**### Implement JPA data access implementation and data layer**

Implemented and Service Layer and DAO layer TravelService and TravelRepository

Please refer packages in source code:

com.afkl.travel.exercise.service

com.afkl.travel.exercise.serviceImpl

com.afkl.travel.exercise.repository

**### Implement the provided `OpenApi` specification.**

Implemented RestController TravelController with the specified methods:

Please refer package in source code:

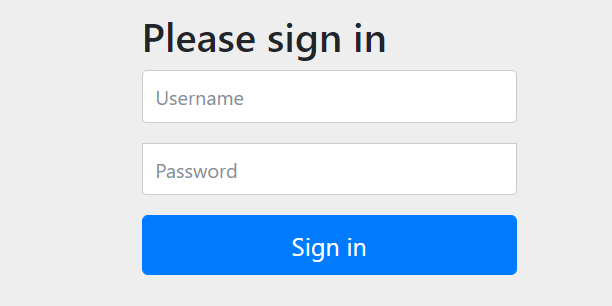
com.afkl.travel.exercise.controllers

**### Add statistics for your backend**

Added functionality for below Actuator functionalities by adding Security class WebSecurityConfiguration, These can be accessible by using configurable username/password in **application.properties.** You will see below login window executing below url in browser.

<http://localhost:8050/actuator/health>

<http://localhost:8050/actuator/metrics>



**Credentials**: ops/psw

**### Make the application configurable**

application.properties has the entire configuration. Security password, DB configuration, actuator configuration, RestController header username/password are all configurable in application.properties.

**### Create a client**

Added a class file **TravelClient.java** to access one of the REST API:

http://localhost:8050/travel/locations/country/US

**### Bonus points**

Docerised the application by adding **Dockerfile** in source code, and installing Docker in local environment

Please run below commands:

**./gradlew build**

**docker build -t case-0.1.0 .**

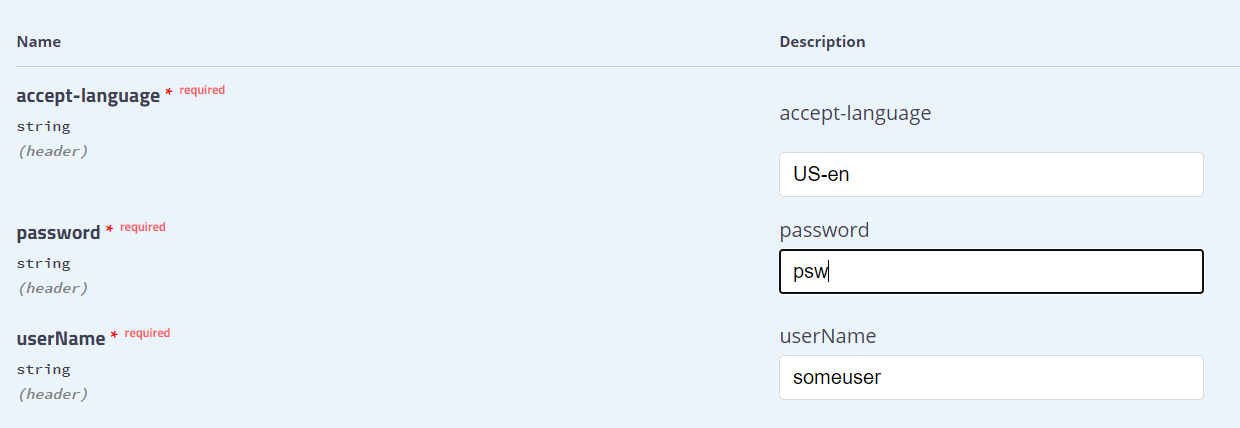
**docker run -i -p 8050:8080 case-0.1.0**

**Added Swagger UI support on below URL:**

<http://localhost:8050/swagger-ui.html#>

1. Please use below data in swagger to call REST API:

**curl -X GET "http://localhost:8050/travel/locations" -H "accept: \*/\*" -H "accept-language: US-en" -H "password: psw" -H "userName: someuser"**



1. Please use below data in swagger to call REST API:

**curl -X GET "http://localhost:8050/travel/locations/country/US" -H "accept: \*/\*" -H "password: psw" -H "userName: someuser"**

