

OpenLM APIコンサンプションのコンバート (v4 ⇒ v5)

注意 : OpenLM Server v5.0以降では SOAPメソッドはサポートされなくなりました。

次のドキュメントは旧バージョン4.xのAPIを使用したアプリケーションの5.0へのコンバートと調整を説明します。

サンプルプロジェクトを含むこちらの[アーカイブ](#)を確認する事もできます。

目次:

XML APIコンサンプションのコンバート

[新しいリンク](#)

[コードの変更](#)

SOAP APIコンサンプションのコンバート

[新しいリンク](#)

[コードの変更](#)

[コードサンプル](#)

XML APIコンサンプションの コンバート

[新しいリンク](#)

[http://\[openlm_server\]:5015/api/easyadminapi/postmessage](http://[openlm_server]:5015/api/easyadminapi/postmessage)

コードの変更

ポート7014でXML APIを以前使用していた場合は、コードの変更は必要ありません。古い[リンク](#)



([http://\[openlm_server\]:7014/OpenLMServer](http://[openlm_server]:7014/OpenLMServer))を新しいリンクに変えてください。全てが通常通り動くはずです。

SOAP APIコンサンプション のコンバート

新しいリンク

[http://\[openlm_server\]:5015/api/easyadminapi/web/\[method\]](http://[openlm_server]:5015/api/easyadminapi/web/[method])

コードの変更

SOAPメソッドはOpenLM Serverバージョン5.xではもはやサポートされておりません。これらのメソッドを消化するためには、新しいAPIエンドポイントへのウェブリクエストが送られる必要があります。このリクエストはJSONフォーマットで行わなければなりません。レスポンスもJSONフォーマットで返ってきます。下記に旧SOAPメソッド *GetDenialsChart* をコンバートするコードサンプルを載せました。

コードサンプル

```
using APISamples.AdminAPI;
using Newtonsoft.Json;
using System;
using System.IO;
using System.Net;
using System.Text;

namespace APISamples
{
    class Example
    {
        private const int _port = 5015;
        private const string _server = "local
        private const string UserName = "Plea
        private const string Password = "Plea
```



```
public void GetTop10DeniedFeaturesInI
{
    var startTime = DateTime.UtcNow.I
    var endTime = DateTime.UtcNow;

    var request = new DenialChartRequ
    {
        BaseInfo = CreateBaseInfo(),

        GetTrueDenialsOnly = true,
        StartTime = new SlimDateTime(
        {
            Year = startTime.Year,
            Month = startTime.Month,
            Day = startTime.Day
        },

        EndTime = new SlimDateTime()
        {
            Year = endTime.Year,
            Month = endTime.Month,
            Day = endTime.Day,
            Hour = endTime.Hour,
            Minute = endTime.Minute,
            Second = endTime.Second
        },
        GroupBy = "Feature"
    };

    var response = GetDenialsChart(re
}

internal DenialsChartResponse GetDeni
{
    return JsonConvert.DeserializeObj
}

internal RequestBaseInfo CreateBaseIr
```



```
{
    return new RequestBaseInfo
    {
        PagingData = new PagingData
        {
            StartRecord = 0,
            NumOfRecord = 10,
            Sort = new string[] { "va"
            Direction = "desc"
        },

        UserLocalSettings = new UserL
        {
            TimezoneStandardName = "U
            ThousandsSeparator = ",",
            DecimalSeparator = ".",
            TimeFormat = "dd/mm/yyyy
        },
        SessionData = new SessionRefr
    };
}
```

```
#region Private methods
private string SendRequest(string jsc
{
    //string url = "http://localhost:
    string url = $"http://{_server}:{
    HttpWebRequest req = (HttpWebRequ

    req.UseDefaultCredentials = true;
    byte[] requestBytes = System.Text
    req.Method = "POST";
    req.ContentType = "text/xml;chars
    req.ContentLength = requestBytes.
    Stream requestStream = req.GetRec
    requestStream.Write(requestBytes,
    requestStream.Close();

    HttpWebResponse res = (HttpWebRes
    StreamReader sr = new StreamReade
```



```
string s = System.Text.Encoding.I
sr.Close();
res.Close();

return s;
}

private string GetSessionID()
{
    LoginFormSettingsResponse respons
    if (response.UserAuthenticationRe
    {
        return AdminAPIAuthenticatio
    }
    else//No authentication required
    {
        return string.Empty;
    }
}

private UserAuthenticationResponse Pe
{
    return JsonConvert.DeserializeObj
}

private LoginFormSettingsResponse Get
{
    var request = new LoginFormSettir
    return JsonConvert.DeserializeObj
}

private string AdminAPIAuthenticatio
{
    var userAuthenticationRequest = r
    if (!useWindowsAuthentication)//C
    {
        userAuthenticationRequest.Use
```



```
        userAuthenticationRequest.Pas
    }/* If OpenLM server authenticati
else
    {
        userAuthenticationRequest.Use
        userAuthenticationRequest.Pas
        userAuthenticationRequest.Tru

    }*/
    UserAuthenticationResponse userAu
    if (userAuthenticationResponse.Er
    {
        throw new Exception(userAuthe
    }
    return userAuthenticationResponse
}

private string GetMessage(string mess
{
    string response = SendRequest(mes
    return response;
}
#endregion
}
}
```

+81 (0)50 5893 6263

sales@openlm.com

