DigiCert Assessment

Problem Statement

Write a minimal RESTful library API to perform query all books in the library and CRUD operations on a single book.

Solution

The application was written in Java using Java 17.

Maven was selected as the build and dependency management tool for the project.

Spring boot framework was chosen as it provides the following benefits:

- Rapid Application development, able to create standalone applications
- Simple to create and configure projects
- Embedded Tomcat server and in memory H2 DB
- Actuator feature for health monitoring and metrics
- Popular framework of choice to containerize microservices – can run directly in Docker

CRUD operations are performed using Hibernate ORM. This was the preferred solution over using JDBC as the code is now database independent, there is no need to write db specific queries. Objects are mapped easily to the database schema.

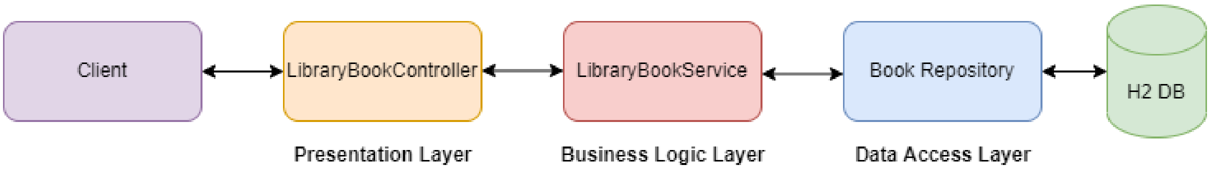
Unit testing of the API's is implemented using Junit and Mockito.

Basic Authentication is enabled on all endpoints in the application, this is added as its best practice to secure API's by an authentication mechanism of choice.

Exceptions are handled in a standardized manner, the API will return a standard response object with the respective error message in the response body.

API documentation is provided by means of a Swagger document. This document describes the requests and responses of the API's.

Project flow and Architecture



The project is structured in this manner to separate the business logic from the presentation layer and the database layer. This creates separation in all the components for ease of testing and maintenance.

- The client (browser/consumer of api) will make an HTTP request to the Controller .
- The Controller will make a call to the business layer
- The Business layer will then call CRUD options in the database layer.

Instructions

The application will run locally on port 8080.

Auth Details

The endpoints are secured with basic authentication

User: appuser

Password: appuser

URL Endpoints

Actuator: http://localhost:8080/actuator

Swagger Doc: http://localhost:8080/swagger-ui/index.html#/

Postman Collection is also included.

Use the following comman to run the JAR file:

```
java -jar library-0.0.1-SNAPSHOT.jar com.assessment.library.LibraryApiApplication
```

Source code can be found in Git Repo:

https://github.com/kayl-rsa18/Assessment.git