Implementing a SOAP Client in C# using Visual Studio 2008

Create a new project

In Visual studio 2008, select File → New. In the new projects dialog navigate to Visual c#, then under windows select Empty Project as shown below: (figure 1)

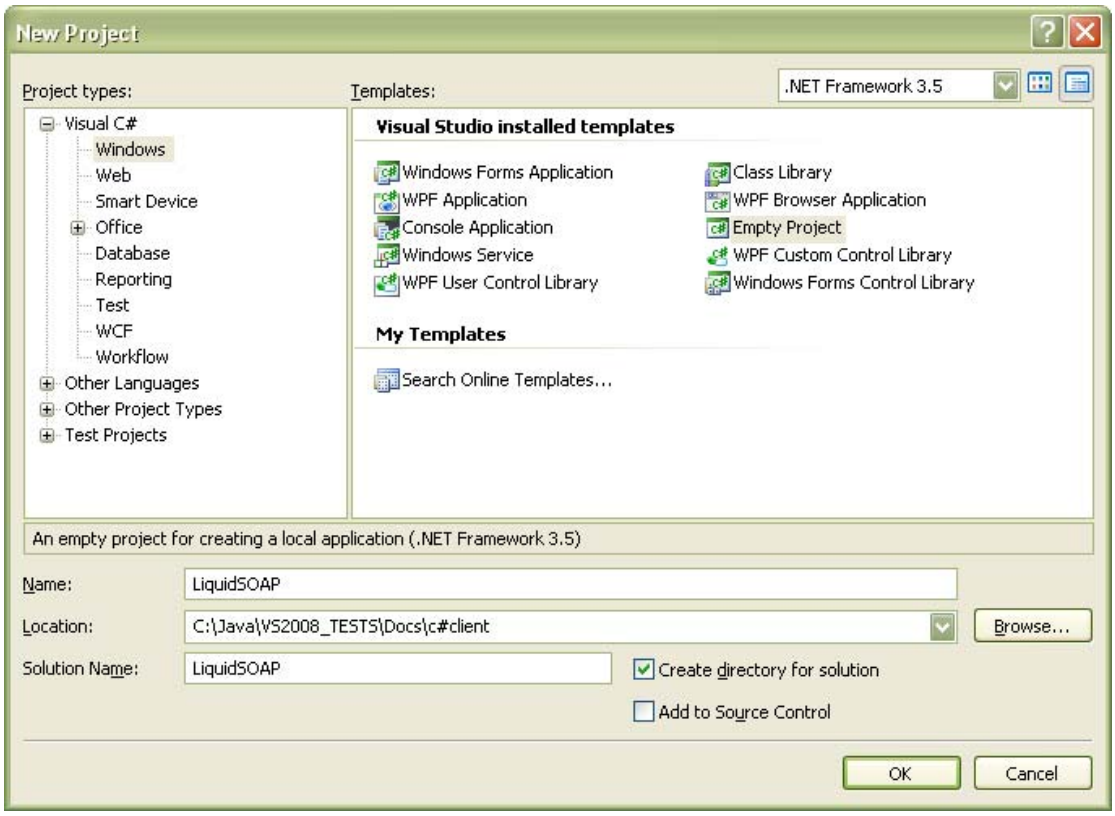

Figure 1

Adding a Service reference

On the solutions explorer, right click on Project name and select Add Service Reference.

On the Service reference dialog, give the following WSDL as the address.(figure 2)

https://soap.oventus.com/LiquidWS/MessageService?WSDL

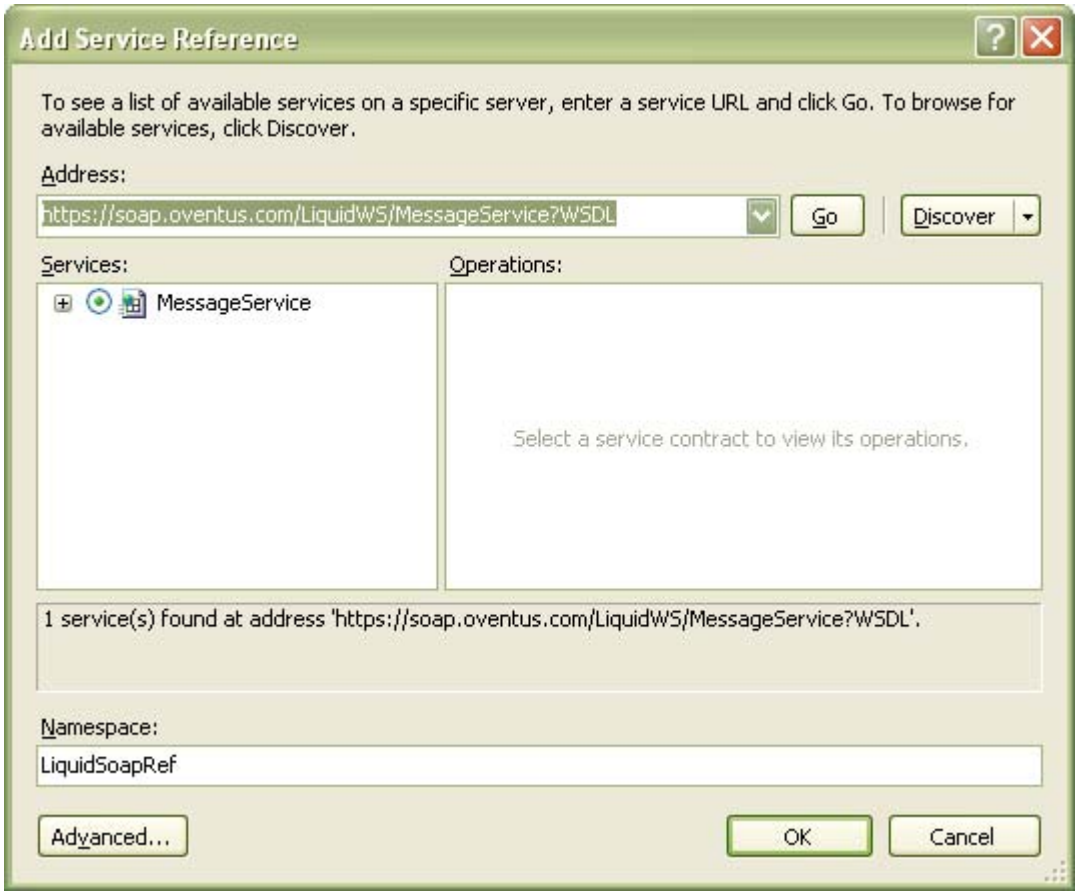

Figure 2

Click on OK. This will generate the required Stubs .

Implement the clients

Right click on project in solutions explorer and add new item (class)
LiquidSOAPImpl.cs

Example- implement login method

```
MessageServicePortTypeClient messageServicePortType = new
LiquidSOAP.LiquidSoapRef.MessageServicePortTypeClient();

pageoneHeader p1Header = null;

loginRequest loginReq = new loginRequest();

loginReq.userid = "username";
loginReq.pwd = "password";

loginResponse loginResponse = new loginResponse();
p1Header = messageServicePortType.login(loginReq, out loginResponse);
```

```
String sessionID = p1Header.sessionid;
statusType statusType = loginResponse.status;
```

Please refer to Appendix A for full implementation of this class.

Appendix A

```
using System;
using System.Collections.Generic;
using System.Text;
using LiquidSOAP.LiquidSoapRef;

namespace LiquidSOAP
{

    class LiquidSOAPImpl
    {
        MessageServicePortTypeClient messageServicePortType = new
LiquidSOAP.LiquidSoapRef.MessageServicePortTypeClient();
        pageoneHeader p1Header = null;

        public pageoneHeader getSessionID()
        {
            try
            {
                if (p1Header == null)
                {
                    loginRequest loginReq = new loginRequest();

                    //PLEASE ENTER A VALID USERNAME AND PASSWORD HERE
                    loginReq.userid = "username";
                    loginReq.pwd = "password";

                    loginResponse loginResponse = new
loginResponse();
                    p1Header = messageServicePortType.login(loginReq,
out loginResponse);

                    String sessionID = p1Header.sessionid;

                    statusType statusType = loginResponse.status;
                    Console.WriteLine(" Status Description is >>: " +
statusType.description);
                    Console.WriteLine(" Session ID is >>:  " +
sessionID);
                }
                return p1Header;
            }
            catch (Exception ex)
            {
                Console.WriteLine("\n\n\n\nPlease check if you have
entered a valid username and password in execute.cs");
                return null;
            }

        }

        public void logOff()
        {
            logoffRequest logOffReq = new logoffRequest();
            logoffResponse logOffResp =
messageServicePortType.logoff(this.getSessionID(), logOffReq);
            Console.WriteLine("Logoff result " + logOffResp.result);
            Console.WriteLine("\n\npress enter to end");
            Console.Read();
```

```
        }

        public void ping()
        {
            pingRequest pingReq = new pingRequest();
            pingResponse pingResp
=messageServicePortType.ping(this.getSessionID(),pingReq);
            Console.WriteLine("Ping result " + pingResp.description +
" value " + pingResp.Value);
            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void isActive()
        {
            isActiveRequest isActiveReq = new isActiveRequest();
            isActiveReq.userid = "username";
            isActiveResponse isActiveResp =
messageServicePortType.isActive(isActiveReq);

            Console.WriteLine("Is Active result " +
isActiveResp.description + " value " + isActiveResp.Value);
            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void getReceipt()
        {
            getReceiptRequest getReceiptReq = new
getReceiptRequest();
            receiptType receiptType =
messageServicePortType.getReceipt(this.getSessionID(),
getReceiptReq);

            Console.WriteLine("Destination "+
receiptType.destination);
            Console.WriteLine("resultCode "+receiptType.resultCode);
            Console.WriteLine("source  "+receiptType.source);
            Console.WriteLine("sentAt " +
receiptType.sentAt.ToString());
            Console.WriteLine("\n\npress enter to end");
            Console.Read();
         }


        public void getReceipts()
        {
            getReceiptsRequest getReceiptsReq = new
getReceiptsRequest();
            receiptType[] receiptType =
messageServicePortType.getReceipts(this.getSessionID(),
getReceiptsReq);

            for (int i = 0; i < receiptType.Length; i++)
            {
                Console.WriteLine("Destination " +
receiptType[i].destination);
                Console.WriteLine("resultCode " +
receiptType[i].resultCode);
```

```
                Console.WriteLine("source  " +
receiptType[i].source);
                Console.WriteLine("sentAt " +
receiptType[i].sentAt.ToString());
                Console.WriteLine("=====");
            }

            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void getInbound()
        {
            getInboundRequest getInboundReq = new
getInboundRequest();
            inboundType inboundType =
messageServicePortType.getInbound(this.getSessionID(),
getInboundReq);

            Console.WriteLine(" destination " +
inboundType.destination);
            Console.WriteLine(" message " + inboundType.message);
            Console.WriteLine(" messageID " + inboundType.messageID);
            Console.WriteLine(" source " + inboundType.source);
            Console.WriteLine(" timeReceived " +
inboundType.timeReceived.ToString());

            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void getInbounds()
        {
            getInboundsRequest getInboundsReq = new
getInboundsRequest();

            inboundType[] inboundType =
messageServicePortType.getInbounds(this.getSessionID(),
getInboundsReq);
            for (int i = 0; i < inboundType.Length; i++)
            {
                Console.WriteLine("destination " +
inboundType[i].destination);
                Console.WriteLine("message " +
inboundType[i].message);
                Console.WriteLine("messageID" +
inboundType[i].messageID);
                Console.WriteLine("source " + inboundType[i].source);
                Console.WriteLine("time received " +
inboundType[i].timeReceived.ToString());
                Console.WriteLine("====");
            }

            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void getMsisdn()
```

```
        {
            getMsisdnRequest getMsisdnReq = new getMsisdnRequest();
            string[] msisdns
=messageServicePortType.getMsisdn(this.getSessionID(), getMsisdnReq);
            for (int i = 0; i < msisdns.Length; i++)
            {
                Console.WriteLine("Msiddn >> " + msisdns[i]);
            }
            Console.WriteLine("\n\npress enter to end");
            Console.Read();
      }


        public void sendMessage()
        {
            string[] addrArr = {"mobile number here"};
            sendMessageType sendMsgType = new sendMessageType();
            sendMsgType.message = "Hello";
            sendMsgType.destinationAddress = addrArr;
            sendMessageResponseType sendMsgRespType =
messageServicePortType.sendMessage(this.getSessionID(), sendMsgType);
            Console.WriteLine("SendMessage Response status " +
sendMsgRespType.status.description);
            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void sendMultiMessage()
        {
            sendMultiMessageType[] sendMultiReqType = new
sendMultiMessageType[2];

            sendMultiMessageType sendType1 = new
sendMultiMessageType();

            string[] addrArr1 = {"mobile address"};
            sendType1.destinationAddress = addrArr1;
            sendType1.flashSMS = false;
            sendType1.message = "first message";
            sendType1.requestID = 1;


            sendMultiMessageType sendType2 = new
sendMultiMessageType();
            string[] addrArr2 = { "mobile address" };
            sendType2.destinationAddress = addrArr2;
            sendType2.flashSMS = true;
            sendType2.message = "second message";
            sendType2.requestID = 2;

            sendMultiReqType[0] = sendType1;
            sendMultiReqType[1] = sendType2;


            sendMultiMessageResponseMessagePacketResponse[] multiResp
= messageServicePortType.sendMultiMessage(this.getSessionID(),
sendMultiReqType);

            for (int i = 0; i < multiResp.Length; i++)
            {
```

```
                Console.WriteLine("description
"+multiResp[i].status.description);
                Console.WriteLine("value
"+multiResp[i].status.Value);
                Console.WriteLine("transactionID " +
multiResp[i].transactionID);
                Console.WriteLine("requestID" +
multiResp[i].requestID);
            }

            Console.WriteLine("\n\npress enter to end");
            Console.Read();


        }

        public void pushWapRequest()
        {
            pushWapType pushWapType = new pushWapType();
            //pushWapType.sourceAddress ="";

            string[] addrArr = { "mobile number here" };
            pushWapType.destinationAddress = addrArr;
            pushWapType.title = "Test";
            pushWapType.url ="http://www.google.com";
            //pushWapType.expireAt

            sendMessageResponseType sendMessageResponseType  =
messageServicePortType.pushWap(this.getSessionID(), pushWapType);

            Console.WriteLine(" description " +
sendMessageResponseType.status.description);
            Console.WriteLine(" Value " +
sendMessageResponseType.status.Value);
            Console.WriteLine(" transactionID " +
sendMessageResponseType.transactionID);

            Console.WriteLine("\n\npress enter to end");
            Console.Read();

       }

        public void pushMultiWapRequest()
        {
            pushMultiWapType[] pushMltiWapType = new
pushMultiWapType[2];

            pushMultiWapType pushMultiType1 = new pushMultiWapType();

            string[] addrArr1 = { "mobile number here" };
            pushMultiType1.destinationAddress = addrArr1;
            pushMultiType1.title = "Test";
            pushMultiType1.url = "http://www.google.com";
            pushMultiType1.requestID = 1;
            pushMultiType1.requestIDSpecified = true;

            //pushMultiType1.expireAt

            pushMultiWapType pushMultiType2 = new pushMultiWapType();
            string[] addrArr2 = { "mobile number here" };
            pushMultiType2.destinationAddress = addrArr2;
```

```
            pushMultiType2.title = "Test";
            pushMultiType2.url = "http://www.google.com";
            pushMultiType2.requestID = 2;
            pushMultiType2.requestIDSpecified = true;
            //pushMultiType2.expireAt

            pushMltiWapType[0] = pushMultiType1;
            pushMltiWapType[1] = pushMultiType2;

            pushMultiWapResponseMessagePacketResponse[] resp =
messageServicePortType.pushMultiWap(this.getSessionID(),
pushMltiWapType);

            for (int i = 0; i < resp.Length; i++)
            {
                Console.WriteLine("requestID "+ resp[i].requestID);
                Console.WriteLine("transactionID " +
resp[i].transactionID);
                Console.WriteLine("description
"+resp[i].status.description);
                Console.WriteLine("value "+resp[i].status.Value);
            }

            Console.WriteLine("\n\npress enter to end");
            Console.Read();

        }

        public void getCredits()
        {
            try
            {
                getCreditsRequest getCreditsReq = new
getCreditsRequest();
                getCreditsResponse getCreditsResp =
messageServicePortType.getCredits(this.getSessionID(),
getCreditsReq);
                Console.WriteLine("Credits Remaining " +
getCreditsResp.creditsRemaining);
                Console.WriteLine("\n\npress enter to end");
                Console.Read();
            }
            catch (Exception ex)
            {
                Console.WriteLine("\n\nException caught press enter
to end\n");
                Console.Read();
            }

        }



    }
}
```

Create an Execute class to access the Services

```
using System;
using System.Collections.Generic;
using System.Text;

namespace LiquidSOAP
{
   class Execute
   {

      public static void Main()
      {
         //SOAP MESSAGE SERVICE
         LiquidSOAPImpl execute = new LiquidSOAPImpl();

          execute.sendMessage();
          execute.isActive();
          execute.ping();
          execute.getCredits();
          execute.getMsisdn();

       }
   }
}
```