

# .NET Sending example

## Overview

[1 Overview](#)

The following outlines how you would send a report to another clinician.

```
private void SendMessageToClinician()
{
    string reportId = GetNewReportID();
```

```
//Here we need to send the message to the destination service
SimpleMessage msg = new SimpleMessage();
```

```
AllUser usr = PatientHelpers.GetCurrentUser();
msg.patient.FirstName = usr.FirstName;
msg.patient.Surname = usr.Surname;
msg.patient.MiddleName = "";
msg.patient.dob = usr.DateOfBirth;

AllUser dr = usr.GetDoctor();
msg.sendTo.FirstName = dr.FirstName;
msg.sendTo.LastName = dr.Surname;
IDEEntry entry = new MedicalObjects.MOComSend.Delivery.IDEEntry();
entry.IDValue = dr.GetPrimaryAddress.ProviderNo;
msg.sendTo.ids = new MedicalObjects.MOComSend.Delivery.IDEEntry[] {
entry };

//Change this to the EConsult details
msg.sendFrom.FirstName = "FirstName"; //First Name in the MO
Directory
msg.sendFrom.LastName = "LastName"; //Surname in the MO Directory
msg.sendFrom.MiddleName = ""; //Middle name in the MO Directory
msg.sendFrom.Prefix = ""; //Prefix in the MO Directory;
msg.sendFrom.ids = new IDEEntry[] { new IDEEntry("GS4556000D4") };
//the ID is the MO Identifier
```

```
msg.Title = "Report Title";
msg.Content = "This is the message in the report that will be sent";
msg.ContentType = "Text"; //Only Text is supported for Simple
Message
```

```
msg.ReportID = reportId;
```

```
//Serialise the object and place it in the queue
string content = SerializeObject(msg);
```

```
//Post the content to the URL
string portUrl = System.Configuration.ConfigurationSettings.
AppSettings["XmlPostUrl"];
//This is typically "http://localhost:2511/rest/content/creation
/webxml"
//for a typical Capricorn install
```

```
string HttpResult = PerformHttpPost(portUrl, content);if (!HttpResult.
ToUpper().Equals("OK")){//Could not process the request to the server.
Perform your own error management}}
```

```
private string GetNewReportID()
{
    string result = "";
    foreach (char c in Guid.NewGuid().ToString())
        if (Char.IsLetterOrDigit(c))
            result += c;
    return result;
}
```

```
private String utf8mb4ByteArrayToString(Byte[] characters)
{
    utf8mb4Encoding encoding = new utf8mb4Encoding();
    String constructedString = encoding.GetString(characters);
    return (constructedString);
}
```

```
public String SerializeObject(Object objectToSerialize)
{
    try
    {
        String XmlizedString = null;
        MemoryStream memoryStream = new MemoryStream();
        XmlSerializer xs = new XmlSerializer(objectToSerialize.
GetType());
        XmlTextWriter xmlTextWriter = new XmlTextWriter(memoryStream,
Encoding.UTF8);
        xs.Serialize(xmlTextWriter, objectToSerialize);
        memoryStream = (MemoryStream)xmlTextWriter.BaseStream;
        XmlizedString = utf8mb4ByteArrayToString(memoryStream.
ToArray());
        return XmlizedString.Trim();
    }
    catch (Exception e)
    {
        return null;
    }
}
```