

# Installatiehandleiding

# **LibraryManager**

---

Marjolijn Muller-Polle  
augustus 2025



## Inleiding

Deze handleiding is opgesteld om je te begeleiden bij de installatie en configuratie van de LibraryManager-applicatie. LibraryManager is een RESTful API, gebouwd met Spring Boot, die dient als het backend-systeem voor een bibliotheekbeheersysteem.

Door dit document te volgen, lees je welke benodigdheden je nodig hebt, hoe je de database correct instelt en hoe je het project kunt bouwen en starten. Daarnaast biedt het een gedetailleerd overzicht van de beschikbare API-endpoints en de bijbehorende toegangsrechten, waardoor je direct aan de slag kunt met het testen van de applicatie.



## Inhoudsopgave

<b>Inleiding.....</b>	<b>1</b>
<b>Inhoudsopgave.....</b>	<b>2</b>
<b>Benodigdheden.....</b>	<b>3</b>
<b>Installatie instructies.....</b>	<b>4</b>
<b>Testgebruikers.....</b>	<b>5</b>
Admin.....	5
Librarian.....	5
Member.....	5
<b>Rest-endpoints.....</b>	<b>6</b>
Authenticatie (inloggen).....	6
Profile/User.....	7
Boeken.....	11
Loans.....	15
Fine.....	18
Invoices.....	20
<b>Postman collecties.....</b>	<b>23</b>



## Benodigdheden

- [Java Development Kit \(JDK\) 17 of hoger](#)  
Zorg ervoor dat de juiste Java-versie is geïnstalleerd.
- [Apache Maven](#)  
Een projectmanagement- en bouwtool voor het compileren van de Java-code en het beheren van afhankelijkheden. De applicatie is afhankelijk van de volgende bibliotheken: Spring Boot, Spring Data JPA, Spring Security, JWT (jjwt), Lombok, en PDFBox.
- [PostgreSQL-database](#)  
De applicatie gebruikt een PostgreSQL-database. Zorg ervoor dat deze is geïnstalleerd en geconfigureerd.
- [IDE \(Integrated Development Environment\)](#)  
Een omgeving zoals IntelliJ IDEA, Eclipse of Visual Studio Code. IntelliJ wordt aanbevolen vanwege de uitstekende ondersteuning voor Spring Boot en Maven.
- [Postman](#)  
Een applicatie voor het testen van API-endpoints. Je zult de meegeleverde collectie gebruiken.

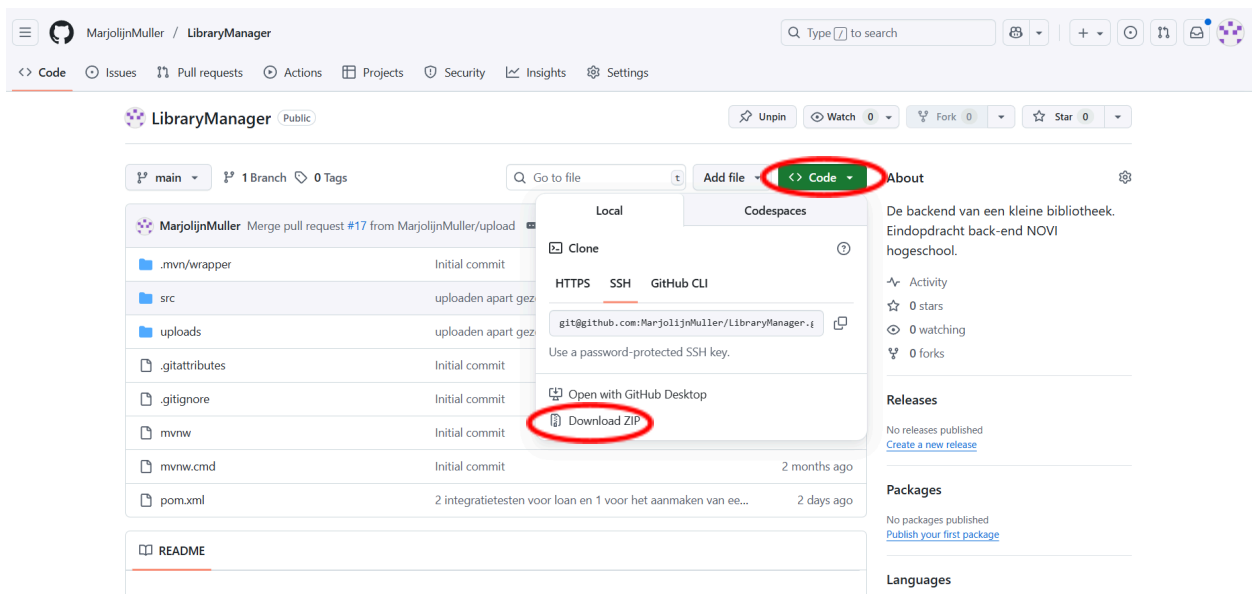
# Installatie instructies

Volg dit stappenplan om de applicatie te installeren en te draaien:

## 1. Download de repository

### a. Optie 1: Download als ZIP-bestand

1. Ga naar de GitHub-pagina in je webbrowser:  
<https://github.com/MarjolijnMuller/LibraryManager>
2. Klik op de groene knop <> Code en vervolgens op Download ZIP.

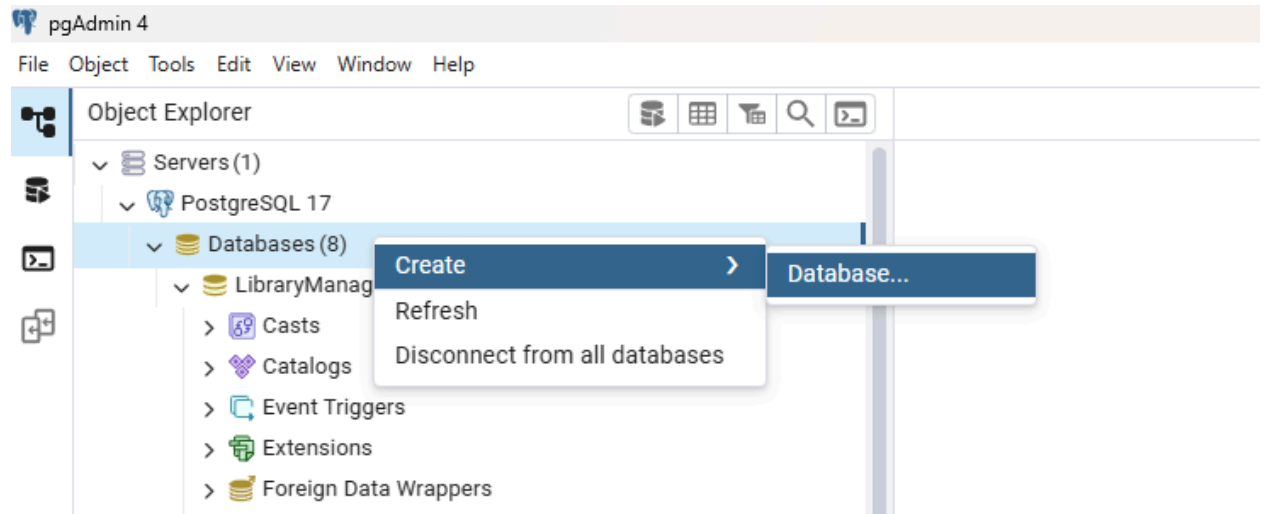


3. Pak het gedownloadde bestand uit op een gewenste locatie op je computer.

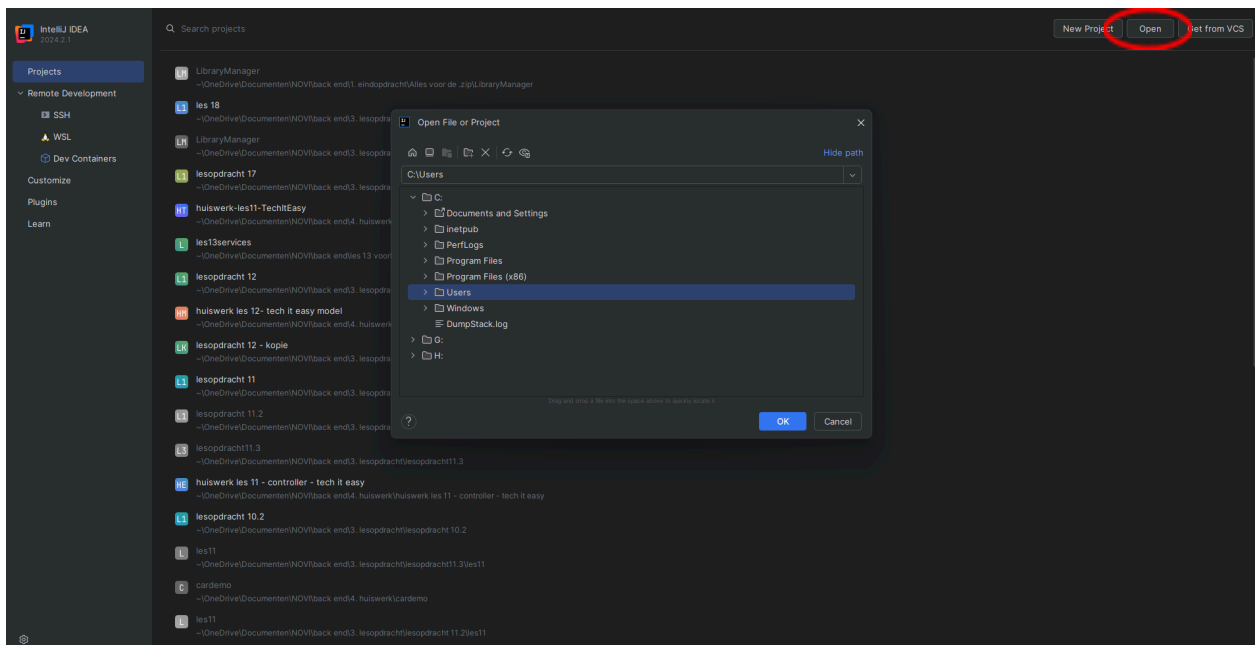
### b. Kloon de repository

1. Open een terminal of command prompt en kloon de GitHub-repository van het project:  
**Bash**  
`git clone https://github.com/MarjolijnMuller/LibraryManager.git`
2. **Configureer de database:**
  - Start je PostgreSQL-database.

- Maak een database aan met de naam **LibraryManager**.  
Om een database aan te maken, klik je met je rechtermuisknop op Databases en volg je de stappen op de afbeelding.



- Open je IDE en importeer het gedownload project als een Maven-project. De IDE zal automatisch de benodigde afhankelijkheden downloaden.

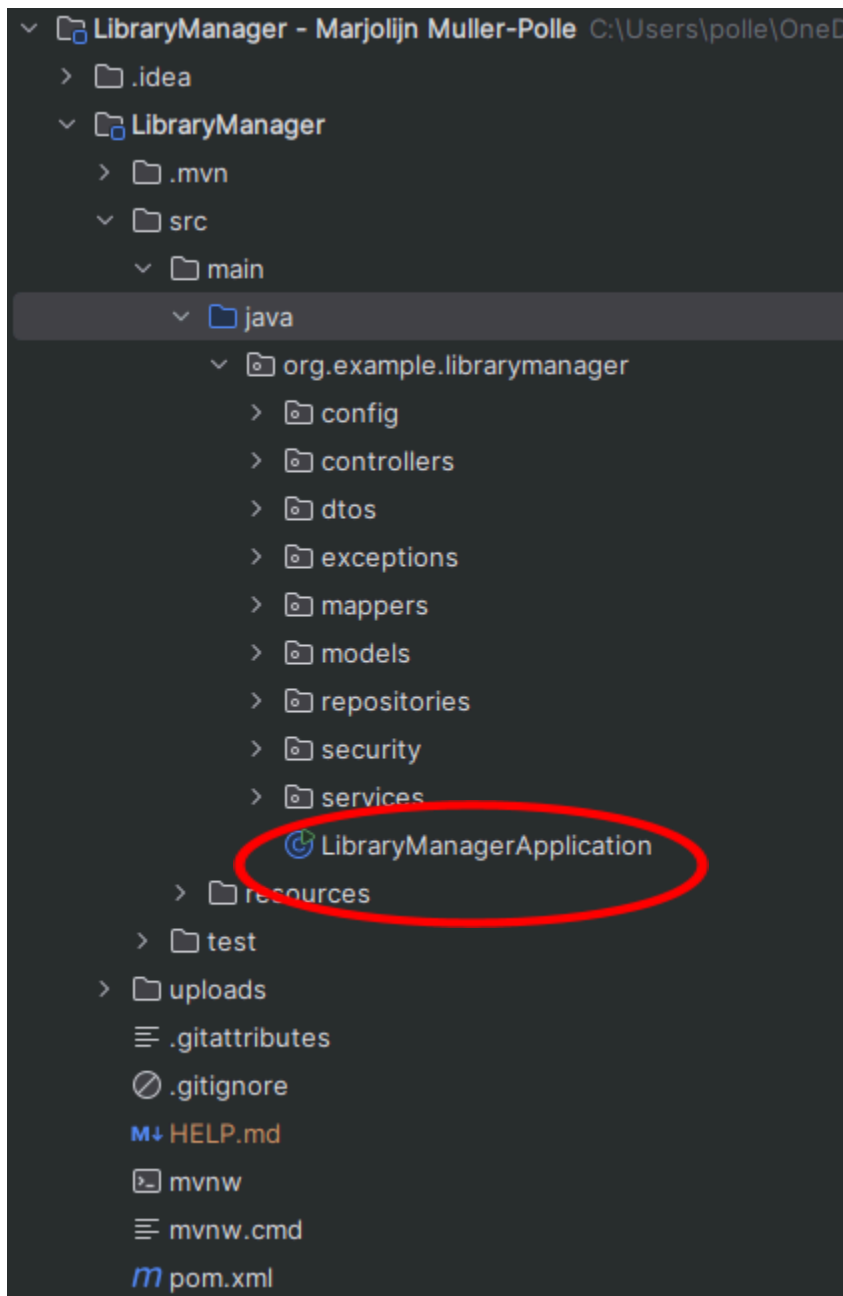


- **Configureer je databaseverbinding:** Stel je gebruikersnaam en wachtwoord in en voeg deze toe aan de omgevingsvariabelen **POSTGRESQL\_USERNAME** en **POSTGRESQL\_PASSWORD** of pas het bestand **application.properties** aan.

### 3. Start de applicatie

Zoek de hoofdklasse van je Spring Boot-applicatie ([LibraryManagerApplication.java](#)) en voer deze uit. De applicatie zal op poort 8080 starten.

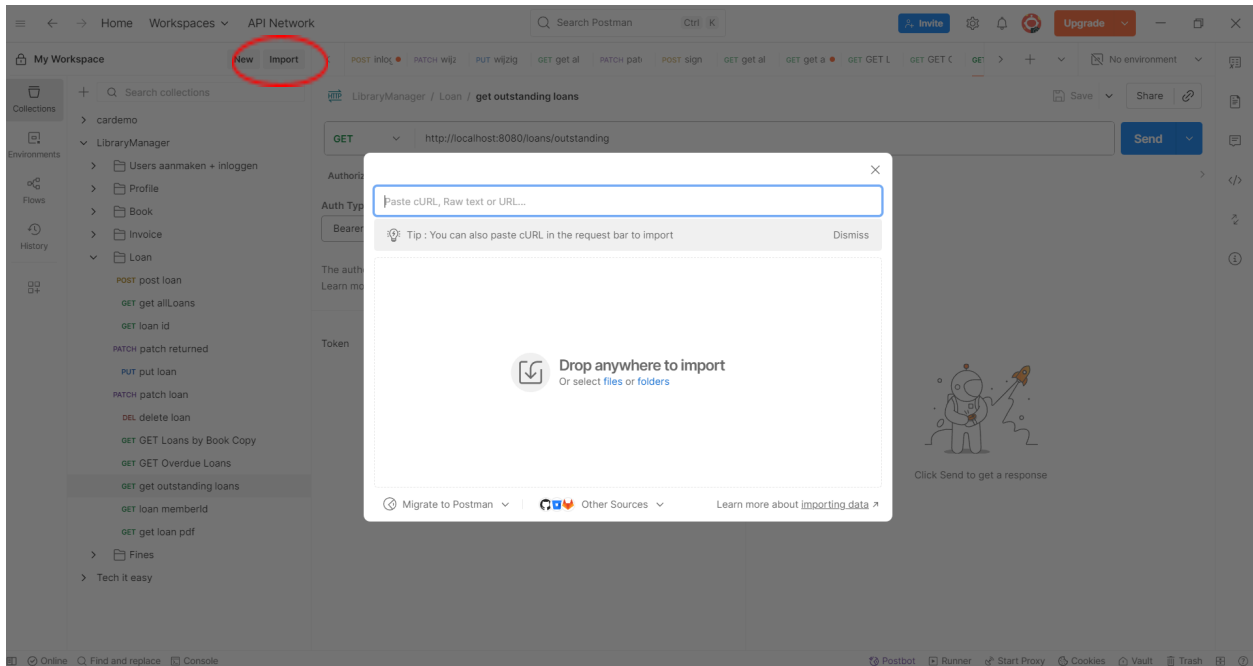
`src → main → java → org.example.librarymanager → LibraryManagerApplication.java`



#### 4. Importeer Postman-collectie

Open Postman en importeer de meegeleverde

[LibraryManager.postman\\_collection.json](#). Deze bevat alle endpoints en voorbeeldrequests.







## Testgebruikers

### Admin

<b>Gebruikersnaam</b>	<b>Wachtwoord</b>
admin	passwordAdmin

### Librarian

<b>Gebruikersnaam</b>	<b>Wachtwoord</b>
emilylib	passwordEmily
davidlib	passwordDavid
sarahlib	passwordSarah
chrislib	passwordChris
lisalib	passwordLisa

### Member

<b>Gebruikersnaam</b>	<b>Wachtwoord</b>
adamember	passwordAdam
ellenmember	passwordEllen
michaelmember	passwordMichael
oliviamember	passwordOlivia
petermember	passwordPeter

## Rest-endpoints

### Authenticatie (inloggen)

Toegangsrechten	Request	Endpoint	Data	Toelichting	Extra informatie
Iedereen	POST	/auth	In body: <ul style="list-style-type: none"><li>• username</li><li>• password</li></ul>	Inloggen van de gebruiker. Jwt token wordt in de header teruggegeven.	

## Profile/User

Toegangsrechten	Request	Endpoint	Data	Toelichting	Extra informatie
ingelogd, ADMIN, LIBRARIAN	GET	/profiles/{userId}	In header: • JWT-token In endpoint: • userId	Het zien van een bepaald profiel. De ingelogde member ziet alleen zijn eigen profiel. Admin en librarian kunnen profielen zien.	
		/profiles/username/{username}	In header: • JWT-token In endpoint: • username	Het zien van een bepaald profiel. De ingelogde member ziet alleen zijn eigen profiel. Admin en librarian kunnen profielen zien.	

ADMIN, LIBRARIAN	POST	/profiles	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In body: <ul style="list-style-type: none"> <li>• username</li> <li>• password</li> <li>• firstName</li> <li>• lastName</li> <li>• street</li> <li>• houseNumber</li> <li>• postalCode</li> <li>• city</li> </ul>		Het aanmaken van een profiel. Om dit goed te laten verlopen, kan niet iedereen dit zomaar doen. Als er geen role wordt ingevuld, dan wordt deze automatisch op member gezet.	
			Optioneel: <ul style="list-style-type: none"> <li>• roles</li> <li>• email</li> <li>• phone</li> </ul>			
	GET	/profiles	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul>		Een lijst van alle profielen	
		/profiles/members	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul>		Een lijst van alle member-profielen.	
		/profiles/librarians	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul>		Een lijst van alle librarian-profielen.	
		/profiles/members/{userId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• userId</li> </ul>		Een profiel van een member met het juiste userId.	

		/profiles/librarians/{userId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• userId</li> </ul>	Een profiel van een bibliothecaris met het juiste userId.	
	PUT	/profiles/{userId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• userId</li> </ul> In body: <ul style="list-style-type: none"> <li>• username</li> <li>• password</li> <li>• firstName</li> <li>• lastName</li> <li>• street</li> <li>• houseNumber</li> <li>• postalCode</li> <li>• city</li> </ul>	Het volledig aanpassen van een profiel.	
			Optioneel: <ul style="list-style-type: none"> <li>• roles</li> <li>• email</li> <li>• phone</li> </ul>		

	PATCH	/profiles/{userId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• userId</li> </ul>	Het deels aanpassen van profiel.	
			Optioneel: <ul style="list-style-type: none"> <li>• username</li> <li>• password</li> <li>• firstName</li> <li>• lastName</li> <li>• street</li> <li>• houseNumber</li> <li>• postalCode</li> <li>• city</li> <li>• roles</li> <li>• email</li> <li>• phone</li> </ul>		
	PATCH	/profiles/{userId}/photo	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• userId</li> </ul> In body: <ul style="list-style-type: none"> <li>• profilePictureFile</li> </ul>	Het toevoegen of wijzigen van een profielfoto.	Het profilePictureFile moet een MultipartFile zijn.
ADMIN	DELETE	/profiles/{userId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• userId in het pad</li> </ul>	Het verwijderen van een profiel.	

## Boeken

Toegangsrechten	Request	Endpoint	Data	Toelichting	Extra informatie
Iedereen	GET	/books		Je krijgt een lijst met alle boeken uit de database.	
		/books/{bookId}	In endpoint: • bookId	Je krijgt een lijst met de boeken met een specifieke bookId	
		/books/{bookId}/copies	In endpoint: • bookId	Je krijgt een lijst van alle exemplaren van een specifiek boek.	
		/books/{bookId}/copies/{followNumber}	In endpoint: • bookId • followNumber	Je krijgt het exemplaar van een specifiek boek.	
		/books/title/{title}	In endpoint: • title	Je krijgt het boek met een specifieke titel.	
		/books/author/{authorLastName}	In endpoint: • authorLastName	Je krijgt een lijst van boeken van een specifieke auteur.	

		/books/isbn/{isbn}	In endpoint: <ul style="list-style-type: none"><li>isbn</li></ul>	Je krijgt het boek met de ISBN.	
		/books/category/{category}	In endpoint: <ul style="list-style-type: none"><li>category</li></ul>	Je krijgt een lijst met boe van die categorie.	Boekcategorieën: <ul style="list-style-type: none"><li>FICTION</li><li>NON_FICTION</li><li>CHILDRENS_BO</li><li>ADULTS_BOOK</li><li>SCIENCE</li><li>HISTORY</li></ul>
		/books/status/{status}	In endpoint: <ul style="list-style-type: none"><li>status</li></ul>	Je krijgt een lijst met exemplaren met die statu	Boekstatussen: <ul style="list-style-type: none"><li>AVAILABLE</li><li>ON_LOAN</li><li>DAMAGED</li><li>MISSING</li><li>IN_REPAIR</li></ul>
ADMIN, LIBRARIAN	POST	/books	In header: <ul style="list-style-type: none"><li>JWT-token</li></ul> In body: <ul style="list-style-type: none"><li>title</li><li>authorFirstName</li><li>authorLastName</li><li>ISBN</li><li>category</li></ul>	Het toevoegen van een boek.	
			Optioneel in body: <ul style="list-style-type: none"><li>publisher</li></ul>		
	PUT	/books/{bookId}	In header: <ul style="list-style-type: none"><li>JWT-token</li></ul> In endpoint: <ul style="list-style-type: none"><li>bookId</li></ul>	Het volledig aanpassen een boek.	



			In body: <ul style="list-style-type: none"> <li>• title</li> <li>• authorFirstName</li> <li>• authorLastName</li> <li>• ISBN</li> <li>• category</li> </ul>		
			Optioneel in body: <ul style="list-style-type: none"> <li>• publisher</li> </ul>		
		/books/{bookId}/copies/{followNumber}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• bookId</li> <li>• followNumber</li> </ul> In body: <ul style="list-style-type: none"> <li>• status</li> <li>• ISBN</li> <li>• title</li> <li>• authorFirstName</li> <li>• authorLastName</li> <li>• category</li> </ul>	Het volledig aanpassen van een exemplaar.	Boekstatussen: <ul style="list-style-type: none"> <li>• AVAILABLE</li> <li>• ON_LOAN</li> <li>• DAMAGED</li> <li>• MISSING</li> <li>• IN_REPAIR</li> </ul>
	PATCH	/books/{bookId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint:	Het deels aanpassen van een exemplaar.	

			<ul style="list-style-type: none"> <li>bookId</li> </ul>		
			Optioneel in body: <ul style="list-style-type: none"> <li>title</li> <li>authorFirstName</li> <li>authorLastName</li> <li>ISBN</li> <li>category</li> <li>publisher</li> </ul>		
	DELETE	/books/{bookId}	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>bookId</li> </ul>	Het volledig verwijderen van een boek.	
		/books/{bookId}/copies/{followNumber}	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>bookId</li> <li>followNumber</li> </ul>		

## Loans

Toegangsrechten	Request	Endpoint	Data	Toelichting	Extra informatie
Iedereen die is ingelogd	POST	/loans	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In body: <ul style="list-style-type: none"> <li>• bookCopyId</li> <li>• userId</li> </ul>	Het aanmaken van een lening met de geleende boeken	
			Optioneel: <ul style="list-style-type: none"> <li>• loanDate</li> <li>• returnDate</li> <li>• isReturned</li> <li>• actualReturnDate</li> <li>• finelId</li> </ul>		
	GET	/loans/user		Het bekijken van alle leningen van de gebruiker.	
ingelogd, ADMIN, LIBRARIAN	GET	/loans/user/{userId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In de body: <ul style="list-style-type: none"> <li>• </li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• </li> </ul>	Een lijst met alle leningen van een bepaalde gebruiker. De ingelogde member kan alleen zijn eigen leningen zien.	
		/loans/{loanId}/pdf	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• loanId</li> </ul>	Het genereren van een pdf met de leningen die uitstaan	

	PATCH	/loans/{loanId}/return	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• loanId</li> </ul>	Het terugbrengen van een boek.	
ADMIN, LIBRARIAN	GET	/loans		Een lijst met alle leningen	
		/loans/{loanId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• loanId</li> </ul>	Geeft een bepaalde lening	
		/loans/bookcopy/{bookCopyId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• bookCopyId</li> </ul>	Geeft de lening waar een bepaald boek is geleend.	
		/loans/overdue	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul>	Geeft een lijst met alle boeken die te laat zijn met inleveren.	
		/loans/outstanding	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul>	Geeft een lijst met alle leningen die uitstaan.	

	PUT	/loans/{loanId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In de body: <ul style="list-style-type: none"> <li>• bookCopyId</li> <li>• userId</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• loanId</li> </ul>	Het volledig aanpassen van een lening.	
			Optioneel: <ul style="list-style-type: none"> <li>• loanDate</li> <li>• returnDate</li> <li>• isReturned</li> <li>• actualReturnDate</li> <li>• fineld</li> </ul>		
	PATCH	/loans/{loanId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• loanId</li> </ul>	Het deels aanpassen van een lening.	
			Optioneel: <ul style="list-style-type: none"> <li>• loanDate</li> <li>• returnDate</li> <li>• isReturned</li> <li>• actualReturnDate</li> <li>• fineld</li> </ul>		
	DELETE	/loans/{loanId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• loanId</li> </ul>	Het verwijderen van een lening.	

## Fine

Toegangsrechten	Request	Endpoint	Data	Toelichting	Extra informatie
ADMIN, LIBRARIAN	GET	/fines	In header: • JWT-token	Geeft een lijst met alle boetes.	
		/fines/{fineId}	In header: • JWT-token In endpoint: • fineId	Geeft een bepaalde boete.	
	PUT	/fines/notPaid	In header: • JWT-token	Geeft een lijst met alle boetes die niet zijn betaald.	
		/fines/{fineId}	In header: • JWT-token In endpoint: • fineId In de body: • fineAmount • fineDate • loanId • invoiceId  Optioneel: • isPaid	Een boete volledig wijzigen.	

	PATCH	/fines/{fineId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• fineId</li> </ul>	Een boete deels wijzigen	
			Optioneel: <ul style="list-style-type: none"> <li>• fineAmount</li> <li>• fineDate</li> <li>• loanId</li> <li>• invoiceld</li> <li>• isPaid</li> </ul>		
		/fines/{fineId}/pay	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• fineId</li> </ul>	Een boete op betaald zetten.	
DELETE		/fines/{fineId}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• fineId</li> </ul>	Een boete verwijderen.	

## Invoices

Toegangsrechten	Request	Endpoint	Data	Toelichting	Extra informatie
Iedereen die is ingelogd	GET	/invoices/user	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul>	Het bekijken van alle facturen. De ingelogde persoon ziet alleen zijn eigen facturen.	
ingelogd, ADMIN, LIBRARIAN	GET	/invoices/{id}	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>invoiceld</li> </ul>	Het bekijken van een specifieke factuur. De ingelogde persoon kan alleen zijn eigen factuur opvragen. Admin en librarian kunnen alle facturen opvragen.	
		/invoices/{id}/pdf	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>invoiceld</li> </ul>	Het verkrijgen van een pdf van de factuur. De ingelogde persoon kan alleen zijn eigen factuur opvragen. Admin en librarian kunnen alle facturen opvragen.	
ADMIN, LIBRARIAN	GET	/invoices/generate	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul>	De facturen worden gegenereerd a.d.h.v. alle boetes die er zijn.	
		/invoices	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul>	Geeft een lijst met alle facturen die er zijn.	
		/invoices/date/{date}	In header: <ul style="list-style-type: none"> <li>JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>date</li> </ul>	Geeft een lijst van alle facturen van een bepaald datum.	



		/invoices/date/before/{date}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• date</li> </ul>	Geeft een lijst van alle facturen van voor een bepaalde datum.	
		/invoices/date/after/{date}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• date</li> </ul>	Geeft een lijst van alle facturen van na een bepaald datum.	
		/invoices/period/{invoicePeriod}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• invoicePeriod</li> </ul>	Geeft een lijst van alle facturen van een bepaald periode die op de factuur staan (bijvoorbeeld augustus 2025)	
	PUT	/invoices/{id}	In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• status</li> </ul>	Geeft een lijst van alle facturen met een bepaald betaalstatus.	Status van betaling: <ul style="list-style-type: none"> <li>• OPEN</li> <li>• PENDING</li> <li>• PAID</li> <li>• OVERDUE</li> <li>• CANCELLED</li> </ul>
			In header: <ul style="list-style-type: none"> <li>• JWT-token</li> </ul> In endpoint: <ul style="list-style-type: none"> <li>• invoiceld</li> </ul>		

			In body: <ul style="list-style-type: none"><li>• invoiceDate</li><li>• userId</li></ul>		<ul style="list-style-type: none"> <li>• OVERDUE</li> <li>• CANCELLED</li> </ul>
			Optioneel: <ul style="list-style-type: none"><li>• invoicePeriod</li><li>• paymentStatus</li></ul>		
	PATCH	/invoices/{id}	In header: <ul style="list-style-type: none"><li>• JWT-token</li></ul> In endpoint: <ul style="list-style-type: none"><li>• invoiceId</li></ul>	Het gedeeltelijk aanpassen van een factuur.	Status van betaling: <ul style="list-style-type: none"> <li>• OPEN</li> <li>• PENDING</li> <li>• PAID</li> <li>• OVERDUE</li> <li>• CANCELLED</li> </ul>
			Optioneel: <ul style="list-style-type: none"><li>• invoicePeriod</li><li>• paymentStatus</li><li>• invoiceDate</li><li>• userId</li></ul>		
		/invoices/{id}/pay	In header: <ul style="list-style-type: none"><li>• JWT-token</li></ul> In endpoint: <ul style="list-style-type: none"><li>• invoiceId</li></ul>	Het automatisch aanpassen dat de factuur betaald is.	
	DELETE	/invoices/{id}	In header: <ul style="list-style-type: none"><li>• JWT-token</li></ul> In endpoint: <ul style="list-style-type: none"><li>• invoiceId</li></ul>	Het verwijderen van een factuur.	



## Postman collecties

De postman collectie heb ik als een aparte file toegevoegd aan het zip-bestand, vanwege de vele data die erin staat.

