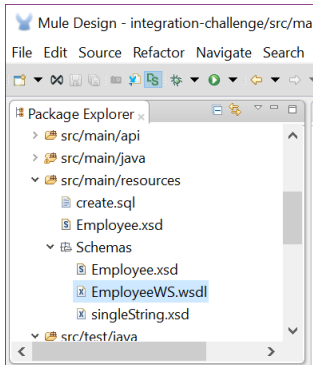
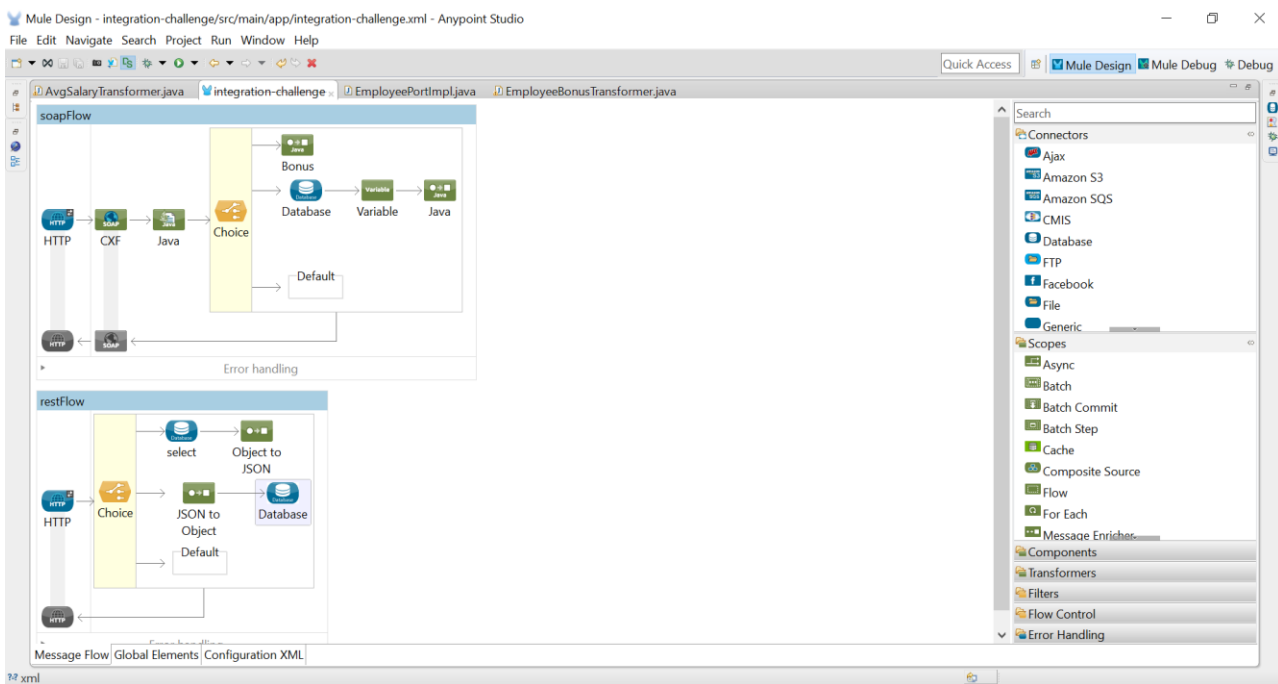


I have used MySQL because I have already installed it on my machine.

I have created the WSDL using Oracle Jdeveloper (soa suite). WSDL and XSDs are in folder Schemas:



Flows design:



I have used soapui to test soap and rest flows.

Test evidence of calculate bonus:

SoapUI 5.2.1

File Project Suite Case Step Tools Desktop Help

Empty SOAP REST Import Save All Forum Trial Preferences Proxy

Search Forum Online Help

Projects

- REST Project 1
 - http://localhost:8081
 - Employees (/employees)
 - Employees
 - Request 1
 - salary
 - EmployeePortServiceSoapBinding
 - calculateEmployeeBonus
 - Request 1
 - getAverageSalary
 - Request 1

Request Properties

| Property | Value |
|---------------------|----------------------------|
| Name | Request 1 |
| Description | |
| Message Size | 540 |
| Encoding | UTF-8 |
| Endpoint | http://localhost:8081/soap |
| Timeout | |
| Bind Address | |
| Follow Redirects | true |
| Username | |
| Password | |
| Domain | |
| Authentication Type | No Authorization |

Request 1

Auth Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)

response time: 9ms (369 bytes)

SoapUI log http log jetty log error log wsrm log memory log

Request 1

```
<?xml version='1.0' encoding='UTF-8'?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <emp:part>
      <!--Optional:-->
      <empl:id/>empl:id
      <empl:firstName/>empl:firstName
      <empl:lastName/>empl:lastName
      <empl:salary/>empl:salary
    </emp:part>
  </soapenv:Body>
</soap:Envelope>
```

Request 1

```
<?xml version='1.0' encoding='UTF-8'?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns3:decimal xmlns="http://www.w3.org/2001/XMLSchema" xmlns:ns2="http://xmlns.oracle.com/Application/Project1/EmployeeWS">42.088817737677700499780201353132724761962890628</ns3:decimal>
  </soap:Body>
</soap:Envelope>
```

Headers (5) Attachments (0) SSL Info WSS (0) JMS (0)

5:7

Test evidence of calculate average salary:

SoapUI 5.2.1

File Project Suite Case Step Tools Desktop Help

Empty SOAP REST Import Save All Forum Trial Preferences Proxy

Search Forum Online Help

Projects

- REST Project 1
 - http://localhost:8081
 - Employees (/employees)
 - Employees
 - Request 1
 - salary
 - EmployeePortServiceSoapBinding
 - calculateEmployeeBonus
 - Request 1
 - calculateAverageSalary
 - Request 1

Request Properties

| Property | Value |
|---------------------|----------------------------|
| Name | Request 1 |
| Description | |
| Message Size | 540 |
| Encoding | UTF-8 |
| Endpoint | http://localhost:8081/soap |
| Timeout | |
| Bind Address | |
| Follow Redirects | true |
| Username | |
| Password | |
| Domain | |
| Authentication Type | No Authorization |

Request 1

Auth Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)

Request 1

```
<?xml version='1.0' encoding='UTF-8'?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <emp:part>
      <!--Optional:-->
      <empl:id/>empl:id
      <empl:firstName/>empl:firstName
      <empl:lastName/>empl:lastName
      <empl:salary/>empl:salary
    </emp:part>
  </soapenv:Body>
</soap:Envelope>
```

Request 1

```
<?xml version='1.0' encoding='UTF-8'?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns3:decimal xmlns="http://www.w3.org/2001/XMLSchema" xmlns:ns2="http://xmlns.oracle.com/Application/Project1/EmployeeWS">120000.000000</ns3:decimal>
  </soap:Body>
</soap:Envelope>
```

Headers (5) Attachments (0) SSL Info WSS (0) JMS (0)

5:7

Test evidence of POST with JSON:

The screenshot shows the SoapUI 5.2.1 interface. The main window displays a POST request to the endpoint `http://localhost:8081/employees`. The request body is a JSON array: `[{"last_name": "Lemeszenski", "first_name": "Daniel", "id": 1, "salary": 120000.00}]`. The response pane on the right shows the raw response, which is a 200 status with a JSON array: `[{"last_name": "Doe", "first_name": "John", "id": 1, "salary": 120000.00}, {"last_name": "Lemeszenski", "first_name": "Daniel", "id": 2, "salary": 120000.00}]`. A blue box highlights the response body with the text "The actual content of the last received response".

Test evidence of GET:

The screenshot shows a web browser window with the address bar displaying `localhost:8081/employees`. The page content shows a JSON array of employee data: `[{"last_name": "Doe", "first_name": "John", "id": 1, "salary": 120000.00}, {"last_name": "Lemeszenski", "first_name": "Daniel", "id": 2, "salary": 120000.00}]`.