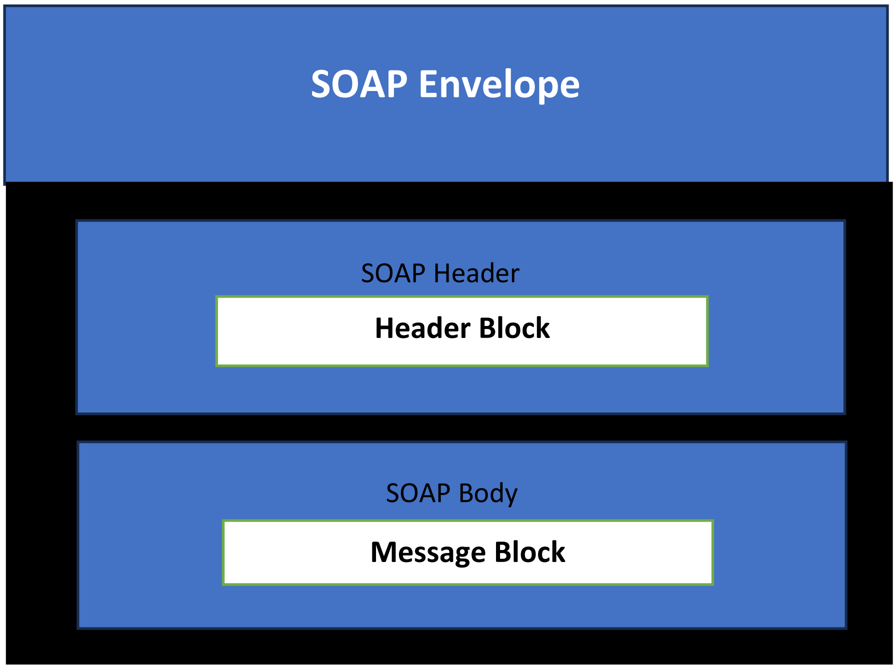
CC Lab 3: Understanding SOAP web services

Web-Service: A service that is made available over the web.

Difference between a website and a webservice is that a website is meant and designed for human interaction/consumption whereas a webservice is meant for code/application-level consumption.

The SOAP message is nothing but a mere XML document which has a structure as shown below:



Here the SOAP Envelope identifies the message as a SOAP message, while the header contains information such as authentication credentials, definition of complex types which could be used in the SOAP message.

A simple SOAP service example of a complex type will be as follows for a structured data type which has a combination of a “Name” and a “Description”.

```
<xsd:complexType>
 <xsd:sequence>
      <xsd:element name="Name" type="string"/>
      <xsd:element name="Description"  type="string"/>
  </xsd:sequence>
</xsd:complexType>
```

A body element will have call and response information. It will contain the actual data being communicated. A response example will be as follows:

```
<soap:Body>
   <GetTutorialInfo>
      <Name>Web Services</Name>
      <Description>All about web services </Description>
   </GetTutorialInfo>
</soap:Body>
```

SOAP Message Structure

SOAP messages are auto-generated by the web service when invoked. A simple SOAP Message has the following elements –

- The Envelope element
- The header element and
- The body element
- The Fault element (Optional)

Following is an example of what overall response message would looks like:

```
<?xml version=”1.0” encoding=”utf-8”?>
<soap:Envelope xmlns:xsi= ”http://www.w3.org/2001/
XMLSchema-instance”>

      <soap:Body>
      <GetTutorialInfo xmlns=”http://example.org/”>
            <ID>int</ID>
      </GetTutorialInfo>
      </soap:Body>
</soap:Envelope>
```

As you can see from the above SOAP response message, our web service “GetTutorialInfo” needs an input “ID” of type “int” to serve us.

It is important to note that if the SOAP message contains a header block it must appear as a 1st child of the root element, i.e., SOAP Envelope.

Note: A v1.1-compliant SOAP processor generates a fault upon receiving a message containing the v1.2 envelope namespace. And a v1.2-compliant SOAP processor generates a Version Mismatch fault if it receives a message that does not include the v1.2 envelope namespace.

Below is a SOAP API message example of version 1.2 compliant SOAP envelope element.

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV= "http://www.w3.org/2001/12/soap-
envelope" SOAP-ENV:encodingStyle="http://www.w3.org/2001/12/soap-
encoding">
        <soap:Body>
        <GetTutorialInfo xmlns="http://example.org/">
                  <ID>int</ID>
        </GetTutorialInfo>
        </soap:Body>
</SOAP-ENV:Envelope>
```

The Fault message

The SOAP Fault message consists of the following elements:

- **<faultCode> -** It defines the type of error thrown. It can be as follows:
    - SOAP-ENV:VersionMismatch – This is when an invalid namespace for the SOAP Envelope element is encountered.
    - SOAP-ENV:MustUnderstand – An immediate child element of the Header element, with the mustUnderstand attribute set to “1”, was not understood.
    - SOAP-ENV:Client – The message was incorrectly formed or contained incorrect information.
    - SOAP-ENV:Server – There was a problem with the server, so the message could not proceed.
- **<faultString>** – This is the text message which gives a detailed description of the error.
- **<faultActor> (Optional)**– This is a text string which indicates who caused the fault.
- **<detail>(Optional)** – This is the element for application-specific error messages.

Following is an example of a fault message when we call a method that does not exist in a web service.

```
<?xml version='1.0' encoding='UTF-8'?>
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
      <SOAP-ENV:Body>
         <SOAP-ENV:Fault>
             <faultcode xsi:type="xsd:string">SOAP-ENV:Client
</faultcode>
             <faultstring xsi:type="xsd:string">
            Failed to locate method (GetTutorialID) in class
(GetTutorial)
         </faultstring>
    </SOAP-ENV:Fault>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```