

# 文件上传和下载在 ASP.NET 中的设计应用

张峥瑶<sup>1,2</sup>, 常利铭<sup>1,2</sup>

(1. 瓦斯灾害监控与应急技术国家重点实验室, 重庆 400039; 2. 中煤科工集团重庆研究院有限公司, 重庆 400039)

**摘要:** 在软件开发平台 VS2012 中进行软件开发, 通过用 JS 和 C# 语言对 Web 前端和后端的网页进行代码的编写开发, 使需要上传和下载的文件通过打开的 Web 网页进行本地的上传和客户端的下载, 从而达到设计应用的目的。对 Web 前端和后端文件上传和下载技术进行了重点阐述, 把这个实现的过程详尽地展示出来。

**关键词:** ASP.NET 编程环境; Web 前端和后端; 本地的上传; 客户端的下载

DOI:10.16184/j.cnki.comprg.2019.04.004

Web 客户端页面对文件的上传或者下载操作是 ASP.NET 开发过程中经常会用到的一项技术。这项技术可以给用户带来非常方便的文件操作, 可以直接在网页上进行文件的上传或者下载。通过 Web 网页上传或者下载的文件一般为除文件夹以外的其他文件, 例如压缩文件、Word 文档、Excel 文件、EXE 应用文件等等。所以在文件进行上传时, 要先对文件进行前期的处理, 使之成为可以上传的非文件夹文件。经过处理后的文件才可以进行上传, 否则进行上传的文件将无法进行上传。

文件的下载是将处理好的非文件夹文件放置在一个指定的文件夹内, 通过 Web 页面显示出来, 点击打开链接然后进行文件的下载和保存。

## 1 文件的上传

### 1.1 文件上传实现过程

在 VS2012 软件开发平台中, 首先要对需要上传的文件进行 Web 页面的控件选择, VS2012 自带的文件上传控件为 FileUpload, 选择此控件后, 还需要再添加一个 Button 按钮, 用来激活所选择需要上传的文件, 使这些控件置于 WebForm 页面上后进行控件的布局, 布局完成后, 再进行代码的编写。



图 1 多文件上传设计图

上传文件选择时, 控件 FileUpload 对文件的选择对话框, 会提示包括非文件夹以外的文件, 选择好需要上

传的文件后, 控件会把所选择的文件名称及类型显示在页面上。点击“文件上传”按钮, 就会把所选择的上传文件上传至服务器所在的指定文件夹里, 文件上传功能就实现了。

其中文件上传界面如图 1 所示。

文中只对非文件夹文件的上传做了详细的代码编写以及上传文件大小的详细说明, 上传文件的选择控件为 VS2012 编程平台自带的 FileUpload 控件, 点击编译过的该控件后程序会自动弹出文件选择对话框, 从而可以让用户自主选择自己需要上传的文件, 例如其中文件上传按钮的原代码如下:

```
protected void FilesUpBut_Click(object sender, EventArgs e)
{
    HttpFileCollection uploadFiles = Request.Files;
    string filePath = Server.MapPath("../UpLoadfiles/files/");
    string fileName = filePath + FileUpload1.FileName; //获得上传文件的文件名
    if (FileUpload1.HasFile || FileUpload2.HasFile || FileUpload3.HasFile || FileUpload4.HasFile)
    {
        for (int i = 0; i < uploadFiles.Count; i++)
        {
            HttpPostedFile userPostedFile = uploadFiles[i];
            if (userPostedFile.ContentLength > 0)
            {

```

**作者简介:** 张峥瑶 (1993-), 女, 本科, 经济学学士, 助理工程师, 研究方向: 安全监控软件的开发以及测试、煤矿安全监控系统产品的检测等。

收稿日期: 2019-01-13

```

        userPostedFile.SaveAs(fileName);
    }
}

Response.Write("<script>alert('恭喜您,上传成功!')</script>");
}
else
{
    Response.Write("<script>alert('请选择要上传的文件!')</script>");
}
}

```

对上传文件按钮的点击事件进行代码编写,首先要将上传文件的路径确定下来,这样才可以在选择好文件后把文件上传到服务器指定的文件夹里。所选择的文件默认为非文件夹选项,当上传成功后,页面会以对话框的形式提示上传成功。此处指定了一次上传 4 个所选择的文件。

## 1.2 上传文件的大小要求

在项目生成时,系统自动生成的 web.config 配置文件里面,默认的上传文件大小为 maxRequestLength="40960",即为 4MB (1MB 为 1024KB) 大小,所以如果在页面的客户端上传的文件大小超过 4MB 时,那么文件上传页面就会出错,所以配置文件中上传文件的大小需要进行更改,才可以上传比较大的文件。

配置文件中上传文件的大小需要对 maxRequestLength="40960" 进行修改,可根据自己的需要修改本文中修改上传文件的大小修改为 40MB,也可以在服务器端的 Internet 信息服务 (IIS) 管理器上进行修改发布,而此发布平台对上传的文件大小的限制默认为 30MB。如果需要进行较大文件的上传,那么可以通过两种途径来实现。一种是在自己的 web.config 配置文件中进行修改,此种方法上传的文件一般为百兆级别,另一种是借助第三方控件例如 NeatUpload 控件,可以上传千兆级别的大文件,此种上传方式的上传速度相比修改配置文件进行文件上传的速度要快很多。

## 2 文件的下载

### 2.1 加载显示的数据及资料下载链接

在文件上传到服务器指定的文件夹以后,就可以在下载页面对需要下载的文件进行下载。本文通过<asp:GridView>网页控件加载显示的数据以及加载 LinkButton 按钮控件,来实现对需要的文件进行下载。其加载

主页面如图 2 所示。

**KJ90-F16(C)矿用本安分站**

序号	产品类型	产品名称	中试报告	产品资料	发表时间	资料下载
2	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
6	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
10	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
14	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
18	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
22	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
26	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载
30	分站	KJ90-F16(C)矿用本安分站	KJ90-F16(C)矿用本安分站中试报告	BOM表等	2018/10/10 0:00:00	资料下载

图 2 资料显示以及下载主界面

### 2.2 资料下载代码编写

点击资料下载即在 LinkButton 按钮点击事件中调用文件下载页面,对需要下载的文件进行选择下载。其中资料下载页面如图 3 所示,LinkButton 按钮点击事件代码如下所示:

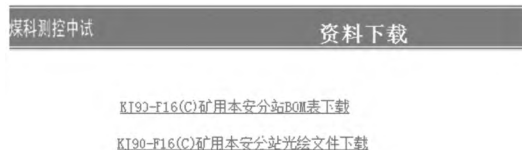


图 3 资料下载界面图

```

/////////////////////////////////LinkButton 按钮点击事件/////////////////////////////////
protected void LinkButton_Click (object sender, EventArgs e)
{
    string strBm1 = string.Format ("@/{0}", PublicClass.strFresh), strBm2 = string.Format ("@/{0}/BOM表", PublicClass.strFresh),
    filename1 = Server.MapPath ("DownLoadFiles" + strBm1),
    filename2 = Server.MapPath ("DownLoadFiles" + strBm2);
    DirectoryInfo root1 = new DirectoryInfo (filename1);
    DirectoryInfo root2 = new DirectoryInfo (filename2);
    bool fiexist1 = root1.Exists, fiexist2 = root2.Exists;
    if (! fiexist1 || ! fiexist2)
    {
        Response.Write("<script>alert('请确认 BOM 表文件所在文件夹是否存在!!','系统提示')</script>");
    }else
    {
        FileInfo[] files = root2.GetFiles();//只获得路径下所有
        //文件名称信息
        if (files.Length == 0)
    }
}

```

(下转第 26 页)

两个算法训练过程中, 所获得分如图 1, 图 2 所示, 从这两幅图中可以看出, 虽然在学习过程中, 两个算法都在较短的时间内学习到最优策略, 并且最终都将得分稳定在较高的水平, 接近设定的最高值, 但是这两个算法的平均得分却表现出了不同的变化情况。具体来说, 就是在 Pong 游戏中, LSTM-DRL 算法中训练网络的收敛速度远比 GA3C 快; 而在 Boxing 游戏中, LSTM-DRL 算法中训练网络的收敛速度则稍慢于 GA3C。但是, 在训练初期, LSTM-DRL 算法平均得分的上升速度都明显高于 GA3C。

### 3 结语

LSTM-DRL 算法得益于 LSTM 保存和处理历史状态信息的出色表现, 较大减小了系统在学习过程中所需的

(上接第 16 页)

```
{
Response.Write("<script>alert(' 请确认 BOM 表文件
是否存在!! ',' 系统提示 ')</script>");
}
else
{
string [] filestr = Directory.GetFiles (file-
name2);
FileStream fs = new FileStream (filestr [0],
FileMode.Open);
byte[] bytes = new byte[(int)fs.Length];
fs.Read(bytes, 0, bytes.Length);
fs.Close();
Response.ContentType = "application/
octet-stream";
Response.Charset = "UTF-8";
//通知浏览器下载文件而不是打开
Response.AddHeader ("Content-Disposition", "attachment; filename=" + Server.UrlEncode(files
[0].Name)); //防止汉字转换时不是乱码出现
Response.BinaryWrite(bytes);
Response.Flush();
Response.End();
} } }
```

以上 LinkButton 点击事件代码编写, 首先要确定下载文件的路径, 如果下载文件路径中的文件不存在, 系统会进行出错提示, 如果下载的资料中下载文件存在, 那么通过文件流读取的方式, 把要下载的文件读取解码

计算规模和存储规模, 学习速度和效果得到了明显的提升。相较于 GA3C, 网络结构复杂度增加, 对复杂环境的特征表达能力也得到了进一步加强。通过实验可以看出, 该算法在求解 POMDP 问题时的表现得到了明显的提高。

### 参考文献

- [1] 刘全, 闫岩, 朱斐, 吴文, 张琳琳. 一种带探索噪声的深度循环 Q 网络 [J/OL]. 计算机学报, 2019: 1-16.
- [2] 徐松林. 深度强化学习概述 [J]. 电脑知识与技术, 2019, 15 (03): 193-194.
- [3] 罗舒俊. 基于深度强化学习的多代理马尔科夫游戏 [J]. 电脑编程技巧与维护, 2018, (12): 10-13.

并保存在客户端系统中。整个资料的下载就完成了。

### 3 资料查询系统的使用效果

资料查询系统的开发和使用, 在实际应用中取得了不错的效果, 使用此系统的所有部门可以进行资料的电子文档保存以及相关资料的系统查询, 不用再去拿纸质文档资料进行比对和复印, 可以直接在查询系统中下载自己需要的资料, 然后打印新的资料更换老的资料等操作, 此系统的应用大大提高了部门人员的办公效率, 得到了相关部门的一致认可。

### 4 结语

非文件夹文件的上传和下载在网页制作过程中是经常要用到的一项常规技术, 详细讲述了文件如何上传和下载的过程, 以及每个功能实现的详细代码说明, 包括制作完成后的网站对于文件的上传、下载的使用情况等。

### 参考文献

- [1] 商杰, 朱忠旭. ASP.NET 平台多文件的上传和下载 [J]. 工业控制计算机, 2013, 12: 42-43.
- [2] 李颖云, 张克. 基于 ASP.NET 的文件上传和下载 [J]. 电脑知识与技术, 2007, 06: 47-50.
- [3] 李向军, 付雪峰. ASP.NET 程序设计 [M]. 清华大学出版社.
- [4] (英) 夏普. Visual C# 2010 从入门到精通 [M]. 清华大学出版社.