

Weblife 1 API Doku

Die Weblife 1 API ist eine, Programmierschnittstelle, die den Zugang zum Weblife1 Framework für externe Applikationen ermöglicht.

Um es einer Applikation zu ermöglichen, zugriff auf das Weblife1 Framework zu haben, ist die Anmeldung eines neuen API Keys notwendig.

Um einen API Key zu erlangen ist ein Weblife1 User notwendig.
Es wird darauf hingewiesen, sich für jede Applikation einen neuen Key zu registrieren.

Zur Dokumentation:

Es wird unterschieden zwischen userbasierter und nicht userbasierter Aufrufe. Für userbasierte Aufrufe wird ein sogenannter "authtoken" benötigt.

Dieser authtoken wird über User - Section / Login - Methode geholt und muss bei allen Aufrufen, die diesen benötigen, mitgegeben werden.

In der Dokumentation vorkommende Beispiele sind in PHP 5.1 geschrieben.

Dokumentation für **Version: 1.14**

Spezifikationen:

Jeder neu angelegte Applikation wird auf 500 Requests pro.
Diese Limitierung kann nach Rücksprache mit den Administratoren erhöht werden.

Anforderungsformate:

REST:

REST ist das am einfachsten zu verwendende Anforderungsformat. Hierbei handelt es sich um eine einfache HTTP GET- oder POST - Aktion.

Die URL für REST Aufrufe lautet `http://rest.api.weblife1.com/`.

Das grundsätzliche Datenformat bei REST ist XML. Befinden sich in einer Antwortvariable HTML bzw. andere Sonderzeichen, so wird dieser Tag als CDATA ausgeliefert.

Anforderungen an die REST API über HTTP GET sind nach folgendem System aufgebaut:

```
http://rest.api.weblife1.com/section/methode/param1/wert1/param2/wert2
```

Um das Section „test“ mit der Methode „echo“ über GET Parameter anzufordern, führen Sie folgenden Aufruf durch:

```
http://rest.api.weblife1.com/test/echo/name/value
```

Anforderungen an die REST API über HTTP POST sind nach folgendem System aufgebaut:

```
http://api.rest.weblife1.com
$_POST['section'] = Sectionname, $_POST['method'] = Methodenname,
$_POST['param1'] = Wert1
```

Um das Section „test“ mit der Methode „echo“ über POST Parameter anzufordern, führen Sie folgenden Aufruf durch:

```
http://rest.api.weblife1.com/
wobei $_POST['section'] = test, $_POST['method'] = echo und $_POST['name']
= value ist
```

Authentifizierung:

Bei jedem Aufruf an die API meldet sich diese beim Weblife1 Framework an. Darum muss jeder Aufruf die Parameter apikey und authsecret enthalten.

authsecret:

Das authsecret ist ein MD5 Hash aus Section, Methode, apikey und apiSecret.

Beispiel:

```
$hash = $section . $methode . $apikey . $apiSecret;  
$authsecret = md5($hash);
```

Generelle Fehler:

Generelle Fehler können bei jedem Aufruf auftreten. Handelt es sich um Aufrufe für die kein auth token notwendig ist, so fallen die Fehler #5 und #104 logischerweise weg.

Auf Grund der Übersichtlichkeit werden diese hier einmal notiert, und in Folge nur mehr auf diese Stelle im Dokument verwiesen.

1	API_GEN_UNKNOWN_ERROR
2	API_GEN_SERVICE_UNAVAILABLE
3	API_GEN_UNKNOWN_CALL
4	API_GEN_TOO_MANY_CALLS
5	API_GEN_AUTH_TOKEN_ERROR
6	API_GEN_TOO_MANY_REQUESTS
7	API_GEN_KEY_LOCKED
100	API_APP_INVALID_PARAM
101	API_APP_MISSING_PARAM
102	API_APP_INVALID_AUTH_SECRET
103	API_APP_INVALID_API_KEY
104	API_APP_INVALID_AUTH_TOKEN
105	API_APP_NO_ACCESS

Section Test:

Method echo:

Section: test
Method: echo

Beschreibung:

Gibt die übergebenen Parameter im XML Format (<key>value</key>) aus.

Beachte: Bei jedem Aufruf müssen der apikey und das authsecret mitgesendet werden.

Darum erscheinen diese auch in der XML Antwort vom Server.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

zusätzlich kann eine (unbegrenzte) Anzahl an Parametern angegeben werden, deren Werte in der XML Antwort vom Server enthalten sind.

Beispielantwort:

```
<test>
  <apikey>myApikey</authtoken>
  <authsecret>a5d2h365h7w98765werh2567fe3aa67e</authsecret>
  <parameter1>value1</parameter1>
  <parameter2>value2</parameter2>
</test>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section System:

Method info:

Section: system
Method: info

Beschreibung:

Diese Methode liefert den aktuelle Systeminformationsblock zurück und informiert den Entwickler über die aktuelle Version und die letzte Änderung der API.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret

Beispielantwort:

```
<system>
  <name>Weblife1 API</name>
  <version>1.5</version>
  <lastchange>2009/09/16 16:18:55.</lastchange>
</system>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section user:

Method login:

Section: user
Method: login

Beschreibung:

Loggt den User ein, und gibt für userbasierte Aufrufe den benötigten authtoken zurück.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
username: der Username des Users
password: das Passwort des Users (md5 Verschlüsselt)

Beispielantwort:

```
<user>
  <authtoken>94d6e59a058519b1c0bed5b3d307b760</authtoken>
  <userid>2069</userid>
  <username>shr3k</username>
</user>
```

Fehlercodes:

200	API_USER_INVALID_USER_OR_PASS
201	API_USER_INVALID_USER
202	API_USER_INVALID_PASS
203	API_USER_USER_LOCKED
204	API_USER_USER_DELETED

Method logout:

Section: user
Method: logout

Beschreibung:

Loggt den User aus (authtoken ist danach nicht mehr verfügbar)

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method getfriendsstatus:

Section: user
Method: getfriendsstatus

Beschreibung:

Liefert die letzten Statusmeldungen der Freunde des eingeloggten (authtoken) Users zurück.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
statusage: Maximales Alter der Statusmeldungen
limit: Anzahl der Statusmeldungen

Beispielantwort:

```
<user>
  <statuscount>2</statuscount>
  <status>
    <statusid>12345</statusid>
    <username>snog</username>
    <userid>12034</userid>
    <message>liegt am Pool</message>
    <commentcount>0</commentcount>
    <time>1249295542</time>
  </status>
  <status>
    <statusid>12346</statusid>
    <username>tonimaroni</username>
    <userid>20111</userid>
    <message>auf dem Weg ins Kino</message>
    <commentcount>4</commentcount>
    <time>1249286142</time>
  </status>
</user>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method getcurrentstatus:

Section: user
Method: getcurrentstatus

Beschreibung:

Liefert die aktuelle Statusmeldung des angegebenen Users zurück.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
username: der Username des Benutzers dessen Statusmeldung man haben möchte.

Beispielantwort:

```
<user>
  <statuscount>1</statuscount>
  <status>
    <statusid>12345</statusid>
    <username>snog</username>
    <userid>12034</userid>
    <message>liegt am Pool</message>
    <commentcount>4</commentcount>
    <time>1249286142</time>
  </status>
</user>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method updatestatus:

Section: user
Method: updatestatus

Beschreibung:

Ändert die Statusmeldung des aktuell eingeloggten Users

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
message: Neue Statusnachricht

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method resetstatus:

Section: user
Method: resetstatus

Beschreibung:

Löscht den aktuellen Status.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method getstatuscomments:

Section: user
Method: getstatuscomments

Beschreibung:

Liefert die Kommentare zur angegebenen Statusmeldung zurück.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
statusid: ID der Statusmeldung
limit: Anzahl der Kommentare die zurückgeliefert werden sollen

Beispielantwort:

```
<user>
  <commentcount>2</commentcount>
  <statuscomment>
    <username>shr3k</username>
    <userid>2069</userid>
    <time>1249291995</time>
    <comment>find ich cool</comment>
  </statuscomment>
  <statuscomment>
    <username>snog</username>
    <userid>12034</userid>
    <time>1249291995</time>
    <comment>macht Spass</comment>
  </statuscomment>
</user>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method addstatuscomment:

Section: user
Method: addstatuscomment

Beschreibung:

Fügt zu einer Statusmeldung einen Kommentar hinzu.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
statusid: ID der Statusmeldung
comment: Kommentar der hinzugefügt werden soll

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method getuserpagethumburl:

Section: user
Method: getuserpagethumburl

Achtung:

Diese Methode ist als deprecated markiert. Sie wird in zukünftigen Versionen nicht mehr unterstützt. Verwenden sie anstatt dieser Method die Methode getuserimage.

Beschreibung:

Liefert die URL für das Thumbnailfoto von der Userpage.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
userid: ID der Users

Beispielantwort:

```
<user>  
  <thumburl>http://80.66.45.86:8001/uptn/1/0/uptn_1.jpg</thumburl>  
</user>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method `getuserpageimage`:

Section: user
Method: `getuserpageimage`

Beschreibung:

Liefert die URL für das Userbild als Thumbnail und als Normalansicht und als Original.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

`apikey`: Ihr API Key
`authsecret`: das `authsecret`
`userid`: ID der Users

Beispielantwort:

```
<user>
  <thumburl>http://80.66.45.86:8001/uptn/1/0/uptn_1.jpg</thumburl>
  <bigurl>http://80.66.45.86:8001/upph/12034/0/upph_12034.jpg</bigurl>
  <origurl>http://80.66.45.86:8001/upor/12034//upor_12034.jpg</origurl>
</user>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method `getuserpagedata`:

Section: user
Method: `getuserpagedata`

Beschreibung:

Liefert die Daten die auf der Userpage zu finden sind.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

`apikey`: Ihr API Key
`authsecret`: das `authsecret`
`userid`: ID der Users

Beispielantwort:

```
<user>
  <username>sepl_3432</username>
  <gender>male</gender>
  <aboutme>
<![CDATA[Ich bin der coole Type von nebenan.
<br>Meld dich bei mir.
<br>-----
<br>]]>
  </aboutme>
  <age>23</age>
  <size>185</size>
```

```
<weight>80</weight>
<haircolor>braun</haircolor>
<hairlength>ohrenlang</hairlength>
<eyecolor>blau</eyecolor>
<lovestate>--</lovestate>
<orientation>--</orientation>
<location>Parties</location>
<job>Angestellter</job>
<type>--</type>
</user>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section pn:

Zum primären Prüfen auf neue Nachrichten soll die Methode check verwendet werden, da diese Aufrufen in die Tageslimits nicht mit eingerechnet werden und weiters die Server wesentlich weniger belasten als zyklische Checks auf getunreadpns.

Private Nachrichten (PNs) können unterschiedlichen Typs sein:

PN Typ	PHP Konstante	Beschreibung
1	TYPE_SINGLE	PN an EINEN Empfänger
2	TYPE_MULTI	PN an MEHRERE Empfänger
3	TYPE_GROUP	Gruppen PN
4	TYPE_MASS	Massen PN
5	TYPE_SYSTEM	System PN
6	TYPE_CHAT	Chateinladung

Method check:

Section: pn
Method: check

Beschreibung:

Holt die Anzahl der Neuen und die Anzahl der ungelesenen PN's.
Sobald der User den Posteingang aufruft, sind die neuen PN's nur mehr als Ungelesene markiert. Sobald eine PN geöffnet wird ist sie weder neu noch ungelesen.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
username: der Username für den die PNs gecheckt werden (muss mit dem authtoken übereinstimmen!)

Beispielantwort:

```
<pn>  
  <newpns>2</newpns>  
  <unreadpns>5</unreadpns>
```

</pn>

Fehlercodes:

Hier können die generellen Fehler auftreten.

Method getfolders:

Section: pn
Method: getfolders

Beschreibung:

Holt die Order im PN Bereich des angegebenen Users.

Beachte: Jeder User hat standardmäßig einen Posteingang und einen Postausgang, sowie einen Papierkorb für den Posteingang und einen Papierkorb für den Postausgang.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)

Beispielantwort:

```
<pn>
  <folder>
    <folderid>0</folderid>
    <name>inbox</name>
    <type>in</type>
    <trash>0</trash>
  </folder>
  <folder>
    <folderid>54</folderid>
    <name>persönlicher Eingangsortner</name>
    <type>in</type>
    <trash>0</trash>
  </folder>
  <folder>
    <folderid>0</folderid>
    <name>trash</name>
    <type>in</type>
    <trash>1</trash>
  </folder>
  <folder>
    <folderid>0</folderid>
    <name>outbox</name>
    <type>out</type>
    <trash>0</trash>
  </folder>
  <folder>
    <folderid>59</folderid>
    <name>persönlicher Ausgangsortner</name>
    <type>out</type>
    <trash>0</trash>
</pn>
```

```
</folder>
<folder>
  <folderid>0</folderid>
  <name>trash</name>
  <type>out</type>
  <trash>1</trash>
</folder>
</pn>
```

Fehlercodes:

Hier können die generellen Fehler auftreten.

Method readfolder:

Section: pn
Method: readfolder

Beschreibung:

Ruft im angegebenen Ordner (folderid) die Privaten Nachrichten ab. Gibt eine Liste der Nachrichten und die Anzahl der im Ordner enthaltenen Nachrichten zurück.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
folderid: die ID des Ordners
limit [optional]: maximale Anzahl an Nachrichten (default 10)
offset [optional]: der Offset (default 0)
type [optional]: 'in' für Posteingang, 'out' für Postausgang (default 'in')
trash [optional]: '1' Wenn es sich um den Papierkorb handelt, '0' wenn nicht (default 0)

Handelt es sich um einen Typ ,in' Ordner so wird bei der Message das Element „sendername“ als Antwort gesendet, handelt es sich hingegen um einen Typ ,out' Ordner so wird das Element „receivername“ als Antwort gesendet.

Beispielantwort (In-Folder):

```
<pn>
  <message>
    <pnid>49628</pnid>
    <sendername>shr3k</sendername>
    <senderid>2069</senderid>
    <type>1</type>
    <isread>1</isread>
    <message>Text</message>
    <time>1218195446</time>
  </message>
  <message>
    <pnid>49627</pnid>
    <sendername>shr3k</sendername>
    <senderid>2069</senderid>
    <type>1</type>
    <isread>0</isread>
```

```

        <message>Nachricht</message>
        <time>1218195429</time>
    </message>
    <pncount>10</pncount>
</pn>

```

Beispielantwort (Out-Folder):

```

<pn>
  <message>
    <pnid>49628</pnid>
    <receivername>snog</receivername>
    <senderid>12034</senderid>
    <type>1</type>
    <isread>1</isread>
    <message>Text</message>
    <time>1218195446</time>
  </message>
  <message>
    <pnid>49627</pnid>
    <receivername>snog</receivername>
    <senderid>12034</senderid>
    <type>1</type>
    <isread>0</isread>
    <message>Nachricht</message>
    <time>1218195429</time>
  </message>
  <pncount>10</pncount>
</pn>

```

Fehlercodes:

303	API_PN_READ_FOLDER_ERROR
-----	--------------------------

Zusätzlich können die generellen Fehler auftreten

Method `getunreadpn`:

Section: pn
Method: getunreadpn

Beschreibung:

Prüft den Posteingang ob ungelesene PNs vorliegen.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)

Beispielantwort:

```

<pn>
  <pncount>1</pncount>
  <message>
    <pnid>376394017</pnid>
    <sendername>shr3k</sendername>
    <senderid>2069</senderid>

```

```
<type>1</type>
<isread>0</isread>
<message>test</message>
<time>1225195026</time>
</message>
</pn>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method readpn:

Section: pn
Method: readpn

Beschreibung:

Ruft die Private Nachricht mit der gegebenen ID ab.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
pnid: die ID der PN
foldermode [optional]: 'in' für Posteingang, 'out' für Postausgang (default 'in')

Beispielantwort:

```
<pn>
  <pnid>49628</pnid>
  <senderid>2069</senderid>
  <sendername>shr3k</sendername>
  <time>1218195446</time>
  <type>1</type>
  <ishtml>>false</ishtml>
  <message>Text</message>
</pn>
```

Fehlercodes:

302	API_PN_READING_ERROR
-----	----------------------

Zusätzlich können die generellen Fehler auftreten.

Method sendpn:

Section: pn
Method: sendpn

Beschreibung:

Sendet eine Private Nachricht an den gewünschten Usernamen.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
multireceiver: gibt an ob diese Nachricht an mehrere Empfänger gesendet werden soll (true/false)
receiver: der Username des Empfängers, oder bei Multireceiver Pn's mit Komma getrennte Usernamen.
message: die Nachricht der PN

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

300	API_PN_SENDING_ERROR
-----	----------------------

Zusätzlich können die generellen Fehler auftreten.

Method movetotrash:

Section: pn
Method: movetotrash

Beschreibung:

Verschiebt die Private Nachricht mit der angegebenen ID in den Papierkorb. Sind mehrere Nachrichten zu verschieben, so sind diese mit ',' (Komma) zu trennen.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
pnid: die ID der PN (für mehrere IDs diese mit ',' [Komma] trennen)
type: Spezifiziert den Typ der Nachricht Eingang oder Ausgang ("in" oder "out")

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

301	API_PN_DELETE_ERROR
-----	---------------------

Zusätzlich können die generellen Fehler auftreten.

Method deletepn:

Section: pn

Method: deletepn

Beschreibung:

Löscht die Private Nachricht endgültig. Diese Methode kann auf Nachrichten angewendet werden die sich in einem normalen Ordner befinden bzw. auch auf Nachrichten die sich Papierkorb befinden.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

authtoken: der authtoken (von user / login)

pnid: die ID der PN (für mehrere IDs diese mit ',' [Komma] trennen)

type: Spezifiziert den Typ der Nachricht Eingang oder Ausgang ("in" oder "out")

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

301	API_PN_DELETE_ERROR
-----	---------------------

Zusätzlich können die generellen Fehler auftreten.

Method restorepn:

Section: pn

Method: restorepn

Beschreibung:

Stellt eine oder mehrere Nachrichten die in den Papierkorb verschoben wurde wieder her.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

authtoken: der authtoken (von user / login)

pnid: die ID der PN (für mehrere IDs diese mit ',' [Komma] trennen)

type: Spezifiziert den Typ der Nachricht Eingang oder Ausgang ("in" oder "out")

folderid [optional]: Zielordner in den die Nachricht wiederhergestellt wird. Wenn kein Ordner angegeben wird dann wird die Nachricht entweder in den Ausgang oder Eingang hineingeschoben.

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

301	API_PN_DELETE_ERROR
-----	---------------------

Zusätzlich können die generellen Fehler auftreten.

Method createfolder:

Section: pn
Method: createfolder

Beschreibung:
Legt einen Ordner im PN System des Users an.

Authentifizierung:
Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:
apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
name: der Name des Ordners
type: der Typ des Ordners ('in' bzw. 'out')

Beispielantwort:
Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

304	API_PN_CREATE_FOLDER_ERROR
-----	----------------------------

Zusätzlich können die generellen Fehler auftreten.

Method renamefolder:

Section: pn
Method: renamefolder

Beschreibung:
Benennt einen Ordner um

Authentifizierung:
Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:
apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
name: der (neue) Name des Ordners
folderid: die ID des Ordners

Beispielantwort:
Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

305	API_PN_RENAME_FOLDER_ERROR
-----	----------------------------

Zusätzlich können die generellen Fehler auftreten.

Method deletefolder:

Section: pn
Method: deletefolder

Beschreibung:
Löscht einen Ordner (samt Inhalt!)

Authentifizierung:
Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:
apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
folderid: die ID des Ordners
type: der Typ des Ordners ('in' bzw. 'out')

Beispielantwort:
Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

306	API_PN_DELETE_FOLDER_ERROR
-----	----------------------------

Zusätzlich können die generellen Fehler auftreten.

Method clearfolder:

Section: pn
Method: clearfolder

Beschreibung:
Löscht den Inhalt eines Ordners

Authentifizierung:
Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:
apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
folderid: die ID des Ordners
type: der Typ des Ordners ('in' bzw. 'out')

trash: '1' Wenn es sich um den Papierkorb handelt, '0' wenn nicht.

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

307	API_PN_CLEAR_FOLDER_ERROR
-----	---------------------------

Zusätzlich können die generellen Fehler auftreten.

Method movepn:

Section: pn
Method: movepn

Beschreibung:

Verschiebt eine PN in einen anderen Ordner

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
pnid: die ID der PN
folderid: die ID des Ordners
type: der Typ des Ordners ('in' bzw. 'out')

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

308	API_PN_MOVING_ERROR
-----	---------------------

Zusätzlich können die generellen Fehler auftreten.

Section friendslist:

Method getfriendslist:

Section: friendslist
Method: getfriendslist

Beschreibung:

Lest die Friendslist des angegebenen Users aus

Authentifizierung:

Für diese Methode ist eine keine Authentifizierung notwendig (nicht userbasierter Aufruf)

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
username: der Username des Users

Beispielantwort:

```
<friendslist>
  <group>
    <groupname>Friends</groupname>
    <groupid>396262</groupid>
    <friend>
      <friendid>277</friendid>
      <friendname>murschi</friendname>
    </friend>
  </group>
  <group>
    <groupname>HTLer</groupname>
    <groupid>396262</groupid>
    <friend>
      <friendid>2284</friendid>
      <friendname>schurl666</friendname>
    </friend>
  </group>
  <friendcount>2</friendcount>
</friendslist>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section userfoto:

Method getalbums:

Section: userfoto
Method: getalbums

Beschreibung:

Lest die Useralben des angegebenen Users aus

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (nicht userbasierter Aufruf)

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
username: der Username des Users

Beispielantwort:

```
<userfoto>
  <albumCount>1</albumCount>
  <adminrights>1</adminrights>
  <album>
    <id>104</id>
    <name>name</name>
    <commentcount>0</commentcount>
    <description>beschreibung</description>
    <viewcount>0</viewcount>
    <fotocount>4</fotocount>
    <permission>0</permission>
    <ownerid>2069</ownerid>
  </album>
</userfoto>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method createalbum:

Section: userfoto
Method: createalbum

Beschreibung:

Legt ein neues Useralbum an für den eingeloggten User an (authtoken).

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)

name: der Name des Albums
desc: die Beschreibung des Albums
maxmedia: wieviele Bilder sollen pro Albenseite sichtbar sein¹
visibility: Sichtbarkeit des Ordners²
password: Wenn das Album Passwortgeschützt sein soll, das Passwort des Albums (kein MD5 - Hash)
comments: 1 für Kommentare im Album erlauben, 0 für nicht erlauben

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

405	API_FOTO_SAVE_ALBUM_ERROR
-----	---------------------------

Hier können nur die generellen Fehler auftreten.

¹ maxmedia kann die werte 4, 8, 12, 16, 20, 24, 28 oder 32 annehmen.

² visibility:

0: für Alle zugänglich

1: für Friends zugänglich

2: nur für mich sichtbar und zugänglich

3: Passwortgeschützt

Method upload:

Section: userfoto

Method: upload

Beschreibung:

Fügt ein neues Bild in das angegebene Album ein.

Wenn Sie den POST – Request manuell erstellen, anstatt eine Dienstbibliothek (z.B. cURL) verwenden sollte dieser in etwa so aussehen (an allen Zeilenenden muss ein \r\n stehen):

```
POST / HTTP/1.0
Host: rest.api.weblife1.com
Content-type: multipart/form-data; boundary=AaB03x
Content-Length: 34567

--AaB03x
content-disposition: form-data; name="section"

userfoto
--AaB03x
content-disposition: form-data; name="method"

upload
--AaB03x
content-disposition: form-data; name="apikey"

... Ihr apikey ...
--AaB03x
content-disposition: form-data; name="authsecret"

... Ihr authsecret ...
--AaB03x
content-disposition: form-data; name="authtoken"

... Ihr authtoken ...
--AaB03x
content-disposition: form-data; name="folderid"

... die folderid (z.B 13) ...
--AaB03x

content-disposition: form-data; name="file";
filename="test.jpg"
Content-Type: image/jpeg

... die originalen Bilddaten (JFIF) ...
--AaB03x--
```

Vergessen Sie das \r\n nach Ihren Bilddaten nicht.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

authtoken: der authtoken (von user / login)
folderid: die ID des Ordners, in den das Foto geladen werden soll
file: die hochzuladende Datei

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

```
<userfoto>
  <success>true</success>
</userfoto>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method readalbum

Section: userfoto
Method: readalbum

Beschreibung:

Lest den Inhalt eines Useralbums aus.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (nicht userbasierter Aufruf), lediglich optional für ein Album welches nur für Freunde zugänglich ist muss eine Authentifizierung vorliegen.

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken[optional]: Wenn eine Friendsüberprüfung stattfinden muss wird der Token zur Authentifizierung benötigt.
folderid: die ID des Ordners der auszulesen ist
password[optional]: das Passwort des Ordners, in dem sich das Bild befindet
offset [optional]: der Offset (default 0)

Beispielantwort:

```
<userfoto>
  <foto>
    <id>1070</id>
    <thumburl>
      http://idev.fs_user3.szene1.co.at/fotoalbum/2/2069/104/tn_1070_
      69cc1.jpg
    </thumburl>
    <viewcount>0</viewcount>
    <commentcount>0</commentcount>
  </foto>
  <foto>
    <id>1071</id>
    <thumburl>
      http://idev.fs_user3.szene1.co.at/fotoalbum/2/2069/104/tn_1071_
      c65d8.jpg
    </thumburl>
    <viewcount>0</viewcount>
    <commentcount>0</commentcount>
  </foto>
```

```

    <fotocount>2</fotocount>
    <fotosperpage>16</fotosperpage>
    <adminrights>1</adminrights>
  </userfoto>

```

Fehlercodes:

400	API_FOTO_NO_ACCESS
402	API_FOTO_ALBUM_NOT_FOUND
404	API_FOTO_INVALID_PASS_ERROR

Zusätzlich können die generellen Fehler auftreten.

Method getfoto

Section: userfoto

Method: getfoto

Beschreibung:

Lest die Bildinformationen eines Bildes aus.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (nicht userbasierter Aufruf)

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

username: der Username des Users

fotoid: die ID des Fotos das auszulesen ist

password[optional]: das Passwort des Ordners, in dem sich das Bild befindet

offset [optional]: der Offset (default 0)

Beispielantwort:

```

<userfoto>
  <adminrights>0</adminrights>
  <id>41218622</id>
  <bigurl>http://fotos.szenel.at/87574/4174325.jpg</bigurl>
  <uploadtime>1216021100</uploadtime>
  <viewcount>131</viewcount>
  <commentcount>1</commentcount>
  <previd>0</previd>
  <prevthumburl></prevthumburl>
  <nextid>44783101</nextid>
  <nextthumburl>http://fotos.szenel.at/130035/tn_4462290.jpg</nextthu
mburl>
  <height>0</height>
  <width>0</width>
  <folderid>1569783</folderid>
  <page>1</page>
</userfoto>

```

Fehlercodes:

400	API_FOTO_NO_ACCESS
401	API_FOTO_NOT_FOUND
404	API_FOTO_INVALID_PASS_ERROR

Zusätzlich können die generellen Fehler auftreten.

Method deletefoto

Section: userfoto
Method: deletefoto

Beschreibung:
Löscht ein Foto aus einem Useralbum

Authentifizierung:
Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:
apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
fotoid: die ID des zu löschenden Fotos
folderid: die ID des Orders der auszulesen ist

Beispielantwort:
Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

403	API FOTO DELETING ERROR
401	API FOTO NOT FOUND

Zusätzlich können die generellen Fehler auftreten.

Method deletealbum

Section: userfoto
Method: deletealbum

Beschreibung:
Löscht ein Useralbum (samt Inhalt!)

Authentifizierung:
Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf)

Parameter:
apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
folderid: die ID des Orders der auszulesen ist

Beispielantwort:
Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

402	API_FOTO_ALBUM_NOT_FOUND
406	API_FOTO_DELETE_ALBUM_ERROR

Zusätzlich können die generellen Fehler auftreten.

Section event:

Events in den Ländern, in denen Weblife1 vertreten ist (Österreich, Slowakei, Schweiz), sind deren Bundesländern zugeordnet:

Österreich:

provinceid	Bundesland
1	Burgenland
2	Kärnten
3	Niederösterreich
4	Oberösterreich
5	Salzburg
6	Steiermark
7	Tirol
8	Vorarlberg
9	Wien

Slowakei:

provinceid	Bundesland
15413	Prešovský kraj
15420	Košický kraj
15421	Žilinský kraj
15422	Banskobystrický kraj
15423	Trenciansky kraj
15424	Nitriansky kraj
15425	Trnavský kraj
15426	Bratislavský kraj

Schweiz (hier sind mehrere Kantons zusammengefasst worden):

provinceid	Bundesland
30	Zürich
87	Bern
21	Graubünden
32, 64, 34, 68	Nordwest – Schweiz
85, 15447, 15367, 20, 40	Ost – Schweiz
15448, 15372, 15432, 15450, 15340	Suisse – Romande
15341	Tessin
15342	Wallis
15449, 36, 15452, 15451, 73, 15453, 5424	Zentral – Schweiz

Channels sind für den Eventbereich eine Zusammenfassung mehrerer Kategorien (fett geschrieben steht für einen Host):

channelid	Channel
4	Szene1.at
7	Autotreffen
18	Szene3

19	EME
20	Bikercom
17	Demonic Nights
11	PokerFieber.com
15	Szene1.ch
13	Szene1.sk
21	Einfachleben.com

Method geteventlist:

Section: event

Method: geteventlist

Beschreibung:

Holt eine Liste von Events aus dem Eventkalender. Diese Events sind immer aufsteigend (ASC) geordnet (vom Aktuellsten zum weniger Aktuellen).

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

page [optional]: die Seite, die angezeigt werden soll (für Blätterfunktion) (default 1)

limit [optional]: das Limit der Events, die pro Seite erscheinen (default 30, max. 100)

relevance [optional]: sollen relevante Events angezeigt werden

channelid [optional]: die ChannelId der Events

locationid [optional]: die LocationId der Events

locationname [optional]: der LocationName der Events

categoryid [optional]: die CategoryId der Events (für mehrere IDs diese mit ',' [Komma] trennen)

country [optional]: Der ISO3166-1 alpha-2 countrycode des Landes

provinceid [optional]: die ProvinceId der Events (für mehrere IDs diese mit ',' [Komma] trennen)

eventname [optional]: der Eventname

from [optional]: ab wann werden die Events angezeigt

to [optional]: bis wann werden die Events angezeigt

radius [optional]: wie groß soll der Suchradius sein? (longitude & latitude müssen gegeben sein, oder der place)

longitude [optional]: Longitude

latitude [optional]: Latitude

place [optional]: Adresse eines Ortes (Name, Postleitzahl, Straße, Hausnummer)

order [optional]: Das Feld nach dem sortiert werden soll

orderdirection [optional]: Die Richtung ('ASC' = Aufsteigend, 'DESC' = Absteigend - auf den order Parameter bezogen)

loadflyers [optional]: Gibt an ob die Flyers mitgeladen werden, („true“, „false“). Wenn die Flyer mitgeladen werden ist die Abfrageperformance schlechter, da auch Abfragen in das Filesystem durchgeführt werden müssen.

Events können nach unterschiedlichen Feldern geordnet werden:

Order	Beschreibung
City	Nach dem Ort
locationname	Nach dem Locationnamen

provinceid	Nach dem Bundesland
country	Nach dem Land
latitude	
longitude	
Eventid	Nach der Event ID
categoryid	Nach der Kategorie ID
eventname	Nach dem Eventnamen
priority	Nach der Priorität
Date	Nach dem Datum
distance	Nach der Entfernung

Beispielantwort:

```

<event>
  <events>
    <id>132766</id>
    <name>Tanz in der Halle</name>
    <city>Sierning</city>
    <location>Bachnergut</location>
    <latitude>47.676067</latitude>
    <longitude>13.098753</longitude>
    <category>3</category>
    <date>2009-06-10</date>
  </events>
  <events>
    <id>132767</id>
    <name>Tanz in der Halle</name>
    <city>Sierning</city>
    <location>Bachnergut</location>
    <latitude>47.676067</latitude>
    <longitude>13.098753</longitude>
    <category>3</category>
    <date>2009-06-13</date>
  </events>
  <eventcount>2</eventcount>
</event>

```

Beispielantwort mit Flyer:

```

<event>
  <events>
    <id>132766</id>
    <name>Tanz in der Halle</name>
    <city>Sierning</city>
    <location>Bachnergut</location>
    <latitude>47.676067</latitude>
    <longitude>13.098753</longitude>
    <category>3</category>
    <date>2009-06-10</date>
    <flyer>
<thumburl>http://files.szenel.at/production/Event/244/tn_Event4b87d93d1c9f1
.jpg</thumburl>
<url>http://files.szenel.at/production/Event/244/Event4b87d93d1c9f1.jpg</ur
l>
    </flyer>
    <flyer>
<thumburl>http://files.szenel.at/production/Event/244/tn_Event4b87d94d1c321
.jpg</thumburl>
<url>http://files.szenel.at/production/Event/244/
Event4b87d94d1c321.jpg</url>

```

```
</flyer>
</events>
<events>
  <id>132767</id>
  <name>Tanz in der Halle</name>
  <city>Sierning</city>
  <location>Bachnergut</location>
  <latitude>47.676067</latitude>
  <longitude>13.098753</longitude>
  <category>3</category>
  <date>2009-06-13</date>
</events>
<eventcount>2</eventcount>
</event>
```

Fehlercodes:

Zusätzlich können die generellen Fehler auftreten.

Method getevent:

Section: event
Method: getevent

Beschreibung:

Gibt das Event mit der gegebenen ID aus.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
eventid: die ID des Events

Beispielantwort:

```
<event>
  <id>132766</id>
  <name>Tanz in der Halle</name>
  <desc>More infos coming soon...</desc>
  <date>Mo, 10. Jun. 2013</date>
  <datetime>2013-06-10 20:00:00</date>
  <starttime>20:00:00</starttime>
  <categoryid>3</categoryid>
  <location>Bachnergut</location>
  <locationid>85298</locationid>
  <latitude>48.412388</latitude>
  <longitude>15.597110</ longitude >
  <flyerurl> http://files.szenel.at/production/Event/244/
Event4b87d93d1c9f1.jpg </flyerurl>
  <flyerthumburl>
http://files.szenel.at/production/Event/244/tn_Event4b87d93d1c9f1.jpg
</flyerthumburl>
  <music>Minze</music>
  <fotocount>0</fotocount>
```

```
<city>Sierning</city>
<district>0</district>
</event>
```

Fehlercodes:

500	API_EVENT_NOT_FOUND
501	API_EVENT_OFFLINE

Zusätzlich können die generellen Fehler auftreten.

Achtung:

Die Felder date und starttime sind als deprecated markiert und sollten daher nicht mehr verwendet werden.

Method getmeetpointuserlist:

Section: event
Method: getmeetpointuserlist

Beschreibung:

Liefert die User zurück die beim angegebenen Event im Meetpoint eingetragen sind.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig.

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
eventid: ID des Events

Beispielantwort:

```
<event>
  <user>
    <userid>12034</name>
    <username>snog</id>
  </user>
  <user>
    <userid>1</name>
    <username>energy</id>
  </user>
  <usercount>2</usercount>
</event>
```

Fehlercodes:

500	API_EVENT_NOT_FOUND
-----	---------------------

Method addtomeetpoint:

Section: event
Method: addtomeetpoint

Beschreibung:

Fügt den aktuell eingeloggtten User zum Meetpoint hinzu.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
eventid: ID des Events

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

500	API_EVENT_NOT_FOUND
-----	---------------------

Method removefrommeetpoint:

Section: event
Method: removefrommeetpoint

Beschreibung:

Löscht den aktuell eingeloggten User vom Meetpoint.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
authtoken: der authtoken (von user / login)
eventid: ID des Events

Beispielantwort:

Diese Methode gibt bei Erfolg eine Erfolgsmeldung zurück. Bei einem Fehler wird eine Fehlermeldung ausgegeben.

Fehlercodes:

500	API_EVENT_NOT_FOUND
-----	---------------------

Method geteventcategories:

Section: event
Method: geteventcategories

Beschreibung:

Gibt die ID der Kategorie mit dem dazugehörigen Namen aus.

Authentifizierung:

Für diese Methode ist eine Authentifizierung notwendig (userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret

language: 'de_AT' für Österreich, 'de_CH' für Schweiz und 'sk_SK' für Slowakei

Beispielantwort:

```
<event>
  <category>
    <name>Disco</name>
    <id>1</id>
  </category>
  <category>
    <name>Electronic</name>
    <id>2</id>
  </category>
</event>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section eventfoto:

Method geteventfotolist:

Section: eventfoto
Method: geteventfotolist

Beschreibung:

Holt eine Liste von Events mit Fotos aus dem Eventkalender. Diese Events sind immer absteigend (DESC) sortiert (vom weniger Aktuellen zum Aktuellsten).

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
page [optional]: die Seite, die angezeigt werden soll (für Blätterfunktion) (default 1)
limit [optional]: das Limit der Events, die pro Seite erscheinen (default 30, max. 100)
relevance [optional]: sollen relevante Events angezeigt werden
channelid [optional]: die ChannelId der Events
locationid [optional]: die LocationId der Events
locationname [optional]: der LocationName der Events
categoryid [optional]: die CategoryId der Events (für mehrere IDs diese mit ',' [Komma] trennen)
country [optional]: Der ISO3166-1 alpha-2 countrycode des Landes
provinceid [optional]: die ProvinceId der Events (für mehrere IDs diese mit ',' [Komma] trennen)
eventname [optional]: der Eventname
from [optional]: ab wann werden die Events angezeigt
to [optional]: bis wann werden die Events angezeigt
radius [optional]: wie groß soll der Suchradius sein? (longitude & latitude müssen gegeben sein, oder der place)
longitude [optional]: Longitude
latitude [optional]: Latitude
place [optional]: Adresse eines Ortes (Name, Postleitzahl, Straße, Hausnummer)
order [optional]: Das Feld nach dem sortiert werden soll
orderdirection [optional]: Die Richtung ('ASC' = Aufsteigend, 'DESC' = Absteigend - auf den order Parameter bezogen)

Events (mit Fotos) können nach unterschiedlichen Feldern geordnet werden:

order	Beschreibung
city	Nach dem Ort
locationname	Nach dem Locationnamen
provinceid	Nach dem Bundesland
country	Nach dem Land
latitude	
longitude	
eventid	Nach der Event ID
categoryid	Nach der Kategorie ID
eventname	Nach dem Eventnamen
priority	Nach der Priorität
fotocount	Nach der Anzahl der Fotos

viewcount	Nach der Anzahl der Views
commentcount	Nach der Anzahl der Comments
date	Nach dem Datum
distance	Nach der Entfernung

Beispielantwort:

```

<eventfoto>
  <events>
    <id>132766</id>
    <name>Tanz in der Halle</name>
    <city>Sierning</city>
    <location>Bachnergut</location>
    <fotocount>1900</fotocount>
    <viewcount>50000</viewcount>
    <priority>1</priority>
    <date>2009-06-10</date>
  </events>
  <events>
    <id>132767</id>
    <name>Tanz in der Halle</name>
    <city>Sierning</city>
    <location>Bachnergut</location>
    <fotocount>1900</fotocount>
    <viewcount>50000</viewcount>
    <priority>1</priority>
    <date>2009-06-13</date>
  </events>
  <eventcount>2</eventcount>
</eventfoto>

```

Fehlercodes:

100	API_APP_INVALID_PARAM
101	API_APP_MISSING_PARAM

Zusätzlich können die generellen Fehler auftreten.

Method geteventfotos:

Section: eventfoto
Method: geteventfotos

Beschreibung:

Holt eine Fotoliste vom Event mit der gegebenen ID.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authsecret
eventid: Die ID des Events, für das Fotos ausgelesen werden sollen.
page [optional]: die Seite, die angezeigt werden soll (für Blätterfunktion)
(default 1)

limit [optional]: das Limit der Fotos, die pro Seite erscheinen (default 15, max. 30)

Beispielantwort:

```
<eventfoto>
  <foto>
    <id>2317910</id>
    <thumburl>
      http://idev.fs_fotos.szenel.co.at/78957/tn_2317910.jpg
    </thumburl>
    <bigurl>
      http://idev.fs_fotos.szenel.co.at/78957/2317910.jpg
    </bigurl>
    <viewcount>204</viewcount>
    <commentcount>5</commentcount>
  </foto>
  <foto>
    <id>2317911</id>
    <thumburl>
      http://idev.fs_fotos.szenel.co.at/78957/tn_2317911.jpg
    </thumburl>
    <bigurl>
      http://idev.fs_fotos.szenel.co.at/78957/2317911.jpg
    </bigurl>
    <viewcount>245</viewcount>
    <commentcount>8</commentcount>
  </foto>
  <foto>
    <id>2317912</id>
    <thumburl>
      http://idev.fs_fotos.szenel.co.at/78957/tn_2317912.jpg
    </thumburl>
    <bigurl>
      http://idev.fs_fotos.szenel.co.at/78957/2317912.jpg
    </bigurl>
    <viewcount>70</viewcount>
    <commentcount>0</commentcount>
  </foto>
  <fotocount>3</fotocount>
</eventfoto>
```

Fehlercodes:

100	API_APP_INVALID_PARAM
101	API_APP_MISSING_PARAM
601	API_EVENTFOTO_NO_FOTOS

Zusätzlich können die generellen Fehler auftreten.

Method geteventfoto:

Section: eventfoto
Method: geteventfoto

Beschreibung:

Holt Bildinformationen vom Eventfoto mit der gegebenen ID.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key

authsecret: das authsecret

fotoid: Die ID des Fotos, für das Informationen ausgelesen werden sollen.

Achtung: Die Felder width und height sind als deprecated markiert, es werden hier keine Daten angeliefert. Diese Felder werden in einer späteren Version entfernt.

Beispielantwort:

```
<eventfoto>
  <id>2317911</id>
  <bigurl>
    http://idev.fs_fotos.szenel.co.at/78957/2317911.jpg
  </bigurl>
  <uploadtime></uploadtime>
  <viewcount>245</viewcount>
  <commentcount>8</commentcount>
  <previd>2317910</previd>
  <prevthumburl>
    http://fotos.szenel.at/142206/tn_4692792.jpg
  </prevthumburl>
  <nextid>2317912</nextid>
  <nextthumburl>
    http://fotos.szenel.at/142206/tn_4692794.jpg
  </nextthumburl>
  <height></height>
  <width></width>
  <eventid>78957</eventid>
</eventfoto>
```

Fehlercodes:

600	API_EVENTFOTO_NOT_FOUND
-----	-------------------------

Zusätzlich können die generellen Fehler auftreten.

Section usersearch:

Method search:

Section: usersearch
Method: search

Beschreibung:

Führt eine Usersuche in der Weblife1 Community durch. (Liefert nur Benutzer die Suchfreigabe auf „öffentlich“ gestellt haben).

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authSecret
forename: Vorname des Users, kann leer sein.
surname: Nachname des Users, kann auch leer sein.

Beispielantwort:

```
<usersearch>
  <user>
    <surname>Mustermann</surname>
    <forename>Max</forename>
    <origin>Perg / Oberösterreich / Österreich</origin>
    <gender>male</gender>
    <userpic>http://user.szenel.at/fotos/thumbnail/1234.jpg</userpic>
    <userpageurl>http://www.szenel.at/user/maxi_mustermann</userpageurl>
    <username>maxi_mustermann</username>
  </user>
  <user>
    <surname>Mustermann</surname>
    <forename>Maria</forename>
    <origin>Perg / Oberösterreich / Österreich</origin>
    <gender>female</gender>
    <userpic>http://user.szenel.at/fotos/thumbnail/382940.jpg</userpic>
    <userpageurl>http://www.szenel.at/user/maria_mustermann</userpageurl>
    <username>maria_mustermann</username>
  </user>
  <usercount>2</usercount>
</usersearch>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section userdata:

Method getuser:

Section: userdata
Method: getuser

Beschreibung:

Liefert die Detaildaten eines bestimmten Users zurück. Hierbei ist die Berechtigung Userdata-Area für die Applikation notwendig.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authSecret
userid: UserId des Benutzers

Beispielantwort:

```
<userdata>
  <username>maxi_mustermann</username>
  <gender>male</gender>
  <zipcode>4030</zipcode>
  <city>Linz</city>
  <street>Franzosenhausweg 49</street>
  <distrinct>Linz-Stadt</distrinct>
  <province>Oberösterreich</province>
  <age>31</age>
  <character>
    <value>Sensibel</value>
  </character>
  <character>
    <value>Selbstbewusst</value>
  </character>
  <character>
    <value>Langschläfer</value>
  </character>
  <hobby>
    <value>Sommersport</value>
  </hobby>
  <hobby>
    <value>Wintersport</value>
  </hobby>
  <drink>
    <value>Non-Alk</value>
  </drink>
  <drink>
    <value>Fruchtsäfte</value>
  </drink>
  <music>
    <value>House</value>
  </music>
  <music>
    <value>Reggae</value>
  </music>
</userdata>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section group:

Method getmembers:

Section: group
Method: getmembers

Beschreibung:

Liefert die Mitglieder der angegebenen Gruppe zurück.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

ACHTUNG: Werden die Offset und Limit Parameter nicht angegeben so werden maximal die ersten 100 Mitglieder zurückgegeben.

Parameter:

apikey: Ihr API Key
authsecret: das authSecret
groupurl: Der eindeutige Gruppenname (nicht Titel) so wie er in der URL übergeben wird.
offset [optional]: Start in der Liste
limit [optional]: Anzahl der Datensätze für diesen Request

Beispielantwort:

```
<group>
  <count>2</count>
  <user>
    <userid>267760969</userid>
    <username>maxi_mustermann</username>
  </user>
  <user>
    <userid>23432369</userid>
    <username>maria_mustermann</username>
  </user>
</group>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method getrandommembers:

Section: group
Method: getrandommembers

Beschreibung:

Liefert eine gewisse Anzahl an zufällig ausgewählten Mitgliedern aus der angegebenen Gruppe. Sind weniger als die angegebenen Anzahl der Mitglieder in der Gruppe so werden alle Mitglieder ausgegeben. Als Standard werden maximal 30 Mitglieder zurückgeliefert.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key

authsecret: das authSecret
groupurl: Der eindeutige Gruppenname (nicht Titel) so wie er in der URL übergeben wird.
Amount [optional]: Anzahl der Mitglieder die zurückgeliefert werden soll

Beispielantwort:

```
<group>
  <count>2</count>
  <user>
    <userid>267760969</userid>
    <username>maxi_mustermann</username>
  </user>
  <user>
    <userid>23432369</userid>
    <username>maria_mustermann</username>
  </user>
</group>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Method getmembercount:

Section: group
Method: getmembercount

Beschreibung:

Liefert die Anzahl der Mitglieder in der Gruppe

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authSecret
groupurl: Der eindeutige Gruppenname (nicht Titel) so wie er in der URL übergeben wird.

Beispielantwort:

```
<group>
  <membercount>21</membercount>
</group>
```

Fehlercodes:

Hier können nur die generellen Fehler auftreten.

Section servicellogin:

Method authenticate:

Section: servicellogin
Method: authenticate

Beschreibung:

Stellt ein Handshakeverfahren zur Verfügung bei dem keine Zugangsdaten (Benutzername und Passwort) benötigt werden. Dazu muss man sich einen passenden Handshakekey vom Szene1 Server anfordern. Hierbei ist die Berechtigung Servicellogin-Area für die Applikation notwendig.

Für das anfordern eines Servicellogin Hashkeys gibt es eine eigene Dokumentation. Diese ist über das Serviceteam von Szene1 erhältlich.

Authentifizierung:

Für diese Methode ist keine Authentifizierung notwendig (kein userbasierter Aufruf).

Parameter:

apikey: Ihr API Key
authsecret: das authSecret
handshakekey: Schlüssel der vom Szenel Server angefordert wurde

Beispielantwort:

```
<servicellogin>  
  <authtoken>94d6e59a058519b1c0bed5b3d307b760</authtoken>  
  <userid>12034</userid>  
  <username>snog</username>  
</servicellogin>
```

Fehlercodes:

700	API_SERVICELOGIN_INVALID_KEY
-----	------------------------------

Zusätzlich können die generellen Fehler auftreten.

Error Code	PHP Konstante	Beschreibung	Section / Methode
1	API_GEN_UNKNOWN_ERROR	Ein unbekannter Fehler tritt auf	Alle
2	API_GEN_SERVICE_UNAVAILABLE	Das Service ist zur zeit nicht Verfügbar	Alle
3	API_GEN_UNKOWN_CALL	Section / Methode nicht bekannt	Alle
4	API_GEN_TOO_MANY_CALLS	Überschreitung der max. möglichen Aufrufe	Alle
5	API_GEN_AUTH_TOKEN_ERROR	Der Aufruf erfordert einen authtoken, es wurde aber keiner gegeben, oder der authtoken ist abgelaufen	Alle
6	API_GEN_TOO_MANY_REQUESTS	Überschreitung der max. möglichen Aufrufe pro Minute.	
7	API_GEN_KEY_LOCKED	Der API Key wurde von der Administration gesperrt	Alle
100	API_APP_INVALID_PARAM	Falscher Parameter bzw. Parameterwert	Alle
101	API_APP_MISSING_PARAM	Fehlender Parameter	Alle
102	API_APP_INVALID_AUTH_SECRET	Falsches Secret	Alle
103	API_APP_INVALID_API_KEY	Falscher Applikationskey	Alle
104	API_APP_INVALID_AUTH_TOKEN	Falscher Authentifizierungs Token	Userbasierende Aufrufe
105	API_APP_NO_ACCESS	Keine Berechtigung für diesen API Bereich	Alle
200	API_USER_INVALID_USER_OR_PASS	Falscher Username bzw. falsches Passwort	user
201	API_USER_INVALID_USER	Falscher Username	user
202	API_USER_INVALID_PASS	Falsches Passwort	user
203	API_USER_USER_LOCKED	User ist gesperrt	user
204	API_USER_USER_DELETED	User wurde gelöscht	user

300	API_PN_SENDING_ERROR	PN konnte nicht zugestellt werden	pn
301	API_PN_DELETING_ERROR	PN konnte nicht gelöscht werden	pn
302	API_PN_READING_ERROR	Es trat ein Fehler beim Lesen der PN auf	pn
303	API_PN_READ_FOLDER_ERROR	Es trat ein Fehler beim Auslesen des Ordners auf	pn
304	API_PN_CREATE_FOLDER_ERROR	Es trat ein Fehler beim Erstellen des Ordners auf	pn
305	API_PN_RENAME_FOLDER_ERROR	Es trat ein Fehler beim Umbenennen des Ordners auf	pn
306	API_PN_DELETE_FOLDER_ERROR	Es trat ein Fehler beim Löschen des Ordners auf	pn
307	API_PN_CLEAR_FOLDER_ERROR	Es trat ein Fehler beim Leeren des Ordners auf	pn
308	API_PN_MOVING_ERROR	Es trat ein Fehler beim Verschieben der PN auf	pn
400	API_FOTO_NO_ACCESS	Kein Zugriff auf das angegebene Foto / Album	userfoto
401	API_FOTO_NOT_FOUND	Das Foto wurde nicht gefunden	userfoto
402	API_FOTO_ALBUM_NOT_FOUND	Das Album wurde nicht gefunden	userfoto
403	API_FOTO_DELETING_ERROR	Es trat ein Fehler beim Löschen des Fotos auf	userfoto
404	API_FOTO_INVALID_PASS_ERROR	Das angegebene Passwort ist falsch	userfoto
405	API_FOTO_SAVE_ALBUM_ERROR	Es trat ein Fehler beim Speichern des Albums auf	userfoto
406	API_FOTO_DELETE_ALBUM_ERROR	Es trat ein Fehler beim Löschen des Albums auf	userfoto
500	API_EVENT_NOT_FOUND	Das Event mit der gegebenen ID wurde nicht gefunden	event
501	API_EVENT_OFFLINE	Das Event mit der gegebenen ID ist offline	event
600	API_EVENTFOTO_NOT_FOUND	Das Event mit der	eventfoto

		gegebenen ID enthält keine Fotos	
601	API_EVENTFOTO_NO_FOTOS	Das Eventfoto mit der gegebenen ID wurde nicht gefunden	eventfoto

Beispielantwort:

```
<error>
  <errorCode>102</errorCode>
  <errorMessage>Authsecret did not match!</errorMessage>
</error>
```