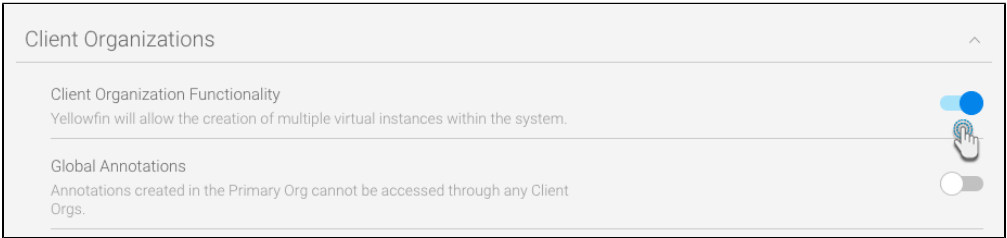


Client Organization Management Services

Yellowfin contains a functionality called [Client Organization](#), which allows multiple virtual instances of Yellowfin to reside in the same server instance. This way private content can be created within one organization, and accessed only by the users of that organization; it will be hidden from other organization users logging into the same server. The following web service calls can be used to manage this functionality.

Note: Ensure that your instance of Yellowfin has the Client Org functionality switched on for this.



Use this web service call to create a new client organization in Yellowfin. The `AdministrationClientOrg` object will be required to provide the details of the new client.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the “web services” role enabled, and must belong to the Default (i.e. Primary) Org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "CREATECLIENT".
Client	AdministrationClientOrg	Object containing details of the new client to be created. See table below .

These are the main parameters that you can set in the `AdministrationClientOrg` object for this web service call:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceID	String	A unique ID used to identify the new client. This is a mandatory parameter.
ClientName	String	Name of the new client organization.
TimeZoneCode	String	The client organization's local time zone code.
DefaultOrg	Boolean	Set this value to true if this organization is the primary organization.

Request Example

Below is a SOAP XML example for this request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>CREATECLIENT</function>
        <client>
          <clientReferenceId>org2</clientReferenceId>
          <clientName>ABC Organization</clientName>
          <defaultOrg>false</defaultOrg>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE

Response Example

The service will return the below response, according to our SOAP example:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>9204e289ced6e9ea7ed52b3cc5765663</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>

```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("CREATECLIENT");
```

- Provide information on the new client:

```
AdministrationClientOrg ac = new AdministrationClientOrg();

ac.setClientReferenceId("org2");           // Mandatory. Other parameters are optional
ac.setClientName("Organization 2");
ac.setTimeZoneCode("AUSTRALIA/SYDNEY");
ac.setDefaultOrg(false);

rsr.setClient(ac);
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: ◦ SUCCESS ◦ FAILURE

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_createclient.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, and client org details according to your environment.
4. Run `http://<host>:<port>/ws_createclient.jsp` from your Internet browser.

```

<%
/*          ws_createclient.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");                          // set to the password of the above account
rsr.setOrgId(1);

rsr.setFunction("CREATECLIENT");

AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org2");                  // mandatory. Other parameters are optional
ac.setClientName("Organization 2");
ac.setTimeZoneCode("AUSTRALIA/SYDNEY");
ac.setDefaultOrg(false);

rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

Use this web service call to retrieve a list of all client organizations within Yellowfin.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the Default (i.e. Primary) Org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "LISTCLIENTS".

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>LISTCLIENTS</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
Clients	AdministrationClientOrg[]	This object array will contain a list of client organizations.

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <clients>
          <clientId>1</clientId>
          <clientName>Default</clientName>
          <defaultOrg>true</defaultOrg>
          <timeZoneCode>AUSTRALIA/BRISBANE</timeZoneCode>
        </clients>
        <clients>
          <clientId>13003</clientId>
          <clientName>ABC Organization</clientName>
          <clientReferenceId>org2</clientReferenceId>
          <defaultOrg>false</defaultOrg>
          <timeZoneCode>AUSTRALIA/BRISBANE</timeZoneCode>
        </clients>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>ce5a636b04d89dd4acd591b6056deb1c</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("LISTCLIENTS");
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
Clients	AdministrationClientOrg[]	List of clients.

- Get client details, as shown in the following snippet:

```
AdministrationClientOrg[] clients = rs.getClients();

for (AdministrationClientOrg client: clients){
    String name = client.getClientName();
    Integer id = client.getClientId();
    String orgRefId = client.getClientReferenceId();
    String timezone = client.getTimeZoneCode();
    boolean isDefault = client.isDefaultOrg();
}
```

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

- Copy the code and save it as `ws_listclients.jsp`.
- Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
- Adjust the host, port, and admin user according to your environment.
- Run `http://<host>:<port>/ws_listclients.jsp` from your Internet browser.

```

<%
/*          ws_listclients.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number

AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();

AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");          // set to the password of the above account

rsr.setOrgId(1);

rsr.setFunction("LISTCLIENTS");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success.<br>Clients:");
    AdministrationClientOrg[] clients = rs.getClients();

    for (AdministrationClientOrg client: clients){
        out.write("<br>");
        out.write("<br>Client Name: " + client.getClientName());
        out.write("<br>Client Id: " + client.getClientId());
        out.write("<br>Org Reference Id: " + client.getClientReferenceId());
        out.write("<br>TimeZoneCode: " + client.getTimeZoneCode());
        out.write("<br>DefaultOrg: " + client.isDefaultOrg());
    }
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}

%>

```

Use this web service call to retrieve details of a specified client organization, based on the client reference ID.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the “web services” role enabled, and must belong to the Default (i.e. Primary) Org.

Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "GETCLIENT".
Client	AdministrationClientOrg	Object containing details of the client whose details are to be retrieved. See table below .

Specify these main parameters for the AdministrationClientOrg object:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceId	String	A unique ID used to identify a client organization.

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>GETCLIENT</function>
        <client>
          <clientReferenceId>org2</clientReferenceId>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
Clients	AdministrationClientOrg	Details of the requested client org.

Response Example

The service will return the below response, according to our above SOAP example:


```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <client>
          <clientId>13003</clientId>
          <clientName>ABC Organization</clientName>
          <clientReferenceId>org2</clientReferenceId>
          <defaultOrg>>false</defaultOrg>
          <timeZoneCode>AUSTRALIA/BRISBANE</timeZoneCode>
        </client>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>43d74cd4d1a49e4119e1c0dd9ca1ba21</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>

```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function:

```

AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("GETCLIENT");

```

- Once the request is configured, perform the call:

```

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
Clients	AdministrationClientOrg	Details of the requested client org.

- Get client details, as shown in the following snippet:

```
AdministrationClientOrg[] clients = rs.getClients();

for (AdministrationClientOrg client: clients){
    String name = client.getClientName();
    Integer id = client.getClientId();
    String orgRefId = client.getClientReferenceId();
    String timezone = client.getTimeZoneCode();
    boolean isDefault = client.isDefaultOrg();
}
```

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_getclient.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, and client org. ID values according to your environment.
4. Run `http://<host>:<port>/ws_getclient.jsp` from your Internet browser.

```

<%
/*          ws_getclient.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");          // set to the password of the above account
rsr.setOrgId(1);

rsr.setFunction("GETCLIENT");

AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org1");
rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success.<br>Clients:");
    AdministrationClientOrg client = rs.getClient();

        out.write("<br>");
        out.write("<br>Client Name: " + client.getClientName());
        out.write("<br>Client Id: " + client.getClientId());
        out.write("<br>Org Reference Id: " + client.getClientReferenceId());
        out.write("<br>TimeZoneCode: " + client.getTimeZoneCode());
        out.write("<br>DefaultOrg: " + client.isDefaultOrg());
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

Use this web service call to delete a client organization from Yellowfin. The client can be specified by providing the client reference ID. Note however, that the default (primary) organization cannot be deleted.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the Default (i.e. Primary) Org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.

Function	String	Web service function. Set this to "DELETECLIENT".
Client	AdministrationClientOrg	Object containing details of the client to be deleted. See table below .

These are the main parameters that you must set in the AdministrationClientOrg object for this web service call:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceId	String	A unique ID used to identify the client to be deleted. This client must already exist in the system.

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>DELETECLIENT</function>
        <client>
          <clientReferenceId>org2</clientReferenceId>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE

Response Example

The service will return the below response, according to our SOAP example:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>2eb12c07838b0721343b66435e86cdd4</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>

```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function:

```

AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("DELETECLIENT");

```

- Define the client to delete:

```

AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org2");           // must be an existing client org ref Id

rsr.setClient(ac);

```

- Once the request is configured, perform the call:

```

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
Clients	AdministrationClientOrg[]	List of clients.

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following steps:

1. Copy the code and save it as `ws_deleteclient.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, and client org. reference ID according to your environment.
4. Run `http://<host>:<port>/ws_deleteclient.jsp` from your Internet browser.

```
<%
/*                               */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");                          // set to the password of the above account

rsr.setOrgId(1);

rsr.setFunction("DELETECLIENT");
AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org2");                  // must be an existing client org ref ID

rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
} else {
    out.write("Failure");
    out.write("Code: " + rs.getErrorCode());
}
%>
```

Use this web service call to update the details of a client organization in Yellowfin.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the Default (i.e. Primary) Org.

Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "UPDATECLIENT".
Client	AdministrationClient Org	Object containing details of the client to be updated. See table below .

These are the main parameters that you can set in the AdministrationClientOrg object for this web service call:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceId	String	A unique ID used to identify the client to update their details. This client must already exist in the system. (Mandatory parameter.)
Use any other parameters to change any detail of the client organization. For example, using ClientName, you can provide a new name for the client org.		

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>UPDATECLIENT</function>
        <client>
          <clientReferenceId>org2</clientReferenceId>
          <clientName>Organization 2</clientName>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>5f0deb399f6ed6afad38a091d836cc5f</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("UPDATECLIENT");
```

- Identify the client whose information needs to be changed:

```
AdministrationClientOrg ac = new AdministrationClientOrg();

ac.setClientReferenceId("org2");           // Mandatory. Other parameters are optional
```

- Specify the client details that need to be updated. For example, you can change the client name:

```
ac.setClientName("Organization 2");

rsr.setClient(ac);
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
------------------	-----------	-------------

StatusCode	String	Status of the web service call. Possible values include: ○ SUCCESS ○ FAILURE
------------	--------	---

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_updateclient.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, and client org reference ID according to your environment.
4. Run `http://<host>:<port>/ws_updateclient.jsp` from your Internet browser.

```
<%
/*          ws_updateclient.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_admin = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_admin.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");          // set to the password of the above account
rsr.setOrgId(1);

rsr.setFunction("UPDATECLIENT");

AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org2");          // mandatory. Other parameters are optional

//the client details to be updated
ac.setClientName("Organization 2");
ac.setTimeZoneCode("AUSTRALIA/SYDNEY");
ac.setDefaultOrg(false);

rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>
```

Use this web service call to list all the users of a specified client organization. Identify the client using the client reference ID.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the Default (i.e. Primary) Org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "LISTUSERSATCLIENT".
Client	AdministrationClient Org	Object containing details of the client. See table below .

These are the main parameters that you must set in the AdministrationClientOrg object for this web service call:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceId	String	A unique ID used to identify the client whose users are to be retrieved. This client must already exist in the system.

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>LISTUSERSATCLIENT</function>
        <client>
          <clientReferenceId>org2</clientReferenceId>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
------------------	-----------	-------------

StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
People	AdministrationPerson[]	This object array will contain a list of users belonging to the client.

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <people>
          <emailAddress>admin@yellowfin.com.au</emailAddress>
          <firstName>System</firstName>
          <initial/>
          <ipId>5</ipId>
          <languageCode>EN</languageCode>
          <lastName>Administrator</lastName>
          <roleCode>YFADMIN</roleCode>
          <salutationCode/>
          <status>ACTIVE</status>
          <timeZoneCode>AUSTRALIA/SYDNEY</timeZoneCode>
          <userId>admin@yellowfin.com.au</userId>
        </people>
        <sessionId>9d7a9ea7f868bfd4cf4f632e9979db0a</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("LISTUSERSATCLIENT");
```

- Define the client to retrieve its user list:

```
AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org1"); // must be an existing client org ref Id

rsr.setClient(ac);
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: ◦ SUCCESS ◦ FAILURE
Clients	AdministrationClientOrg[]	List of clients.

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_listclientusers.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, and client org. reference ID according to your environment.
4. Run `http://<host>:<port>/ws_listclientusers.jsp` from your Internet browser.

```

<%
/*          ws_listclientusers.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");          // set to the password of the above account

rsr.setOrgId(1);

rsr.setFunction("LISTUSERSATCLIENT");
AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org1");          // must be an existing client org ref ID

rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
    AdministrationPerson[] people = rs.getPeople();
    out.write("<br>Number of Users: " + people.length + "<br>");
    for (AdministrationPerson p: people){
        out.write(p.getUserId());
    }
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

Use this call to add an existing Yellowfin user to a specified client organization.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the Default (i.e. Primary) Org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "ADDUSERACCESS".
Person	AdministrationPerson	Object containing details of the user to grant them access to a client org. See table below .

Client	AdministrationClient Org	Object containing details of the client org. See table below .
--------	--	--

These are the main parameters that you must set in the **AdministrationPerson** object for this web service call:

AdministrationPerson Element	Data Type	Description
UserID	String	To identify an existing user who is to be granted access to a client org.

These are the main parameters that you must set in the **AdministrationClientOrg** object for this web service call:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceId	String	To identify an existing client organization to add the user to.

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>ADDUSERACCESS</function>
        <person>
          <userId>admin@yellowfin.com.au</userId>
        </person>
        <client>
          <clientReferenceId>org2</clientReferenceId>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>ac657b27615227113530b79e1aa87fa4</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("ADDUSERACCESS");
```

- Define the user who is to be added to a client org:

```
AdministrationPerson ap = new AdministrationPerson();
ap.setUserId("admin@yellowfin.com.au");           //must be an existing user

rsr.setPerson(ap);
```

- Specify the client org to which to add the user:

```
AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org3");                  // must be an existing client org

rsr.setClient(ac);
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: ○ SUCCESS ○ FAILURE

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_adduseraccess.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, user to grant access and client org. reference ID according to your environment.
4. Run `http://<host>:<port>/ws_adduseraccess.jsp` from your Internet browser.

```
<%
/*          ws_adduseraccess.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_admin = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_admin.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");                          // set to the password of the above account

rsr.setOrgId(1);

rsr.setFunction("ADDUSERACCESS");
AdministrationPerson ap = new AdministrationPerson();
ap.setUserId("admin@yellowfin.com.au");          // must be an existing user
rsr.setPerson(ap);

AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org2");          // must be an existing client org

rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>
```


This web service call is used to list all the client organizations which a specified user belongs to. You can identify the user by using the AdministrationPerson object.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	A Yellowfin admin account for web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the Default (i.e. Primary) Org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. Primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "GETUSERACCESS".
Person	AdministrationPerson	Object containing details of the user. See table below .

These are the main parameters that you must set in the AdministrationPerson object for this web service call:

AdministrationPerson Element	Data Type	Description
UserID	String	To identify the user whose client organizations are to be retrieved.

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>GETUSERACCESS</function>
        <person>
          <userId>admin@yellowfin.com.au</userId>
        </person>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The response returned will contain these parameters:

Response Element	Data Type	Description
------------------	-----------	-------------

StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE
People	AdministrationPerson[]	This object array will contain all the client organizations a user belongs to.

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <clients>
          <clientId>1</clientId>
          <clientName>Default</clientName>
          <defaultOrg>true</defaultOrg>
          <timeZoneCode>AUSTRALIA/BRISBANE</timeZoneCode>
        </clients>
        <clients>
          <clientId>13004</clientId>
          <clientName>Organization 2</clientName>
          <clientReferenceId>org2</clientReferenceId>
          <defaultOrg>false</defaultOrg>
        </clients>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>fa52a1be31b08a3c44cf5198f42164</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("GETUSERACCESS");
```

- Define the user to retrieve their client list:

```
AdministrationPerson ap = new AdministrationPerson();
ap.setUserId("admin@yellowfin.com.au");

rsr.setPerson(ap);
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: ◦ SUCCESS ◦ FAILURE
Clients	AdministrationClientOrg[]	List of clients the user is a member of.

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_getuseraccess.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, and client org. reference ID according to your environment.
4. Run `http://<host>:<port>/ws_getuseraccess.jsp` from your Internet browser.

```

<%
/*          ws_getuseraccess.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");                          // set to the password of the above account

rsr.setOrgId(1);

rsr.setFunction("GETUSERACCESS");
AdministrationPerson ap = new AdministrationPerson();
ap.setUserId("admin@yellowfin.com.au");           // get user access to admin user's client list

rsr.setPerson(ap);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
    AdministrationClientOrg[] orgs = rs.getClients();
    out.write("<br>Number of Orgs: " + orgs.length + "<br>");
    out.write("<br>Org Name (Org Reference Id):");

    for (AdministrationClientOrg ac: orgs){

        out.write("<br>" + ac.getClientName() + " (" + ac.getClientReferenceId() + ")");

    }
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

This web service call is used to remove a specified user's access to a client organization. The user and the client org can be identified using the AdministrationPerson and AdministrationClientOrg objects, respectively.

This removed user account will remain in the system, even if that user doesn't belong to any other client organizations. To delete that user account from the system, you can use the DELETEUSER call. Or you could even add that user to the default organization by using the ADDUSERACCESS call.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
-----------------	-----------	-------------

LoginId	String	An admin account to connect to Yellowfin web services. This can be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "REMOVEUSERACCESS".
Person	AdministrationPerson	Object containing details of the user. See table below .
Client	AdministrationClientOrg	Object containing details of the client org. See table below .

These are the main parameters that you must set in the **AdministrationPerson** object for this web service call:

AdministrationPerson Element	Data Type	Description
UserID	String	To identify the user whose access is to be removed from the client org. This could be the user ID or email address, depending on the Logon ID method.

These are the main parameters that you must set in the **AdministrationClientOrg** object for this web service call:

AdministrationClientOrg Element	Data Type	Description
ClientReferenceId	String	To identify an existing client organization to remove the user from.

Request Example

Below is a SOAP XML example for this request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>REMOVEUSERACCESS</function>
        <person>
          <userId>binish.sheikh@yellowfin.com.au</userId>
        </person>
        <client>
          <clientReferenceId>org2</clientReferenceId>
        </client>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: - SUCCESS - FAILURE

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>a30aafab603330389d2bfb5a3e0faae7</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Start with a basic request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);

rsr.setFunction("REMOVEUSERACCESS");
```

- Define the user that is to be removed from the client org:

```
AdministrationPerson ap = new AdministrationPerson();
ap.setUserId("admin@yellowfin.com.au");

rsr.setPerson(ap);
```

- Then specify which the client organization:

```
AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org3");           // must be an existing client org

rsr.setClient(ac);
```

- Once the request is configured, perform the call:

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- The response returned will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: ◦ SUCCESS ◦ FAILURE

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_removeuseraccess.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, admin user, user whose access is to be removed and the client org. reference ID according to your environment.
4. Run `http://<host>:<port>/ws_removeuseraccess.jsp` from your Internet browser.

```
<%
/*          ws_removeuseraccess.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");                          // set to the password of the above account
rsr.setOrgId(1);

rsr.setFunction("REMOVEUSERACCESS");
AdministrationPerson ap = new AdministrationPerson();
ap.setUserId("admin@yellowfin.com.au");

rsr.setPerson(ap);
AdministrationClientOrg ac = new AdministrationClientOrg();
ac.setClientReferenceId("org3");                  // must be an existing client org

rsr.setClient(ac);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("Success");
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>
```