

# Báo cáo tóm tắt bằng hình ảnh

## - WebService Pháp - Anh

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [TraTuPhapAnh](#)

**This web service is using <http://tempuri.org/> as its default namespace.**

**Recommendation: Change the default namespace before the XML Web service is made public.**

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for XML Web services that are under development, but published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services creating using ASP.NET, the default namespace can be changed using the WebService attribute's Namespace property. The WebService attribute is an attribute applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to "http://microsoft.com/webservices/":

**C#**

```
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}
```

**Visual Basic**

```
<WebService(Namespace="http://microsoft.com/webservices/")> Public Class MyWebService
    ' implementation
End Class
```

**C++**

```
[WebService(Namespace="http://microsoft.com/webservices/")]
public ref class MyWebService {
    // implementation
};
```

For more details on XML namespaces, see the W3C recommendation on [Namespaces in XML](#).

For more details on WSDL, see the [WSDL Specification](#).

For more details on URIs, see [RFC 2396](#).

WebService1

Click [here](#) for a complete list of operations.

### TraTuPhapAnh

**Test**

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
Tu:	<input type="text" value="Bonjour"/>

**SOAP 1.1**

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /WebService1.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/TraTuPhapAnh"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <TraTuPhapAnh xmlns="http://tempuri.org/">
      <Tu>string/Tu</Tu>
    </TraTuPhapAnh>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

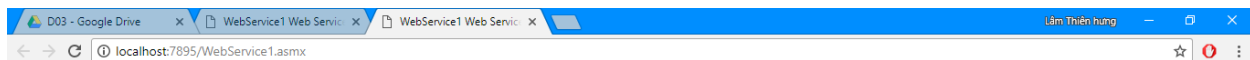
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <TraTuPhapAnhResponse xmlns="http://tempuri.org/">
      <TraTuPhapAnhResult>string</TraTuPhapAnhResult>
    </TraTuPhapAnhResponse>
  </soap:Body>
</soap:Envelope>
```

**SOAP 1.2**

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<string xmlns="http://tempuri.org/">Hello</string>
```



## WebService1

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [TraTuAnhViet](#)

This web service is using <http://tempuri.org/> as its default namespace.

**Recommendation: Change the default namespace before the XML Web service is made public.**

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for XML Web services that are under development, but published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services creating using ASP.NET, the default namespace can be changed using the WebService attribute's Namespace property. The WebService attribute is an attribute applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to "http://microsoft.com/webservices/":

```
C#
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}
```

Visual Basic

```
<WebService(Namespace="http://microsoft.com/webservices/")> Public Class MyWebService
    implementation
End Class
```

C++

```
[WebService(Namespace="http://microsoft.com/webservices/")]
public ref class MyWebService {
    // implementation
};
```

For more details on XML namespaces, see the W3C recommendation on [Namespaces in XML](#).

For more details on WSDL, see the [WSDL Specification](#).

For more details on URIs, see [RFC 2396](#).



## WebService1

Click [here](#) for a complete list of operations.

### TraTuAnhViet

#### Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
Tu:	<input type="text" value="hello"/>

Invoke

#### SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /WebService1.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/TraTuAnhViet"

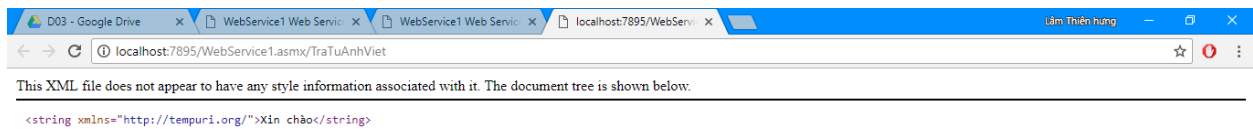
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <TraTuAnhViet xmlns="http://tempuri.org/">
      <Tu>string/Tu</Tu>
    </TraTuAnhViet>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <TraTuAnhVietResponse xmlns="http://tempuri.org/">
      <TraTuAnhVietResult>string</TraTuAnhVietResult>
    </TraTuAnhVietResponse>
  </soap:Body>
</soap:Envelope>
```

#### SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.



- Phần mềm dịch từ Pháp – Việt

Two screenshots of a Windows application window titled 'Form1'. The top screenshot shows the application with two text input fields. The first field is labeled 'Tiếng Pháp :' and is empty. The second field is labeled 'Tiếng Việt :' and is also empty. Below these fields is a button labeled 'Tra Từ'. The bottom screenshot shows the same application window, but now the 'Tiếng Pháp :' field contains the text 'Bonjour' and the 'Tiếng Việt :' field contains the text 'Xin chào'. The 'Tra Từ' button is now highlighted with a blue border.

Form1

Pháp - Việt

Tiếng Pháp :

Tiếng Việt :

Form1

Pháp - Việt

Tiếng Pháp :

Tiếng Việt :