项目管理文档

成员： 谢志杰 1352975

计鹏玥 1352914

王思尧 1352896

谢明玥 1352937

徐锦程 1353012

一、项目简介

1、功能要求

项目要求实现一个client+server的通信项目。可以实现多个客户端与一个服务端的链接，即服务器可以向多个客户端 发送广播消息，客户端也可以向其他客户端发送消息。类似于聊天室的那种功能，实现信息能在多个客户端之间共享。

工作过程中，客户端循环可以不停输入向服务器发送消息，并且启一个线程，专门用来监听服务器端发来的消息并打印输出。服务器端启动时，启动一个监听何时需要向客户端发送消息的线程。每次接受客户端连接请求，都启一个线程进行处理，并且将客户端信息存放到公共集合中。当客户端发送消息时，服务器端将消息顺序存入队列中，当需要输出时，从队列中取出广播到各客户端处。

2、开发环境

Github作为项目版本控制工具

Java作为开发语言：JDK8

Junit作为测试工具：Junit4

二、项目计划

* 13.16~3.18 项目组在github上建立repository，初步完成分工，同时学习相关技术，例如socket通信等等。
* 3.19~3.23 各成员分别完成界面类、IO操作类、客户端、服务端的编写。此时程序已经具备多客户端与服务端通信的基本功能，部分要求还没有实现。发送接收消息尚基于命令行，界面类还未加入。
* 3.24~3.26 加入界面类，使得系统的输入输出基于界面。同时完善客户端与服务端的细节，实现需求规约里的所有要求。
* 3.27~3.29 使用Junit对系统进行测试，同时完成相关文档的撰写。

三、项目分工

本项目为小组项目，由我们小组五名成员共同合作完成，分工情况如下。

* 谢志杰：负责服务端的编写和测试以及项目的统筹安排，文档修改。
* 谢明月、计鹏玥：负责客户端的编写以及使用Junit完成客户端测试。
* 徐锦程：完成界面的设计以及相关java类的编写，以及撰写管理文档。
* 王思尧：完成文件IO相关操作类的编写，以及撰写复用文档、程序文档。

四、项目完成情况

* 3.16~3.18 项目组在github上建立repository，初步完成分工。
* 3.19~3.23谢志杰：Server端基本完成。
* 3.24~3.24 徐锦程：GUI完成，等待客户端完成后开始集成。
* 3.25~3.26谢明月、计鹏玥：客户端原型完成。
* 3.26~3.28谢明月、计鹏玥：客户端重构，漏洞修复。
* 3.26~3.28谢志杰：服务端重构，模块化。
* 3.28~3.30王思尧，徐锦程文档撰写基本完成。

五、项目总结

1、交付产品

代码：源代码、测试代码

文档：复用文档、程序文档、管理文档

2、总结

本项目开始与三月十六日，历时两周，经过小组五名成员的共同努力，我们最终按时完成了项目，实现了需求中的所有功能。总体而言，此项目的开发比较成功，但是还是存在着一些问题，造成这些问题的原因也是多方面的，例如，前期我们对需求的理解出现了偏差，导致在开发过程中绕了少许的弯路。再比如，由于对socket的理解不够深入，导致开发遇到了一点困难。不过，这些问题最终都得以解决。

这个项目也给我们带来了很多新的认识。在项目确立后，我们就需要做好项目开发计划，需求调研用时，开发用时，测试用时，实施用时，维护用时。在我们做好了计划后，我们要随时的跟踪计划任务的完成进度，从而使我们的项目进度掌控在我们的开发周期范围之内。此外，除了对总体进度的把握，项目组成员之间的有效沟通必不可少。好的沟通能够加快项目的进度，使我们尽量地去避免一些隐藏的问题，及时地发现问题，解决问题，从而按时或者提前完成项目的开发。

随着项目的开发完成，我们积累了更多的经验，也学习到很多新的知识。相信这些经验与技巧，都将成为我们日后工作学习或者项目开发中宝贵的财富。

3、改进方向

* 代码部分可以进行进一步优化，减少冗余代码，提高代码质量。
* 通过改进代码结构可以使项目的模块化程度进一步提高，符合软件复用的思想。