

Uknow InfoHub

Experience and Conclusion Report

BIXLRSMB Team

Contents

1	Introduction	2
1.1	Document Introduction	2
2	Conclusions	2
2.1	陈俊男(vera, 2011011235)	2
2.2	周昕宇(zxytim, 2011011252)	3
2.3	赵一开(blahgeek, 2011011262)	3
2.4	吴育昕(ppwwyyxx, 2011011271)	3
2.5	贾开(jiakai, 2011011275)	4
2.6	刘啸宇(vuryleo, 2011011434)	4

1 Introduction

1.1 Document Introduction

This is the experience and conclusion that provided by everyone in develop team, which shows what we have learned and what we wish.

The order is based on student ID.

2 Conclusions

2.1 陈俊男(vera, 2011011235)

Although I am the weakest in programming skills among our team, I have still learnt a lot with the great help of my teammates during this period working with everybody in Uknow Infohub.

The first thing that benefits me greatly is the ability in team work. Without any doubt, team work is truly important in modern life. This working experience in Software Engineering Team provided me an access to experience this wonderful days. It begins with a definition of "team". In Uknow Infohub, we define a team as a group of individuals that support and respect each other.

We made a detailed schedule on when we should discuss or assign works. Thanks to this good habit, our team work possesses are harmonious and complimentary skills. We work in harmony with each other and our various skills could actually compliment and advance the process and the project forward with purpose.

Another thing that benefited me greatly is that I learnt about the use of UML. It is a little amazing when I figured out that The Unified Modeling Language - UML - is OMG's most-used specification, and the way the world models not only application structure, behavior, and architecture, but also business process and data structure. When we started using UML to manage our process, the benefit started to appear. With UML, our team can achieve a higher level of abstraction that is critical for proper planning of Uknow Infohub. First of all, using UML made us know exactly where we are getting with a lower development costs. Besides, our Uknow Infohub could behave exactly as we expected it to. And by using models, the right decisions are made before we started writing poor code which lower down the overall costs. Working with a new developer would be easier when we set the models. Last communication with programmers and outside contractors would be more efficient.

In the end, I wanted to thank Miss Bai and my great teammates who have helped me a lot during these weeks. I hope everyone can get useful knowledge after taking the wonderful experience.

2.2 周昕宇(zxytim, 2011011252)

软工项目中,我主要负责后台 fetcher、prefilter 的实现、测试工作. 由于使用 python 作为开发语言,极大地提高工作效率,可以明显的感觉整个项目的开发周期大大缩短. python 作为“胶水语言”,可以方便地调用大量的第三方库,如自动标注标签的 prefilter 就利用了 nltk 库,大大减少了工作量.

以前从未对自己的工程做过系统的单元测试. 而这次在有单元测试的情况下,api 的正确性得到了有力的保证,代码覆盖情况一目了然,使得对于系统不断更新后的一致性得到了保障. 压力测试中也反映出 api 服务器的负载是相对均衡的,也说明了我们服务的支持的高并发性.

由于大家处于不同的班级以及不同的寝室,交流不如都在一个办公室里方便. 邮件等沟通方式由于其高延时和非强制性以及工程本身概念的抽象性,使得使得信息交流不畅,从而项目在初期进展不够快. 后来采取的类似敏捷软件开发(缺少其价值观中的客户协作)的开发方法后,项目推进速度极具加快. 事实验证敏捷开发对于一个人数不多的团队在产品原型搭建上是非常有效的.

2.3 赵一开(blahgeek, 2011011262)

第一次与多人合作写这样一个“大”项目,感触颇深.给“大”字加上引号,因为该项目的代码行数并不多,其大只是因为它的模块很多,功能很多,逻辑性很强. 完成这样一个项目,给我最多的感触是 Git 这样一个去中心化的版本控制工具非常方便多人在不同时间不同地点共同完成代码. 另外,Issue Tracker 这样一个系统对于标记和解决特定问题也非常有效率.但是我也感受到这两个工具有一些不足之处:

1. 对于两个人修改了相同部分的代码但是逻辑上无关时 merge 比较痛苦(比如全局修改变量名).
2. Issue Tracker 在代码增长阶段用处比较少,更加适合维护期的代码.

2.4 吴育昕(ppwwyyxx, 2011011271)

这次软工项目中,我与几位同学合作设计并开发完成了一个信息整合系统. 开发过程中,我深深意识到大型系统设计中架构的重要性. 有一个良好的架构,可以让后期的开发变得极为方便,也使得整个系统更加灵活和强大. 我们的信息收集模块,正是由于 fetcher-prefilter-postfilter 三层架构的存在,才能够分布式处理不同种类的用户定制的信息.

另外,这个项目也让我感受到前端开发的难处. 相比后端,网页前端受到的限制更多,编写、调试起来都更加麻烦. 前端的界面设计要想做的具有扩展性和维护性,更远非一日之功,这方面日后还需多加训练.

这次项目中我收获不小,但如果项目内容能够更加丰富多样,让我们能够有机会做出一个真正十分有用的东西,那将会更加有趣.

2.5 贾开(jiakai, 2011011275)

这次软工项目中我主要负责了后端系统的整体架构和部分实现,基于大量开源框架开发了一套扩展性较强的分布式信息抓取系统. 由于之前有类似的工作经验,整个过程较为顺利.

最让人开心的是今年软工终于放弃了对语言的限制,可以使用 python,而且使用 git 作为项目管理工具,这极大地提高了我们的开发热情和开发效率,缩短了开发时间. 希望以后也可以继续这样灵活下去.

在这次项目中,我最大的收获还是在团队协作和信息沟通方面. 组内成员的班级和寝室都不相同,平时主要通过邮件列表等方式通信布置任务. 而在具体实现方面,要想让别人理解自己的代码并且让大家都能共同开发,就需要有比较详细的内嵌于代码的文档,但最终发现最有效的交流办法还是直接面谈或者用一些即时聊天工具. 如何能把软件这种抽象的概念能简明扼要地向别人表达清楚,确实是一门值得长期研究和体会的艺术.

2.6 刘啸宇(vuryleo, 2011011434)

这次软工项目自己做为组长,切实体验了一次做为项目经理而非普通开发者的角色.遇到了各种合作交流上的问题,多数都是之前没有想到的. 比如邮件沟通的长延时和表述能力不佳,多人协作代码的整合等. 最终我们采用了敏捷开发的流程,项目进度大大加快.

不过幸运的是组员的个人能力都非常强劲,每个人可以完成完整的一套功能. 而且由于之前互相比较了解,每个组员的专长也很清晰,在安排分工时能够让每人处理其擅长的部分.这都给本次项目的顺利开发奠定了基础.

感谢能力逆天的开哥(贾开),感谢前端大神小开哥(赵一开),感谢机器学习强人 Tim(周昕宇),感谢网站达人听听(吴育昕),感谢设计师豆包(陈俊男). 正是开开听听组合加上设计师的给力才能让本次项目顺利进行.

通过本次课程,自己对较大的软件项目开发的流程有了深刻的理解,对软件开发中的非技术因素有了一些认识. 相信自己的队员对此也有些心得与体会.

本次课程开放 python 做为可选语言之一出乎我们的意料,也使得我们的开发心情比较愉悦,希望能够继续保持并最好能够提供更多选择. 另外,助教做为 reporter 加入软工项目组的意义似乎不是非常清晰,应该通过项目网站在线检查的一些信息依旧要求提交文档(指项目管理文档). 如果之后的课程能够在这方面加以改进相信将会更加优秀.

最后,感谢赵华凯助教的耐心解答与白老师的传授与指点.