个人总结 by zyx

文档方面自己主要负责需求规格文档的整合，写部分设计文档内容。代码方面，自己主要写的是数据层的代码，包括文档持久化和网络接口。

文档方面可说的不多，感觉比上学期文档写起来顺手多了，可能也是经验足的原因吧，总之文档方面问题不多（其实讲道理迭代一设计文档还没写完）。

代码方面，由于写网络连接，爬虫部分，接触了大量新的知识，包括java中调用python代码（处理json），访问url，甚至还花了不少于20个小时自学了html标记语言以有助于以后迭代直接爬虫抓取网页源码。感觉收获非常多，毕竟这些满满的都是知识，有一种不断打开新世界大门的感觉。另外在软工二的团队合作中我们组是采用的按功能点分着写代码，虽然采用了分层架构但是并没有分层写，所以每个人的交互都不多。软工二的项目只需要自己做三层的内容不需要和别人交互，效率方面也是完完全全不考虑。但这次软工三我们是分层写代码，其实自己挺想写中间一层能够同时交互上下两层的吧，但是写最底层也非常棒，由于所有人都要依赖自己的代码所以对自己的代码的稳定性，普适性，以及效率的要求非常高。自己也因为为了一点效率的提升把url访问，处理从url上读取信息这两个效率敏感的部分用了不同的方法写以获得最高效率，所以这是完全不同的体验啊。做一个服务的提供者的感觉是很好的。

代码方面的不足其实也是有的，我觉得最主要的就是持续集成方面做得不好（其实是压根没做），但是这次项目彻底地让我感受到了持续测试的重要性。经常是我写的底层服务提供的代码，自己写个main类随意地跑那些方法就当做测试了，但这样在之后更改代码的时候又常常忘了测试，导致其他人反映我的代码出bug，非常不好意思。出bug后还要自己再去读懂很久以前写的测试代码再去测试，有时全都忘了自己在什么地方做过那个修改，非常浪费时间，所以下一个迭代要持续地利用junit测试自己的接口，真切地让每个接口都在每一个发布之前都处于能通过测试的状态，所以要先写好测试代码，测试驱动开发。

2016.3.9