Adms

Adms

> Assist Of Adx


```
1. http http://restful.adleida.com:8008/
2.
3. HTTP/1.0 200 OK
4. Content-Length: 41
5. Content-Type: application/json
6. Date: Tue, 09 Jun 2015 09:09:44 GMT
7. Server: Werkzeug/0.10.4 Python/2.7.6
8.
9. "Welcome to adleida restful API service!"
```

特性

- 成熟的数据管理系统 妥善保存 dsp 信息
- 使用高性能 gridfs 完整存储物料
- 物料经由审核流程确保安全合法
- cdn 大范围分发 大大降低等待时延
- 支持高并发 HTTP 请求

客户端准备工作

```
1. sudo apt install python
2. wget https://bootstrap.pypa.io/get-pip.py
3. sudo python get-pip.py
4. sudo pip install httpie
```

使用指南

Dsp 相关操作

1. 注册 Dsp 基本信息

1. 数据准备

> 注册信息采用 json 数据格式，示例模型如下所示，`name` 为 Dsp 方的名称，`burl` 为 Dsp 方竞价的服务器地址

```
{
    "name": "dspname",
    "burl": "http://example.dsp.com/path/to/bid/"
}
```

2. 调用 `RESTful-API` 进行注册（假设现在你已经完成一份 `json` 格式的 `Dsp` 数据信息，这里暂时命名为 `Dsp-info.json`）

```
http POST http://restful.adleida.com:8008/v1/dsp/ < dsp-info.json
```

3. 确认并保留返回信息

> 您将会接收到如同下列形式的返回信息。其中 `id` 是在我方服务端自行生成的唯一标识，`access_token` 作为以后对该注册信息操作的凭据，请自行妥善保存

```
HTTP/1.0 200 OK
Content-Length: 103
Content-Type: application/json
Date: Tue, 09 Jun 2015 10:44:01 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

{
    "id": "5576c371c44b1a5756805130",
    "access_token": "d19a1398-ccf5-4c47-868c-a4abaf24e011",
    "message": "successfully create your dsp information to adexchange"
}
```

2. 查询 Dsp 信息

1. 通过返还的 id 查询信息

```
http GET http://restful.adleida.com:8008/v1/dsp/<id>
```

> 其中 `<id>` 为注册返回的 id 值 下面的例子直接使用了上述返回的 id 结果

```
http GET http://restful.adleida.com:8008/v1/dsp/5576c371c44b1a5756805130
```

2. 你会接受到如同下列形式的返回信息

```
HTTP/1.0 200 OK
Content-Length: 98
Content-Type: application/json
Date: Tue, 09 Jun 2015 10:52:58 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

{
    "burl": "http://example.dsp.com/path/to/bid/",
    "id": "5576c371c44b1a5756805130",
    "name": "dspname"
}
```

3. 修改已注册的 Dsp 信息

1. 发送请求参数，提交 `PUT` 请求

> 修改只需要将你想改变的字段作为参数交给请求一并发送至服务器即可，这里依然使用上述返回的 `id` 作为演示，请注意修改操作是需要提供 `access_token` 作为凭据的。下方示例修改了 `name` 字段的值

```
http PUT http://restful.adleida.com:8008/v1/dsp/ access_token:d19a1398-ccf5-4c47-868c-a4abaf24e011 id=5576c371c44b1a5756805130 name=newdspname
```

2. 返回提示

> 修改操作成功后返回提示信息如下所示

```
HTTP/1.0 200 OK
Content-Length: 38
Content-Type: application/json
Date: Wed, 10 Jun 2015 02:45:32 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

"successfully update your information"
```

3. 查询确认

> 成功更新字段后查询进行确认

```
http GET http://restful.adleida.com:8008/v1/dsp/5576c371c44b1a5756805130

HTTP/1.0 200 OK
Content-Length: 98
```

```
Content-Type: application/json
Date: Wed, 10 Jun 2015 02:45:42 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

{
    "burl": "http://example.dsp.com/path/to/bid/",
    "id": "5576c371c44b1a5756805130",
    "name": "newdspname"
}
```

4. 删除记录

1. 发起 `DELETE` 请求

> 删除动作只需要提供 `id` 和 `access_token` 即可完成操作，如下例所示

```
http DELETE http://restful.adleida.com:8008/v1/dsp/ access_token:d19a1398-ccf5-4c47-868c-a4abaf24e011 id=5576c371c44b1a5756805130
```

2. 确认返回结果

> 成功删除该条记录后会接收到如下提示信息

```
HTTP/1.0 200 OK
Content-Length: 38
Content-Type: application/json
Date: Wed, 10 Jun 2015 03:32:29 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

"successfully delete your information"
```

3. 查询验证

> 此时便无法查询到该条记录

```
http GET http://restful.adleida.com:8008/v1/dsp/5576c371c44b1a5756805130

HTTP/1.0 404 NOT FOUND
Content-Length: 39
Content-Type: application/json
Date: Wed, 10 Jun 2015 03:32:34 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

{
    "message": "Not Found",
    "status": 404
```

```
}
```

adm 相关操作

1. 注册 adm 基本信息

1. 执行脚本快速批量上传 点击此处下载上传脚本

> 使用该脚本非常方便，你只需要准备一份索引文件 `index.yaml` 和你所需要上传的物料置于文件夹 `images` 下即可，目录结构如下所示

```
.
├── images
│   ├── 1.jpg
│   ├── 2.jpg
│   └── 3.png
├── index.yaml
└── uploader.py
```

> 索引文件 `index.yaml` 内容包含你所提交的 `adm` 相关信息 数据示例模型如下所示

```yaml
-
  data:
    img: "images/1.jpg"
    text: "some app description"
  did: "5576c371c44b1a5756805130"
  type: 1
-
  data:
    img: "images/2.jpg"
    text: "some app description"
  did: "5576c371c44b1a5756805130"
  type: 1
-
  data:
    img: "images/3.png"
    text: "some app description"
  did: "5576c371c44b1a5756805130"
  type: 1
```

> 需要注意 `index.yaml` 格式的缩进是强制性的，其中 `did` 字段需要与上述示例中返回的 `id` 保持一致。
> 接下来便可以直接调用 `uploader.py` 进行物料上传，简单使用如下示例

```
usage: uploader.py [-h] [-o] [-t] [-s] [-m]

adleida simple command line uploader

optional arguments:
  -h, --help        show this help message and exit
  -o, --output      decide to generate result.yaml from response
  -t, --test        test to mock local secne
  -s, --substitute  consider whether script substitute value of field img
```

直接运行 `python` 命令执行该脚本，在检测本地所需上传的物料合法性之后会提示上传，执行前可以指定参数 `-o` 决定上传后将返回结果输出到本地统计目录下的文件 `result.yaml` 中，流程如下所示

```
python uploader.py -o

start checking validation of images ..

finish checking, upload now?    [Y/n]

begin ..

finish uploading
```

`result.yaml` 会将上传信息在服务器生成的 `id` 和取得物料的路径 `img` 显示出来

```
- data:
    img:                                http://restful.adleida.com:8008/v1/media/<media-id>
    text: "some app description"
  did: "5576c371c44b1a5756805130"
  id: <adm-id>
  type: 1
- data:
    img: http://restful.adleida.com:8008/v1/media/<media-id>
    text: "some app description"
  did: "5576c371c44b1a5756805130"
  id: <adm-id>
  type: 1
- data:
    img: http://restful.adleida.com:8008/v1/media/<media-id>
    text: "some app description"
  did: "5576c371c44b1a5756805130"
  id: <adm-id>
```

```
18.    type: 1
```

> 其中 `<adm-id>` 为每条注册的 `adm` 信息独立的身份标识

2. web 界面上传
    - TODO

2. 查询 adm 信息

1. 发送 `GET` 请求通过返还的 `adm-id` 查询信息

```
1. http GET http://restful.adleida.com:8008/v1/adm/<adm-id>
```

2. 确认返回信息

```
1. HTTP/1.0 200 OK
2. Content-Length: 261
3. Content-Type: application/json
4. Date: Wed, 10 Jun 2015 10:14:06 GMT
5. Server: Werkzeug/0.10.4 Python/2.7.6
6.
7. {
8.     "data": {
9.         "img": "http://restful.adleida.com:8008/v1/media/<media-id>",
10.        "text": "some app description"
11.    },
12.    "did": "5576c371c44b1a5756805130",
13.    "id": <adm-id>,
14.    "type": 1
15. }
```

> 需要注意的是，查询出来的返回信息应该与 `result.yaml` 中对应的 `<adm-id>` 结果相同

2. 删除 adm 记录

1. 发送 `DELETE` 请求

> 与删除 `Dsp` 信息操作相同，需要提供 `id`（`<adm-id>`）和 `access_token` 发起删除操作请求

```
1. http DELETE http://restful.adleida.com:8008/v1/adm/ id=<adm-id> access_token:d19a1398-ccf5-4c47-868c-a4abaf24e011
```

2. 确认返回结果

```
HTTP/1.0 200 OK
Content-Length: 38
Content-Type: application/json
Date: Wed, 10 Jun 2015 10:23:48 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

"successfully delete your information"
```

3. 查询验证

```
http GET http://restful.adleida.com:8008/v1/adm/<adm-id>

HTTP/1.0 404 NOT FOUND
Content-Length: 39
Content-Type: application/json
Date: Wed, 10 Jun 2015 10:26:18 GMT
Server: Werkzeug/0.10.4 Python/2.7.6

{
    "message": "Not Found",
    "status": 404
}
```

3. 获取物料展示

- 直接打开浏览器访问返回的 `img` 结果，若物料已经通过服务端审核将予以展示，否则将返回 `404` 错误码

License

MIT