

CURSO : 1924 - Desarrollo de Servicios Web II
PROFESOR : Jesus Antonio Alpaca Rendon
SEMESTRE : 2024-05
CICLO : Quinto
SECCION : T6JA - T6JB
GRUPO : 01
DURACION : 50 minutos
FECHA : 10/10/2024

NOTA

ALUMNO (A) : WILMER OCAMPO QUISPE

CASO DE LABORATORIO T2

Consideraciones generales:

- Se considerará el orden, la claridad de las respuestas y las buenas prácticas.
- Las preguntas deben resolverse de acuerdo con los conceptos discutidos o analizados en clase. Para ello, aplicará su propio criterio para dar una solución a los problemas planteados
- Para el desarrollo de la siguiente aplicación cree el proyecto Spring Boot con el nombre **DSWII_T2_XXXX** (donde XXXX serán sus apellidos y nombres) cree la estructura de carpetas necesarias, enviar en un comprimido (con la misma estructura **DSWII_T2_XXXX**) su examen de word al blackboard donde incluya la URL del repositorio publico donde estará la resolución para revisar los commits y el código fuente

LOGRO DE LA EVALUACION:

Al término de la evaluación, el alumno demuestra su capacidad para crear Servicios Web SOAP y Rest y consumir el primero para obtener datos de alguna BD u otra fuente. Además de poder procesar esos datos y brindar otra respuesta.

Consolidado

Pregunta	Puntaje		Llenar solo en caso de Recalificación justificada	
	Máximo	Obtenido	Sustento	Puntaje
1	20			
Nota Recalificada				

DSW-II T2 REPOSITORIO GIT HUB:

[WilmerOcampo/DSWII T2 OcampoWilmer \(github.com\)](https://github.com/WilmerOcampo/DSWII_T2_OcampoWilmer)

Pregunta 01

Se tiene un sistema de registro de pagos el cual cuenta con un web service SOAP para devolver la lista de pagos que se tiene hasta la fecha.

Por otro lado se tiene una API REST que debe procesar esos pagos, hacer algunos cálculos y devolver en JSON la información solicitada. Para ello, la API REST debe consumir el servicio de SOAP para obtener los datos y hacer ese procesamiento.

Desarrollar un web service SOAP que devuelva la lista de pagos de una BD, La tabla contendrá las siguientes columnas:

- idPago
- fecha de Pago(puede ser String si se desea)
- monto total del pago
- moneda(soles o dólares)(String recomendable)

Debe crear una aplicación API REST que consuma el web service.

Una vez que tenga la lista de pagos, deberá procesar estos datos de la siguiente manera:

El monto total del pago, puede estar en soles o dólares, dependiendo el valor de la columna "moneda" (tipo de cambio: 3.85), la aplicación API REST deberá devolver los datos en JSON de esta forma:

Tenemos que considerar que el monto original será el monto total que recibimos de SOAP, este puede ser soles o dólares, hay que indicar y apartir de ese monto se calcula su IGV(18%) y obtener el valor neto o monto base.

Ejemplo:

Total: 847.50

Impuesto: 847.50×0.18

Valor Neto: Total – Impuesto -> 694.95

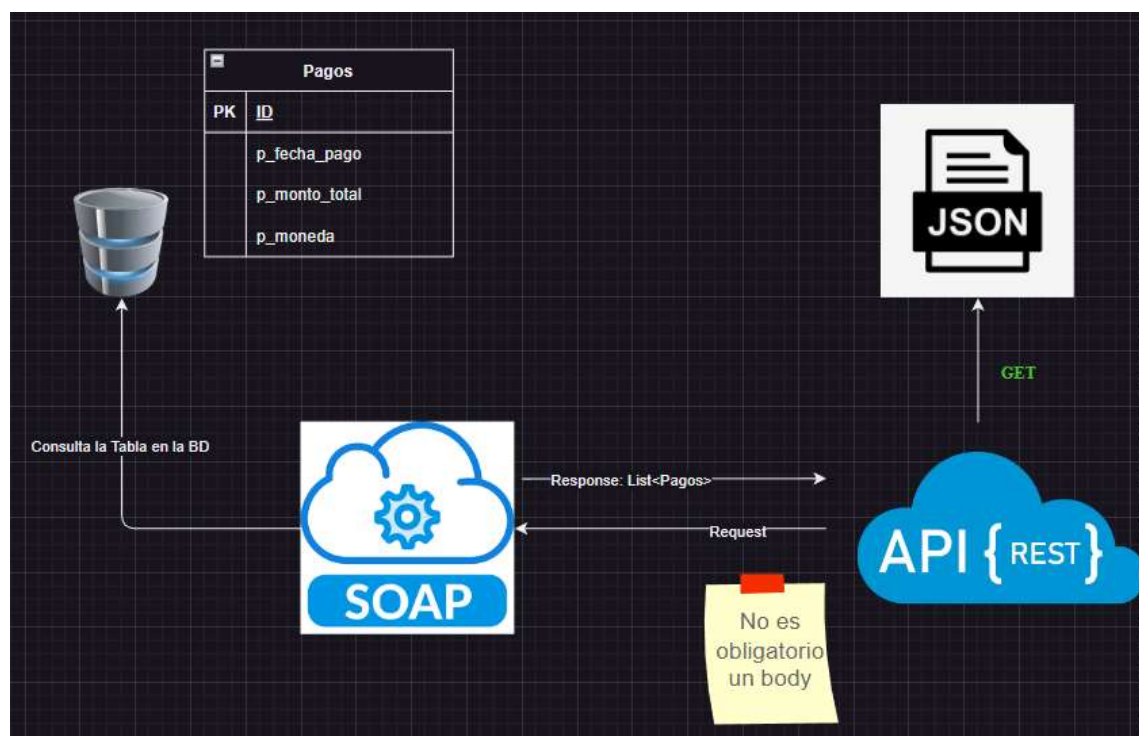
```
[
  {
    "monedas": {
      "soles": 980.056,
      "dolares": 254.56
    },
    "montoOriginal": 254.56,
    "monedaOriginal": "Dolares",
    "impuestos": 45.82,
    "valorNeto": 208.74,
    "fechaVenta": "2023-10-26"
  },
  {
    "monedas": {
      "soles": 847.50,
      "dolares": 220.13
    },
    "montoOriginal": 847.50,
```

```

    "monedaOriginal": "Soles",
    "impuestos": 152.55,
    "valorNeto": 694.95,
    "fechaVenta": "2023-10-25"
  },
  {
    "monedas": {
      "soles": 1797.30,
      "dolares": 466.83
    },
    "montoOriginal": 1797.30,
    "monedaOriginal": "Soles",
    "impuestos": 323.514,
    "valorNeto": 1473.786,
    "fechaVenta": "2023-10-26"
  }
]

```

La arquitectura del proyecto seria de esta forma:



RÚBRICA DE LA PRIMERA EVALUACIÓN

Rúbrica					
Pregunta	puntaje	12 puntos	9 puntos	4 puntos	0 puntos
1	10 puntos	Realiza el control de versiones al inicio, la estructura del proyecto, la bd, JPA, el menu y commit al final	Realiza el control de versiones al inicio, la estructura del proyecto y la bd	Realiza el control de versiones al inicio, la estructura del proyecto y la bd	Ninguna clase se encuentra mapeada. El proyecto no puede ejecutarse

		4 puntos	3 puntos	1 punto	0 puntos
2	10 puntos	Responde la pregunta claramente, y explica las razones correctamente, no es ambigua la respuesta	La respuesta es un poco ambigua, ya que no indica exactamente la solución	No es la respuesta correcta.	La pregunta no se respondió

SOLUCIÓN

SCRIPT MYSQL

```
CREATE SCHEMA `wo_soap_db` DEFAULT CHARACTER SET utf8 ;

SELECT * FROM wo_soap_db.payments;

INSERT INTO `wo_soap_db`.`payments` (`payment_date`, `total_amount`, `currency`) VALUES ('2024-10-10', '950.09', 'Soles');
INSERT INTO `wo_soap_db`.`payments` (`payment_date`, `total_amount`, `currency`) VALUES ('2024-10-11', '860.52', 'Dólares');
INSERT INTO `wo_soap_db`.`payments` (`payment_date`, `total_amount`, `currency`) VALUES ('2024-10-11', '3200.23', 'Soles');
INSERT INTO `wo_soap_db`.`payments` (`payment_date`, `total_amount`, `currency`) VALUES ('2024-10-11', '520.03', 'Dólares');
INSERT INTO `wo_soap_db`.`payments` (`payment_date`, `total_amount`, `currency`) VALUES ('2024-10-11', '1025.99', 'Soles');

SELECT * FROM wo_soap_db.payments WHERE `payment_date` = '2024-10-11';
```

SOAP UI: <http://localhost:8090/ws/payment.wsdl>

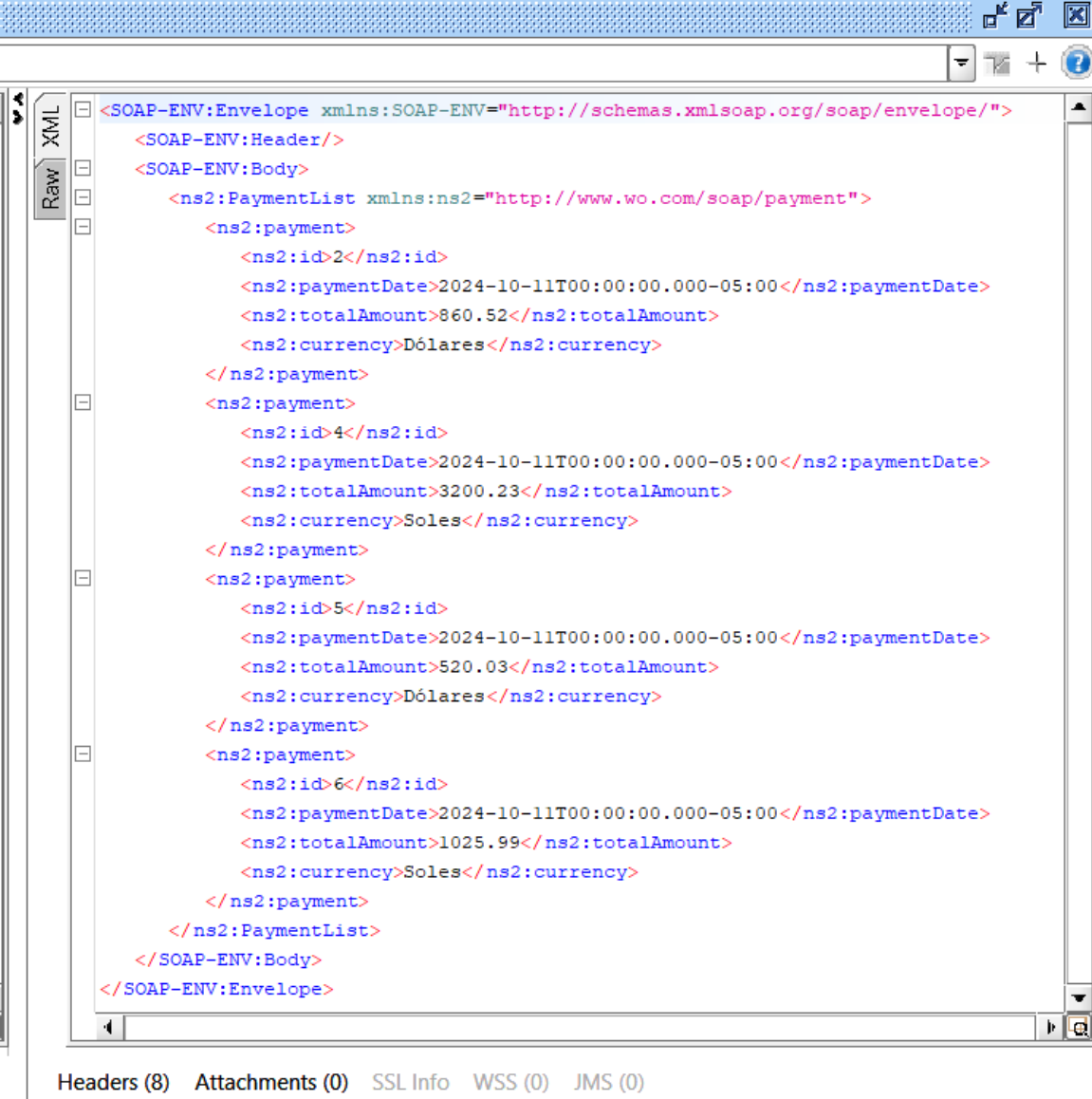
REQUEST



The screenshot shows the SOAP UI interface for 'Request 1'. The address bar displays 'http://localhost:8090/ws/payment'. The XML view is expanded, showing the following structure:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:pay=
  <soapenv:Header/>
  <soapenv:Body>
    <pay:PaymentListRequest>
      <pay:paymentDate>2024-10-11</pay:paymentDate>
    </pay:PaymentListRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

RESPONSE



The screenshot shows the SOAP UI interface for the response. The XML view is expanded, showing the following structure:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:PaymentList xmlns:ns2="http://www.wo.com/soap/payment">
      <ns2:payment>
        <ns2:id>2</ns2:id>
        <ns2:paymentDate>2024-10-11T00:00:00.000-05:00</ns2:paymentDate>
        <ns2:totalAmount>860.52</ns2:totalAmount>
        <ns2:currency>Dólares</ns2:currency>
      </ns2:payment>
      <ns2:payment>
        <ns2:id>4</ns2:id>
        <ns2:paymentDate>2024-10-11T00:00:00.000-05:00</ns2:paymentDate>
        <ns2:totalAmount>3200.23</ns2:totalAmount>
        <ns2:currency>Soles</ns2:currency>
      </ns2:payment>
      <ns2:payment>
        <ns2:id>5</ns2:id>
        <ns2:paymentDate>2024-10-11T00:00:00.000-05:00</ns2:paymentDate>
        <ns2:totalAmount>520.03</ns2:totalAmount>
        <ns2:currency>Dólares</ns2:currency>
      </ns2:payment>
      <ns2:payment>
        <ns2:id>6</ns2:id>
        <ns2:paymentDate>2024-10-11T00:00:00.000-05:00</ns2:paymentDate>
        <ns2:totalAmount>1025.99</ns2:totalAmount>
        <ns2:currency>Soles</ns2:currency>
      </ns2:payment>
    </ns2:PaymentList>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

At the bottom of the window, the following tabs are visible: Headers (8), Attachments (0), SSL Info, WSS (0), JMS (0).

API REST: <http://localhost:8081/api/payment/payment-date>

REST API SOAP DSW-II / Post FindByPaymentDate

POST <http://localhost:8081/api/payment/payment-date> [Send](#)

Params Auth Headers (9) **Body** Scripts Settings Cookies

raw JSON Beautify

```
1 {
2   "paymentDate": "2024-10-11"
3 }
```

Body 200 OK · 43 ms · 808 B

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "monedas": {
4       "soles": 3313.0,
5       "dolares": 860.52
6     },
7     "montoOriginal": 860.52,
8     "monedaOriginal": "Dólares",
9     "impuestos": 154.89,
10    "valorNeto": 705.63,
11    "fechaVenta": "2024-10-11"
12  },
13  {
14    "monedas": {
15      "soles": 3200.23,
16      "dolares": 831.23
17    },
18    "montoOriginal": 3200.23,
19    "monedaOriginal": "Soles",
20    "impuestos": 576.04,
21    "valorNeto": 2624.19,
22    "fechaVenta": "2024-10-11"
23  },
24  {
25    "monedas": {
26      "soles": 2002.12,
27      "dolares": 520.03
28    },
29    "montoOriginal": 520.03,
30    "monedaOriginal": "Dólares",
31    "impuestos": 93.61,
32    "valorNeto": 426.42,
33    "fechaVenta": "2024-10-11"
34  },
35  {
36    "monedas": {
37      "soles": 1025.99,
38      "dolares": 266.49
39    },
40    "montoOriginal": 1025.99,
41    "monedaOriginal": "Soles",
42    "impuestos": 184.68,
43    "valorNeto": 841.31,
44    "fechaVenta": "2024-10-11"
45  }
46 ]
```