

WW 接口 API 设计 v1

2015.11.5, by: 姜成昱

概述

Where to Watch 接口部分用 REST API 给出，REST API 可以支持任何支持发送 HTTP 请求的设备，降低了服务端开发和客户端开发的耦合，尽量使得服务端内部实现对于客户端透明。实现服务器端后，Web 可通过 JavaScript 来获取 JSON 格式的数据数据，也容易在此 API 基础上进一步进行原生移动端的开发。

接口列表

URL	HTTP	功能
/user	POST	用户注册
/login	GET	用户登录
/user/<userId>	DELETE	用户删除
/series	POST	添加剧集
/series	GET	查询剧集
/series/<seriesId>	PUT	更新剧集
/series/<seriesId>	DELETE	删除剧集

用户管理部分

用户注册

创建新用户需要向 user 路径发送 POST 请求，username 和 password 字段是必须的，如：

```
curl -X POST \
-H "Content-Type: application/json" \
-d '{"username":"abc","password":"fdasdf@#$"}' \
http://api.example.com/v1/users
```

创建成功时，HTTP 返回 201 Created，返回主体是一个 JSON 对象，包括一个 sessionId 字段，用来验证用户随后的请求，以及一个用户 ID 字段 userId。

失败 HTTP 返回 400 INVALID REQUEST，表示请求有错误（如字段不规范）。

用户登录

在用户注册后,可以通过向 `login` 路径发送 GET 请求登陆,必须附上 `username` 和 `password` 作为 URL 编码后的参数,如:

```
curl -X GET \  
-G \  
--data-urlencode 'username=abc' \  
--data-urlencode 'password=fdasdf@#$' \  
http://api.example.com/v1/login
```

登陆成功返回 200 OK, 主体包括 `username` 字段、`userId` 字段和 `sessionToken` 字段。

登录失败返回 401 Unauthorized, 表示用户没有权限(令牌、用户名、密码错误)。

用户删除

通过向 `user/<userId>` 发送 DELETE 请求可以删除用户, 请求头必须附上 `sessionToken` 验证, 如:

```
curl -X DELETE \  
-H "sessionToken: SESSIONTOKEN" \  
http://api.example.com/v1/user/USERID
```

删除成功返回 204 NO CONTENT, 同时用户相创建的剧集也会一并删除。

失败返回 401 Unauthorized, 或 404 NOT FOUND 表示要删除的用户不存在。

剧集信息管理部分

添加剧集

发送 POST 到 `series` 路径下可以添加剧集。添加剧集需要在登录的条件下进行, 也就是请求时必须附上有效的 `sessionToken`。另外, 还需要发送的字段包括剧集名 `name`, 当前集数 `current`, 是否公开 `public`, 以及可选的预期集数字段 `total`, 可选备注字段 `note`, 如:

```
curl -X POST \  
-H "sessionToken: SESSIONTOKEN" \  
-H "Content-Type: application/json" \  
-d '{"name": "person of interest SE 1", "current": "1", "total": "24", "note": "exciting!"}' \  
http://api.example.com/v1/series
```

创建成功时, HTTP 返回 201 Created, 返回的主体包括添加剧集时发送的所有字段、创建者用户 ID 字段 `createdBy`、创建时间 `createdAt` 字段、更新时间 `updatedAt` 字段、以及剧集 ID 字段 `seriesId`, 共同构成**剧集信息字段**(`name`, `current`, `public`, `total`, `note`, `createdBy`, `createdAt`, `updatedAt`, `seriesId`,)。

创建失败时, 返回 401 Unauthorized, 或 400 INVALID REQUEST。

更新剧集

发送 PUT 请求到 `series/<seriesId>` 可更新剧集信息，需要在登录的条件下进行，请求时必须附上创建者有效的 `sessionToken`。请求体中字段**剧集信息字段**中的 `name`, `current`, `total`, `note`, `createdAt`, `updatedAt`, 所有字段均为可选。如果没有相应字段则表示不更新该字段的信息。如不指定 `updatedAt`, 该字段则更新为当前服务器时间。

成功时，HTTP 返回 201 Created。返回更新后的**剧集信息字段**。

更新失败时，返回 400 INVALID REQUEST, 或 401 Unauthorized, 或 404 NOT FOUND。

查询剧集

发送 GET 到 `series` 可获取所有剧集信息，请求时可附上用户的 `sessionToken`, 获取自己的追剧情况和其他公开的追剧情况；若不附上 `sessionToken` 则只返回所有公开的追剧情况。

可通过 URL 编码后的参数过滤用户和记录数量，如：

`?limit=10`：指定返回剧集记录的数量

`?offset=10`：指定返回剧集记录的开始位置。

`?createdBy=abc`：指定筛选用户

注意：若指定了 `createdBy=abc` 只会返回由 `abc` 创建的公开条目，无论是否当前附上了 `abc` 的 `sessionToken`。

成功时，HTTP 返回 200 OK。返回结果主体的是一个 JSON 数组，每个数组元素字段定义同**剧集信息字段**，如：

```
"series":[
  {
    "seriesId": "32",
    "name": "POI SE1",
    "current": 12,
    "public": true,
    "total": 24,
    "note": "exciting",
    "createdBy": "abc",
    "createdAt": "2011-05-26T07:56:00.123Z",
    "updatedAt": "2011-05-27T11:16:00.123Z"
  },
  {
    "seriesId": "34",
    "name": "POI SE2",
    "current": 14,
    "public": true,
    "total": 24,
    "note": "exciting!",
    "createdBy": "abc",
    "createdAt": "2013-05-26T08:50:00.123Z",
    "updatedAt": "2014-07-27T17:46:00.123Z"
  }
]
```

查询失败时，返回 401 Unauthorized, 或 404 NOT FOUND。

删除剧集

发送 DELETE 到 `series/<seriesId>` 可删除剧集信息，需要在登录的条件下进行，请求时必须附上创建者有效的 `sessionToken`。删除成功时，返回 204 NO CONTENT。

失败时，返回 401 Unauthorized, 或 404 NOT FOUND 表示要删除的剧集不存在。

错误处理

如果状态码是 4xx, 会向用户返回出错信息，返回体 `error` 为出错信息字段，如：

```
{error: "Invalid userId"}
```