



JOOMEO SERVICES

JOOMEO A.P.I. VERSION 0.5 BETA



www.jooméo.com

contact@jooméo.com

SOMMAIRE

JOOMEO SERVICES	3
GETTING STARTED	3
API KEY	3
CURRENT RELEASE	3
API OVERVIEW	4
CONVENTIONS	4
API METHODS	6
VERSION	6
SESSIONS	9
USER	15
USER.COLLECTION	38
USER.ALBUM	38
USER.CONTACT	54
USER.FILE	59
API ERRORS CODES	64
ANNEXES	66

JOOMEO SERVICES

Joomeo's Application Programming Interface (API) allows you to build your own creative applications around the content created on Joomeo. You can retrieve, explore albums, pictures from a Joomeo space.

You'll need an [API Key](#) to get started, and need to use [XML-RPC \(1\)](#) to interact with the service.

GETTING STARTED

Anyone is free to write application to do cool stuff with available Joomeo data via our API. Follow the steps below to build your first application :

- ▶ Read the Joomeo API documentation
- ▶ Sign up for a user account
- ▶ Register for an API Key (Public or Commercial)

API KEY

To use Joomeo's API, you need an API key. Each individual application or use of the API can use its own key. It also helps us track how it's being used so we can improve it.

We encourage you to develop Commercial applications to build your business on top of Joomeo. There are few exceptions on commercial applications and we invite you to contact us for clarification.

Also note that the Joomeo name, the Joomeo logo are owned and trademarked by Airsens and may not be used without permission.

[Have questions?](#)

Please contact us with anything you might want to know : api_developer@joomeo.com.

CURRENT RELEASE

Joomeo API version: 0.5 Beta

API OVERVIEW

The Joomeo API consists of a set of callable methods that allow you to make remote procedure calls to the Joomeo XML-RPC Server endpoint over the internet. Every HTTP Request must use the method POST.

Joomeo XMLRPC API endpoint is available at the URL: <http://api.joomeo.com/xmlrpc.php>

CONVENTIONS

AUTHENTICATION

Every Joomeo API call must include the API Key. There is no anonymous access type.

LIMITS

The maximum life time of a session is limited to 60 minuts.

XML-RPC REQUEST FORMAT

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>method_name</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>param1_name</name>
          <value><type_param>value</type_param></value>
        </member>
        <member><name>param2_name</name>
          <value>< type_param >value</string></type_param>
        </member>
        .....
        .....
        <member><name> paramn_name </name>
          <value>< type_param >value</type_param ></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

XML-RPC RESPONSE FORMAT

```
<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value>[array | struct... ]
      <type>value</type>
    </value>
  </param>
</params>
</methodResponse>
```

ERROR MESSAGES

Error messages are returned as XML messages with an error code and an error message. The error message is for debugging only. It can change without notice and it is also not suitable for displaying to the user.

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <fault>
    <value>
      <struct><member><name>faultCode</name>
        <value><int>value</int></value>
      </member>
      <member>
        <name>faultString</name>
        <value><string>string</string></value>
      </member>
    </struct>
  </value>
</fault>
</methodResponse>
```

API METHODS

VERSION

ABOUT

Returns the current API version and the list of available methods.

Method Name : [joomeo.version.about](#)

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
  <methodCall>
    <methodName>
      joomeo.version.about
    </methodName>
    <params>
    </params>
  </methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
  <param>
    <value><struct>
      <member>
        <name>apiDocsUrl</name>
        <value>
          <string>http://api.joomeo.com/JoomeoAPI.pdf</string>
        </value>
      </member>
      <member>
        <name>apiName</name>
        <value>
          <string>JOOMEO XML-RPC API</string>
        </value>
      </member>
      <member>
        <name>apiVersion</name>
        <value><string>0.5</string></value>
      </member>
      <member>
        <name>apiStatus</name>
        <value><string>Beta</string></value>
      </member>
    </struct>
  </value>
</param>
</params>
</methodResponse>
```

- **Method Name** : system.listMethods

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>
    system.listMethods
  </methodName>
  <params>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
  <param>
    <value>
      <array>
        <data>
          <value><string>joomeo.session.init</string></value>
          <value><string>joomeo.session.getProperties</string></value>
          <value><string>joomeo.session.initContact</string></value>
          <value><string>joomeo.session.kill</string></value>
          <value><string>joomeo.user.getStatistics</string></value>
          <value><string>joomeo.user.getAlbumList</string></value>
          <value><string>joomeo.user.getCollectionList</string></value>
          <value><string>joomeo.user.getFilesList</string></value>
          <value><string>joomeo.user.getNumberOfFiles</string></value>
          <value><string>joomeo.user.getHubList</string></value>
          <value><string>joomeo.user.getContactList</string></value>
          <value><string>joomeo.user.addContact</string></value>
          <value><string>joomeo.user.addAlbum</string></value>
          <value><string>joomeo.user.addCollection</string></value>
          <value><string>joomeo.user.getAllowedFileType</string></value>
          <value><string>joomeo.user.uploadBinary</string></value>
          <value><string>joomeo.user.album.getFilesList</string></value>
          <value><string>joomeo.user.album.getNumberOfFiles</string></value>
          <value><string>joomeo.user.album.saveUploadedFile</string></value>
          <value><string>joomeo.user.album.update</string></value>
          <value><string>joomeo.user.album.delete</string></value>
          <value><string>joomeo.user.album.deleteFile</string></value>
          <value><string>joomeo.user.album.allowContactAccess</string></value>
          <value><string>joomeo.user.album.removeContactAccess</string></value>
          <value><string>joomeo.user.album.getAllowedContactAccessList</string></value>
          <value><string>joomeo.user.collection.getFilesList</string></value>
          <value><string>joomeo.user.collection.getNumberOfFiles</string></value>
```

```
<value><string>joomeo.user.collection.saveUploadedFile</string></value>
<value><string>joomeo.user.contact.update</string></value>
<value><string>joomeo.user.contact.delete</string></value>
<value><string>joomeo.user.contact.sendInvite</string></value>
<value><string>joomeo.user.file.delete</string></value>
<value><string>joomeo.user.file.getBinary</string></value>
<value><string>joomeo.version.about</string></value>
<value><string>system.listMethods</string></value>
<value><string>system.methodHelp</string></value>
<value><string>system.methodSignature</string></value>
<value><string>system.multicall</string></value>
<value><string>system.getCapabilities</string></value>
</data>
</array>
</value>
</param>
</params>
</methodResponse>
```

SESSIONS

INIT

Method used to initialize a user session and sets up internal control structures. Init must be the first method of the API call in the application.

Method Name : [joomeo.session.init](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
spacename (Required)	string	Joomeo space name
login (Required)	string	Your login associated to the Joomeo space name.
password (Required)	string	Your password.

Returns : The session id and the type of the session

Structure :

Member name	Type	Description
sessionId	string	Unique identifiant of the session
sessionType	int	0 - User Session, 1 - Contact Session
spacename	string	Joomeo space name
spaceURL	string	Joomeo space URL

Request format

```
<methodCall>
<methodName>joomeo.session.init</methodName>
<params>
  <param>
    <value><struct>
      <member>
        <name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
      <member>
        <name>spacename</name>
        <value><string>MyFamily</string></value>
      </member>
      <member>
        <name>login</name>
        <value><string>John</string></value>
      </member>
      <member>
        <name>password</name>
        <value><string>MyPassword</string></value>
      </member>
    </struct></value>
  </param>
</params>
```

```
</methodCall>
```

Response format

```
<methodResponse>
  <params>
    <param>
      <value><struct>
        <member>
          <name>sessionid</name>
          <value><string>adb7bf8f65f858b13bfb0c271de01206</string></value>
        </member>
        <member><
          name>sessionType</name>
          <value><int>0</int></value>
        </member>
        <member>
          <name>spacename</name>
          <value><string>MyFamily</string></value>
        </member>
        <member>
          <name>spaceURL</name>
          <value><string>http://joomeo.com/MyFamily</string></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodResponse>
```

GETPROPERTIES

Method is used to get or re-initialize a user session. GetProperties can be the first method of the API call in the application if you ever get a sessionid.

Method Name : [joomeo.session.getProperties](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

Returns : The session id and the type of the session

Structure :

Member name	Type	Description
sessionid	string	Unique identifiant of the session
sessionType	int	0 - User Session, 1 - Contact Session
spacename	string	Joomeo space name
spaceURL	string	Joomeo space URL

Request format

```

<methodCall>
<methodName>joomeo.session.getProperties</methodName>
<params>
  <param>
    <value>
      <struct>
        <member><name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member><name>sessionid</name>
          <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
        </member>
      </struct>
    </value>
  </param>
</params>
</methodCall>

```

Response format

```

<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>sessionid</name>
            <value><string>adb7bf8f65f858b13bfb0c271de01206</string></
value>
          </member>
          <member>
            <name>sessionType</name>
            <value><int>0</int></value>
          </member>
          <member>
            <name>spacename</name>
            <value><string>MyFamily</string></value>
          </member>
          <member>
            <name>spaceURL</name>
            <value><string>http://joomeo.com/MyFamily</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>

```

INITCONTACT

Method used to initialize a contact session and sets up internal control structures. InitContact must be used to explore hubs.

Method Name : [joomeo.session.initContact](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
contactid (Required)	string	Contactid returned from the getHubList Method.

Returns : The session id and the type of the session

Structure :

Member name	Type	Description
sessionid	string	Unique identifiant of the session
sessionType	int	1 - Contact Session
spacename	string	Joomeo space name
spaceURL	string	Joomeo space URL

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.session.initContact</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member><name>sessionid</name>
          <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
        </member>
        <member><name>contactid</name>
          <value><string>NTc4YjlyZDDzriVfcogW7A%3D%3D%0D%0A</string></
value>
      </member>
    </struct></value>
  </param>
</params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
```

```

        <member>
            <name>sessionid</name>
            <value><string>3c2cb6d2fc47f6769e107538f6f6656d</string></
value>
        </member>
        <member>
            <name>sessionType</name>
            <value><int>1</int></value>
        </member>
        <member>
            <name>spacename</name>
            <value><string>MyFamily</string></value>
        </member>
        <member>
            <name>spaceURL</name>
            <value><string>http://joomeo.com/MyFamily</string></value>
        </member>
    </struct>
</value>
</param>
</params>
</methodResponse>

```

KILL

Method used to kill an existant session. Kill must be used before leaving your application.

Method Name : `joomeo.session.kill`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

Returns : success if the session is killed with success

Structure :

Member name	Type	Description
success	int	1 - session killed with success

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
    <methodName>joomeo.session.kill</methodName>
    <params>
        <param>
            <value><struct>
                <member><name>apikey</name>
                    <value><string>12345a12345b12345c12345d</string></value>
                </member>
                <member><name>sessionid</name>

```

```
        <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
      </member>
    </struct></value>
  </param>
</params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>success</name>
            <value><int>1</int></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

USER

GETSTATISTICS

Method to get statistics about the Joomeo space associated to the user session.

Method Name : [joomeo.user.getStatistics](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

Returns : The number of photos of the given sessionid..

Structure :

Member Name	Type	Description
numberOfFiles	int	Number of files in the Joomeo space
numberOfAlbums	int	Number of albums in the Joomeo space
numberOfContacts	int	Number of contacts in the Joomeo space (this member is present only with a sessionType = 0)
totalUsedSpace	double	total size of the stored files (in byte)
freeSpace	double	free space to upload new files (in byte)
spaceAllowed	double	space allowed per month or total space for enterprise offers

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.getStatistics</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
      <member><name>sessionid</name>
      <value><string>0f16a26a812a1621b673211123501af9</string></value>
    </member>
    </struct></value>
  </param>
</params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
  <param><value><struct>
    <member>
      <name>numberOfFiles</name>
```

```

        <value><int>14739</int></value>
    </member>
    <member>
        <name>numberOfAlbums</name>
        <value><int>276</int></value>
    </member>
    <member>
        <name>numberOfContacts</name>
        <value><int>124</int></value>
    </member>
    <member>
        <name>totalUsedSpace</name>
        <value><double>3523460232</double></value>
    </member>
</struct></value></param>
</params>
</methodResponse>

```

GETCOLLECTIONLIST

This method is deprecated. Use `joomeo.user.getAlbumList` instead.

GETALBUMLIST

Method used to get the list of albums of the specified sessionid.

Method Name : `joomeo.user.getAlbumList`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
orderby	string	"date" or "name" - (date by default)

Returns : The list of albums of the specified sessionid sort on <orderby>.

Array of Structure :

Member name	Type	Description
albumid	string	Identifiant of the album
label	string	label of the album
orderby	string	sort by criteria
date	double	date of creation by the user
createddate	double	date of creation in the database
public	int	0 (private) - 1 (public)
allowdownload	int	0 (No) - 1 (Yes)
allowupload	int	0 (No) - 1 (Yes)
folderid	string	Identifiant of the folder

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
    <methodName>joomeo.user.getAlbumList</methodName>

```

```

<params>
  <param>
    <value><struct>
      <member><name>apikey</name>
      <value><string>12345a12345b12345c12345d</string></value>
    </member>
    <member><name>sessionid</name>
    <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
    </member>
    <member><name>orderby</name>
    <value><string>date</string></value>
    </member>
  </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><array>
        <data>
          <value><struct>
            <member><name>albumid</name>
            <value><string>NTc4YjlyZDCXMkyzQfxmWA%3D%3D%0D%0A</string></value>
          </member>
            <member><name>label</name>
            <value><string>Album 1</string></value>
          </member>
            <member><name>orderby</name>
            <value><string>date</string></value>
          </member>
            <member><name>date</name>
            <value><double>1.218201944E+12</double></value>
          </member>
            <member><name>createddate</name>
            <value><double>1.218201944E+12</double></value>
          </member>
            <member><name>public</name>
            <value><int>0</int></value>
          </member>
            <member><name>allowdownload</name>
            <value><int>1</int></value>
          </member>
            <member><name>allowupload</name>
            <value><int>0</int></value>
          </member>
        </data>
      </value>
    </param>
  </params>
</methodResponse>

```

```

    <member><name>folderid</name>
    <value></value>
  </member>
</struct></value>
<value><struct>
  <member><name>albumid</name>
  <value><string>NTc4YjlyZDAmo07w43O0dw%3D%3D%0D%0A</string></value>
</member>
  <member><name>label</name>
  <value><string>Album 2</string></value>
</member>
  <member><name>orderby</name>
  <value><string>date</string></value>
</member>
  <member><name>date</name>
  <value><double>1.218201944E+12</double></value>
</member>
  <member><name>createddate</name>
  <value><double>1.218201944E+12</double></value>
</member>
  <member><name>public</name>
  <value><int>0</int></value>
</member>
  <member><name>allowdownload</name>
  <value><int>1</int></value>
</member>
  <member><name>allowupload</name>
  <value><int>0</int></value>
</member>
  <member><name>folderid</name>
  <value><string>NTc4YjlyZDBCu0sCh9%2FR2w%3D%3D%0D%0A</string></value>
</member>
</struct></value>
</data>
</array></value>
</param>
</params>
</methodResponse>

```

GETFILESLIST

Method used to get the list of photos/videos of a given sessionid.

Method Name : [joomeo.user.getFilesList](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

orderby	string	“date” or “name” - (date by default)
random	int	Generate a Random result sort order : 0 (No) - 1 (Yes) - (0 by default)
startresult	int	Start index to get files list (used if maxresult >0) (0 byDefault)
maxresult	int	Number of photos to return : 0 or N - (0 - unlimited by default)

Returns : A list of photos properties.

Array of Structure :

Member name	Type	Description
fileid	string	Identifiant of the file
filename	string	Name of the file
rotation	int	rotation in degree
rating	int	rate of the picture
type_mime	string	Mime type of the file
size	int	size in bytes
width	int	width of the picture
height	int	height of the picture
date_shooting	double	shooting date of the picture
date_creation	double	creation date of the picture
allowdownload	int	0 (No) - 1 (Yes)
legend	string	caption of the photo
collectionid	string	Parent Album Id (this member is deprecated, Use albumid instead)
albumid	string	Parent Album Id

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.GetFilesList</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
        <member><name>sessionid</name>
        <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
      </member>
        <member><name>orderby</name>
        <value><string>date</string></value>
      </member>
        <member><name>random</name>
        <value><int>0</int></value>
      </member>
        <member><name>startresult</name>
        <value><int>20</int></value>
      </member>
        <member><name>maxresult</name>
        <value><int>20</int></value>
      </member>
    </param>
  </params>
</methodCall>
```

```

        </struct></value>
    </param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
    <param>
        <value><array>
            <data>
                <value><struct>
                    <member><name>fileid</name>
                    <value><string>NTc4YjlyZDCE2zKHkYzRiw%3D%3D%0D%0A</string></value>
                </member>
                <member><name>filename</name>
                <value><string>AnnaMarie.jpg</string></value>
                </member>
                <member><name>rotation</name>
                <value><int>0</int></value>
                </member>
                <member><name>rating</name>
                <value><int>0</int></value>
                </member>
                <member><name>type_mime</name>
                <value><string>image/jpeg</string></value>
                </member>
                <member><name>size</name>
                <value><int>597810</int></value>
                </member>
                <member><name>width</name>
                <value><int>1440</int></value>
                </member>
                <member><name>height</name>
                <value><int>900</int></value>
                </member>
                <member><name>date_shooting</name>
                <value><double>1.085775444E+12</double></value>
                </member>
                <member><name>date_creation</name>
                <value><double>1.218201944E+12</double></value>
                </member>
                <member><name>allowdownload</name>
                <value><int>1</int></value>
                </member>
                <member><name>legend</name>
                <value><string>caption of the photo</string></value>
                </member>
            </data>
        </value>
    </param>
</params>
</methodResponse>

```

```
<member><name>collectionid</name>
<value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
</member>
<member><name>albumid</name>
<value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
</member>
</struct></value>
<value><struct>
<member><name>fileid</name>
<value><string>NTc4YjlyZDB0TDCs31thug%3D%3D%0D%0A</string></value>
</member>
<member><name>filename</name>
<value><string>BlackDog.jpg</string></value>
</member>
<member><name>rotation</name>
<value><int>0</int></value>
</member>
<member><name>rating</name>
<value><int>0</int></value>
</member>
<member><name>type_mime</name>
<value><string>image/jpeg</string></value>
</member>
<member><name>size</name>
<value><int>228704</int></value>
</member>
<member><name>width</name>
<value><int>2560</int></value>
</member>
<member><name>height</name>
<value><int>1600</int></value>
</member>
<member><name>date_shooting</name>
<value><double>1.214466532E+12</double></value>
</member>
<member><name>date_creation</name>
<value><double>1.223021747E+12</double></value>
</member>
<member><name>allowdownload</name>
<value><int>1</int></value>
</member>
<member><name>legend</name>
<value><string>caption of the photo</string></value>
</member>
<member><name>collectionid</name>
<value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
</member>
<member><name>albumid</name>
<value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
```

```

</member>
</struct></value>
<value><struct>
<member><name>fileid</name>
<value><string>NTc4YjlyZDAXNvzDYEaopQ%3D%3D%0D%0A</string></value>
</member>
<member><name>filename</name>
<value><string>BlueridgeMountains.jpg</string></value>
</member>
<member><name>rotation</name>
<value><int>0</int></value>
</member>
<member><name>rating</name>
<value><int>0</int></value>
</member>
<member><name>type_mime</name>
<value><string>image/jpeg</string></value>
</member>
<member><name>size</name>
<value><int>1331154</int></value>
</member>
<member><name>width</name>
<value><int>2560</int></value>
</member>
<member><name>height</name>
<value><int>1600</int></value>
</member>
<member><name>date_shooting</name>
<value><double>1.214466532E+12</double></value>
</member>
<member><name>date_creation</name>
<value><double>1.218201862E+12</double></value>
</member>
<member><name>allowdownload</name>
<value><int>1</int></value>
</member>
<member><name>legend</name>
<value><string>caption of the photo</string></value>
</member>
<member><name>collectionid</name>
<value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
</member>
<member><name>albumid</name>
<value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
</member>
</struct></value>
</data>
</array></value>
</param>

```

```
</params>
</methodResponse>
```

GETNUMBEROFFILES

Method to get the number of the photos of a given sessionid.

Method Name : `joomeo.user.getNumberOfFiles`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

Returns : The number of photos of the given sessionid..

Structure :

Member Name	Type	Description
nbfiles	int	Number of files in the album

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.getNumberOfFiles</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
        <member><name>sessionid</name>
        <value><string>0f16a26a812a1621b673211123501af9</string></value>
      </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
  <param><value><struct>
    <member><name>nbfiles</name>
    <value><int>147</int></value>
  </member>
  </struct></value></param>
</params>
</methodResponse>
```

GETHUBLIST

Method used to get the list of the hubs.

This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.getHubList](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

Returns : The list of the hubs for a given sessionid.

Array of Structure :

Member name	Type	Description
contactid	string	Identifiant of the contact
username	string	Name of the Joomeo space
email	string	Email
firstname	string	FirstName
lastname	string	LastName
lastvisit time	double	Date when the contact has been connected for the last time

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.getHubList</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
        <member><name>sessionid</name>
        <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
      </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><array>
        <data>
          <value><struct>
            <member><name>contactid</name>
```

```
<value><string>MmUzN2M3ZWRQqBxaKXd6mw%3D%3D%0D%0A</string></value>
</member>
<member><name>username</name>
<value><string>myspace</string></value>
</member>
<member><name>email</name>
<value><string>john.doe@company.com</string></value>
</member>
<member><name>firstname</name>
<value><string>John</string></value>
</member>
<member><name>lastname</name>
<value><string>DOE</string></value>
</member>
<member><name>lastvisit</name>
<value><double>1.214466532E+12</double></value>
</member>
</struct></value>
<value><struct>
<member><name>contactid</name>
<value><string>MmUzN2M3ZWQk8k69Izpcbw%3D%3D%0D%0A</string></value>
</member>
<member><name>username</name>
<value><string>perauscher</string></value>
</member>
<member><name>email</name>
<value><string>peter.rausher@gmail.com</string></value>
</member>
<member><name>firstname</name>
<value><string>Peter</string></value>
</member>
<member><name>lastname</name>
<value><string>Rausher</string></value>
</member>
<member><name>lastvisit</name>
<value><double>1.214466532E+12</double></value>
</member>
</struct></value>
<value><struct>
<member><name>contactid</name>
<value><string>MmUzN2M3ZWRAHGjI6Sh7TA%3D%3D%0D%0A</string></value>
</member>
<member><name>username</name>
<value><string>Steeve</string></value>
</member>
<member><name>email</name>
<value><string>steeve.compo@joomeo.com</string></value>
</member>
<member><name>firstname</name>
```

```

    <value><string>Steeve</string></value>
  </member>
  <member><name>lastname</name>
  <value><string>Compo</string></value>
</member>
  <member><name>lastvisit</name>
  <value><double>1.214466532E+12</double></value>
</member>
</struct></value>
</data>
</array></value>
</param>
</params>
</methodResponse>

```

GETCONTACTLIST

Method used to get the list of the contacts.

This method is allowed with a sessionType equals to 0.

Method Name : `joomeo.user.getContactList`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
orderby	string	default order to get the contact list (lastname (default), lastVisit, firstname, email, login)

Returns : The list of the contacts for a given sessionid.

Array of Structure :

Member name	Type	Description
contactid	string	Identifiant of the contact
login	string	login of the contact
password	string	contact password to connect him to the Joomeo space associated to the current user session
email	string	Email
firstname	string	FirstName
lastname	string	LastName
lastvisit	double	Date when the contact has been connected for the last time

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.getContactList</methodName>
  <params>
    <param>
      <value><struct>
        <member>

```

```

        <name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
    </member>
    <member>
        <name>sessionid</name>
        <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
    </member>
    <member>
        <name>orderby</name>
        <value><string>email</string></value>
    </member>
</struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
    <params>
        <param><value><array><data>
            <value><struct>
                <member>
                    <name>contactid</name>
                    <value><string>MmUzN2M3ZWRQqBxaKXd6mw%3D%3D%0D%0A</
string></value>
                </member>
                <member>
                    <name>login</name>
                    <value><string>john.doe</string></value>
                </member>
                <member>
                    <name>password</name>
                    <value><string>aw5Dt</string></value>
                </member>
                <member>
                    <name>email</name>
                    <value><string>john.doe@company.com</string></value>
                </member>
                <member>
                    <name>firstname</name>
                    <value><string>John</string></value>
                </member>
                <member>
                    <name>lastname</name>
                    <value><string>DOE</string></value>
                </member>
                <member>
                    <name>lastvisit</name>

```

```

                <value><double>1.214466532E+12</double></value>
            </member>
        </struct></value>
        <value><struct>
            <member>
                <name>contactid</name>
                <value><string>MmUzN2M3ZWRQqBxaKXd6mw%3D%3D%0D%0A</
string></value>
            </member>
            <member>
                <name>login</name>
                <value><string>john.doe</string></value>
            </member>
            <member>
                <name>password</name>
                <value><string>aw5Dt</string></value>
            </member>
            <member>
                <name>email</name>
                <value><string>john.doe@company.com</string></value>
            </member>
            <member>
                <name>firstname</name>
                <value><string>John</string></value>
            </member>
            <member>
                <name>lastname</name>
                <value><string>DOE</string></value>
            </member>
            <member>
                <name>lastvisit</name>
                <value><double>1.214466532E+12</double></value>
            </member>
        </struct></value>
    </data></array></value></param>
</params>
</methodResponse>

```

ADDCONTACT

Method used to add a new contact in the Joomeo Space.
This method is allowed with a sessionType equals to 0.

Method Name : `joomeo.user.addContact`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

email (Required)	string	A valid address email
firstname	string	the firstname of the contact to add
lastname	string	the lastname of the contact to add
login	string	the username used to connect to the Joomeo space
password	string	the password used to connect to the Joomeo space

*The arguments : **firstname, lastname, login and password are required if one of them are passed***

Returns : The properties of the created contact

Structure :

Member Name	Type	Description
exist	int	1 if contact already exist, 0 if not
contactid	string	Identifiant of the contact
login	string	login of the contact
password	string	contact password to connect him to the Joomeo space associated to the current user session
email	string	Email
firstname	string	FirstName
lastname	string	LastName

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.addContact</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>0f16a26a812a1621b673211123501af9</string></value>
        </member>
        <member>
          <name>email</name>
          <value><string>john.doe@doe.com</string></value>
        </member>
        <member>
          <name>firstname</name>
          <value><string>john</string></value>
        </member>
        <member>
          <name>lastname</name>
          <value><string>Doe</string></value>
        </member>
        <member>
          <name>login</name>
```

```

        <value><string>johon.doe</string></value>
    </member>
    <member>
        <name>password</name>
        <value><string>wxAs78</string></value>
    </member>
    </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
    <param><value><struct>
        <member>
            <name>exist</name>
            <value><int>0</int></value>
        </member>
        <member>
            <name>contactid</name>
            <value><string>MmUzN2M3ZWTOwiAp9ts3%2BA%3D%3D%0D%0A</string></
value>
        </member>
        <member>
            <name>email</name>
            <value><string>john.doe@doe.com</string></value>
        </member>
        <member>
            <name>firstname</name>
            <value><string>john</string></value>
        </member>
        <member>
            <name>lastname</name>
            <value><string>Doe</string></value>
        </member>
        <member>
            <name>login</name>
            <value><string>johon.doe</string></value>
        </member>
        <member>
            <name>password</name>
            <value><string>wxAs78</string></value>
        </member>
    </struct></value></param>
</params>
</methodResponse>

```

ADDCOLLECTION

This method is deprecated. Use `joomeo.user.addAlbum` instead.

ADDALBUM

Method used to add a new album in the Joomeo Space.

This method is allowed with a `sessionType` equals to 0.

Method Name : [joomeo.user.addAlbum](#)

Arguments

Name	Type	Description
<code>apikey</code> (Required)	string	Your application API key.
<code>sessionid</code> (Required)	string	Session Id
<code>label</code> (Required)	string	Name of the album
<code>previewSize</code>	int	Maximum size of the slideshow and the previews (1920 by default)
<code>albumDate</code>	double	Date of the album (date of the day by default) (unix timestamp)
<code>public</code>	int	1 to add an album accessible by all the Joomeo space's contacts, 0 to add a private album (0 by default)

Returns : The albumid of the new album

Structure :

Member Name	Type	Description
<code>albumid</code>	string	Identifiant of the album
<code>label</code>	string	Name of the album
<code>orderby</code>	string	default order to display the album's photos
<code>previewSize</code>	int	Maximum size of the slideshow and the previews (1920 by default)
<code>albumDate</code>	double	date of the album (unix timestamp)
<code>createdDate</code>	double	crated date of the album (unix timestamp)
<code>public</code>	int	1 accessible by all contacts, 0 private to a contact selection
<code>allowdownload</code>	int	0 current user is not allowed to download original photos from this album, 1 current user is allowed to download original photos
<code>allowupload</code>	int	0 current user is not allowed to upload new photos into this album, 1 current user is allowed to upload

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.addAlbum</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
```

```

        <value><string>0f16a26a812a1621b673211123501af9</string></value>
    </member>
    <member>
        <name>label</name>
        <value><string>my album</string></value>
    </member>
    <member>
        <name>previewSize</name>
        <value><int>1920</int></value>
    </member>
    <member>
        <name>albumDate</name>
        <value><double>1.214466532E+12</double></value>
    </member>
    <member>
        <name>public</name>
        <value><int>1</int></value>
    </member>
    </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
    <param><value><struct>
        <member>
            <name>albumid</name>
            <value><string>MmUzN2M3ZWTOwiAp9ts3%2BA%3D%3D%0D%0A</string></
value>
        </member>
        <member>
            <name>label</name>
            <value><string>my album</string></value>
        </member>
        <member>
            <name>orderBy</name>
            <value><string>date</string></value>
        </member>
        <member>
            <name>previewSize</name>
            <value><int>1920</int></value>
        </member>
        <member>
            <name>albumDate</name>
            <value><double>1.214466532E+12</double></value>
        </member>
    </struct>
    </param>
</params>
</methodResponse>

```

```

    <member>
      <name>createdDate</name>
      <value><double>1.214466532E+12</double></value>
    </member>
    <member>
      <name>public</name>
      <value><int>1</int></value>
    </member>
    <member>
      <name>allowdownload</name>
      <value><int>1</int></value>
    </member>
    <member>
      <name>allowupload</name>
      <value><int>0</int></value>
    </member>
  </struct></value></param>
</params>
</methodResponse>

```

GETALLOWEDFILETYPE

Method used to get the allowed file type to upload in the Joomeo space of the current user.

Method Name : [joomeo.user.getAllowedFileType](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id

Returns : The list of the files types allowed to be uploaded in the current Joomeo space (their extensions with an array of associated mime type).

Array of Structure :

Member name	Type	Description
extension	string	Identifiant of the file
typemimes	array of structure	The list of mime type allowed with the extension

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.getAllowedFileType</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>

```

```

        <member>
            <name>sessionid</name>
            <value><string>0f16a26a812a1621b673211123501af9</string></value>
        </member>
    </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
    <params><param><value><array><data>
        <value><struct>
            <member>
                <name>extension</name>
                <value><string>avi</string></value>
            </member>
            <member>
                <name>type_mimes</name>
                <value><array>
                    <data><value><struct><member>
                        <name>type_mime</name>
                        <value><string>video/avi</string></value>
                    </member></struct></value></data>
                </array>
            </value>
        </member>
    </struct></value>
    <value><struct>
        <member>
            <name>extension</name>
            <value><string>jpg</string></value>
        </member>
        <member>
            <name>type_mimes</name>
            <value><array>
                <data><value><struct><member>
                    <name>type_mime</name>
                    <value><string>image/jpeg</string></value>
                </member></struct></value></data>
            </array></value>
        </member>
    </struct></value>
    <value><struct>
        <member>
            <name>extension</name>
            <value><string>m4v</string></value>
        </member>
    </struct>

```

```

<member>
  <name>type_mimes</name>
  <value><array><data>
    <value><struct><member>
      <name>type_mime</name>
      <value><string>video/m4v</string></value>
    </member></struct></value>
    <value><struct><member>
      <name>type_mime</name>
      <value><string>video/x-m4v</string></value>
    </member></struct></value>
  </data></array></value>
</member>
</struct></value>
<value><struct>
  <member>
    <name>extension</name>
    <value><string>mpeg</string></value>
  </member>
  <member>
    <name>type_mimes</name>
    <value><array><data>
      <value><struct><member>
        <name>type_mime</name>
        <value><string>video/mp2p</string></value>
      </member></struct></value>
      <value><struct><member>
        <name>type_mime</name>
        <value><string>video/mpeg</string></value>
      </member></struct></value>
      <value><struct><member>
        <name>type_mime</name>
        <value><string>video/msvideo</string></value>
      </member></struct></value>
      <value><struct><member>
        <name>type_mime</name>
        <value><string>video/x-msvideo</string></value>
      </member></struct></value>
    </data></array></value>
  </member>
</struct></value>
</data></array></value></param>
</params>
</methodResponse>

```

UPLOADBINARY

Method used to upload the binary file to save in an album with the method saveUploadedFile.

Method Name : [joomeo.user.uploadBinary](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
data (Required)	base64	The binary data in base64 corresponding to the binary serialized file
filename (Required)	string	The name of the binary file with the extension
size (Required)	int	The size of the binary data (in byte)

Returns : An uploadid used in the method saveUploadedFile

Structure :

Member Name	Type	Description
uploadid	string	Temporary identifiant of the uploaded file

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.uploadBinary</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
      <member><name>sessionid</name>
      <value><string>0f16a26a812a1621b673211123501af9</string></value>
      </member>
      <member><name>data</name>
      <value><base64>binary data</base64></value>
      </member>
      <member><name>filename</name>
      <value><string>filename.jpg</string></value>
      </member>
      <member><name>size</name>
      <value><int>1277900</int></value>
      </member>
    </struct></value>
  </param>
</params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
  <param><value><struct>
    <member><name>uploadid</name>
    <value><string>1227616985_419</string></value>
```

```
        </member>
      </struct></value></param>
</params>
</methodResponse>
```

USER.COLLECTION

All the method of this class are deprecated. Use joomeo.user.album instead.

USER.ALBUM

GETFILESLIST

Method to get the list of the photos contained in given album.

Method Name : [joomeo.user.album.GetFilesList](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id
orderby	string	"date" ,"name" or "manual" - (date by default)
random	int	Generate a Random result sort order : 0 (No) - 1 (Yes) - (0 by default)
startresult	int	Start index to get files list (used if maxresult >0) (0 byDefault)
maxresult	int	Number of photos to return : 0 or N - (0 - Unlimited by default)

Returns : The list of photos of the corresponding album.

Array of Structure :

Member name	Type	Description
fileid	string	Identifiant of the file
filename	string	Name of the file
rotation	int	rotation in degree
rating	int	rate of the picture
type_mime	string	Mime type of the file
size	int	size in bytes
width	int	width of the picture
height	int	height of the picture
date_shooting	double	shooting date of the picture
date_creation	double	creation date of the picture
allowdownload	int	0 (No) - 1 (Yes)
legend	string	caption of the photo
albumid	string	Parent album Id

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<methodCall>
```

```
  <methodName>joomeo.user.album.GetFilesList</methodName>
```

```
  <params>
```

```
    <param>
```

```
      <value><struct>
```

```
        <member>
```

```
          <name>apikey</name>
```

```
          <value><string>12345a12345b12345c12345d</string></value>
```

```
        </member>
```

```

<member>
  <name>sessionid</name>
  <value><string>07fcec8470e06fba7e7326d74e9b8865</string></value>
</member>
<member>
  <name>albumid</name>
  <value><string>MmUzN2M3ZWS1Q63BmNKHuA%3D%3D%0D%0A</string></value>
</member>
<member>
  <name>orderby</name>
  <value><string>date</string></value>
</member>
<member>
  <name>random</name>
  <value><int>1</int></value>
</member>
<member>
  <name>maxresult</name>
  <value><int>3</int></value>
</member>
</struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><array>
        <data>
          <value><struct>
            <member>
              <name>fileid</name>
              <value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></
value>
            </member>
            <member>
              <name>filename</name>
              <value><string>AnnaMarie.jpg</string></value>
            </member>
            <member>
              <name>rotation</name>
              <value><int>0</int></value>
            </member>
            <member>
              <name>rating</name>

```

```

        <value><int>0</int></value>
    </member>
    <member>
        <name>type_mime</name>
        <value><string>image/jpeg</string></value>
    </member>
    <member>
        <name>size</name>
        <value><int>597810</int></value>
    </member>
    <member>
        <name>width</name>
        <value><int>1440</int></value>
    </member>
    <member>
        <name>height</name>
        <value><int>900</int></value>
    </member>
    <member>
        <name>date_shooting</name>
        <value><double>1.085775444E+12</double></value>
    </member>
    <member>
        <name>date_creation</name>
        <value><double>1.218201944E+12</double></value>
    </member>
    <member>
        <name>allowdownload</name>
        <value><int>1</int></value>
    </member>
    <member>
        <name>legend</name>
        <value><string>caption of the photo</string></value>
    </member>
    <member>
        <name>albumid</name>
        <value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></
value>
    </member>
</struct></value>
</param>
</params>
</methodResponse>

```

GETNUMBEROFFILES

Method to get the number of the photos contained in given album.

Method Name : joomeo.user.album.getNumberOfFiles

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id

Returns : The number of photos of the corresponding album.

Structure :

Member Name	Type	Description
nbfiles	int	Number of files in the album

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.album.getNumberOfFile</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
        <member><name>sessionid</name>
        <value><string>0f16a26a812a1621b673211123501af9</string></value>
      </member>
        <member><name>albumid</name>
        <value><string>OTQxMDRiMDZAzRHI8NYTkW%3D%3D%0D%0A</string></value>
      </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
  <param><value><struct>
    <member><name>nbfiles</name>
    <value><int>147</int></value>
  </member>
  </struct></value></param>
</params>
</methodResponse>
```

SAVEUPLOADEDFILE

Method to save an uploaded file in a album.

Method Name : [joomeo.user.album.saveUploadedFile](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id
uploadid (Required) upload.php	string	Upload Id return by the method uploadBinary or by the PHP script upload.php
fileName (Required)	string	name of the uploaded file (with extension)
legend	string	caption of the photo

Returns : The file id of the added file.

Structure :

Member Name	Type	Description
fileid	string	Identifiant of the file
filename	string	Name of the file
rotation	int	rotation in degree
rating	int	rate of the picture
type_mime	string	Mime type of the file
size	int	size in bytes
width	int	width of the picture
height	int	height of the picture
date_shooting	double	shooting date of the picture
date_creation	double	creation date of the picture
allowdownload	int	0 (No) - 1 (Yes)
legend	string	caption of the photo
albumid	string	Parent album Id

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.album.saveUploadedFile</methodName>
  <params>
    <param>
      <value><struct>
        <member><name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
        <member><name>sessionid</name>
        <value><string>0f16a26a812a1621b673211123501af9</string></value>
      </member>
        <member><name>albumid</name>
        <value><string>OTQxMDRiMDZAzRHI8NYTkW%3D%3D%0D%0A</string></value>
      </member>
        <member><name>uploadid</name>
        <value><string>1227616985_419</string></value>
      </member>
        <member><name>fileName</name>
        <value><string>filename.jpg</string></value>
      </member>
    </param>
  </params>
</methodCall>
```

```

        </member>
        <member><name>legend</name>
        <value><string>caption of the photo</string></value>
        </member>
    </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
<params>
    <param><value><struct>
        <member><name>fileid</name>
        <value><string>MmUzN2M3ZWTOwiAp9ts3%2BA%3D%3D%0D%0A</string></
value>
        </member>
        <member>
            <name>filename</name>
            <value><string>BlueridgeMountains.jpg</string></value>
        </member>
        <member>
            <name>rotation</name>
            <value><int>0</int></value>
        </member>
        <member>
            <name>rating</name>
            <value><int>0</int></value>
        </member>
        <member>
            <name>type_mime</name>
            <value><string>image/jpeg</string></value>
        </member>
        <member>
            <name>size</name>
            <value><int>1331154</int></value>
        </member>
        <member>
            <name>width</name>
            <value><int>2560</int></value>
        </member>
        <member>
            <name>height</name>
            <value><int>1600</int></value>
        </member>
        <member>
            <name>date_shooting</name>
            <value><double>1.214466532E+12</double></value>

```

```

</member>
<member>
  <name>date_creation</name>
  <value><double>1.218201862E+12</double></value>
</member>
<member>
  <name>allowdownload</name>
  <value><int>1</int></value>
</member>
<member>
  <name>legend</name>
  <value><string>caption of the photo</string></value>
</member>
<member>
  <name>albumid</name>
  <value><string>MmUzN2M3ZWTRvA5sYBbJOw%3D%3D%0D%0A</string></value>
</member>
</struct></value></param>
</params>
</methodResponse>

```

UPDATE

Method used to update a given album.

This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.album.update](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id
label	string	Name of the album
previewSize by default)	int	Maximum size of the slideshow and the previews (1920)
albumDate	double	Date of the album (date of the day by default) (unix timestamp)
public	int	1 to add an album accessible by all the Joomeo space's contacts, 0 to add a private album (0 by default)

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - album updated with success

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.album.update</methodName>

```

```

<params>
  <param>
    <value><struct>
      <member>
        <name>apikey</name>
        <value><string>12345a12345b12345c12345d</string></value>
      </member>
      <member>
        <name>sessionid</name>
        <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
      </member>
      <member>
        <name>albumid</name>
        <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</
string></value>
      </member>
      <member>
        <name>label</name>
        <value><string>my album</string></value>
      </member>
      <member>
        <name>previewSize</name>
        <value><int>1920</int></value>
      </member>
      <member>
        <name>albumDate</name>
        <value><double>1.214466532E+12</double></value>
      </member>
      <member>
        <name>public</name>
        <value><int>1</int></value>
      </member>
    </struct></value>
  </param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><struct>
        <member>
          <name>success</name>
          <value><int>1</int></value>
        </member>
      </struct></value>
    </param>

```

```
</params>
</methodResponse>
```

DELETE

Method used to delete an album from the Joomeo space of the current user session. This method is allowed with a sessionType equals to 0.

Method Name : `joomeo.user.album.update`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - album deleted with success

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.album.delete</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
        </member>
        <member>
          <name>albumid</name>
          <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</
string></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
```

```

<params>
  <param>
    <value><struct>
      <member>
        <name>success</name>
        <value><int>1</int></value>
      </member>
    </struct></value>
  </param>
</params>
</methodResponse>

```

DELETEFILE

Method used to delete a file from a given album.
This method is allowed with a sessionType equals to 0.

Method Name : `joomeo.user.album.deleteFile`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Parent Album Id
fileid (Required)	string	File Id of the photo

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - file deleted with success

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.album.deleteFile</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
        </member>
        <member>
          <name>albumid</name>

```

```

        <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</
string></value>
    </member>
    <member>
        <name>fileid</name>
        <value><string>YmRiNWE4NzfjHivguObbFw%3D%3D%0D%0A</
string></value>
    </member>
</struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
    <params>
        <param>
            <value><struct>
                <member>
                    <name>success</name>
                    <value><int>1</int></value>
                </member>
            </struct></value>
        </param>
    </params>
</methodResponse>

```

ALLOWCONTACTACCESS

Method used to allow a contact to access to a given album.
This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.album.allowContactAccess](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id
contactid (Required)	string	Contact id to be allowed to access to the album
allowdownload	int	allow contact to download original file from the album
allowupload	int	allow contact to upload new files into the album

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - file deleted with success

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.album.allowContactAccess</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
        </member>
        <member>
          <name>albumid</name>
          <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</
string></value>
        </member>
        <member>
          <name>contactid</name>
          <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</
string></value>
        </member>
        <member>
          <name>allowdownload</name>
          <value><int>1</int></value>
        </member>
        <member>
          <name>allowupload</name>
          <value><int>0</int></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><struct>
        <member>
          <name>success</name>
          <value><int>1</int></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodResponse>

```

```

        </param>
    </params>
</methodResponse>

```

REMOVECONTACTACCESS

Method used to remove the contact access to a given album.

This method is allowed with a sessionType equals to 0.

Method Name : `joomeo.user.album.removeContactAccess`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id
contactid (Required)	string	Contact id to be allowed to access to the album

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - file deleted with success

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
    <methodName>joomeo.user.album.removeContactAccess</methodName>
    <params>
        <param>
            <value><struct>
                <member>
                    <name>apikey</name>
                    <value><string>12345a12345b12345c12345d</string></value>
                </member>
                <member>
                    <name>sessionid</name>
                    <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
                </member>
                <member>
                    <name>albumid</name>
                    <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</
string></value>
                </member>
                <member>
                    <name>contactid</name>
                    <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</
string></value>
                </member>
            </struct></value>
        </param>
    </params>
</methodCall>

```

```

        </param>
    </params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
    <params>
        <param>
            <value><struct>
                <member>
                    <name>success</name>
                    <value><int>1</int></value>
                </member>
            </struct></value>
        </param>
    </params>
</methodResponse>

```

GETALLOWEDCONTACTACCESS

Method used to get the contact list who are allowed to access to a given album. This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.album.getAllowedContactAccessList](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
albumid (Required)	string	Album Id

Returns : The list of the allowed contact id and rules to access for a given albumid.

Array of Structure :

Member name	Type	Description
contactid	string	Identifiant of the contact
allowdownload	int	allow contact to download original file from the album
allowupload	int	allow contact to upload new files into the album

Request format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName> joomeo.user.album.getAllowedContactAccessList</methodName>
    <params>
        <param>
            <value><struct>
                <member>
                    <name>apikey</name>
                    <value><string>12345a12345b12345c12345d</string></value>

```

```

        </member>
        <member>
            <name>sessionid</name>
            <value><string>50d289fd9bca5ce9841f4c3b586c0646</string></value>
        </member>
        <member>
            <name>albumid</name>
            <value><string>YmRiNWE4NzljHivguObbFw%3D%3D%0D%0A</
string></value>
        </member>
    </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
    <params>
        <param><value><array><data>
            <value><struct>
                <member>
                    <name>contactid</name>
                    <value><string>MmUzN2M3ZWRQqBxaKXd6mw%3D%3D%0D%0A</
string></value>
                </member>
                <member>
                    <name>allowdownload</name>
                    <value><int>1</int></value>
                </member>
                <member>
                    <name>allowupload</name>
                    <value><int>1</int></value>
                </member>
            </struct></value>
        <value><struct>
            <member>
                <name>contactid</name>
                <value><string>MmUzN2M3ZWRQqBxaKXd6mw%3D%3D%0D%0A</
string></value>
            </member>
            <member>
                <name>allowdownload</name>
                <value><int>0</int></value>
            </member>
            <member>
                <name>allowupload</name>
                <value><int>1</int></value>
            </member>
        </struct>
    </param>

```

```
        </struct></value>
      </data></array></value></param>
    </params>
  </methodResponse>
```

USER.CONTACT

UPDATE

Method used to update a given contact.

This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.contact.update](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
contactid (Required)	string	Contact Id to update
email (Required)	string	A valid address email
firstname (Required)	string	the firstname of the contact to update
lastname (Required)	string	the lastname of the contact to update
login (Required)	string	the username used to connect to the Joomeo space
password (Required)	string	the password used to connect to the Joomeo space

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - if contact updated with success

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.contact.update</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>0f16a26a812a1621b673211123501af9</string></value>
        </member>
        <member>
          <name>contactid</name>
          <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</
string></value>
        </member>
        <member>
          <name>email</name>
          <value><string>john.doe@doe.com</string></value>

```

```

        </member>
        <member>
            <name>firstname</name>
            <value><string>john</string></value>
        </member>
        <member>
            <name>lastname</name>
            <value><string>Doe</string></value>
        </member>
        <member>
            <name>login</name>
            <value><string>john.doe</string></value>
        </member>
        <member>
            <name>password</name>
            <value><string>wxAs78</string></value>
        </member>
    </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
    <params>
        <param>
            <value><struct>
                <member>
                    <name>success</name>
                    <value><int>1</int></value>
                </member>
            </struct></value>
        </param>
    </params>
</methodResponse>

```

DELETE

Method used to delete a given contact.

This method is allowed with a sessionType equals to 0.

Method Name : `joomeo.user.contact.delete`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
contactid (Required)	string	Contact Id to update

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - if contact deleted with success

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
<methodName>joomeo.user.contact.update</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>0f16a26a812a1621b673211123501af9</string></value>
        </member>
        <member>
          <name>contactid</name>
          <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</
string></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><struct>
        <member>
          <name>success</name>
          <value><int>1</int></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodResponse>
```

SENDINVITE

Method used to send an email to the contact. This email is automatically filled with the access information to the Joomeo space associated to the user session.

This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.contact.sendInvite](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
contactid (Required)	string	Contact id (get with joomeo.user.getContactList method)
subject	string	Your personal subject for the email
message	string	Your personal message attached to the email
lang	string	The language used to write the email (fr or en). If no data the user preference will be used.
sendCopy	int	0 (by default) or 1 (1 to send a copy of this email to the manager and 0 to don't send it)

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - email sented with success

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<methodCall>
```

```
  <methodName>joomeo.user.contact.sendInvite</methodName>
```

```
  <params>
```

```
    <param>
```

```
      <value><struct>
```

```
        <member>
```

```
          <name>apikey</name>
```

```
          <value><string>12345a12345b12345c12345d</string></value>
```

```
        </member>
```

```
        <member>
```

```
          <name>sessionid</name>
```

```
          <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
```

```
        </member>
```

```
        <member>
```

```
          <name>contactid</name>
```

```
          <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</
```

```
string></value>
```

```
        </member>
```

```
        <member>
```

```
          <name>subject</name>
```

```
          <value><string>Invitation to my personal Joomeo space</string></value>
```

```
        </member>
```

```
      </member>
```

```
<name>message</name>
<value><string>Hi,
```

Please, connect you to my Joomeo space to watch my holiday's album !

Regards,

John Doe</string></value>

```
</member>
```

```
<member>
```

```
<name>lang</name>
```

```
<value><string>en</string></value>
```

```
</member>
```

```
<member>
```

```
<name>sendCopy</name>
```

```
<value><int>1</int></value>
```

```
</member>
```

```
</struct></value>
```

```
</param>
```

```
</params>
```

```
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<methodResponse>
```

```
<params>
```

```
<param>
```

```
<value><struct>
```

```
<member>
```

```
<name>success</name>
```

```
<value><int>1</int></value>
```

```
</member>
```

```
</struct></value>
```

```
</param>
```

```
</params>
```

```
</methodResponse>
```

USER.FILE

DELETE

Method used to delete a file from the Joomeo space of the current user session. All the instances of this files will be deleted from their albums.

This method is allowed with a sessionType equals to 0.

Method Name : [joomeo.user.file.delete](#)

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
fileid (Required)	string	File Id of the photo

Returns : Success or error

Structure :

Member Name	Type	Description
success	int	1 - file deleted with success

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodCall>
  <methodName>joomeo.user.file.delete</methodName>
  <params>
    <param>
      <value><struct>
        <member>
          <name>apikey</name>
          <value><string>12345a12345b12345c12345d</string></value>
        </member>
        <member>
          <name>sessionid</name>
          <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
        </member>
        <member>
          <name>fileid</name>
          <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</
string></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodCall>
```

Response format

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
```

```

    <param>
      <value><struct>
        <member>
          <name>success</name>
          <value><int>1</int></value>
        </member>
      </struct></value>
    </param>
  </params>
</methodResponse>

```

GETBINARY

Method used to download the binary file of a photo contained in given album.

If parameter albumid is Null the method will take into account the current album id of the photo.

Method Name : `joomeo.user.file.getBinary`

Arguments

Name	Type	Description
apikey (Required)	string	Your application API key.
sessionid (Required)	string	Session Id
fileid (Required)	string	File Id of the photo
collectionid	string	Parent Album Id (this member is deprecated, Use albumid instead)
albumid	string	Parent Album Id
type (Required)	string	Format type : "large" (by default) - size from 720x720 to 1920x1920, "original" - original file, "small" - size 240x240, "medium" - size 500x500, "degraded" - size 50x50

Returns : The binary data in base64 corresponding to the binary serialized photo.

Structure :

Member Name	Type	Description
data	base64	Binary data
mime_type	string	Mime type of the file
size	int	size in bytes of the file

Request format

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<methodCall>
```

```
  <methodName>joomeo.user.file.getBinary</methodName>
```

```
  <params>
```

```
    <param>
```

```
      <value><struct>
```

```
        <member><name>apikey</name>
```

```
        <value><string>12345a12345b12345c12345d</string></value>
```

```

    </member>
    <member><name>sessionid</name>
    <value><string>2aa89193aca45ace5995d905a4e45707</string></value>
    </member>
    <member><name>fileid</name>
    <value><string>YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A</string></value>
    </member>
    <member><name>albumid</name>
    <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</string></value>
    </member>
    <member><name>collectionid</name>
    <value><string>YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A</string></value>
    </member>
    <member><name>type</name>
    <value><string>large</string></value>
    </member>
  </struct></value>
</param>
</params>
</methodCall>

```

Response format

```

<?xml version="1.0" encoding="UTF-8"?>
<methodResponse>
  <params>
    <param>
      <value><struct>
        <member><name>data</name>
        <value><base64>binary data</base64></value>
      </member>
      <member><name>mime_type</name>
      <value><string>image/jpeg</string></value>
    </member>
    <member><name>size</name>
    <value><int>304432</int></value>
  </member>
  </struct></value>
    </param>
  </params>
</methodResponse>

```

FILE.PHP

Script PHP used to download a photo directly from its URL.

Arguments

Name	Type	Description	Method
apikey (Required)	string	Your application API key.	GET
sessionid (Required)	string	Session Id	GET
fileid (Required)	string	File Id of the photo	GET
collectionid instead)	string	Albumid Id (this member is deprecated, use albumid instead)	GET
albumid	string	Albumid Id	GET
type (Required) default (large)	string	Format type : "original", "large", "small", "medium", "degraded" by default	GET
rotation	int	photo rotation in degrees (0, 90, 180 or 270)	GET

Returns : The data file.

URL access

<http://api.joomeo.com/file.php>

Request format

[http://api.joomeo.com/file.php?](http://api.joomeo.com/file.php?apikey=12345a12345b12345c12345d&sessionid=2aa89193aca45ace5995d905a4e45707&fileid=YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A&albumid=YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A&type=large&rotation=0)

[apikey=12345a12345b12345c12345d&sessionid=2aa89193aca45ace5995d905a4e45707&fileid=YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A&albumid=YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A&type=large&rotation=0](http://api.joomeo.com/file.php?apikey=12345a12345b12345c12345d&sessionid=2aa89193aca45ace5995d905a4e45707&fileid=YmRiNWE4NzFjHivguObbFw%3D%3D%0D%0A&albumid=YmRiNWE4NzeG5zZzirOmLQ%3D%3D%0D%0A&type=large&rotation=0)

UPLOAD.PHP

Script PHP used to upload a file.

Arguments

Name	Type	Description	Method
apikey (Required)	string	Your application API key.	GET
sessionid (Required)	string	Session Id	GET

Posted data

upload.php must receive a request POST with the following parameters :

- an element Content-Type with multipart/form-data
- an element Content-Disposition with the attribut name defined on "Filedata" and an attribut name defined on the original file name
- the binary data of the file to upload

Returns : An upload id used in the method saveUploadedFile

URL access

<http://api.joomeo.com/upload.php>

Request format

<http://api.joomeo.com/upload.php?apikey=12345a12345b12345c12345d&sessionid=2aa89193aca45ace5995d905a4e45707>

Posted data example

```
Content-Type: multipart/form-data; boundary=AaB03x
--AaB03x
Content-Disposition: form-data; name="Filedata"; filename = "example.jpg"
Content-Type: application/octet stream
... contents of example.jpg ...
--AaB03x--
```

Response format

```
Success :
<uploadid>1227616985_419</uploadid>
Error :
<error>200</error>
```

API ERRORS CODES

100: INVALID XML RPC OBJECT

The passed signature was invalid.

101: INVALID XML RPC METHOD SIGNATURE

The passed signature was invalid.

102: UNKNOWN XML RPC METHOD

Given method doesn't exist.

200: INVALID API KEY

The API key passed is not valid.

201: API KEY DO NOT EXIST

The API key passed doesn't exist.

202 : SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while initializing the user session.

203 : SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while initializing the contact session.

204 : SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or can't create a session for the contact.

205 : SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or can't get the session properties.

206 : SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or is not allowed to make this action.

299 : SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while killing the user session.

300: LOGIN / AUTHENTICATION FAILED.

Invalid connection parameters

301: MISSING / EMPTY REQUIRED PARAMETER.

Missing required parameter param

400: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or can't Initialize the user album

401: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or can't get the user albums list

402: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or can't Initialize the files list for the album

403: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or can't get the user hubs list

404: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while trying to get the files list

405: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while trying to get the binary data file

406: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while trying to add a new album

407: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while trying to add a file into an album

407001: INVALID PARAMETER UPLOADID

The uploadid is not valid

407002: FILE FORMAT ERROR

The uploaded file format is not supported

407003: INVALID PARAMETER COLLECTIONID

The albumid is not valid

407004: MONTHLY UPLOAD LIMIT IS REACHED

The Joomeo space has reached its monthly upload limit

408: SERVICE CURRENTLY UNAVAILABLE

The API service is currently unavailable or an error occurred while trying to upload binary data

408001: SAVE UPLOADED FILE ERROR

An error occurred while trying to save uploaded file in temporary directory

408002: MONTHLY UPLOAD LIMIT IS REACHED

The Joomeo space has reached its monthly upload limit

408003: FILE FORMAT ERROR

The uploaded file format is not supported

409: ERROR WHILE GETTING ALLOWED FILE TYPE TO UPLOAD

An error occurred while trying to get allowed file type list to upload

410: ERROR WHILE GETTING JOOME0 SPACE STATISTICS

An error occurred while trying to get statistics from the Joomeo space of the user session

411: ERROR WHILE DELETING THE ALBUM

An error occurred while trying to delete the album from the Joomeo space of the user session

412: ERROR WHILE DELETING THE FILE

An error occurred while trying to delete the file from its album

413: ERROR WHILE DELETING THE FILE

An error occurred while trying to delete the file from the Joomeo space of the user session

414: ERROR WHILE GETTING THE CONTACTS LIST

An error occurred while getting the contact list of the Joomeo space of the user session

415: ERROR WHILE ADDING CONTACT

An error occurred while adding the contact into the Joomeo space of the user session

415001: THE LOGIN AND PASSWORD ARE ALREADY USED FOR ANOTHER CONTACT

The login and password are already used for another contact

416: ERROR WHILE TRYING TO SEND AN EMAIL TO THE CONTACT

An error occurred while trying to send an email to the contact

417: ERROR WHILE TRYING TO ALLOW CONTACT TO ACCESS TO THE ALBUM

An error occurred while trying to allow contact to access to the album

418: ERROR WHILE TRYING TO REMOVE THE CONTACT ACCESS TO THE ALBUM

An error occurred while trying to remove the contact access to the album

419: ERROR WHILE TRYING TO GET THE CONTACT LIST WHO ARE ALLOWED TO ACCESS TO THE ALBUM

An error occurred while trying to get the contact list who are allowed to access to the album

420: ERROR WHILE TRYING TO UPDATE THE CONTACT

An error occurred while trying to update the contact

420001: A CONTACT WITH THIS EMAIL ALREADY EXIST

A contact with this email already exist

420002: A CONTACT WITH THIS LOGIN AND PASSWORD ALREADY EXIST

A contact with this login and password already exist

ANNEXES

(1) XML-RPC

XML-RPC is a format devised by Userland Software for achieving remote procedure call via XML using HTTP as the transport. XML-RPC has its own web site, www.xmlrpc.com.



Joomeo



airsens™

www.airsens.com
www.joomeo.com

contact@joomeo.com