

# API\_doc for LibraryWebService

Author: Tommy Steger, Anna Thompson, Olof Wahlund Level: Intermediate. Technologies: JPA, EJB, JPA and JAX-RS. Database: MySQL Summary: A practical REST service example of library solution. Target Project: WildFly.  
Source: <https://github.com/annaThompsonLex/LibraryWebService>.

## SECURITYINFORMATION

**Add Username & password before all request:**

[Username]:[password]@localhost:8080/LibraryWebService/rest/loan/takeloan/{[bookId]}/{[userId]}

## LOAN

GET	Make a new Loan
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/loan/takeloan/{[bookId]}/{[userId]}	

GET	Return a new Loan
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/loan/returnloan/{[bookId]}/{[userId]}	

GET	Serch for a loan			
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/loan?parameter=[yourkey]				
<b>Poster API requests:</b> localhost:8080/LibraryWebService/rest/loan				
Parameters				
Serch by parram				
Parameter	Required	Valid Options	Default value	Description
bookid	No		<empty>	Book id of loan to search for.
userid	No		<empty>	User id of loan to search for.

GET	Get all expired loans
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/loan?expired	

## BOOK

GET

Serch for book

Send all data requests to:

localhost:8080/LibraryWebService/rest/book?parameter=[yourkey]

Poster API requests:

localhost:8080/LibraryWebService/rest/book

Parameters

Serch parram

Parameter	Required	Valid Options	Default value	Description
title	No		<empty>	Book title to search for.
author	No		<empty>	Author to search for.
genre	No		<empty>	Genre to serch for.
isbn	No		<empty>	A valid ISBN ID
<empty>	No		<empty>	List all Books.

GET	Serch for book by id		
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/book/[id]			

POST	Cerate book
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/book	
<b>Accepts JSON and XML</b>	<b>Example:</b> { "ISBN": [isbn], "title": [title], "author": [author], "genre": [genre], "shelf": [shelf], "copies": [copies], "picUrl": [picUrl]}

PUT					Update book				
<b>Send all data requests to:</b>									
localhost:8080/LibraryWebService/rest/book/[id]?parameter=[yourkey]									
<b>Accepts JSON and XML</b>					<b>Example:</b>				
{ "shelf": [shelf]}									
Parameters									
Update parram									
Parameter	Required	Valid Options			Default value	Description			
author	No				<empty>	New author.			
copies	No				<empty>	New number of copies.			
shelf	No				<empty>	New shelf.			

DELETE	Delete book
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/book/[id]	

## USER

GET

Serch for user

Send all data requests to:

localhost:8080/LibraryWebService/rest/user?parameter=[yourkey]

Poster API requests:

localhost:8080/LibraryWebService/rest/user

Parameters

Serch by ID or parram

Parameter	Required	Valid Options	Default value	Description
firstName	No		<empty>	Users first name
lastName	No		<empty>	Users last name
email	No		<empty>	Users email
<empty>	No		<empty>	List all Users

GET	Serch for user by id
<b>Send all data requests to:</b>	
localhost:8080/LibraryWebService/rest/user/[id]	

POST	Cerate user
<b>Send all data requests to:</b>	
localhost:8080/LibraryWebService/rest/user	
<b>Accepts JSON and XML</b>	<b>Example:</b>
{ "firstName": [first name], "lastName": [lastName], "email": [email], "password": [password]}	

PUT		Update user		
Send all data requests to:				
localhost:8080/LibraryWebService/rest/user/[id]?parameter=[yourkey]				
Accepts JSON and XML		Example:		
{ "email": [email]}				
Parameters				
Update parram				
Parameter	Required	Valid Options	Default value	Description
lastName	No		<empty>	New last name.
email	No		<empty>	New email.

DELETE	Delete user
<b>Send all data requests to:</b> localhost:8080/LibraryWebService/rest/user/[id]	