

PR̄MIUM



**EnviroChain[®]
Lab Web Service
API Documentation**

Contents

Introduction	1
EnviroChain® Methods	1
Lab Web Service	1
API Documentation	1
Introduction	3
EnviroChain Methods	3
1. GetAutoLogExcelCoC	3
2. GetPhotosOfCOCID	7
3. GetSignatureOfCOCID	12
4. LabAuthenticate	15
5. LabGetCoCID	18
4. LabGetCoCIDDetails	21

Introduction

This EnviroChain documentation is organized by methods.

Urls for web service of EnviroChain Lab Production is as below:

<http://api.envirochain.com:81/Service.asmx>

EnviroChain Methods

EnviroChain methods discussed in this document are:

- [GetAutoLogExcelCoC](#)
- [GetCoCPDF](#)
- [GetPhotosOfCOCID](#)
- [GetSignatureOfCOCID](#)
- [LabAuthenticate](#)
- [LabGetCoCID](#)
- [LabGetCoCIDDetails](#)

Each method is explained in more detail in its own section.

1. GetAutoLogExcelCoC

This method will return all CoCs within specifies date range submitted to the lab in zip format. One zip file will contain multiple CoCs in each laboratory's specified EDD format.

In the GetAutoLogExcelCoC method, input parameters will be TokenID, StartDate, and EndDate.

The date input parameters will have a format of mm/dd/yyyy.

Output will be zip file in Base64bit string for file data, error code, and error messages.

Users will have to convert Base64bit to binary then need to write zip file.

TokenID is retrieved from the LabAuthenticate method.

To test, paste the output of base64 into the text area in the website below.

<http://www.motobit.com/util/base64-decoder-encoder.asp>

Select decode functionality for Source data and output data as export to binary file. Provide filename with .zip as extension.

Input Parameters:

- TokenID
- StartDate
- EndDate

Output Parameters:

- ErrorCode
- ErrorMessage
- FileData

Note: Both input and output parameters are case sensitive.

Successful Response

```
<errorCode>000</errorCode>
<errorMessage>Successful.</errorMessage>
<fileData>...</fileData>
```

<fileData> will contain Base64 string data of zip file.

Production Demo deployed in the following location:

<http://api.envirochain.com:81/CoCAutoLogExcel.aspx>

Request

```
TokenId=string&StartDate=string&EndDate=string
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<GetResponse xmlns="http://api.envirochain.com/">
  <errorCode>string</errorCode>
  <errorMessage>string</errorMessage>
  <fileData>string</fileData>
</GetResponse>
```

Send all input parameters to by POST method.

Sample .net Web Client Code:

```
try
{
    HttpWebRequest httpReq =
    (HttpWebRequest)WebRequest.Create("http://api.envirochain:81.com/EnviroChain
    Lab/Service.asmx/GetAutoLogExcelCoC");

    ASCIIEncoding encoding = new ASCIIEncoding();

    string strparam = "TokenId=" + txtTokenID.Text.Trim() + "&StartDate=" +
    txtStartDate.Text.Trim() + "&EndDate=" + txtEndDate.Text.Trim() + "";

    byte[] postData = encoding.GetBytes(strparam);

    httpReq.ContentType = "application/x-www-form-urlencoded";
    httpReq.Method = "POST";
    httpReq.ContentLength = postData.Length;

    // convert the request to a stream object and send it on its way
    Stream ReqStrm = httpReq.GetRequestStream();
    ReqStrm.Write(postData, 0, postData.Length);
    ReqStrm.Close();

    // get the response from the web server and
    // read it all back into a string variable
    DataSet dsWeather = new DataSet();
    byte[] binaryData = null;
    using (HttpWebResponse response = httpReq.GetResponse() as
    HttpWebResponse)
    {
        // Load data into a dataset

        dsWeather.ReadXml(response.GetResponseStream());
        if (dsWeather.Tables[0].Rows.Count > 0)
        {
            string bytedata =
            dsWeather.Tables[0].Rows[0]["fileData"].ToString();
            if (bytedata.Length > 0)
            {
                binaryData =
                System.Convert.FromBase64String(dsWeather.Tables[0].Rows[0]["fileData"].ToSt
                ring());

                String FileName = "EnvironChainLab.zip"; ;
            }
        }
    }
}
```


2. GetCoCPDF

This method will return a PDF version of the CoC submitted to the lab.

Input Parameters:

TokenID
CoCID

Output Parameters:

errorCode
errorMessage
fileData

Note: Both input and output parameters are case sensitive

Successful Response

```
<errorCode>000</errorCode>  
<errorMessage>Successfull.</errorMessage>  
<fileData>...</fileData>
```

TokenID is retrieved from the LabAuthenticate method. CoCID is retrieved from the LabGetCoCID method.

Output will be base64string in fileData tag.

<fileData> will contain Base64 string data of zip file.

Note: This will need to be converted into a PDF file, as shown in the Sample .net Web Client Code below.

Production demo deployed in the following location:

<http://api.envirochain.com:81/CoCPDF.aspx>

Sample .net Web Client Code:

```
try
    {
        HttpWebRequest httpReq =
        (HttpWebRequest)WebRequest.Create("http://api.envirochain.com:81/Service.asmx" +
        "\\GetCoCPDF");

        ASCIIEncoding encoding = new ASCIIEncoding();
        string strparam = "TokenId=" + txtTokenID.Text.Trim() + "&CoCID=" +
        txtCoCID.Text.Trim() + "";
        byte[] postData = encoding.GetBytes(strparam);

        httpReq.ContentType = "application/x-www-form-urlencoded";
        httpReq.Method = "POST";
        httpReq.ContentLength = postData.Length;
        httpReq.Timeout = Timeout.Infinite;
        //httpReq.UserAgent = "api.envirochain.com";
        httpReq.KeepAlive = true;

        // convert the request to a stream object and send it on its way
        Stream ReqStrm = httpReq.GetRequestStream();
        ReqStrm.Write(postData, 0, postData.Length);
        ReqStrm.Close();
        // Thread.Sleep(540000);
        // get the response from the web server and
        // read it all back into a string variable
        DataSet dsWeather = new DataSet();
        byte[] binaryData = null;

        using (HttpWebResponse response = httpReq.GetResponse() as
        HttpWebResponse)
        {
            // Load data into a dataset

            dsWeather.ReadXml(response.GetResponseStream());
            if (dsWeather.Tables[0].Rows.Count > 0)
            {
                string bytedata =
                dsWeather.Tables[0].Rows[0]["fileData"].ToString();
                if (bytedata.Length > 0)
                {
                    binaryData =
                    System.Convert.FromBase64String(dsWeather.Tables[0].Rows[0]["fileData"].ToString());
                }

                String FileName = txtCoCID.Text.Trim()+".pdf"; ;
                String FilePath = AppDomain.CurrentDomain.BaseDirectory
            }
        }
    }
}
```



```
        string filenameex = FilePath + FileName;
        FileStream fsx = new FileStream(filenameex,
        FileMode.Create, FileAccess.Write);

        ////Read block of bytes from stream into the byte array
        fsx.Write(binaryData, 0, binaryData.Length);

        ////Close the File Stream
        fsx.Close();

        System.Web.HttpResponse response1 =
        System.Web.HttpContext.Current.Response;
        response1.ClearContent();
        response1.Clear();
        response1.ContentType = "text/pdf";
        response1.AddHeader("Content-Disposition", "attachment;
        filename=" + FileName + ";");
        response1.TransmitFile(FilePath + FileName);
        response1.Flush();
        response1.End();
    }
    else
    {

        Response.Write(dsWeather.Tables[0].Rows[0]["errorMessage"].ToString());
    }
    else
    {
        Response.Write("No Files are Generated");
    }
}
}
catch (WebException ex)
{
    if (ex.Status == WebExceptionStatus.ProtocolError)
    {
        Console.WriteLine("Status Code : {0}",
        ((HttpWebResponse)ex.Response).StatusCode);
        Console.WriteLine("Status Description : {0}",
        ((HttpWebResponse)ex.Response).StatusDescription);
    }
    Logger.WriteError(ex.Message);
    Response.Write(ex.Message);
}
}
```

3. GetPhotosOfCoCID

This method will return zip file of photos for a specified CoC.

Input Parameters:

- TokenID
- CoCID
- ErrorCode
- ErrorMessage
- FileData

Output Parameters:

Production demo deployed in the following location:

<http://api.envirochain.com:81/CoCSignature.aspx>

Request

```
TokenId=string&CoCID=string
```

Response

```
<?xml version="1.0" encoding="utf-8"?>  
<GetPhotoResponse xmlns="http://api.envirochain.com/">  
  <errorCode>string</errorCode>  
  <errorMessage>string</errorMessage>  
  <fileData>string</fileData>  
</GetPhotoResponse>
```

Note: A successful response will only be for PaidCoC.

Send all input parameters to by POST method.

Sample .net Web Client Code:

```
try  
{
```

```

    HttpWebRequest httpReq =
    (HttpWebRequest)WebRequest.Create("http://api.envirochain.com/EnviroChainLabProd/Service.aspx/GetPhotosOfCOCID");

    ASCIIEncoding encoding = new ASCIIEncoding();
    string strparam = "TokenId=" + txtTokenID.Text.Trim() + "&CoCID=" +
    txtCoC.Text.Trim() + "";
    byte[] postData = encoding.GetBytes(strparam);

    httpReq.ContentType = "application/x-www-form-urlencoded";
    httpReq.Method = "POST";
    httpReq.ContentLength = postData.Length;

    // convert the request to a stream object and send it on its way
    Stream ReqStrm = httpReq.GetRequestStream();
    ReqStrm.Write(postData, 0, postData.Length);
    ReqStrm.Close();

    // get the response from the web server and
    // read it all back into a string variable
    DataSet dsWeather = new DataSet();
    byte[] binaryData = null;
    using (HttpWebResponse response = httpReq.GetResponse() as HttpWebResponse)
    {
        // Load data into a dataset

        dsWeather.ReadXml(response.GetResponseStream());
        if (dsWeather.Tables[0].Rows.Count > 0)
        {
            string bytedata =
            dsWeather.Tables[0].Rows[0]["fileData"].ToString();
            if (bytedata.Length > 0)
            {
                binaryData =
                System.Convert.FromBase64String(dsWeather.Tables[0].Rows[0]["fileData"].ToString());

                String FileName = "EnvironChainPhoto.zip"; ;
                String FilePath = AppDomain.CurrentDomain.BaseDirectory; ;

                string filenameex = FilePath + FileName;
                FileStream fsx = new FileStream(filenameex, FileMode.Create,
                FileAccess.Write);

                ////Read block of bytes from stream into the byte array
                fsx.Write(binaryData, 0, binaryData.Length);

                ////Close the File Stream
                fsx.Close();

                System.Web.HttpResponse response1 =
                System.Web.HttpContext.Current.Response;
                response1.ClearContent();
                response1.Clear();
                response1.ContentType = "text/plain";
            }
        }
    }

```

```
        response1.AddHeader("Content-Disposition", "attachment;
filename=" + FileName + ";");
        response1.TransmitFile(FilePath + FileName);
        response1.Flush();
        // response1.End();
    }
    else
    {
        Response.Write(
dsWeather.Tables[0].Rows[0]["errorMessage"].ToString());
    }
    else
    {
        Response.Write("Photo Files not Generated");
    }
}
}
catch(Exception ex)
{
    Logger.WriteError(ex.Message);
    Response.Write(ex.Message);
}
```

4. GetSignatureOfCoCID

This method will return zip of signature images of specific CoCID.

Input Parameters:

- TokenID
- CoCID

Output Parameters:

- ErrorCode
- ErrorMessage
- FileData
- SignatureDetails

Signature details Contains:

SignatureID	SignatureDateTime
RelinquishedName	IPAddress
ReceivedName	SamplerSignatureName
RelinquishedCompany	SamplerSignature
ReceivedCompany	

Production demo deployed in the following location:

<http://api.envirochain.com:81/CoCPhoto.aspx>

Request

```
TokenId=string&CoCID=string
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<GetSignatureResponse xmlns="http://api.envirochain.com/">
  <errorCode>string</errorCode>
  <errorMessage>string</errorMessage>
  <signatureDetails>string</signatureDetails>
  <fileData>string</fileData>
</GetSignatureResponse
```

Note: A successful response will only be for PaidCoC.

Send all input parameters to by POST method.

Sample .net Web Client Code:

```
try
{
    HttpWebRequest httpReq =
    (HttpWebRequest)WebRequest.Create("http://api.envirochain.com:81
/Service.asmx/GetSignatureOfCOCID");
```

```
        ASCIIEncoding encoding = new ASCIIEncoding();
        string strparam = "TokenId=" + txtTokenID.Text.Trim() + "&CoCID=" +
txtCoC.Text.Trim() + "";
        byte[] postData = encoding.GetBytes(strparam);

        httpReq.ContentType = "application/x-www-form-urlencoded";
        httpReq.Method = "POST";
        httpReq.ContentLength = postData.Length;

        // convert the request to a stream object and send it on its way
        Stream ReqStrm = httpReq.GetRequestStream();
        ReqStrm.Write(postData, 0, postData.Length);
        ReqStrm.Close();

        // get the response from the web server and
        // read it all back into a string variable
        DataSet dsWeather = new DataSet();
        byte[] binaryData = null;
        using (HttpWebResponse response = httpReq.GetResponse() as HttpWebResponse)
        {
            // Load data into a dataset

            dsWeather.ReadXml(response.GetResponseStream());
            if (dsWeather.Tables[0].Rows.Count > 0)
            {
                string bytedata =
dsWeather.Tables[0].Rows[0]["fileData"].ToString();
                if (bytedata.Length > 0)
                {
                    binaryData =
System.Convert.FromBase64String(dsWeather.Tables[0].Rows[0]["fileData"].ToString());

                    String FileName = "EnvironChainSignature.zip"; ;
                    String FilePath = AppDomain.CurrentDomain.BaseDirectory; ;

                    string filenameex = FilePath + FileName;
                    FileStream fsx = new FileStream(filenameex, FileMode.Create,
FileAccess.Write);

                    ////Read block of bytes from stream into the byte array
                    fsx.Write(binaryData, 0, binaryData.Length);

                    ////Close the File Stream
                    fsx.Close();

                    System.Web.HttpResponse response1 =
System.Web.HttpContext.Current.Response;
                    response1.ClearContent();
                    response1.Clear();
                    response1.ContentType = "text/plain";
```

```
        response1.AddHeader("Content-Disposition", "attachment;  
filename=" + FileName + ";");  
        response1.TransmitFile(FilePath + FileName);  
        response1.Flush();  
        // response1.End();  
    }  
    else  
    {  
        Response.Write(  
dsWeather.Tables[0].Rows[0]["errorMessage"].ToString());  
    }  
    else  
    {  
        Response.Write("No Signature Files are Generated");  
    }  
    }  
    }  
    catch(Exception ex)  
    {  
        Logger.WriteError(ex.Message);  
        Response.Write(ex.Message);  
    }  
}
```

5. LabAuthenticate

The LabAuthenticate method will authenticate lab user against credentials and return token. In this method, a Lab user will input for username and password.

Response will be with error code and error messages. It also returns Token. Token will generate only if user has laboratory access.

If Authentication is successful, it will return a message stating that it was successful, along with with a TokenID.

Input Parameters:

- UserName
- Password

Output Parameters:

- ErrorCode

- ErrorMessage
- TokenID

Note: Both input and output parameters are case sensitive.

Successful Response

```
<errorCode>000</errorCode>
<errorMessage>Successful</errorMessage>
<tokenID>95e390c5-a4b6-4a1
```

Response is in GetAuthenticateResponse tag as below.

This token will be valid for specific time. For every other hit of method, this session time will also update.

Currently, the token is valid up to 15 minutes. This token is required for use with further methods.

Unsuccessful response

```
<errorCode>002</errorCode>
<errorMessage>Unsuccessful</errorMessage>
<tokenID></tokenID>
```

Request

```
UserName=string&Password=string
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<GetAuthenticateResponse xmlns="http://api.envirochain.com/">
  <errorCode>string</errorCode>
  <errorMessage>string</errorMessage>
  <tokenID>string</tokenID>
</GetAuthenticateResponse>
```


Client for production testing can be found in the following location:

<http://api.envirochain.com:81/Authenticate.aspx>

Sample .net Web Client Code:

```
try
{
    HttpWebRequest httpReq =
(HttpWebRequest)WebRequest.Create("http://api.envirochain.com:81/Service.aspx/LabAuthenticate");

    ASCIIEncoding encoding = new ASCIIEncoding();
    string strparam = "UserName=" + txtUserName.Text.Trim() +
"&Password=" + txtPassword.Text.Trim()+"";
    byte[] postData = encoding.GetBytes(strparam);

    httpReq.ContentType = "application/x-www-form-urlencoded";
    httpReq.Method = "POST";
    httpReq.ContentLength = postData.Length;

    // convert the request to a stream object and send it on its way
    Stream ReqStrm = httpReq.GetRequestStream();
    ReqStrm.Write(postData, 0, postData.Length);
    ReqStrm.Close();

    // get the response from the web server and
    // read it all back into a string variable
    DataSet dsWeather = new DataSet();

    using (HttpWebResponse response = httpReq.GetResponse() as
HttpWebResponse)
    {
        // Load data into a dataset

        dsWeather.ReadXml(response.GetResponseStream());
        if (dsWeather.Tables[0].Rows.Count > 0)
        {
            string Token =
dsWeather.Tables[0].Rows[0]["tokenID"].ToString();
            if (Token.Length > 0)
            {
                Response.Write("Successful Authentication with Token
:- " + Token);
            }
            else
            {
                Response.Write("Not Authenticated.");
            }
        }
    }
}
```

```
        }  
    }  
}  
catch (Exception ex)  
{  
    Logger.WriteError(ex.Message);  
    Response.Write(ex.Message);  
}
```

6. LabGetCoCID

This method will return list of CoCIDs with UserID, CompanyName, ClientName, ProjectName, CoCRef, DateTime, and NoOfSamples.

Input Parameters:

- TokenID
- StartDate
- EndDate

Output Parameters:

- ErrorCode
- ErrorMessage
- COCResponse

Production demo deployed in the following location:

<http://api.envirochain.com:81/CoCIDList.aspx>

Send all input parameters to by POST method.

Request

```
TokenId=string&StartDate=string&EndDate=string
```

Response

```
<?xml version="1.0" encoding="utf-8"?>  
<GetCOCIDeResponse xmlns="http://api.envirochain.com/">  
  <errorCode>string</errorCode>  
  <errorMessage>string</errorMessage>
```

```
<COCResponse>string</COCResponse>  
</GetCOCIDeResponse>
```

Sample .net Web Client Code:

```
try  
{  
    HttpWebRequest httpReq =  
(HttpWebRequest)WebRequest.Create("http://api.envirochain.com:81/  
Service.asmx/LabGetCoCID");  
  
    ASCIIEncoding encoding = new ASCIIEncoding();  
    string strparam = "TokenId=" + txtTokenID.Text.Trim() + "&StartDate=" +  
txtStartDate.Text.Trim() + "&EndDate=" + txtEndDate.Text.Trim() + "";  
    byte[] postData = encoding.GetBytes(strparam);  
  
    httpReq.ContentType = "application/x-www-form-urlencoded";  
    httpReq.Method = "POST";  
    httpReq.ContentLength = postData.Length;  
  
    // convert the request to a stream object and send it on its way  
    Stream ReqStrm = httpReq.GetRequestStream();  
    ReqStrm.Write(postData, 0, postData.Length);  
    ReqStrm.Close();  
  
    // get the response from the web server and  
    // read it all back into a string variable  
    DataSet dsWeather = new DataSet();  
  
    using (HttpWebResponse response = httpReq.GetResponse() as HttpWebResponse)  
    {  
        // Load data into a dataset  
  
        dsWeather.ReadXml(response.GetResponseStream());  
        if (dsWeather.Tables[0].Rows.Count > 0)  
        {  
            string COCResponse =  
dsWeather.Tables[0].Rows[0]["COCResponse"].ToString();  
            if (COCResponse.Length > 0)  
            {  
                // Response.ClearHeaders();  
  
                Response.Write(ConvertXmlToHtmlTable(COCResponse));  
  
            }  
            else  
            {  
                Response.Write("No CoCID(s) found.--" +  
dsWeather.Tables[0].Rows[0]["errorMessage"].ToString());  
            }  
        }  
    }  
}
```

```
        }
    }
    else
    {
        Response.Write("Not CoCID(s) found.--");
    }
}

}
catch (Exception ex)
{
    Logger.WriteError(ex.Message);
    Response.Write(ex.Message);
}

// below function for showing output in table format
protected string ConvertXmlToHtmlTable(string xml)
{
    StringBuilder html = new StringBuilder("<table align='center' " +
        "border='1' class='xmlTable'>\r\n");
    try
    {
        XDocument xDocument = XDocument.Parse(xml);
        XElement root = xDocument.Root;

        var xmlAttributeCollection = root.Elements().Attributes();

        foreach (var ele in root.Elements())
        {
            if (!ele.HasElements)
            {
                string elename = "";
                html.Append("<tr>");

                elename = ele.Name.ToString();

                if (ele.HasAttributes)
                {
                    IEnumerable<XAttribute> attribs = ele.Attributes();
                    foreach (XAttribute attrib in attribs)
                        elename += Environment.NewLine + attrib.Name.ToString() +
                            "=" + attrib.Value.ToString();
                }

                html.Append("<td>" + elename + "</td>");
                html.Append("<td>" + ele.Value + "</td>");
                html.Append("</tr>");
            }
            else
            {
                string elename = "";
```

```
        html.Append("<tr>");

        elename = ele.Name.ToString();

        if (ele.HasAttributes)
        {
            IEnumerable<XAttribute> attribs = ele.Attributes();
            foreach (XAttribute attrib in attribs)
                elename += Environment.NewLine + attrib.Name.ToString() +
"=" + attrib.Value.ToString();
        }

        html.Append("<td>" + elename + "</td>");
        html.Append("<td>" + ConvertXmlToHtmlTable(ele.ToString()) +
"</td>");

        html.Append("</tr>");
    }
}

html.Append("</table>");
}
catch (Exception e)
{
    return xml;
    // Returning the original string in case of error.
}
return html.ToString();
}
```

7. LabGetCoCIDDetails

In this method, we have to pass COCID and Token for getting details of CoC.

Input Parameters:

- TokenID
- CoCID

Output Parameters:

- ErrorCode
- ErrorMessage
- COCDetailResponse
- SampleDetailResponse

Production demo deployed in the following location:

<http://api.envirochain.com:81/CoCIDDetails.aspx>

Send all input parameters to by POST method.

Request

```
TokenId=string&CoCID=string
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<GetCOCDetailsResponse xmlns="http://api.envirochain.com/">
  <errorCode>string</errorCode>
  <errorMessage>string</errorMessage>
  <COCDetailResponse>string</COCDetailResponse>
  <SampleDetailResponse>string</SampleDetailResponse>
</GetCOCDetailsResponse>
```

Note: A successful response will only be for PaidCoC.

Sample .net Web Client Code:

```
try
{
    HttpWebRequest httpReq =
    (HttpWebRequest)WebRequest.Create("http://api.envirochain.com:81/
    Service.aspx/LabGetCoCIDDetails");

    ASCIIEncoding encoding = new ASCIIEncoding();
    string strparam = "TokenId=" + txtTokenID.Text.Trim() + "&CoCID=" +
    txtCOCID.Text.Trim() + "";
    byte[] postData = encoding.GetBytes(strparam);

    httpReq.ContentType = "application/x-www-form-urlencoded";
```

```
httpReq.Method = "POST";
httpReq.ContentLength = postData.Length;

// convert the request to a stream object and send it on its way
Stream ReqStrm = httpReq.GetRequestStream();
ReqStrm.Write(postData, 0, postData.Length);
ReqStrm.Close();

// get the response from the web server and
// read it all back into a string variable
DataSet dsWeather = new DataSet();

using (HttpWebResponse response = httpReq.GetResponse() as HttpWebResponse)
{
    // Load data into a dataset

    dsWeather.ReadXml(response.GetResponseStream());
    if (dsWeather.Tables[0].Rows.Count > 0)
    {
        string COCResponse =
dsWeather.Tables[0].Rows[0]["COCDetailResponse"].ToString();
        if (COCResponse.Length > 0)
        {

            Response.Write(ConvertXmlToHtmlTable(COCResponse));

        }
        else
        {

            Response.Write(
dsWeather.Tables[0].Rows[0]["errorMessage"].ToString());
        }
        string SampleDetailResponse =
dsWeather.Tables[0].Rows[0]["SampleDetailResponse"].ToString();
        if (SampleDetailResponse.Length > 0)
        {

            Response.Write(ConvertXmlToHtmlTable(SampleDetailResponse));

        }
        else
        {

            Response.Write(
dsWeather.Tables[0].Rows[0]["errorMessage"].ToString());
        }
    }
    else
```

```
        {
            Response.Clear();
            Response.Write("Not CoC Detail(s) found.--");
        }
    }

}
catch (Exception ex)
{
    Response.Clear();
    Logger.WriteError(ex.Message);
    Response.Write(ex.Message);
}

}
protected string ConvertXmlToHtmlTable(string xml)
{
    StringBuilder html = new StringBuilder("<table align='center' " +
        "border='1' class='xmlTable'>\r\n");
    try
    {
        XDocument xDocument = XDocument.Parse(xml);
        XElement root = xDocument.Root;

        var xmlAttributeCollection = root.Elements().Attributes();

        foreach (var ele in root.Elements())
        {
            if (!ele.HasElements)
            {
                string elename = "";
                html.Append("<tr>");

                elename = ele.Name.ToString();

                if (ele.HasAttributes)
                {
                    IEnumerable<XAttribute> attribs = ele.Attributes();
                    foreach (XAttribute attrib in attribs)
                        elename += Environment.NewLine + attrib.Name.ToString() +
                            "=" + attrib.Value.ToString();
                }

                html.Append("<td>" + elename + "</td>");
                html.Append("<td>" + ele.Value + "</td>");
                html.Append("</tr>");
            }
            else
            {
                string elename = "";
                html.Append("<tr>");
            }
        }
    }
}
```



```
        elename = ele.Name.ToString();

        if (ele.HasAttributes)
        {
            IEnumerable<XmlAttribute> attribs = ele.Attributes();
            foreach (XmlAttribute attrib in attribs)
                elename += Environment.NewLine + attrib.Name.ToString() +
"=" + attrib.Value.ToString();
        }

        html.Append("<td>" + elename + "</td>");
        html.Append("<td>" + ConvertXmlToHtmlTable(ele.ToString()) +
"</td>");
        html.Append("</tr>");
    }
}

html.Append("</table>");
}
catch (Exception e)
{
    return xml;
    // this function for showing out put in table format
}
return html.ToString();
}
```