**API Testing on TMS (Training Management System)**

1. Log in to the service using different users…

Endpoint: http://localhost:8081/api/v2/login

Method: POST

Request Body: Send **email (admin@gmail.com)** and **password (admin)** in **JSON** format.

1. Test the login endpoint for successful login [**using Admin**]
2. Test the login endpoint for the Not found error. Ex: User is not found associated with this email
3. Test email and password validation to login to the web service.
4. Users’ registration - **new user**

Endpoint: http://localhost:8081/api/user/registerUser

Method: POST

Request Body: ModelAttribute = {email, firstName, lastName, gender, password, phoneNo, profilePicture,role}. Have to select form-data in body in postman.

1. Test the user’s registration for the provided data.
2. Test the failed for existing users.
3. Get all user’s data

Endpoint: http://localhost:8081/api/user/get-all-users

Method: GET

Request Body: As usual pass the access token in authentication.

1. Test all the user’s data.

JSON responses:

[

    {

        "id": 1,

        "fullName": "first admin",

        "email": "admin@gmail.com",

        "phoneNo": null,

        "role": "ADMIN",

        "image": null

    },

    {

        "id": 2,

        "fullName": "first trainee",

        "email": "a@gmail.com",

        "phoneNo": "92392321",

        "role": "TRAINEE",

        "image": "created-user.png"

    },

    {

        "id": 3,

        "fullName": "first trainer",

        "email": "strainer@gmail.com",

        "phoneNo": "92392321",

        "role": "TRAINER",

        "image": "created-user.png"

    },

    {

        "id": 4,

        "fullName": "second trainee",

        "email": "b@gmail.com",

        "phoneNo": "213123213",

        "role": "TRAINEE",

        "image": "no-image"

    },

    {

        "id": 5,

        "fullName": "third trainee",

        "email": "c@gmail.com",

        "phoneNo": "213123213",

        "role": "TRAINEE",

        "image": "batchCreation.png"

    }

]

1. Create a new batch

Endpoint: http://localhost:8081/api/batch/create-batch

Method: POST

Request Body: RequestBody = {batch-name, start-time, end-time}

1. Test the API with some valid and invalid data.
2. Test this API with trainee login token to check the result.
3. Assign trainees to a batch

Endpoint: http://localhost:8081/api/classroom/assign-trainees-to-batch

Method: POST

Request: RequestBody = [{batch-name, trainee-email}]

1. Test this API with some valid trainee data.
2. Pass one trainee who is already assigned to another batch and test the result.
3. Create a Course

Endpoint: http://localhost:8081/api/course/create-course

Method: POST

Request Body: RequestBody = {courseName, courseDescription}

1. Test the API with some valid and invalid data.
2. Test this API with trainee login token to check the response.
3. Now create a Schedule

Endpoint: http://localhost:8081/api/schedules/create-schedule

Method: POST

RequestBody = {name, start-time, end-time, batch-name, course-name, trainer-name}

1. Test the API with some valid and invalid data.
2. Test this API with trainee/trainer login token to check the response.
3. Create notice

Endpoint: <http://localhost:8081/api/classroom/create-notice>

Method: POST

RequestBody = {classroom-id, notice}

1. Check classroom data from database and execute this API.
2. Create post in classroom

Endpoint: <http://localhost:8081/api/classroom/create-attachments>

Method: POST

RequestBody = {classroom-id, message, attached-file}

1. Test the API with attached file data and without this data.
2. Reply to the post in the classroom

Endpoint: <http://localhost:8081/api/classroom/create-reply>

RequestBody = {classroom-post-id, reply, trainee-email}

1. Test the API with some random data.