# Assignment 1.1 - Web Services (WDSL/SOAP)

## Web services and Cloud-based systems - 2020

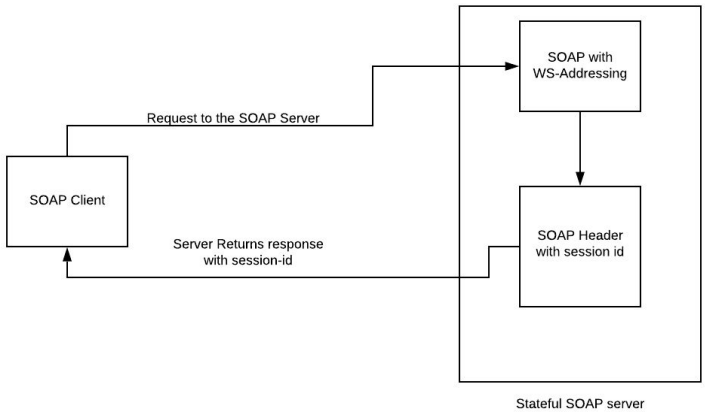
**Done by,**
**Haritha Jayaraman - 12975052**
**Kailainathan Muthiah Kasinathan - 12937827**
**Richard Bieringa -10691065**

## Design & Implementation of a calculator using Soap:

Bottom-Up Approach was followed when implementing the Web Service. It is a bottom-up approach as the code was written first and then WSDL was generated from it. The reason to do so is it was relatively easy as the code for Calculator in Python is pretty easy in comparison to generating a WSDL file(As I am not familiar with creating one). Implementation was executed using Python programming language.The server side was first set up. Ladon was used for this purpose. The file calculator.py containing the code for the operations in a calculator was written. Next the runserver.py contains the code to start the server. First the required modules were imported and then the directory name of the absolute path was stored as it should be known to call the calculator.py service. The service module is then created with the same name as the service which is a calculator in our case. Then a Web Service Gateway Interface( WSGI) is used for running python Web Applications. The WSGI Application can be invoked by the WSGI server which is also subsequently created. Thus a stand-alone server on port 8080 is created by starting the server. Next, the client side implementation is done. This is executed using Zeep. Once the server is up and running it generates a SOAP11 url and description.The description consists of the url for the WSDL file through which it can be accessed. This URL is copied and included in the client file, so the client can know the Web Service Description of the client and hence can be used to invoke the methods of the Calculator Application. So here the client invokes the four operations of the Calculator Application namely add, sub, mul, div. Now the Web Service is completely implemented. By using the command line , run the client python code using python3 client.py .

## Making Soap Stateful:

Soap web services can be made stateful by using WS-Addressing. WS-Addressing is the standard way in which one can include the message routing data into the SOAP Headers. Basically the session-id can be stored in the SOAP Header this way. The process starts with the SOAP client sending a request . Upon request the SOAP session will be




initiated by the SOAP Server. The response from SOAP server to the client will be a header with session id. This makes SOAP stateful.