|  |  |
| --- | --- |
|  |  |
|  | |
| RESTful接口开发规范 | |
|  | |
| TP 系统  2018年03月 | |

|  |
| --- |
| 目 次 |

[1 引言 1](#_Toc432577387)

[1.1 编写目的 1](#_Toc432577388)

[1.2 阅读对象 1](#_Toc432577389)

[2 工具说明 1](#_Toc432577390)

[2.1 开发工具 1](#_Toc432577391)

[2.1.1 JDK 1](#_Toc432577392)

[2.1.2 Eclipse 1](#_Toc432577393)

[2.1.3 Tomcat 1](#_Toc432577394)

[2.1.4 Maven 1](#_Toc432577395)

[2.2 第三方软件包 2](#_Toc432577396)

[2.3 测试工具 2](#_Toc432577397)

[3 编码规范 2](#_Toc432577398)

[3.1 通用Java编码规范 2](#_Toc432577399)

[3.1.1 排版 2](#_Toc432577400)

[3.1.2 类与接口 2](#_Toc432577401)

[3.1.3 注释 2](#_Toc432577402)

# 引言

* 1. 编写目的

规范RESTful接口开发

* 1. 阅读对象

开发人员

# 工具说明

## 开发工具

## 第三方软件包

## 测试工具

# RESTful API规范

## 基本规范

### 协议

尽量使用HTTPS协议。

### 版本

将API的版本号放入URL。如<https://api.trendytech.com/v1/>

### 认证

使用OAuth2.0框架进行身份认证。

### 数据格式

请求体和返回体都采用JSON格式。

## 请求规范

### 路径（Endpoint）

路径以资源为单位定义，尽可能简短，若为词组用下划线“\_”隔开，并保持形式统一（一般为复数形式）。如：

数据库中用户表为users，则url定义为：<https://api.trendytech.com/v1/users>

### HTTP行为

最常用的行为：

* GET（SELECT）：从服务器取出资源（一项或多项）。
* POST（CREATE）：在服务器新建一个资源。
* PUT（UPDATE）：在服务器更新资源（客户端提供改变后的完整资源）。
* PATCH（UPDATE）：在服务器更新资源（客户端提供改变的属性）。
* DELETE（DELETE）：从服务器删除资源。

另外还有：

* HEAD：获取资源的元数据。
* OPTIONS：获取信息，关于资源的哪些属性是客户端可以改变的。

示例：

GET /users 获取用户列表

GET /users/{userId} 获取指定用户

POST /users 新建用户

PUT /users/{userId} 更新指定用户信息，提供完整信息

PATCH /users/{userId} 更新指定用户信息，提供修改部分信息

DELETE /users/{userId} 删除指定用户

GET /users/{userId}/roles 获取指定用户的角色列表

### 查询过滤条件和分页

如果记录数量很多，服务器不可能都将所有数据返回给用户。API应该提供参数，过滤返回结果。

条件参数定义如下：

* 分页

|  |  |  |
| --- | --- | --- |
| 参数名 | 描述 | 示例 |
| limit | 记录总数，设置默认值 | ?limit=1000 |
| offset | 记录开始位置 | ?offset=2 |

* 排序

|  |  |  |
| --- | --- | --- |
| 参数名 | 描述 | 示例 |
| sortby | 排序字段 | ?sortby=id |
| order | 排序顺序，升序asc，降序desc | ?order=asc |

## 响应规范

### 返回状态码

服务器需要向用户返回状态码和提示信息，常用状态码如下：

* 200 OK - [GET]：服务器成功返回用户请求的数据，该操作是幂等的（Idempotent）。
* 201 CREATED - [POST/PUT/PATCH]：用户新建或修改数据成功。
* 202 Accepted - [\*]：表示一个请求已经进入后台排队（异步任务）
* 204 NO CONTENT - [DELETE]：用户删除数据成功。
* 400 INVALID REQUEST - [POST/PUT/PATCH]：用户发出的请求有错误，服务器没有进行新建或修改数据的操作，该操作是幂等的。
* 401 Unauthorized - [\*]：表示用户没有权限（令牌、用户名、密码错误）。
* 403 Forbidden - [\*] 表示用户得到授权（与401错误相对），但是访问是被禁止的。
* 404 NOT FOUND - [\*]：用户发出的请求针对的是不存在的记录，服务器没有进行操作，该操作是幂等的。
* 406 Not Acceptable - [GET]：用户请求的格式不可得（比如用户请求JSON格式，但是只有XML格式）。
* 410 Gone -[GET]：用户请求的资源被永久删除，且不会再得到的。
* 422 Unprocesable entity - [POST/PUT/PATCH] 当创建一个对象时，发生一个验证错误。
* 500 INTERNAL SERVER ERROR - [\*]：服务器发生错误，用户将无法判断发出的请求是否成功。

### 返回结果

GET请求应返回资源对象或列表，创建或更新操作应返回资源对象。

* GET /users：返回资源对象的列表（数组）
* GET /users/{userId}：返回单个资源对象
* POST /users：返回新生成的资源对象
* PUT /users/{userId}：返回完整的资源对象
* PATCH /users/{userId}：返回完整的资源对象
* DELETE /users/{userId}：返回一个空文档

### 异常信息

当服务器返回4xx或5xx系列的状态码，则需要提供可理解的异常信息。

如：

返回状态码401，返回体{error: “User Unauthorized”}