1. **Folder Explore操作：**
   1. URL: admin/filemanager/(folder/)
   2. Client->server

请求方法: POST

请求参数: action=”folder”(url如果没有指定的情况下必须通过此参数指明操作)，path=请求文件夹的相对路径

* 1. Server->client

消息 = {folder: 文件夹数组}

文件夹数组 = [文件夹对象, 文件夹对象, … ]

文件夹对象 = { name: 字串, leaf: 布尔值(是否包含文件夹，不包含文件夹为true) , path：字串(相对于系统上传文件根文件夹的目录信息)}

1. **File Explore操作：**
   1. URL: admin/filemanager/(file/)
   2. Client->server

请求方法: POST

请求参数: action=”file”(url如果没有指定的情况下必须通过此参数指明操作)，json=请求内容

请求内容=[path对象]

Path对象={ path: 字串(请求文件夹的相对路径) }

* 1. Server->client

消息 = {file: 文件数组}

文件数组 = [文件对象, 文件对象, … ]

文件夹对象 = { name: 字串, url: 字串(相对于系统上传文件根文件夹的路径信息，由于后继步骤需要用此项构造请求文件的url，所以必须urlencode), thumbUrl: 字串}

1. **File Upload操作：**
   1. URL: admin/filemanager/(upload/)
   2. Client->server

请求方法: POST

请求参数: action=”upload”(url如果没有指定的情况下必须通过此参数指明操作)，json=请求内容

请求内容=[path对象]

Path对象={ path: 字串(请求文件夹的相对路径) }

* 1. Server->client

消息 = {file: 文件数组}

文件数组 = [文件对象, 文件对象, … ]

文件夹对象 = { name: 字串, url: 字串(相对于系统上传文件根文件夹的路径信息，由于后继步骤需要用此项构造请求文件的url，所以必须urlencode), thumbUrl: 字串}

1. **File Remove操作：**
   1. URL: admin/filemanager/(fileremove/)
   2. Client->server

请求方法: POST

请求参数: action=”fileremove”(url如果没有指定的情况下必须通过此参数指明操作)，json=请求内容

请求内容=[ path对象, [字串, 字串, …… ]]

Path对象={ path: 字串(请求文件夹的相对路径) }

* 1. Server->client

消息 = {file: 文件数组}

文件数组 = [文件对象, 文件对象, … ]

文件夹对象 = { name: 字串, url: 字串(相对于系统上传文件根文件夹的路径信息，由于后继步骤需要用此项构造请求文件的url，所以必须urlencode), thumbUrl: 字串}