Test Report of My Favorite Movies App

COMP. 6215 Assignment-3: Deliverable (3)

Edited by @ Emerald Chen

21 / 06 / 2019

Table of Contents

1 Testing the API 3

1.1 Get-Routes 3

1.2 Post-Routes 12

1.2.1 Post-Routes (updated) 12-13

1.3 Put-Routes 14

1.4 Delete-Routes 14

2 Testing the application 16

2.1 Testing the application (updated) 17

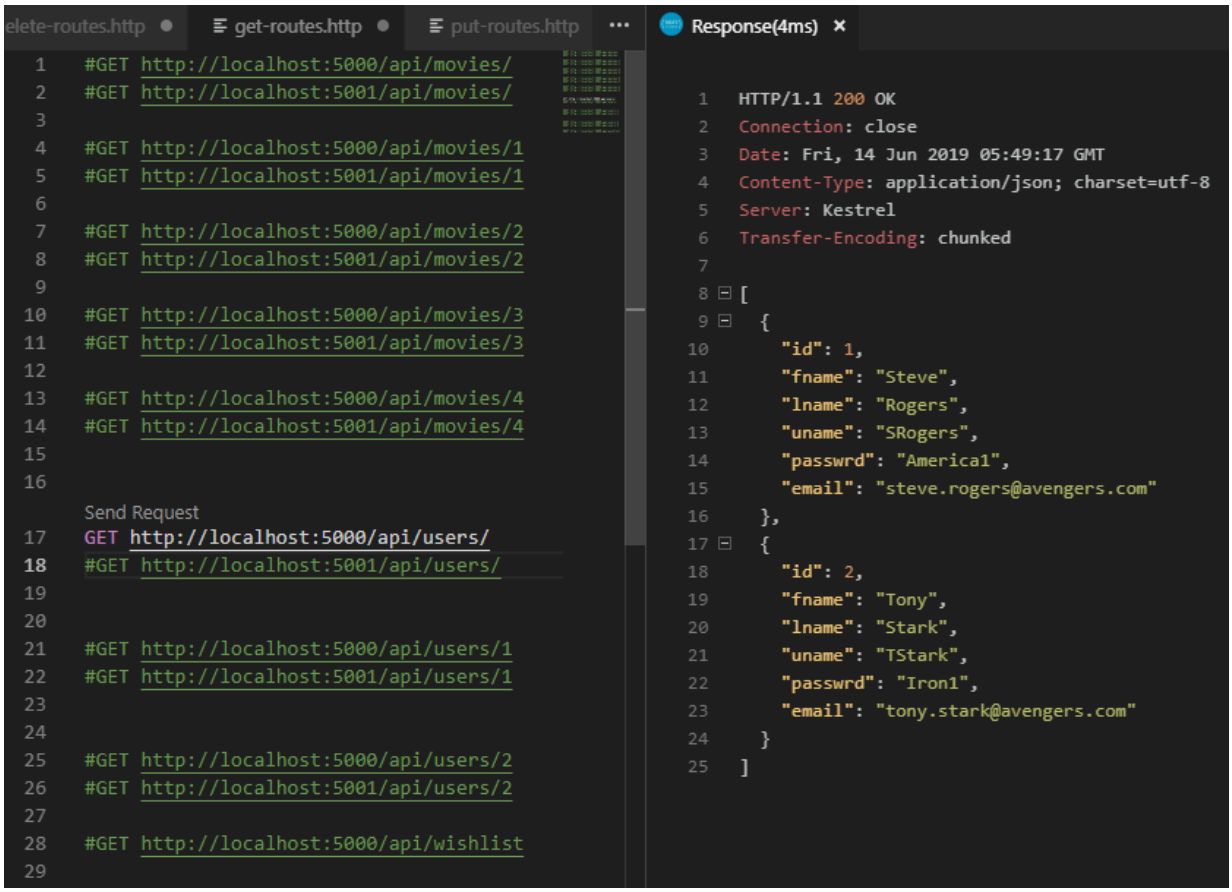
3 The reasons for choosing WPF for GUI 18

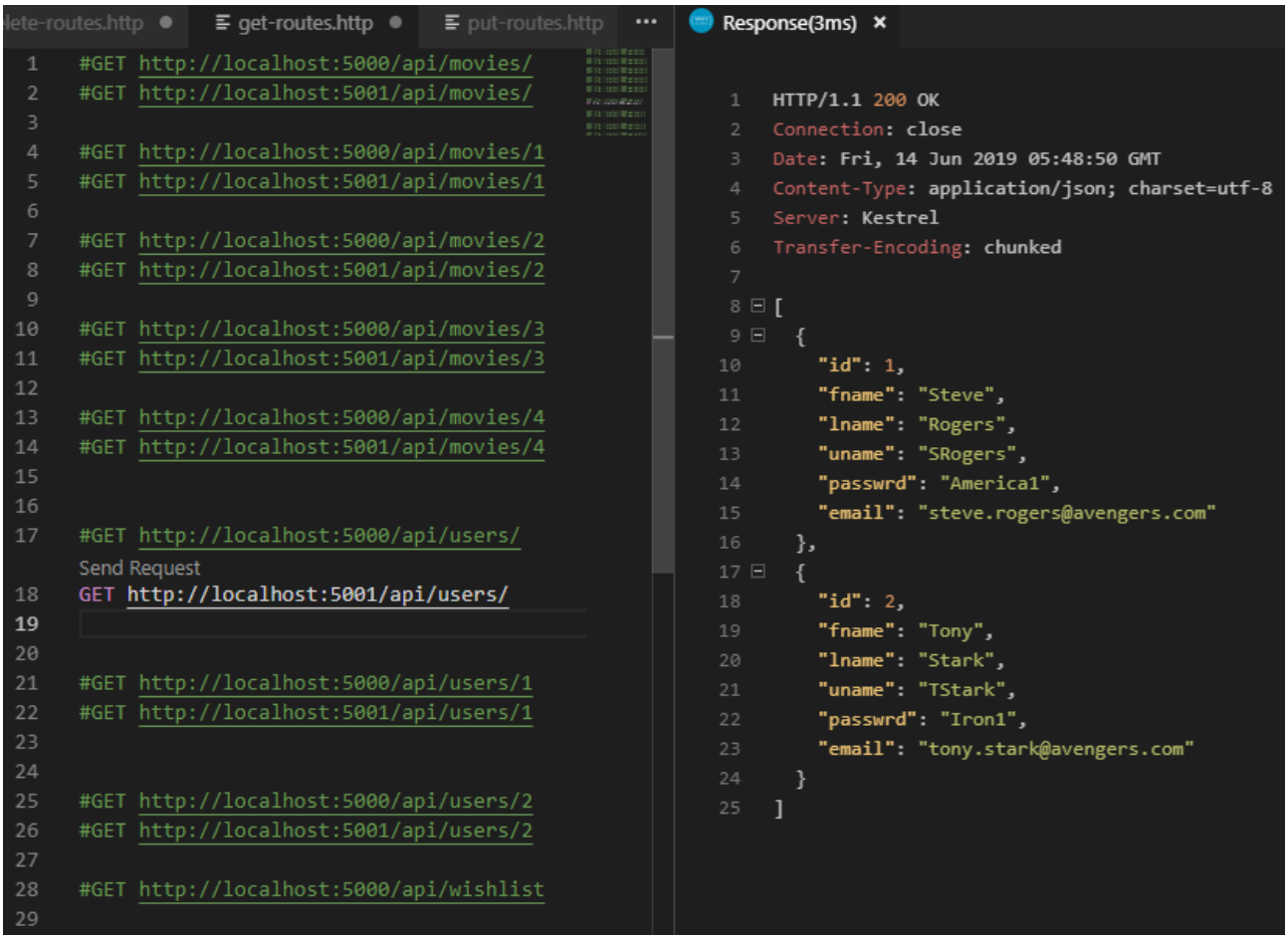
1. Testing the API

Below are a series of GET / POST / PUT and DELETE scripts that will test the API.

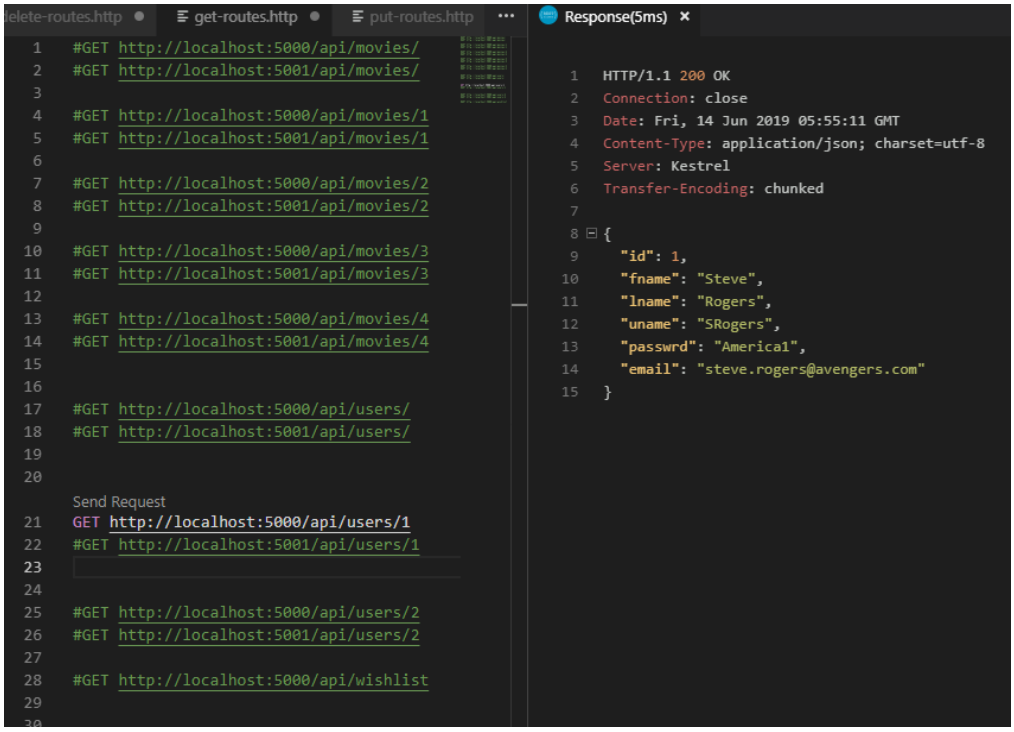
1.1 Get-Routes

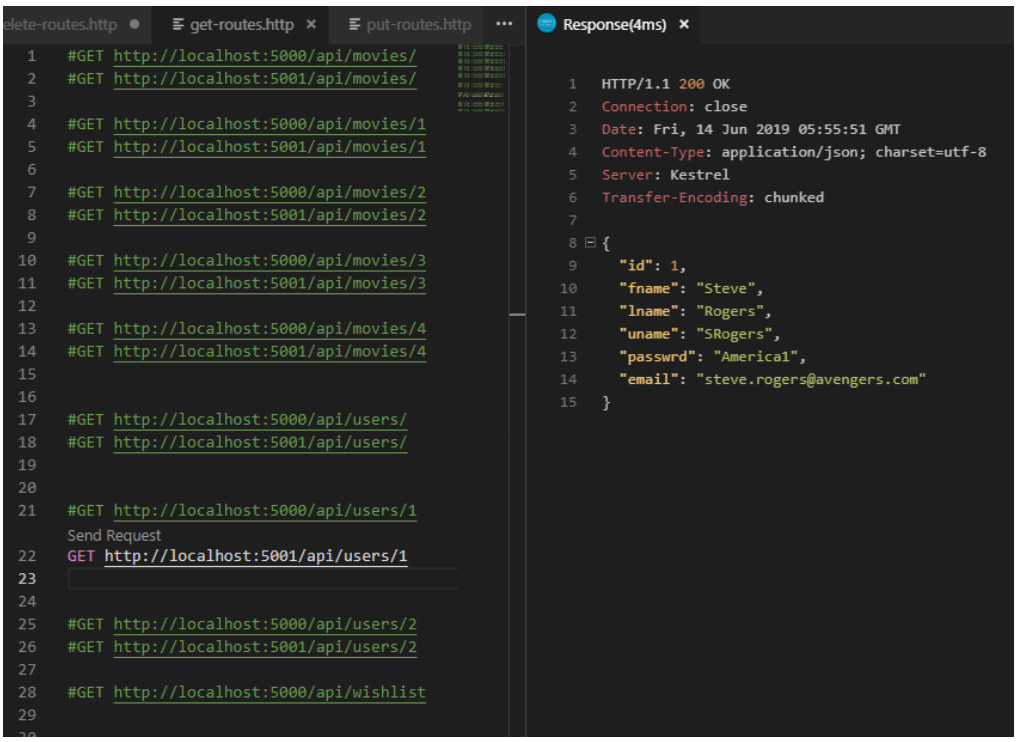
1) GET http://localhost:5000/api/users and http://localhost:5001/api/users



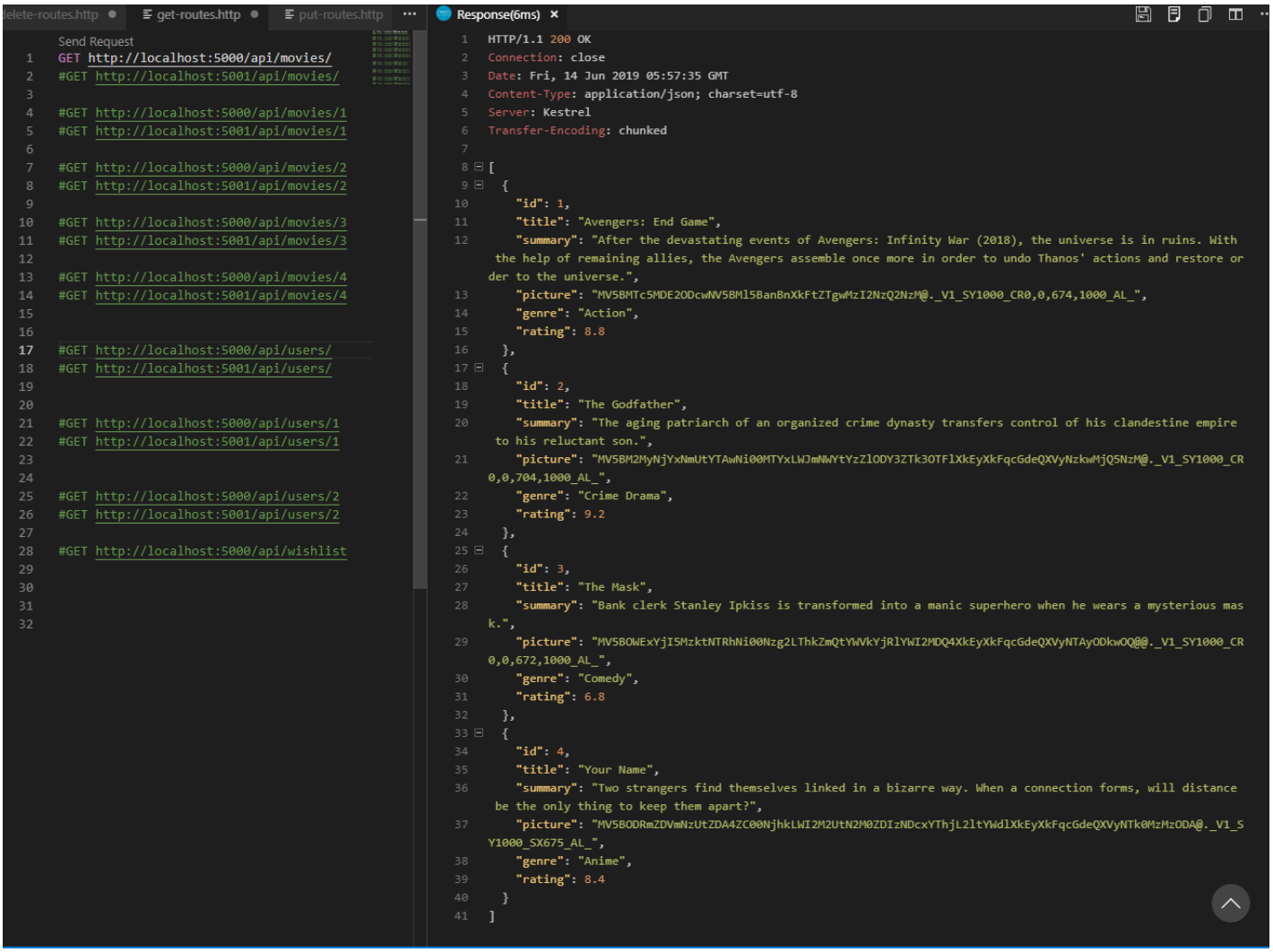

lete-routes.http
get-routes.http
put-routes.http
#GET http://localhost:5000/api/movies/
#GET http://localhost:5001/api/movies/
#GET http://localhost:5000/api/movies/1
#GET http://localhost:5001/api/movies/1
#GET http://localhost:5000/api/movies/2
#GET http://localhost:5001/api/movies/2
#GET http://localhost:5000/api/movies/3
#GET http://localhost:5001/api/movies/3
#GET http://localhost:5000/api/movies/4
#GET http://localhost:5001/api/movies/4
#GET http://localhost:5000/api/users/
Send Request
GET http://localhost:5001/api/users/
#GET http://localhost:5000/api/users/1
#GET http://localhost:5001/api/users/1
#GET http://localhost:5000/api/users/2
#GET http://localhost:5001/api/users/2
#GET http://localhost:5000/api/wishlist
Response(3ms)
HTTP/1.1 200 OK
Connection: close
Date: Fri, 14 Jun 2019 05:48:50 GMT
Content-Type: application/json; charset=utf-8
Server: Kestrel
Transfer-Encoding: chunked
[
{
"id": 1,
"fname": "Steve",
"lname": "Rogers",
"uname": "SRogers",
"passwrd": "America1",
"email": "steve.rogers@avengers.com"
},
{
"id": 2,
"fname": "Tony",
"lname": "Stark",
"uname": "TStark",
"passwrd": "Iron1",
"email": "tony.stark@avengers.com"
}
]

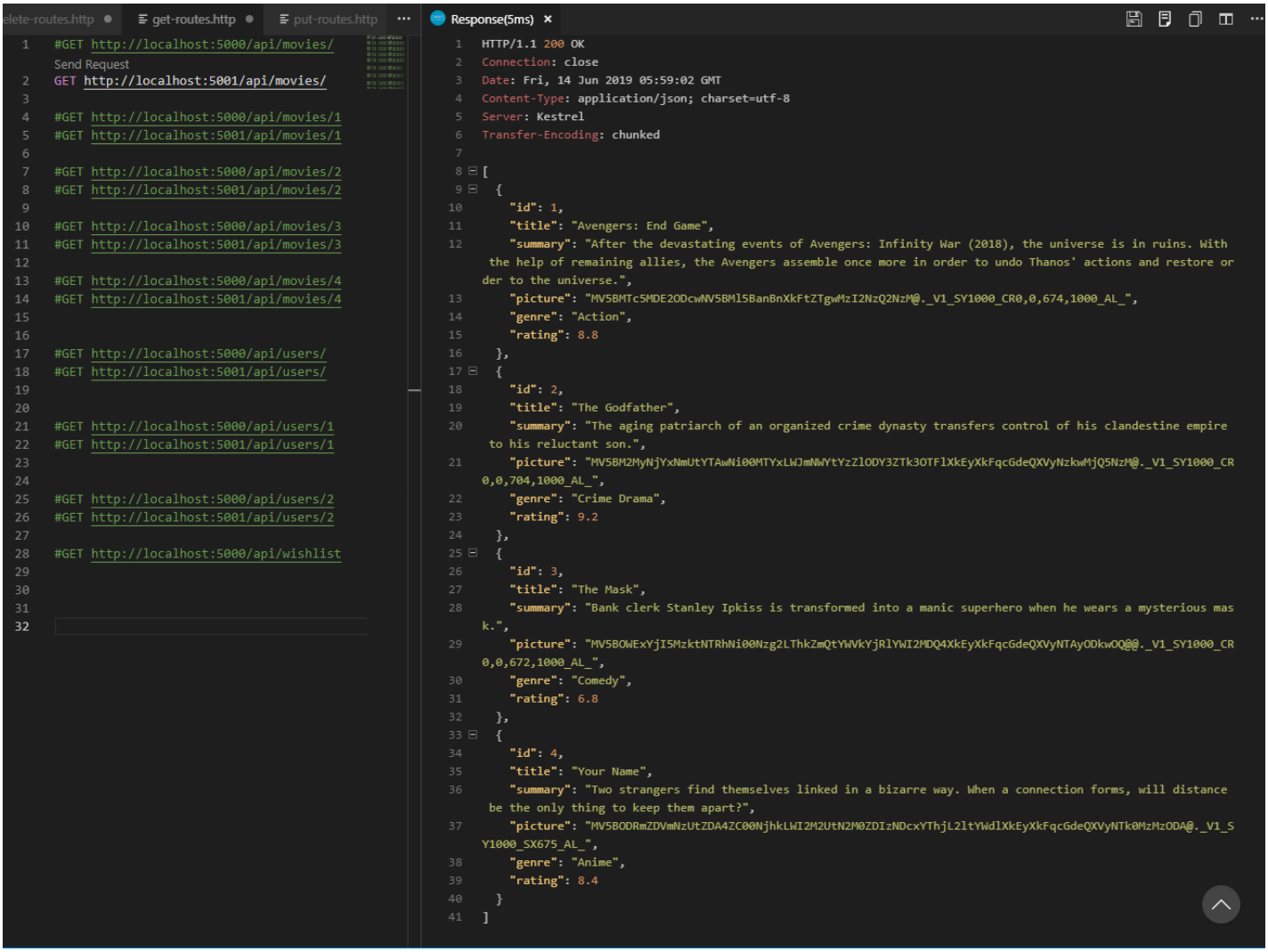
2) GET http://localhost:5000/api/users/1 and
http://localhost:5001/api/users/1



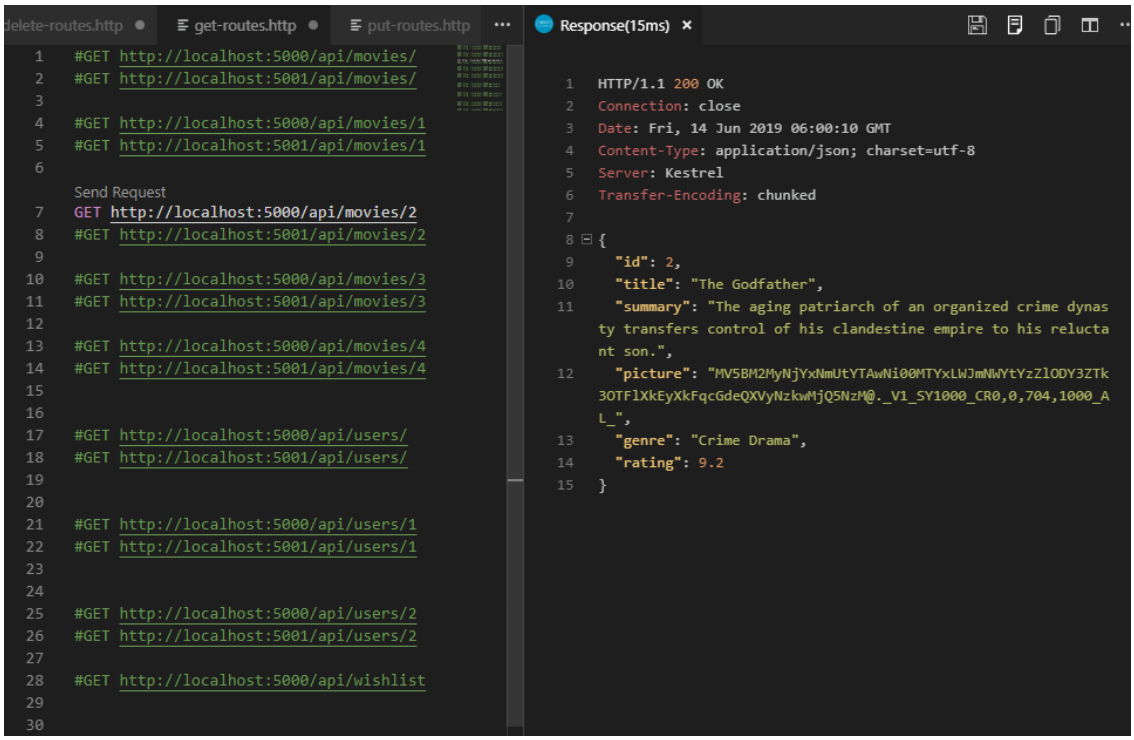


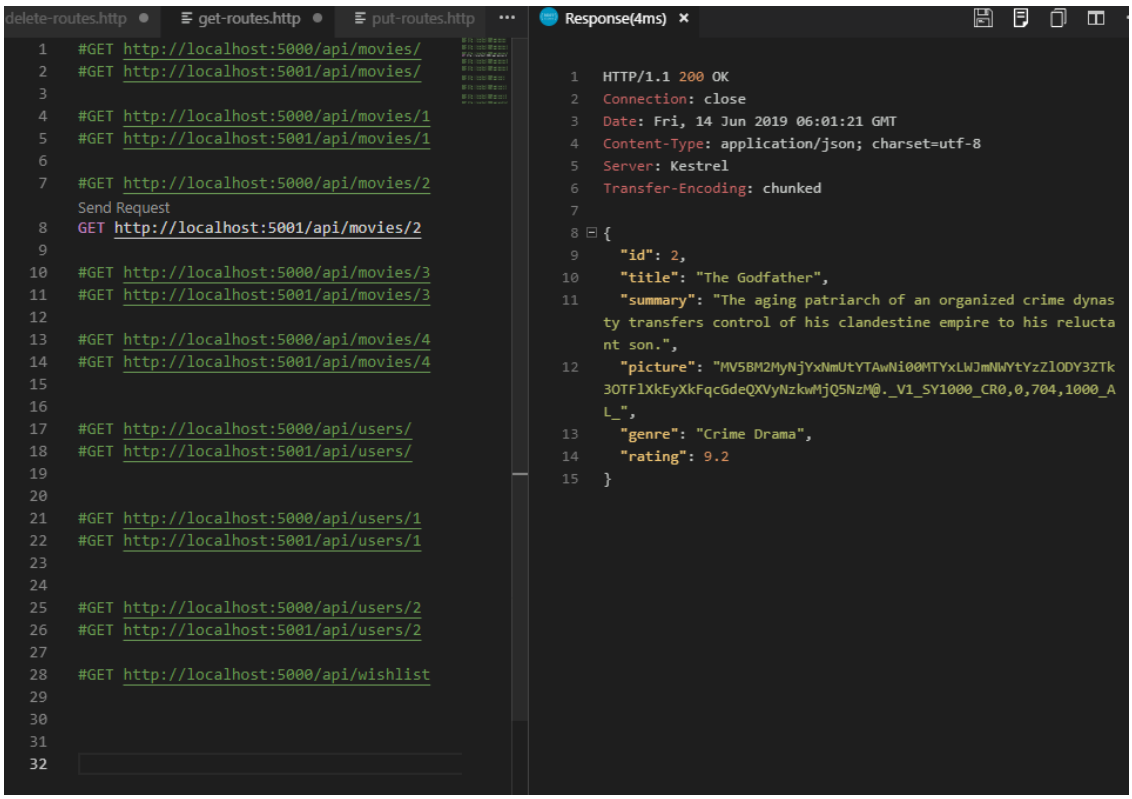
3) GET http://localhost:5000/api/movies and
http://localhost:5001/api/movies



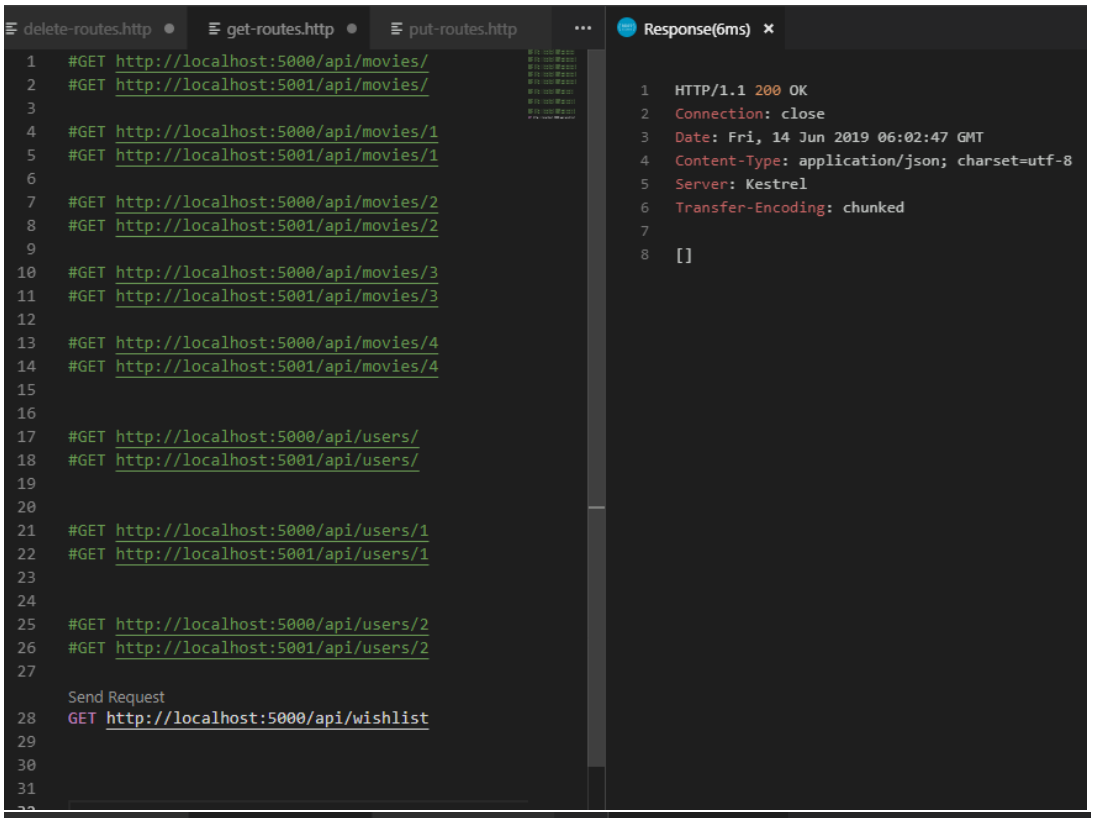

delete-routes.http
get-routes.http
put-routes.http
#GET http://localhost:5000/api/movies/
Send Request
GET http://localhost:5001/api/movies/
#GET http://localhost:5000/api/movies/1
#GET http://localhost:5001/api/movies/1
#GET http://localhost:5000/api/movies/2
#GET http://localhost:5001/api/movies/2
#GET http://localhost:5000/api/movies/3
#GET http://localhost:5001/api/movies/3
#GET http://localhost:5000/api/movies/4
#GET http://localhost:5001/api/movies/4
#GET http://localhost:5000/api/users/
#GET http://localhost:5001/api/users/
#GET http://localhost:5000/api/users/1
#GET http://localhost:5001/api/users/1
#GET http://localhost:5000/api/users/2
#GET http://localhost:5001/api/users/2
#GET http://localhost:5000/api/wishlist
Response(5ms)
HTTP/1.1 200 OK
Connection: close
Date: Fri, 14 Jun 2019 05:59:02 GMT
Content-Type: application/json; charset=utf-8
Server: Kestrel
Transfer-Encoding: chunked
[
{
"id": 1,
"title": "Avengers: End Game",
"summary": "After the devastating events of Avengers: Infinity War (2018), the universe is in ruins. With the help of remaining allies, the Avengers assemble once more in order to undo Thanos' actions and restore order to the universe.",
"picture": "MV5BMTc5MDE2ODcwNV5BMl5BanBnXkFtZTgwMzI2NzQ2NzM@._V1_SY1000_CR0,0,674,1000_AL_",
"genre": "Action",
"rating": 8.8
},
{
"id": 2,
"title": "The Godfather",
"summary": "The aging patriarch of an organized crime dynasty transfers control of his clandestine empire to his reluctant son.",
"picture": "MV5BM2MyNjYxNmUtYTAwNi00MTYxLWJmNWYtYzZlODY3ZTk3OTFlXkEyXkFqcGdeQXVyNzkwMjQ5NzM@._V1_SY1000_CR0,0,704,1000_AL_",
"genre": "Crime Drama",
"rating": 9.2
},
{
"id": 3,
"title": "The Mask",
"summary": "Bank clerk Stanley Ipkiss is transformed into a manic superhero when he wears a mysterious mask.",
"picture": "MV5BOWExYjI5MzktNTRhNi00Nzg2LThkZmQtYWVkYjRlYWI2MDQ4XkEyXkFqcGdeQXVyNTAyODkwOQ@@._V1_SY1000_CR0,0,672,1000_AL_",
"genre": "Comedy",
"rating": 6.8
},
{
"id": 4,
"title": "Your Name",
"summary": "Two strangers find themselves linked in a bizarre way. When a connection forms, will distance be the only thing to keep them apart?",
"picture": "MV5BODRmZDVmNzUtZDA4ZC00NjhkLWI2M2UtN2M0ZDIzNDcxYThjL2ltYWdlXkEyXkFqcGdeQXVyNTk0MzMzODA@._V1_SY1000_SX675_AL_",
"genre": "Anime",
"rating": 8.4
}
]

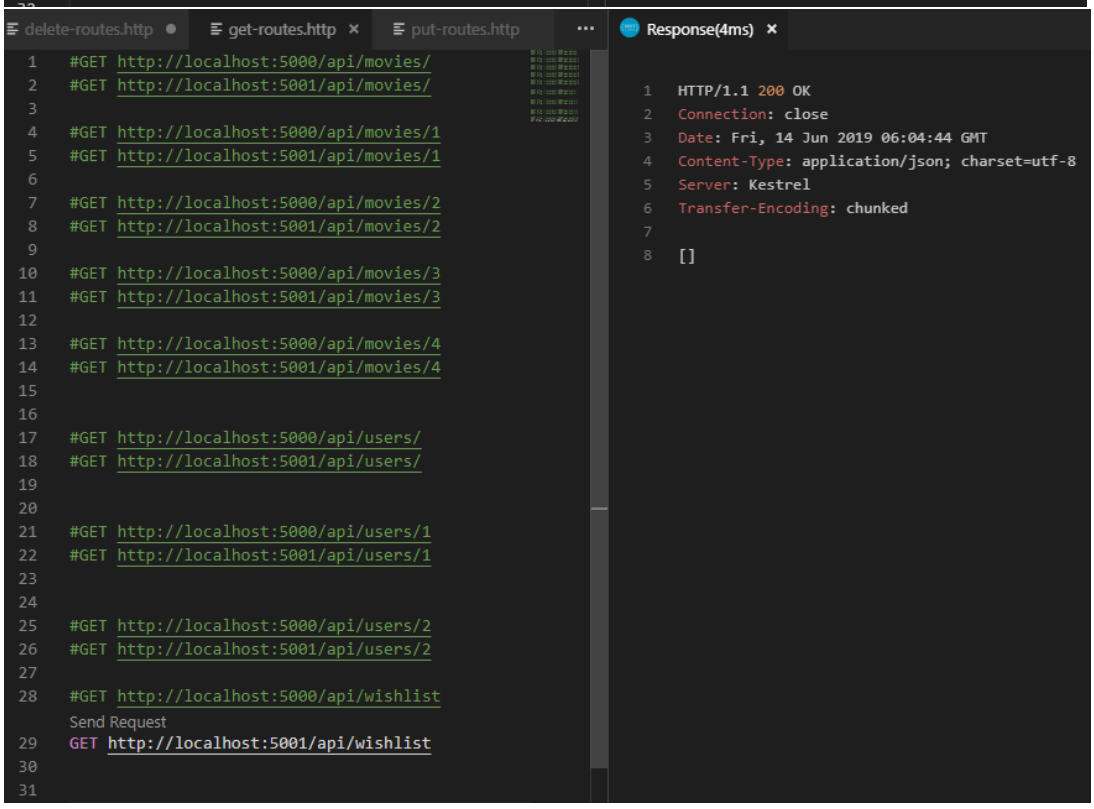
4) GET http://localhost:5000/api/movies/2 and
http://localhost:5001/api/movies/2





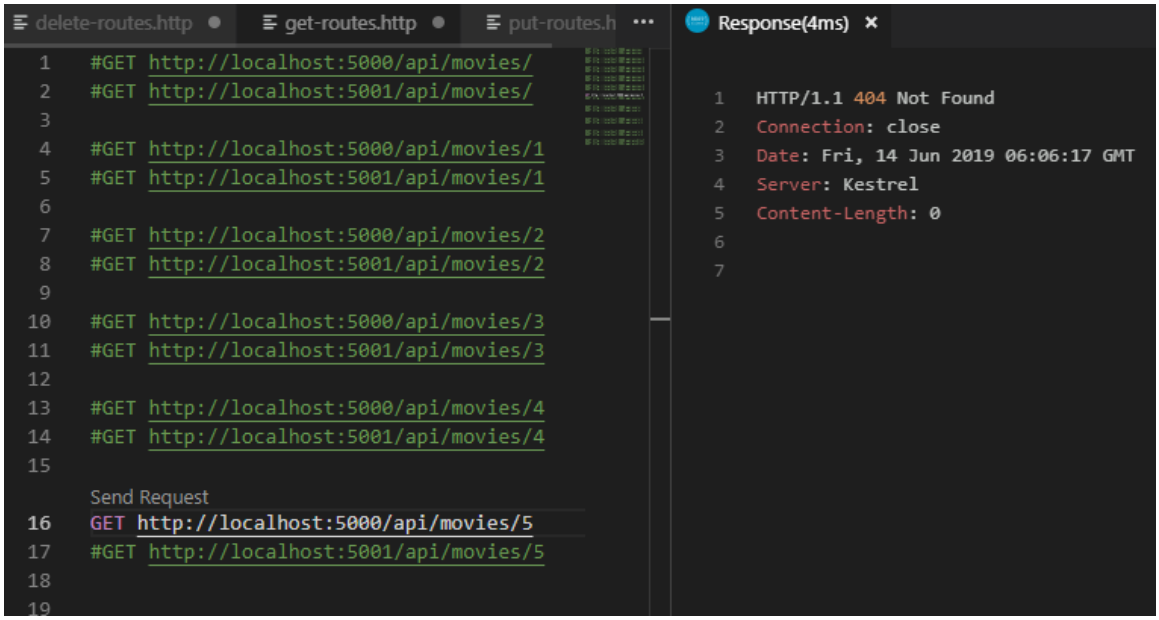
5) GET http://localhost:5000/api/wishlist and
http://localhost:5001/api/wishlist

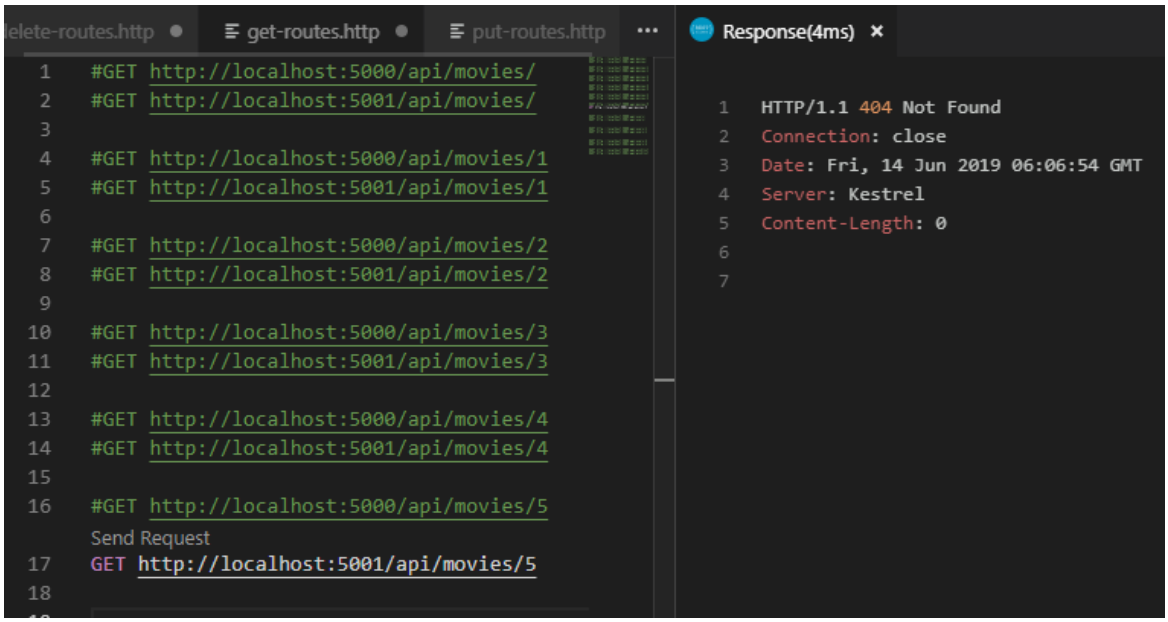




6) GET http://localhost:5000/api/movies/5 and
http://localhost:5001/api/movies/5

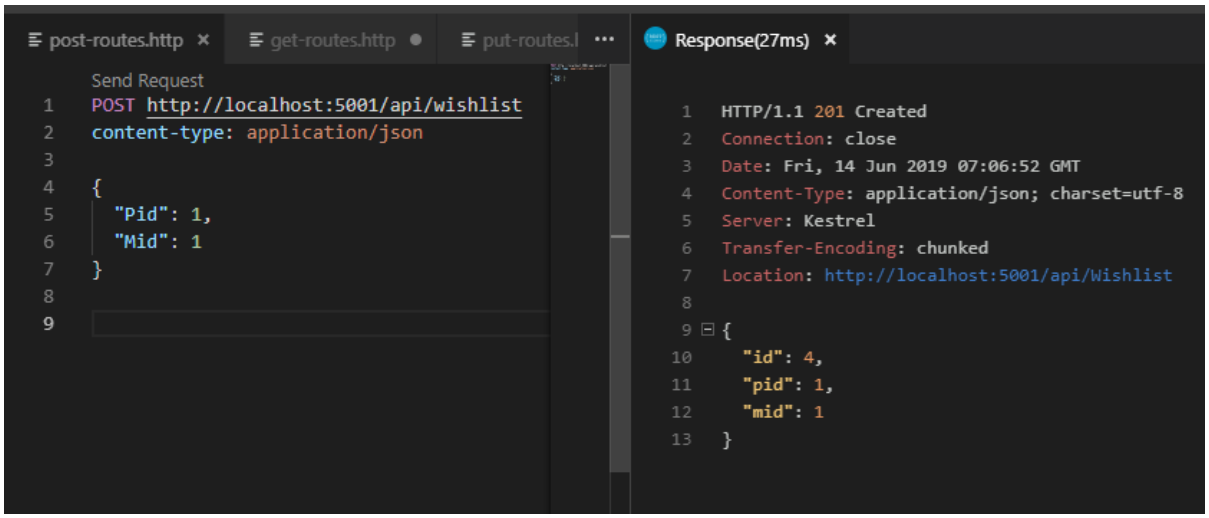
The last id in the Movies table is 4, that's why it's not found.

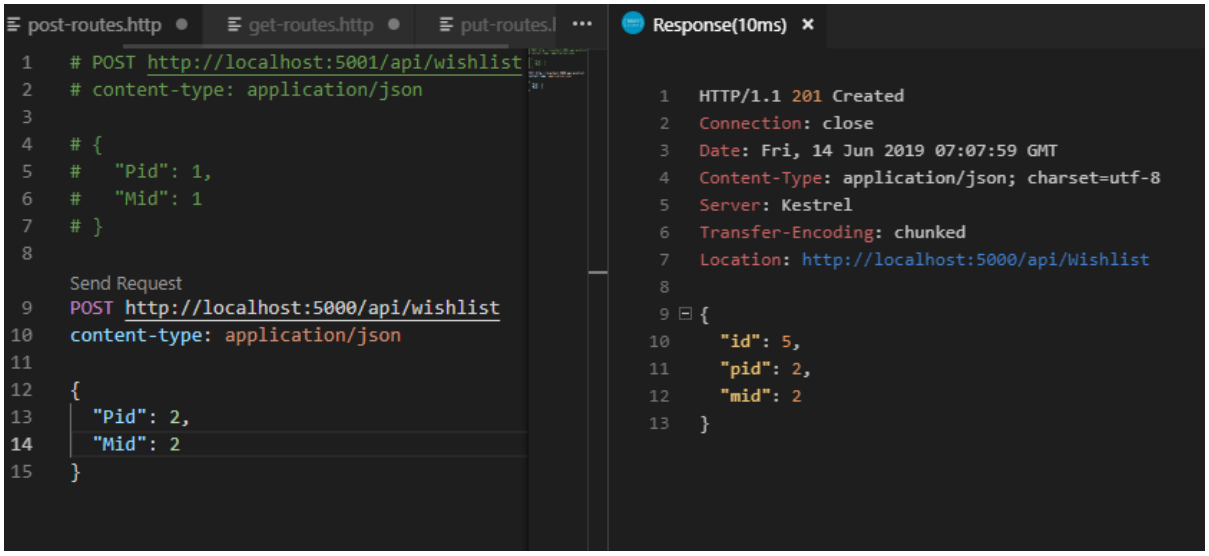




1.2 Post-Routes

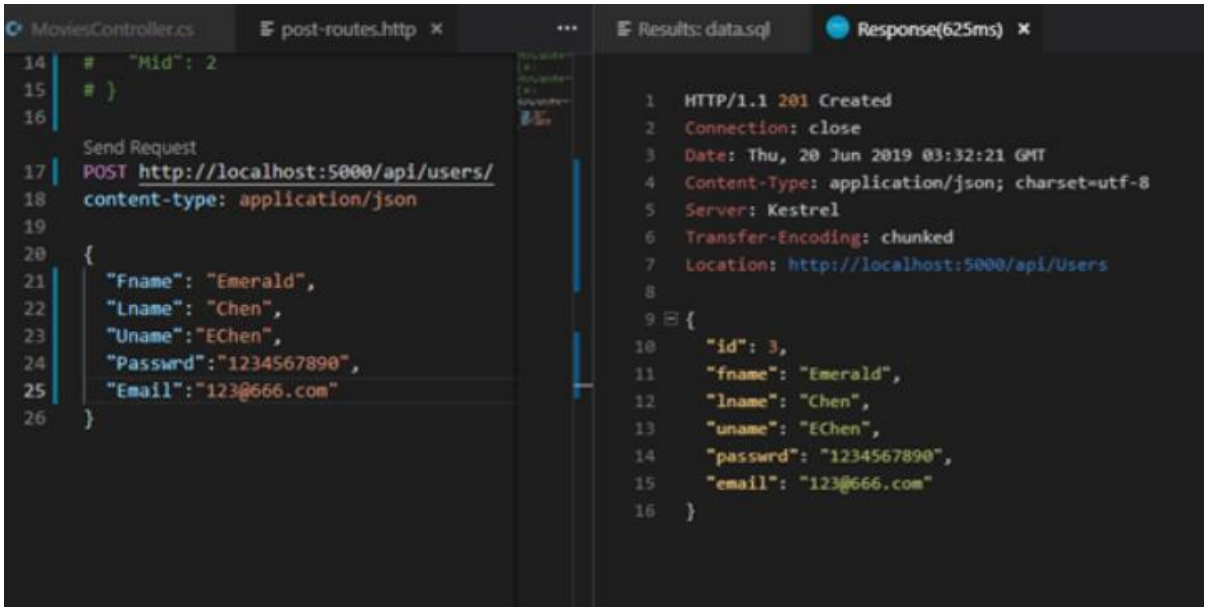
1) POST http://localhost:5000/api/wishlist and
http://localhost:5001/api/wishlist

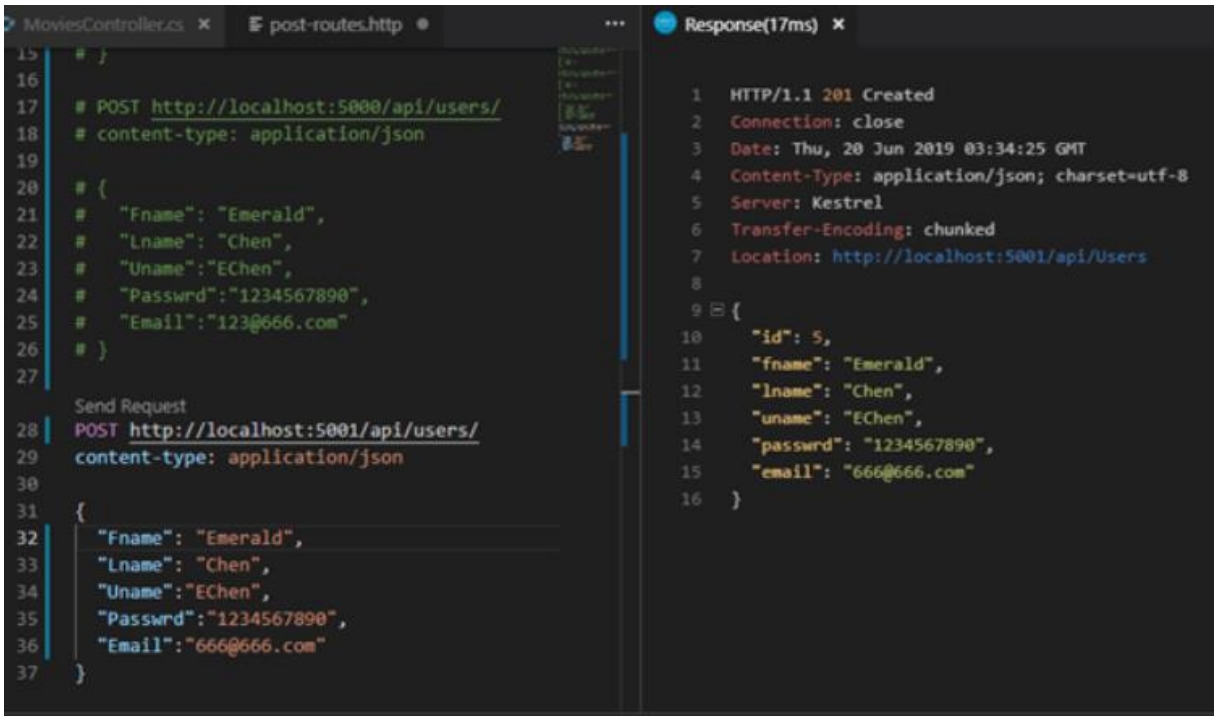




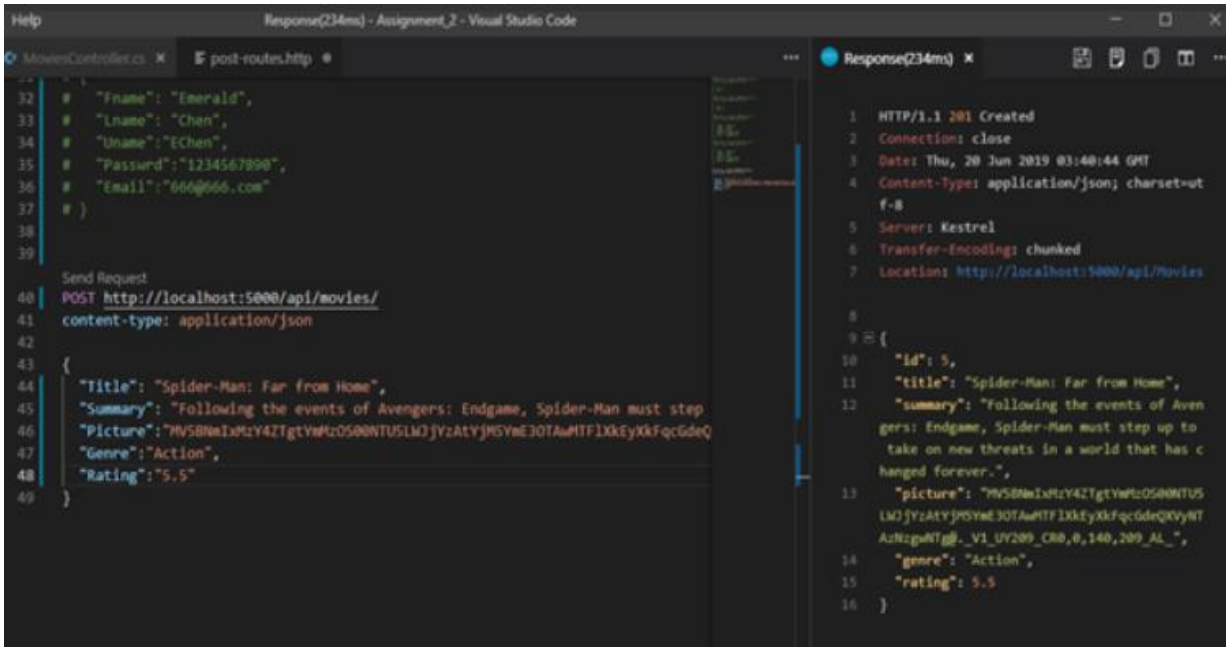
Below are updates for Assignment 3

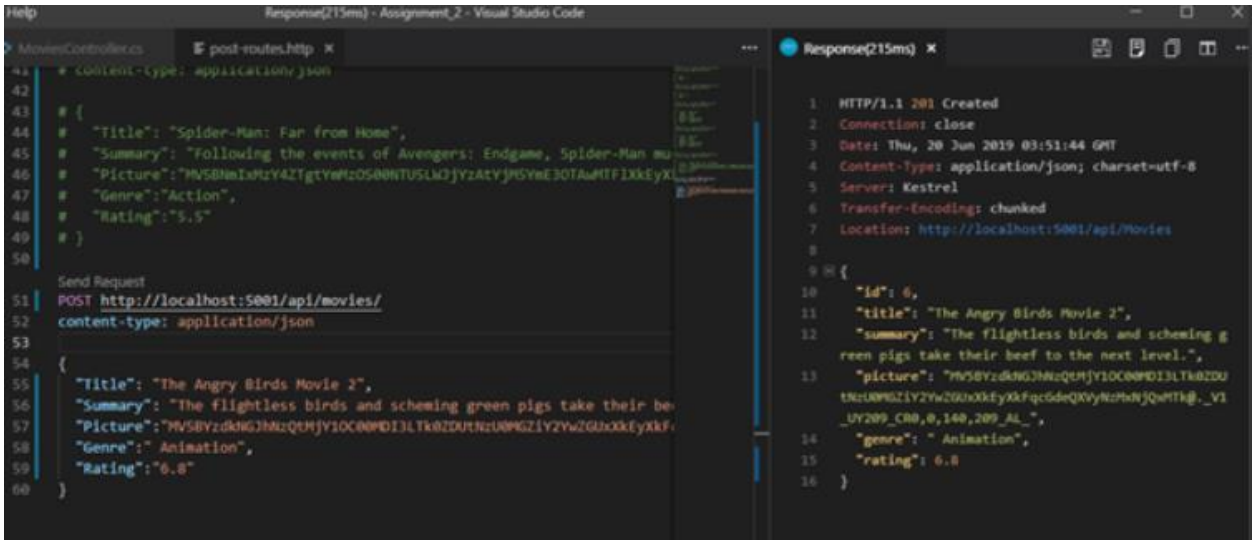
2) POST http://localhost:5000/api/users and http://localhost:5001/api/users





3) POST http://localhost:5000/api/movies and http://localhost:5001/api/movies

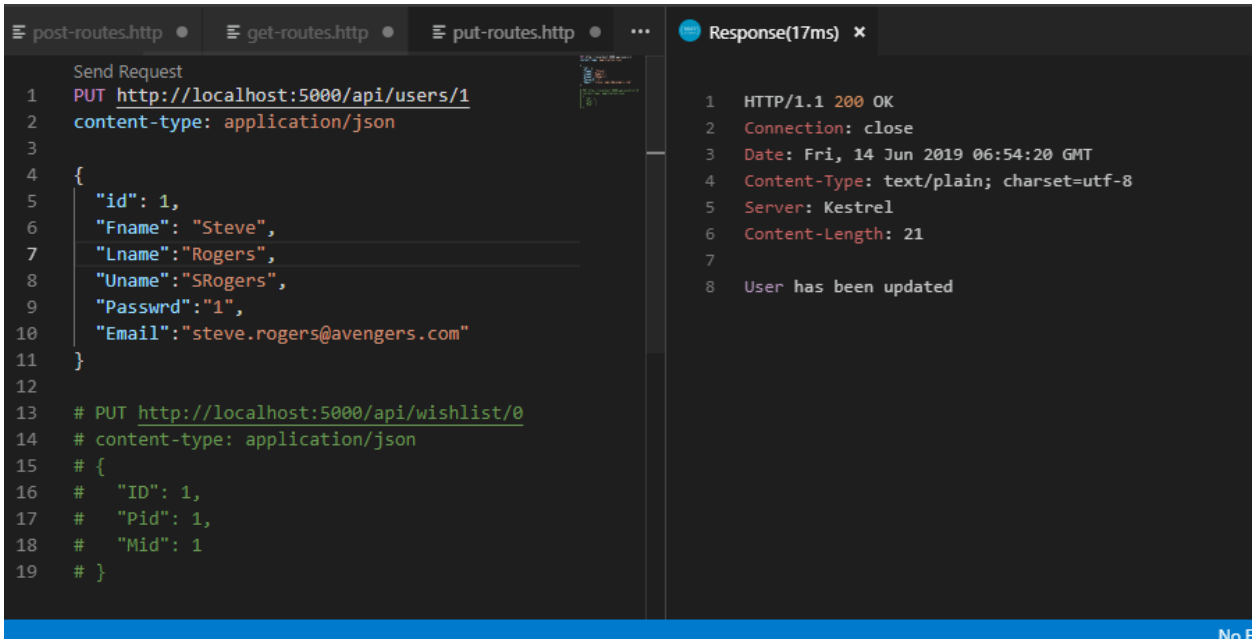


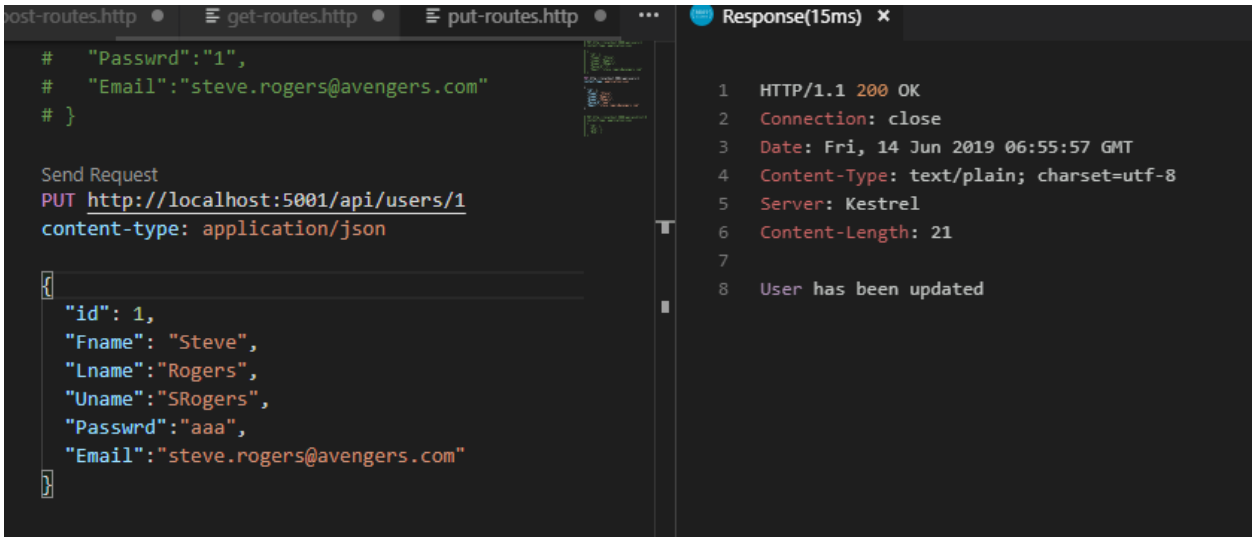


Above are updates for Assignment 3

1.3 Put-Routes

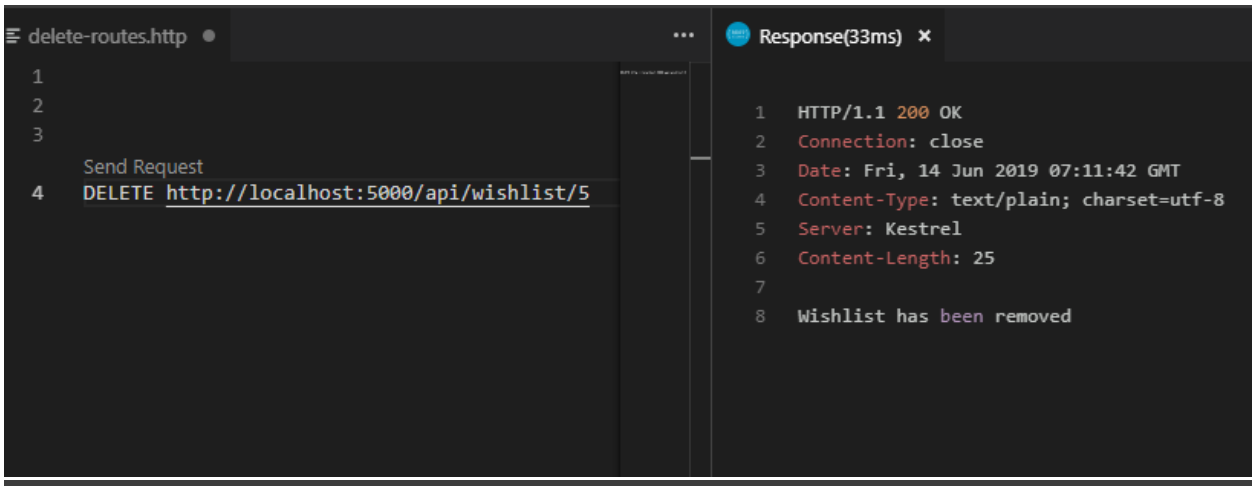
1) PUT http://localhost:5001/api/users/1 and
http://localhost:5000/api/users/1

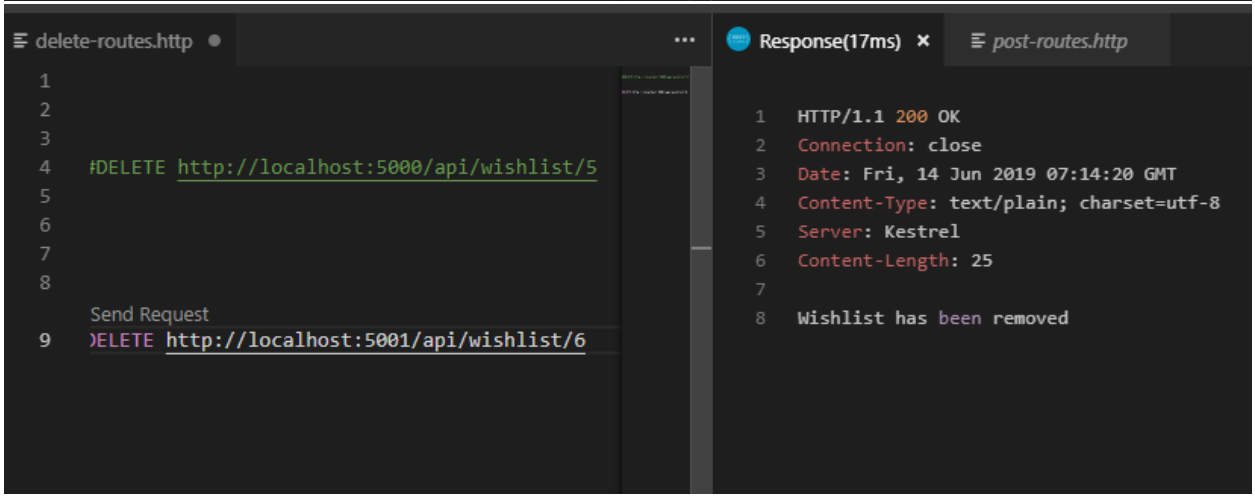


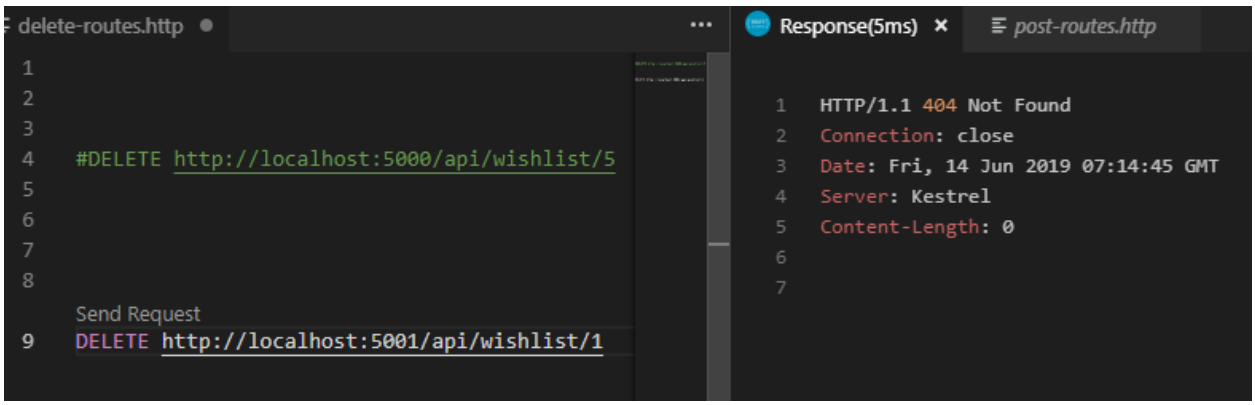


1.4 Delete-Routes

1) DELETE http://localhost:5000/api/wishlist and
http://localhost:5001/api/wishlist







2. Testing the application

The application has been tested by Bodie Keys and Leo Cookson.

Also, they've answered the below 6 testing questions

Testing Question	Test Case	Expected Result	User/Tester Feedbacks
1. Does the login process work properly?	Test the Login functionality to verify login with valid username and password	User can log in by entering the username and the password that stored in the provided database, and then the user can see his or her details in My Account. On the other hand, if either the username or password is invalid, user will be prompted to retry the login.	Yes, it works as expected.
2. Is the user able to update his or her own details but not change his or her username?	Test the Update account details functionality to verify update with valid input	User can update his or her first name, last name, password and email address only. If any field of those details is blank, the user will be prompted to fill in that field.	Yes, it works as expected.
3. Is the user able to search for a movie?	Test the Search movies functionality to verify search with valid input	User can search for a movie that matches the movie title in the provided database. If a movie is not found, the user will be prompted to try again.	Yes, it works as expected.
4. Is the user able to add a movie to his/her wish list?	Test the Add movies functionality	User can add a found movie into My Wishlist. When a movie is added successfully a confirm box will be popped up.	Yes, it works as expected.
5. Is the user able to view a selected movie's details?	Test the View movie details functionality	User can select a movie in My Wishlist and view its title, rating, genres, summary and image. If the user doesn't select any movie then click View details button, he/she will be prompted to select a movie before clicking the button.	Yes, it works as expected.

6. Is the user able to remove a movie from his/her wish list?	Test the Remove functionality	User can select a movie in My Wishlist and remove it from his/her wish list. If the user doesn't select any movie then click Remove button, he/she will be prompted to select a movie before clicking the button.	The movie could be removed but then the app crashes sometimes.
Below are updates for Assignment 3			
7. Is the App user able to register as a new user?	Test the Registration functionality to verify Register with valid input	User can register as a new user by entering the details in the required fields. On the other hand, if any of inputs is invalid, user will be prompted to retry the register	Yes, it works as expected.
8. Is the user able to login as a newly registered user without restarting the application?	Test if the App UI will be updated synchronously with API when a new user is added	After registering as a new user, he or she could click the Login button to the new page which allows the user to log in with his or her new username and password without restarting the App	Yes, it works as expected.
9. Is the user able to add a new movie to the library?	Test if the Add to Database button works right	User can add new movies to the library with valid info copied from imdb.com On the other hand, if any filed is blank when clicking Add to Database button, user will be prompted to fill in all fields	Yes, it works as expected.
10. Is the user able to add the newly added movie to a user's wish list?	Test if the App UI will be updated synchronously with API when a new movie is added	User can add the movie that is newly added to the database to his or her wish list without restarting the application and view that movie's details after selecting it	No really. User can only add the new movie to his or her wish list after restarting the App.

The issue for Question 10 probably could be fixed if the user page was divided into three pages. However, the code for assignment 2 must remain the same. So I didn't revise the code to test it that way.

Above are updates for Assignment 3

Unfortunately, the app might get crashed after removing a movie from the wish list.

The error is caused by some outranged index. Not enough time to fix that bug.

3. The reasons for choosing WPF for GUI

1) In WPF (Windows Presentation Framework), UI elements are designed in XAML while behaviours can be implemented in C#
2) WPF data binding provides a simple and consistent way for applications to present and interact with data
3) WPF provides extensively scalable for the UI elements in applications
4) I've learned the basics of WPF so that it saved me time to pick it up and used it for this assignment