CloudBank API

Version: 0.1
Date: 16 October 2009
Author: Marcus Winter

Interactive Technologies Research Group, University of Brighton
http://itrg.brighton.ac.uk

Introduction

CloudBank is a mobile and web-based crowd-sourced information system for advanced language learners. The system allows learners to collect, annotate and tag interesting or intriguing language- and culture-related content found in everyday life, including text, images and other media, and to upload these content items to a repository. From the repository, the content can be syndicated in a number of ways.

The CloudBank repository exposes its data via an Application Programming Interface (API). This document is the official reference for that functionality. It is written for developers who like to write services or applications that integrate with the repository.
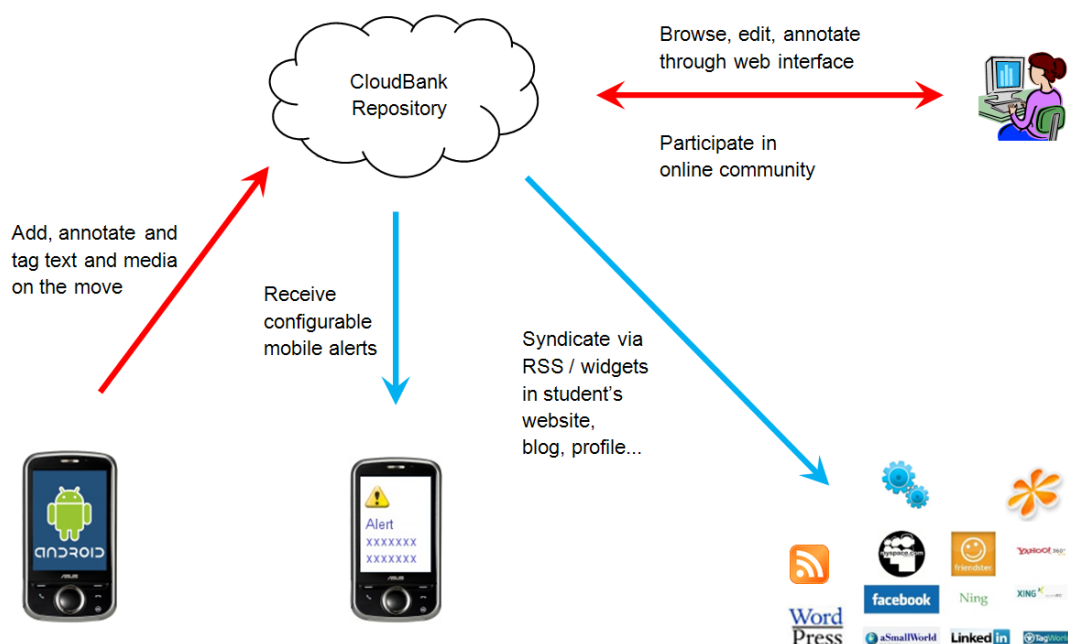

Fig. 1: system overview

Getting started

In order to avoid unauthorised misuse of the CloudBank API, developers need an API key to query the CloudBank REST server. API keys can be requested by email from the Interactive Technologies Research Group at the University of Brighton.

Basics

The CloudBank API is entirely HTTP-based. Method calls to the CloudBank API require a POST request. Other request formats return an error.

The CloudBank API conforms to the design principles of Representational State Transfer (REST). The API is designed to support multiple formats that can be specified as part of the request. The present API implementation supports XML only.

API methods take both, required standard parameters and optional request-dependent parameters. Parameter values should be converted to UTF-8 and URL encoded.

The endpoint for all CloudBank API calls is
http://itrg.brighton.ac.uk/cloudbank/restserver.php

Authentication

All Cloudbank API methods require user authentication to access or publish data. To authenticate yourself you need two pieces of information, which you receive when you apply for an API key:

- Public API key
- Private API secret

The public API key is appended to every API call to identify the developer. The private API secret is known only to the developer and should never be disclosed.

Authentication involves three parameters required for every API method call:
- Parameter 'key': the public API key
- Parameter 'cid': the call id for this request. The call id is a random string to identify the call and is used as input for the signature. The random string must be at least 16 characters long and should not exceed 64 characters.
- Parameter 'sig': the signature for this request. The signature string is calculated from the call id (i.e. the 'cid' parameter) and the API secret.

Algorithm for generating a signature string:

```
String cid = CreateRandomString(32)
String val = Concatenate(cid, api_secret)
Byte[] hash = MD5(val)
String sig = BytesToHexString(hash)
```

Example implementations for Java and PHP are available on request from the CloudBank developer team.

Common parameters for all methods

name	ver
description	API version (currently 1.0)
required	yes
example	ver=1.0

name	key
description	developer API key
required	yes
example	key=12345678

name	uid
description	user-ID, username or telephone number of current user
required	yes
example	uid=07789990314

name	cid
description	the call id for this request. The call id is a random string to identify the call and is used as input for the signature. The random string must be at least 16 characters long and should not exceed 64 characters.
required	yes
example	cid=A87gkj2G5hg7Hr5d

name	sig
description	signature for this request (* see Authentication)
required	yes
example	sig=1234567890abcdef1234567890abcdef1234567890abcd...

name	fmt
description	requested format (currently only XML is supported)
required	yes
example	fmt=xml

Method-specific parameters

getItemList

Use this method to retrieve a list of items currently in the Cloudbank repository. The item list is sorted in descending order according to the last modified date.

Parameters:

name	q
description	method name
required	yes
example	q=getItemList

name	tag
description	filter item list by tag. can be either a tag or tag_id. numeric values are interpreted as tag_id
required	no
example	tag=5 -or- tag=football

name	search
description	search term for full-text search over fields 'items.item' and 'items.description'. If tag parameter is set, then search applies only to the specific subset .
required	no
example	search=hello

name	limit
description	limit number of returned items
required	no
example	limit=10

Returns (example):

```xml
<?xml version="1.0" encoding="UTF-8"?>
<dataset>
<info><name>getItemList</name></info>
<data>
<row>
<_id>1</_id>
<item>to nutmeg</item>
</row>
<row>
<_id>2</_id>
<item>cheers</item>
</row>
</data>
</dataset>
```

getTagList

Use this method to retrieve a list of tags currently in the Cloudbank repository. The item list is sorted in descending order by tag popularity, i.e. how many items are tagged with them.

Parameters:

name	q
description	method name
required	yes
example	q=getTagList

name	limit
description	limit number of returned tags
required	no
example	limit=10

Returns (example):

```xml
<?xml version="1.0" encoding="UTF-8"?>
<dataset>
<info><name>getTagList</name></info>
<data>
<row>
```

```
<_id>18</_id>
<tag>media</tag>
</row>
<row>
<_id>2</_id>
<tag>pub</tag>
</row>
<row>
<_id>5</_id>
<tag>football</tag>
</row>
</data>
</dataset>
```

getItem

Use this method to retrieve a list of tags currently in the Cloudbank repository. The item list is sorted in descending order by tag popularity, i.e. how many items are tagged with them.

Parameters:

name	q
description	method name
required	yes
example	q=getItem

name	item
description	identify the item to return. item or item-id. numeric values are interpreted as _id
required	yes
example	item=5 -or- item=nutmeg

Returns (example):

```
<?xml version="1.0" encoding="UTF-8"?>
<dataset>
<info><name>getItem</name></info>
<data>
<row>
<_id>5</_id>
<item>to nutmeg</item>
```

```
<description>in football: to play the ball through an opponent's
legs</description>
<link>http://en.wikipedia.org/wiki/Nutmeg_(football)</link>
<image>app/media/to_nutmeg.jpg</image>
<audio>app/media/to_nutmeg.3gp</audio>
<taglist>football, nutmeg</taglist>
</row>
</data>
</dataset>
```

putItem

Use this method to insert or update an item in the Cloudbank repository. Returns the
_id of the item stored.

Parameters:

name	q
description	method name
required	yes
example	q=putItem

name	_id
description	_id of the item to insert or update. If(_id < 0) item is inserted, else item is updated.
required	yes
example	_id=-1 (negative value: insert new item)

name	item
description	the item to insert
required	yes
example	item=to%20nutmeg

name	description
description	definition of the new item
required	yes
example	description=in%20football:%20to%20play...

name	link
description	URL to an online resource
required	yes
example	link=http://en.wikipedia.org/wiki/Nutmeg_(football)

name	image
description	an image related to this item
required	no
example	JPEG image Base64 encoded

name	audio
description	an audio recording related to this item
required	no
example	3GPP audio file Base64 encoded

name	taglist
description	list of comma-separated tags related to this item
required	yes
example	taglist=football,nutmeg

Returns (example):

```
<?xml version="1.0" encoding="UTF-8"?>
<dataset>
<info><name>putItem</name></info>
<data>
<row>
<_id>5</_id>
</row>
</data>
</dataset>
```

Return values

Success

The CloudBank REST server currently only supports XML. If the request could be executed, the response is structured as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<dataset>
<info><name>method-name</name></info>
<data>
<row>
<field_name1>value</field_name1>
...
<field_nameN>value</field_nameN>
</row>
...
<row>
<field_name1>value</field_name1>
...
<field_nameN>value</field_nameN>
</row>
</data>
</dataset>
```

Error

If the request for some reason could not be executed, the response is structured as follows::

```xml
<?xml version="1.0" encoding="UTF-8"?>
<error_response>
<error_msg>error-message</error_msg>
<error_response>
```

Contact

Cloudbank is developed by the Interactive Technologies Research Group at the University of Brighton. Find out more at http://itrg.brighton.ac.uk/cloudbank/

To get in touch with the developers please contact Lyn Pemberton, Marcus Winter or Sanaz Fallahkhair {lp22,mw159,sf61}@brighton.ac.uk or phone +44 1273 642491

University of Brighton
Interactive Technologies Research Group
School of Computing, Mathematical and Information Sciences
Watts Building, Lewes Road
Brighton, BN2 4GJ
United Kingdom