# Overview of program:

Backend: Spring boot,

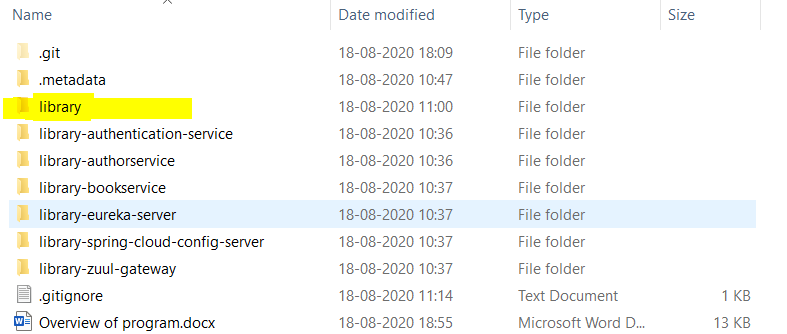
Database: PostgreSQL

Frontend: Angular

Tools used:

Maven, Spring data JPA, Spring tool suite

Front end:

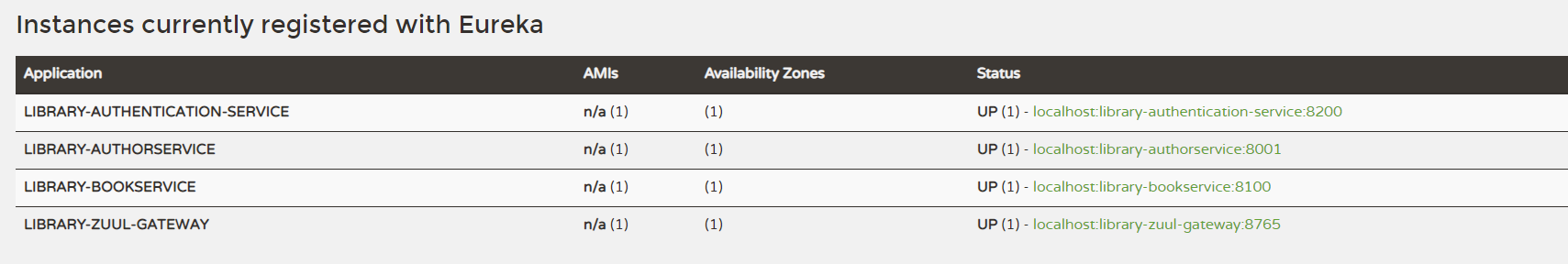


Library folder contains the frontend code

Open and

Run npm install

# Backend Microservices:



Client-side Load Balancing is done using Ribbon

# Ports used:

Author service-8001

Book service – 8100

Authentication service 8200

Zuul gate way -8765

Eureka -8761

# Backend:

## Library-eureka-server:

This is the naming server.

## Library-zuul-gateway:

API gateway this forwards the request appropriately to rest of the microservices

## Library-authentication-service:

This microservice take cares of the authentication

### Functionalities:

1. login signup
2. generating JWT token
3. validating the token

## library-authorservice:

This microservice serves the request related to author details

### Functionalities:

1. add a author
2. editing author details
3. deleting author

## library-bookservice:

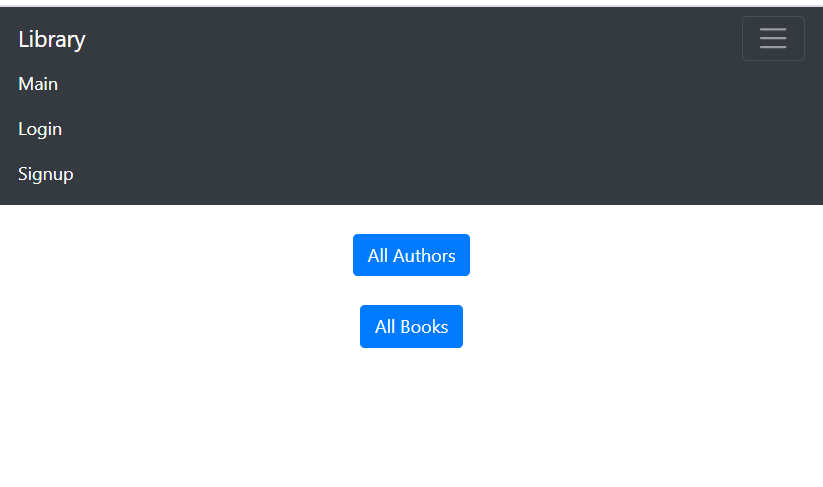
This microservice serves the request related to Books

### Functionalities:

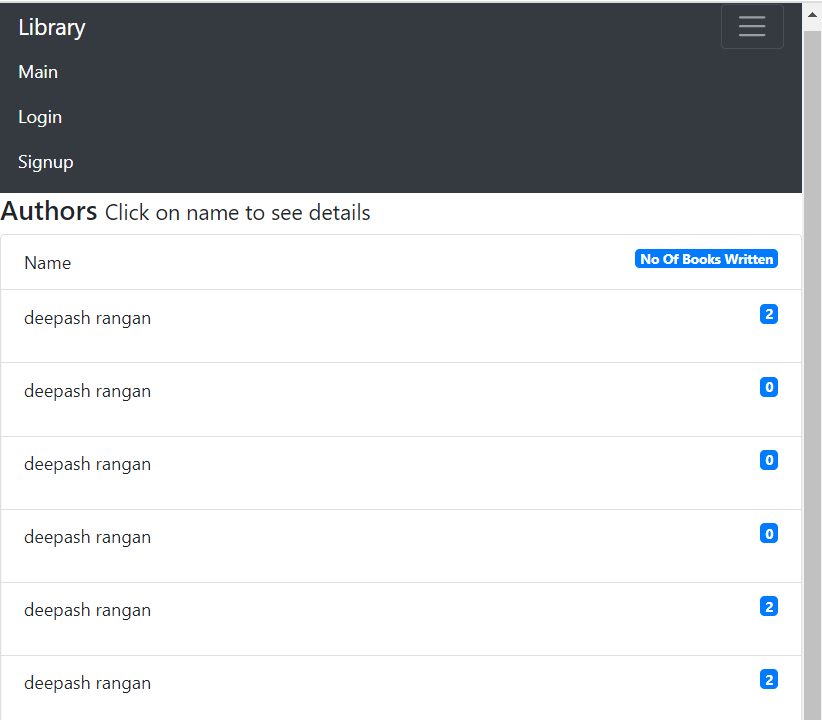
1. add a book
2. editing book details
3. deleting a book

# output:

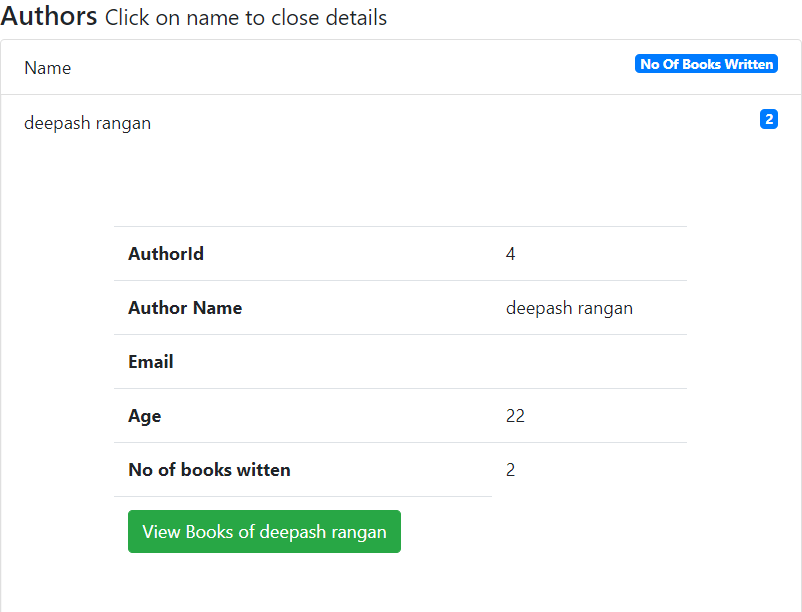
## Main Page:



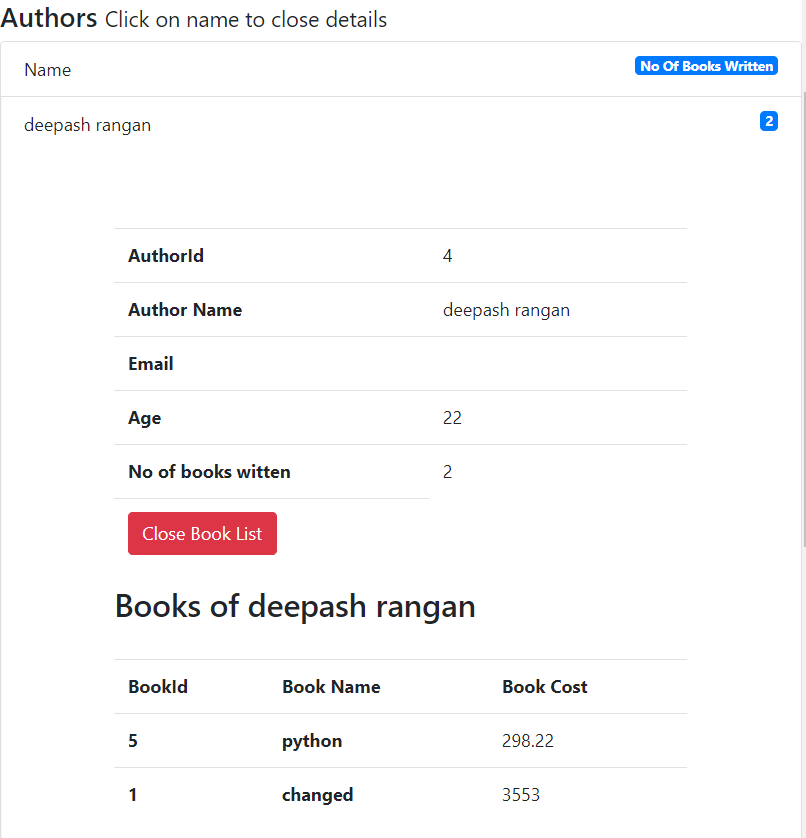
### All Authors:



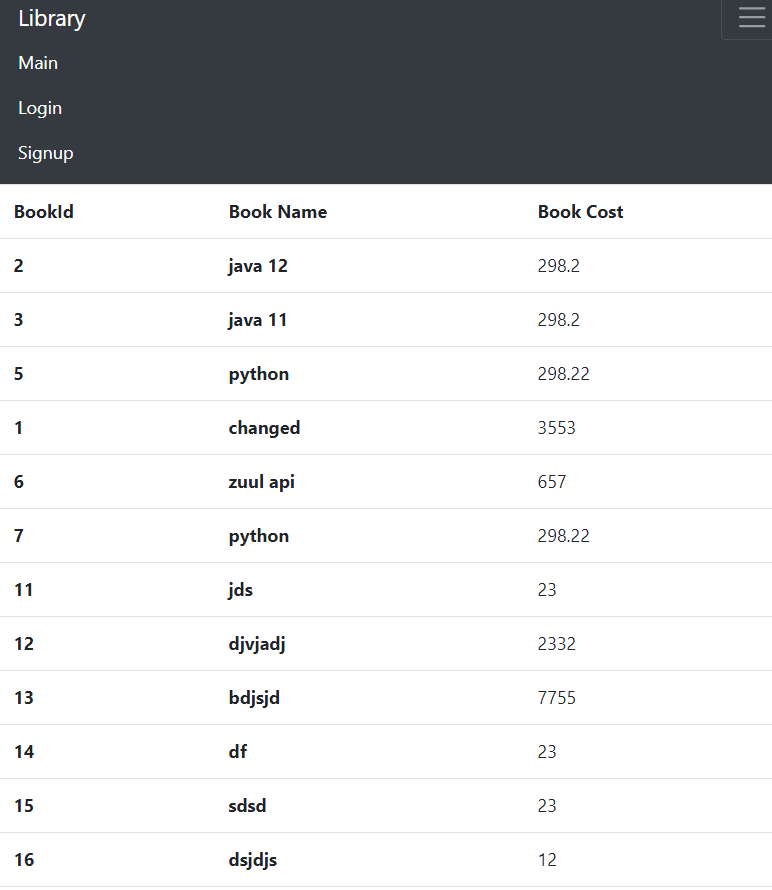
Click on author name to see the details of him:



View books to see books written by that author:



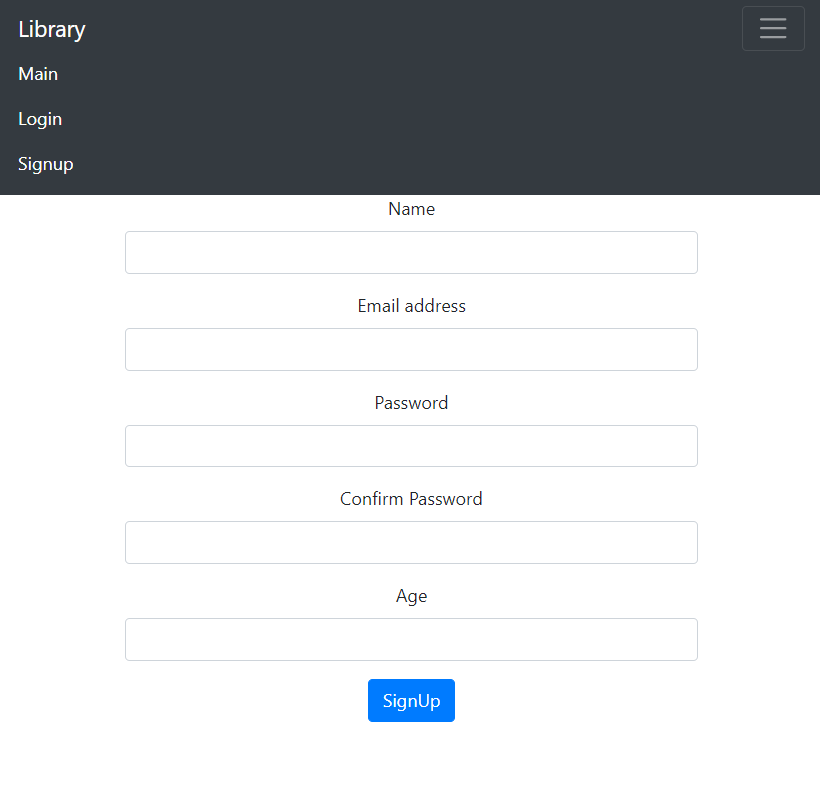
### All Books:



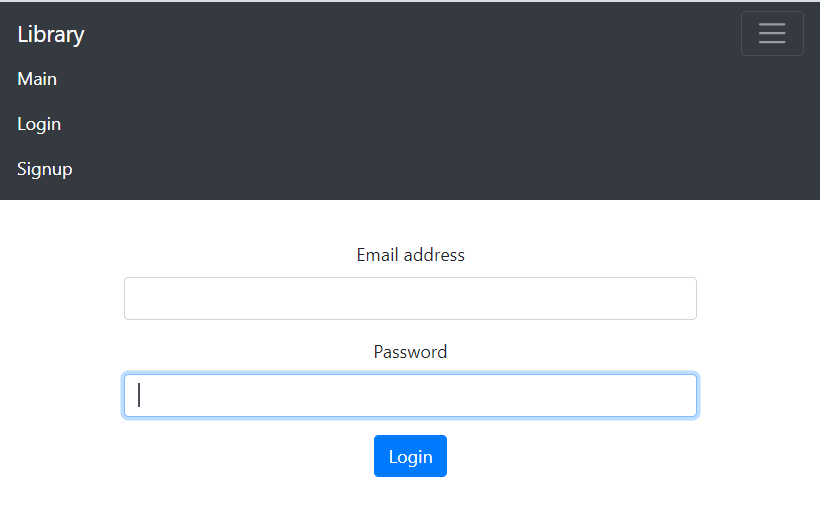
This returns all the list of Books

## Signup:

Author can register himself



## Login

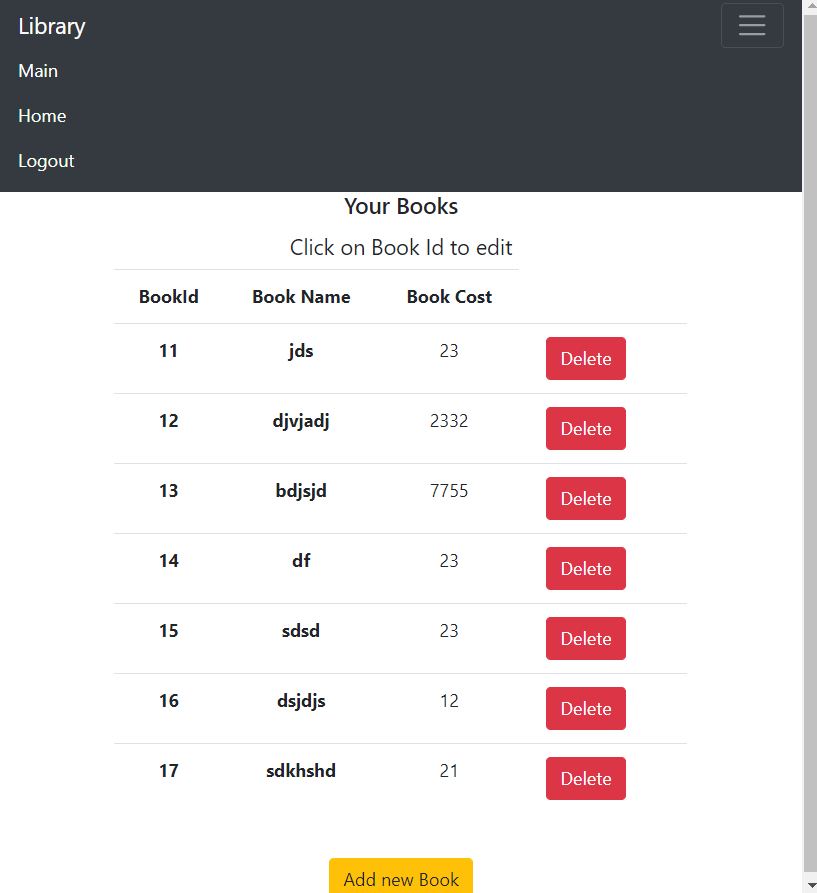


## Author Home:

After logging In

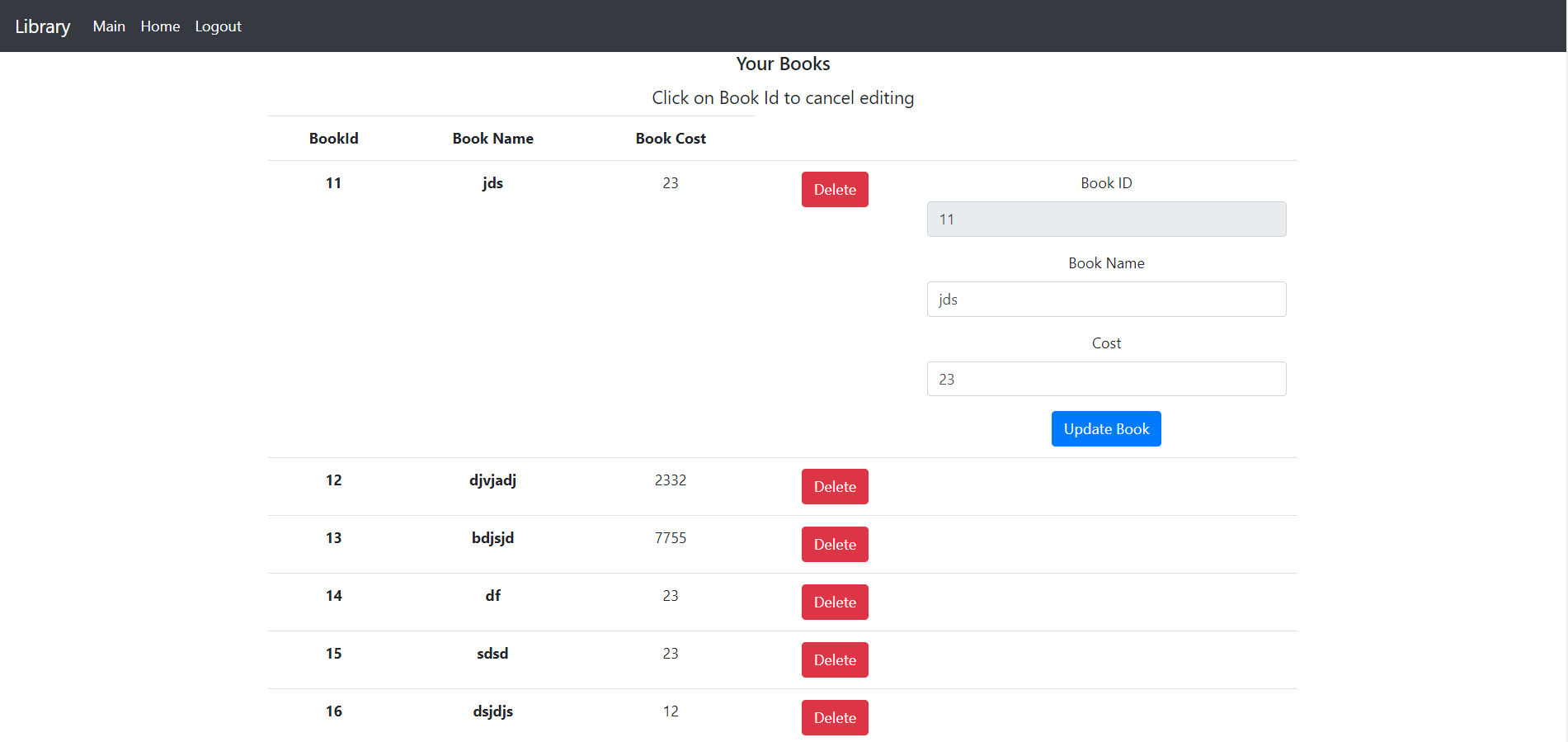
The Token is generated and stored in session storage:





Author home here he sees the list of books added by Him

### Click on Book Id to edit the details of a Book:



### Add Book:

