

## 实训第四周报告

学号	姓名	导师	研究方向	报告周期
17343105	田皓	余阳	Go-Online 在线编程项目开发	第 4 周 (11.24 - 12.1)

## 本周内容

### 关于签到系统

巨大问题：dwebsocket 弃用了！

在开发过程中遇到这样的问题：前端断开连接后后端报错，虽然不影响使用但是实在太难看了。

基本把网上的 dwebsocket 的博客都看了，也有人说了同样的问题，但是解决方法一个比一个不对，最后在 dwebsocket 的 github 上看到了这句话，找到了 dwebsocket 依赖于这个地址：

The dwebsocket module based django-websocket development, django-websocket module in <https://github.com/gregmuellegger/django-websocket>. Thank you Gregor Müllegger.

然后，作者弃用了，只能怪自己图快用了 dwebsocket，上一周就当交学费了。

### django-websocket (ABANDONED, do not use)

THIS PROJECT IS ABANDONED! Please use [django-channels](#) to implement websockets with Django on the server.

For legacy reasons, you can access the old documentation in the [README of version 0.3.0](#).

没办法只能乖乖使用 channels。

不过也并不复杂，按照教程搞好，写一个自己的 consumer 就可以了。

具体实现思路：

连接开始，判断各种信息符合要求，新开一个线程每隔固定时间向前端发送签到码，收到断开信号之后关闭子线程并断开连接。

遇到的问题是连接断开后后端还一直在 print 签到码，所以得想办法结束子线程，最后还是用一个比较笨的方法解决：发送签到码的子线程用一个全局变量 flag 控制，disconnect 的时候 flag = 0。下面是部分代码：

```
flag = 1
```

```
def send_code(self, courseID, classID):  
    while True and flag == 1:
```

```
time.sleep(3)
utils.generateCode(courseID, classID)
checkCode = utils.getCheckCode()
print(checkCode)
self.send(text_data=json.dumps({
    'data': checkCode
}))
```

在本地测试已无问题，需要再与前端商议一下验证身份的方式。

## 其他

### 简单学习了 **docker**

安装 Community Edition CE 社区版

- 安装教程 [菜鸟 阮一峰](#)

避免每次 `sudo` 加入用户组 [docker 官方文档](#)

```
sudo groupadd docker
sudo usermod -aG docker $USER
```

重启电脑 或者 `newgrp docker` 激活对组的更改

docker 是 CS 架构，与 mysql 一样，需要本机启动 docker 服务

```
sudo service docker start
sudo systemctl start docker
```

- image 文件

```
# 列出本机的所有 image 文件。
docker image ls
```

修改默认仓库

打开 `/etc/default/docker` 加上：

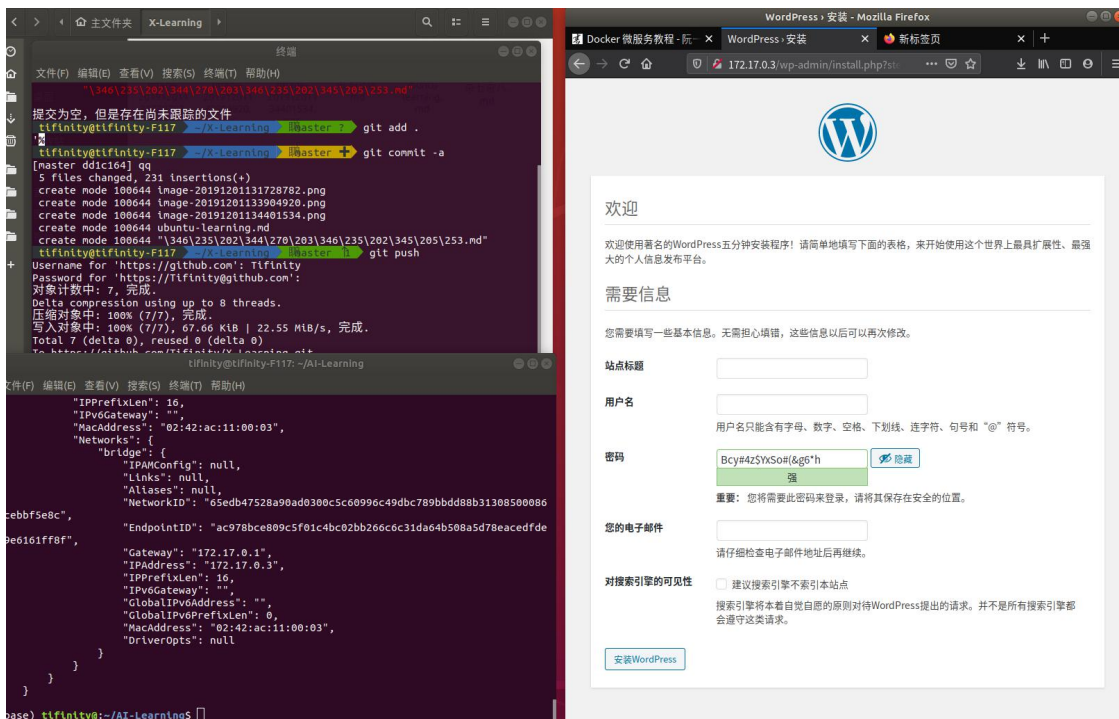
```
DOCKER_OPTS="--registry-mirror=https://registry.docker-cn.com"
```

然后重启 docker 服务

```
sudo service docker restart
```

- container

镜像与容器就像程序与进程，跟着阮一峰的教程做了 wordpress，对 docker 有了一个初步的认识。



## 学习收获

- chanel 简单实现签到功能
- 初步使用 docker

## 存在问题

- 对别人的模块理解还是太表面，一度陷入盲目求快的陷阱中

## 下周计划

- 完善项目