仿AirDrop功能网页项目

翟小慧 41911269 苌浅杭41911268

# 1 关键功能

## 1.1 文件传输

在桌面端编写一个jsp网页，当手机与电脑处于同一个局域网时，可以通过网页的方式发送文件。通过手机端访问该网页，可以在本地选择需要上传的文件或进行即时拍照，然后即可进行上传操作，或可以根据需要删除已选择文件从而便于进行重新选择。

## 1.2 复制粘贴

在文件传输的基础功能之上，在该jsp网页上添加复制粘贴的功能模块，该功能模块中只有一个按钮“复制”，能够实现在手机端复制文字后在电脑端进行粘贴的功能。在手机访问网页后，网页能够监听手机的复制操作，手机进行复制后，能够自动获取设备的剪切板内容，而后即可在电脑端右击粘贴或使用Ctrl-V进行粘贴。而网页上显示的复制粘贴面板中的内容也会及时更新。

# 2 关键代码

## 2.1文件传输

首先需要创建文件上传的保存路径，我们使文件的保存路径在WEB-INF目录下，这是因为在该路径下能够提高文件传输过程中的安全性能，具体代码如下：

        String uploadPath =this.getServletContext().getRealPath("/WEB-INF/upload");

        File uploadFile = new File(uploadPath);

        if (!uploadFile.exists()){

            uploadFile.mkdir(); //创建这个目录

        }

接着需要处理已上传的文件，解析请求时，第一步先获取文件名及文件后缀名，如果文件后缀名不是我们需要的，则直接return返回告诉用户文件类型不正确；如果文件后缀名正确，则获取文件上传的流，判断是否上传成功并返回上传结果，上传成功后关闭文件上传流，具体代码如下：

public static String uploadParseRequest(ServletFileUpload upload, HttpServletRequest request, String uploadPath)

            throws FileUploadException, IOException {

        String msg = "";

        // 解析前端请求

        List<FileItem> fileItems = upload.parseRequest(request);

        for (FileItem fileItem : fileItems) {

            //fileItem每一个表单对象

            if (fileItem.isFormField()) {// 判断上传的文件是否是带文件的表单

                // getFieldName指的是前端表单控件的name;

                String name = fileItem.getFieldName();

                String value = fileItem.getString("UTF-8"); // 处理乱码

                System.out.println(name + ": " + value);

            } else {// 判断是文件

                // ============处理文件==============

                // 拿到文件名

                String uploadFileName = fileItem.getName();

                System.out.println("上传的文件名: " + uploadFileName);

                if (uploadFileName.trim().equals("") || uploadFileName == null) {

                    continue;

                }

                // 获得文件上传的流

                InputStream inputStream = fileItem.getInputStream();

                // 创建一个文件输出流

                FileOutputStream fos = new FileOutputStream(realPath + "/" + fileName);

                System.out.println("path:"+realPath + "/" + fileName);

                // 创建一个缓冲区

                byte[] buffer = new byte[1024 \* 1024];

                // 判断是否读取完毕

                int len = 0;

                // 如果大于0说明还存在数据;

                while ((len = inputStream.read(buffer)) > 0) {

                    fos.write(buffer, 0, len);

                }

                // 关闭流

                fos.close();

                inputStream.close();

                msg = "文件上传成功!";

                fileItem.delete(); // 上传成功

## 2.2 复制粘贴

在实现复制粘贴功能时，需要设置监听器，对手机端发生的复制事件进行监听，当监听到动作时，通过JavaScript获取当前复制粘贴面板的内容，并向服务器发送请求，将该面板上的内容通过clipboard在电脑端实现粘贴功能。具体代码如下：

function copyToClipboard(text,callback) {

    var copyInput = document.getElementById('COPY\_INPUT');

    if (!copyInput) {

        copyInput = document.createElement('input');

        copyInput.setAttribute('id', 'COPY\_INPUT');

        copyInput.style.position = 'fixed';

        copyInput.style.left = '-100%';

        copyInput.style.top = '0';

        copyInput.style.zIndex = -100;

        copyInput.style.opacity = 0;

        document.body.appendChild(copyInput);

        }

    copyInput.value = text;

    copyInput.focus();

    copyInput.select();

    // document.execCommand 可能会被废弃

    if (document.execCommand('copy')) {

        document.execCommand('copy');

        }

    copyInput.blur();

    callback && callback(true);

    }

在使用clipboard时，如果复制粘贴面板的内容为空，则返回复制失败提示；若不为空，则点击复制按钮时，将会返回复制成功提示，具体代码如下：

document.addEventListener('copy',function(e){

        // 获得clipboardData 对象

        let clipboardData = e.clipboardData || window.clipboardData;

        var form=document.getElementById("clipboard\_form");

        // 如果 未复制或者未剪切，直接 return

        if(!clipboardData) return ;

        // 获得 Selection 对象 表示用户选择的文本范围或光标的当前位置。

        // 将用户输入的剪切或者复制的文本转化为字符串

        let text = window.getSelection().toString();

        if(text){

            // 如果文本存在，首先取消默认行为

            e.preventDefault();

            console.log("可以获取",text);

            document.getElementById("clipboard").value=text;

            alert('复制成功');

            form.submit();

        }

    })

# 3 单元测试

单元测试是在软件开发过程中要进行的最低级别的测试活动，指对软件中的最小可测试单元进行检查和验证，软件的独立单元将在与程序的其他部分相隔离的情况下进行测试。单元测试的要点是进行单元模块所有数据项的正确性、完善性测试，主要关注模块的算法细节和模块接口间流动的数据。其测试目的在于发现模块内部存在的各种错误。

在本项目中，我们对两个方法进行了单元测试。在对getDiskFileItemFactory方法进行的单元测试里创建DiskFileItemFactory对象，处理文件路径或大小限制，输出创建工厂。在对getServletFileUpload进行的单元测试里，输出了获取的ServletFileUpload的参数变量。

    @Test

    void getDiskFileItemFactory() {

        DiskFileItemFactory factory = new DiskFileItemFactory();

        System.out.println(factory);

    }

    @Test

    void getServletFileUpload() {

        DiskFileItemFactory factory = new DiskFileItemFactory();

        ServletFileUpload upload = new ServletFileUpload(factory);

        System.out.println(upload);

    }

# 4 使用教程

在手机端访问充当服务器的JSP页面192.168.43.29:8080/airdrop/index.jsp，即可进入到文件传输功能页面和复制粘贴功能页面，如下图所示。



## 4.1 文件传输

在手机端点击开始上传按钮即可选择需要上传的文件，文件可以是PDF、docx、png、jpg等格式，当选择好文件后，在文件路径框内自动生成拟定好的上传文件路径，点击提交，即可开始向电脑端传输文件。手机端上传文件图示如下：

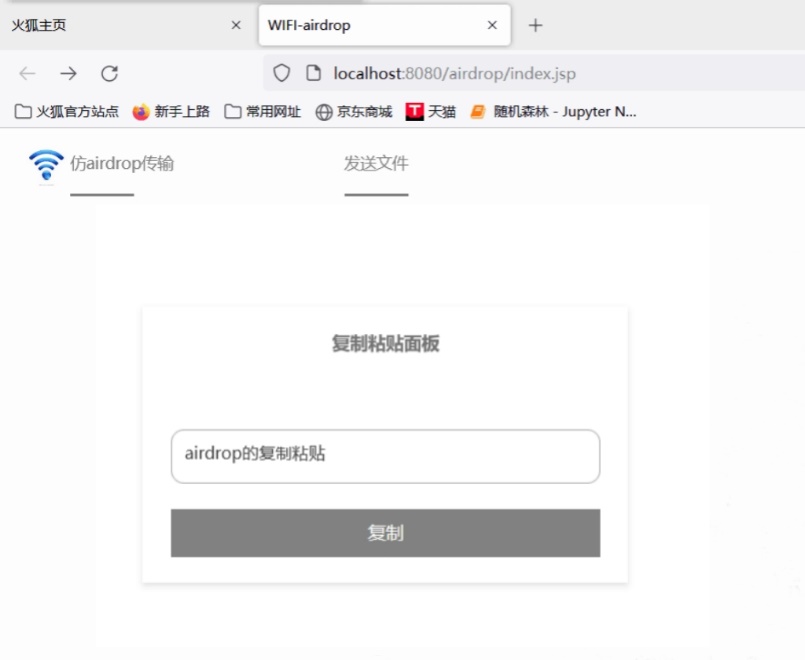


若上传的文件正确，则在显示上传结果，如下图所示：



## 4.2 复制粘贴

用户在手机端将需要复制的文字输入到复制粘贴面板中的文本框内，接着点击复制按钮，而后页面会自动向服务器提前当前的复制内容，成功提交后，手机端会出现“复制成功弹窗”，电脑端则会重新加载页面，显示从手机端粘贴过来的内容，如下图所示：

若复制粘贴面板的内容为空时，点击复制按钮，则会弹出复制失败的提示，如下图所示：

