

- Author: Yunpeng Chen
- Contact: mathcyp2008[at]hotmail.com
- Changelog:
 - 2014.07.15 Initial version.
 - 2014.07.31 Edited by Ji Yang (yangji9181@gmail.com)

一,URL说明

url相对路径	HTTP variable	功能
/register/	POST	注册
/login/	POST	登陆
/logout/	POST	登出
/edit/	POST	编辑资料(密码除外)
/edit/password/	POST	修改密码
/info/	POST	获取个人资料
/fblogin/	POST	通过facebook登录

二，数据格式

(1)注册功能:

请求数据格式: /register/

```

{
  email: string,      //required, 这个在数据库中是要保证唯一的
  password: string,   //required
  nick_name: string,  //required
  first_name: string, //optional, default 'Unknown'
  last_name: string,  //optional, default 'Unknown'
  gender: string,     //optional, default 'Unknown'
  phone: string,      //optional, default 'Unknown'
  eas_id: string,     //学校教务系统登录账号, optional, default 'Unknown'
  tpa_type: string,   // 第三方登录类型 (如: 'facebook'), optional, default
'Unknown'
  tpa_id: string,    // 第三方登录账户id, optional, default 'Unknown'
  image: string,     //头像地址, optional, default 'Unknown'
  university: string, // 必须是shortname, 比如uiuc, ucb, purdue; optional,
default 'Unknown'
  major: string,     // 必须是shortname, 比如cs, ee; optional, default 'Unknown'
}

```

返回数据格式:

```

JSON:
{
  status: string,
  message : string,
}

```

(2)登陆功能: /login/

请求数据格式:

```

{
  email: string, //required
  password: string //required
}

```

返回数据格式:

```
JSON:
{
  status: string,
  message : string,
  data : {
    user: string, //用户id
    ip:      string,
    token:   string //登出前一直有效, 请妥善保管
  }
}
```

(3)登出功能: /logout/

请求数据格式:

```
{
  token: string // 登录时获得
}
```

返回数据格式:

```
JSON:
{
  status: string,
  message : string
}
```

(4)编辑信息功能: /edit/ (修改非密码信息)

请求数据格式:

```

{
  token: string,          //登录时获得, required
  password: string,       //required
  email: string,          //required
  nick_name: string,      //required
  first_name: string,     //required
  last_name: string,      //required
  gender: string,         //required
  phone: string,          //required
  eas_id: string,         //required
  tpa_type: string,       //required
  tpa_id: string,         //required
  image: string,          //required
  university: string,     //required, shortname
  major: string,          //required, shortname
}
}

```

返回数据格式:

```

JSON:
{
  status: string,
  message : string,
}

```

(5)修改密码: /edit/password/

请求数据格式:

```

{
  token: string,          //登录时获得, required
  old_password: string,   //required
  new_password: string    //required
}

```

返回数据格式:

```
JSON:
{
  status: string,
  message : string,
}
```

(6)获取个人资料: /info/ 请求数据格式:

```
{
  token: string, // 登录时获得
}
```

返回数据格式:

```
JSON:
{
  status: string,
  message : string,
  data: {
    first_name: string,
    last_name: string,
    nick_name: string,
    password: string,
    gender: string,
    image: string,
    eas_id: string,
    tpa_type: string,
    tpa_id: string,
    university: integer,
    email: string,
    phone: string
  }
}
```

(7)facebook登录: /fblogin/ 请求数据格式:

```
{
  code: string, // 客户端通过sdk或访问授权页获取的短期令牌
}
```

返回数据格式：

```
JSON:
{
  status: string,
  message : string,
  data : {
    user_id: string,
    ip:      string,
    token:   string // 本系统用户令牌，登出后失效
  }
}
```

三，返回数据说明

1， status 字段

status code	功能
"success"	请求成功
"fail"	请求失败（客户端提交数据错误等）
"error"	服务器错误（数据库不一致等异常情况）

2， message 字段

用于请求发生错误时返回错误信息

3， data

部分请求中用于返回客户端所需的数据。 例如：登录时，需要返回'token'变量给客户端，于是返回信息data['token']中可获取客户端所需token信息。

四，解释说明

1， /register 普通注册

- tpa_type和tpa_id用于第三方登陆，普通注册时上述字段可为空。
- 普通注册中，email是用户的唯一标识信息，需保持唯一
- University及Major域需要给简称，不能识别的一律存为'Unknown'，所有注册时optional的域没有上传的话默认存为'Unknown'。
- 提交的用户数据需严格与personal.models文件中的规定保持一致。因后续功能需要，

model仍有可能做微调。

2, /login & /logout 用户登录与登出

- 用户登录时，获取的token长期有效。
- 同一用户允许同时在不同地点登录。

3, /edit & /edit/password 修改用户信息

- /edit/接口用于修改除用户密码之外的其他用户信息
- 用户email不可修改

4, /fblogin/ 使用facebook第三方登录

- 此模块使用了第三方库, 依赖库的安装参考链接[链接](#)
- 根据在facebook developer注册的appid、secret key以及指定的跳转uri生成确定的 Authorization URL，客户端程序可通过此URL获取code值并将此code作为第三方登录请求数据发送至后端程序。
- 用户第一次登录时，后端程序会从facebook获取用户基本信息并为用户创建本系统中的账户。以后再次使用同样的facebook id进行第三方登录，则直接使用第一次创建的账户直接登录系统。
- [facebook token参考文档](#)