

用户端小程序接口列表

接口统一前缀地址: <https://domain/api/user>

请求统一参数:

token (wxLogin接口除外)

统一返回结果:

```
{
  code: 状态码 (0表示成功, 其他为失败),
  message: 返回信息(成功时一般为请求成功, 失败时为失败信息),
  data: 返回数据, 始终是一个对象(成功时存在, 失败时一般不存在, 除非特殊情况需要数据时才存在)
}
```

全局状态码:

- 0 成功
- 500 未知错误
- 10000 接口不存在
- 10001 参数不合法
- 10003 未登录
- 10004 账号不存在
- 10005 密码错误
- 10006 账号已存在
- 10007 token无效或不存在
- 70010 微信openId不存在(此时需重新调用 wx.login 并调用/wxLogin接口以获取微信openId)
- 70011

用户端登陆机制 (即 token , openId, 用户账号之间的关系):

步骤:

1. token获取: 调用 wx.login, 获取code及appid, 然后请求 /wxLogin 接口, 获取token, 并缓存在本地
2. 绑定openId 到账户: 通过 /login 接口, 将用户手机号(即用户实际账号)与openId相关联

流程:

首次进入小程序后, 首先 获取token, 然后 绑定openId 到账户

后续每次进入小程序, 首先获取本地token缓存, 若不存在则 获取token , 获取后若不存在用户信息,则需要登录 (即步骤2)

登陆相关

☐ 微信登陆接口

接口地址 **POST** /wxLogin

参数:

code wx.login 获取到的code, wx.login需要使用withCredentials选项获取
appid wx.login 获取到的appid, 用于区分运营中心

返回值:

```
data: {  
  token: 用户令牌,  
  userInfo: 用户信息, 微信openId已绑定用户的情况下返回  
}
```

☐ 登陆接口(绑定手机号)

接口地址: POST /login

参数:

```
mobile 手机号  
verify_code 验证码
```

返回值

```
data: {  
  userInfo: 用户信息  
}
```

☐ 短信接口

接口地址: POST /sms/verify_code

参数:

```
mobile 手机号
```

返回值:

```
data: {  
  verify_code: 验证码, 测试时存在  
}
```

☒ 获取地区列表(树形结构)

接口地址: GET /area/tree

参数: tier 要获取的层级数 1-3 (省/市/县), 默认为3

返回:

```
data: {  
  list: [  
    {  
      id: id,  
      area_id: 地区ID,  
      name: 地区名,  
      sub: [  
        子地区列表, 有下级时存在, 结构同父级结构  
      ]  
    }  
  ]  
}
```

商家模块

- ☐ 获取商家类别列表
- ☐ 获取商家列表 (关键字搜索, 附近商家等)
- ☐