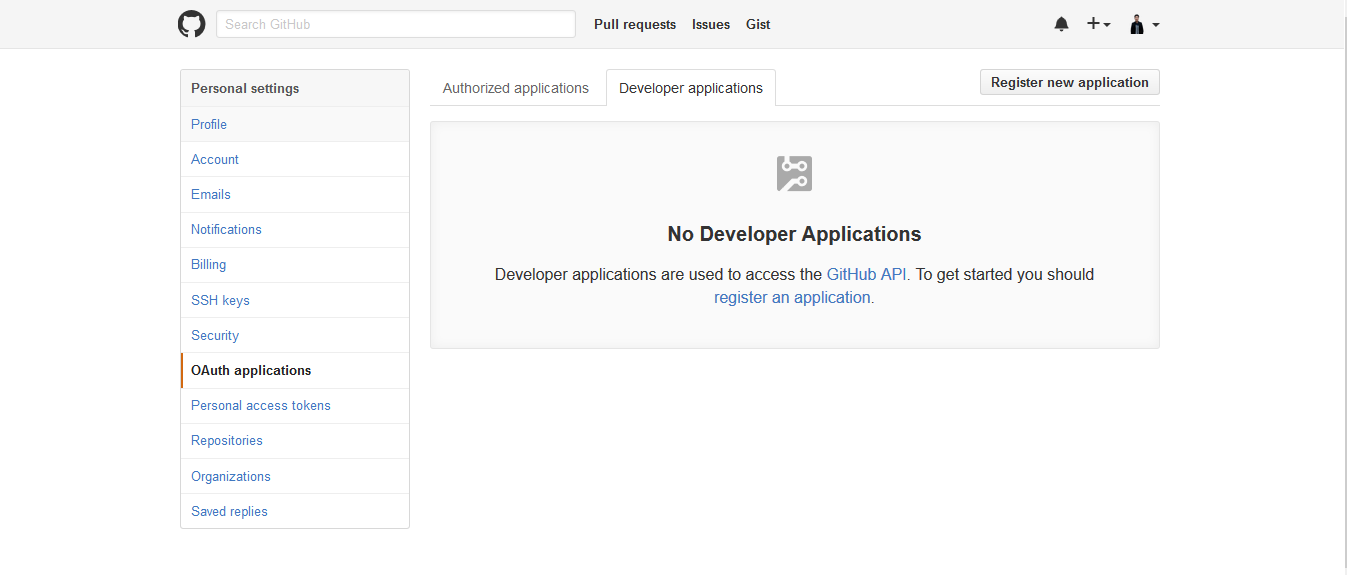
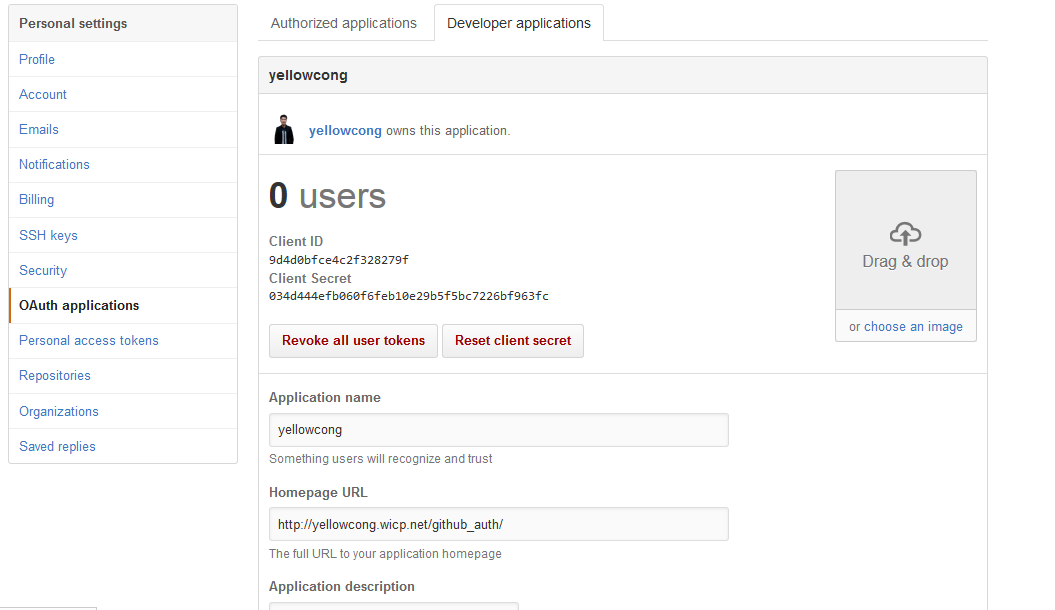
# 注册Application



# 创建应用



Client ID

9d4d0bfce4c2f328279f

Client Secret

034d444efb060f6feb10e29b5f5bc7226bf963fc

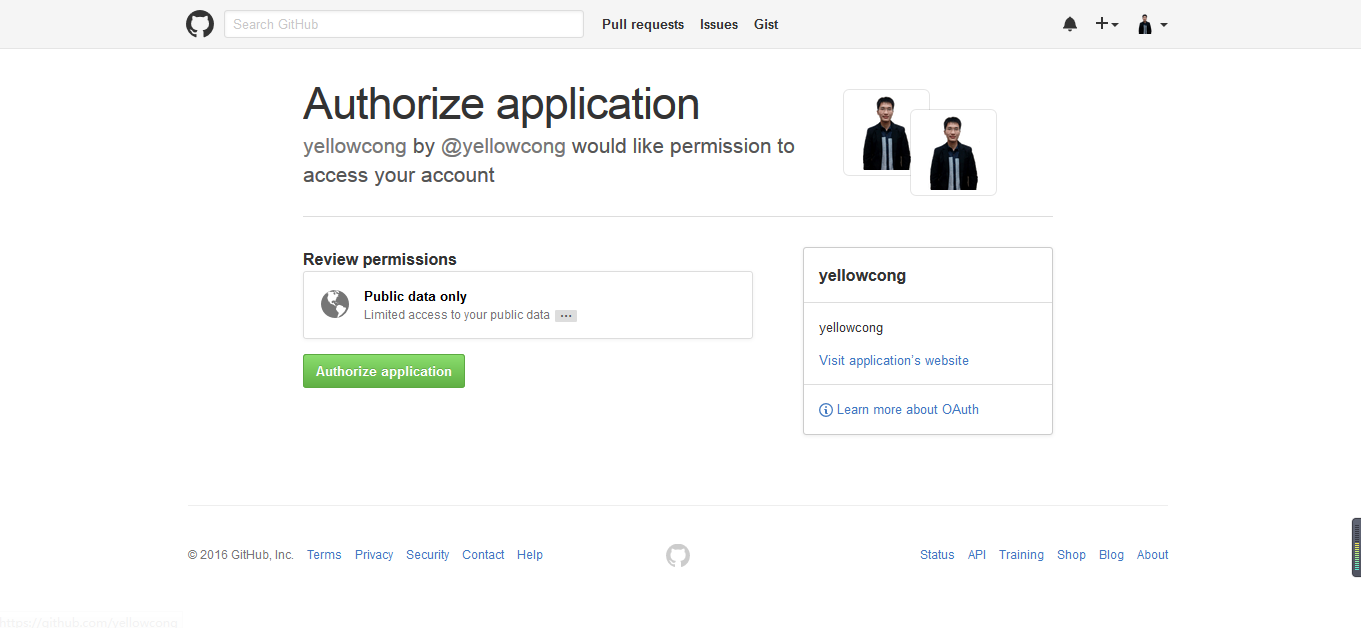
# 授权的思路

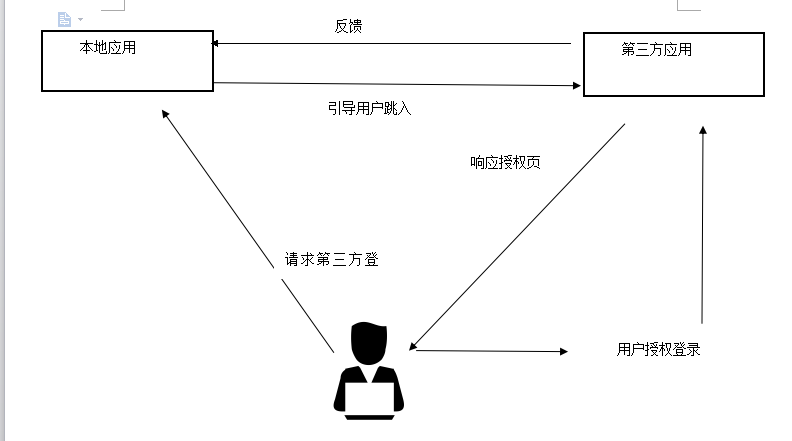
获取授权

获取token

获取用户id

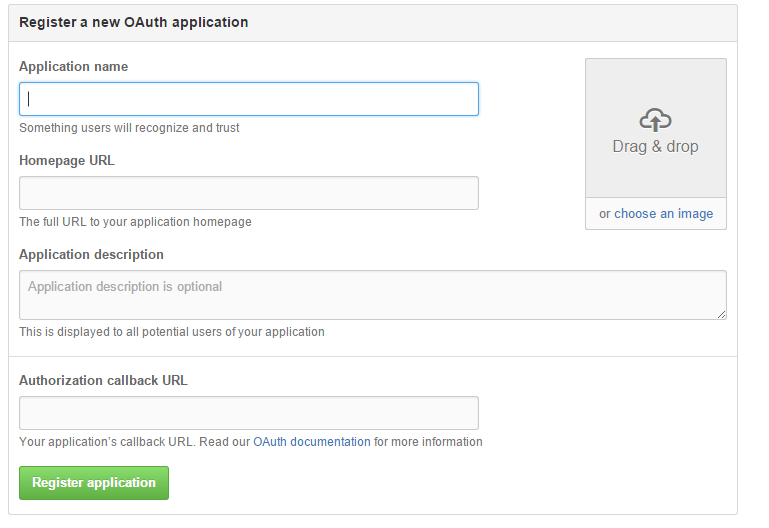
跳转到授权页面



流程图如下：[](http://static.oschina.net/uploads/space/2015/0828/174836_eNK9_1778309.png)

说明：本文以github登录为例

**注册github账号,登录后在个人信息页面一次点击Applications>Developer applications>Register a new application弹出如下界面**

[](http://static.oschina.net/uploads/space/2015/0828/175824_LL3r_1778309.png)

前三项就依据个人应用添加      最后callback URL 就是回调的地址了   添加对应平台的接口  github允许填写本地连接地址（开发中直接填[http://127.0.0.1:8080](http://127.0.0.1:8080/api/xxx)/xxx/xxx   这个地址要和授权的时候的传入的地址参数一致 否则授权失败）

注册之后 会得到  github提供的client id 和client secret 有了这两个东东就可以换取更多的信息了

流程图说明：

1、用户点击github登录本地应用引导用户跳转到第三方授权页这里就不拐歪了 跳转地址为：

https://github.com/login/oauth/authorize?client\_id=xxxxx&state=xxx&redirect\_uri=xxxx

    参数说明  :client\_id  就不说是啥了吧   redirect\_uri：就是上面刚刚填的啦   state:随机字符串（授权成功会原样返回）

2 、授权成功后会回调我们平台，响应信息中会带code其他参数不理他了哈哈，我们带着这个code再次访问github 地址：

https://github.com/login/oauth/access\_token?client\_id=xxxx&client\_secret=xxx&code=xxx&redirect\_uri=xxxx

参数不解释了，这次会得到响应的access\_token

3、成功获取access\_token后用这玩意再去换取用户信息 地址：

https://api.github.com/user?access\_token=xxx

之后我们就能得到用户信息了，接下来就可以做想做的事情了

其他平台思路都一样了

# 封装

1.我们可以将github的数据封装到properties的文件中，然后建立一个类，静态加载他， 然后获取到里面的数据

2.将我们的url封装到Auth这个类中