An-Najah National University

Distributed Operating Systems

A Multi-tier Online Book Store

Instructor: Samer Arandi.

Student: Laila Abu Safiya.

Introduction:

In this lab we are required to design Bazar.com using two tier we design (front-end and back-end), also we are recommended to use a lightweight micro web framework.

So, in my solution I use a Node JS frame work, its uses JavaScript on the server also its run on various platforms, also I have a good knowledge on it.

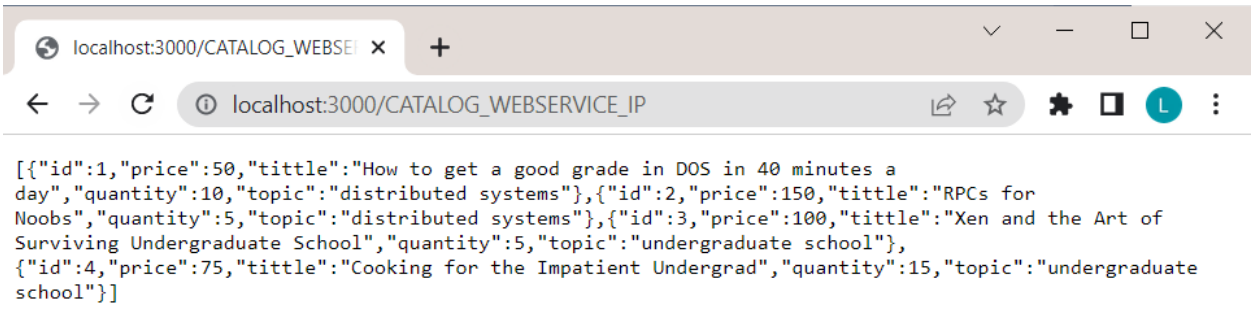
In my program I have two files (catalog.csv & order.csv) which its consider as the database of servers that have the information for our program. At first in my program, I read each of them and store them in array of objects (catalog, Book_Sold) in JSON format to use it in our API's.

Also I use Express which is flexible Node.js framework, and its help to create a robust API in quickly and easy way.

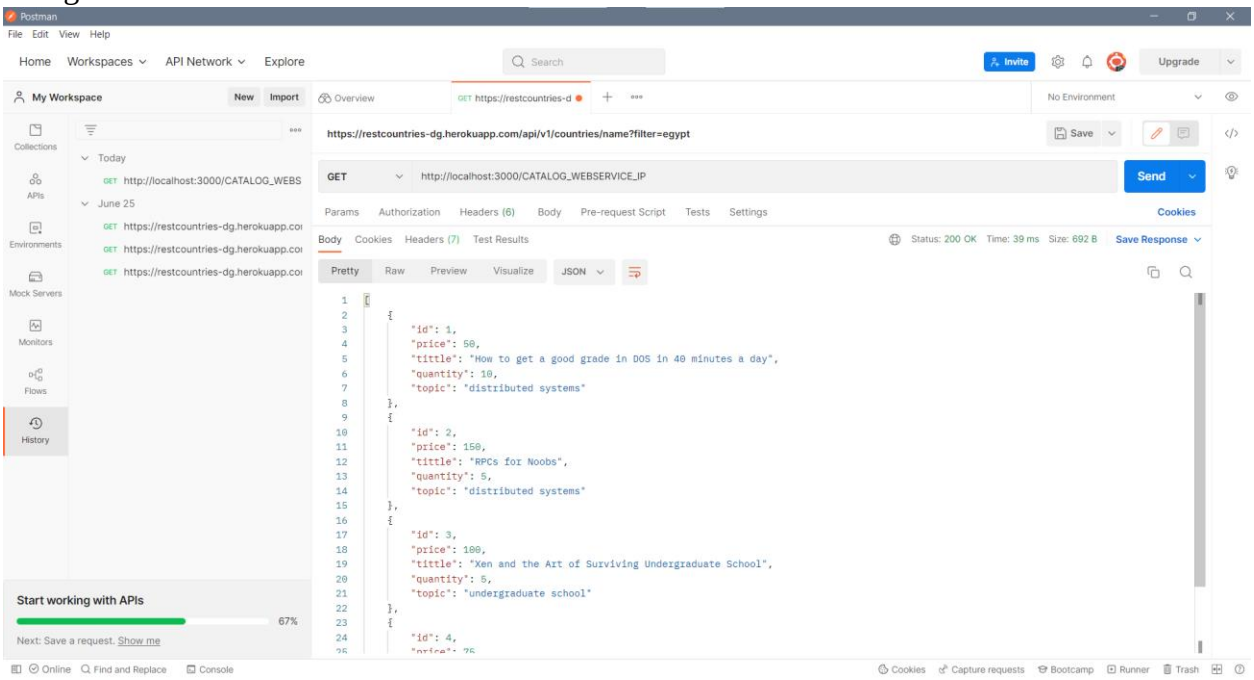
Procedure:

Then I build the microservices requested:

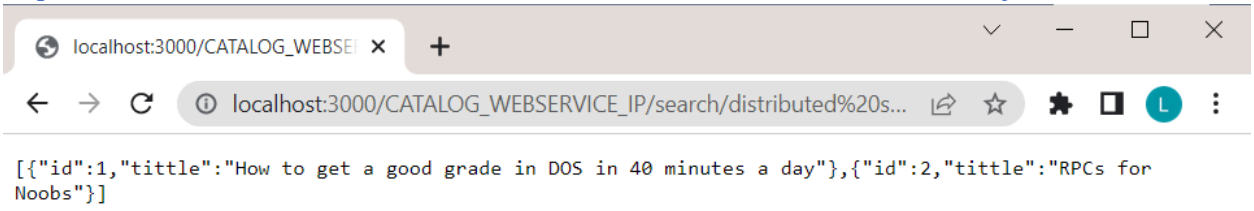
- /CATALOG_WEBSERVICE_IP:
  This URL will return the list of all Book in our store from catalog server.



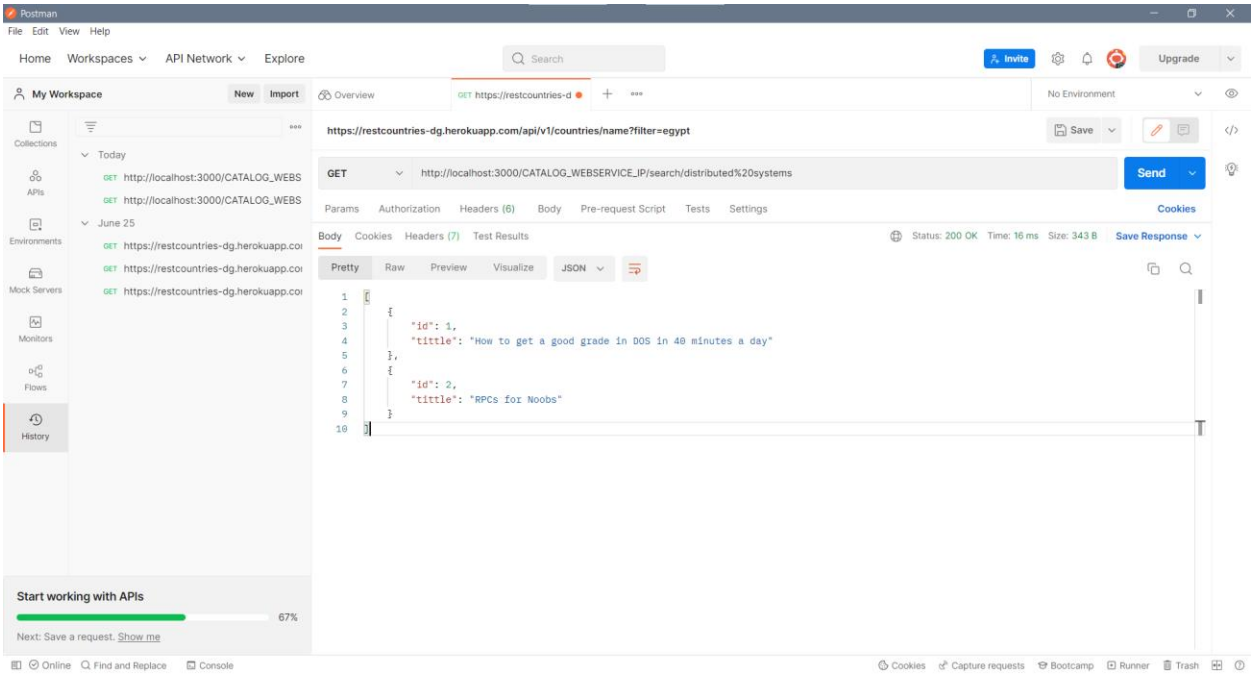
Using Postman:



- /CATALOG_WEBSERVICE_IP/search/:itemName
  Using this we can search about specific books that belong to a specific group using the group topic, so it will return all information about the books, here it will return (id, tittle), in case the group not found its will return an empty array and set the status to **not found.**
  Example:
  http://localhost:3000/CATALOG_WEBSERVICE_IP/search/distributed%20systems

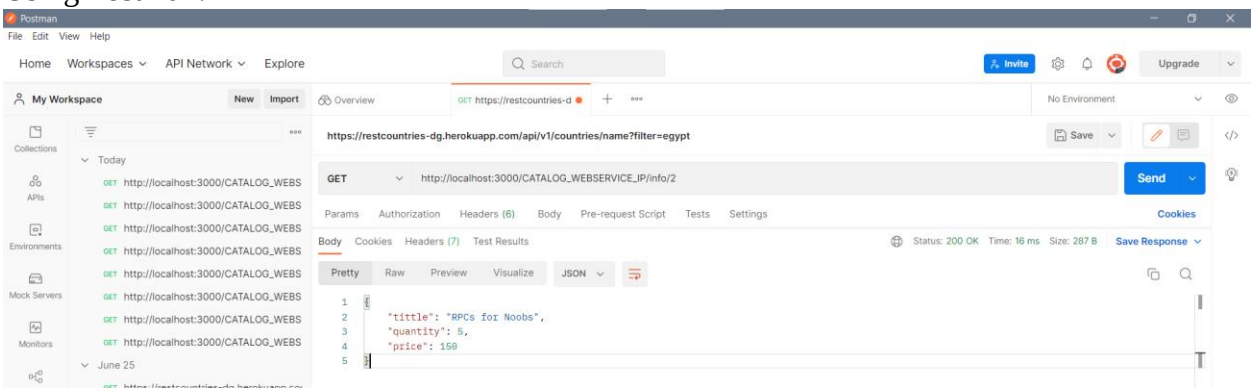


Using Postman:

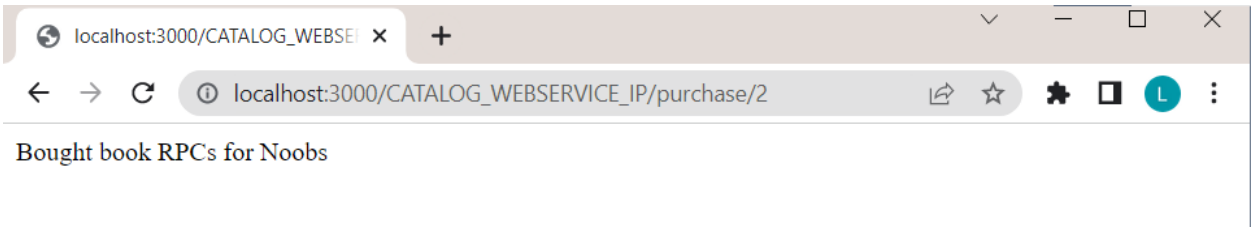


- /CATALOG_WEBSERVICE_IP/info/:itemNUM
  Using this microservices we can get all information about specific book by sending book number in URL.
  Example:
  http://localhost:3000/CATALOG_WEBSERVICE_IP/info/2

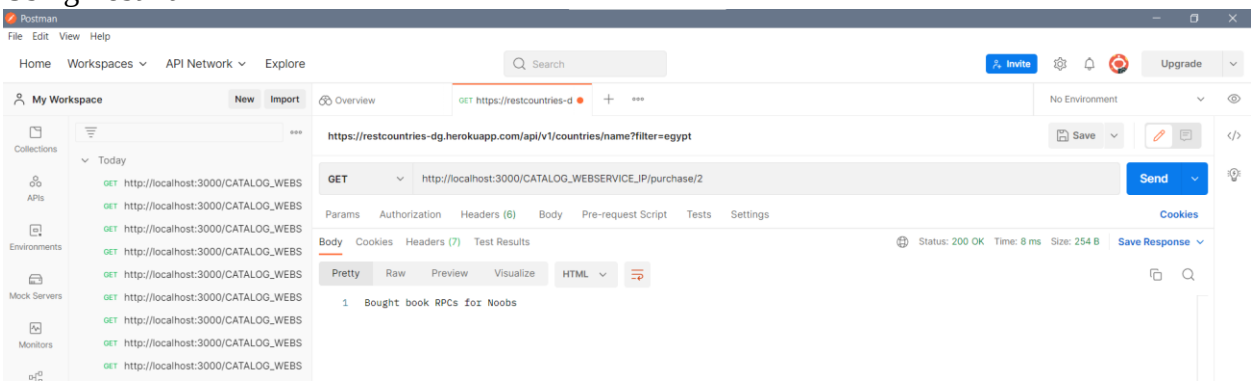
Using Postman:



- /CATALOG_WEBSERVICE_IP/pruchase/:itemNUM
  We can use this microservices to specifies an item number for purchase, so its will increment the number of books sold in order server and decrement the quantity exist in catalog server, we do this operation by rewrite the new information to our files.
  Example:
  http://localhost:3000/CATALOG_WEBSERVICE_IP/purchase/2
  it will display Bought book (name of the book).



Using Postman



Improvement:
We can improve our program by send the quantity of how many books we will buy, that's when we send a purchase in URL, we can send another number which consider of how many books we will buy.
It will be something like this:
http://localhost:3000/CATALOG_WEBSERVICE_IP/purchase/2/3
which mean that we will buy 3 books from the book that have the id = 2.