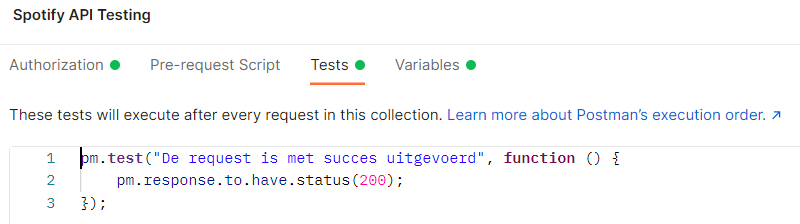
#### Overview

Afbeelding met tekst

Automatisch gegenereerde beschrijving

#### Extra

In een collection kunnen we ook tests schrijven die voor elke request zal gebruikt worden. In mijn geval heb ik hier de test die gaat kijken of een request de code 200 meegeeft ingezet.



Wanneer ik dit uitvoerde zagen we telkens bij elke request mijn algemene collection test terugkomen.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Zowel voor de GET als voor de PUT gaf hij een PASS. Voor de POST gaf hij spijtig genoeg geen PASS. Dit was al snel duidelijk omdat als je een POST request uitvoert en als deze ook PASSED geeft hij een code 201 terug. Dus om dit te vermijden heb ik de code veranderd naar:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Ik heb ook de test voor te controleren of de reactie tijd onder de 200ms is ook toegevoegd. Daarnaast heb ik deze 2 testen wel bij alle request nog laten staan dus u zal meerdere keren dezelfde test zien terugkomen. Normaal gezien zou ik dit wegdoen maar dan zou ik het u niet meer kunnen tonen in de demo.

Nu zou het resultaat moeten zijn dat hij bij elke test dat slaagt een PASS moet teruggeven.

Afbeelding met tekst, tafel

Automatisch gegenereerde beschrijving

Zoals ik had gedacht krijgen we nu wel overal een PASS waar dit zou moeten.