1. Overview of The Project

As a part of designing a new electric system, a new user register and retrieving user by surname needs to be established. This exercise covers building the API services to support this new register, retrieving user details by surname and retrieving all data from Database.

1.2 Build With

This project is built with the following framework, build tool, server and database.
- Sping-boot framework. (3.2.4)
- Apache Maven (3.9.6)
- Tomcat
- MySQL

2. Getting Started

2.1 Prerequisites

Spring Boot 3.0 requires Java 17 as a minimum version. If you are currently using Java 8 or Java 11, you'll need to upgrade your JDK before you can develop Spring Boot 3.0 applications.

2.2 Installation

Java 17 for both OS (Mac & Window)
https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html

Maven Installation if required.
:brew install maven for Mac OS .
https://maven.apache.org/download.cgi for window OS.

IDE installations
IntelliJ for Mac: https://www.jetbrains.com/idea/download/?section=mac
IntelliJ for window: https://www.jetbrains.com/idea/download/?section=windows

Database installations
MySQL installation : https://downloads.mysql.com/archives/installer/
MySQL Workbench : https://dev.mysql.com/downloads/workbench/

Postman Installations
https://www.postman.com/downloads/

Github Link
https://github.com/ymphyow/userinfoapi

3. Usage of the Project

3.1 Create Schema and table in MySQL as below:

```
CREATE SCHEMA `electric_system` DEFAULT CHARACTER SET utf8mb3 ;

USE electric_system;
CREATE TABLE users (
id INT AUTO_INCREMENT PRIMARY KEY,
surname VARCHAR(255),
firstname VARCHAR(255),
email VARCHAR(255)
);
```

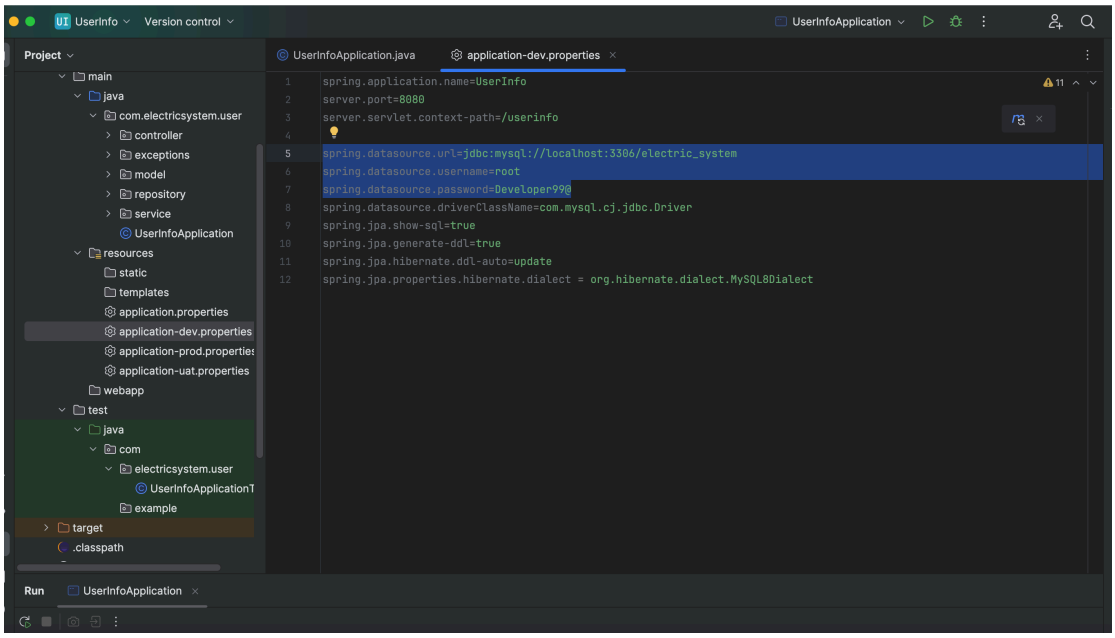
3.2 git clone https://github.com/ymphyow/userinfoapi.git

3.3 import the project into IntelliJ IDE.

3.3 Four properties files are located in the src/main/resources directory.

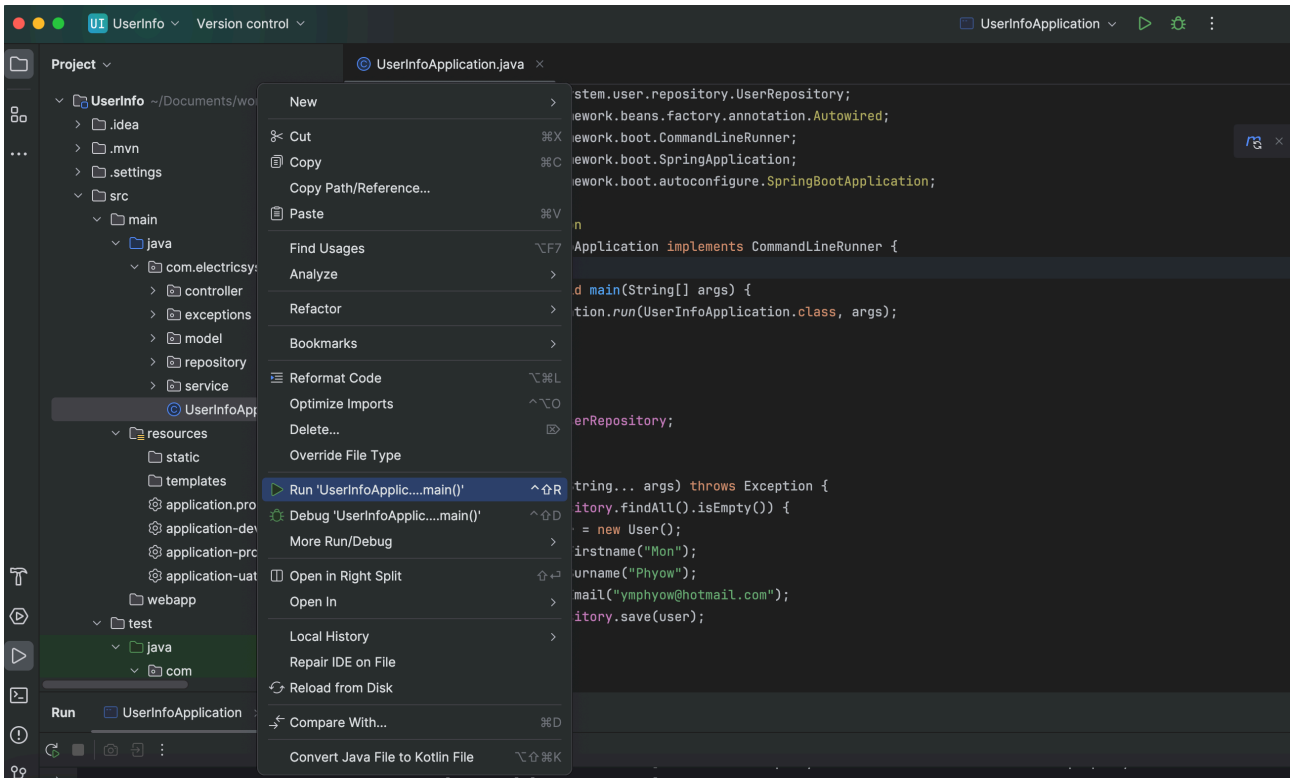
application.properties (Specify the environment in this file.)
application-dev.properties (Configure credentials for development environment)
application-uat.properties (Configure credentials for user Testing environment))
application-prod.properties (Configure credentials for Production environment))

- Open the properties file -> application-dev.properties
- Update the application.properties file in the src/main/resources directory with your database connection details.
- Ensure SDK version is 17 in project setting.



4. Running the Application

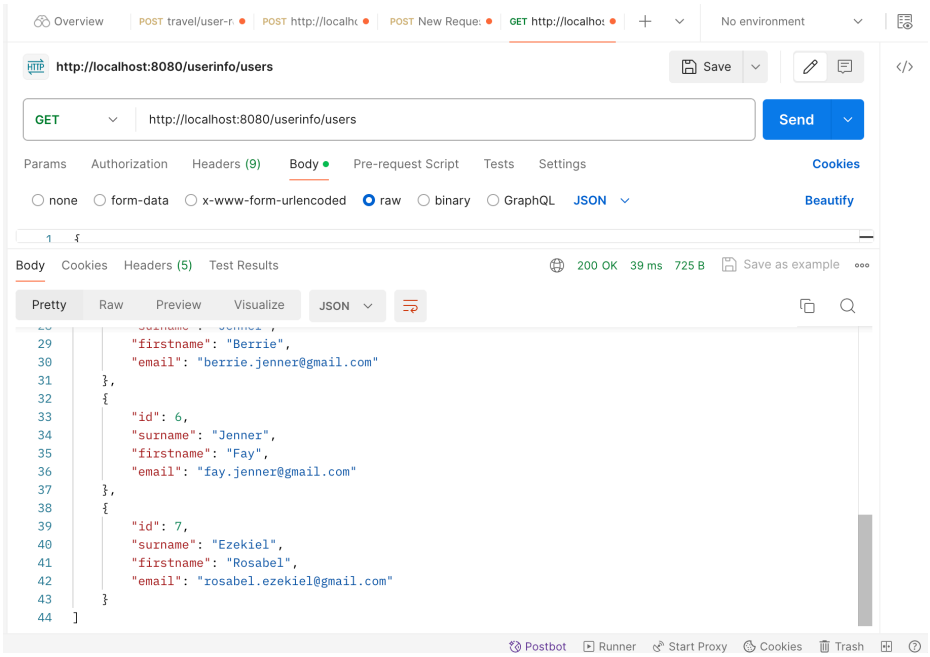
Ensure that you have a MySQL or PostgreSQL database set up and running.
Run the application using Maven/Tomcat. Default port is 8080.



5. Using the API

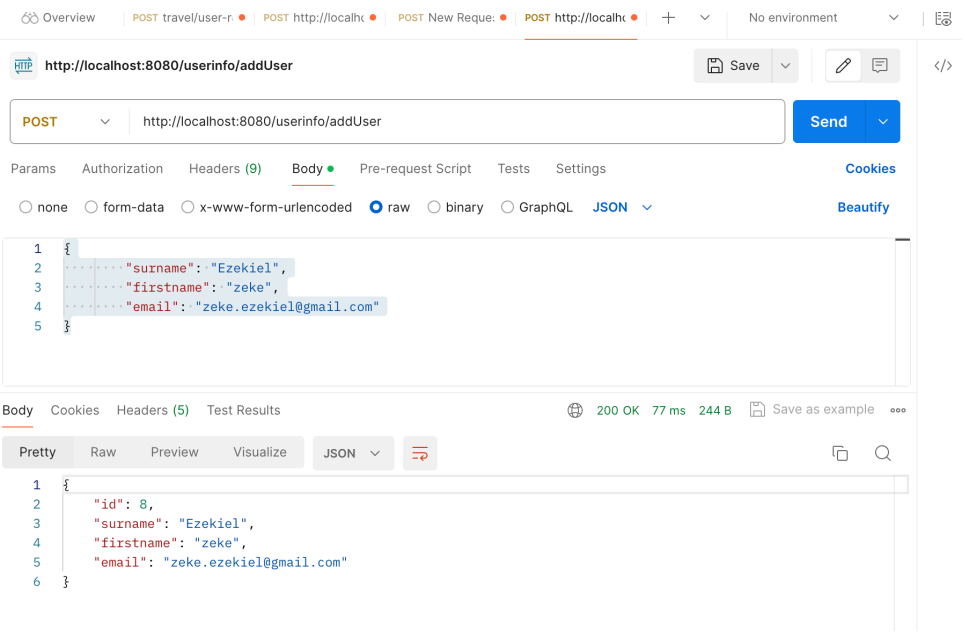
5.1 Testing in postman.

To retrieve all user info using GET method: http://localhost:8080/userinfo/users

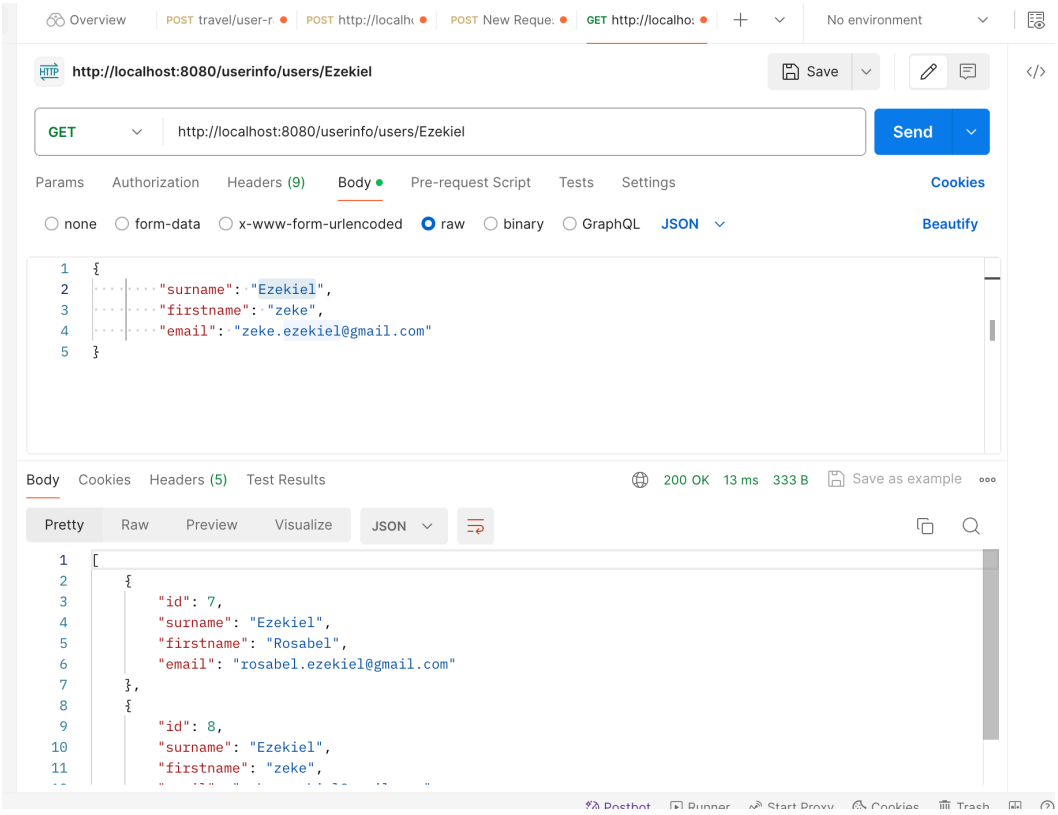


To add user info using POST method: http://localhost:8080/userinfo/addUser
See below JSON example object.

```
{
    "surname": "Ezekiel",
    "firstname": "zeke",
    "email": "zeke.ezekiel@gmail.com"
}
```



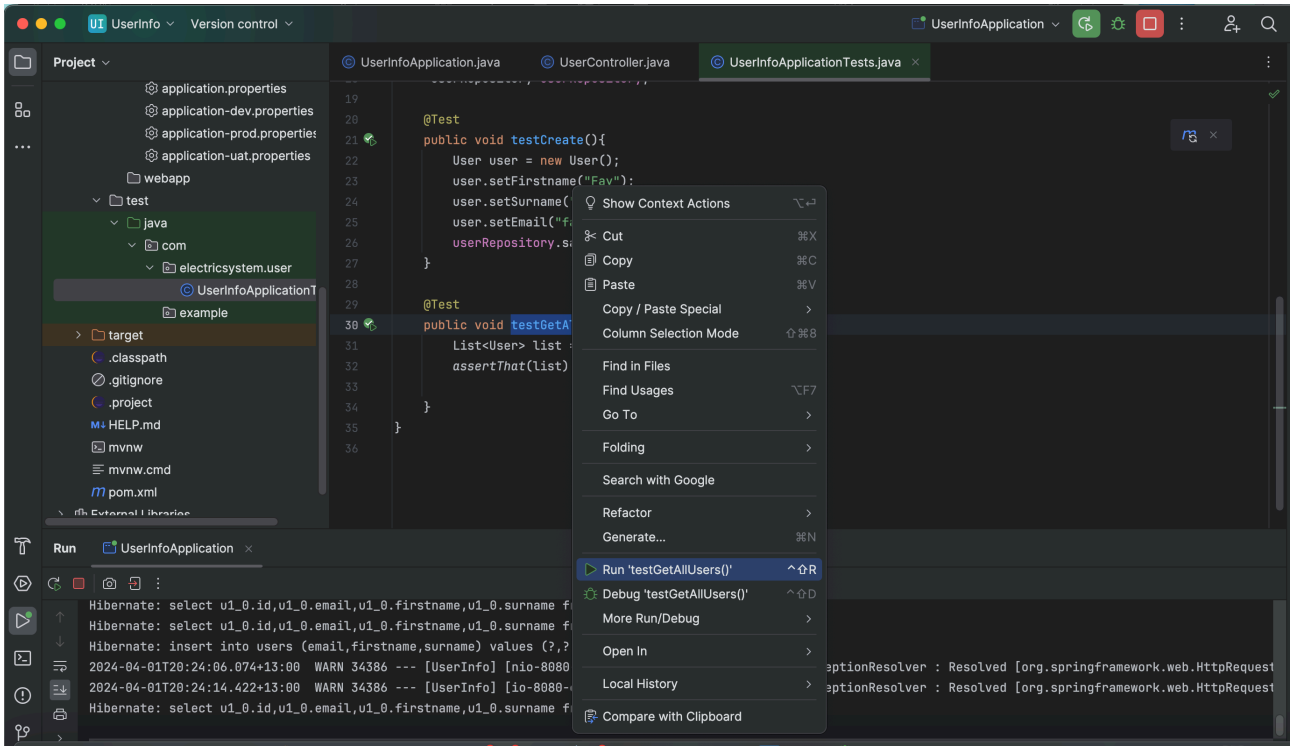
To retrieve user detail by surname using GET method.
Ex: http://localhost:8080/userinfo/users/Ezekiel

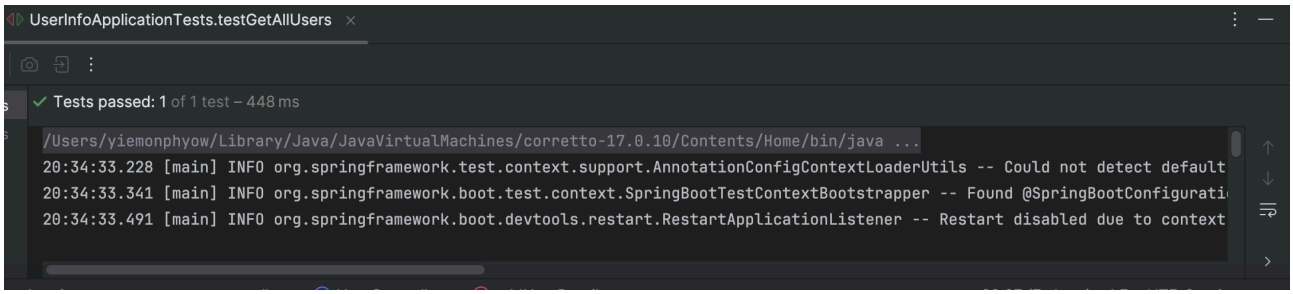


6. Run the Test Case

Run UserInfoApplicationTests file in the test/java/com/electricsystem/user directory.

Choose the test case and check the status as below:





Feel free to customize or give feedback this README file with more detailed instructions or additional information specific to this application.