



AXPERT GENERIC API

For Interfacing



Table of Contents

1 Introduction..... 2

 1.1 Purpose..... 2

 1.2 Scope..... 2

2 Webservices..... 2

 2.1 Login..... 3

 2.1.1 Request..... 3

 Request format..... 3

 Response..... 4

 Response format..... 5

 2.2 GetIView..... 6

 2.2.1 Request..... 6

 Request format..... 6

 2.2.2 Response..... 7

 Response format..... 9

 2.3 SaveData..... 11

 2.3.1 Request..... 11

 Request format..... 11

 2.3.2 Response..... 13

 Response format..... 13

 2.4 ChangePassword..... 14

 1.1.1 Request..... 14

 Request format..... 14

 1.1.2 Response..... 16

 Response format..... 16

1 Introduction

1.1 Purpose

This document explains about the available Axpert restful services and how to use them to interface with axpert application.

1.2 Scope

This document is intended for users who want to consume the restful services provided by Axpert for projects that uses the Axpert as back end.

2 Webservice

Axpert provides the below services.

1. Login
2. GetIView
3. SaveData.

The Login should validate the username/password and return the menu information which assigned for user.

The web service request parameter formats are mentioned into “_parameter” json array. The web service not requires sending the “_parameter” json array from the client application.

2.1 Login

Login validates the username/password and returns either error or menu details. The current login functionality will work as mentioned below,

1. Login should validate the username and password in Axpert.
2. On successful login, provide the sessionId(s) and return the fetched menu information.

2.1.1 Request

Method	URL
POST	https://{AxpertScriptsfolder}/ASBMenuRest.dll/datasnap/rest/TASBMenuREST/Login

Request format

```
{
  "_parameters": [{
    "login": {
      "axpapp": "{projectname}",
      "username": "{username}",
      "password": "{md5hash}",
      "seed": "1983",
      "other": "Chrome"
    }
  ]
}
```

Params	Values	Length	Remarks
axpapp	string	50	Project name
username	string	50	Login username
password	string	50	The password uses double MD5 encryption. The logic is MD5(seed + MD5(password))
seed	string	4	Password encryption string
other	string	200	Anything needed record about the login session. In the web service call from web, this column will have info about the browser.

Response

Status	Sample Response
200	<p>Success</p> <pre> { "result": { "status": "Success", "s": "906614.400080.110709", "ugroup": "default", "uroles": "default", "globalvars": { "m_courtname": "ACEM ,जयपुर(द्वितीय)", "m_courttype": "ACEM", "hello": "Hai, Mobile WS!", "responsibilities": "default", "rolename": "default", "ax_evalcopy": "F" }, "usersvars": { }, "axmmenu": { "row": [{ "axmpagehrid": "13776000000000", "menu_caption": "City Master", "sname": "City", "struct_type": "S", "modifiedon": "15\07\2015 10:30:19 AM", "html_view": "", "button_param": "", "mapview": "", "forcecamera": "", "geo": "F" }] } } </pre>
200	<p>Error</p> <pre> { "error": { "status": "Failed", "msg": "Specified project not available" } } </pre>

Response format

Params	Values	Length	Remarks
status	string	10	Status of the service call
s	string	30	Generated random key from server, this key considers to maintain the session
ugroup	string	30	User group (Comma separated)
urole	string	250	User role (Comma separated)
axmmenu	String		Fetches menu details
row	String		Array of menu details, each menu record contains properties of the menu.
globalvars	String		List of application variable

2.2 GetIView

This webservice is used to fetch the data from an Iview.

2.2.1 Request

Method	URL
POST	http://{AxpertWebScripts}/ASBIViewRest.dll/datasnap/rest/TASBIViewREST/getiview

Request format

```

{
  "_parameters": [{
    "getiview": {
      "name": "{iviewname}",
      "axpapp": "{projectname}",
      "username": "",
      "password": "",
      "seed": "",
      "s": "906614.400080.110709",
      "pageno": "1",
      "pagesize": "5",
      "sqlpagination": "true",
      "params": {
        "gen": "Male"
      }
    }
  ]
}

```

Params	Values	Length	Remarks
axpapp	String	50	Project name
name	String	15	Report name
username	String	50	Login username
password	String	50	If there is a value present in the seed, the password uses double MD5 encryption. The logic is MD5(seed + MD5(password)). If seed is empty, it assumes the value to be MD5(password).
s	String	30	It is a Random key which the client will be getting on login web service call; it helps to validate the session details of the every web service call. Instead of

			<p>providing the username/password details for every web service call, to be sent the session details to validate.</p> <p>The web service will be checked with the session details, if it is empty, it requires username/password. If both are empty, it will throw the error message.</p>
pageno	Integer	5	Page number; mention which page want to generate
pagesize	Integer	5	Page size of the report. Mention the number of records in the page.
sqlpagination	Boolean		If sqlpagination= 'true' then the report will consider pageno, pagesize otherwise it will all the records as result.
params	String		Parameter of the report

2.2.2 Response

Status	Response
200	<pre> { "headrow": { "rowno": { "hide": "false" }, "axrowtype": { "hide": "true", "width": "80", "dec": "0", "type": "c" }, "axp__font": { "axp__font": "axp__fontdetails", "hide": "true", "width": "80", "dec": "0", "type": "c" }, "citymsid": { "citymsid": "ID", "width": "80", "dec": "0", "align": "Right", "type": "n", "hide": "false" }, "cityname": { </pre>

	<pre> "cityname": "CITY", "width": "125", "dec": "0", "align": "Left", "type": "c", "hide": "false" }, "reccount": "5", "totalrows": "16", "datarows": "16" }, "row": [{ "citymsid": "12662000000000", "cityname": "New Delhi", "pincode": "110001", "country": "INDIA" }, { "citymsid": "12662000000001", "cityname": "Mumbai", "pincode": "400001", "country": "INDIA" }, { "citymsid": "12662000000002", "cityname": "Chennai", "pincode": "600001", "country": "INDIA" }, { "citymsid": "12662000000003", "cityname": "Kolkata", "pincode": "700001", "country": "INDIA" }, { "citymsid": "12729000000000", "cityname": "aaa", "pincode": "1111", "country": "INDIA" }] } </pre>
200	<pre> { "error": { "status": "Failed", "msg": "Specified project not available" } } </pre>

Response format

Params	Values	Length	Remarks
headrow	String		It contains the header details of the iview
rowno	String		To show the serial number
axrowtype	String		Column width of serial number (rowno)
axp__font	String		To set font style
Columns	String		<p>We will get the columns properties from this section (JSON Pair). For each column, we added one JSON Pair. Under this we are giving following details,</p> <ol style="list-style-type: none"> 1. Column caption 2. Column width (in pixels) 3. Decimal (No of decimal places) 4. Alignment 5. Type of column data (n – numeric / c – character / d – Datetime) 6. Hide <p>For example citymsid is one column in the iview. Added the JSON Pair under this column for column property.</p> <ol style="list-style-type: none"> 1. ID is caption of the column 2. 80 – width 3. Decimal places 4. Right alignment 5. n – numeric data 6. Hidden false.
recount	Integer		It is page size
totalrows	Integer		It is number of total records
datarows	Integer		Number of total records

row	String	<p>Array or iview records</p> <p>This is JSON Array which will be having list of JSON objects as rows. Each JSON objects will be having list of pairs which specifies column name in JSON key and column value in JSON value.</p> <p>For example,</p> <pre> "citymsid": "12662000000000", "cityname": "New Delhi", "pincode": "110001", "country": "INDIA" </pre> <p>Citymsid, cityname, pincode and country are the column names.</p>
-----	--------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.3 SaveData

This web service is used for save the data into tstructs

2.3.1 Request

Method	URL
POST	http://{axpertwebscripts}/ASBTStructRest.dll/datasnap/rest/TASBTStruct/savedata

Request format

```

{
  "_parameters": [
    {
      "savedata": {
        "axpapp": "{projectname}",
        "username": "",
        "password": "",
        "seed": "1983",
        "s": "906614.400080.110709",
        "transid": "{transid}",
        "recordid": "0",
        "reodata": [
          {
            "axp_recid1": [
              {
                "rowno": "001",
                "text": "0",
                "columns": {
                  "name": "Student Name",
                  "dob": "07/05/1991",
                  "photo": "sample.jpg",
                  "address": "BANGALORE",
                  "pincode": {
                    "text": "400001",
                    "id": "12662000000001"
                  }
                }
              }
            ]
          }
        ]
      }
    }
  ]
}

```

Params	Values	Length	Remarks
--------	--------	--------	---------

Axpapp	string	50	Project name
Username	string	50	Login username
password	string	50	If there is a value present in the seed, the password uses double MD5 encryption. The logic is MD5(seed + MD5(password)). If the seed is empty, it assumes the value to be MD5(password).
seed	string	4	Password encryption string
s	string	30	It is a Random key which the client will be getting on login web service call; it helps to validate the session details of the every web service call. Instead of providing the username/password details for every web service call, to be sent the session details to validate. The web service will be checked with the session details, if it is empty, it requires username/password. If both are empty, it will throw the error message.
transid	String	5	Name of the transaction
recordid	Integer		Unique sequence number of the transaction. If it is a new transaction, it should be 0.
recdata	string		It represents the column details with data. Axp_recid is a data container (dc) of the screen. If it is 1 st dc assign with axp_recid1 and if it is 2 nd dc assign with axp_recid2. (axp_recid + dc number) Rowno is a record count. If it is not a grid dc then always "001" only. Text is record id of dc (if it is first dc it should be the same as recordid value) Columns are representsarray of column data. Left side details are column names and the right side details are column values. If it is normalized data, send text, id, oldid and ov details. If it is non normalized data, it contains text and ov details. Details of column properties, <i>text</i> : Value <i>id</i> : Normalized unique value

			<i>oldid</i> : Old normalized unique value (modification) <i>ov</i> : Old value (modification)
--	--	--	---------------------------------------------------------------------------------------------------

2.3.2 Response

Status	Response
200	<pre>{ "message": [{ "msg": "Student Master Saved", "recordid": "13329000000000" }] }</pre>
200	<pre>{ "error": { "status": "Failed", "msg": "Specified project not available" } }</pre>

Response format

Params	Values	Length	Remarks
msg	string		Return the transaction message
recordid	Integer		Return the saved recorded
Autogenerate columns	string		Autogenerate column value if anything there in the transaction; it will be returned in the response. For example: REF1516002134, INV1516/01324

2.4 ChangePassword

This web service is used for change the user’s password and forget password. The process will be differentiated by sending “service” value. Axprops table has to be configured in axpert for mailing process.

The client has to call the web service with user details and mail content. The web service will be validating the user details and updated the latest password and it will send the mail to intimate the user.

1.1.1 Request

Method	URL
POST	http://{axpertwebscripts}/ASBUserRest.dll/datasnap/rest/TASBUser/ChangePassword

Request format

```

{
  "changepassword": {
    "direct": "t",
    "axpapp": "{projectname}",
    "service": "f",
    "username": "admin",
    "password": "22723bbd4217a0abf6d3e68073c7603d",
    "seed": "",
    "s": "996000.454333.329006",
    "user": "john",
    "usermailid": "john@asdf.com",
    "subject": "Reset Password",
    "body": {
      "I0": "Hi John,",
      "I1": "Your password has been changed 4141 as per your request",
      "I2": "Please login immediately and change the password of your choice for security",
      "I3": "Regards",
      "I4": "Support Team."
    },
    "pwd": "22723bbd4217a0abf6d3e68073c7603d",
    "changebyadmin": "yes"
  }
}

```

Params	Values	Length	Remarks
axpapp	string	50	Project name
username	string	50	Login username

password	string	50	If there is a value present in the seed, the password uses double MD5 encryption. The logic is MD5(seed + MD5(password)). If seed is empty, it assumes the value to be MD5(password).
seed	string	4	Password encryption string
s	string	30	It is a Random key which the client will be getting on login web service call; it helps to validate the session details of the every web service call. Instead of providing the username/password details for every web service call, to be sent the session details to validate. The web service will be checked with the session details, if it is empty, it requires username/password. If both are empty, it will throw the error message.
service	String	1	f- forgot password, c- change password Password/Session details will be validated only when the service = "c"
usermailid	String		The password detail mail will be sent to usermailid
Subject	String		Mail subject
Body	String		Mail details
Pwd	String		New password (Refer to password column to encrypt the pwd data)
changedbyadmin	String		To validate the new password text
direct	String		Default "t" – as true

1.1.2 Response

Status	Response
200	<pre>{ "result": { "status": "Success", "msg": "Password is reset and sent to your email" } }</pre>
200	<pre>{ "error": { "status": "Failed", "msg": "Invalid Username / email" } }</pre>

Response format

Params	Values	Length	Remarks
status	string		Return the status of web call
Msg	String		Return the success message or error message