



Sri Lanka Institute of Information Technology

## **Electro Grid (EG)**

### **Project Report**

IT3030 – Programming Applications and Frameworks 2022  
Batch No: Y3.S1.WE.DS.04

Group ID: **181**

Team Members:

1. IT20213008- Wijendra R.H.
2. IT20060008- Wickramaratne D.D.W.
3. IT20207854- De Silva M.
4. IT20225506- Thirimanne S.U.
5. IT20257286- Athukorala A.S.U.
6. IT20258580- Kanakaratne M.R

## Table of Contents

Project Scenario .....	3
Work Breakdown Chart.....	4
Version Controlling System.....	5
SE Methodology .....	5
Gantt chart .....	5
Onion Diagram (Stakeholder Analysis) .....	5
Requirements Analysis.....	6
Functional Requirements.....	6
Non-Functional Requirements.....	6
Tools and Technologies.....	6
Use Case Diagram (Requirements Modelling) .....	7
System Overview .....	8
ER Diagram.....	9
Individual Sections .....	10
IT20213008 - Wijendra R.H. : Billing Service.....	10
IT20060008- Wickramaratne D.D.W. : Schedule Service.....	14
IT20207854- De Silva M. : Power Generator Service .....	19
IT20225506- Thirimanne S.U. : New Connection Service .....	23
IT20257286- Athukorala A. S. U. : Branch Service .....	27
IT20258580- Kanakaratne M.R. : Customer Service .....	29
References .....	34
Appendix .....	34

## Project Scenario

Our Electro-grid system will handle the work from where electricity is generated in the powerhouses to providing new connections to our customers. Moreover, we will be handling the services below managing the electricity power given from the power houses to providing it accordingly to our customers. Now let us dive deep into the services to get a better understanding of the system.

## Identifying Services

### 1. Customer Service

Customer service manages all the things related to customers in Electro Grid System. Customer service has mainly two target stakeholders. They are registered customers and unregistered customers. In here system admin have got all rights to electro grid system. Also, admin can add, delete, update customers and he have access to view all details regarding customer service. And anyone can log on to the system as an unregistered customer and see all the updated details related to Electro Grid System. If he is willing to register, he can register to the system also. When a user becomes a registered customer, he can request customer service.

### 2. New Connection Service

New connection service provides customers to request new electricity connections to their domestic houses or industrial sites. One can request by filling in the details and clarifying whether it is domestic or industrial. Once completed our administrative staff will get in touch with them via email and contact methods and later update the status of the connection via our administrative.

### 3. Billing Service

Billing Service is to generate monthly bills to the registered customers by the admin of the Electro Grid Company. The admin has the right to generate, update, delete and read the bill data.

### 4. Power Generator Service

Power Generator Service manages the Power generators that provide electricity to ElectroGrid System. The power generators can be of different types such as coal-powered, wind-powered, fuel powered etc. The power generator company staff can log into the system and add the relevant details such as name, type and cost per unit of electricity generated. . They can also can update details such as unit cost as and when required, and also delete the details in the case of the power generator is shutting down. The admin of the system can view the newly added power generator details and validate the details.

### 5. Schedule Service

Schedule service provides for customers to new updates of the power outages are made through the power-cut schedule. Also change schedules and dates are notified instantly.

### 6. Branch Service

Branch details service provides all the details of branches, Customers can search for a branch detail by simply typing the branch name. An Admin is the person in charge of inserting, updating, and deleting relevant information. Admin can also see the added details. Customers can send a message to request clarification. Admin has access to those messages and can respond to them via email addresses.

## Work Breakdown Chart

Student ID	Name	Function	CRUD Operations
IT20213008	Wijendra R.H.	Billing Service	<ul style="list-style-type: none"> <li>• Read Bill Data</li> <li>• Insert Bill Data</li> <li>• Update Bill Data</li> <li>• Delete Bill Data</li> </ul>
IT20060008	Wickramaratne D.D.W.	ScheduleService	<ul style="list-style-type: none"> <li>• Read Schedule details</li> <li>• Insert Schedule details</li> <li>• Update Schedule details</li> <li>• Delete Schedule details</li> </ul>
IT20207854	De Silva M.	Power Generator Service	<ul style="list-style-type: none"> <li>• Read Power Generator Details</li> <li>• Insert New Power Generator</li> <li>• Update Power Generator Details</li> <li>• Delete Power Generator</li> <li>• User Login</li> </ul>
IT20225506	Thirimanne S.U.	New Connection Service	<ul style="list-style-type: none"> <li>• Read New Connection Entry</li> <li>• Insert New Connection Entry</li> <li>• Update New Connection Entry</li> <li>• Delete New Connection Entry</li> </ul>
IT20257286	Athukorala A.S.U.	Branch Service	<ul style="list-style-type: none"> <li>• Read Branch details</li> <li>• Insert Branch details</li> <li>• Update Branch details</li> <li>• Delete Branch details</li> </ul>
IT20258580	Kanakaratne M.R	Customer Service	<ul style="list-style-type: none"> <li>• Read Customer Details</li> <li>• Insert Customer Details</li> <li>• Update Customer Details</li> <li>• Delete Customer Details</li> </ul>

## Version Controlling System

GIT is used for version controlling.

Github Repository link - <https://github.com/SLIIT-Y3S1-PAF-Project/ElectroGrid-Project.git>

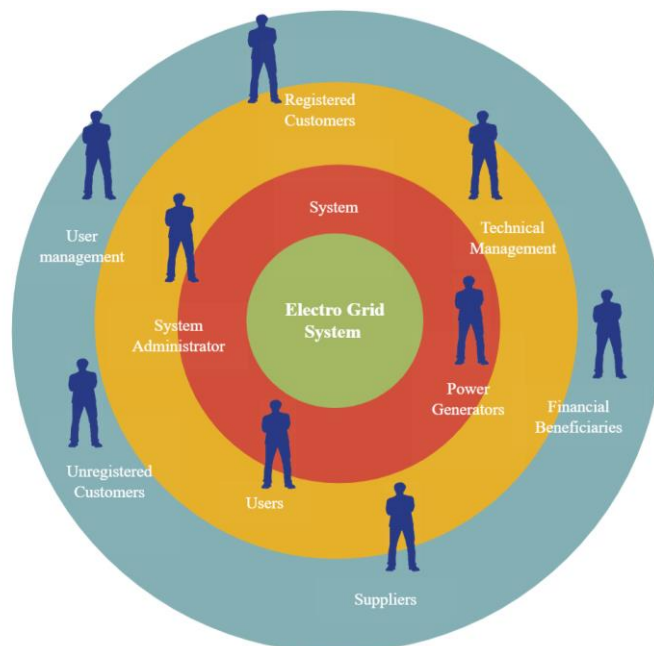
## SE Methodology

As a team we chose Agile development methodology as an iterative approach to project management and software development that helped us deliver the best product. We chose this methodology instead of any other because rather than betting everything on a one big launch, we could work in small, but consumable, increments as agile developers because it suits us best in our workflow. Furthermore, we chose this methodology because we can respond to changes easily under feedback and integrate it into one complete working product with minimum effort.

## Gantt chart

	WEEK 01	WEEK 02	WEEK 03	WEEK 04	WEEK 05	WEEK 06	WEEK 07	WEEK 08	WEEK 09	WEEK 10
REQUIREMENTS GATHERING										
DESIGN										
FIND SUPPORT RESOURCES AND SELF-STUDY										
ASSIGN INDIVIDUAL FUNCTIONS										
CODING										
TESTING										
PREPARE REPORT										
FINALIZE PROJECT										

## Onion Diagram (Stakeholder Analysis)



# Requirements Analysis

## Functional Requirements

### 1. Customer Service

Admin should be able to add, update, view, and delete customer details. Any customer should be able to log into the system and see all the updated information.

### 2. New Connection Service

Customers can use this service to request new electricity connections for their domestic houses or industrial sites. The service's status should be updated by the admin.

### 3. Billing Service

Billing Service is responsible for generating monthly bills for registered customers. Admin is allowed to add details to monthly bills, as well as update, delete, and generate them.

### 4. Power Generator Service

There are several types of power generators, such as coal-powered, wind-powered, fuel-powered, etc. Power generators should be able to enter relevant information such as name, type, and cost per unit of electricity generated. View those, update the unit cost details, and delete the details if the power generator is shutting down. The admin has the authority to validate the information.

### 5. Schedule Service

Schedule service is responsible for notifying the change schedules and dates.

This service should provide the latest updates on the power outages to the customers.

### 6. Branch Service

Customers should have access to see the details of branches. Admin is allowed to insert branch details update those and delete those.

Any clarification should be asked through the message box and the admin should have access to view those.

## Non-Functional Requirements

### 1. Availability

The system must be accessible 24x7

### 2. Performances

Response time should be high

### 3. Security

Users should have access to their relevant functions according to their authentication level.

### 4. Reliability

## Tools and Technologies

IDE – Eclipse

Frameworks – Java-JAX-RS(Jersey)

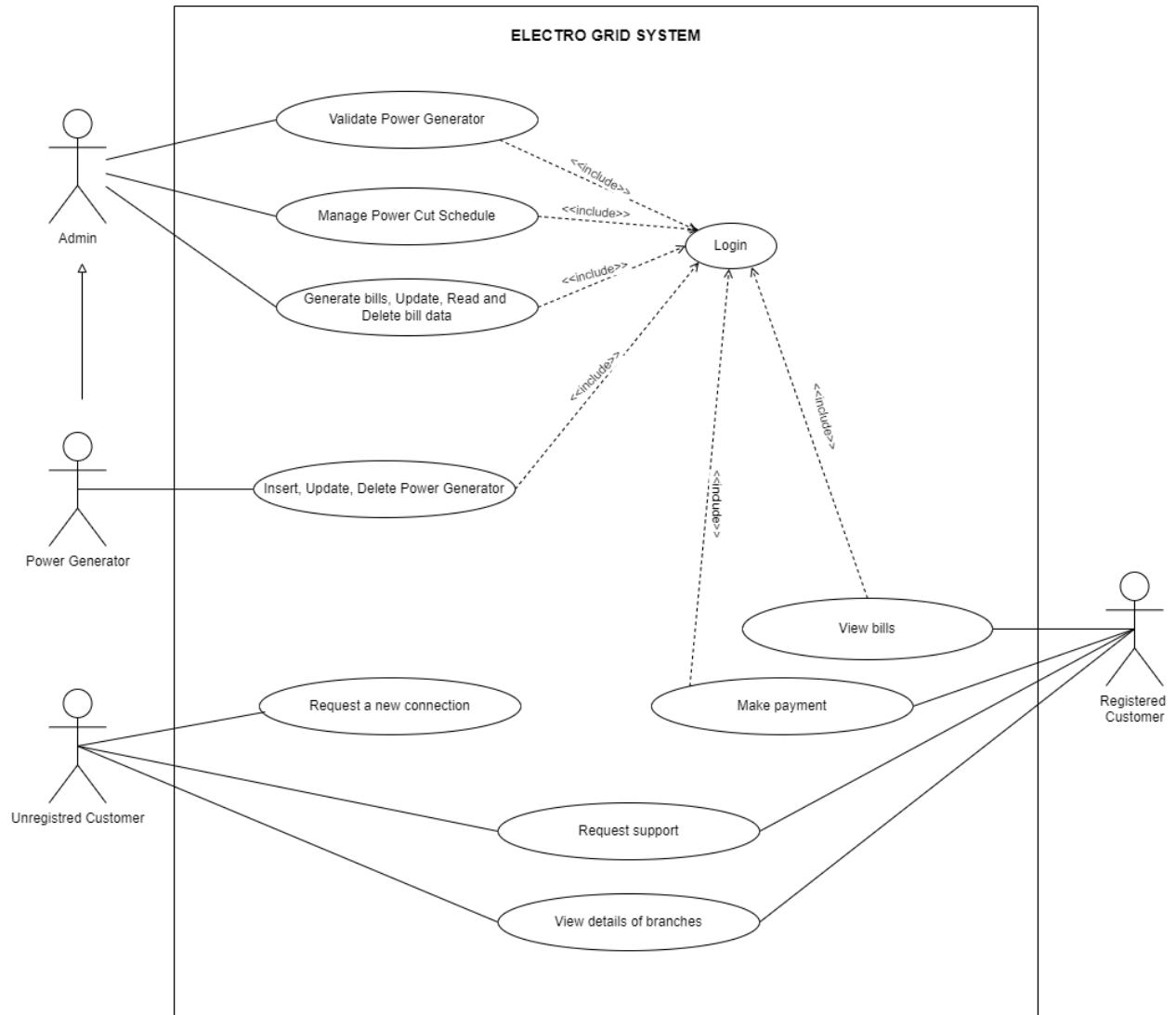
Server - Tomcat version 9.0

Database - MySQL

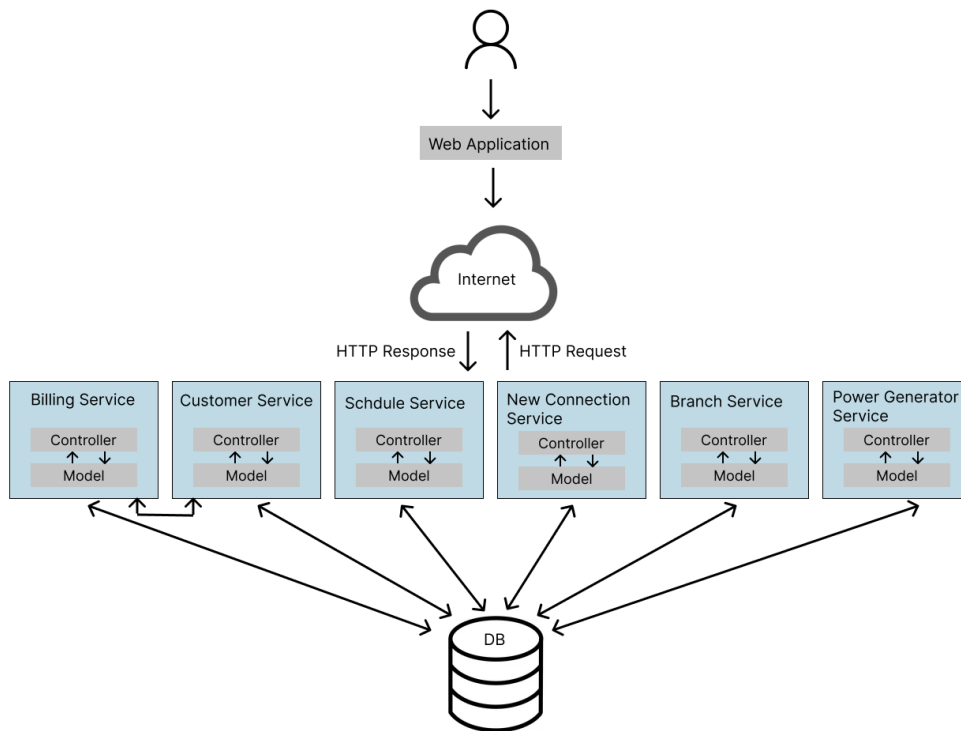
Testing – Postman

Other tools – XAMPP, Maven

## Use Case Diagram (Requirements Modelling)

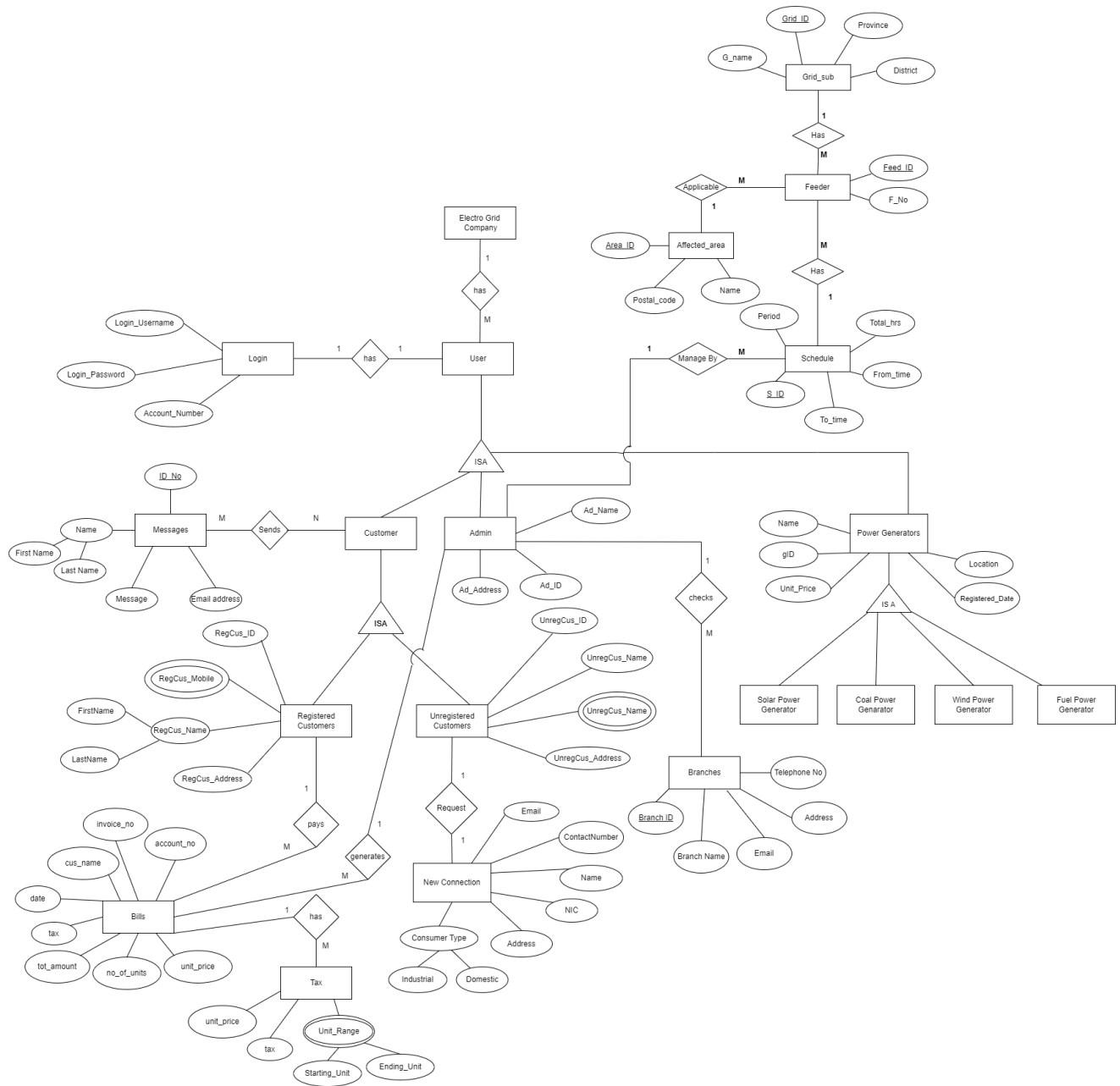


## System Overview





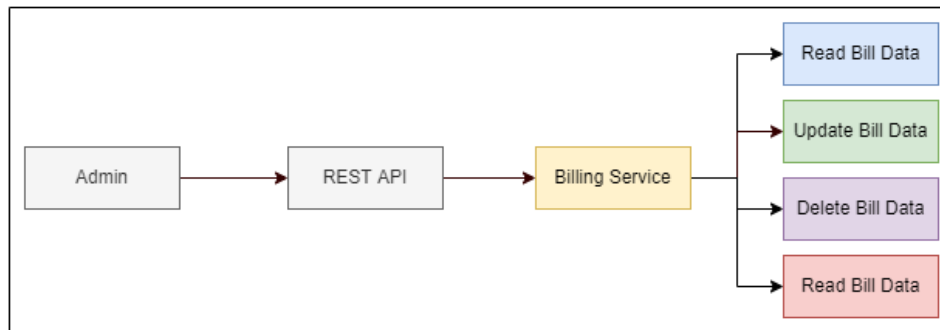
## ER Diagram



## Individual Sections

IT20213008 - Wijendra R.H. : Billing Service

### Service Design



### CRUD

GET – View all the Bill Data	POST – Insert Bill Data																																																								
<p>URL: http://localhost:8095/BillingService/billingService/bills/readBills</p> <p>Resource: Billing API</p> <p>Request: GET BillingService/billingService/bills/readBills</p> <p>Media: TEXT_HTML</p> <p>Response:</p> <table><tr><th>Invoice No</th><th>Date</th><th>Customer Name</th><th>Account No</th><th>Units</th><th>Unit Price</th><th>Tax</th><th>Total Amount</th></tr><tr><td>1</td><td>2022-04-01</td><td>H.L.Fernando</td><td>236589123</td><td>152</td><td>32.0</td><td>480.0</td><td>5344.0</td></tr><tr><td>4</td><td>2022-02-01</td><td>P.K.Oman</td><td>985463255</td><td>32</td><td>5.0</td><td>60.0</td><td>220.0</td></tr><tr><td>5</td><td>2022-02-01</td><td>Jhon</td><td>978954628</td><td>100</td><td>28.0</td><td>480.0</td><td>3280.0</td></tr><tr><td>6</td><td>2022-03-05</td><td>N Somapala</td><td>125478412</td><td>100</td><td>28.0</td><td>480.0</td><td>3280.0</td></tr><tr><td>7</td><td>2022-02-01</td><td>Nimali Kumara</td><td>974812628</td><td>60</td><td>50.0</td><td>60.0</td><td>90.0</td></tr><tr><td>9</td><td>2022-04-24</td><td>Kumara Y.K.</td><td>745814782</td><td>134</td><td>32.0</td><td>480.0</td><td>4768.0</td></tr></table> <p>"Error while reading Bill Data"</p>	Invoice No	Date	Customer Name	Account No	Units	Unit Price	Tax	Total Amount	1	2022-04-01	H.L.Fernando	236589123	152	32.0	480.0	5344.0	4	2022-02-01	P.K.Oman	985463255	32	5.0	60.0	220.0	5	2022-02-01	Jhon	978954628	100	28.0	480.0	3280.0	6	2022-03-05	N Somapala	125478412	100	28.0	480.0	3280.0	7	2022-02-01	Nimali Kumara	974812628	60	50.0	60.0	90.0	9	2022-04-24	Kumara Y.K.	745814782	134	32.0	480.0	4768.0	<p>URL: http://localhost:8095/BillingService/billingService/bills/insertBills</p> <p>Resource: Billing API</p> <p>Request: POST BillingService/billingService/bills/insertBills</p> <p>Media: APPLICATION_FORM_URLENCODED</p> <p>Response: " Inserted successfully " "Error while inserting Bill Data"</p>
Invoice No	Date	Customer Name	Account No	Units	Unit Price	Tax	Total Amount																																																		
1	2022-04-01	H.L.Fernando	236589123	152	32.0	480.0	5344.0																																																		
4	2022-02-01	P.K.Oman	985463255	32	5.0	60.0	220.0																																																		
5	2022-02-01	Jhon	978954628	100	28.0	480.0	3280.0																																																		
6	2022-03-05	N Somapala	125478412	100	28.0	480.0	3280.0																																																		
7	2022-02-01	Nimali Kumara	974812628	60	50.0	60.0	90.0																																																		
9	2022-04-24	Kumara Y.K.	745814782	134	32.0	480.0	4768.0																																																		
PUT – Update Bill Data	DELETE – Delete Bill Data																																																								
<p>URL: http://localhost:8095/BillingService/billingService/bills/updateBills</p> <p>Resource: Billing API</p> <p>Request: PUT</p>	<p>URL: http://localhost:8095/BillingService/billingService/bills/deleteBills</p> <p>Resource: Billing API</p> <p>Request: DELETE</p>																																																								

BillingService/billingService/bills/updateBills  Media: APPLICATION_JSON  Response: " Updated successfully " "Error while updating Bill Data "	BillingService/billingService/bills/deleteBills  Media: APPLICATION_XML  Response: " Deleted successfully " " Error while deleting Bill Data "
--	--

## Testing

Test ID	Test Description/ Test Steps	Test Input(s)	Expected Output(s)	Actual Output(s)	Result (Pass/Fail)
01	View all Bill Data	URL for the API Send Get request.	Display all Bill Data.	Display all Bill Data.	pass
02	Add Bill Data	Attributes of Bill Data	"Inserted successfully"	"Inserted successfully"	pass
03	Update Bill Data	Attributes to be updated along with the Invoice No	"Updated successfully"	"Updated successfully"	pass
04	Delete Bill Data	Given Invoice No of the bill to be deleted.	"Deleted successfully"	"Deleted successfully"	pass

## GET

GET http://localhost:8095/BillingService/billingService/bills/readBills

Params Authorization Headers (6) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL HTML

1

Body Cookies Headers (5) Test Results Status: 200 OK Time: 14 ms

Pretty Raw Preview Visualize

Invoice No	Date	Customer Name	Account No	Units	Unit Price	Tax	Total Amount
1	2022-04-01	H.L.Fernando	236589123	152	32.0	480.0	5344.0
4	2022-02-01	P.K.Oman	985463255	32	5.0	60.0	220.0
5	2022-02-01	Jhon	978954628	100	28.0	480.0	3280.0
6	2022-03-05	N Somapala	125478412	100	28.0	480.0	3280.0
7	2022-02-01	Nimali Kumara	974812628	60	50.0	60.0	90.0
9	2022-04-24	Kumara Y.K.	745814782	134	32.0	480.0	4768.0

## POST

POST http://localhost:8095/BillingService/billingService/bills/insertBills ...

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

cusName	Kumara Y.K.
accNo	74581478
noOfUnits	134
unitPrice	32
tax	480.00
totalAmount	4768.00

Body Cookies Headers (5) Test Results Status: 200 OK Time: 175 ms

Pretty Raw Preview Visualize Text

1 Inserted successfully

## PUT

PUT http://localhost:8095/BillingService/billingService/bills/updateBills ...

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   ... "invoiceNo": "7",
3   ... "date": "2022-02-01",
4   ... "cusName": "Nimali Kumara",
5   ... "accNo": "974812628",
6   ... "noOfUnits": "60",
7   ... "unitPrice": "60.00",
8   ... "tax": "60.00",
9   ... "totalAmount": "90.00"
10 }
```

Body Cookies Headers (5) Test Results Status: 200 OK

Pretty Raw Preview Visualize Text

1 Updated successfully

## DELETE

DELETE http://localhost:8095/BillingService/billingService/bills/deleteBills ...

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL XML

```
1 <billData>
2 | <invoiceNo>8</invoiceNo>
3 </billData>
```

Body Cookies Headers (5) Test Results Status: 200 OK Time: 4

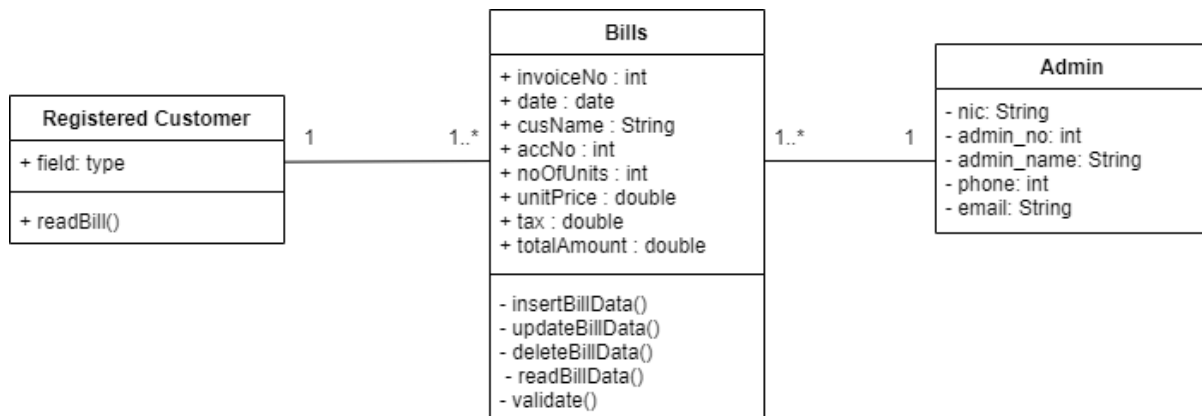
Pretty Raw Preview Visualize Text

1 Deleted successfully

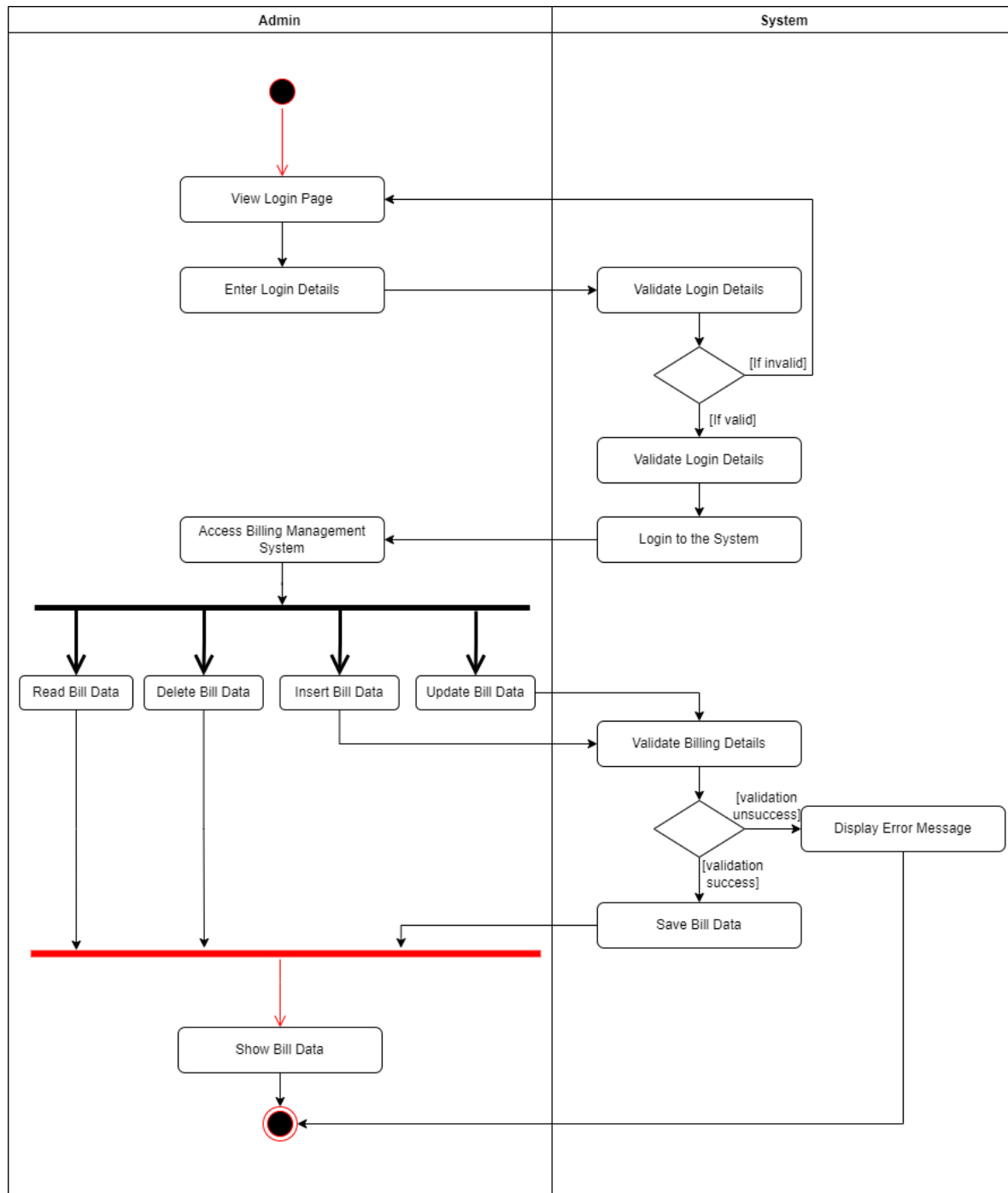
## Assumptions and other details

- Only the admin performs the CRUD operations.

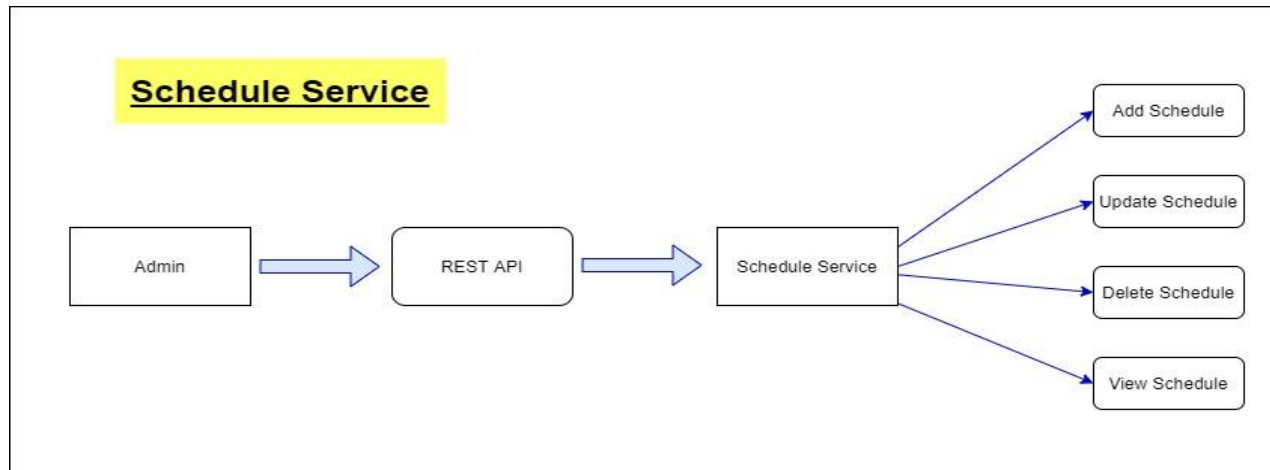
## Class Diagram



## Activity Diagram



## Service Design



## CRUD

GET - View all Power-cut Schedule	POST – Insert new Schedule
<p><b>URL:</b> http://localhost:8081/PAF-Projec/ScheduleService/Schedules</p> <p><b>Resource:</b> Schedule Management API</p> <p><b>Request:</b> GET PAF-Projec/ScheduleService/Schedules</p> <p><b>Media:</b> TEXT_HTML</p> <p><b>Response:</b> {   "sID": "7",   "sPeriod": "12toMidnight",   "sTotHrs": "3Hours",   "sFromTime": "7pm",   "sToTime": "10pm",   "sArea": "Narahepita",   "sSub": "Colombo A zone",   "sProvince": "Western" }</p> <p>"Error while reading the user."</p>	<p><b>URL:</b> http://localhost:8081/PAF-Projec/ScheduleService/Schedules</p> <p><b>Resource:</b> Schedule Management API</p> <p><b>Request:</b> POST PAF-Projec/ScheduleService/Schedules</p> <p><b>Media:</b> APPLICATION_FORM_URLENCODED</p> <p><b>Response:</b> "Inserted successfully" "Error while inserting the User."</p>

PUT – Update Scheduled details	DELETE – Delete Schedule details
URL: http://localhost:8081/PAF-Projec/ScheduleService/Schedules  User Resource: User Management API Request: PUT PAF-Projec/ScheduleService/Schedules  Media: APPLICATION_JSON  Response: "Updated successfully" "Error while updating the User."	URL: http://localhost:8081/PAF-Projec/ScheduleService/Schedules  Resource : User Management API  Request: DELETE PAF-Projec/ScheduleService/Schedules  Media: APPLICATION_XML  Response: "Deleted successfully" "Error while deleting the User."

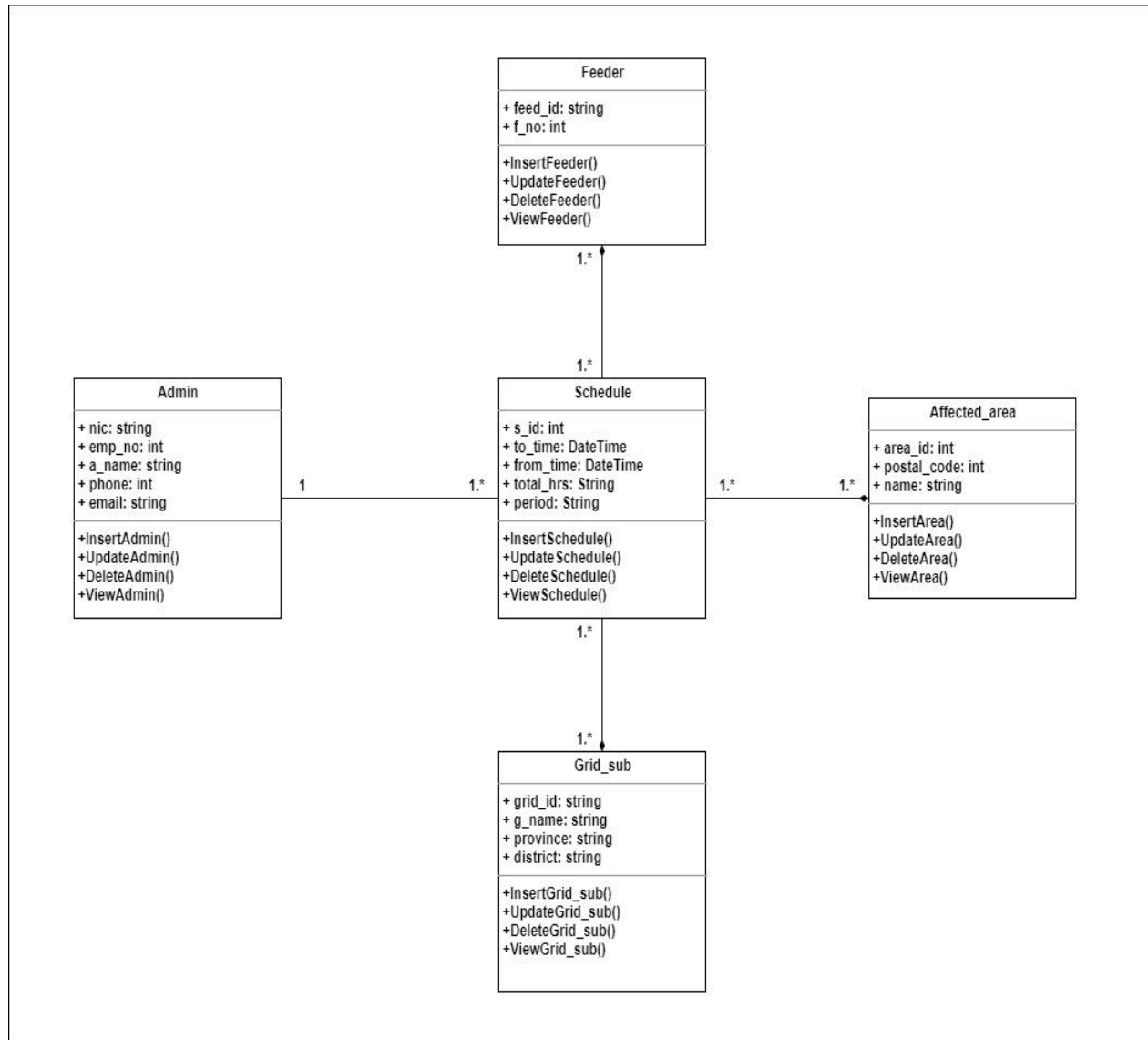
## **Testing**

Test ID	Test Description/ Test Steps	Test Input(s)	Expected Output(s)	Actual Output(s)	Result (Pass/Fail)
01	View All Power-cut Schedule	URL for the API Send Get request.	Display all Power-cut schedule details.	Display all Power-cut schedule details.	pass
02	Add details.	Attributes of user	"Inserted successfully"	"Inserted successfully"	pass
03	Update details.	Attributes to be updated along with the ID	"Updated successfully"	"Updated successfully"	pass
04	Delete details.	Given ID of the product to be deleted.	"Deleted successfully"	"Deleted successfully"	pass

## **Assumption and discussion**

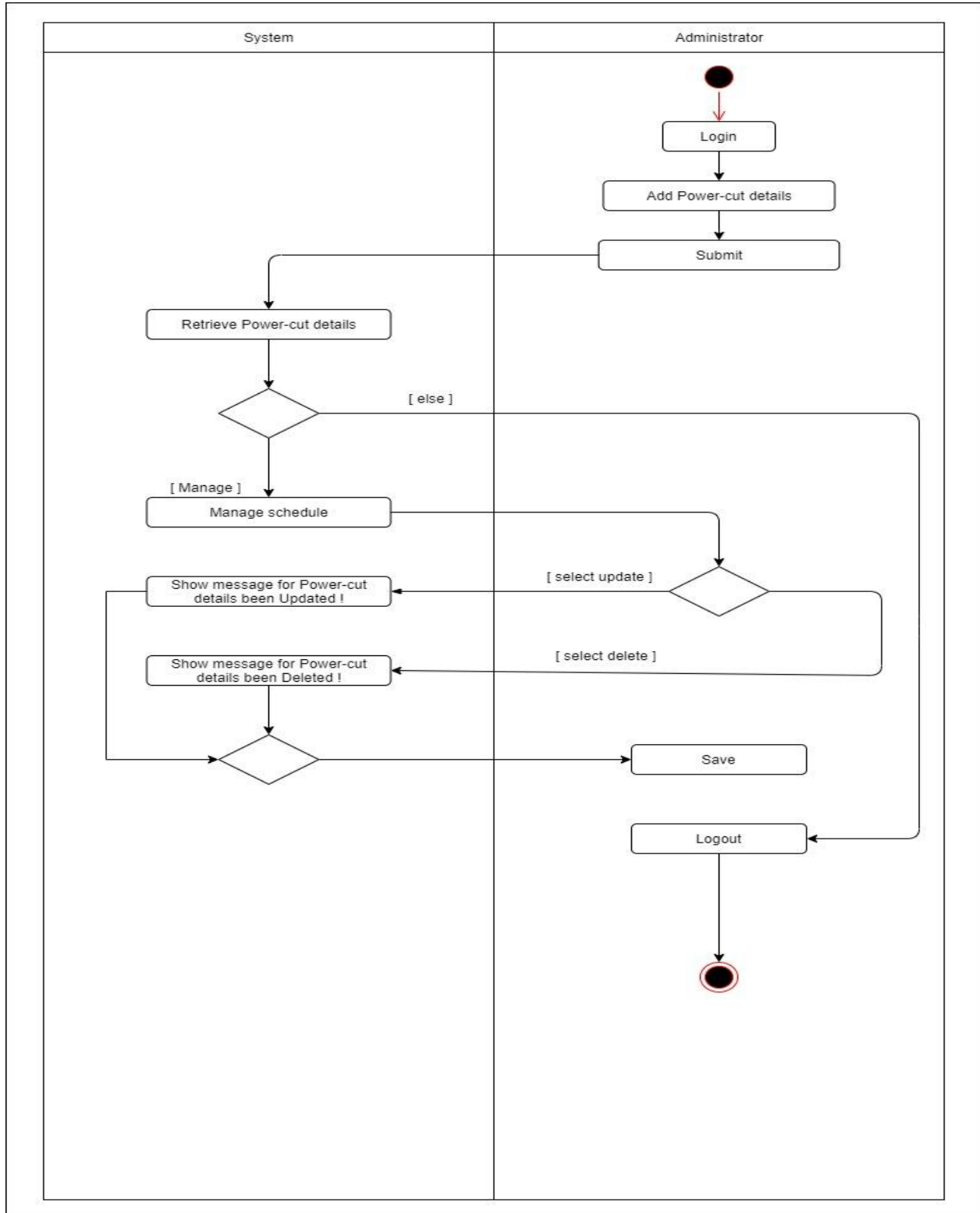
- Admin add Users to the system.
- Only admin can see the User Details.
- If we need to change product details, we can simply change it through product id.

## Class Diagram





## Activity Diagram



## POST

POST

http://localhost:8081/PAF-Project/ScheduleService/Schedules

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

KEY	VALUE
<input checked="" type="checkbox"/> sPeriod	12toMidnight
<input checked="" type="checkbox"/> sTotHrs	3
<input checked="" type="checkbox"/> sFromTime	7pm
<input checked="" type="checkbox"/> sToTime	10pm
<input checked="" type="checkbox"/> sArea	Homagama
<input checked="" type="checkbox"/> sSub	Colombo Zone 3
<input checked="" type="checkbox"/> sProvince	Western
Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

1 Inserted successfully !

## PUI

PUT

http://localhost:8081/PAF-Project/ScheduleService/Schedules

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Se

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

```
1 {
2   ....sID:"7",
3   ....sPeriod:"12toMidnight",
4   ....sTotHrs:"3Hours",
5   ....sFromTime:"7pm",
6   ....sToTime:"10pm",
7   ....sArea:"Narahepita",
8   ....sSub:"Colombo A zone",
9   ....sProvince:"Western"
10 }
```

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

1 Updated successfully!

## DELETE

DELETE

http://localhost:8081/PAF-Project/ScheduleService/Schedules

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Setti

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

```
1 <sData>
2 <sID>7/sID
3 </sData>
```

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

1 Deleted successfully

## GET

GET

http://localhost:8081/PAF-Project/ScheduleService/Schedules

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

KEY	VALUE
Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

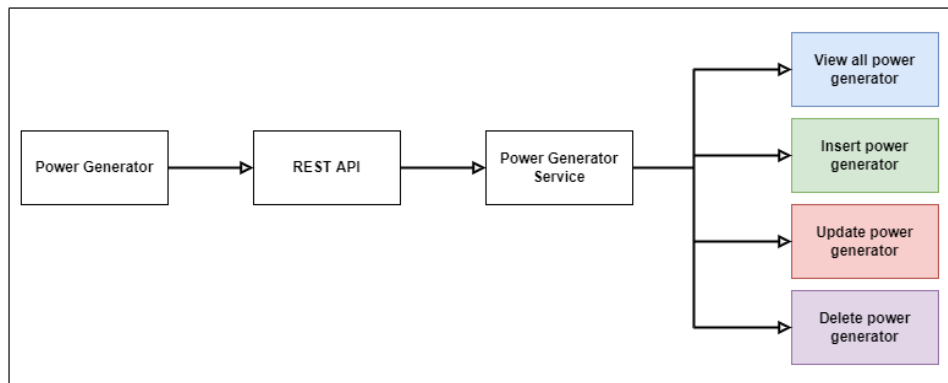
Preview

Visualize

sPeriod	sToHrs	sFromTime	sToTime	sArea	sSub	sProvince	Update	Remove
1	1	1	1	1	1	1	<button>Update</button>	<button>Remove</button>
1	1	1	1	1	1	1	<button>Update</button>	<button>Remove</button>
3Hours	Every 1	4.30pm	7.30pm	Colombo1-5	Colombo A zone	Western	<button>Update</button>	<button>Remove</button>
dfsdf	444	sfse	fsdfe	fefsg	fesfeg	fefes3	<button>Update</button>	<button>Remove</button>
3	12toMidnight	7pm	10pm	Homagama	Colombo Zone 3	Western	<button>Update</button>	<button>Remove</button>

sPeriod	sToHrs	sFromTime	sToTime	sArea	sSub	sProvince	Update	Remove
1	1	1	1	1	1	1	<button>Update</button>	<button>Remove</button>
1	1	1	1	1	1	1	<button>Update</button>	<button>Remove</button>
3Hours	Every 1	4.30pm	7.30pm	Colombo1-5	Colombo A zone	Western	<button>Update</button>	<button>Remove</button>
dfsdf	444	sfse	fsdfe	fefsg	fesfeg	fefes3	<button>Update</button>	<button>Remove</button>
3Hours	12toMidnight	7pm	10pm	Narahepita	Colombo A zone	Western	<button>Update</button>	<button>Remove</button>

## Service Design

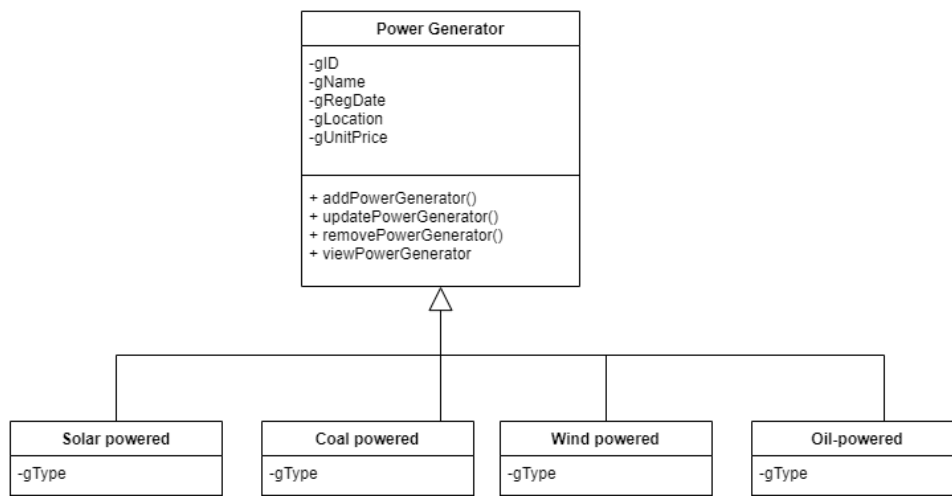


## CRUD

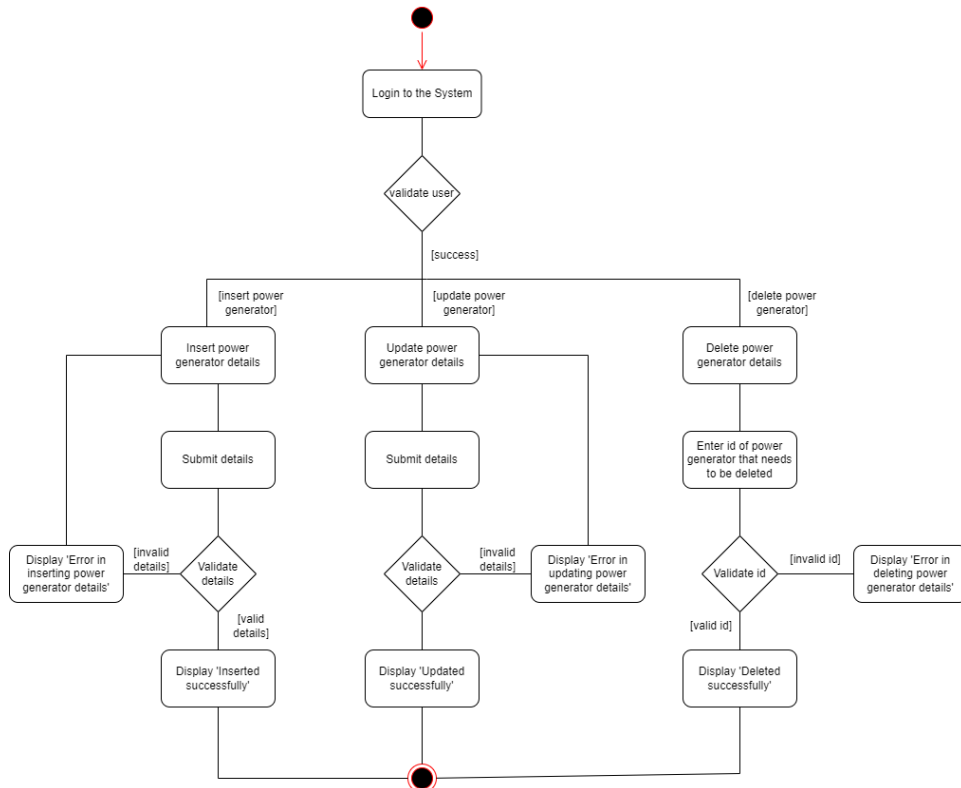
GET – View all the Power Generator Details	POST – Insert new Power Generator																																																
<p>URL: http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/readPowerGenerators</p> <p>Resource: Power Generator API</p> <p>Request: GET PowerGenerators/PowerGeneratorService/PowerGenerators/readPowerGenerators</p> <p>Media: TEXT_HTML</p> <p>Response:</p> <table><thead><tr><th>Power Generator Code</th><th>Power Generator Name</th><th>Power Generator Type</th><th>Power Generator Location</th><th>Unit Price</th><th>Registered Date/Modified Date</th></tr></thead><tbody><tr><td>g1225</td><td>Kelanitissa</td><td>Oil-fired</td><td>Colombo</td><td>24.75</td><td>2012-12-12</td></tr><tr><td>g572</td><td>Lakvijaya</td><td>Coal-fired</td><td>Puttalam</td><td>27.5</td><td>2007-07-25</td></tr><tr><td>g212</td><td>Sampur</td><td>Coal-fired</td><td>Trincomalee</td><td>26.0</td><td>2015-09-22</td></tr><tr><td>g2393</td><td>Kotmale</td><td>Hydroelectric</td><td>Kotmale</td><td>26.0</td><td>2009-03-02</td></tr><tr><td>g792</td><td>Victoria</td><td>Hydroelectric</td><td>Adhikarigama</td><td>24.75</td><td>2017-02-21</td></tr><tr><td>g475</td><td>Hambantota</td><td>Solar-powered</td><td>Hambantota</td><td>27.9</td><td>2008-02-07</td></tr><tr><td>g1234</td><td>test_generator</td><td>test_type</td><td>test_location</td><td>123.45</td><td>2022-04-23</td></tr></tbody></table> <p>"Error while reading Power Generators"</p>	Power Generator Code	Power Generator Name	Power Generator Type	Power Generator Location	Unit Price	Registered Date/Modified Date	g1225	Kelanitissa	Oil-fired	Colombo	24.75	2012-12-12	g572	Lakvijaya	Coal-fired	Puttalam	27.5	2007-07-25	g212	Sampur	Coal-fired	Trincomalee	26.0	2015-09-22	g2393	Kotmale	Hydroelectric	Kotmale	26.0	2009-03-02	g792	Victoria	Hydroelectric	Adhikarigama	24.75	2017-02-21	g475	Hambantota	Solar-powered	Hambantota	27.9	2008-02-07	g1234	test_generator	test_type	test_location	123.45	2022-04-23	<p>URL: http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/addPowerGenerator</p> <p>Resource: Power Generator API</p> <p>Request: POST PowerGenerators/PowerGeneratorService/PowerGenerators/addPowerGenerator</p> <p>Media: APPLICATION_FORM_URLENCODED</p> <p>Response: "Inserted successfully " "Error while adding the power generator"</p>
Power Generator Code	Power Generator Name	Power Generator Type	Power Generator Location	Unit Price	Registered Date/Modified Date																																												
g1225	Kelanitissa	Oil-fired	Colombo	24.75	2012-12-12																																												
g572	Lakvijaya	Coal-fired	Puttalam	27.5	2007-07-25																																												
g212	Sampur	Coal-fired	Trincomalee	26.0	2015-09-22																																												
g2393	Kotmale	Hydroelectric	Kotmale	26.0	2009-03-02																																												
g792	Victoria	Hydroelectric	Adhikarigama	24.75	2017-02-21																																												
g475	Hambantota	Solar-powered	Hambantota	27.9	2008-02-07																																												
g1234	test_generator	test_type	test_location	123.45	2022-04-23																																												
PUT – Update Power Generator Details	DELETE – Delete Power Generator																																																

<p>URL: http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/updatePowerGenerator</p> <p>Resource: Power Generator API</p> <p>Request: PUT PowerGenerators/PowerGeneratorService/PowerGenerators/updatePowerGenerator</p> <p>Media: APPLICATION_JSON</p> <p>Response: " Updated successfully " "Error while updating the Power Generator "</p>	<p>URL: http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/deletePowerGenerator</p> <p>Resource: Power Generator API</p> <p>Request: DELETE PowerGenerators/PowerGeneratorService/PowerGenerators/deletePowerGenerator</p> <p>Media: APPLICATION_XML</p> <p>Response: " Deleted successfully " " Error while deleting the Power Generator "</p>
---	--

## Class Diagram



## Activity Diagram



## GET

Postman GET request to `http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/headPowerGenerators`. The response is a 200 OK status with a table of power generator data.

Power Generator Code	Power Generator Name	Power Generator Type	Power Generator Location	Cost Price	Registered Date/Modified Date	Update	Remove
g1225	Kelantan	Oil-fired	Colombo	24.75	2012-12-12	Update	Remove
g572	Lakshya	Coal-fired	Punnam	27.5	2007-07-25	Update	Remove
g212	Saspar	Coal-fired	Tancomole	28.0	2015-09-22	Update	Remove
g2393	Kotmale	Hydroelectric	Kotmale	28.0	2009-03-02	Update	Remove
g792	Victoria	Hydroelectric	Athkarigama	24.75	2017-02-21	Update	Remove
g475	Manhattan	Solar-powered	Manhattan	27.9	2008-02-07	Update	Remove

## POST

Postman POST request to `http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/addPowerGenerator`. The response is a 200 OK status with a body indicating the generator was inserted successfully.

```

{
  "message": "Power generator inserted successfully"
}
  
```

## PUT

Postman PUT request to `http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/updatePowerGenerator`. The response is a 200 OK status with a body indicating the generator was updated successfully.

```

{
  "message": "Power generator updated successfully"
}
  
```

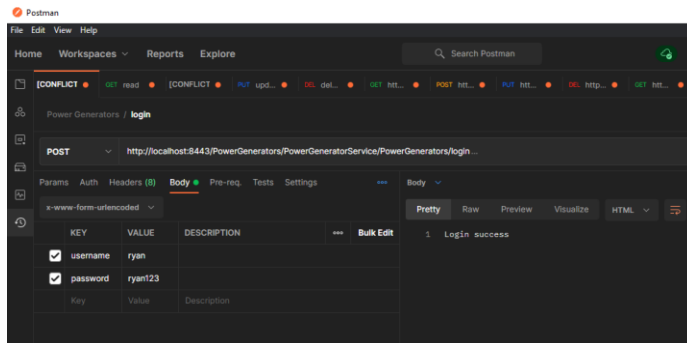
## DELETE

Postman DELETE request to `http://localhost:8443/PowerGenerators/PowerGeneratorService/PowerGenerators/deletePowerGenerator`. The response is a 200 OK status with a body indicating the generator was deleted successfully.

```

{
  "message": "Power generator deleted successfully"
}
  
```

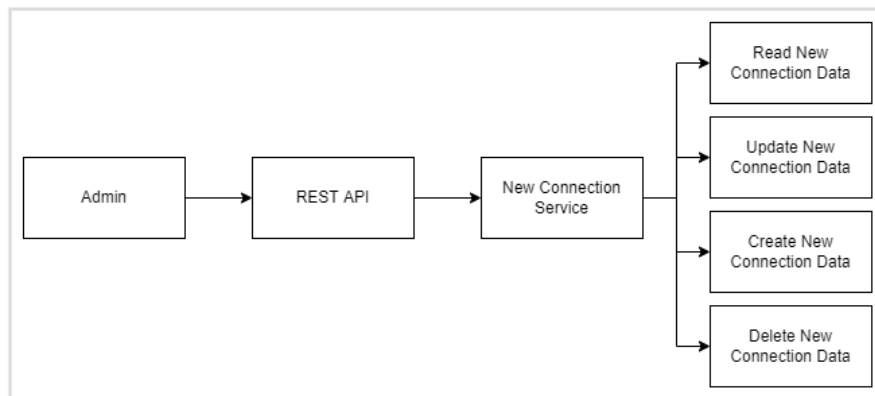
## Login



## Testing

Test ID	Test Description/ Test Steps	Test Input(s)	Expected Output(s)	Actual Output(s)	Result (Pass/Fail)
01	View all power generators	URL for the API Send Get request.	Display all power generator details.	Display all Power-cut schedule details.	pass
02	Add power generator.	Attributes of new power generator	"Inserted successfully"	"Inserted successfully"	pass
03	Update power generator.	Attributes to be updated along with the ID	"Updated successfully"	"Updated successfully"	pass
04	Delete power generator.	Given ID of the power generator to be deleted.	"Deleted successfully"	"Deleted successfully"	pass

## Service Design



## CRUD

GET - View all New Connections Requests	POST – Insert New Connection Request																																												
<p>URL: http://localhost:8090/NewConnectionService/NewConnection/Connection</p> <p>Resource: New Connection API</p> <p>Request: GET NewConnectionService/NewConnection/Connection</p> <p>Media: TEXT_HTML</p> <p>Response:</p> <table><tr><th>NC ID</th><th>NC Name</th><th>NC Email</th><th>NC Address</th><th>NC City</th><th>NC Mobile</th><th>NC Type</th><th>NC RegDate</th><th>NC Status</th><th>Update</th><th>Remove</th></tr><tr><td>4</td><td>Shanila</td><td>shanila@gmail.com</td><td>Piliyandala</td><td>Colombo</td><td>112519800</td><td>Domestic</td><td>2022-04-12</td><td>Pending</td><td><a href="#">Update</a></td><td><a href="#">Remove</a></td></tr><tr><td>8</td><td>Nalaka</td><td>Nalaka@gmail.com.lk</td><td>Dehiwala</td><td>Colombo</td><td>112519344</td><td>Industrial</td><td>2022-04-13</td><td>Pending</td><td><a href="#">Update</a></td><td><a href="#">Remove</a></td></tr><tr><td>9</td><td>Udaraka</td><td>Udaraka@gmail.com.lk</td><td>Kottawa</td><td>Colombo</td><td>112519000</td><td>Industrial</td><td>2022-01-13</td><td>Pending</td><td><a href="#">Update</a></td><td><a href="#">Remove</a></td></tr></table> <p>"Error while reading New Connection."</p>	NC ID	NC Name	NC Email	NC Address	NC City	NC Mobile	NC Type	NC RegDate	NC Status	Update	Remove	4	Shanila	shanila@gmail.com	Piliyandala	Colombo	112519800	Domestic	2022-04-12	Pending	<a href="#">Update</a>	<a href="#">Remove</a>	8	Nalaka	Nalaka@gmail.com.lk	Dehiwala	Colombo	112519344	Industrial	2022-04-13	Pending	<a href="#">Update</a>	<a href="#">Remove</a>	9	Udaraka	Udaraka@gmail.com.lk	Kottawa	Colombo	112519000	Industrial	2022-01-13	Pending	<a href="#">Update</a>	<a href="#">Remove</a>	<p>URL: http://localhost:8090/NewConnectionService/NewConnection/Connection</p> <p>Resource: New Connection API</p> <p>Request: POST NewConnectionService/NewConnection/Connection</p> <p>Media: APPLICATION_FORM_URLENCODED</p> <p>Response: "Inserted successfully" "Error while inserting New Connection."</p>
NC ID	NC Name	NC Email	NC Address	NC City	NC Mobile	NC Type	NC RegDate	NC Status	Update	Remove																																			
4	Shanila	shanila@gmail.com	Piliyandala	Colombo	112519800	Domestic	2022-04-12	Pending	<a href="#">Update</a>	<a href="#">Remove</a>																																			
8	Nalaka	Nalaka@gmail.com.lk	Dehiwala	Colombo	112519344	Industrial	2022-04-13	Pending	<a href="#">Update</a>	<a href="#">Remove</a>																																			
9	Udaraka	Udaraka@gmail.com.lk	Kottawa	Colombo	112519000	Industrial	2022-01-13	Pending	<a href="#">Update</a>	<a href="#">Remove</a>																																			
PUT – Update New Connection Request	DELETE – Delete New Connection Request																																												
<p>URL: http://localhost:8090/NewConnectionService/NewConnection/Connection</p> <p>User Resource: New Connection API I</p> <p>Request: PUT NewConnectionService/NewConnection/Connection</p> <p>Media: APPLICATION_JSON</p>	<p>URL: http://localhost:8090/NewConnectionService/NewConnection/Connection</p> <p>Resource: New Connection API</p> <p>Request: DELETE NewConnectionService/NewConnection/Connection</p> <p>Media: APPLICATION_XML</p>																																												

Response: "Updated successfully" "Error while updating New Connection. "	Response: "Deleted successfully" "Error while deleting New Connection. "
--	--

## Testing

Test ID	Test Description/ Test Steps	Test Input(s)	Expected Output(s)	Actual Output(s)	Result (Pass/Fail)
01	View all New Connections	URL for the API Send Get request.	Display all New Connection Requests	Display all New Connection Requests	pass
02	Request New Connection	Attributes entered by new customer	"Inserted successfully"	"Inserted successfully"	pass
03	Update New Connection request	Attributes to be updated along with the ID	"Updated successfully"	"Updated successfully"	pass
04	Delete New Connection request	Given ID of the product to be deleted.	"Deleted successfully"	"Deleted successfully"	pass

## GET

GET <http://localhost:8090/NewConnectionService/NewConnection/Connection>

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Headers [↔ 6 hidden](#)

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Content-Type	application/xml	
Key	Value	Description

Body Cookies Headers (5) Test Results Status: 200 OK Time: 30 ms Size: 1.62 KB

[Pretty](#) [Raw](#) [Preview](#) [Visualize](#)

NC ID	NC Name	NC Email	NC Address	NC City	NC Mobile	NC Type	NC RegDate	NC Status	Update	Remove
4	Shanila	shanila@gmail.com	Piliyandala	Colombo	112519800	Domestic	2022-04-12	Pending	<a href="#">Update</a>	<a href="#">Remove</a>
8	Nalaka	Nalaka@gmail.com.lk	Dehiwala	Colombo	112519344	Industrial	2022-04-13	Pending	<a href="#">Update</a>	<a href="#">Remove</a>
9	Udaraka	Udaraka@gmail.com.lk	Kottawa	Colombo	112519000	Industrial	2022-01-13	Pending	<a href="#">Update</a>	<a href="#">Remove</a>

## POST

POST <http://localhost:8090/NewConnectionService/NewConnection/Connection>

Params Authorization Headers (8) Body ☒ Pre-request Script Tests Settings

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

<input type="checkbox"/> ncID	01
<input checked="" type="checkbox"/> ncName	Thirimanne
<input checked="" type="checkbox"/> ncEmail	Thirimanne@gmail.com
<input checked="" type="checkbox"/> ncAddress	Kottawa
<input checked="" type="checkbox"/> ncCity	Colombo
<input checked="" type="checkbox"/> ncMobile	0112511000
<input checked="" type="checkbox"/> ncType	Industrial
<input checked="" type="checkbox"/> ncRegDate	2022-01-01
<input checked="" type="checkbox"/> ncStatus	Pending
Key	Value

Body Cookies Headers (5) Test Results Status: 200 OK Time: 17 ms

[Pretty](#) [Raw](#) [Preview](#) [Visualize](#) [Text](#) [JSON](#)

```
1 Inserted successfully
```



## PUT

PUT ▼ http://localhost:8090/NewConnectionService/NewConnection/Connection/

Params Authorization Headers (9) **Body** ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL ● JSON ▼

```
1 {
2   ... "ncID": "10",
3   ... "ncName": "UpdatedThiri",
4   ... "ncEmail": "Thiri@gmail.com",
5   ... "ncAddress": "UpdatedAddress",
6   ... "ncCity": "UpdatedCity",
7   ... "ncMobile": "0112519844",
8   ... "ncType": "Updated",
9   ... "ncRegDate": "2022-04-01",
10  ... "ncStatus": "Completed"
11 }
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize Text ▼ ≡

```
1 Updated successfully
```

## DELETE

DELETE ▼ http://localhost:8090/NewConnectionService/NewConnection/Connection

Params Authorization Headers (9) **Body** ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL ● XML ▼

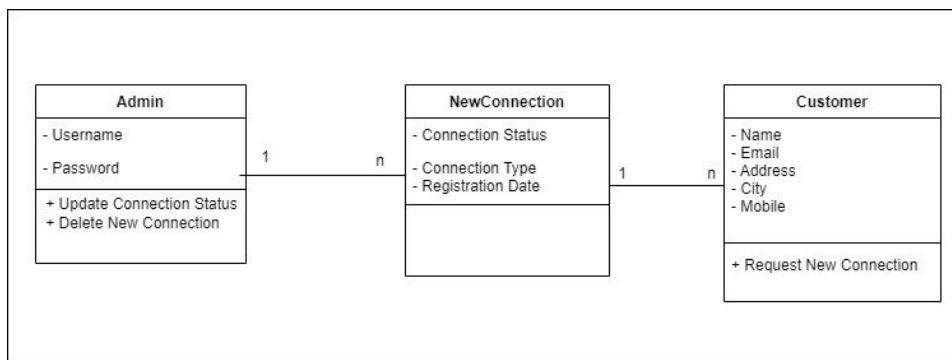
```
1 <itemData>
2   ... <ncID>10</ncID>
3 </itemData>
```

Body Cookies Headers (5) Test Results

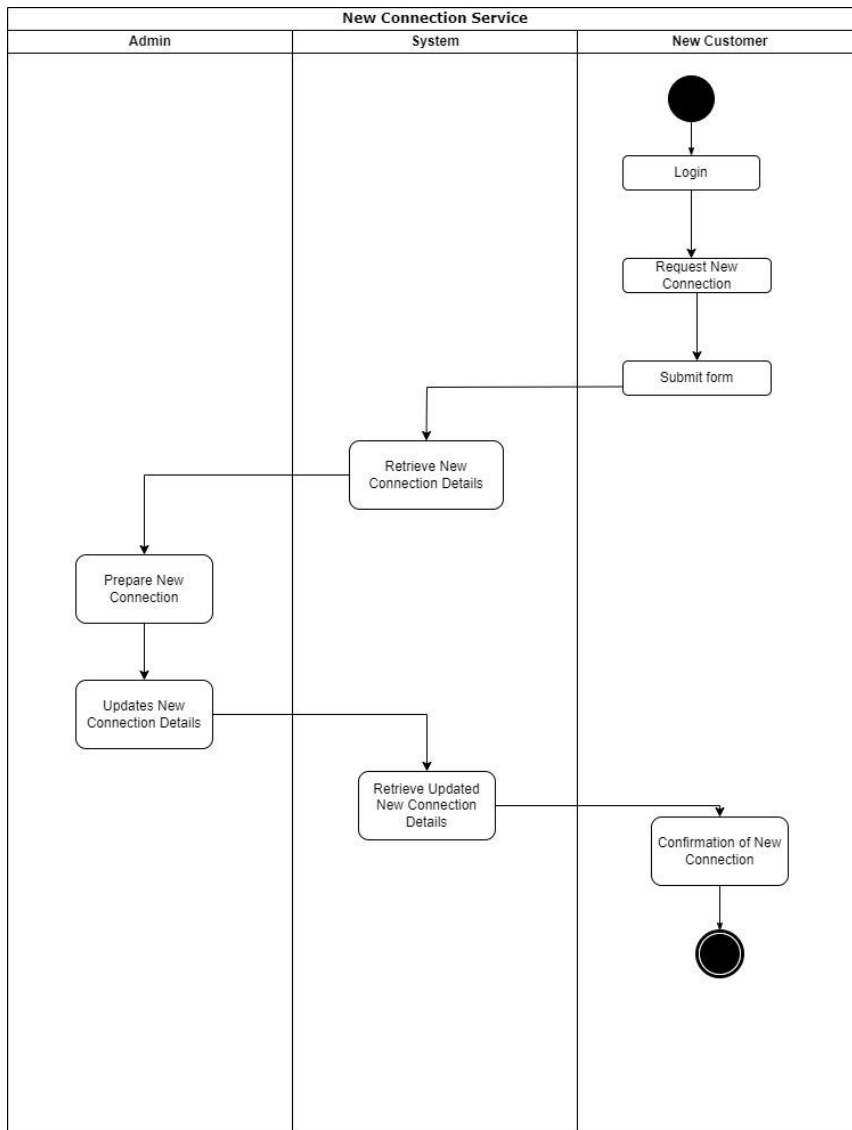
Pretty Raw Preview Visualize Text ▼ ≡

```
1 Deleted successfully
```

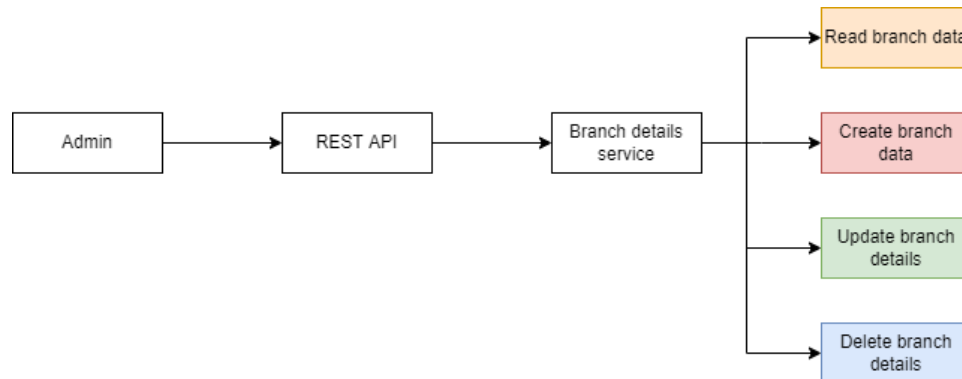
## Class Diagram



## Activity Diagram



## Service Design



## CRUD

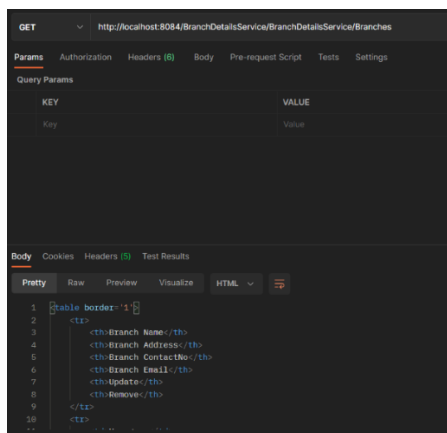
GET – View all branches' details	POST – Insert New Connection Request																																												
<p>URL: http://localhost:8084/BranchDetailsService/BranchDetailsService/Branches</p> <p>Resource: Branch Details API</p> <p>Request: GET BranchDetailsService/BranchDetailsService/Branches</p> <p>Media: TEXT_HTML</p> <p>Response:</p> <table><tr><th>NC ID</th><th>NC Name</th><th>NC Email</th><th>NC Address</th><th>NC City</th><th>NC Mobile</th><th>NC Type</th><th>NC RegDate</th><th>NC Status</th><th>Update</th><th>Remove</th></tr><tr><td>4</td><td>Shanila</td><td>shanila@gmail.com</td><td>Piliyandala</td><td>Colombo</td><td>112519800</td><td>Domestic</td><td>2022-04-12</td><td>Pending</td><td><input type="button" value="Update"/></td><td><input type="button" value="Remove"/></td></tr><tr><td>8</td><td>Nalaka</td><td>Nalaka@gmail.com.lk</td><td>Dehiwala</td><td>Colombo</td><td>112519344</td><td>Industrial</td><td>2022-04-13</td><td>Pending</td><td><input type="button" value="Update"/></td><td><input type="button" value="Remove"/></td></tr><tr><td>9</td><td>Udaraka</td><td>Udaraka@gmail.com.lk</td><td>Kottawa</td><td>Colombo</td><td>112519000</td><td>Industrial</td><td>2022-01-13</td><td>Pending</td><td><input type="button" value="Update"/></td><td><input type="button" value="Remove"/></td></tr></table> <p>"Error while reading the branch."</p>	NC ID	NC Name	NC Email	NC Address	NC City	NC Mobile	NC Type	NC RegDate	NC Status	Update	Remove	4	Shanila	shanila@gmail.com	Piliyandala	Colombo	112519800	Domestic	2022-04-12	Pending	<input type="button" value="Update"/>	<input type="button" value="Remove"/>	8	Nalaka	Nalaka@gmail.com.lk	Dehiwala	Colombo	112519344	Industrial	2022-04-13	Pending	<input type="button" value="Update"/>	<input type="button" value="Remove"/>	9	Udaraka	Udaraka@gmail.com.lk	Kottawa	Colombo	112519000	Industrial	2022-01-13	Pending	<input type="button" value="Update"/>	<input type="button" value="Remove"/>	<p>URL: http://localhost:8084/BranchDetailsService/BranchDetailsService/Branches</p> <p>Resource: Branch Details API</p> <p>Request: POST BranchDetailsService/BranchDetailsService/Branches</p> <p>Media: APPLICATION_FORM_URLENCODED</p> <p>Response: "Inserted successfully" "Error while inserting the branch"</p>
NC ID	NC Name	NC Email	NC Address	NC City	NC Mobile	NC Type	NC RegDate	NC Status	Update	Remove																																			
4	Shanila	shanila@gmail.com	Piliyandala	Colombo	112519800	Domestic	2022-04-12	Pending	<input type="button" value="Update"/>	<input type="button" value="Remove"/>																																			
8	Nalaka	Nalaka@gmail.com.lk	Dehiwala	Colombo	112519344	Industrial	2022-04-13	Pending	<input type="button" value="Update"/>	<input type="button" value="Remove"/>																																			
9	Udaraka	Udaraka@gmail.com.lk	Kottawa	Colombo	112519000	Industrial	2022-01-13	Pending	<input type="button" value="Update"/>	<input type="button" value="Remove"/>																																			
PUT – Update New Connection Request	DELETE – Delete New Connection Request																																												
<p>URL: http://localhost:8084/BranchDetailsService/BranchDetailsService/Branches</p> <p>Resource: Branch Details API</p> <p>Request: POST BranchDetailsService/BranchDetailsService/Branches</p> <p>Media: APPLICATION_FORM_URLENCODED</p>	<p>URL: http://localhost:8084/BranchDetailsService/BranchDetailsService/Branches</p> <p>Resource: Branch details API</p> <p>Request: DELETE BranchDetailsService/BranchDetailsService/Branches</p> <p>Media: APPLICATION_XML</p>																																												

Response: "Inserted successfully" "Error while inserting the branch"	Response: "Deleted successfully" "Error while deleting the branch "
--	---

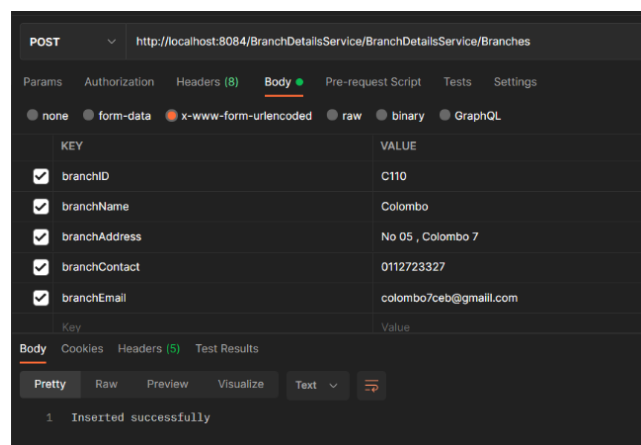
## Testing

Test ID	Test Description/ Test Steps	Test Input(s)	Expected Output(s)	Actual Output(s)	Result (Pass/Fail)
01	View branch details	URL for the API Send Get request.	Display all branches' details.	Display all branches' details.	pass
02	Add a branch detail.	Attributes of branch	"Inserted successfully"	"Inserted successfully"	pass
03	Update branch detail.	Attributes to be updated along with the ID	"Updated successfully"	"Updated successfully"	pass
04	Delete branch detail.	Given ID of the branch to be deleted.	"Deleted successfully"	"Deleted successfully"	pass

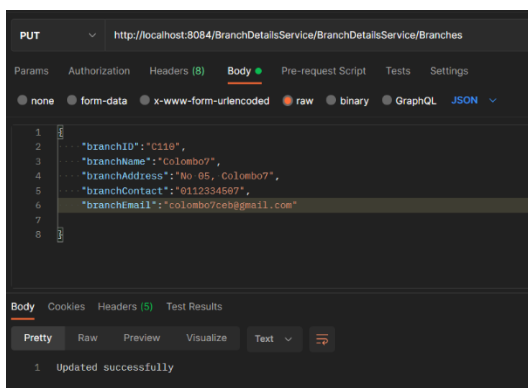
## GET



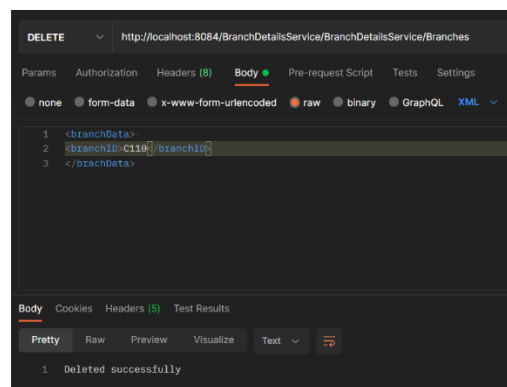
## POST



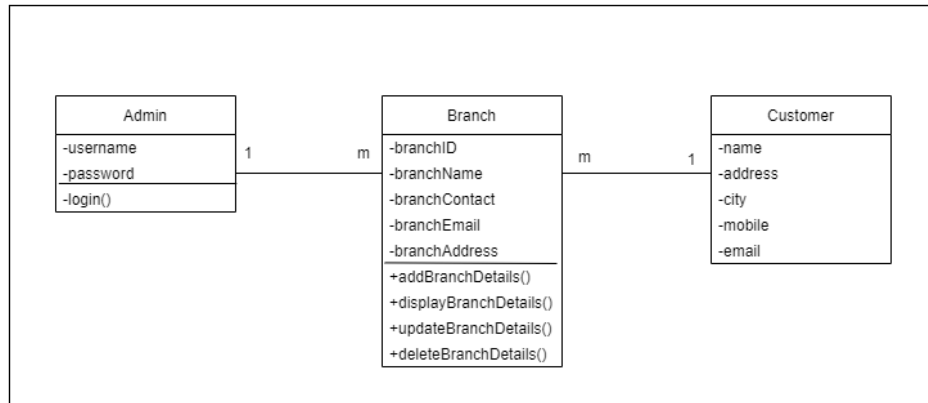
## PUT



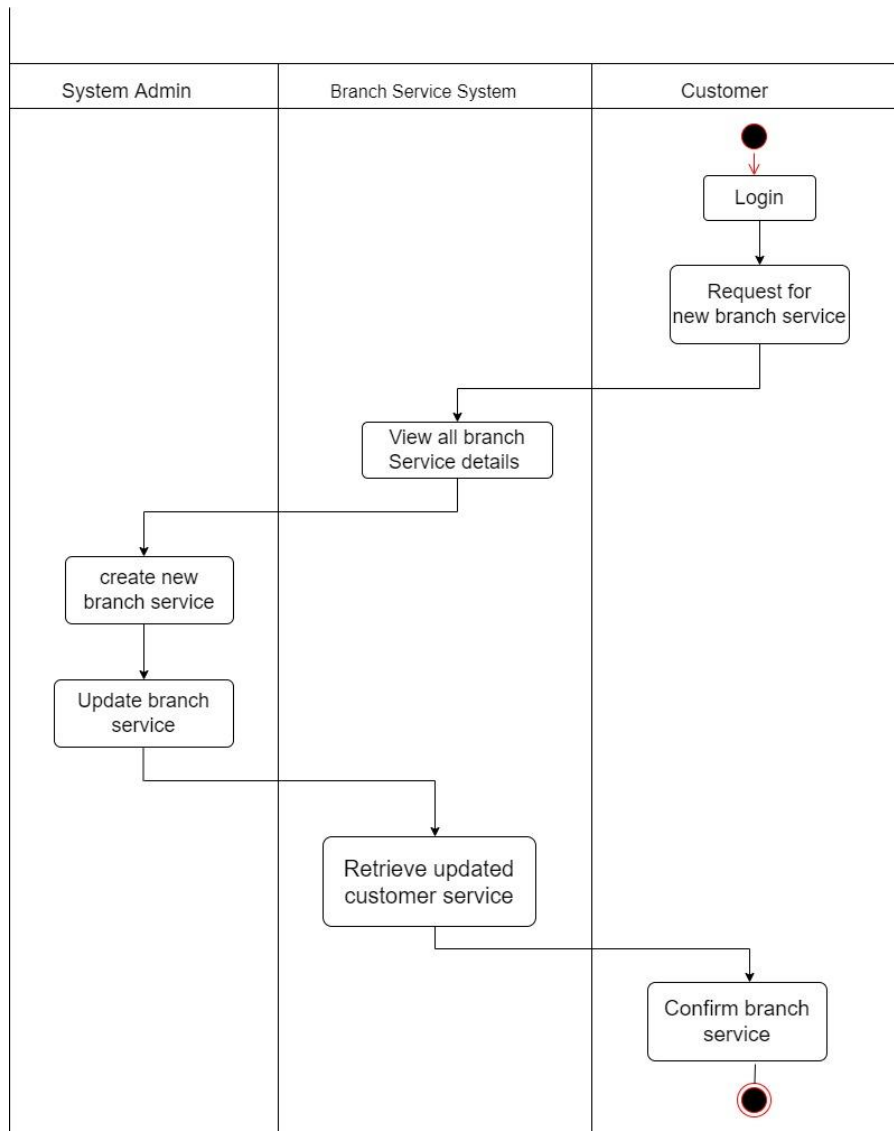
## DELETE



## Class Diagram

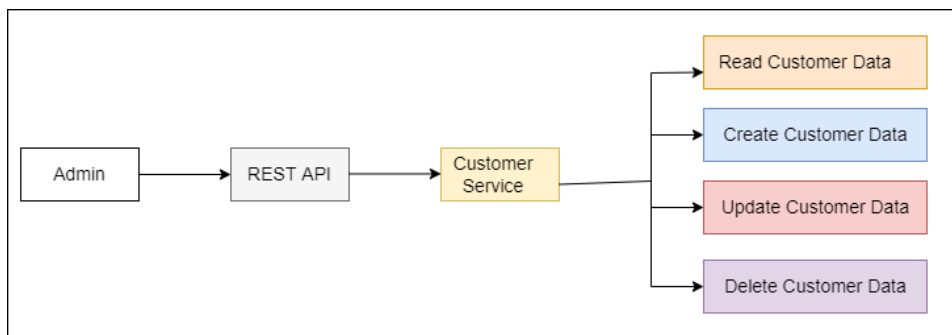


## Activity Diagram



## IT20258580- Kanakaratne M.R. : Customer Service

### Service Design



### CRUD

GET - View all customer details	POST – Insert new customer																		
<p>URL: http://localhost:8090/CustomerService/myService/Customers</p> <p>Resource: Customer Service API</p> <p>Request: GET CustomerService/myService/Customers</p> <p>Media: TEXT_HTML</p> <table><thead><tr><th>Customer Code</th><th>Customer Name</th><th>Customer Email</th><th>Customer Address</th><th>Update</th><th>Remove</th></tr></thead><tbody><tr><td>1093</td><td>Netthani De Silva</td><td>netthani@gmail.com</td><td>Ambalangoda, Galle</td><td>Galle</td><td>073546567</td></tr><tr><td>8097</td><td>Sachith Liyanage</td><td>sachith345@gmail.com</td><td>Kottawa, Piliyandala</td><td>Colombo</td><td>0918948948</td></tr></tbody></table> <p>"Error while reading the user."</p>	Customer Code	Customer Name	Customer Email	Customer Address	Update	Remove	1093	Netthani De Silva	netthani@gmail.com	Ambalangoda, Galle	Galle	073546567	8097	Sachith Liyanage	sachith345@gmail.com	Kottawa, Piliyandala	Colombo	0918948948	<p>URL: http://localhost:8090/CustomerService/myService/Customers</p> <p>Resource: Customer Service API</p> <p>Request: POST CustomerService/myService/Customers</p> <p>Media: APPLICATION_FORM_URLENCODED</p> <p>Response: "Inserted successfully" "Error while inserting the Customer."</p>
Customer Code	Customer Name	Customer Email	Customer Address	Update	Remove														
1093	Netthani De Silva	netthani@gmail.com	Ambalangoda, Galle	Galle	073546567														
8097	Sachith Liyanage	sachith345@gmail.com	Kottawa, Piliyandala	Colombo	0918948948														
PUT – Update customer details	DELETE – Delete customer details																		
<p>URL: http://localhost:8090/CustomerService/Customers</p> <p>Resource: Customer Service API</p> <p>Request: PUT CustomerService/myService/Customers</p> <p>Media: APPLICATION_JSON</p> <p>Response: "Updated successfully" "Error while updating the Customer."</p>	<p>URL: http://localhost:8090/CustomerService/myService/Customers</p> <p>Resource: Customer Service API</p> <p>Request: DELETE CustomerService/myService/Customers</p> <p>Media: APPLICATION_XML</p> <p>Response: "Deleted successfully" "Error while deleting the Customer."</p>																		

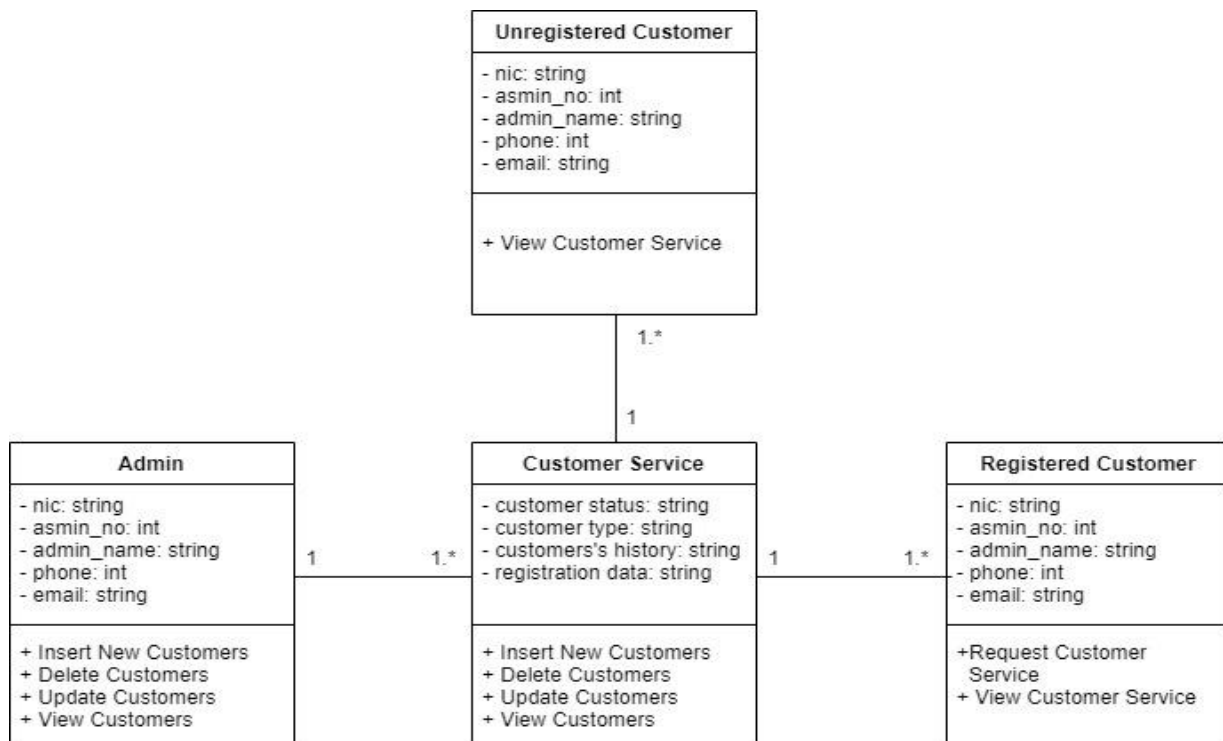
## Testing

Test ID	Test Description/ Test Steps	Test Input(s)	Expected Output(s)	Actual Output(s)	Result (Pass/Fail)
01	View all customer details	URL for the API Send Get request.	Display all Customer details.	Display all Customer details.	pass
02	Add customer details.	Attributes of customer	"Inserted successfully"	"Inserted successfully"	pass
03	Update customer details.	Attributes to be updated along with the ID	"Updated successfully"	"Updated successfully"	pass
04	Delete customer details.	Given ID of the product to be deleted.	"Deleted successfully"	"Deleted successfully"	pass

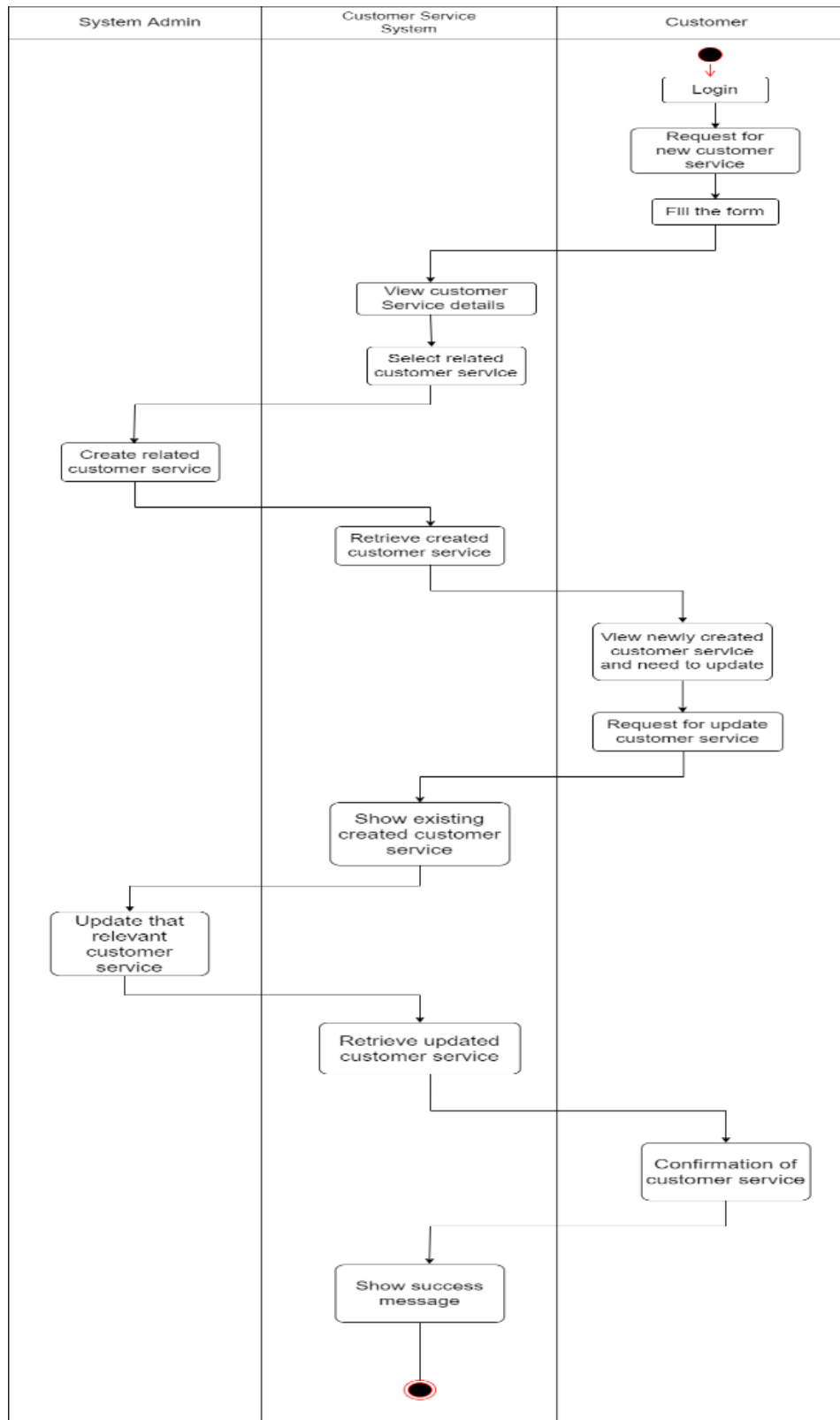
## Assumption and discussion

- Admin adds customers to the system.

## Class Diagram

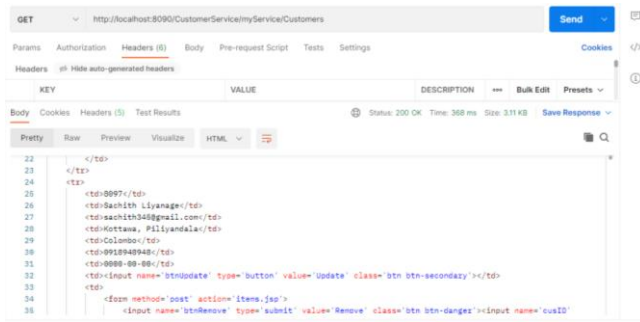


## Activity Diagram

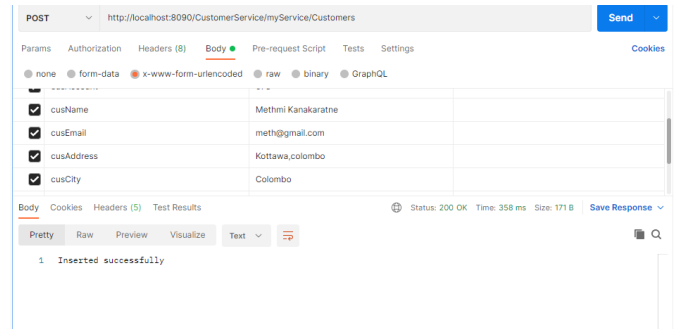




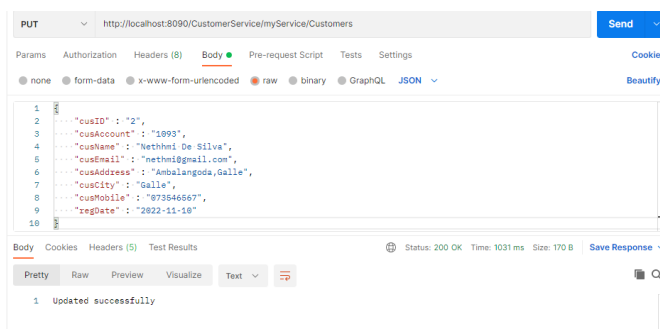
## Get



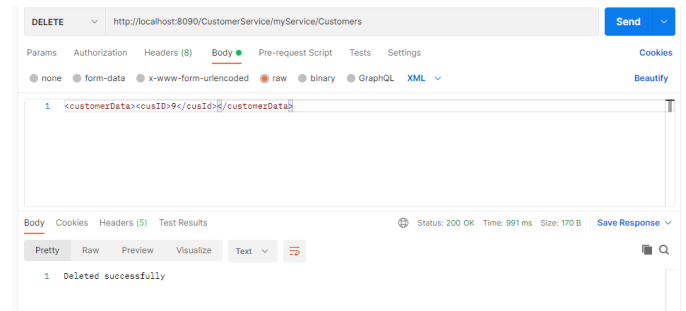
## Post



## Put



## Delete



## References

- CRUD operations – Lab sheet 5 & 6
- Onion Diagram - [Introduction of Stakeholder Analysis - Onion Diagram & Power Interest Matrix](#)

## Appendix