# 仿AirDrop文件传输及共享剪切板

## 一、软件是什么

### 1.1软件功能简介

该软件是基于Web页面实现的（由于各种限制，只支持Google浏览器）。主要的功能有：文件传输、共享剪切板。

文件传输功能实现流程：当手机和电脑处于同一个局域网下时，手机（电脑）访问发布出来的Web页面的网址，点击上传文件即可将手机（电脑）中的文件上传到服务器中，上传的文件也可以通过访问下载页面的网址，点击下载。

共享剪切板的实现流程：当手机和电脑处于同一个局域网下时手机（电脑）可以通过表单将需要复制的内容提交到服务器端储存，在通过客户端读取服务器端储存的内容，实现手机和电脑之间的数据同步，接着通过clipboard API实现将传来的内容写入剪切板实现复制粘贴功能。

### 1.2界面展示

软件界面如下：

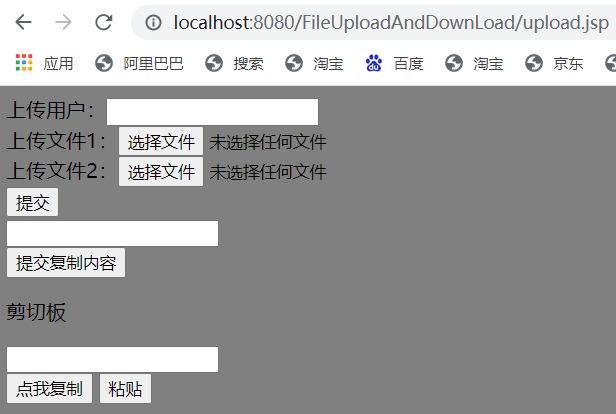


图1 PC端文件传输及剪切板页面

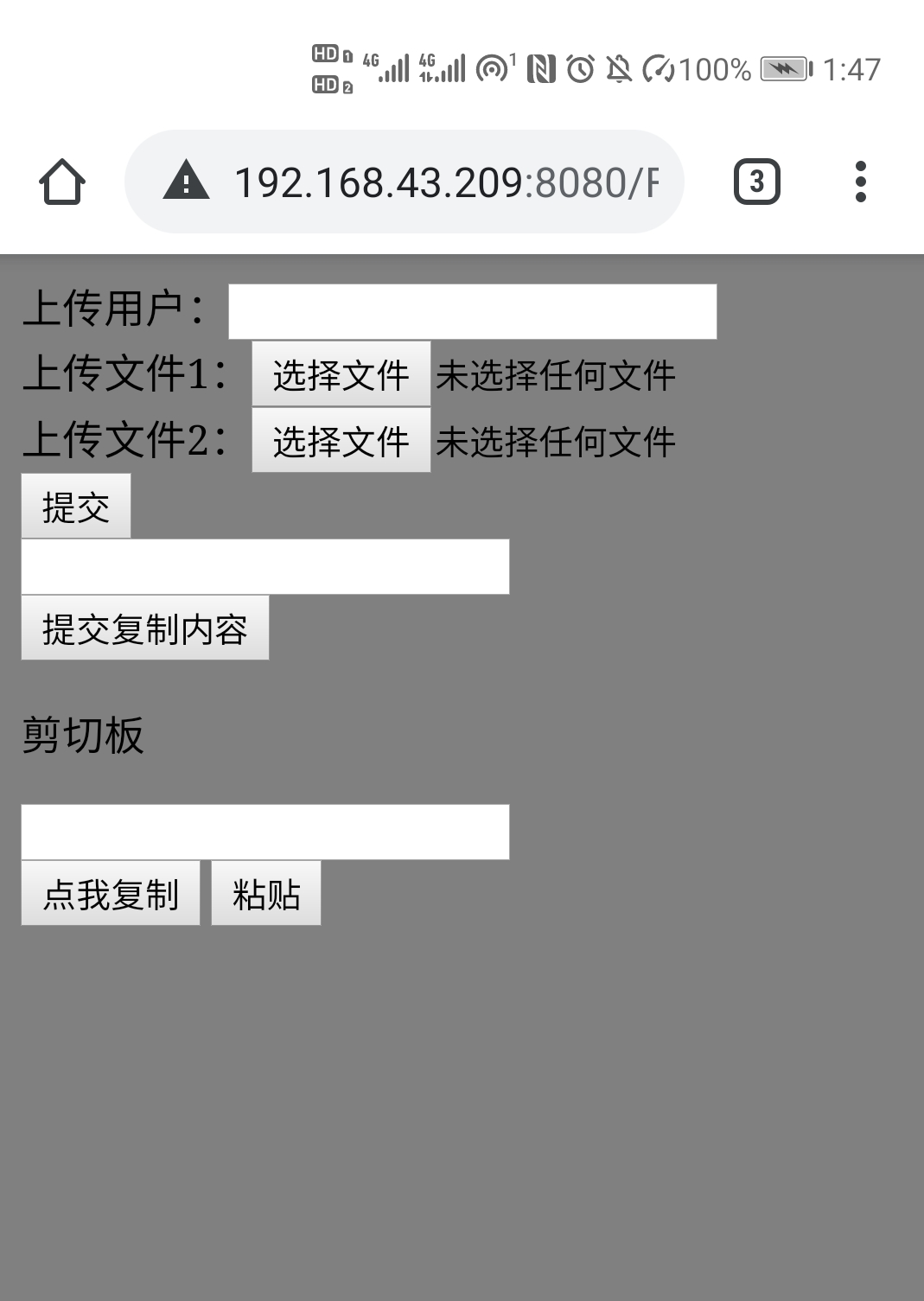


图2 手机端文件传输及剪切板页面

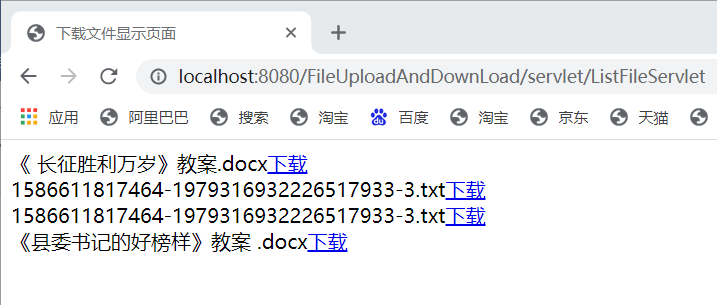


图3 PC端文件下载页面

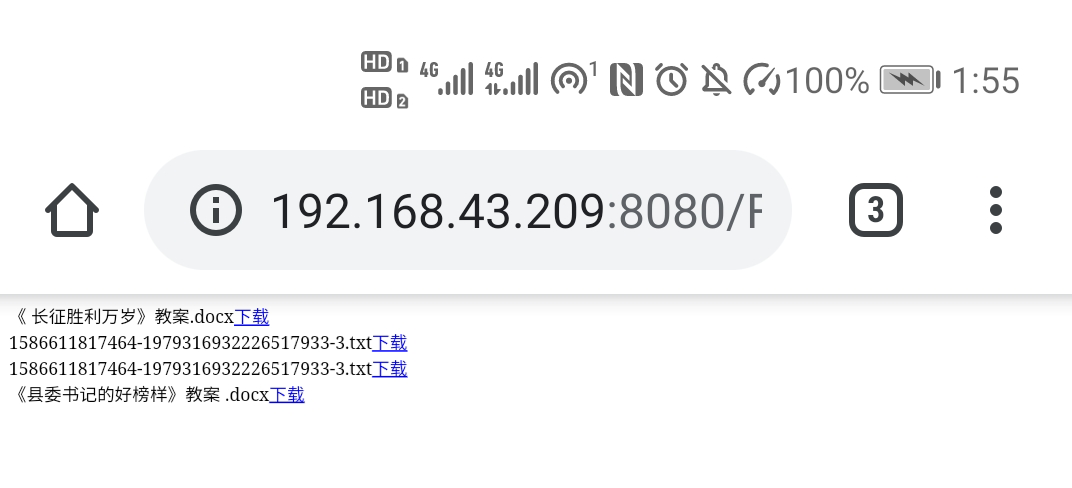


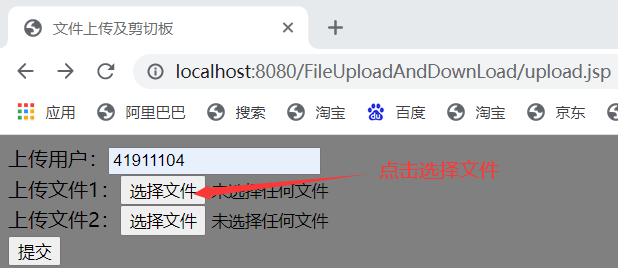
图4 手机端文件下载页面

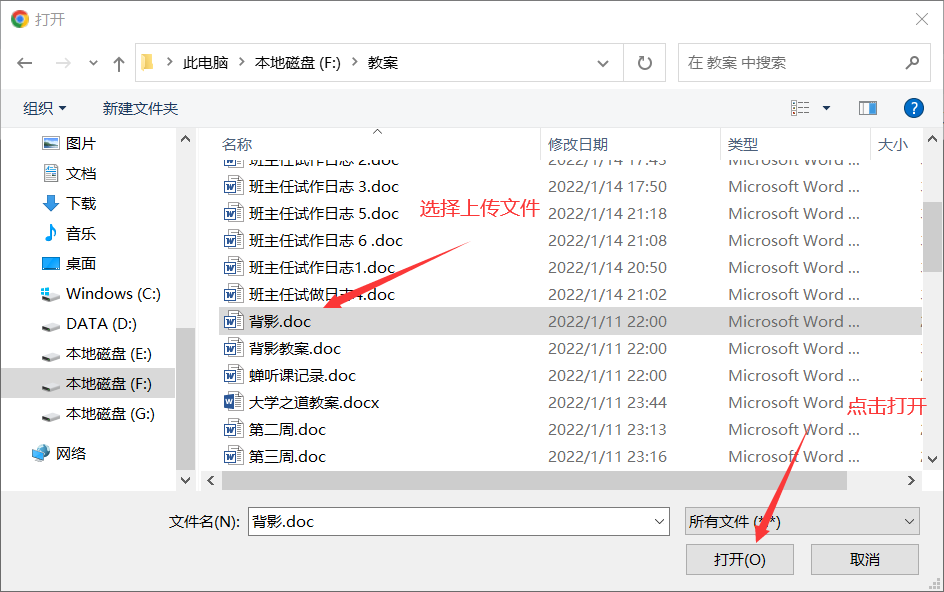
## 二、使用方式

### 2.1文件传输及下载

#### 2.1.1 文件传输

填入用户名，点击上传文件，从手机（PC端）已有的文件中选择需要上传的文件，点击提交按钮即可上传。由于设置了单个上传文件的大小的最大值，目前是设置为1024\*1024字节，因此传输的单个文件最大为1MB。PC端流程如下(手机端的流程与之相似)：





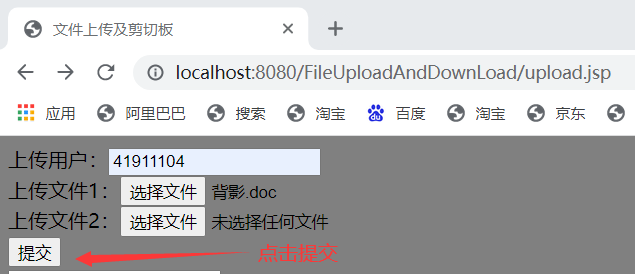


图5 文件传输流程

#### 2.1.2文件下载

在手机（电脑）谷歌浏览器的网址栏中输入下载页面的网址，即会出现下载的网页，显示出所有上传到服务器的文件，选择所需要的文件点击下载，即可下载到本地。

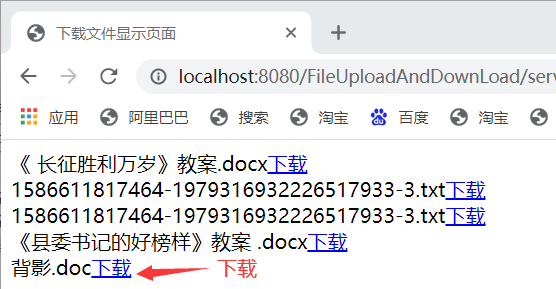


图6 下载文件流程图

### 2.2共享剪切板

#### 2.2.1 pc端剪切板

单独在电脑可实现的剪切板功能：功能一：在表单填入需要复制的内容，点击“提交复制内容”，将要复制的内容传到下方剪切板中（也可直接在剪切板的文本框中写入要复制的内容，该功能主要是为了实现共享剪切板），点击“点我复制”，即可将复制内容存入剪切板，点击“粘贴”实现在本网页上显示复制内容（流程见图7）。功能二：能获得在电脑其他页面复制的内容在本页面粘贴，只需点击粘贴，即可将其它页面复制的内容粘贴在页面上。

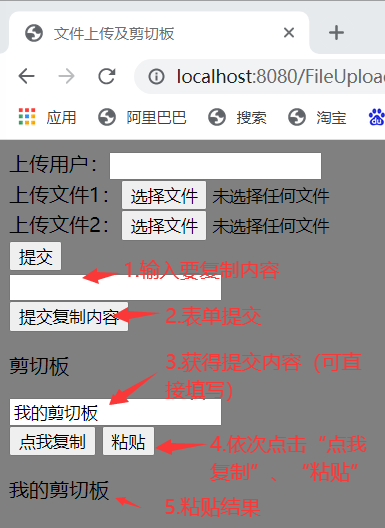
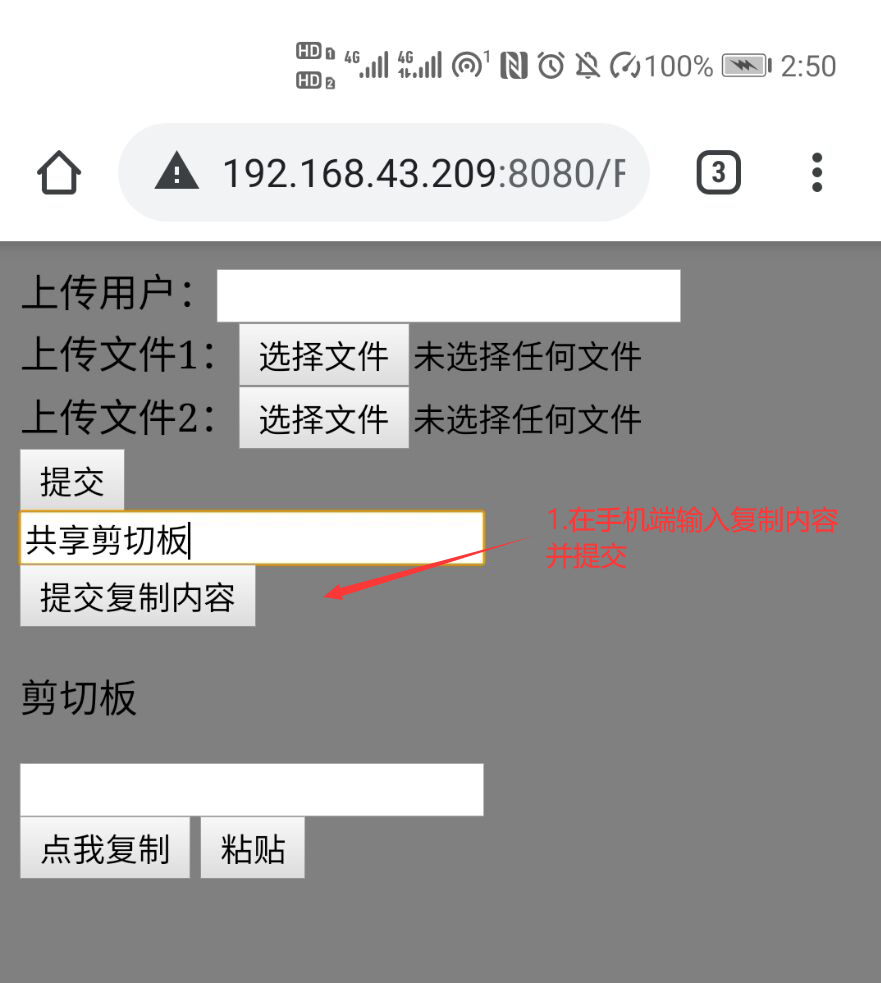


图7 PC端剪切板使用流程图

#### 2.2.2共享剪切板

共享剪切板可以实现手机端（PC端）提交复制内容PC端（手机端）可以得到它并进行复制粘贴。具体的实现手机端提交复制内容PC端粘贴的使用流程如下（PC端传手机端操作类似）：首先在手机端打开软件网页，按照图7流程图中的步骤一、二输入复制内容、提交表单数据；接着再PC端按照图7流程图的步骤四、五将上传内容复制到剪切板中进行粘贴。



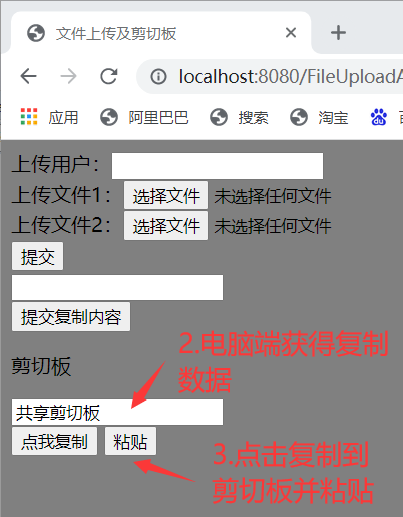


图8 共享剪切板实现流程

## 三、关键代码

### 3.1 网页端代码upload.jsp

upload.jsp是实现这个软件主要前端界面的代码。主要用来让用户上传文件、实现表单提交及剪切板。



图9 upload.jsp代码截图

### 3.2UploadHandleServlet.java

该servlet实现从客户端获得得到要上传文件的保存目录，将上传的文件存放于WEB-INF目录下，不允许外界直接访问。同时将限制传输文件的单个大小限制为1MB；通过文件的扩展名来判断上传的文件类型是否合法；为防止文件覆盖的现象发生，通过文件名的hash值为文件分配内容地址、为上传文件产生一个唯一的文件名。

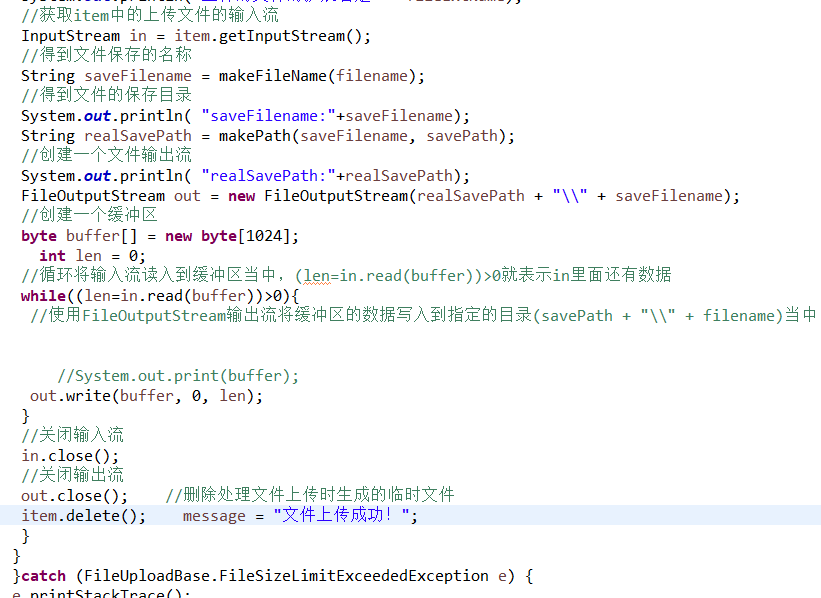


图10 UploadHandleServlet.java核心上传代码

### 3.3 ListFileServlet.java

该servlet主要实现将上传到服务器的文件及其下载的链接发送到客户端，使用户能够点击下载上传的文件。



图11ListFileServlet.java核心上传代码

### 3.4clip.js

该js通过clipboard API实现实时监听剪切板内容，进行复制粘贴操作。代码如下：



图12 clip.js

## 四、单元测试

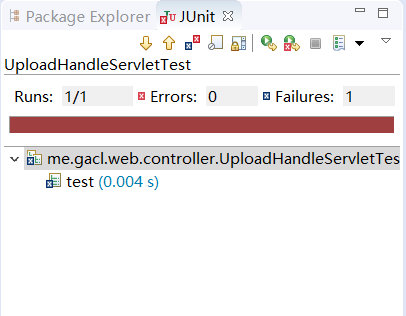
通过eclipse的JunitTest对进行单元测试ListFileServlet.java以及UploadHandleServlet.java进行单元测试 ，测试成功。测试结果如下图：  
 

图13 UploadHandleServlet.java测试结果

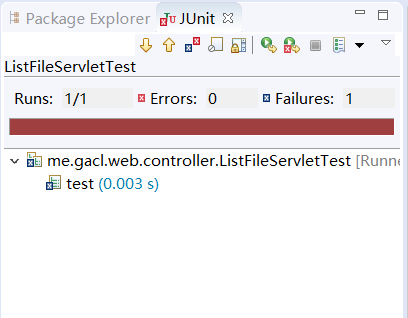


图14 ListFileServlet.java测试结果

## 五、持续集成

用git命令将整个源代码提交到GitHub的个人仓库上在通过GitHub自带的GitHub Action持续集成工具进行集成。通过 Github Action 集成的做法：在个人仓库的根目录下，创建一个 .github 文件夹，里面放一个 \*.yaml文件, 通过 Yaml 文件配置 Github Action 所用的文件。

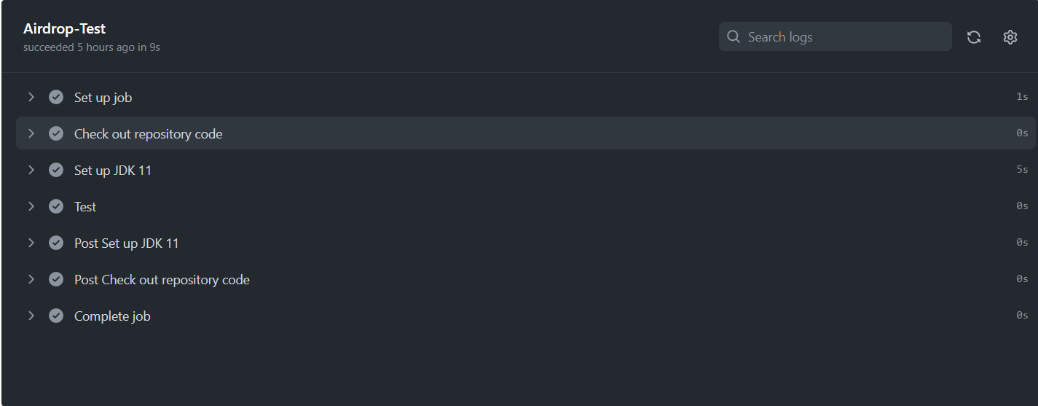


图15 Github Action 集成结果

## 六、亮点

本软件有两个亮点，其中之一是在文件上传的过程中通过文件的hash值分配地址，为文件设置唯一的文件名解决了文件覆盖的问题。第二个是在网页写的剪切板可以获得在本设备其它部分写入剪切板的内容，从而在网页端可以粘贴出来。

## 七、总结与反思

本项目在实现仿AirDrop文件上传基本没有问题，但在实现共享剪切板时任有问题存在。首先在剪切板功能中使用了clipboard API对复制、粘贴操作进行监听，由于非https的网页（除localhost外）不能使用clipboard API，而通过Tomcat发布的网页只能通过http访问，因此在手机端的剪贴板功能不能实现。其次由于浏览器的兼容问题，文件传输功能及剪切板只能在谷歌浏览器运行。最后剪切板的共享功能只能通过表单提交实现，并不能实现理想状态下的手机端复制内容，电脑端直接就可以粘贴的功能。因此本软件使用的方案不是个很好的方案，仍有很大的改进空间。