计算机科学与工程学院课程设计报告

**题目全称：** 简单文件服务器和客户端

**题目难度等级：** 5

**指导老师：**  徐扬  **职称：** 教授 博士生导师

**学生姓名：** 王克 **学号：** 2021080909008

**专业：**计算机科学与工程学院（网络空间安全学院） **班号（大类分流后班号）：** 2021080909

|  |  |  |  |
| --- | --- | --- | --- |
| **A计算机使用技能（100分）** | **B课程设计报告（100分）** | **C计算机专业技能（100分）** | **总分（A\*10%+B\*30%+C\*60%）** |
|  |  |  |  |

**备注：如参加答辩，请答辩老师给出C计算机专业技能的分数。请指导教师给出除计算机专业技能以外的其他分数。**

**如参加答辩，请答辩老师签字：**

**主要任务：**

实现一个简单文件服务器和客户端。设计一个简单文件协议，利用网络编程，实现一个客户端和服务端，支持文件上传下载，使用客户端实现对服务器内储存的文件的操作。

**详细功能描述：**

1、支持创建单层目录

2、支持一次上传单份文件

3、支持一次上传多个文件或文件夹

4，支持删除某个文件

5，支持删除某个目录

6，支持浏览文件和文件

**预期成果或目标：**

设计协议，完成客户端与服务端的网络连接，客户端对服务端发送指令，实现本地对网络上服务器内文件的上传下载和浏览。

**指导老师评语：**

**指导教师签字：**

摘 要

计算机技术的快速发展促进了各个行业的多方面发展，而网络技术的发展促进了各行各业的信息融合，这两门技术相辅相成。计算机技术中各种高级编程语言的发展，使得普通人也有机会模拟网络的传输过程，并且利用现有的工具直接利用网络编程实现自己想要的功能，编写自己的软件。在JAVA中，就有网络编程的相关类，程序设计人员可以利用该类设计自己的协议，搭建自己的服务器，开发出一套简单的文件管理系统，其他使用者可以使用相对应的客户端对服务端上的文件进行远程管理。同时，利用JAVA提供的GUI工具与多线程技术，编程人员可以实现多用户链接，并且为服务端和客户端设计比较优美的操作界面。

**关键词：**JAVA，网络编程，协议，GUI，多线程

目 录

[计算机科学与工程学院课程设计报告 1](#_Toc8004)

[摘 要 2](#_Toc24437)

[第一章 绪 论 1](#_Toc25942)

[1.1 服务端与客户端概述 1](#_Toc14538)

[1.2 网络概述 1](#_Toc9691)

[1.3 JAVA概述 1](#_Toc9292)

[第二章 系统分析 2](#_Toc9275)

[2.1 功能分析 2](#_Toc3612)

[2.1.1 基础操作 2](#_Toc16665)

[2.1.2 拓展操作 2](#_Toc3376)

[2.1.3 实现功能 2](#_Toc5366)

[2.2 类关系分析 2](#_Toc10233)

[2.2.1 类概述 2](#_Toc25758)

[2.2.2 工具类-MyStreamMethod与MyActionMethod 3](#_Toc12178)

[2.2.3 服务端与客户端类-MySever、MyClient 3](#_Toc26179)

[2.2.4 特殊实现类-MyRunnable、InputWindow 4](#_Toc27893)

[2.3网络协议分析 4](#_Toc17466)

[第三章 详细设计及实现 5](#_Toc31051)

[3.1场景与视窗 5](#_Toc29351)

[3.2 背景类 5](#_Toc8536)

[3.3 小球类 5](#_Toc22440)

[3.4 弹珠类 5](#_Toc30083)

[3.5 平板类 5](#_Toc444)

[3.6 生命值及关卡数显示 6](#_Toc29472)

[3.6游戏控制类 6](#_Toc18041)

[3.6.1各元素的协调 6](#_Toc22738)

[3.6.2键盘事件 6](#_Toc13257)

[3.6.3碰撞检测 6](#_Toc1865)

[.......... 6](#_Toc20558)

[3.6.4游戏状态的切换 6](#_Toc6310)

[...... 6](#_Toc19350)

[第四章 测试 6](#_Toc27723)

[4.1 游戏应用环境的构建 6](#_Toc1354)

[4.1.1 游戏需要的硬件环境 6](#_Toc4581)

[... 6](#_Toc19071)

[4.1.2 游戏需要的软件环境 6](#_Toc5605)

[........ 6](#_Toc25287)

[4.2 游戏界面显示 6](#_Toc723)

[4.3 平板移动测试 7](#_Toc4806)

[........ 7](#_Toc17945)

[4.4 小球碰撞测试 7](#_Toc10312)

[........ 7](#_Toc23605)

[4.5 游戏失败测试 7](#_Toc1070)

[......... 7](#_Toc1912)

[4.6 游戏通关测试 7](#_Toc30895)

[........ 7](#_Toc11703)

[4.7 小结 7](#_Toc18117)

[参考文献 8](#_Toc7327)

第一章 绪 论

1.1 服务端与客户端概述

客户端（Client）是发送请求（request），服务器端（Service）是响应请求（response），返回相应的资源数据。服务端是为客户端服务的，服务的内容诸如向客户端提供资源，保存客户端数据。

客户端或称为用户端，是指与服务器相对应，为客户提供本地服务的程序。除了一些只在本地运行的应用程序之外，一般安装在普通的客户机上，需要与服务端互相配合运行[1]。

客户端与服务端相互联系，相互配合，完成服务端能提供和客户端想实现的功能。客户端与服务端一般通过网络发送指令完成功能。

1.2 网络概述

计算机网络是指将不同地理位置，具有独立功能的多台计算机及网络设备通过通信线路(包括传输介质和网络设备连接起来)，在网络操作系统、网络管理软件及网络通信协议的共同管理和协调下实现资源共享和信息传递的计算机系统。[2]

通过网络，用户可以跨地区传达信息，大大方便了人类的生活，提高了信息传达效率，实现了天涯若比邻的实时传输。

网络通信协议是根据网络上的节点进行通信的一组规则，每种设备都可以根据通信协议识别其他设备的信息。使用相同协议的加算计可以识别彼此之间传递的信息。[3]方便了用户与多个用户之间信息传递，是网络中的一门通用语言。

1.3 JAVA概述

Java是是Sun公司推出的能够跨越多平台的、可移植性最高的一种面向对象的编程语言。通过JAVA，程序员可以写出更利于人类阅读的代码，并将其编译为机器语言通过计算机运行，代替人进行工作。

JAVA编程语言发展迅速，技术人员编写了许多功能强大且实用的库，供后来的学习者学习使用。

第二章 系统分析

2.1 功能分析

2.1.1基础操作

1.客户端与服务端窗口实现及按钮监听实现

2.客户端输入服务端IP地址

3.客户端与服务端通过网络连接

4.进行操作后双端进行日志打印

5.双端对本地文件进行操作

6.双端通过网络信道发送接收byte流

2.1.2拓展操作

1.协议实现，规定byte流格式，长度及信息

2.服务端通过网络传输byte流后分析操作指令

3.服务端通过指令进行操作并发回反馈

2.1.3实现功能

1.在客户端远程创建单层目录

2.在客户端远程上传单份文件

3.在客户端远程上传多个文件或文件夹

4.支持删除某个文件或文件夹

5.在客户端浏览服务端文件及文件夹

2.2 类关系分析

2.2.1 类概述

本设计的软件实现分为客户端与服务端，而两个软件的实现中有许多共通之处，因此笔者在将软件功能实现后，将两个软件中使用的相同功能以及复杂功能分化出来，分类放到其他源文件中，以此来减少单个文件的代码行数，也降低了代码阅读的难度和时间。下图是有关本软件内部类的关系图：

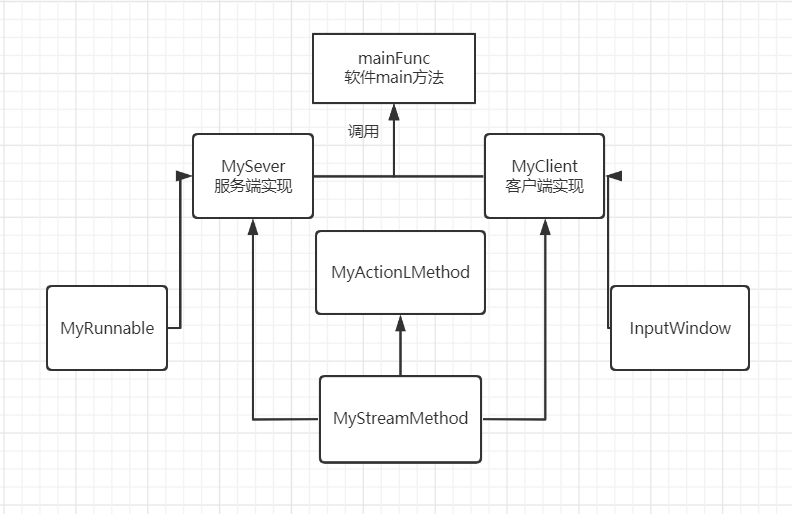


图2.2.1

2.2.2 工具类-MyStreamMethod与MyActionMethod

在软件代码实现中，工具类主要分为两类，一类是对本地文件及网络进行操作的代码，主要包括对文件的遍历、读取、删除、复制，以及利用网络信道发送及接收byte流。这一类方法的本质都是利用JAVA中的流进行操作,因此将这一类方法归到同一工具类中。另一类则是两个软件中对按钮监听的方法实现，这一类代码的特征是类型多，占用的代码量比较大，因此将其单独归类，阅读者只需要知道方法的用途，分析时再去寻找对应的源码即可。

MyActionMethod类中的方法，大多数在客户端MyClient类中被调用，也调用了MyStreamMethod中的方法，因此MyStreamMethod在整个软件中为最底层实现。

2.2.3 服务端与客户端类-MySever、MyClient

这两个类都继承于JFrame类，实现了ActionListener接口方法。

在这两个类中，主要包括了初始化方法，在初始化方法中，为窗口设置大小参数，对按钮进行排列，对按钮实现监听功能，监听的具体实现被分到MyActionMethod工具类中，监听到按钮操作后调用对应函数实行功能。

在客户端初始化之前，需要弹出小窗口，让用户输入服务器IP地址，此软件因在本地，未获得公网静态IP，因此避免出现更换网络后无法连接的情况，需要客户获取服务器IP再进行连接。

初始化之后，客户端申请与服务端连接，服务端等待申请连接并阻塞直到接收下一个申请。之后等待用户在客户端进行操作。

2.2.4 特殊实现类-MyRunnable、InputWindow

这两个类都只调用过一次，分别在服务端与客户端。

MyRunnable类继承Runnable接口，实现run（）方法，是服务端的具体功能实现方法，当有一个新的客户端进行连接，便在服务端的线程池中加入一个新的MyRunnable类等待对应客户操作。其阻塞等待接收客户端发出的byte流，根据byte流的信息对本地文件进行操作，以此实现客户端对服务端内文件的操作。

InputWindow类在客户端初始化时调用，弹出窗口，接收用户输入的IP地址，传递到客户端，以便客户端进行连接。

2.3网络协议简述

在本软件实现中，需要频繁的在两端之间传递byte流，每次传递的信息都不尽相同，因此设计了一个简易协议帮助软件识别需要进行的操作，调用不同的方法。此简易协议规定了每次发送byte流的长度以及格式，在这里不仔细分析，具体讲述在第三章。

第三章 详细设计及实现

3.1视窗与布局

MySever类、MyClient类与InputWindow类三个类继承自JFrame类，内部设置对应的蚕食，添加了对话框按钮，增加监听。用户可以在对应的部位添加内容，供软件运行。

3.2 双端初始化与连接

客户端和服务端在进行连接时使用的是TCP协议，其是一种可靠的面向连接的传送服务，具体实现方式这里不做赘述，只需知道客户端与服务端是通过这种协议在网络中建立连接。

初始化时服务端新初始化一个监听本地IP，端口为5050的SocketSever，即TCP协议的服务端端口，等待服务端的连接，使用阻塞方法SocketSever.accept()方法，在未连接时一直等待。并在此时显示窗口，展示本机IP地址。

客户端初始化时先弹出客户端窗口，并初始化InputWindow类，弹出小窗口用于获取用户输入的IP地址。输入IP之后客户端会创建连接该IP的Socket，连接服务端。双端成功连接之后都会在窗口中打印成功连接的对话。

服务端在接收到一个请求之后，会将会话加入线程池，等待下一个连接，只需要开启一次服务端，客户端可随时开启连接，并且连接多次，但暂时不讨论多用户同时操作服务端内文件的冲突问题。

3.3简单文件传输协议实现

在用户连接之后，客户端和服务端都进入静默状态，等待用户在客户端进行操作。客户端监听到按钮按下之后，需要服务端进行相对应的操作，服务端发送完成反馈。这些指令都蕴藏在双方发送的byte流中。

协议规定双方发送的单次byte流长度都为128。其中在操作指令中，首位即第一位是常驻操作数，其代表本次服务器需要进行的操作。服务端收到请求后会进行对应操作，在操作结束后返回一个128byte的数组，代表本次操作成功或失败。在进行操作过程中，可能会遇到一些情况需要用户判断，会需要第二位也作为操作位，这些指令会在之后进行说明。

有些操作需要为服务端操作提供信息，其信息储存在操作数之后的位数，例如文件地址，文件名称等。这些信息占用的位数不能确定，无法详细规定多少位可以包含所有内容，但协议中所有有效信息全部包含在前127比特。

在一个byte串中加入操作数和信息位后，剩余部分会补齐，协议使用‘@’符号进行填充，并且为了避免信息混淆，最后几位不会填充符号而是将该byte串无效位的位数储存进去。例如有125位有效信息，则填充2位‘@’和一个数字3。除此之外，协议规定如果前127位都为有效位，则最后一位为‘@’，不需要填充数字1。

非操作指令中，舍弃了前面的操作数位，127比特全是有效信息位，后面一位填充‘@’，如果遇到不满127位的，按照前文所述的填充方法填充。使非操作指令和操作指令具有相似的格式，能够用同一种方法操作传送的byte流。

3.4 流式工具类实现-MyStreamMethod

MyStreamMethod工具类是整个软件最基础的类，其他几个重要的类中都应用到了该类中写的方法，因此首先说明该类有哪些方法，具体怎么实现。

该类定义了两种send()方法和两个receive()方法，相同名称的方法对应的参数和作用都不相同；还定义了获取本地文件夹内所有文件的方法、在对话框中添加文字的方法和弹出选择文件地址窗口的类。

send和receive方法分别对应发送和接收单个byte串和整个文件或者文件夹。

3.5 平板类

平板类继承于QGraphicsItem，需要重写三个虚函数。平板的绘制类似于小球，直接插入图片到画刷画出图形。因为要进行与小球的碰撞检测，所以shape函数要返回精确轮廓。

.......

3.6 生命值及关卡数显示

.......

3.6游戏控制类

.......

3.6.1各元素的协调

......

3.6.2键盘事件

......

3.6.3碰撞检测

..........

3.6.4游戏状态的切换

......

第四章 测试

4.1 游戏应用环境的构建

4.1.1游戏需要的硬件环境

...

4.1.2 游戏需要的软件环境

........

4.2 游戏界面显示

......

4.3 平板移动测试

........

4.4 小球碰撞测试

........

4.5 游戏失败测试

.........

4.6 游戏通关测试

........

4.7 小结

.....

参考文献

1. CSDN.服务端和客户端的区别及介绍.

https://blog.csdn.net/manerzi/article/details/127599455

1. 知乎.计算机网络概述. https://zhuanlan.zhihu.com/p/52244814.
2. 杨连刚.Qt开发的国内外知名软件[EB/OL].2018-04-19.
3. wiki百科.Qt for Python Signals and Slots. https://wiki.qt.io/Qt\_for\_Python\_Signals\_and\_Slots.
4. 邱升.求点关于直线坐标的一种简捷方法[J].中学数学, 2000.