Makersharks Assessment

Name: G Rushivardhan

Email: rushivardhan18@gmail.com

Phone number: 9603366515

Task: Implement a user registration and user details fetch endpoints for a RESTful API using Spring Boot, following the requirement specified below.

API Endpoints:

1. /api/user/register [POST]
2. /api/user/fetch [GET]

Solution:

I created a spring boot project "Assignment" using "Spring Initializer" where I added all the required dependencies for my project and downloaded as a jar folder. Then In Assignment project I added all the packages and classes required for it. Then added the Postgres properties in application properties file, and all other requirements

Requirements:

- Java 17
- Spring Boot
- Postgres
- GitHub

Entity:

User Entity class shows all the required fields for the endpoint requirement.

Functionality:

As given in the assignment instructions, firstly the input is accepted in the form of JSON.

Email and Phone number are validated using the validated function to verify their format

Password is encrypted using **BCryptPasswordEncoder** and the encrypted password is stored in the Postgres.

The fetch endpoint will return the username, email and phone number and these are returned using a HashMap, here password is not returned due to security measures.

Setup Instructions:

1.Clone the Repository:

https://github.com/Rushi1820/MakersharksAssignment_Rushivardhan.git

2.Build the Project:

  Command: mvn clean install

3.Run the project

API Endpoints

1. Register User

Endpoint: /api/user/register

Method: POST

URL: http://localhost:8080/api/user/register

Description: Registers a new user with the provided details.

Request Body:

```
{
    "username": "rushi18",
    "email": "rushie@example.com",
    "password": "seurepassword",
    "phno": "9603366515"
}
```

Response:

If the details are new and not existing, then it will accept the details and return

```
{
    "message": "User registered successfully."
}
```

Else:

```
{
  "message": "Username, email, or phone number already exists."
}
```

2. Fetch User

Endpoint: /api/user/fetch

Method: GET

Description: Fetches details of a user by their username.

URL: http://localhost:8080/api/user/fetch?username=rushi18

Query Parameter: username

Response:

```
{
    "phno": 9603366515,
    "email": "rushie@example.com",
    "username": "rushi18"
}
```

Please find the attached postman collection for easy access of endpoints

---

Assignment Submitted to: Makersharsks Inc

Student Name: G Rushivardhan

Email: rushivardhan18@gmail.com

Phone number: 9603366515