Sri Lanka Institute of Information Technology

Electro Grid (EG) System Web Services

Project Report

IT3030 - Programming Applications and Frameworks 2021
Group Assignment

Group Number: 190

Group Members:

1. IT20098582 – Nissanka L.N.A.T.A.

1. Workload Distribution

Registration Number	Name with Initials	Web Service Allocated
IT20098582	Nissanka L.N.A.T.A.	Users Service • User Management. • User Role Management. • User Authentication and Authorization. Suppliers Service • Manage/ calculate unit prices by considering the start date, last date and the number units used

2. Version Controlling

- Public GitHub Repository Link: https://github.com/lnathilina/PAF-22.git

3. SE Methodology

The waterfall model was selected since the requirement set of the project was identified upfront and is non-frequently changing.

Moreover, to develop and maintain the code, mainly the object-oriented programming paradigm was used while dividing the classes according to MVC architecture.

4. Requirements

4.1.1. Functional Requirements

Functional requirements of each web service are shown in a tabular format along with the stakeholder type who requires it as follows.

Web Service	Functional Requirement	Type of the user
Supplier Service	Add/Remove/View/Update Supplier	Administrators Consumers

4.1.2. Non-functional Requirements

Availability

The web services should be available and accessible at any time, except during a system maintenance. A given service should be working fine independently even though another web service of the system is down or under maintenance except when the inter-service communication is required to fulfill a functional requirement.

Maintainability

Documentation of the APIs of the web services should be well-written and comprehensible by the users or the developers who are implementing client programs to communicate with the services and should mention the correct methods of troubleshooting, types of media consumed by the end points and the URI format.

Portability

Any type of client program should be able to call the API end points and request information from the web services regardless of the language or the technology stack used to develop the client programs.

4.1.3. Technical Requirements

Technical requirements for the development of the web services are as follows.

IDE:	Eclipse versions between 2023.3
Server:	Apache Tomcat Server Version 9
Java Development Kit:	JDK8
Java Runtime Environment:	JRE 1.8
Operating System:	Any OS that supports eclipse versions mentioned above
Libraries:	MySQL connector (JDBC)
Dependency Management:	Maven 3
Version Control:	Git and GitHub
Database Management:	MySQL Workbench/ WAMP Server/ XAMPP Server

Recommended Technical Requirements for implementing the web services of Electro Grid are as follows.

Server:	Apache Tomcat Server Version 9 or above (if hosted locally. Any

	cloud provided who supports war package hosting will be compatible as well.)
Java Runtime Environment:	JRE 1.8
Operating System:	Any OS that supports JDK 8/JRE 1.8 and Apache Tomcat Server 9
Database Server:	MySQL server/ PhpMyAdmin (If hosted locally) or Any cloud provider that supports MySQL database hosting

5. Individual Sections

5.1. Supplier Service

Supplier Service is responsible for managing all users who are directly interacting with all web services of system. It does maintain Service Development and Testing

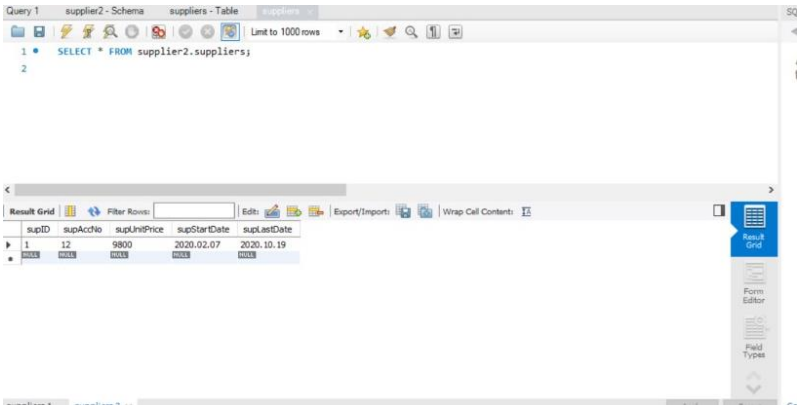
5.1.1. Service Development and Testing

- **Eclipse, JDK 8, JRE 1.8, Tomcat 9**
- **GitHub** repository to manage the version controlling and team collaboration.
- **Postman** to test the developed API endpoints to make sure they are working as expected.

6. Appendix

Appendix A: Supplier Service Diagrams

Figure A-1 Database design of the Funding Service



Database Design of the Funding Service

Appendix B: Testing and Results

B-1. POSTMAN Test results of User Service

Users endpoints

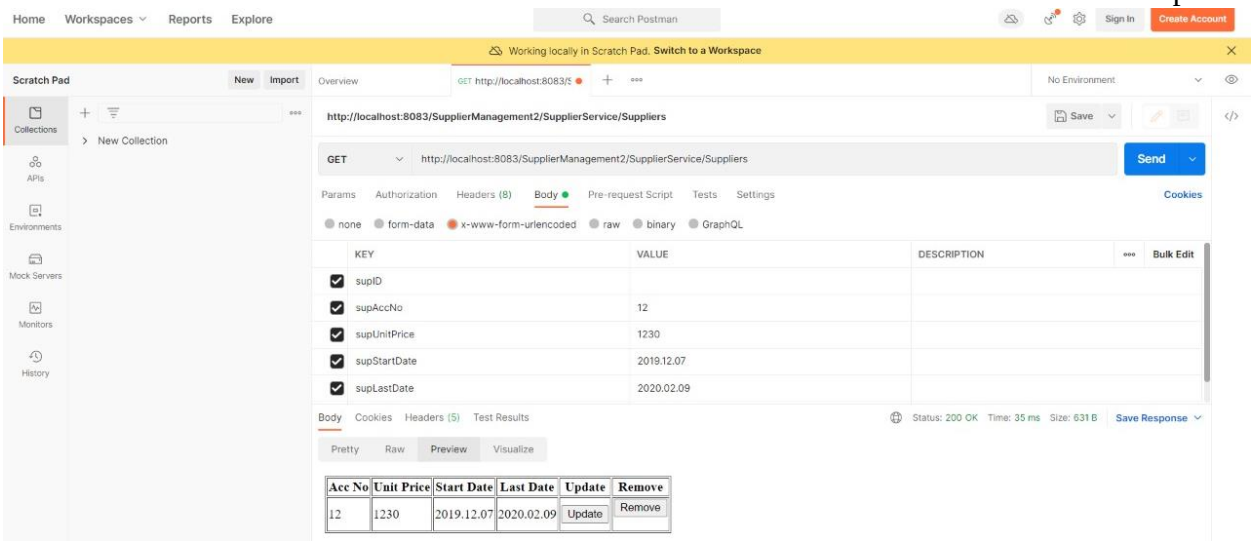


Figure B-1 Supplier service: getting summarized details

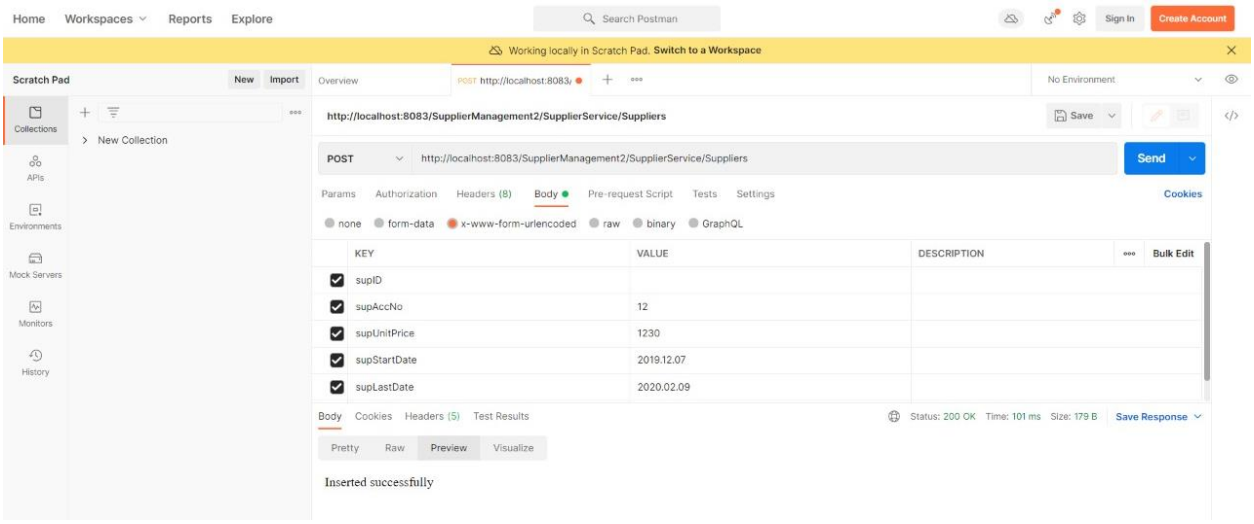


Figure B- 2 user service: authentication (login as an ADMIN). The response will contain the JWT token which needs to be included in every other request where authentication is needed.

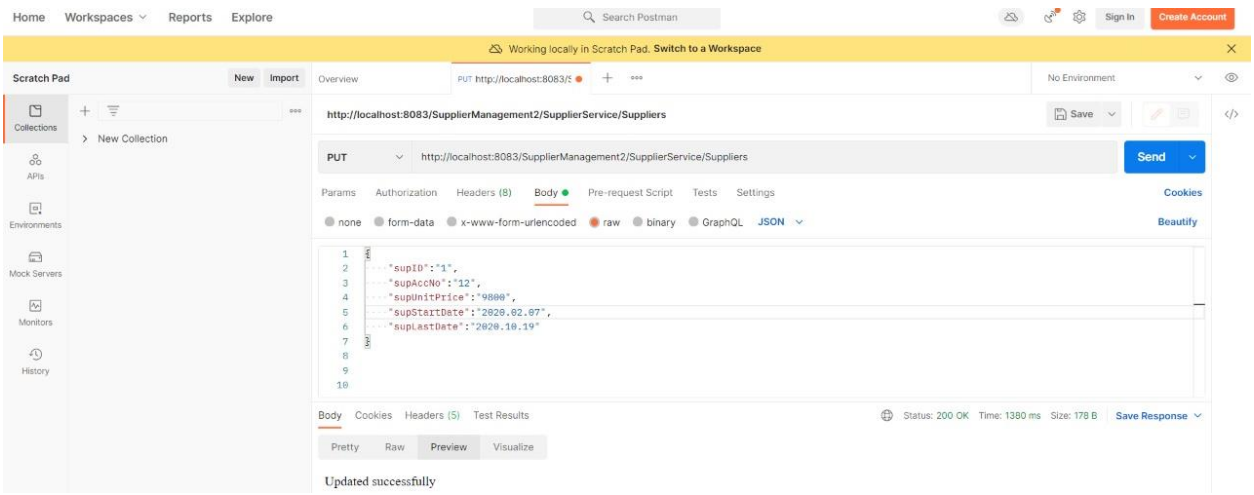


Figure B- 3 user service: resetting user password (any user can perform this task after login)

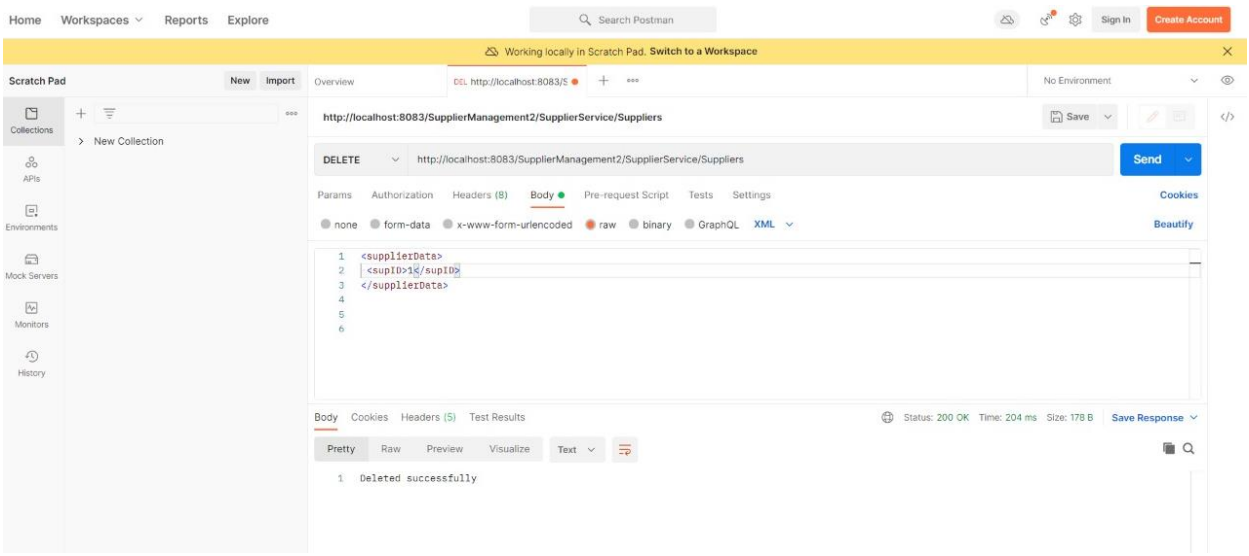


Figure B- 4 user service: deactivating a user account. (only administrators can do this)