LOGEALA CARDS BACKEND

API DOCUMENTATION

*Prepared by: Cheruiyot Kelvin Collins*

*Date:15/06/2023*

***Introduction.***

This document explains all available APIs on Cards Backend project, the project has been build using Java spring boot framework and MySQL db., for build maven has been used. It has two user roles (ADMIN and MEMBER). Admin has been granted access to register, log in, view all users, delete a user and view all cards while Member has only rights to register, log in, create a card (Card name being mandatory while color and description being optional, if color is provided is should start with # and should consist of a 6 digit alphanumeric characters).

Members are also able to view, edit and delete only the cards they created while admin can view list of all cards, all users and are able to edit and delete a user.

For Authentication JWT has been used and once a user has been authorized, bearer token will be generated and this should be send in the header of every request.

**APIs endpoints Table**

|  |  |  |  |
| --- | --- | --- | --- |
| HTTP VERB | **URL** | **DESCRIPTION** | **AUTHORITIES** |
| POST | <http://localhost:8080/api/auth/signup> | New user creation | MEMBER,ADMIN |
| POST | <http://localhost:8080/api/auth/signin> | Log in a user | MEMBER,ADMIN |
| GET | <http://localhost:8080/api/auth/list/all> | List all users | ADMIN |
| GET | [http://localhost:8080/api/auth/deleteById/{id}](http://localhost:8080/api/auth/deleteById/%7Bid%7D) | Delete a user | ADMIN |
| POST | <http://localhost:8080/api/card/add> | Create a card | MEMBER |
| PUT | <http://localhost:8080/api/card/update> | Update a card | MEMBER |
| GET | [http://localhost:8080/api/card/find HYPERLINK "http://localhost:8080/api/card/findById/%7Bid%7D"ById/{id}](http://localhost:8080/api/card/findById/%7Bid%7D) | Find card by id | MEMBER |
| DELETE | [http://localhost:8080/api/card/deleteById/{id}](http://localhost:8080/api/card/deleteById/%7Bid%7D) | Delete card by id | MEMBER |
| GET | <http://localhost:8080/api/card/list/all> | List all cards | ADMIN |
| GET | [http://localhost:8080/api/card/filterByName{name}](http://localhost:8080/api/card/filterByName%7Bname%7D) | Filter member’s card by name | MEMBER |
| GET | <http://localhost:8080/api/card/> filterByStatus/{status} | Filter member’s card by status | MEMBER |
| GET | <http://localhost:8080/api/card/> filterByColor/{color} | Filter member’s card by status | MEMBER |
| GET | <http://localhost:8080/api/card/> filterByCreationDate/{creationDate} | Filter member’s card by creation date | MEMBER |
| GET | <http://localhost:8080/api/card/listAllMemberCards> | List all members cards | MEMBER |

**SQL Scripts.**

Below are SQL Script to be run before and after starting the application

Before starting the application run below SQL script to create the database

***create database logicea\_db;***

**Starting the application**

Please make sure to change DB user credentials to the one you are using in application.properties file under resource folder then to start the application please run the following maven command in cmd while inside project’s root directory.

***mvn spring-boot:run***

after the application has started successfully hibernate DLL will auto generate all the tables, run below SQL scripts to add two roles (ADMIN, MEMBER) and create two users one with ADMIN and one with MEMBER privileges.

***INSERT INTO `logicea\_db`.`roles` (`id`, `name`) VALUES ('1', 'MEMBER');***

***INSERT INTO `logicea\_db`.`roles` (`id`, `name`) VALUES ('2', 'ADMIN');***

**Adding Users**

user below payload in post man to add two users:

endpoint: <http://localhost:8080/api/auth/signup>

**Admin**

*{*

*"name":"Administrator",*

*"username":"admin",*

*"email":"kelvincollins@gmail.com",*

*"roleId":2,*

*"password":"admin"*

*}*

**Member**

***{***

***"name":"Member",***

***"username":"member",***

***"email":"kelvincollins1@gmail.com",***

***"roleId":1,***

***"password":"member"***

***}***

**Signing in**

End point(POST): <http://localhost:8080/api/auth/signin>

admin

***{***

***"usernameOrEmail":"admin",***

***"password":"admin"***

***}***

member

***{***

***"usernameOrEmail":"member",***

***"password":"member"***

***}***

when you successfully sign in you will get a bearer token of this format:

***{***

***"accessToken": "eyJhbGciOiJIUzM4NCJ9.eyJzdWIiOiJrZWx2aW5jb2xsaW5zQGdtYWlsLmNvbSIsImlhdCI6MTY4NjgzMTMwNSwiZXhwIjoxNjg3NDM2MTA1fQ.glAiojgPcl5uz\_KIOmBKYk\_p657LoB7v9U2QY\_CeOs5uur9kT3g11RtjmPaFSGT\_",***

***"tokenType": "Bearer"***

***}***

To access the rest of the endpoints bearer token needs to be included in every request

**Adding Card**

Endpoint(POST): <http://localhost:8080/api/card/add>

Request

***{***

***"name":"Duncan Kibet",***

***"description":"Go Shopping",***

***"color":"#white"***

***}***

Response

**{**

**"status": "ACCEPTED",**

**"description": "Card Created Successfully"**

**}**

**Finding Card By Id**

End Point(GET): <http://localhost:8080/api/card/findById/1>

Response

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

**List Cards For A Particular Member**

End Point(GET): <http://localhost:8080/api/card/list/all/for/member>

response

***[***

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

***]***

**List All Cards (Admin)**

End Point(GET): <http://localhost:8080/api/card/list/all>

Response

***[***

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

***]***

**Member Delete Card By ID**

End Point(DELETE): <http://localhost:8080/api/card/deleteById/1>

Response

**Filter By Name**

End Point (GET): <http://localhost:8080/api/card/filterByCardName/Duncan> Kibet

Response

***[***

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

***]***

**Filter By Status**

End Point (GET): <http://localhost:8080/api/card/filterByStatus/To> Do

Response

***[***

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

***]***

**Filter By Color**

End Point (GET): <http://localhost:8080/api/card/filterByColor/white>

Response:

***[***

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

***]***

**Filter By Creation Date**

End Point (GET): <http://localhost:8080/api/card/filterByCreationDate/2023-06-15>

Response:

***[***

***{***

***"id": 1,***

***"name": "Duncan Kibet",***

***"description": "Go Shopping",***

***"color": "#white",***

***"status": "To Do",***

***"creationDate": "2023-06-15"***

***}***

***]***