WSO2 ESB Assignment

The purpose of this assignment is to learn and adapt to the WSO2 ESB. Also, this assignment link to the previous banking API. Before starting this assignment, you should learn the following concepts.

- XML and SOAP services.
- WSO2 ESB proxy services.
- WSO2 ESB Mediators.

You can use the following tools for development.

- WSO2 EI
- WSO2 Integration Studio IDE
- SOAP UI

1 - Transform XML data to JSON banking API format

Business Requirement	Description
1.0	The data format is given in the xml format and it should transform and insert to banking system. (See sample xml data format).
1.1	Type of the account should be correctly map with the API account type value (See account type mappings).
1.2	A proxy service should be implemented to accept the XML payload.

Task 2 - Get the JSON formatted data and transform to XML format

Business Requirement	Description
1.0	The user data should be transformed into the following XML format.
1.1	Type of the account should be correctly map to source system format. (See account type mappings)
1.2	A proxy service should be implemented to receive a list of user's account with XML format.

Account Type Mappings

Source System (XML data)	Target System (Banking API)
Isuru	1
Nirogya	2

Sample XML format.

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
    <user id="1">
        <name>Nuwan Gunasekara</name>
        <password>$456SSSS</password>
        <accountType>Isuru</accountType>
    </user>
    <user id="2">
```

```
        <name>Gayan  Perera</name>
        <password>$55dddfww</password>
        <accountType>Nirogya</accountType>
    </user>
    <user  id="3">
        <name>Ruwan  Weerasekara</name>
        <password>45dddd@@</password>
        <accountType>Isuru</accountType>
    </user>
    <user  id="4">
        <name>Gihan  Dias</name>
        <password>sdfgee@@45</password>
        <accountType>Nirogya</accountType>
    </user>
    <user  id="5">
        <name>Dimuth  Rupasinghe</name>
        <password>ssfd##4ssd</password>
        <accountType>Isuru</accountType>
    </user>
</users>
```

Answers

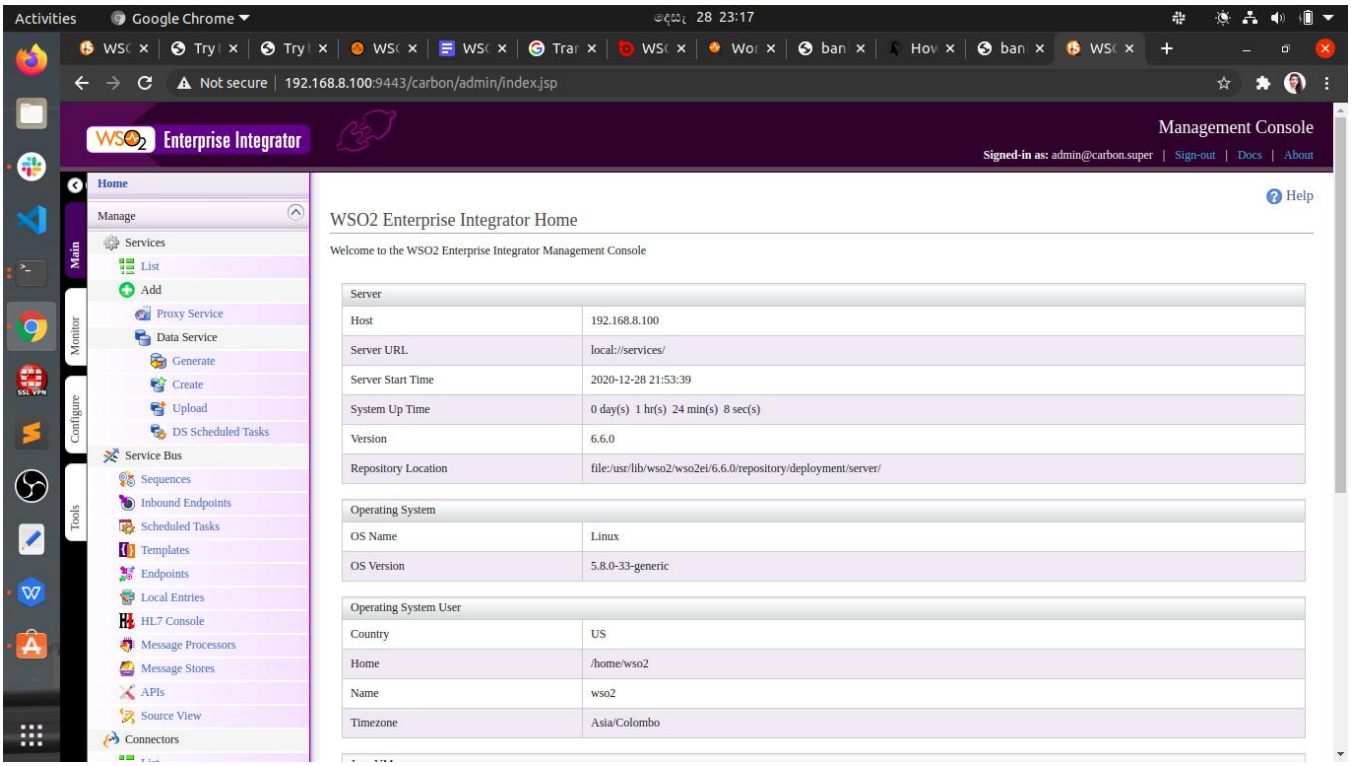
Step 01 :- Download WSO2 Enterprise Integrator.

Step 02 :- Open a terminal and type **'sudo wso2ei-6.6.0-integrator'**

Step 03 :-

```
INFO {org.wso2.carbon.core.internal.StartupFinalizerServiceComponent} - WSO2 Carbon started in 21 sec
INFO {org.wso2.carbon.ui.internal.CarbonUIServiceComponent} - Mgt Console URL  : https://192.168.8.100:9443/carbon/
INFO {org.wso2.carbon.core.services.util.CarbonAuthenticationUtil} - 'admin@carbon.super [-1234]' logged in at [2020
```

Copy the above Mgt Console URL and paste it in the browser and log into the system by giving 'admin' for both user name and password the window will look like below.



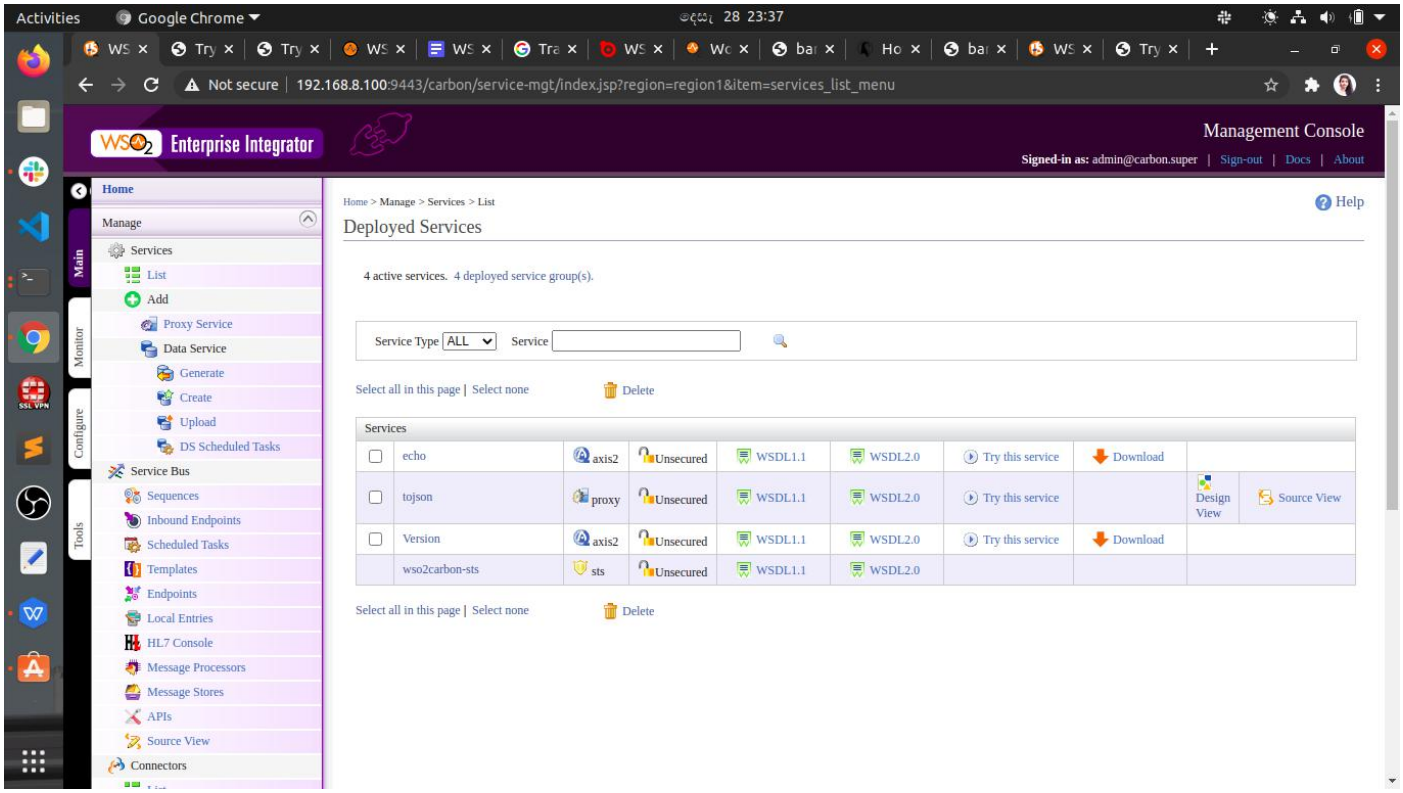
Task 01:-

Go to Proxy Services -> Custom Proxy -> Switch to design view

then type the following code there and save it.

```
<proxy xmlns="http://ws.apache.org/ns/synapse"
       name="tojson"
       transports="https,http"
       statistics="disable"
       trace="disable"
       startOnLoad="true">
   <target>
      <inSequence>
         <property name="messageType" value="application/json" scope="axis2"/>
         <respond/>
      </inSequence>
   </target>
   <description/>
</proxy>
```

then we can see that proxy service in the list as follow.



- Save the given xml file as banking_data.xml
- Open a terminal where banking_data.xml file is located

- Get the url of the service that we added by clicking the try this service button. It will be look like this
  https://192.168.8.100:9443/services/tojson?tryit
- Then type as follows

curl --insecure -v -X POST -H "Content-Type:application/xml" -d@banking_data.xml "https://192.168.8.100:9443/services/tojson"

- Then the result will be as follows

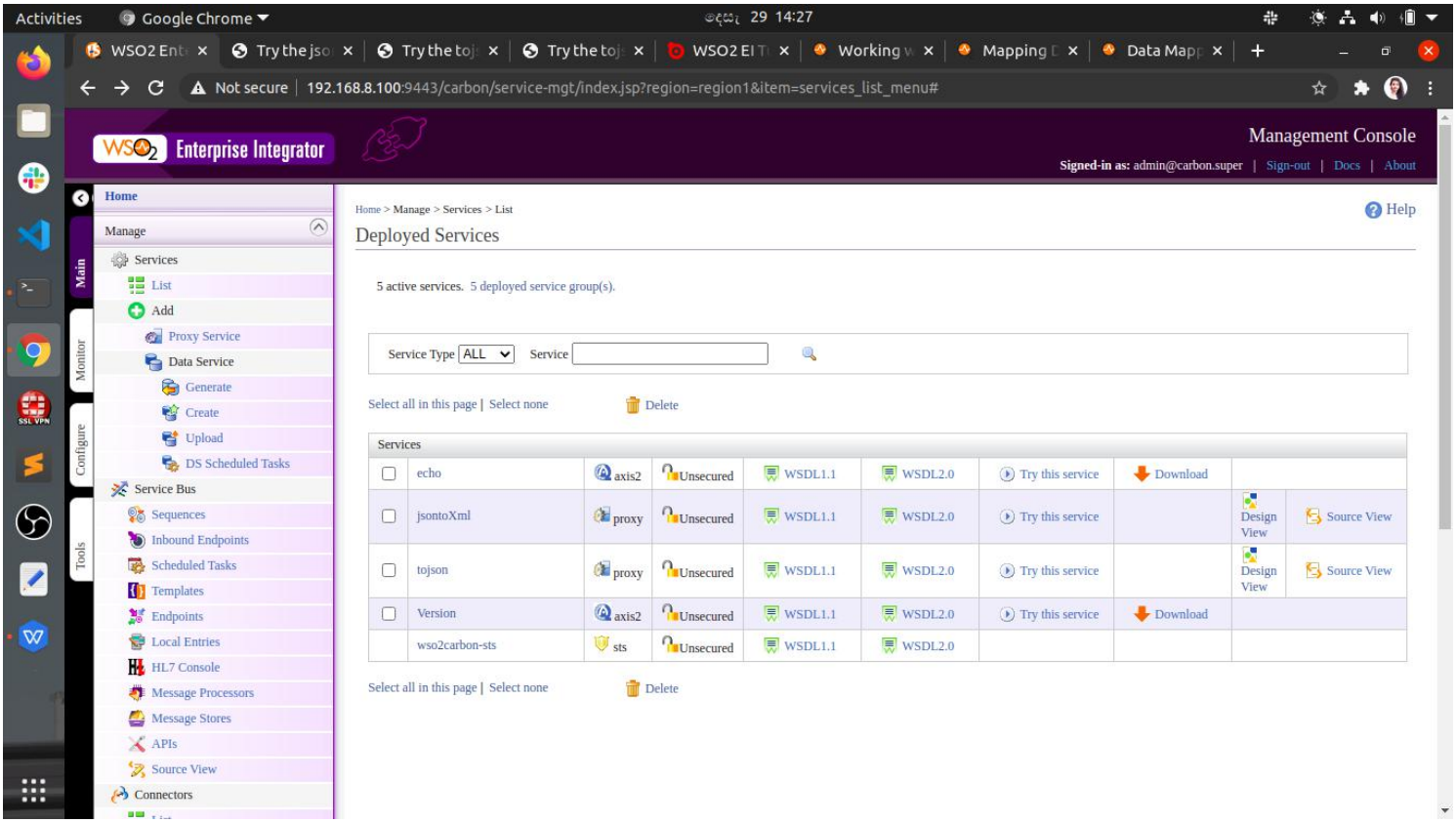
Task 02 :-

Go to Proxy Services -> Custom Proxy -> Switch to design view then type the following code there and save it.

```
<?xml version="1.0" encoding="UTF-8"?>
<proxy xmlns="http://ws.apache.org/ns/synapse" name="jsontoXml"
startOnLoad="true" statistics="disable" trace="disable"
transports="http,https">
    <target>
        <inSequence>
            <property name="messageType" scope="axis2"
value="application/xml"/>
            <respond/>
        </inSequence>
```

```
    </target>
    <description/>
</proxy>
```

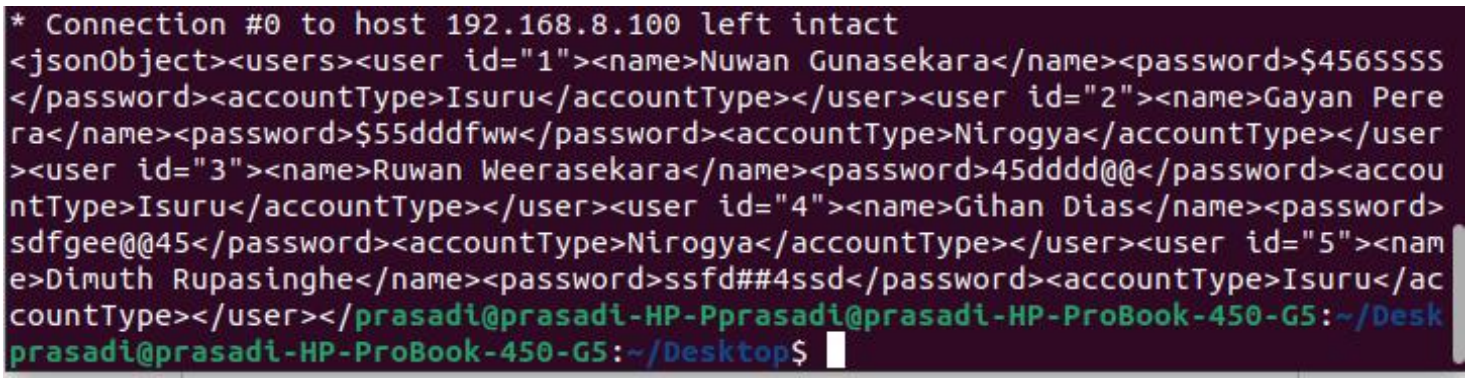
then we can see that proxy service in the list as follow.



- Save the json formatted data that we got in to a file called **banking_data.json**
- Open a terminal where banking_data.json file is located
- Get the url of the service that we added by clicking the try this service button. It will be look like this **https://192.168.8.100:9443/services/jsontoXml?tryit**
- Then type as follows

curl --insecure -v -X POST -H "Content-Type:application/json" -d@banking_data.json "https://192.168.8.100:9443/services/jsontoXml"

➢ the result Then will be as follows

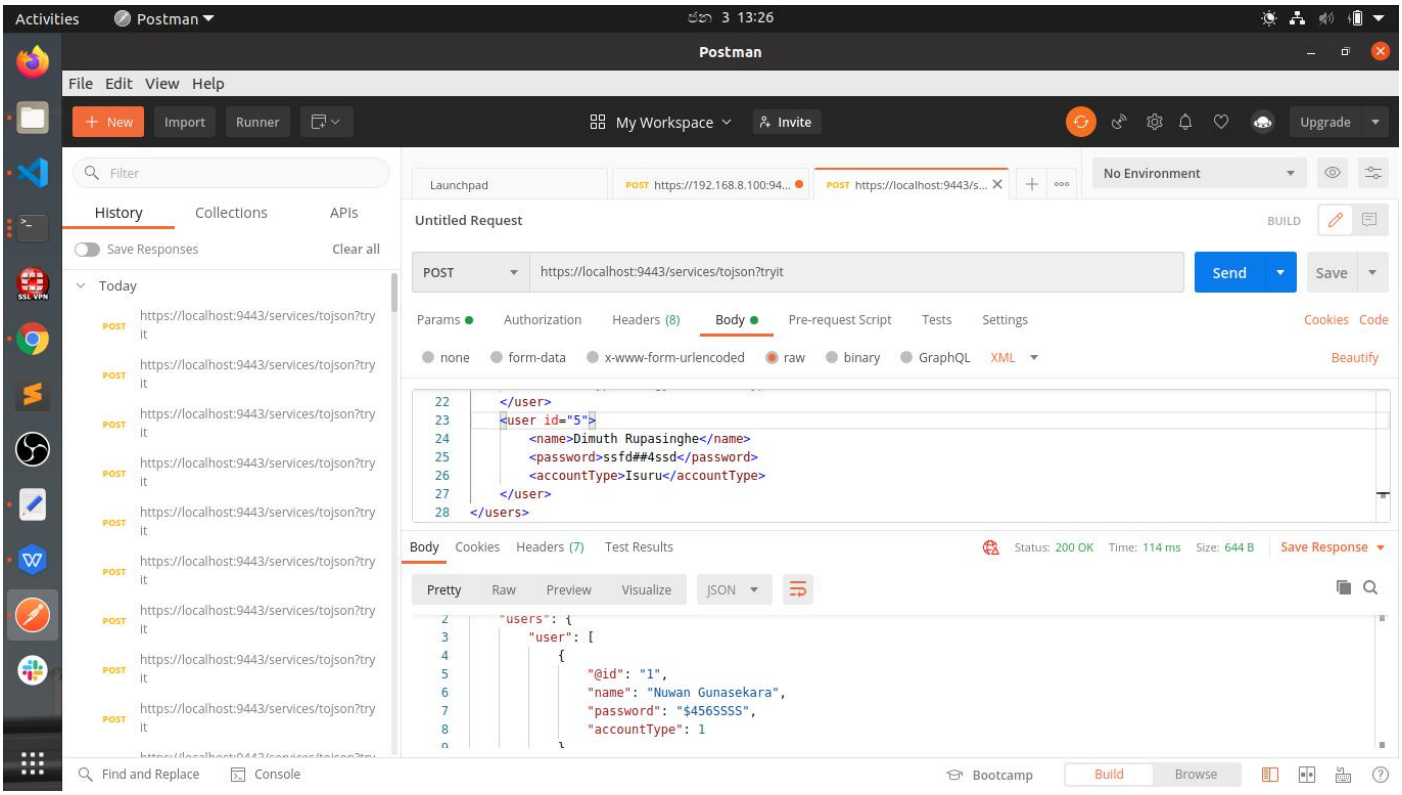


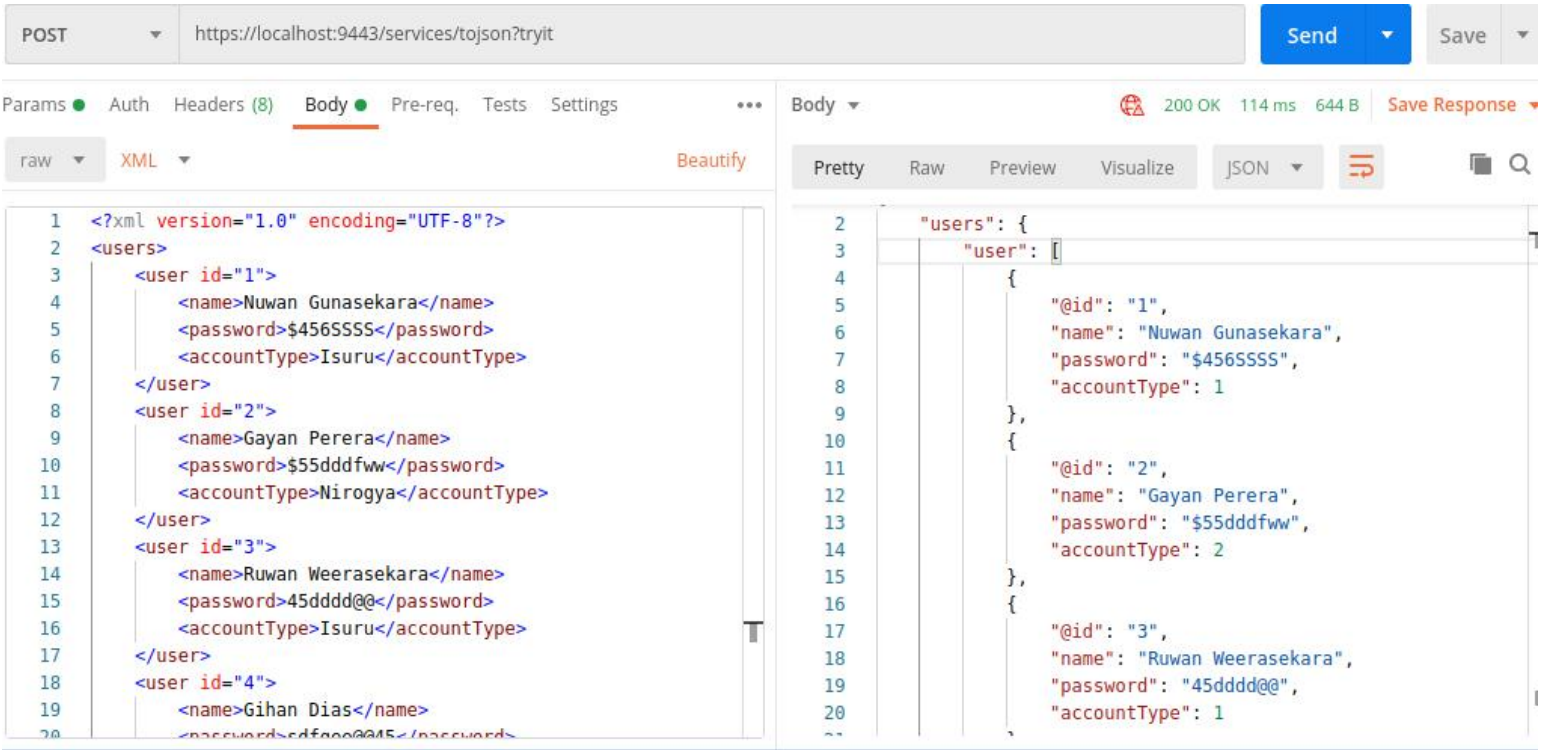
Data Type mapping And The Finalized Result

➢ I did the Data mapping part by using the

* Pay Load Factory Mediator (To format The output by putting Variables to it)
* Filter Mediator (Filter the account_type property and Assign it a value 1 and 2 according to it name by using a regex)
* For Each Mediator (Iterate Through the list)

Following is the Result I got When I sent the XML pay load





➢ Following I have attached the Code that I use for this in **tojson** proxy service

```
<?xml version="1.0" encoding="UTF-8"?>
<proxy xmlns="http://ws.apache.org/ns/synapse" name="tojson"
startOnLoad="true" statistics="disable" trace="disable"
transports="http,https">
<target>
<inSequence>
<foreach expression="//users/user" id="foreach_1">
<sequence>
<property expression="//user/@id" name="identificatio"/>
<property expression="//user/name" name="name"/>
<property expression="//user/password" name="password"/>
<property expression="//user/accountType" name="accountType"/>

<filter regex="Isuru" source="$ctx:accountType">
<then>
<property name="updatedAccountType" value="1"/>
</then>
</filter>
<filter regex="Nirogya" source="$ctx:accountType">
<then>
<property name="updatedAccountType" value="2"/>
</then>
</filter>
<payloadFactory media-type="xml">
<format>
<user id="$1">
<name>$2</name>
<password>$3</password>
<accountType>$4</accountType>
```

```
</user>
</format>
<args>
<arg  evaluator="xml"  expression="$ctx:identificatio"/>
<arg  evaluator="xml"  expression="$ctx:name"/>
<arg  evaluator="xml"  expression="$ctx:password"/>
<arg  evaluator="xml"  expression="$ctx:updatedAccountType"/>
</args>
</payloadFactory>
</sequence>
</foreach>
<property name="messageType" scope="axis2" value="application/json"/>
<respond/>
</inSequence>
</target>
<description/>
</proxy>
```