

Restful APIs: By Alex Nguyen, Michelle Lu, Michael Su, Adrian Groch

Url/path for every action

Method

Field names/values

As an authorised user, I want to log in/logout of my account

Path: /api/user/login

Method: GET, POST

Fields: Email, Password

Serve response codes in HTTP header:

200 - successful

401 - no match, unable to authenticate

Logout handled on frontend

As an unauthorised user, I want to register an account

Path: /api/user/register

Method: POST

Fields: email/password/university/user location

Server response:

201 - successful

400 - error

Other User API's:

Get user details

Path: /api/user/<userid>

Method: POST

Fields: user_id (returns json with user details)

Server response:

200 - successful

404 - error

Update user details

Path: /api/user/update/<userid>

Method: POST

Fields: (see DB planning, User table)

Server response:

200 - successful

404 - error

401 - not logged in

Delete user details

Path: /api/user/delete/<userid>

Method: POST

Fields: (see DB planning, User table)

Server response:

200 - successful

404 - error

401 - not logged in

List users (debug)

Path: /api/user/list

Method: POST

Fields: (see DB planning, User table)

Server response:

200 - successful

404 - error

Being able to add books

Path: /api/books/create

Method: POST

Fields: (see DB planning, book table)

Server response:

200 - success

2011 - field is missing

401 - not logged in

Add book to wishlist

Path: /api/user/wishlist/create

Method: POST

Fields: (see DB planning, Book table)

Server response:

201 - successful

404 - error

401 - not logged in

Book image adding

Path: /api/books/image/<bookid>

Method: POST

Fields: (see DB planning, Book_Image table)

Server response:

200 - successful

404 - error

Update book details

Path: /api/books/update/<bookid>

Method: POST

Fields: (see DB planning, Book table)

Server response:

201 - successful

404 - error

401 - not logged in

Delete book details

Path: /api/books/delete/<bookid>

Method: POST

Fields: (see DB planning, Book table)

Server response:

200 - successful

404 - error

401 - not logged in

Get book details

Path: /api/books/<bookid>

Method: POST

Fields: (see DB planning, Book table)

Server response:

200 - successful

404 - error

Search for book using query

Path: /api/books/search/<query>

Method: POST

Fields: (see DB planning, Book table)

Server response:

200 - successful

204 - no content found

List all book inventory based on transaction type

Path: /api/books/list/<transaction_type>

Method: POST

Fields: (see DB planning, Book_List table)

Server response:

404 - error

List all book inventory

Path: /api/booklistings

Method: POST

Fields: (see DB planning, Book_List table)

Server response:

404 - error

Get book details based on user details and joins

Path: /api/booklistings/<bookid>

Method: POST

Fields: (see DB planning, Book_List table and User table)

Server response:

404 - error

User Action Stuff

Request Book

Path: /api/request/<book_id>

Method: POST

Fields: (see DB planning, Book_List table, Notification, Transaction)

Server response:

201 - successful

404 - error

401 - not logged in

Respond to Book Request

Path: /api/response/<notification_id>

Method: POST

Fields: (see DB planning, User table, Notification, Transaction)

Server response:

201 - successful

404 - error

401 - not logged in

Get details of notification

Path: /api/request/notifications/<user_id>

Method: POST

Fields: (see DB planning, Notification, Transaction)

Server response:

404 - error

Get details of wishlist for given user

Path: /api/user/wishlist/<user_id>

Method: POST

Fields: (see DB planning, Book_List)

Server response:

404 - error

Get transactions for a particular user

Path: /api/user/transactions/<user_id>

Method: POST

Fields: (see DB planning, Book, Transaction tables, User)

Server response:

404 - error

401 - not logged in