Assignment-4

Date
Page

1) SOAP fault is caused due to client or server failure. state T/F with justification.

→ SOAP provides a model for handling situations when fault arises in the processing of a message.

2) The SOAP fault model requires that all SOAP-specific and application-specific faults be reported using a special-purpose element called env:fault.

3) The env:Fault element is a reserved element predefined by the SOAP specification whose purpose is to provide an extensible mechanism for transporting structured and unstructured information about problems that have arisen during the processing of a SOAP message.

Here is a sample SOAP fault element (fault element marked in bold):

```
<? XML version = "1.0"?>
<env:Envelope xmlns:env = "http // www. w3. org/
2001/12/soap/-envelope">
<env:Body>
<env:Fault>
<env:code>
<env:value> env:sender </env:value>
</env:code>
<env:Reason>
<env:Text xml:lang="en-us"> Processing error</
env:Text>
<env:Text xml:lang="dd"> Processerings-fejl
</env:Text>
</env:Fault>
</env:Body>
</env:Envelope>
```

2) Give the use of SOAP actor attribute.

→ The SOAP actor global attribute can be used to indicate the recipient of a header element the value of the SOAP actor attribute is a URL.

3) What do you meant by Wire Protocol and transport Protocol?

→ 1) Wire Protocol specifies means that SOAP process only two fundamental properties it can send & receive HTTP transport Protocol Packets & process XML Messages.

Wire Protocol specifies the form or shape of data to be exchanged between disparate applications & eventually the term transport protocol signifies the method by which the data is transferred from system to system.

4) What is a SOAP message path?

→ The SOAP message path is the set of SOAP nodes through which a single SOAP message passes including the initial SOAP sender, zero or more SOAP intermediaries & an ultimate SOAP receiver.

5) Give the use of SOAP must understand attribute

→ The must understand attribute means that any node(computer) processing the SOAP message must understand the given header block. A "node" may not may not always be the final receiver of the SOAP message.

the message might be routed via intermediate nodes before ending up at the receiving/processing node (the final Web service).

6) Explain in short Apache Axis environment
→ the use of java for developing SOAP environment applications enables scalable and portable applications to be built that also can interoperate with heterogeneous Applications residing on different platforms by resolving the platform-specific incompatibilities.
- Apache Axis is SOAP toolkit that makes it easy to create, deploy, & consume Web services. By using Axis, we will be able to quickly convert existing java functionality into web services, deploy those services and be able to communicate with them remotely via the internet usually through firewalls.

7) How errors are handled using SOAP faults give an example for adding fault in XML of SOAP message?
→ SOAP provides a model for handling situations when faults arise in the processing of a message. The SOAP <Body> element has another distinguishing role in that it is the place where fault information is placed.
1) the SOAP fault model requires that all SOAP-specific and application-specific faults be reported using a special-purpose element called env: fault.
2) The env: fault element is reversed element

predefined by the SOAP specification whose purpose is to provide an extensible mechanism for transporting structured and unstructured information about problems that have arisen during the processing of a SOAP message.
Here is a sample SOAP fault element (fault element marked in bold)

```
<?xml version="1.0"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/12/soap-envelope">
<env:Body>
<env:Fault>
<env:Code>
<env:Value> env:Sender</env:Value>
</env:Code>
<env:Reason>
<env:Text xml:lang="en-US"> Processing error
</env:Text>
<env:Text xml:lang="da"> Processing - fejl
</env:Text>
</env:Reason>
</env:Fault>
</env:Body>
</env:Envelope>
```

2) What are advantages and disadvantages of SOAP?

→ Advantages -

1) Simplicity - SOAP is simple as it is based on XML, which is highly structured and easy to parse.

2) Portability - SOAP is portable without any dependencies on the underlying platform like byte ordering issue or machine word widths. Today, XML parsers exist for virtually any platform mainframes to write-watch-size devices.

3) firewall-friendly-Posting data over HTTP means not only that the delivery mechanism is widely available but also that is able to get past firewalls that pose problems for other methods.

4) use of open standards - SOAP uses the open standard of XML to format the data, which makes it easily extentable and well supporte -d.

5) universal acceptance - SOAP is the most widely accepted standard in the message communication domain.

~~6) universal acceptance - SOAP is the most widely~~

6) Resilience to changes - changes to SOAP infrastructure will likely not affect applications using the protocol, unless significant serializati on changes are made to the SOAP specification.

disadvantages -

1) SOAP was initially tied to HTTP and this mandate a request/response architecture that was not appropriate for all situations. HTTP is a relatively slow protocol and of course the performance of SOAP

suffered. The latest version of the SOAP specificati loosens this requirement.

2) SOAP is stateless. The stateless nature of SOAP requires that the requesting application must reintroduce itself to other applications when more connections are required as if it had never been connected before

3) What is SOAP? Give the structure of SOAP message, explain it.

→ i SOAP is simply an attempt to create the usage of existing Internet technologies to standardized distributed communication over the web, rather than being a new technological advancement.

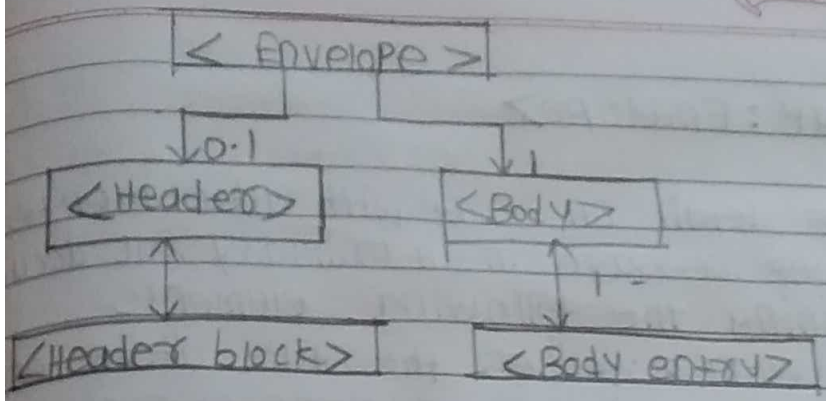
A SOAP message is an ordinary XML containing following elements.

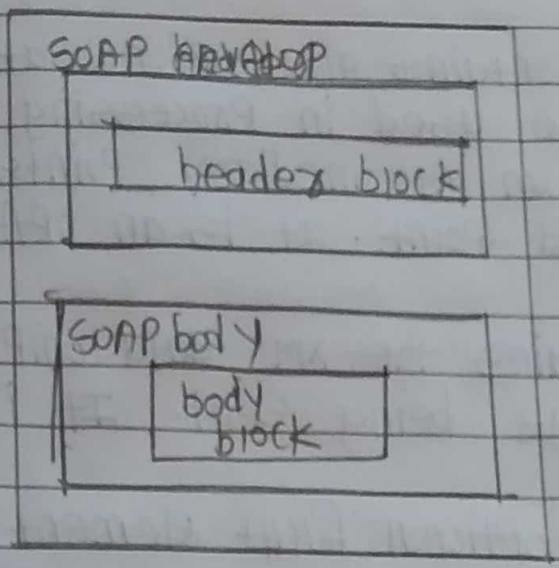
1. Envelope - Defines the start and the end of the message. It is a mandatory element.

2. Header - contain any optional attributes of the message used in processing the message, either at an intermediary point or at the ultimate point. It is an optional element.

iii) Body - certain the XML data comprising the message being sent. It is a mandatory ele

iv) fault - An optional fault element that provid information about errors that occur while processing the message.





Here is a simple sample SOAP message.

```
<?xml version="1.0"?>
<SOAP:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope">
  <soap:Header>
  </soap:Header>
  <soap:Body>
  </fault element is optional, used only if a fault occurs in web service.
  -->
  <soap:Fault>
  </soap:Fault>
  </soap:Body>
```

<Soap:Envelope>

4) Write detail on SOAP with attachments.

→ A SOAP message in an ordinary XML document containing the following elements.

I) Envelope: defines the start & the end of message. It is a mandatory element.

II) Header - Contains any optional attributes of the message used in processing the messa either at an intermediary points or at the ultimate end-point. It is an optional element

III) Body - Contains the XML data comprising with the message being sent. It is mandato element

IV) Fault - An optional fault element that provi information about errors that occur while processing the message.

5)

Give an example XML code snipet for erro handling in SOAP using fault element also exp it

→ SOAP provides a model for handling situations whe faults arise in the processing of a messa The SOAP <Body> element has another distinguishing role in that it is place wh fault information is placed.

I) The SOAP fault model requires that all SOAP specific and application-specific faults be reported using a special purpose element

called env:Fault.

D) The env:Fault element is a reversed element predefined by the SOAP specification whose purpose is to provide an extensible mechanism for transporting structured and unstructured about problems that have arisen during the processing of a SOAP message

6) write an example of document styled SOAP body.

→ The SOAP body is the area of the SOAP message where the application-specific XML data being exchanged in the message is placed. The <Body> element must be present & must be an immediate child of the develope.
The SOAP Body element is the element in a SOAP message that contains the main part to be processed by either client or web service. While a Header element is optional, a Body element is mandatory. You MUST have a Body element in a SOAP message.
Here is a sample SOAP Body element (Body element marked in bold):

```
<?xml version="1.0"?>
<env:Envelope xmlns:env="http://www.w3.org/2011/12/soap-envelope">
<env:Header>
</env:Header>
<env:Body>
</env:Body>
</env:Envelope>
```

The body of a SOAP message can consist of pretty

much whatever XML you feel like putting in there as long as it is valid. However you cannot put text inside the Body element. Text should be nested inside child element of the Body element.

7) Draw the structure of SOAP with attachement message, give an example & explain it.

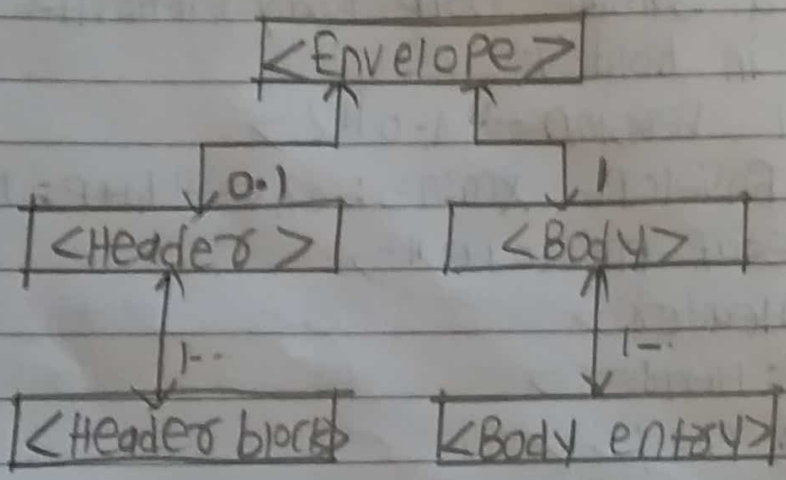
→ A SOAP message is an ordinary XML document containing the following element.

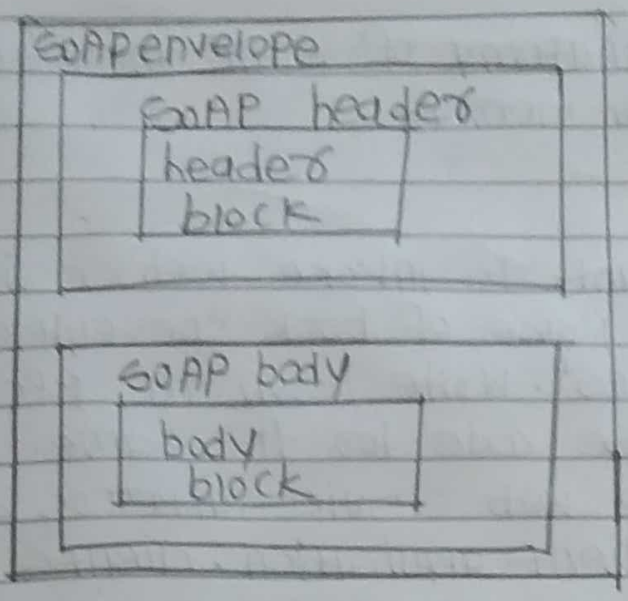
i) Envelope - Defines the start and the end of the message it is mandatory element.

ii) Header - contains any optional attributes of the message used in processing the message, either at an intermediary point or at the ultimate end-point. It is an optional element.

iii) Body - contains the XML data comprising the message being sent. It is a mandatory element.

iv) Fault - An optional Fault element that provides information about errors that occur while processing the message.





Here is a simple SOAP message:

```
<?xml version="1.0" ?>
<Soap : Envelope
xmlns : soap="http://www.w3.org/2001/12/soap-envelope">
<Soap: Header>
</Soap: Header>
<Soap : Body>
<!-- fault element is optional.
used only if a fault occurs in web service
-->
<Soap: fault>
</Soap: Fault>
</Soap: Body>
</Soap : Envelope>
```

8) Write the anatomy of SOAP message and describe each element.

→

8) A client want to invoke web service for requesting price of book "developing Java Web services". Write down the RPC request and response code for the same.

→ An RPC-style web service appears as a remote object to a client application. clients express this request as a method call with a set of arguments which return a response containing a return value. these are represented as sets of XML elements embedded within a SOAP message.

i) RPC style supports automatic serialization /deserialization of messages, permitting developers to express a request as a method call with a set of parameters. which returns a response containing a return value.

ii) Because of this type of bilateral communication between the client and web service, RPC style web service require a tightly coupled (synchronous) model of communication between the client and service provider.

eg - SOAP <Body> specification. It illustrates a price quote service that requests the Product Price be quote service that requests the product price be associated with a specified Plastic Product. As shown from this listning

the SOAP <Body> element contains the actual method call.
Note the existence of the message namespace qualifiers which is the part of the information needed for the method call to be successful. the namespace identifies the URL of the target object. In addition, the method call requires the method name (GetProdPrice) & the the Parameters (Prod-Id)

```
<env-Envelope
  XMLNS: SOAP="http://www.w3.org/2003/05/
SOAP-envelope"
XMLNS:m="http://www.Plastics_supply.com/Prod-
Prices">
<env-Header>
<tx: Transaction-id
XMLNS:t="http://www.transaction.com/transaction".
 env: mustUnderstand='1'>
  512
</tx: Transaction-id>
</env: Header>
 <env: Body>
  <m: Get-Prod Price>
<Prod-id> 450R60P </Prod-id>
</m: GetProdPrice>
</env: Body>
  </env: Envelope>
```