

Installatiehandleiding



Auteur **Serkan Ertas**
Datum **1 februari 2022**

Inhoudsopgave

Inleiding	3
Lijst van benodigdheden	4
Installatie en setup van de benodigde software	5
Benodigde software installeren	5
Het project opzetten	7
Gebruikersrollen en inloggegevens	13
REST endpoints	14

Inleiding

Dit is de installatiehandleiding voor het Full stack Filmspring project.

Dit document bevat informatie over de benodigde software, installatie en setup proces, inloggegevens en REST endpoints.

Lijst van benodigdheden

Systeemvereisten (aanbevolen)

Geheugen	8GB
CPU	Elke moderne processor
Besturingssysteem	Windows x86-64 of macOS (in deze installatiehandleiding nemen we Windows als richtlijn)
Internetverbinding	vereist

Downloads

IntelliJ IDEA Community Edition 2021.3.2 (of hoger) download:

<https://www.jetbrains.com/idea/download/download-thanks.html?platform=windows&code=IIC>

JDK 17.02 download:

https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi

PostgreSQL 14.1 download:

<https://www.enterprisedb.com/postgresql-tutorial-resources-training?uuid=db55e32d-e9f0-4d7c-9aef-b17d01210704&campaignId=7012J000001NhszQAC>

pgAdmin 4 v6.4 download:

<https://ftp.postgresql.org/pub/pgadmin/pgadmin4/v6.4/windows/pgadmin4-6.4-x64.exe>

Postman 9.11.0 (of hoger) download: <https://dl.pstmn.io/download/latest/win64>

Node.js 16.13.2 (of hoger) download:

<https://nodejs.org/dist/v16.13.2/node-v16.13.2-x64.msi>

NPM is inbegrepen bij de Node.js installatie

Installatie en setup van de benodigde software

Benodigde software installeren

JDK

1. Wanneer de download klaar is, open het .msi bestand en druk op **Next**.
2. Druk nogmaals op **Next**.
3. Als Windows om toestemming vraagt, druk op **Yes**. De JDK wordt nu geïnstalleerd.
4. Wanneer de installatie voltooid is, druk op **Close**.

Node.js

1. Wanneer de download klaar is, open het .msi bestand en druk op **Next**.
2. Druk nogmaals op **Next**.
3. **Accepteer** de licentieovereenkomst en druk op **Next**.
4. Druk nogmaals op **Next**.
5. Alle onderdelen worden geïnstalleerd; druk op **Next**.
6. Houd het vakje *Automatically install the necessary tools ...* leeg. Druk op **Next**.
7. Druk op **Install** om de software te installeren. Als Windows om toestemming vraagt, druk op **Yes**.
8. Node.js is nu geïnstalleerd.

PostgreSQL

1. Wanneer de download klaar is, open het .exe bestand en druk op **Next**. Als Windows eerst om toestemming vraagt, druk op **Yes**.
2. Druk nogmaals op **Next**.
3. Vink alles aan **behalve Stack Builder** en druk op **Next**.
4. Druk nogmaals op **Next**.
5. Voer in de velden:
 - a. *Password:* **password**
 - b. *Retype password:* **password**Druk op **Next**.
6. Verander de *port* naar **5432** als iets anders is ingesteld en druk op **Next**.
7. Laat de *Locale* staan op **[Default locale]** en druk op **Next**.
8. Vervolgens is er een overzicht van alle ingestelde instellingen. Druk op **Next**.
9. Druk nogmaals op **Next**. PostgreSQL zal nu worden geïnstalleerd.

10. Wanneer de installatie voltooid is, druk op **Finish**. PostgreSQL is nu geïnstalleerd.

pgAdmin

1. Wanneer de download klaar is, open het .exe bestand en kies voor **Install for me only {recommended}**.
2. Druk op **Next**.
3. **Accepteer** de licentieovereenkomst en druk op **Next**.
4. Druk nogmaals op **Next**.
5. Druk nogmaals op **Next**.
6. Druk op **Install**. PgAdmin wordt nu geïnstalleerd.
7. Wanneer de installatie voltooid is, druk op **Finish**. PgAdmin is nu geïnstalleerd.

IntelliJ IDEA

1. Wanneer de download klaar is, open het .exe bestand en druk op **Next**. Als Windows eerst om toestemming vraagt, druk op **Yes**.
2. Druk op **Next**.
3. Druk nogmaals op **Next**.
4. De vakjes kun je leeg laten. Als je het software icoon op je desktop wil voor snellere toegang, kun je *Create Desktop Shortcut* aanvinken. Druk op **Next**.
5. Druk vervolgens op **Install**.
6. IntelliJ IDEA is nu geïnstalleerd. Druk op **Finish**.

Postman

1. Open het **Postman** bestand. Installatie is niet nodig.
2. Een account aanmaken is niet nodig. Druk op *Skip signing in* e.d.

Het project opzetten

pgAdmin: database importeren en server opzetten

Nu gaan we PostgreSQL database opzetten met test data met behulp van de pgAdmin interface.

1. Open de pgAdmin 4 applicatie.
2. Je wordt gevraagd een wachtwoord in te stellen. Voer in het veld *password*: **password** in.
3. Vouw het Servers menu uit. Je ziet de *PostgreSQL 14* server. Klik met de rechtermuisknop op *PostgreSQL 14* en klik op **Properties**.
4. Navigeer naar het *Connection* tabblad en wijzig de velden naar:
 - a. *Hostname/address*: **localhost**
 - b. *Port*: **5432**
 - c. *Username*: **postgres**
5. Druk vervolgens op **Save** om deze instellingen op te slaan. Vervolgens moet je de server afsluiten en weer opstarten. Dit doe je door met de rechtermuisknop op *PostgreSQL 14* te drukken en te kiezen voor **Disconnect Server**. Vervolgens start je de server weer op door nogmaals op *PostgreSQL 14* te drukken en daarna **Connect Server**.

Nu gaan we een database importeren.

1. Eerst moeten we de binary path voor PostgreSQL opzetten. Hiervoor gaan we naar **File - Preferences** bovenaan de applicatie.
2. In het menu, klik op **Binary paths** onder het kopje *Paths*.
3. Scroll naar beneden. Onderaan zie je de database server **PostgreSQL 14**. Druk op *Select binary path...* voor deze rij en plak het volgende erin:
 - a. C:\Program Files\PostgreSQL\14\bin\Druk op **Save**.
4. Nu gaan we de database maken. Onder het kopje *PostgreSQL 14 > Databases(1)* zien we al een database: *postgres*. Wij gaan er nog een toevoegen door met de rechtermuisknop op *Databases (1)* te klikken en vervolgens op **Create - Database....**
5. Het veld *Database* is momenteel leeg. Typ in het *Database* veld **Filmspring** en druk op **Save**.
6. Nu zien wij ook de database **Filmspring** onder het kopje *Databases (2)*. Klik met de rechtermuisknop op **Filmspring** en vervolgens op **Restore....**
7. Wij zien nu 4 velden onder het kopje *General*. Voor het tweede veld, *Filename*, moeten wij klikken op de drie puntjes en daarna navigeren wij naar de locatie van het *compressed_testdata.sql* bestand. Wanneer je het bestand gevonden hebt, klik je op **Restore**.

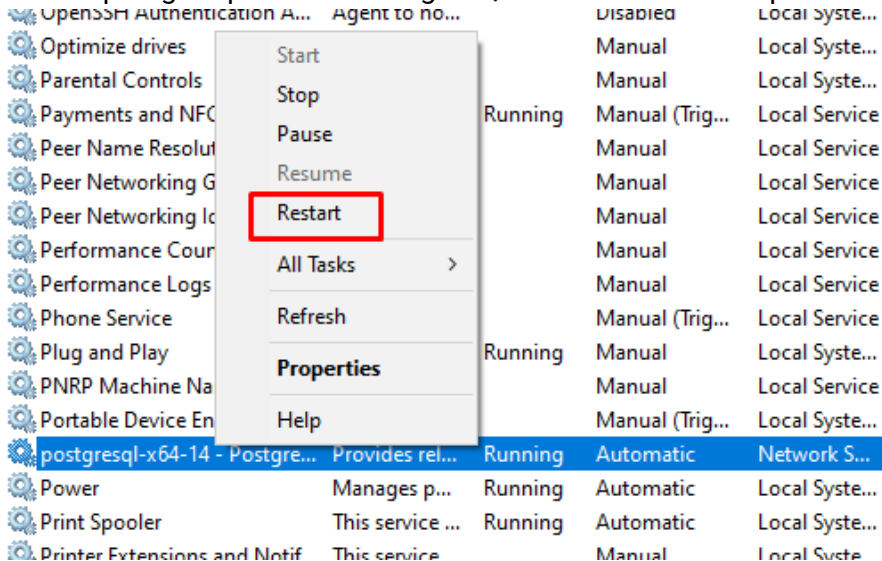
Nu hebben wij onze database gevuld met de benodigde testdata. Zorg ervoor dat de server actief blijft. Wij gaan nu verder met het instellen van Spring boot.

Spring boot applicatie draaien met IntelliJ IDEA

1. Open IntelliJ IDEA Community Edition en accepteer de licentieovereenkomst.
2. Klik op **Don't Send**. De applicatie zal nu opstarten.
3. In het midden van het scherm zie je 3 knoppen. Druk op **Open**.
4. Navigeer naar en open het **pom.xml** bestand in de *Filmspring-server* map.
5. Klik op **Open as Project**. Druk op **Trust this project** als hier om gevraagd wordt.
6. Maven zal bezig zijn met alle nodige dependencies te indexeren en klaar te zetten. Rechtsonder kun je de voortgang bekijken. Wanneer alles gereed is, navigeer naar **src - main - java - nl.serkanertas.filmspringserver - FilmSpringServerApplication**. Open dit bestand.
7. Bovenaan wordt er gevraagd om een *SDK* op te zetten. Druk op **Setup SDK** en kies voor **17**. Wederom kan het een tijdje duren terwijl het programma alle benodigdheden opzet. Wanneer dit gereed is, klik je op de groene pijl en daarna op **Run 'FilmspringServ....main()'**. Nu draait zowel de Spring boot applicatie als de database server.

Troubleshooting: PostgreSQL server start niet

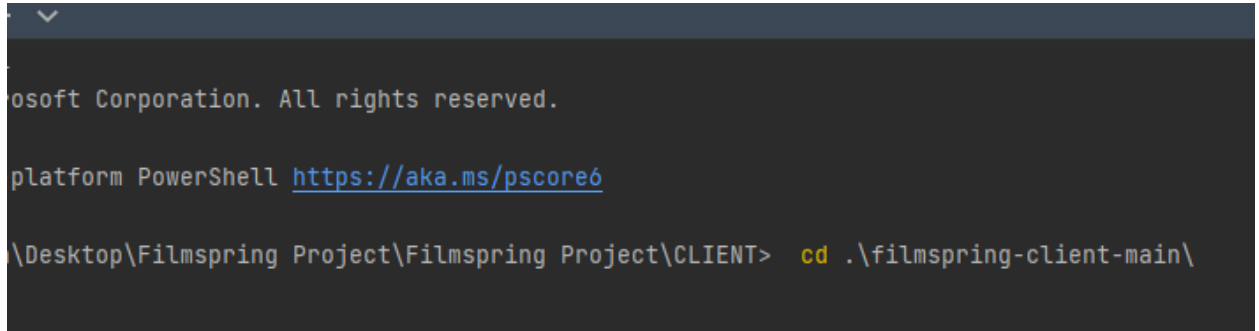
Lukt het niet om de server op te starten via pgAdmin? Bekijk deze oplossing:

Stap 1	<p>Druk op de Windows + R toetsen. Dit opent een dialoogvenster. Kopieer en plak de volgende locatie: <i>C:\Program Files\PostgreSQL\14\data\postgresql.conf</i> en druk op enter. Open het bestand met Notepad of een andere tekstbewerker.</p>
Stap 2	<p>Scroll omlaag naar "CONNECTIONS AND AUTHENTICATION". Het kan zijn dat <i>port</i> een waarde van 8080 heeft. Dit is niet de bedoeling. Verander deze waarde naar 5432:</p> <pre>#----- # CONNECTIONS AND AUTHENTICATION #----- # - Connection Settings - listen_addresses = '*' # what IP address(es) to listen on; # comma-separated list of addresses; # defaults to 'localhost'; use '*' for all # (change requires restart) port = 5432 # (change requires restart) max_connections = 100 # (change requires restart)</pre> <p>Sla het bestand op.</p>
Stap 3	<p>Druk vervolgens op de Windows toets en zoek naar "Services". Open dit programma.</p>
Stap 4	<p>Scroll naar "postgresql-x64-14 - PostgreSQL Server 14" en klik op Restart.</p> 
Stap 5	<p>Probeer het opnieuw om de server op te starten in pgAdmin 4.</p>

Frontend React applicatie draaien met IntelliJ IDEA

Zorg ervoor dat zowel de Spring boot applicatie en database aan het draaien zijn.

1. In IntelliJ IDEA, klik bovenaan op **File - Open...**, en open de map *CLIENT* in een **New Window**.
2. Klik op de *Terminal* knop onderaan IntelliJ en plak het volgende commando in de terminal: **cd .\filmspring-client-main**.



```
Microsoft Corporation. All rights reserved.  
  
platform PowerShell https://aka.ms/pscore6  
  
\Desktop\Filmspring Project\Filmspring Project\CLIENT> cd .\filmspring-client-main\
```

Typ vervolgens **npm install**. Dit commando kan een tijdje duren.

3. Wanneer het programma klaar is, typ **npm start**.

Troubleshooting: HTTPS self-signed certificate waarschuwingen

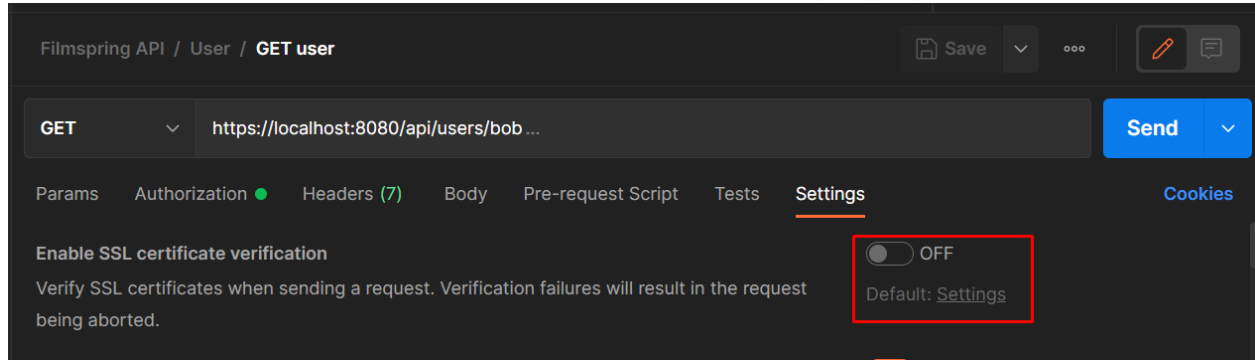
De backend wordt beveiligd met een 'Self-signed' HTTPS certificaat. Het is dus geen echt certificaat dat uitgegeven is door een autoriteit. Om deze reden zullen browsers zoals Firefox en Chrome beveiligingsmaatregelen nemen waardoor de website niet goed (of helemaal niet) kan functioneren zoals gewenst.

Voor **Firefox** betekent dit meestal een waarschuwing. Voor **Chrome** zullen we een instelling moeten aanpassen:

Stap 1	Kopieer en plak deze link in de navigatiebalk van Chrome: <code>chrome://flags/#allow-insecure-localhost</code>
Stap 2	<p>Je krijgt de optie 'Allow invalid certificates for resources loaded from localhost.' te zien. Klik op Enable.</p>  <p>The screenshot shows the Chrome flags page with the flag 'Allow invalid certificates for resources loaded from localhost.' highlighted in yellow. Below the flag name, it says 'Allows requests to localhost over HTTPS even when an invalid certificate is presented. - Mac, Windows, Linux, Chrome OS, Android'. At the bottom right, there is a blue button labeled 'Enabled' with a dropdown arrow. The URL '#allow-insecure-localhost' is visible at the bottom left of the flag card.</p>

Troubleshooting: Foutmelding Postman HTTPS

Er kan een foutmelding ontstaan tijdens het gebruik van Postman, omdat het HTTPS certificaat self-signed is. Om dit op te lossen, moeten we *Enable SSL certificate verification* in **Settings** uitzetten:



Gebruikersrollen en inloggegevens

Gebruikersrollen

Rol	Toelichting
ROLE_ADMIN	Deze gebruiker mag alle methodes aanroepen.
ROLE_OWNER-GROUP-X	Deze gebruiker mag de groep en gebruikers in zijn groep verwijderen. Ook mag hij het groepsprofiel bewerken.
ROLE_MEMBER-GROUP-X	Deze gebruiker mag andere gebruikers uitnodigen om bij zijn groep aan te sluiten. Deze gebruiker mag ook zijn groep verlaten.
ROLE_INVITED-X	Deze gebruiker mag een uitnodiging van een groep accepteren of weigeren.
ROLE_USER	Deze gebruiker mag zijn eigen profiel bijwerken en een nieuwe groep aanmaken, waarvan hij automatisch de group owner wordt.
<i>Hierbij is X het groeps ID nummer.</i>	

Inloggegevens

Gebruikersnaam	Onversleuteld Wachtwoord	Rol(len)
admin	admin123	<ul style="list-style-type: none">• ROLE_ADMIN• ROLE_USER
john	johnsmith123	<ul style="list-style-type: none">• ROLE_INVITED-37• ROLE_USER
alice1	alice123	<ul style="list-style-type: none">• ROLE_MEMBER-GROUP-37• ROLE_USER
bob	bobdoe123	<ul style="list-style-type: none">• ROLE_OWNER-GROUP-37• ROLE_MEMBER-GROUP-37• ROLE_USER

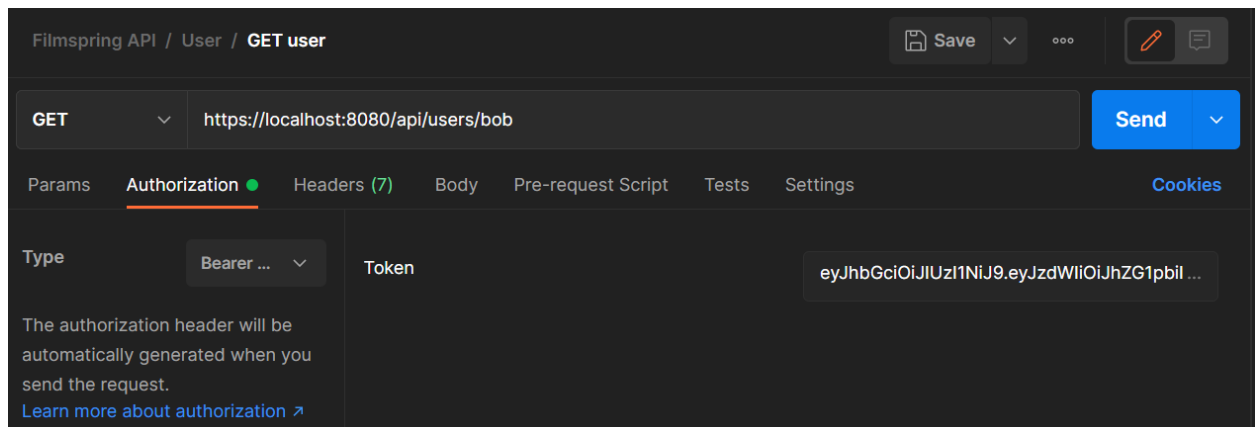
REST endpoints

Aandachtspunten

- De basis URL is: **https://localhost:8080/api**.
- Er wordt aangenomen dat bij elke request (behalve bij **/users/login**), een geldige JWT token aanwezig is in de **Authentication** header en dat de eigenaar van de JWT token de juiste rollen heeft bij de betreffende request.
- De maximale grootte van een bestand dat je kan uploaden is 5 MB.

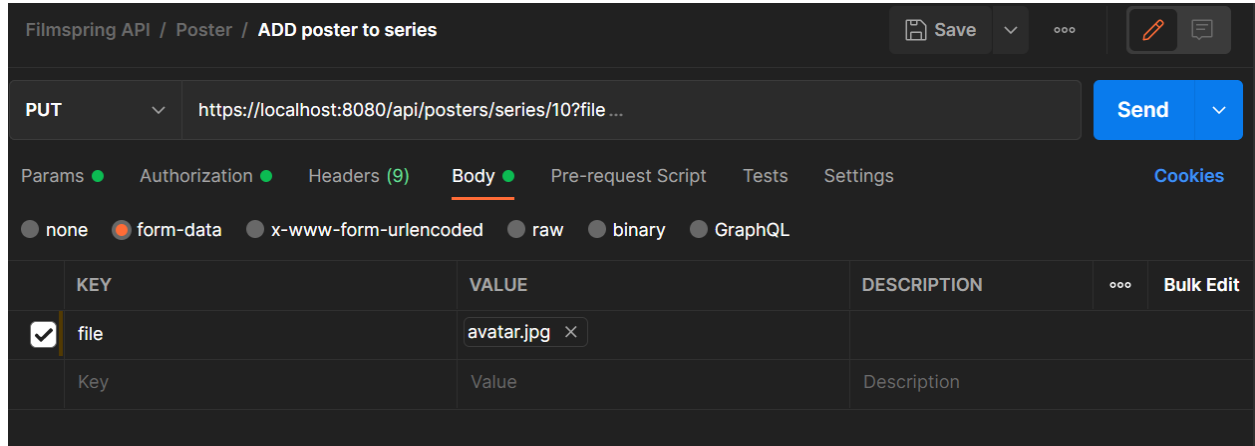
JWT-token toevoegen aan request

1. Navigeer naar **Authorization**.
2. Bij *Type* kies voor **Bearer Token**.
3. Plak jouw token in het *Token* veld.



Bestanden uploaden via Postman

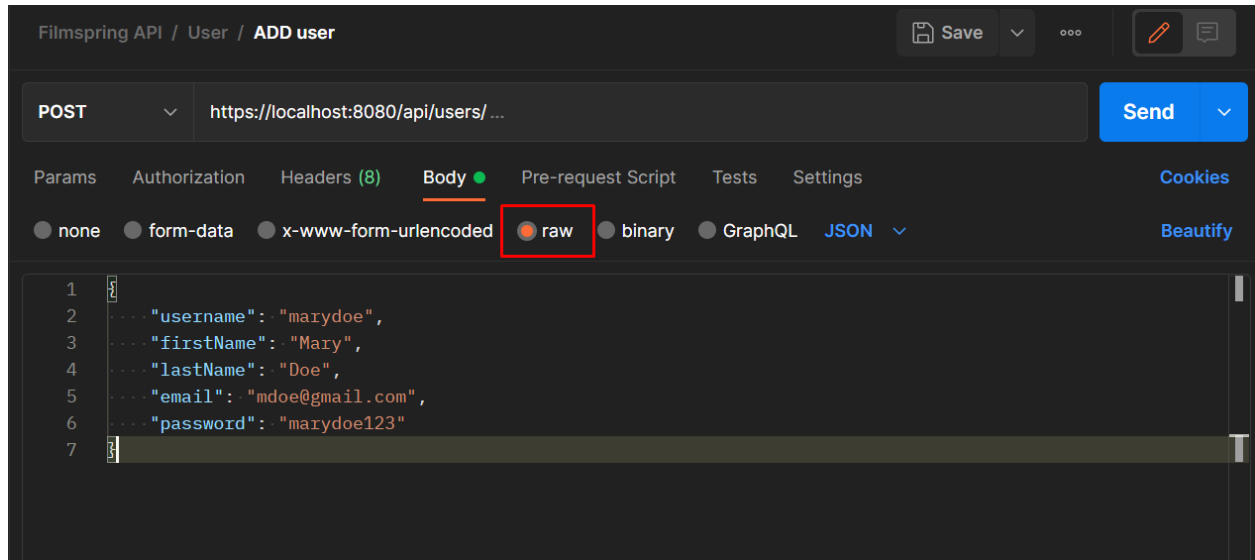
1. Navigeer naar **Body**.
2. Kies voor form-data.
3. Kies het bestand dat je wil uploaden.



JSON payload versturen via Postman

Sommige requests verwachten JSON data. Dit is hoe je JSON toevoegt aan een request:

1. Navigeer naar **Body**.
2. Kies voor raw.
3. Plak de JSON payload bij de bijbehorende request in het tekstveld.



Alle endpoints per controller

AuthController

Login gebruiker

POST /users/login (verwacht: 200 OK)

Hier logt de gebruiker in. Bij een succesvolle request wordt er een JWT-token gestuurd die gebruikt kan worden bij alle andere requests.

JSON payload

```
{  
  "username": "admin",  
  "password": "admin123"  
}
```

ActorController

Krijg alle acteurs

GET /actors (verwacht: 200 OK)

Alle bestaande acteurs worden weergegeven.

Krijg acteur door ID

GET /actors/{actor_id} (verwacht: 200 OK)

Krijg een acteur op basis van zijn of haar ID.

Voeg acteur toe aan film

PUT /actors/films/{film_id}/actors/{actor_id} (verwacht: 204 NO CONTENT)

Voeg een acteur toe aan een film.

Verwijder acteur van film

DELETE /actors/films/{film_id}/actors/{actor_id} (verwacht: 204 NO CONTENT)

Verwijder een acteur van een film.

Voeg acteur toe aan serie

PUT /actors/series/{series_id}/actors/{actor_id} (verwacht: 204 NO CONTENT)

Voeg een acteur toe aan een serie.

Verwijder acteur van Serie

DELETE /actors/series/{series_id}/actors/{actor_id} (verwacht: 204 NO CONTENT)

Verwijder een acteur van een serie.

AvatarController

Krijg profielfoto van gebruiker

GET /avatars/users/{user_id} (verwacht: 200 OK)

Krijg de profielfoto van een gebruiker in Base64.

Upload profielfoto voor gebruiker

PUT /avatars/users/{user_id}?file (verwacht: 200 OK)

Upload een foto bestand als profielfoto voor een gebruiker.

Verwijder profielfoto van gebruiker

DELETE /avatars/users/{user_id} (verwacht: 204 NO CONTENT)

Verwijder de profielfoto van een gebruiker. De gebruiker krijgt dan de standaard profielfoto.

Krijg profielfoto van groep

GET /avatars/groups/{group_id} (verwacht: 200 OK)

Krijg de profielfoto van een groep in Base64.

Upload profielfoto voor groep

PUT /avatars/groups/{group_id}?file (verwacht: 200 OK)

Upload een foto bestand als profielfoto voor een groep.

Verwijder profielfoto van groep

DELETE /avatars/groups/{group_id} (verwacht: 204 NO CONTENT)

Verwijder de profielfoto van een groep. De groep krijgt dan de standaard profielfoto.

FilmController

Krijg film door ID

GET /films/{film_id} (verwacht: 200 OK)

Krijg een film op basis van ID.

Krijg film door zoekterm

GET /films?search={query} (verwacht: 200 OK)

Krijg een film of meerdere films op basis van een zoekterm.

Krijg gebruikersbeoordeling van film

GET /films/{film_id}/rating/{user_id} (verwacht: 200 OK)

Krijg de beoordeling van een gebruiker voor een film.

Update gebruikersbeoordeling van film

PUT /films/{film_id}/rating/{user_id} (verwacht: 200 OK)

Update de beoordeling van een gebruiker voor een film. Rating moet tussen de 1.0 en 5.0 zijn.

JSON payload

```
{  
  "rating": 3.5  
}
```

Krijg alle film entiteiten

GET /films/raw (verwacht: 200 OK)

Krijg alle film entiteiten.

Krijg film entiteit door ID

GET /films/raw/{film_id} (verwacht: 200 OK)

Krijg een film entiteit op basis van ID.

Krijg alle acteurs van film

GET /films/{film_id}/actors (verwacht: 200 OK)

Krijg alle acteurs van een film.

GroupController

Krijg groep door ID

GET /groups/{group_id} (verwacht: 200 OK)

Krijg een groep op basis van ID.

Krijg alle groepen

GET /groups (verwacht: 200 OK)

Krijg alle groep entiteiten.

Krijg groep entiteit door ID

GET /groups/raw/{group_id} (verwacht: 200 OK)

Krijg een groep entiteit op basis van ID.

Maak nieuwe groep aan

POST /groups (verwacht: 201 CREATED)

Maak een nieuwe groep aan.

JSON payload

```
{  
  "groupName": "ExampleGroup"  
}
```

Verwijder groep door ID

DELETE /groups/{group_id} (verwacht: 204 NO CONTENT)

Verwijder een groep.

Krijg alle gebruikers van groep

GET /groups/{group_id}/users (verwacht: 200 OK)

Krijg alle gebruikers binnen een groep.

Voeg gebruiker toe aan groep

PUT /groups/{group_id}/users/{user_id} (verwacht: 200 OK)

Voeg een gebruiker toe, dus zonder uitnodiging, aan een groep.

Verwijder gebruiker van groep

DELETE /groups/{group_id}/users/{user_id} (verwacht: 204 NO CONTENT)

Verwijder een gebruiker van een groep.

Verlaat groep

DELETE /groups/{group_id}/users/{user_id}/leave (verwacht: 204 NO CONTENT)

Verlaat de groep als gebruiker.

Verzend groep uitnodiging naar gebruiker

PUT /groups/{group_id}/invite/{user_id} (verwacht: 200 OK)

Verzend een groep uitnodiging naar een gebruiker.

Accepteer groep uitnodiging

PUT /groups/{group_id}/invite/{user_id}/accept (verwacht: 200 OK)

Accepteer een groep uitnodiging en word lid van die groep.

Weiger groep uitnodiging

DELETE /groups/{group_id}/invite/{user_id}/reject (verwacht: 200 OK)

Weiger een groep uitnodiging en word niet lid van die groep.

MediaGroupsController

Voeg film toe aan groep Planned

PUT /groups/{group_id}/films/planned/{film_id} (verwacht: 200 OK)

Voeg een film toe aan de Planned groepslijst.

Verwijder film van groep Planned

DELETE /groups/{group_id}/films/planned/{film_id} (verwacht: 204 NO CONTENT)

Verwijder een film van de Planned groepslijst.

Voeg serie toe aan groep Planned

PUT /groups/{group_id}/series/planned/{series_id} (verwacht: 200 OK)

Voeg een serie toe aan de Planned groepslijst.

Verwijder serie van groep Planned

DELETE /groups/{group_id}/series/planned/{series_id} (verwacht: 204 NO CONTENT)

Verwijder een serie van de Planned groepslijst.

MediaUsersController

Voeg film toe aan gebruiker Watched

PUT /users/{user_id}/films/watched/{film_id} (verwacht: 200 OK)

Voeg een film toe aan de Watched gebruikerslijst.

Verwijder film van gebruiker Watched

DELETE /users/{user_id}/films/watched/{film_id} (verwacht: 204 NO CONTENT)

Verwijder een film van de Watched gebruikerslijst.

Voeg film toe aan gebruiker Planned

PUT /users/{user_id}/films/planned/{film_id} (verwacht: 200 OK)

Voeg een film toe aan de Planned gebruikerslijst.

Verwijder film van gebruiker Planned

DELETE /users/{user_id}/films/planned/{film_id} (verwacht: 204 NO CONTENT)

Verwijder een film van de Planned gebruikerslijst.

Voeg film toe aan gebruiker Favorites

PUT /users/{user_id}/films/favorites/{film_id} (verwacht: 200 OK)

Voeg een film toe aan de Favorites gebruikerslijst.

Verwijder film van gebruiker Favorites

DELETE /users/{user_id}/films/favorites/{film_id} (verwacht: 204 NO CONTENT)

Verwijder een film van de Favorites gebruikerslijst.

Voeg serie toe aan gebruiker Watched

PUT /users/{user_id}/series/watched/{series_id} (verwacht: 200 OK)

Voeg een serie toe aan de Watched gebruikerslijst.

Verwijder serie van gebruiker Watched

DELETE /users/{user_id}/series/watched/{series_id} (verwacht: 204 NO CONTENT)

Verwijder een serie van de Watched gebruikerslijst.

Voeg serie toe aan gebruiker Planned

PUT /users/{user_id}/series/planned/{series_id} (verwacht: 200 OK)

Voeg een serie toe aan de Planned gebruikerslijst.

Verwijder serie van gebruiker Planned

DELETE /users/{user_id}/series/planned/{series_id} (verwacht: 204 NO CONTENT)

Verwijder een serie van de Planned gebruikerslijst.

Voeg serie toe aan gebruiker Favorites

PUT /users/{user_id}/series/favorites/{series_id} (verwacht: 200 OK)

Voeg een serie toe aan de Favorites gebruikerslijst.

Verwijder serie van gebruiker Favorites

DELETE /users/{user_id}/series/favorites/{series_id} (verwacht: 204 NO CONTENT)

Verwijder een serie van de Favorites gebruikerslijst.

PosterController

Krijg poster van film door ID

GET /posters/films/{film_id} (verwacht: 200 OK)

Krijg de poster van een film op basis van ID.

Voeg poster toe aan film

PUT /posters/films/{film_id}?file (verwacht: 200 OK)

Upload een foto bestand als poster voor een film.

Verwijder poster van film

DELETE /posters/films/{film_id} (verwacht: 204 NO CONTENT)

Verwijder de poster van een film.

Krijg poster van serie door ID

GET /posters/series/{series_id} (verwacht: 200 OK)

Krijg de poster van een serie op basis van ID.

Voeg poster toe aan serie

PUT /posters/series/{series_id}?file (verwacht: 200 OK)

Upload een foto bestand als poster voor een serie.

Verwijder poster van serie

DELETE /posters/series/{series_id} (verwacht: 204 NO CONTENT)

Verwijder de poster voor een serie.

SeriesController

Krijg serie door ID

GET /series/{series_id} (verwacht: 200 OK)

Krijg een serie op basis van ID.

Krijg serie door zoekterm

GET /series?search={query} (verwacht: 200 OK)

Krijg een serie of meerdere series op basis van een zoekterm.

Krijg gebruikersbeoordeling van serie

GET /series/{series_id}/rating/{user_id} (verwacht: 200 OK)

Krijg de gebruikersbeoordeling van een gebruiker voor een serie.

Update gebruikersbeoordeling van serie

PUT /series/{series_id}/rating/{user_id} (verwacht: 200 OK)

Update de beoordeling van een gebruiker voor een serie. Rating moet tussen de 1.0 en 5.0 zijn.

JSON payload

```
{  
  "rating": 3.5  
}
```

Krijg alle serie entiteiten

GET /series/raw (verwacht: 200 OK)

Krijg alle serie entiteiten.

Krijg serie entiteit door ID

GET /series/raw/{series_id} (verwacht: 200 OK)

Krijg een serie entiteit op basis van ID.

Krijg alle acteurs in serie

GET /series/{series_id}/actors (verwacht: 200 OK)

Krijg alle acteurs in een serie.

UserController

Krijg gebruiker door ID

GET /users/{user_id} (verwacht: 200 OK)

Krijg een gebruiker op basis van ID.

Maak nieuwe gebruiker aan

POST /users (verwacht: 201 CREATED)

Creëer een nieuw account met JSON data. "lastName" is niet verplicht. Er zal een foutmelding getoond worden als de username en/of email al bestaat.

JSON payload

```
{
  "username": "marydoe",
  "firstName": "Mary",
  "lastName": "Doe",
  "email": "mdoe@gmail.com",
  "password": "marydoe123"
}
```

Update gebruiker gegevens

PATCH /users/{user_id}/account (verwacht: 204 NO CONTENT)

Update de account en profielgegevens van een gebruiker. Dit is beter te doen via de frontend, waar alle relevante velden gekoppeld zijn, in plaats van afzonderlijk met gewone JSON.

JSON payload

```
{
  "firstName": "Alisee",
  "lastName": "Mountain",
  "bio": "Hi I'm Alisee and this is my updated bio.",
  "hideMediaPreference": true,
  "email": "alm@email.com",
  "oldPassword": "alice123",
  "newPassword": "alisee123"
}
```

Verwijder gebruiker account

*(Omdat het wachtwoord verzonden wordt via JSON, gaat het verwijderen van een account met een POST-request en niet met een DELETE-request.)

POST /users/{user_id}/account (verwacht: 204 NO CONTENT)

Verwijder de account van een gebruiker. Hiervoor is een wachtwoord vereist, gegeven met JSON.

JSON payload

```
{  
  "password": "wachtwoord van gebruiker hier"  
}
```

Krijg gebruiker door zoekterm

GET /users?search={query} (verwacht: 200 OK)

Krijg een gebruiker of meerdere gebruikers op basis van een zoekterm.

Krijg alle gebruiker entiteiten

GET /users/raw (verwacht: 200 OK)

Krijg alle gebruiker entiteiten.

Logout gebruiker

*(Basis-URL is hier niet van toepassing.)

POST https://localhost:8080/logout (verwacht: 200 OK)

Log de gebruiker uit door een POST-request te sturen met bovenstaande URL. Het logout proces wordt behandeld door Spring boot.

ExceptionHandler

Voor de ExceptionController zijn geen endpoints aanwezig. Wanneer er een fout optreedt wordt deze controller automatisch aangeroepen.