文档编号：10003

GO API

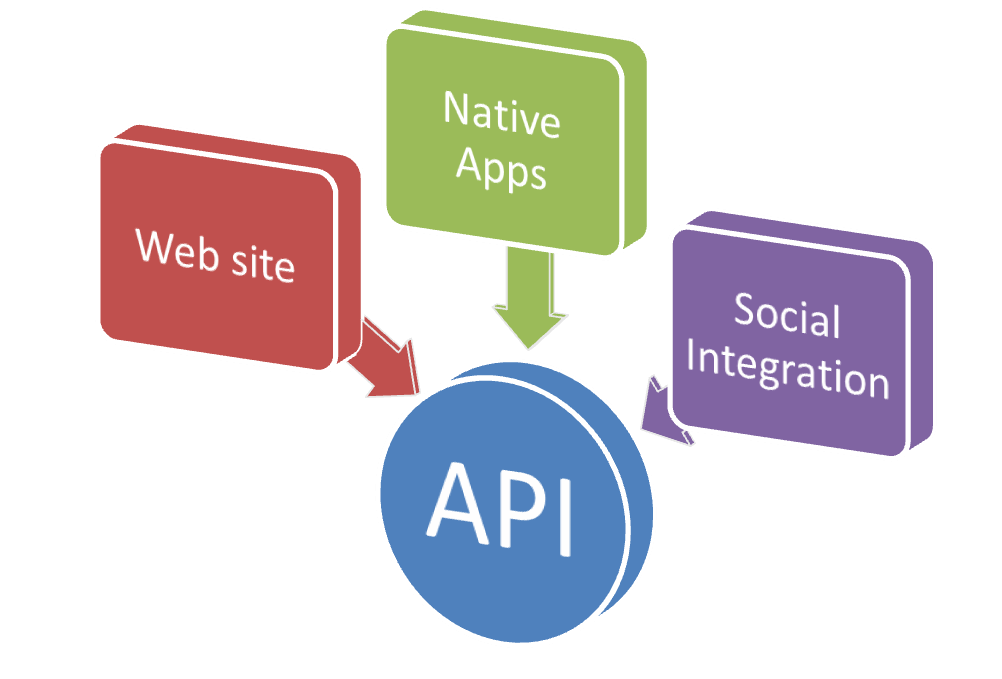
1.GO API 接口总览

GO API接口为web端、iOS端和Android端提供多种接口服务。

概述

GO API 接口服务器假设在阿里云服务器上，采用Restful协议框架，基于Maven web项目，通过Spring、SpringMVC和Mybatis框架实现。GO API主要支持四种HTTP协议方法：POST（创建操作）、GET（查询操作）、PUT（更新操作）、DELETE（删除操作）等操作对应数据库典型的CRUD操作，提供相应接口服务。GO API集成客户端重复使用的服务，大大减少项目工作量，同时，使GO项目尽可能地前后端分离。

GO API 接口可以通过采用不同的HTTP协议方法进行URL访问，获得相应的接口服务。接口以JSON数据包方式返回相应数据和返回码，web端可以采用js解析，iOS和Android端可以采用相应第三方框架解析。



参考文献（ctrl+左键 访问）：

1、[什么才是真正的 RESTful 架构](http://blog.csdn.net/lz0426001/article/details/52370193)

更新日志

2017.3.16

>1测试版

>2内容：GET、POST、PUT、DELETE四种操作，以及测试类Test。

2017.4.6

>1正式版v1.0

>2内容：个人用户、组织用户相关接口。

数据规范

调用接口时，数据或参数必须符合API接口文档规范，尽量保证顺序前后一致（乱序也有效）。

API接口服务程序设计各相关类，需保证无相同名称变量。

接口简介

响应状态码（返回码）说明

用户每次调用接口时，都会获得正确或错误的状态码。用户可以根据相应的状态码和错误信息，排查错误。

|  |  |  |
| --- | --- | --- |
| 响应状态码 | 信息 | 备注 |
| 200 | OK | 请求成功 |
| 10000 | invalid parameters | 无效的参数 |
| 10001 | database error | 数据库错误 |
| 10002 | query failed | 查询失败 |
| 10003 | verification failed | 认证失败 |
| 10004 | login failed | 登录失败 |
|  |  |  |

2.接口指南

通过http相应请求方式访问API接口地址，获得响应的接口服务。

接口调用地址

GO API接口基地址

http://shnuecp.com/comingo%20Maven%20Webapp/

支持的四种HTTP请求方式：

GET 查询资源

POST 创建资源

PUT 更新资源

DELETE 删除资源

接口调用样例

http请求（GET方法）URL地址：

http://shnuecp.com/comingo%20Maven%20Webapp/test?id=0b117007-6de6-4763-b332-cb109a9e7bfe

返回Json包：

{

"msg": "OK",

"statuscode": 200,

"test": {

"createTime": {

"date": 22,

"day": 3,

"hours": 21,

"minutes": 19,

"month": 2,

"nanos": 0,

"seconds": 15,

"time": 1490188755000,

"timezoneOffset": -480,

"year": 117

},

"id": "6157f400-97a2-4e34-a55f-0aaf433f3578",

"remark": "test",

"username": "zhangchi"

}

}

请求成功，返回如下信息

响应状态码 响应信息

200 OK

如没有id信息将返回如下信息：

响应状态码 响应信息

10000 invalid paramters

接口测试用例

Test类

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/test

GET方法

说明：根据id查询用户信息。

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 是否必须 | 选项 | 备注 |
| id | 用户id | 是 |  |  |

返回JSON数据包参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 类型 | 选项 | 备注 |
| id | 用户id | String |  |  |
| username | 用户名 | String |  |  |
| remark | 备注 | String |  |  |
| createTime | 创建时间 | Int |  |  |

POST方法

说明：创建新用户。

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 是否必须 | 选项 | 备注 |
| username | 用户名 | 否 |  |  |
| remark | 备注 | 否 |  |  |

返回响应状态码：

200

PUT方法

说明：更新用户信息。

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 是否必须 | 选项 | 备注 |
| id | 用户id | 必须 |  |  |
| username | 用户名 | 否 |  | 如不更改，需置为原username |
| remark | 备注 | 否 |  | 如不更改，需置为原remark |

返回响应状态码：

200

DELETE方法

说明：根据用户id删除用户。

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 是否必须 | 选项 | 备注 |
| id | 用户id | 是 |  |  |

返回响应状态码：

200

3.具体接口文档

3.1个人用户

3.1.1个人用户信息获取

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/userinfo

GET方法

说明：根据id查询个人用户信息。

请求参数：

响应参数：

样例：

3.1.2个人用户注册

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/userinfo

POST方法

说明：个人用户注册。

请求参数：

响应参数：

样例：

3.1.3个人用户登录

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/userlogin

POST方法

说明：个人用户登录。

请求参数：

响应参数：

样例：

3.2组织用户

3.2.1组织用户信息获取

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/orginfo

GET方法

说明：根据id查询组织用户信息。

请求参数：

响应参数：

样例：

3.2.2组织用户注册

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/orginfo

POST方法

说明：组织用户注册。

请求参数：

响应参数：

样例：

3.2.3组织用户登录

请求URL：

http://shnuecp.com/comingo%20Maven%20Webapp/orglogin

POST方法

说明：组织用户登录。

请求参数：

响应参数：

样例：

3.3