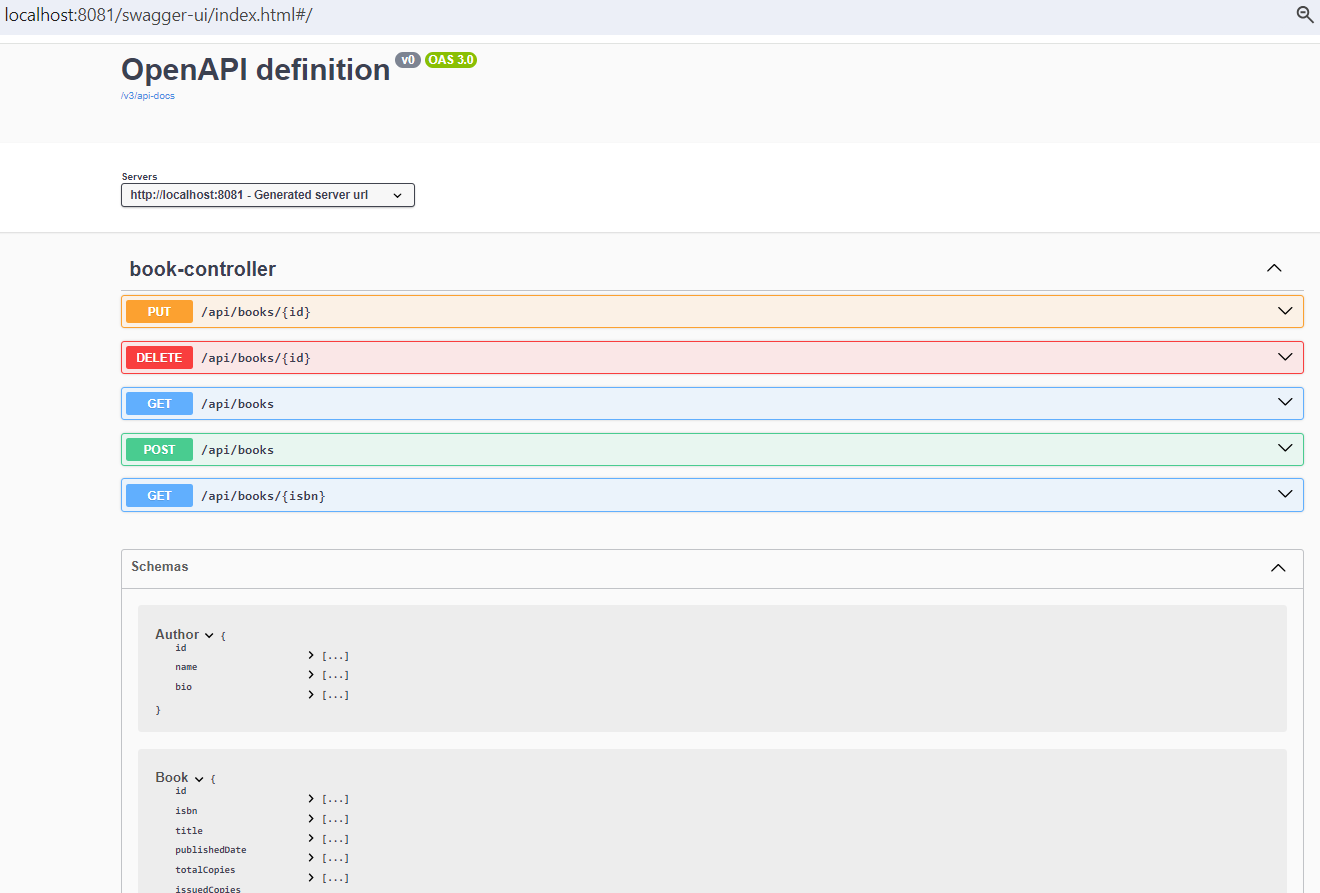
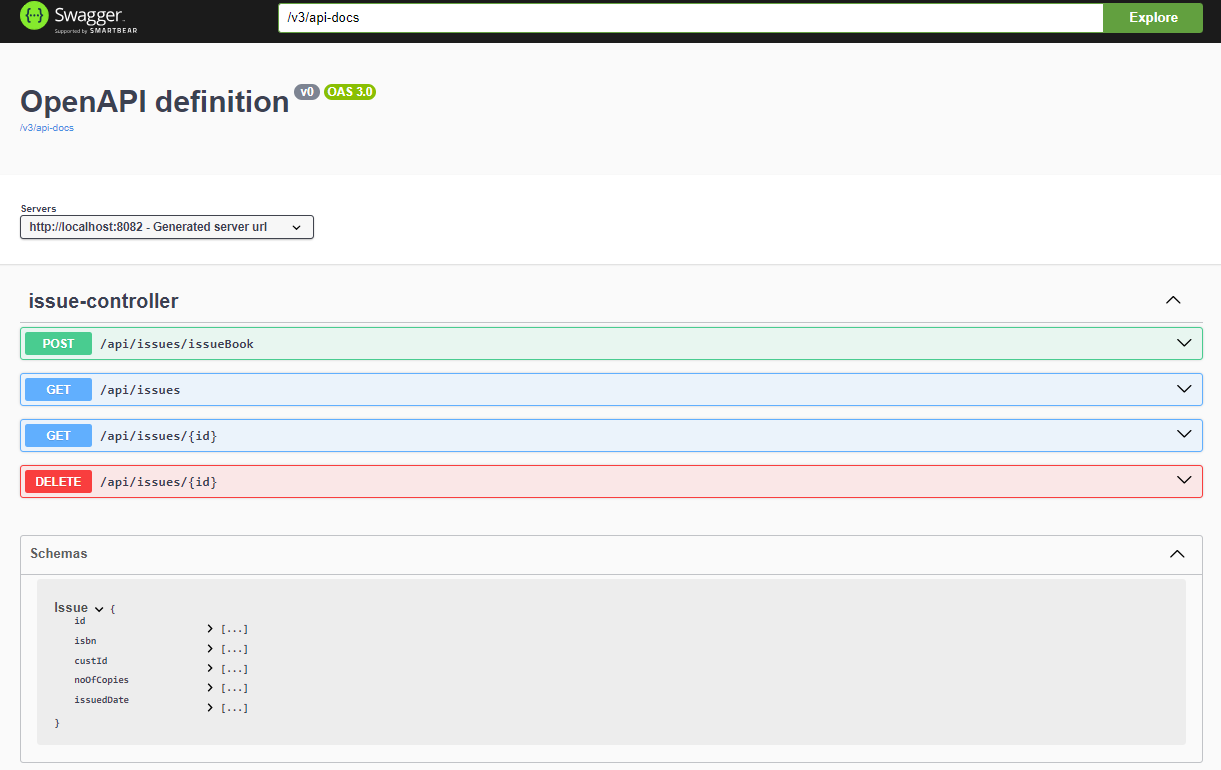
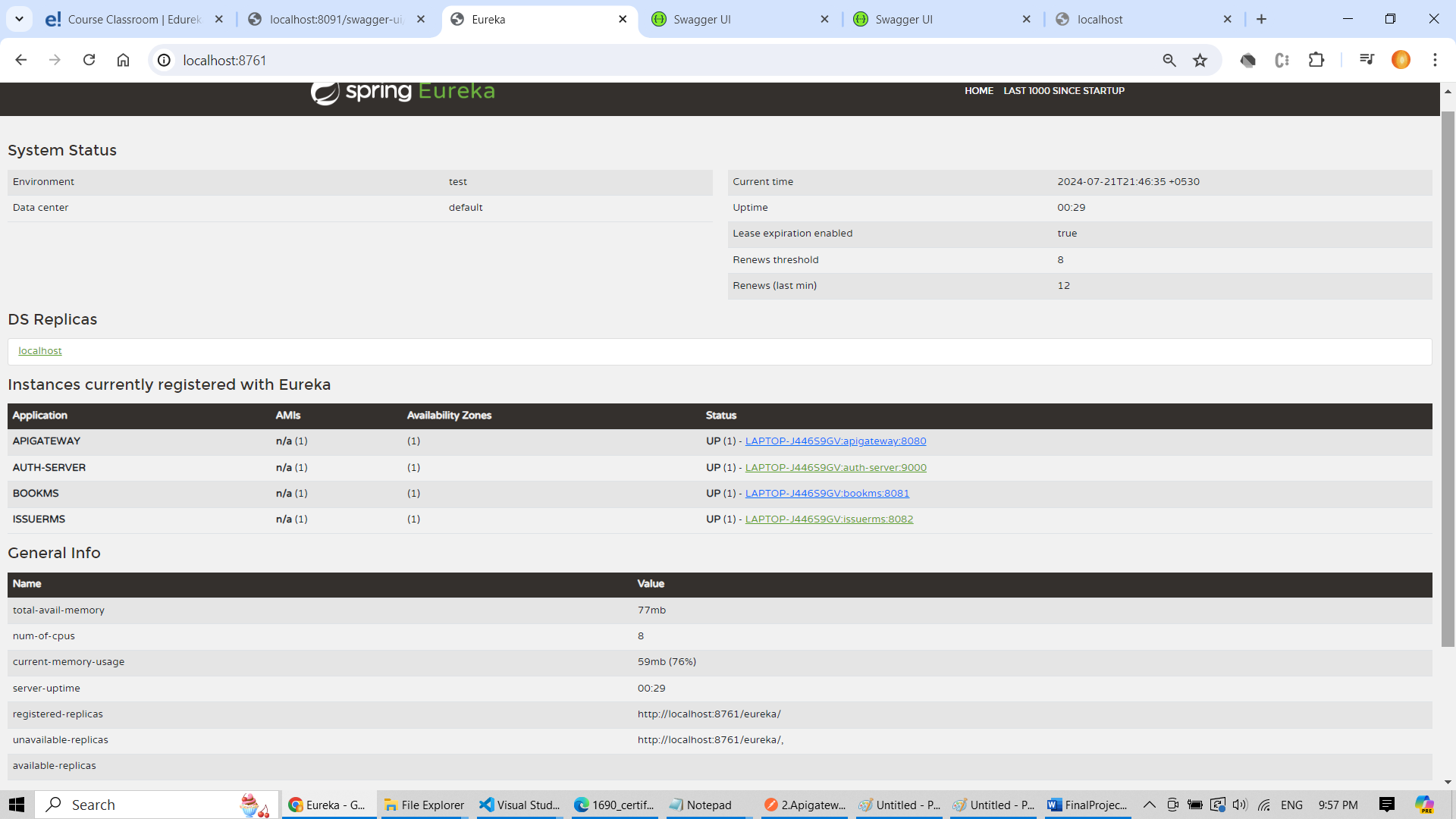
1. Built bookms with following functionalities and datamodel



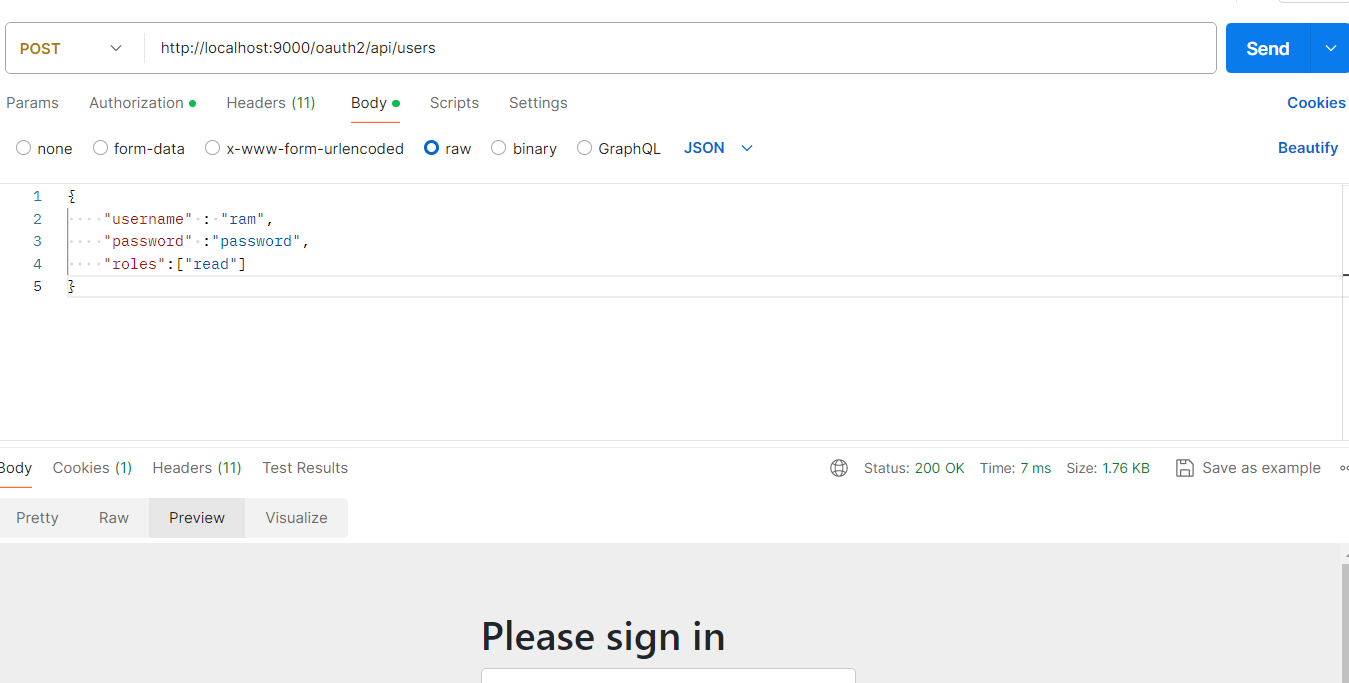
1. Built issuerms with following functionalities and datamodel



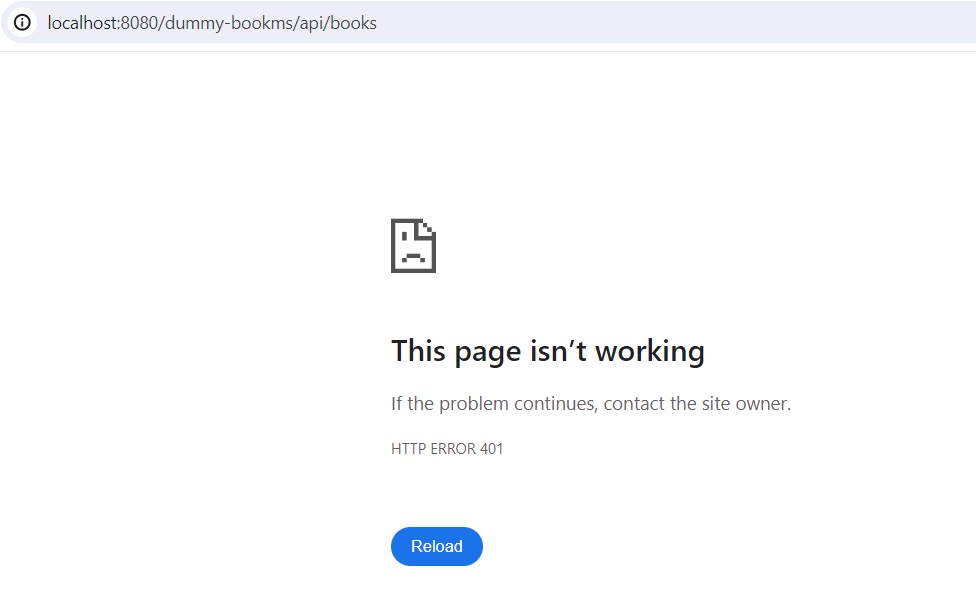
1. Built eureka server and made auth-server, api gateway , bookms, issuerms as eureka clients. Issuer ms is a load balanced web-client which can call bookms server to issue the books/to cancel the issues.Apigateway addresses all the cross cutting concerns



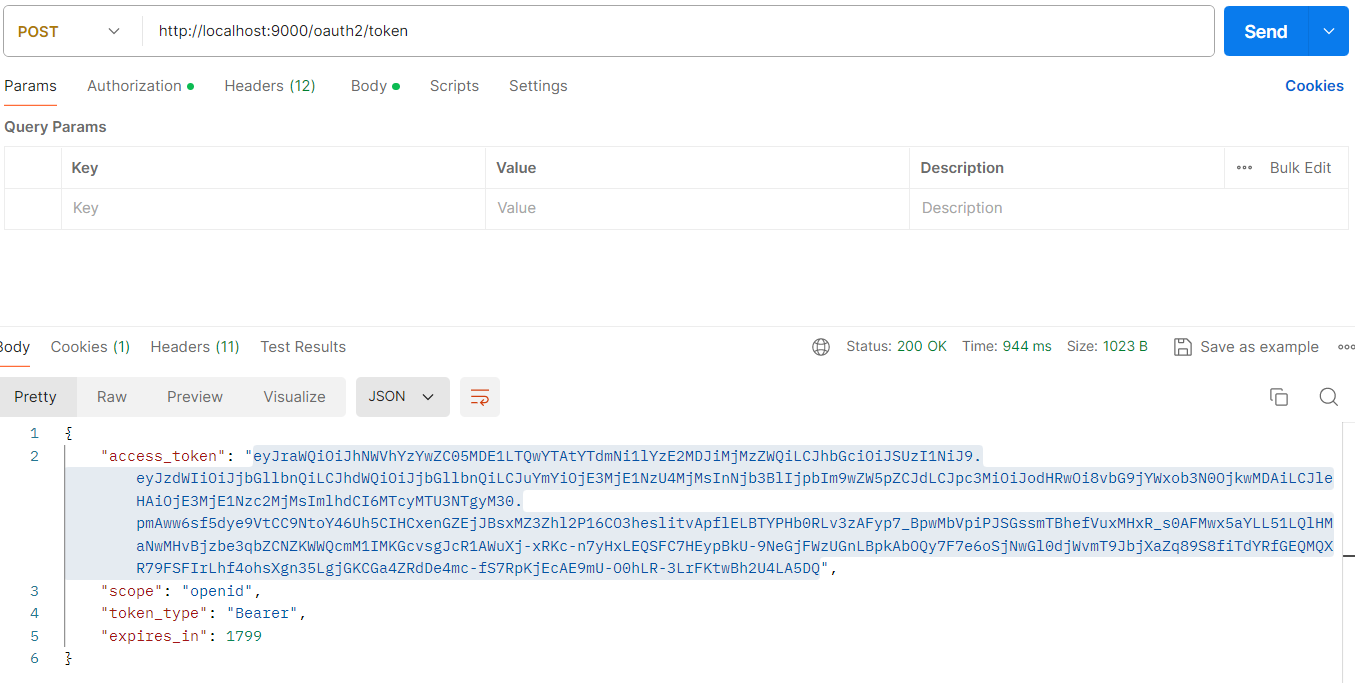
1. Auth-server ms is equipped to add /update / delete users. Following screenshot for adding user.

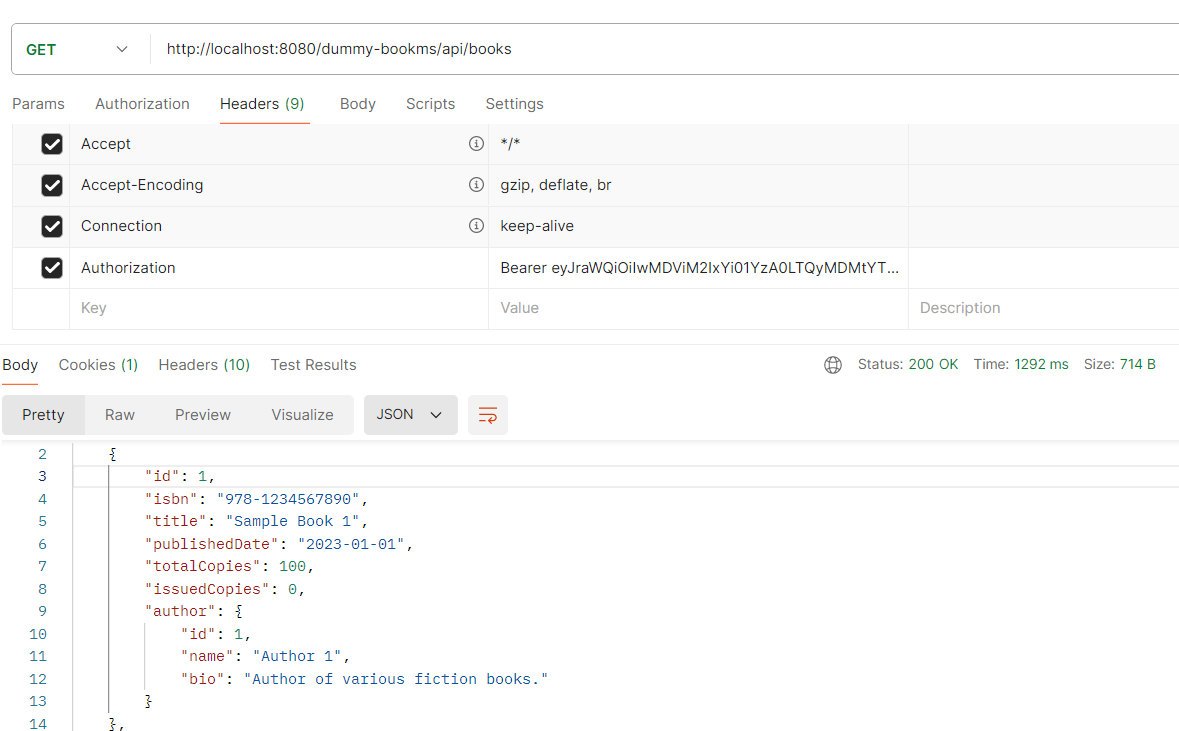


1. Once the apigateway is configured as resource server, its not possible to retrieve webservers endpoints either from webclient or by human user

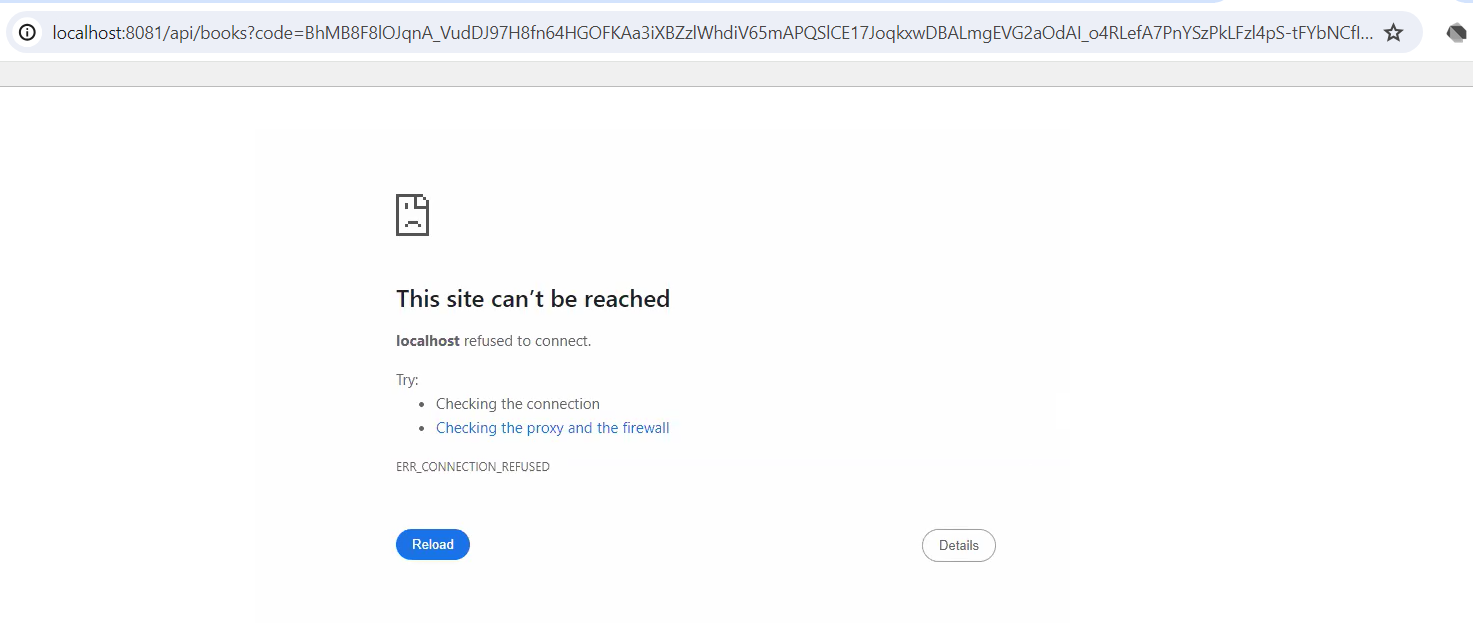


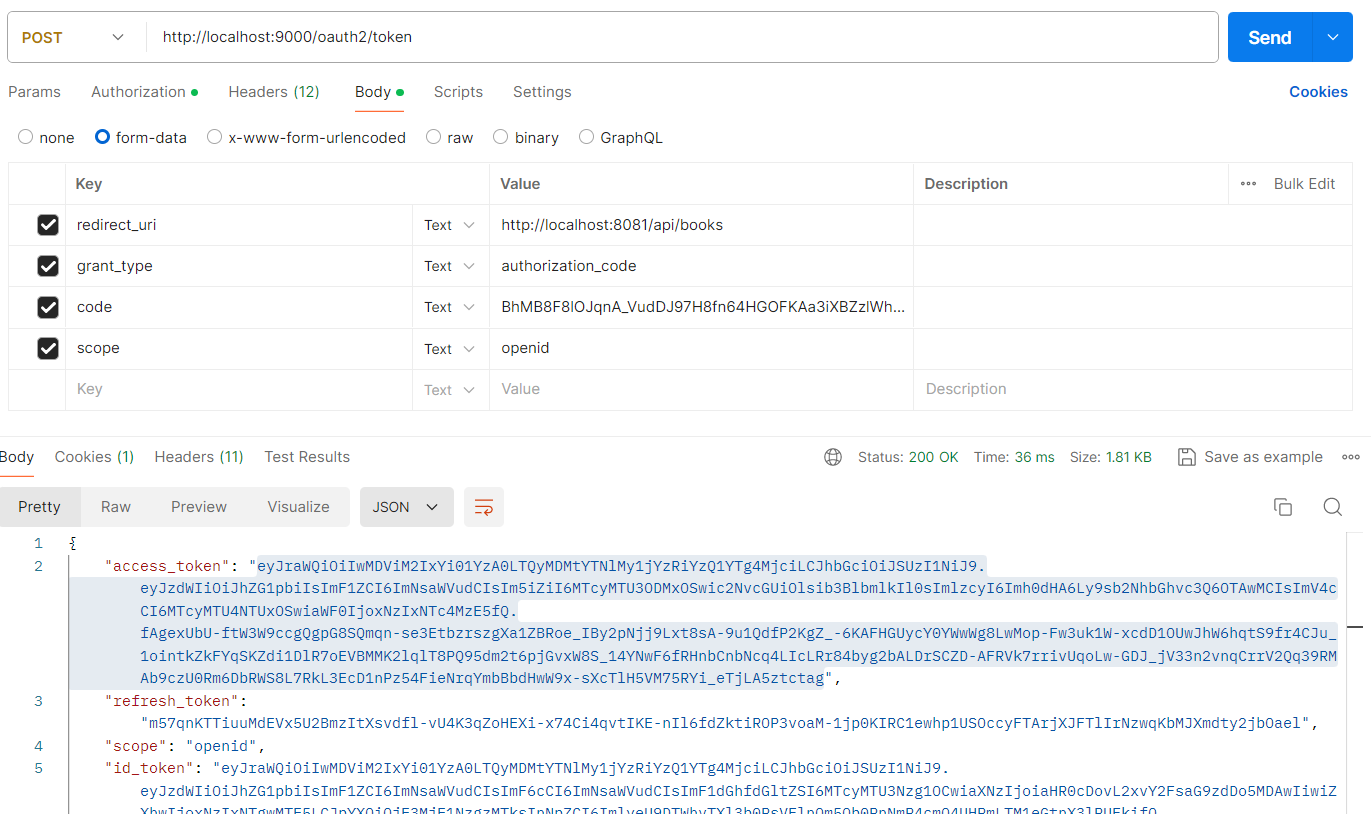
1. Webclient accessing webserver using client-credentials



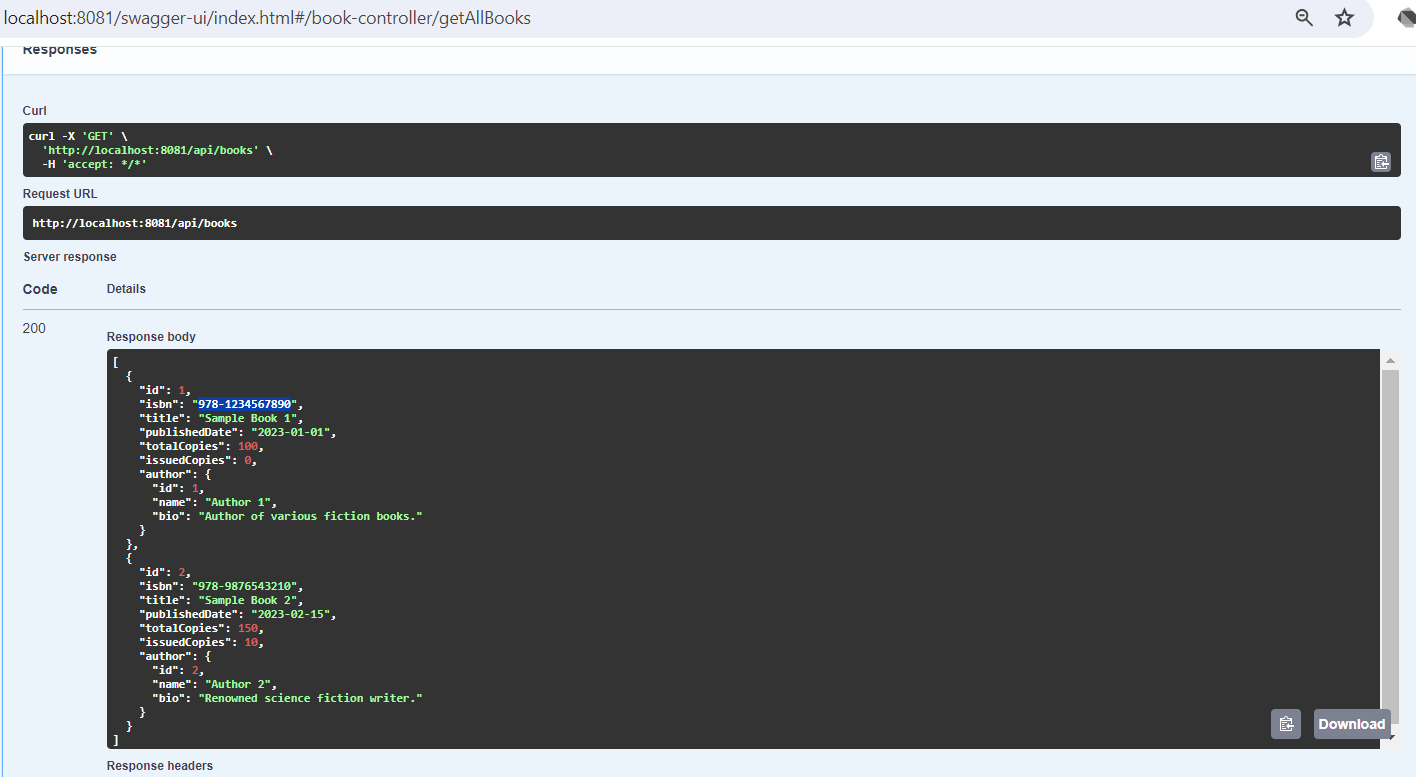


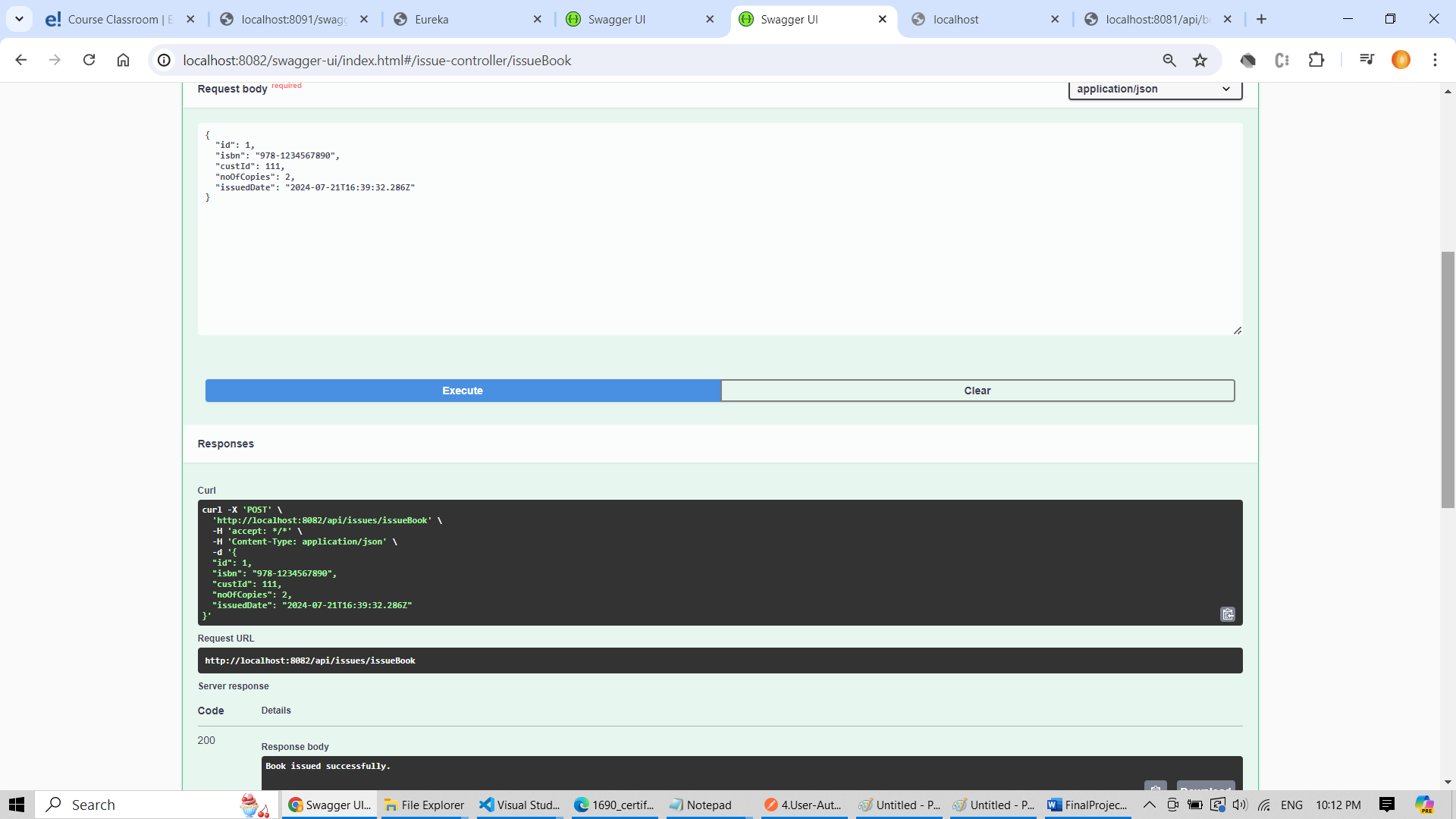
1. Humanuser accessing webserver using Temporary token,Accesstoken,Refreshtoken . Hitting <http://localhost:9000/oauth2/authorize?response_type=code&client_id=client&scope=openid&redirect_uri=http://localhost:8081/api/books> from browser and passing the credentials of newly created user “ram” whose password is “password”

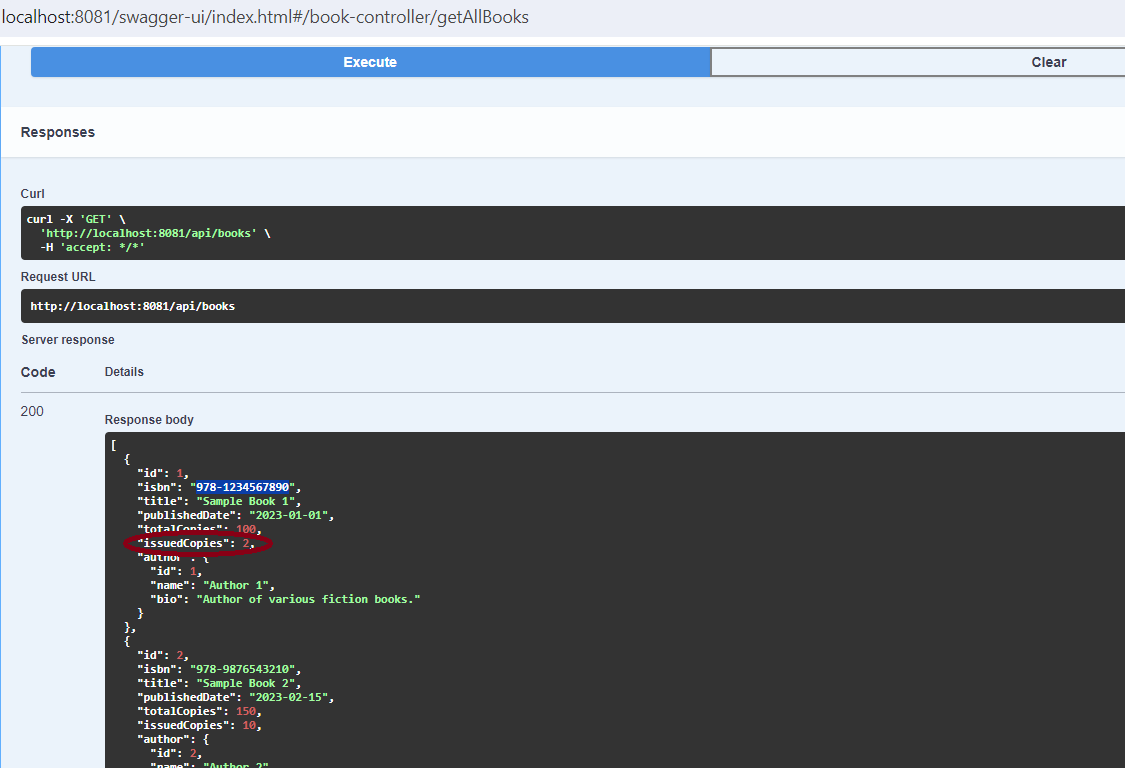




1. Issuerms can update the bookms using webflux. In Bookms, for isbn, issuedCopies initially zero. Once its issued using issuerms, its changed to 2. This is illustrated in below screenshots







1. Deploy the instances in docker and launch it

