**Steps to run the Basic API with Authentication**

1. Execute mysql script named “**98labs.sql**” to mysql server, this contains the database and table scripts.

2. Open a command prompt and navigate to the folder where you extract the 98labs folder and run a command “npm run dev” to start the project.

3. To test the apis first generate a token using a sample user account. You can use Postman/Fiddler or other means to call an APIs. Please see below:

Request:

url: <http://localhost:3000/users/authenticate>

method: POST

body:

{

"username": "test",

"password": "test"

}

Response:

{

"id": 1,

"username": "test",

"firstName": "Test",

"lastName": "User",

"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOjEsImlhdCI6MTU5NzM4MjY3MSwiZXhwIjoxNTk3OTg3NDcxfQ.NvOAt\_HGqPJz\_x72olRWMry7Q91k5ZWdnZkS20xvecw"

}

4. To test the token, copy the token and use it as Authrorization Bearer Token. Please see sample below:

Request:

url: <http://localhost:3000/users>

method: GET

Authorization: Bearer **<Token>**

Response:

[

{

"id": 1,

"username": "test",

"firstName": "Test",

"lastName": "User"

}

]

5. To test for customers, copy the token and use it as Authrorization Bearer Token. Please see sample below:

**Get all with paging and sorting.**

Request:

url: <http://localhost:3000/customers?sortby=firstname&sortdir=desc>

method: GET

Authorization: Bearer **<Token>**

Response:

{

"totalItems": 0,

"list": [],

"totalPages": 0,

"currentPage": 0

}

**Insert new customer.**

Request:

url: <http://localhost:3000/customers/>

method: PUT

Authorization: Bearer **<Token>**

body:

{

"firstname": "Test",

"lastname": "Test",

"date\_of\_birth": "12/15/2020",

"is\_active": 1

}

Response:

{

"id": null,

"firstname": "Test",

"lastname": "Test",

"date\_of\_birth": "2020-12-14T16:00:00.000Z",

"is\_active": true

}

**Update existing customer.**

Request:

url: <http://localhost:3000/customers/1>

method: PATCH

Authorization: Bearer **<Token>**

body:

{

"firstname": "Jose",

"lastname": "Riza;",

"date\_of\_birth": "12/26/2020",

"is\_active": 1

}

Response:

{

"message": "Customer was updated successfully."

}

**Delete a customer.**

Request:

url: <http://localhost:3000/customers/1>

method: DELETE

Authorization: Bearer **<Token>**

Response:

{

"message": "Customer was deleted successfully!"

}