|  |
| --- |
| **成绩** |
|  |



中国农业大学

课程论文

（2019 -2020学年夏季学期）

**论文题目：** 应用开发实训报告

**课程名称：** 农业信息化应用开发实训I

**任课教师：** 雷宏洲

**班　　级：** 试验192

**学　　号：** 2019308160121

**姓　　名：** 朱婷婷

**课程论文格式要求（封皮的背面）：**

1．课程论文采用统一封面，以左侧为准装订成册。

2．课程论文一律使用标准A4复印纸打印或使用标准A4复印纸手写稿形式上交。

3．论文打印的格式要求：

论文标题（使用隶书二号加黑；一级标题、二级标题、三级标题分别使用宋体三号、四号及小四号并加加黑）

摘要、关键字（需使用宋体小四号）

正文（使用宋体小四号，行距20磅）

参考文献（使用宋体五号）

**一．基础概念题**

（1）根据题意，需要使用java.awt.Graphics类为绘图对象，绘图工具paint()通过Grapgics对象绘制具体的图形。由所给类的关系加上个人理解，可以通过继承绘制不同形状图形，减少重复操作。由基本图形出发可划分各个子类，完成从宽泛到细节的图形描述。继承关系如图1-1所示。

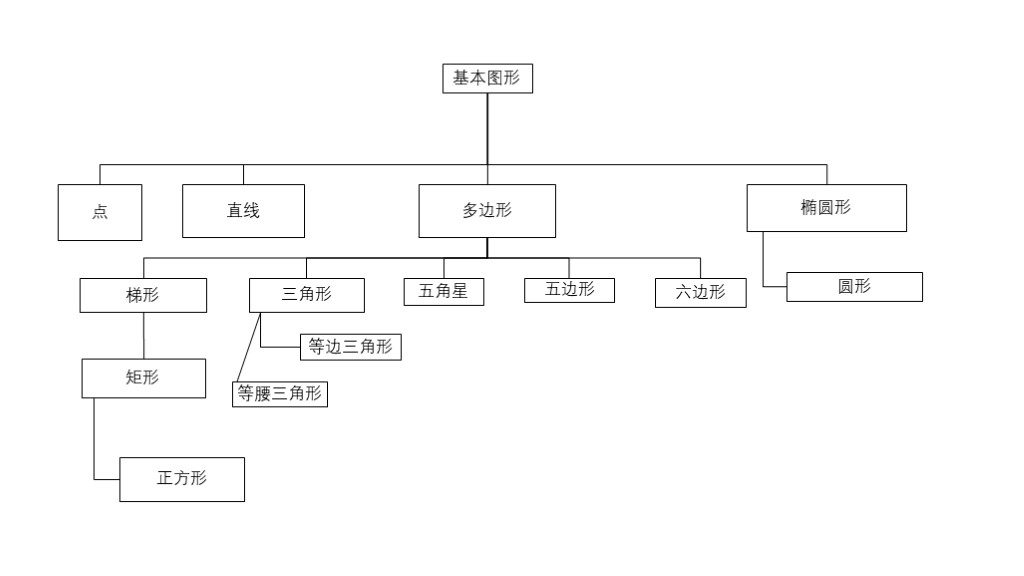


图 1-1 继承关系

代码实现时，基本图形与多边形的范围较广，设置为抽象类，其中包含draw(Graphics g)通过继承完成所有图形的绘制。在绘制过程中，简单的图形如三角形、圆形、矩形能通过直接设置坐标参数完成；而对于五角星、五边形、六边形，很难直接找到对应顶点坐标并依次连接，并对绘制结果在同一水平线增加了难度，因此需要涉及到外接圆半径和角度的关系，通过数学关系设置参数中心坐标(x, y)和半径r，经过计算获取所有顶点坐标（如图1-2），即可绘制所有边长的直线，正六边形和五角星可类推。

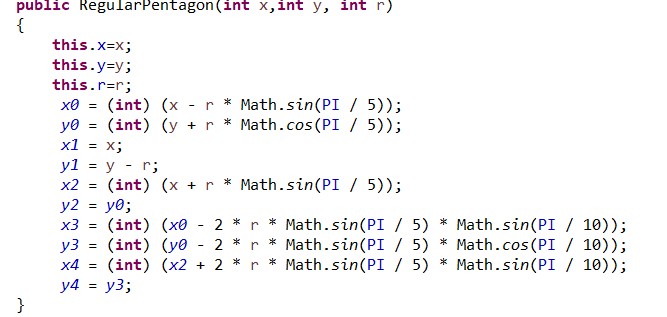


图 1-2 正五边形数学关系

对图片轮廓完成绘制后，为达到题中所示的颜色，先设置RGB为70,130,180的颜色，再对三个图形进行填充，如图1-3。

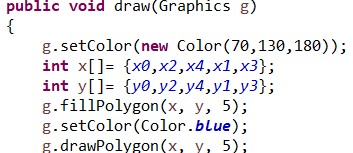


图 1-3 颜色填充

对于题目要求两种实例化方式，一种使用了new完成实例化，另一种则在每个类中加入复制构造函数并以原类返回，通过主函数已完成实例化的对象复制得到新对象，如图1-4。

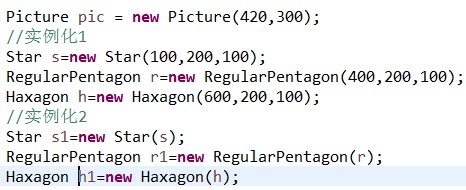


图 1-4 两种实例化方式

在文件1中运行Mypicture.java，得到绘制图形如图1-5。

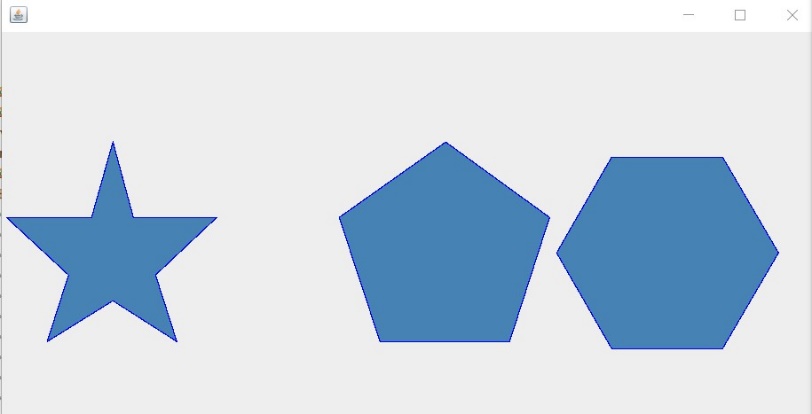


图 1-5 图形

（2）事件处理机制三要素有：事件源（能够产生事件的组件）、事件（用户对组件的操作）和事件监听器（接受、解析并处理事件的对象）。当组件的特定动作（如单机组件、移动鼠标或者敲击键盘）发生时，组件产生ActionEvent事件并传递给ActionListener对象，再由ActionListener定义的方法相应处理事件，实现和读者的交互。

在文件2中，以选择攻击为例，事件源、事件和事件处理分别为：

事件源：一技能按钮、二技能按钮、三技能按钮；

事件：点击一技能、点击二技能、点击三技能；

事件处理：文字提示“对方速度减慢”、文字提示“对方眩晕”、文字提示“对方受到10000点伤害”。

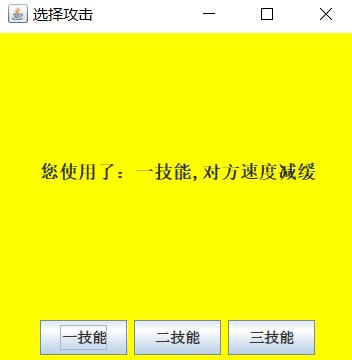
当用户点击不同的按钮时，组件会获取相应产生事件并传递给监听器，监听器通过getsource()函数获取事件源，再由规定的方法，输出不同的信息提示，如图2-1。



图 2-1 设置文字提示

运行Event.java,以点击一技能按钮为例，显示结果如图2-2。

图 2-2 点击一技能按钮



（3）利用Socket方式，首先服务器端创建ServerSocket，设置建立连接的端口号，客户机端创建客户端Socket对象，设置绑定的主机名称或IP地址，指定连接端口号并发起连接请求；建立连接后服务器端和客户端建立数据输入流和数据输出流，并利用其进行数据传输；最后关闭Socket和ServerSocket，完成C/S模式的连接请求和响应。

举例：远程数据通信，完成服务器和客户端的信息传输。

在文件3中，服务器端建立的连接端口号为8888，与客户端建立连接后，通过数据输入和输出流传输数据，以TestServer.java中部分代码为例（图3-1），服务器向线路发送“你好，我是服务器”，并将在线路中读取的信息在系统中输出。



图 3-1 服务器端主要代码

依次运行服务器端和客户端后，运行结果如图3-2、3-3，完成简易的C/S模式程序。



图 3-2 客户端运行结果

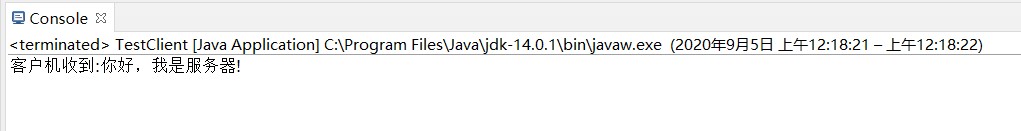


图 3-3 服务器端运行结果

（4）对JSP程序解释如下：

1. <%@ page contentType="text/html; utf-8"%><!--定义页面的依赖属性，使页面显示为中文。contentType用于定义网络文件的类型和网页的编码，告诉客户端实际返回内容的内容属性。-->

2. <&@ page import=" java.util.\*"%><!--导入util包下的类（日期）-->

3. <html>

4. <body>

5. <center><!--对包括的内容水平居中-->

6. <%! String name; %><!--声明string类的成员变量name-->

7. <%

8. Date today=new Date();<!--获取当前日期并将当前日期保存为其初始值-->

9. name=request.getParameter("UserName");<!--获取用户姓名并以字符串形式返回赋值给name，因URL中没有name属性，返回值为null。-->

10. %>

11. Welcome<%=name%>,今天是<%=today.getDate()%>,星期<%=today.getDay()%><!--输出语句，today.getDate()用于获取当前日，today.getDay()用于获取当前星期-->

12. </center>

13. </body>

14. </html>

运行结果如图4：

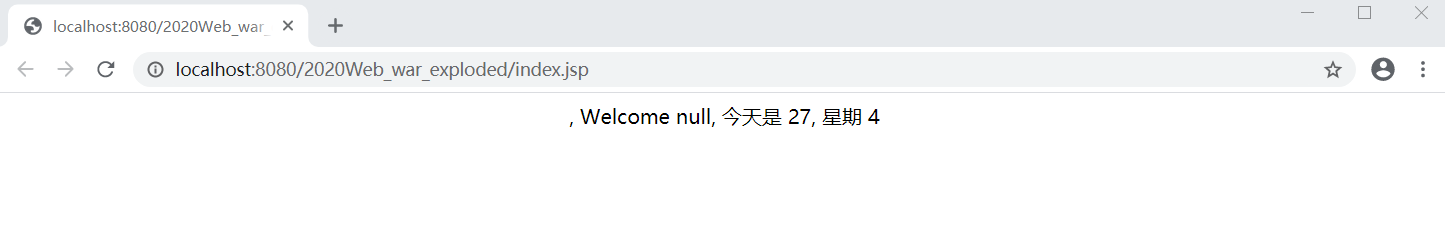


图 4 运行结果

**二．实践题**

**1.项目描述：题目名称及背景概况。**

题目背景：基于java的小型宿舍文件仓储系统

背景概况: 如今的数据规模高速增长，原有的人工管理文件方式处理这样大的数据不可避免增加了管理的工作量及复杂程度，伴随着不严密性带来各种工作中的疏漏。为了解决这样的问题，以各类网盘为代表的应用软件实现了人们文件分享的需求，使得文件能在群体间进行各种管理和操作，极大提高了工作效率。目前，各类软件系统仍在不断优化的过程中，满足人们多样化的需求。

**2.项目分析：项目的目的、用途或需要解决的问题以及需求。**

项目目的：开发一个小型宿舍文件仓储系统，使同宿舍成员能够完成文件的分享和传输。

项目用途/需要解决的问题：用于小群体间上传、下载、重命名及删除文件等操作，专门储存学习资料，实现对学习资料的储存和共享。

需求：对用户进行密码验证，登录后实现小群体间的数据传输，更具有针对性，简洁易操作，保护隐私和数据安全。

**3.设计：功能设计、界面设计、数据结构和数据库设计、对象和类设计、文件系统目录结构设计。**

（1）功能设计

登录：实现注册、修改密码

文件管理：查看文件储存目录，上传、下载、删除文件，修改文件名、客户端权限管理

（2）界面设计

登录界面：

先运行“server.java”，再运行“IP.java”，显示输入IP地址界面（图5-1）：



图 5-1 输入IP地址

输入服务器提供的IP后用户进行登录（图5-2）：密码框隐藏，点击显示密码后取消隐藏。

注册和修改密码界面（图5-3、5-4）：



图 5-3 注册界面

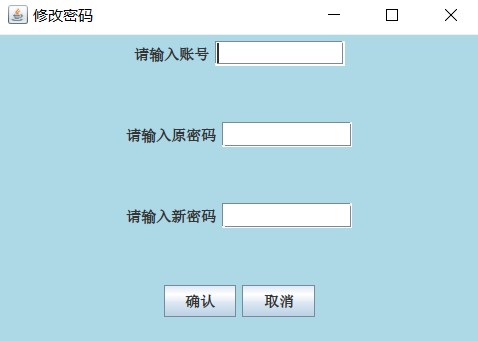


图 5-4 修改密码界面

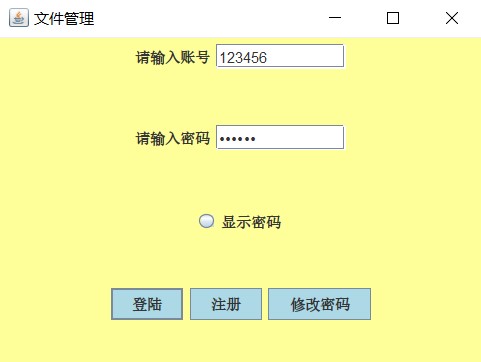


图 5-2 用户登录界面

客户端界面：登录成功后，客户端上方显示个人已上传的文件，下方点击“目录”按钮在显示共享文件，如图5-5。

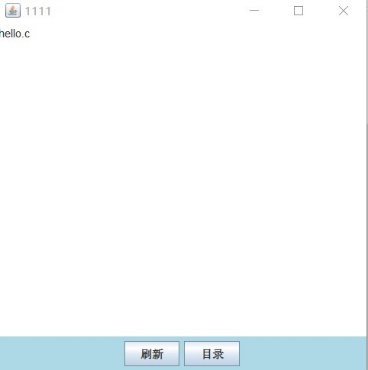


图 5-5 客户端界面

服务器端界面：运行Server后，服务器窗口可显示IP地址和用户的操作，并设置用户下载及上传权限，如图5-6。



图 5-6 服务器端界面

（3）数据结构和数据库设计：

代码中涉及到数据结构的部分只有FileInfo类，考虑到文件的存储目录是一种类似多叉树的结构，该类使用两个HashMap，即SubDirectory和Parent，构造了多叉树的结构，且可以快速的查找文件的parent和child，以及获得指定文件的路径。

（4）对象和类设计：

完成此系统共设计了九个类，其中包含：

JFrame类LoginWindow（登录界面）、registerWindow（注册界面）、changePasswordWindow（修改密码界面）和PromptWindow（提示窗口）以文本框和按钮为主，通过GUI设计和事件监听完成用户登录和信息提示；

服务器地址类IP实现不同机客户端与同一服务器连接，适合同宿舍群体间使用；

为储存客户的信息及文件信息，ClientInfo和FileInfo分别实现对用户名与密码、文件名及路径的储存，便于服务器和客户端的传输。其中，ClientInfo类主要含有成员变量用户密码、账号以及在线状态；FileInfo类将个人文件与共享文件区分，使文件仅能由上传者重命名和删除，共享文件接受所有客户进行下载。

最后，作为一个C/S系统，Client和Server实现了系统与读者的交互，完成整个功能操作。

（5）文件系统目录结构设计：

点击客户端的“目录”按钮后，通过“root”窗口可实现对文件的管理，如图5-6。初始化界面有“上传”、“新建文件夹”两个按钮，其中点击新建文件夹可作为嵌套，依旧含有这两个按钮。当用户点击上传时可上传文件，目录中会弹出相应文件按钮，点击文件即可实现下载、重命名和删除的操作。

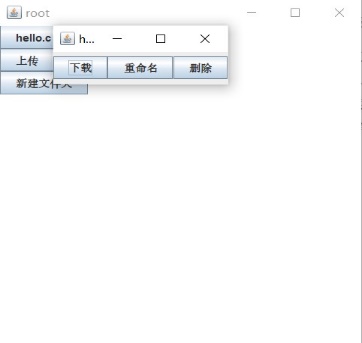
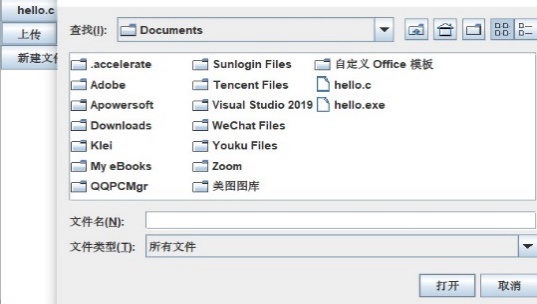


图 5-6 文件系统目录结构



**4.主要功能或算法描述及流程。**

（1）FileInfo类：该类主要存有三个HashMap，分别为SubDirectory（文件夹对应其所包含的文件或文件夹名），Parent（文件或文件夹对应其所在文件夹），以及personalFile（用户对应其所上传的文件），通过三个HashMap，即可保存用户操作相关的文件信息。其中Parent主要负责记录一个文件或者文件夹的路径，通过反向查找，即可输出文件的全路径。

（2）客户端获得文件存储目录：当用户通过登录界面进入客户端界面后，服务器会通过ObjectOutputSream将自己的FileInfo类传给客户端，从而客户端可以获得文件存储目录。

（3）服务器和客户端更新文件目录：当客户端对文件进行修改操作时，会先将修改信息传输给服务器（不同的操作对应不同的操作符），服务器收到信息后，首先修改自身的FileInfo信息，之后再将信息传输给所有客户端，以此来实现服务器和所有客户端的信息同步。

（4）客户端上传文件：客户端需要上传文件时，首先发送申请信息给服务器，服务器返回对应信息后，开始接收文件，客户端收到反馈后开始传输文件，传输结束后再把相关的信息添加到FileInfo。

（5）客户端下载文件：与上传文件相似，都采用先跟服务器确认，然后进行对应操作，所下载的文件将保存在文件夹“Download”中。

（6）socket接收消息的主要结构：服务器端，主线程不断接收新的socket连接，之后进行登陆，注册，修改密码的验证。子线程为很多个内部类组成，该内部类存有对应socket，接收除上述操作外的所有信息。客户端只有一个接收socket信息的入口，该入口接收服务器传来的不同标识的字符串，之后做出对应的操作。

相关类的uml图（图5-7），见文件“源程序-实践-UML图”：

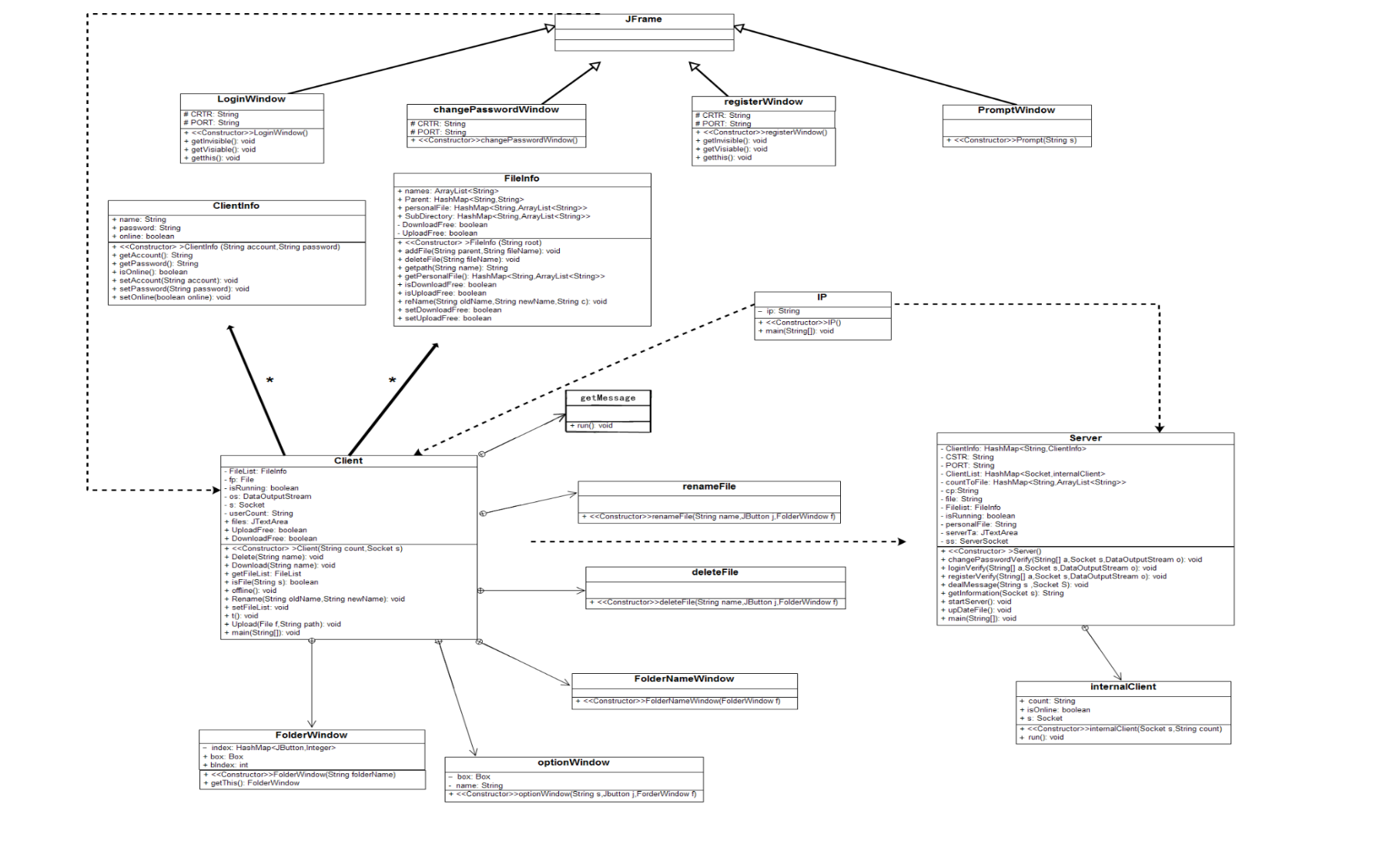


图 5-7 UML图

相关流程图（图5-8）：

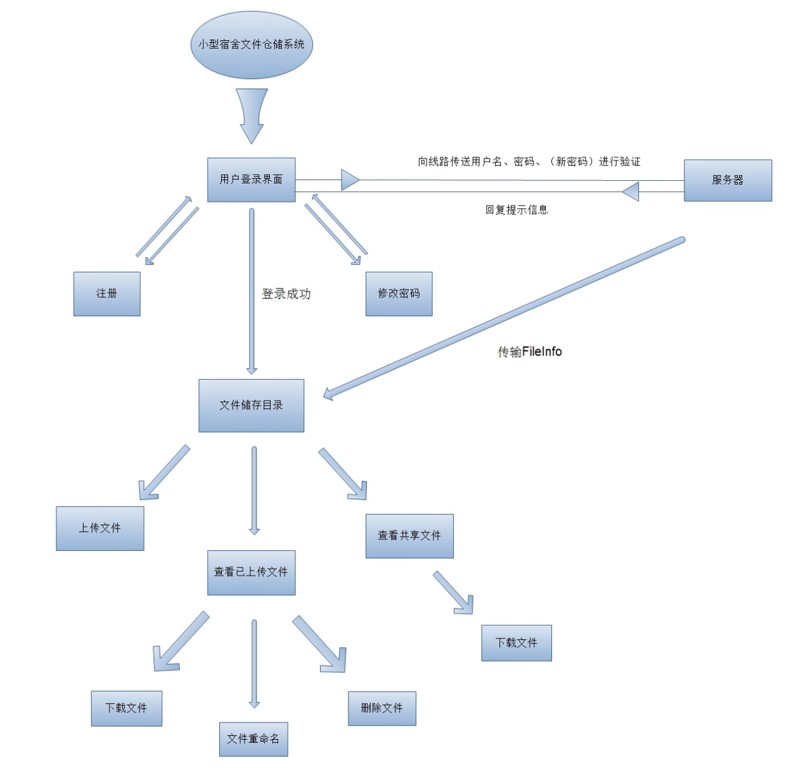


图 5-8 系统流程图

5. 文献资料

【1】汪钰斌. 学生信息管理系统[J]. 消费电子, 2013(16):63-63.

【2】邵维忠, 杨芙清. 面向对象的系统分析[M]. 清华大学出版社, 2006.

【3】王枫, 罗家融. Linux下多线程Socket通讯的研究与应用[J]. 计算机工程与应用, 2004, 040(016):106-109.

【4】綦宏伟, 代亚非, 李晓明. 基于Java/Swing的通用文件管理器设计模式[J]. 计算机工程与应用, 2003(08):108-111.

【5】肖玉朝, 何伟. Java中实现事件监听的多种方法剖析[J]. 电脑知识与技术:学术交流, 2009(11):2898-2900.