

使用 IntraWeb 实现通用的 HTML 文本编辑器

高 勇

摘 要: 在 Delphi7 下, 使用 IntraWeb 实现了两种形式的 HTML 文本编辑器, 前者不可上传文件, 后者能够上传文件。详述了其实现的关键步骤并给出了所有源代码。

关键词: IntraWeb; HTML; 文本编辑器; Ckeditor

看到网上论坛中的 HTML 文本编辑器, 很是喜欢, 可笔者用的是 IntraWeb, 它又没有带这样的控件, 在好奇心的驱动下, 于是决定自己亲手试一下。经过不懈的努力, 已基本实现当初的设想。

先看一下运行效果如图 1 所示。



图 1 运行效果

1 编辑器的实现

新建一个 IntraWeb 工程, 命名为 Project1.dpr。请打开源代码自己对照来看。

以下是关键步骤:

将“控件”放到 files 文件夹中(将 images 文件夹拷贝到 files 文件夹下), 不要改变其中任何内容, 这是必要条件。

请看 Unit1 单元:

将 TIWMemo 控件放在该页面上, 并命名为 IWMemo110 (切记一定要是这个名字), 将其拉伸到最小, 放到其他可视控件的后面, 这里是放在 IWRectangle1 控件的后面, 移动 IWRectangle1 控件即可看到。切记一定要放在可视控件的后面, 这样做是为了在程序运行时不被用户看到。如果它在其他可视控件的前面, 请在 IWMemo110 上点击右键, 选择 Control 下的 Send to Back 即可。

将 IWRectangle 控件放在页面上, 这里面将放置 HTML 多功能编辑器, 它的宽度为程序运行时的实际宽度

编写如下函数: (导入 HTML 多功能编辑器的代码)

当页面生成时运行时下代码:

```
procedure TIWForm1.IWAppFormCreate(Sender: TObject);
var
  AText:TStringList;
  AFile:String;
begin
  //找到“控件”中的 edithtml.htm 文件。WebApplication.Ap
  //plicationPath 为主程序所在根目录
  AFile:=WebApplication.ApplicationPath+'files/images/post/
  edithtml.htm';
  //如果文件存在,则将其内容导入到 AText,再将 AText.Text
  //的值赋予 IWRectangle1.Text,这样,程序运行时就会在 I
  //WRectangle1 中显示我们的 HTML 文本编辑器
  if FileExists(AFile) then begin
    AText:=TStringList.Create;
    AText.LoadFromFile(AFile);
    IWRectangle1.Text:=AText.Text;
    AText.Free;//释放临时创建的 AText 变量
  end;
end;
```

在窗口中放置一个提交用的按钮 IWButton1, 其 TIWScriptEvents 中的属性 onMouseDown 中加入如下代码:

```
Dvbbs_CopyData('Body');
return Checkpostdata();
```

程序中对 IWMemo110.text 的操作就是对 HTML 多功能编辑器中的内容的 HTML 代码的操作, 将 IWMemo110.text 内容保存在数据库中, 也就是 HTML 多功能编辑器中所显示的内容的 HTML 代码被保存在数据库中了。以下是点击提交按钮后的对 IWMemo110.tex 的示例操作:

```
procedure TIWForm1.IWButton1Click(Sender: TObject);
begin
  //显示多功能编辑器中的内容的 HTML 代码,这只是为了验
  //证我们的程序
```



```
webapplication.ShowMessage(IWMemo110.Text);
//将 IWMemo110.Text 中的内容用 IWRectangle2 控件输
//出,程序运行时就会看到 HTML 多功能编辑器中所显示的内容
IWRectangle2.Text:=IWMemo110.Text;
end;
```

由于 images 文件夹中的内容基本不用修改,其中的内容就不作过多讲述,有兴趣可自行审查。

这个例子没有自动绑定本机 IP,大家可以采用 3.2 中的方法,在 ServerController 单元中加入 LocalIP 函数,并进行相应的赋值操作。这样如果本机有网络的话,其地址就可以被其他机器访问,大家可以在其他机器中看到运行效果,应该是和本机一样的。

补充说明:

(1) 如果一定要改变 IWMemo 控件的名字,不想用 IWMemo110 这个名字,例如改为 IWMemook,就要在 images 文件夹中找到 edithtml.htm 和 DhtmlEdit1.js,将其中的 IWMEMO110 替换成 IWMEMOOK (一定要全部大写)。再次调试程序时请在 IE 的 internet 选项中删除 internet 临时文件。否则调试会出错。

(2) 如果要限制输入文字的多少,请在 images 文件夹中找到 edithtml.htm 修改第 4 句:

MaxConlength=1624000000;数字即为文字的数量,当然,不是很精确,只是大概。这里默认数字很大,目的是不限字数。

(3) 同样,Project_dll.dpr 为生成 dll 版本的工程文件。要用 IIS 发布才能查看。所有代码在 WinXPSP2+Dlphi7.0 + IntraWeb9.0.45 下调试通过。IWServerController 的属性 ComInitialization: 发布到 Windows XP SP2, Windows2000, 请选择 ciMultiThreaded, 发布到 Windows 2003 SP2 请选择 ciNormal。

2 上传功能的实现

众所周知,IntraWeb 下自带的在线编辑器是 TMS 公司 TMS IntraWeb 控件包所提供的,其功能对于目前的 Web 应用稍嫌不足,主要界面如图 2 所示。

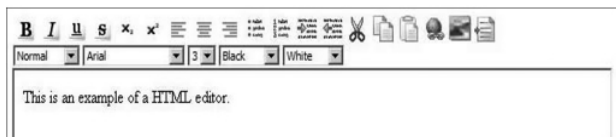


图 2 自带在线编辑器

实现过程:

新建一个 IntraWeb 工程,命名为 IWWeb.dpr。请打开源代码自己对照来看。

以下是关键步骤:

(1) 修改 ServerController 的 AllowSubFolders 项为 True, CharSet 为 gb2312, 勾选 SupportedBrowsers 下的要支持的浏览器项, 修改 ServerController 的 SessionTrackingMethod 属性为 tmCookie。

在项目下建立 Files 目录,并在其下建立 ckeditor 和 Upload 两个目录用来保存 CKEditor 在线编辑器及上传文件, Upload 下再分别建立 Media、File、Image 三个目录分别用来保存 CKEditor 上传的 Flash、一般文件、图片文件。CKEditor 可在 <http://ckeditor.com/> 下载,将压缩包解压至 ckeditor 目录下(要升级后面有详解)。

(2) 指定 IWForm 中 Ckeditor 所要用的编辑框。

在 IWForm 中加入 IWMemo 控件,在 IWForm 的 Extra-Header 中输入代码<script type=" text/javascript" src=" ../Files/ckeditor/ckeditor.js" ></script>。

在 IWForm 中的 OnRender 添加,如在 OnCreate 添加,则只要刷新过页面,编辑器的文件上传功能就失效。

```
procedure TfrmDefault.IWAppFormRender (Sender: TObject);
var jsString:string;
begin
    jsString:='CKEDITOR.replace("MEMEDITOR",{filebrowser-
ImageUploadUrl:"";
    jsString:=jsString+ '/Default.Html/?Action=UploadImage",
';
    jsString:=jsString + 'filebrowserUploadUrl:"/Default.Html/?
Action=UploadFile",';
    jsString:=jsString + 'filebrowserFlashUploadUrl:"/Default.
Html/?Action=UploadFlash"});';
    Self.AddToInitProc(jsString);
end;
//MEMEDITOR 为 iwmemo 控件名大写
```

(3) 提交前的处理

在窗口中放置一个提交用的按钮 IWButton1,其 TILWScripEvents 中的属性 onMouseDown 中加入如下代码 (// MEMEDITOR 为 iwmemo 控件名大写):

```
var oEditor = CKEDITOR.instances.MEMEDITOR;
document.getElementById ("MEMEDITOR").value = oEditor.
getData();
```

这样做的目的是什么呢?大家可以试一下,如果没有这一步,将编辑框内的值取出来时空,且刷新页面后编辑框内的内容为空。这是将编辑框内的 html 代码取出来的关键一步。

(4) 配置 CKEditor

打开 Files\ckeditor 下 config.js 文件对编辑器作如下配置:

```
CKEDITOR.editorConfig = function( config )
{
```

*****NETWORK & COMMUNICATION*****

```
// Define changes to default configuration here. For exam
//ple:
// config.language = 'fr';
// config.uiColor = '#AADC6E';
config.language = 'zh-cn'; // 配置语言
//config.uiColor = '#FFF'; // 背景颜色
config.width = 'auto'; // 宽度
config.height = '300px'; // 高度
//config.skin = 'office2003';界面 v2,kama,office2003
config.toolbar = 'Full';
config.font_names = 'Arial;Times New Roman;Verdana;宋
体;楷体;黑体;仿宋体;隶书;微软雅黑';
config.fontSize_defaultLabel = '12px';
config.fontSize_sizes = '8/8px;9/9px;10/10px;11/11px;12/
12px;14/14px;16/16px;18/18px;20/20px;22/22px;24/24px;26/
26px;28/28px;36/36px;48/48px;72/72px';
//config.filebrowserBrowseUrl='ckfinder/ckfinder.html';
config.toolbar_Full =
[
    ['Source','-', 'Save','NewPage','Preview','-', 'Tem-
plates'],
    ['Cut','Copy','Paste','PasteText','PasteFromWord','-',
    'Print','SpellChecker','Scayt'],
    ['Undo','Redo','-', 'Find','Replace','-', 'SelectAll','Re-
moveFormat'],
    ['Form','Checkbox','Radio','TextField','Textarea',
    'Select','Button','ImageButton','HiddenField'],
    '/',
    ['Bold','Italic','Underline','Strike','-', 'Subscript','Su-
perscript'],
    ['NumberedList','BulletedList','-', 'Outdent','Indent',
    'Blockquote'],
    ['JustifyLeft','JustifyCenter','JustifyRight','Justify-
Block'],
    ['Link','Unlink','Anchor'],
    ['Image','Flash','flvPlayer','Table','HorizontalRule',
    'Smiley','SpecialChar','PageBreak'],
    '/',
    ['Styles','Format','Font','FontSize'],
    ['TextColor','BGColor'],
    ['Maximize','ShowBlocks']
];
//config.extraPlugins = 'flvPlayer' 这是另外为 ckeditor 开发
//的插入 FLV 视频插件
//config.filebrowserBrowseUrl='';以下三行为配置
//ckeditor 文件管理的地址
//config.filebrowserImageBrowseUrl='ckfinder/ckfinder.
//html?Type=Images';
//config.filebrowserFlashBrowseUrl='ckfinder/ckfind
//er.html?Type=Flash';
};
```

(5) 为 Ckeditor 实现上传图片、文件，及 Flash 功能，
在 ServerController 的添加上传代码，具体如下：

```
unit ServerController;
unit ServerController;
interface
uses
    SysUtils, Classes, IWServerControllerBase, IWBaseForm,
    HTTPApp,
    // For OnNewSession Event
    WinSock, ShellAPI, Variants, StrUtils, //新加入的：
    UserSessionUnit, IWApplication, IWAppForm, IniFiles;
type
    TIWServerController = class(TIWServerControllerBase)
    procedure IWServerControllerBaseNewSession (ASes-
    sion: TIWApplication;
        var VMainForm: TIWBaseForm);
    procedure IWServerControllerBaseBeforeRender (ASes-
    sion: TIWApplication;
        AForm: TIWBaseForm; var VNewForm: TIWBase-
    Form);
    procedure IWServerControllerBaseCreate (Sender: TOB-
    ject);
    procedure IWServerControllerBaseBackButton (ASubmit-
    tedSequence,
        ACurrentSequence: Integer; AFormName: string; var
    VHandled,
        VExecute: Boolean);
    private
    public
    end;
function UserSession: TIWUserSession;
function IWServerController: TIWServerController;
implementation
{$R *.dfm}
uses
    IWInit, IWGlobal, IWForm, IWContainer, Upload;
function getRandFileName: string; //获得随机文件名
var strDateTime: string;
begin
    DateTimeToString (strDateTime, 'YYYYMMDDHH-
    MMSSZZZ', Now);
    Dat eTimeToString (strDateTime, 'YYYYMMDDHH-
    MMSSZZZ', Now);
    Result := strDateTime + Char(97 + Random(26)) + Char
    (97 + Random(26)) + Char(97 + Random(26)) + Char(97 +
    Random(26)) + Char(97 + Random(26)) + Char(97 + Random
    (26)) + Char(97 + Random(26)) + Char(97 + Random(26)) +
    Char(48 + Random(10)) + Char(48 + Random(10));
end;
function BuildPage(jsScript: string; Text: string): string;
//输出页面格式
```



```

begin
    Result := '<HTML><head><META HTTP-EQUIV="Content-Type" CONTENT="text/html; ' +
        'CHARSET=gb2312"><meta http-equiv="cache-control" content="no-cache"><meta http-equiv="pragma" content="no-cache"> <meta http-equiv="Content-Script-Type" content="text/javascript">' + jsScript + '</head><BODY>' +
        Text + '</BODY></HTML>';
end;
function IWServerController: TIWServerController;
begin
    Result := TIWServerController(GServerController);
end;
function UserSession: TIWUserSession;
begin
    Result := TIWUserSession(WebApplication.Data);
end;
procedure TIWServerController.IWServerControllerBaseBackButton(
    ASubmittedSequence, ACurrentSequence: Integer;
    AFormName: string;
    var VHandled, VExecute: Boolean);
const
    DefaultFormName = 'frmDefault';
type
    TIWFormClass = class of TIWForm;
var
    LForm: TIWForm;
begin
    // This is a generic back-button handler that can easily
    // be adapted to
    // your application
    VHandled := True;
    VExecute := True;
    if AFormName = '' then
        AFormName := DefaultFormName;
    if WebApplication.FindComponent(AFormName) <> nil
    then
        begin
            WebApplication.SetActiveForm(WebApplication.FindComponent(AFormName) as
            TIWContainer);
        end else begin
            try
                LForm := TIWFormClass(FindClass('T' + AFormName)).Create(WebApplication);
                WebApplication.SetActiveForm(LForm);
            except
                VHandled := false;
            end;
        end;
    end;
end;
end;

```

```

procedure TIWServerController.IWServerControllerBaseBeforeRender(
    ASession: TIWApplication; AForm: TIWBaseForm; var
    VNewForm: TIWBaseForm);
const
    imgFileExt = '.gif .bmp .png .jpg .jpeg .jpe';
    mediaFileExt = '.swf .flv';
    normalFileExt = '.rar .zip .7z .doc .docx .xls .xlsx .ppt .pptx .rtf .txt';
var nameSavedFile: string;
    fileExt, jsScript, funNum, messageNotice: string;
    fileStream: TFileStream;
    dirSavedFile: string;
    allowFileExt: string;
begin
    { ----- 在线编辑器上传文件 ----- }
    if Pos('CKEditorFuncNum', ASession.Request.Query) > 0
    then
        begin
            { ----- 2 为 FireFox 浏览器 1 为其他浏览器 ----- }
            if Pos('CKEditorFuncNum=1', ASession.Request.Query)
            > 0 then
                funNum := '1'
            else
                funNum := '2';
            ASession.RunParams.Clear; //清除运行参数
            ASession.RunParams.Add('funNum=' + ASession.Request.QueryFields.Values['CKEditorFuncNum']); //TISAPIRe
            //quest 可以正确获取浏览器参数值,但 StandAlone 模式无法
            //获取,只能在 Query 中查询
            //运行参数用于向上传页面传递返回编辑器的值,不能如
            //StandAlone 模式下临时构建一个页面

            { ----- 设定上传文件目录 ----- }
            if Pos('Action=UploadImage', ASession.Request.Query)
            > 0 then
                begin
                    dirSavedFile := 'Image';
                    allowFileExt := imgFileExt;
                end
            else if Pos('Action=UploadFlash', ASession.Request.Query) > 0 then
                begin
                    dirSavedFile := 'Media';
                    allowFileExt := MediaFileExt;
                end
            else
                begin
                    dirSavedFile := 'File';
                end
            end;
        end;
    end;
end;

```

*****NETWORK & COMMUNICATION*****

```

allowFileExt := normalFileExt;
end;
if ASession.Request.Files.Count = 0 then
Exit
else
begin
fileExt := LowerCase(ExtractFileExt(ASession.Request.
Files[0].FileName));
if Pos(fileExt, allowFileExt) = 0 then
begin
ASession.RunParams.Add('messageNotice=禁止上传
该文件类型! ');
ASession.RunParams.Add('backUrl= ');
messageNotice := '禁止上传该文件类型! ';
end
else
begin
nameSavedFile := getRandFileName() + fileExt;
fileStream := TFileStream.Create(ASession.Applica-
tionPath +
'Files\Upload\' + dirSavedFile + '\' + nameSaved-
File, fmCreate or fmShareDenyWrite);
try
FileStream.CopyFrom (ASession.Request.Files [0].
Stream, 0);
{ 0 = 从开始复制所有数据 }
ASession.RunParams.Add('messageNotice=文件上
传成功! ');
ASession.RunParams.Add('backUrl=../Files/Upload/
' + dirSavedFile + '/' + nameSavedFile);
nameSavedFile := 'Files/Upload/' + dirSavedFile +
'/' + nameSavedFile;
messageNotice := '文件上传成功! ';
finally
FileStream.Free;
end;
end;
jsScript := '<script type="text/javascript">' +
'window.parent.CKEDITOR.tools.callFunction (' +
funNum + ', "' + nameSavedFile + '", ' +
'"' + messageNotice + '");</script>';
ASession.Response.Content := BuildPage(jsScript, '');
//StandAlone 应用程序实现时用这个
//VNewForm:=TfrmUpload.Create(ASession);显示返回页面
//接收参数 ISAPI 的实现时用这个
Exit;
end;
end;
{-----}
end;
procedure TIWServerController.IWServerControllerBase-

```

```

Create(Sender: TObject);
function GetLocalIP: string; //得到本机 IP 地址
type
TaPInAddr = array[0..10] of PInAddr;
PaPInAddr = ^TaPInAddr;
var
phe: PHostEnt;
pptr: PaPInAddr;
Buffer: array[0..63] of char;
I: Integer;
GInitData: TWSADATA;
begin
WSAStartup($101, GInitData);
Result := '';
GetHostName(Buffer, SizeOf(Buffer));
phe := GetHostByName(Buffer);
if phe = nil then Exit;
pptr := PaPInAddr(phe^.h_addr_list);
I := 0;
while pptr[I] <> nil do begin
Result := StrPas(inet_ntoa(pptr[I]));
Inc(I);
end;
WSACleanup;
end;
begin
self.CacheDir := self.GetAppPath + 'cache';
//更改程序的 cache 目录
self.BoundIP := GetLocalIP; //自动绑定本机 IP
end;
procedure TIWServerController.IWServerControllerBase-
NewSession(
ASession: TIWApplication; var VMainForm: TIWBase-
Form);
begin
ASession.Data := TIWUserSession.Create(nil);
end;
initialization
TIWServerController.SetServerControllerClass;
Randomize;
end.

```

注: IntraWeb 的 ISAPI 的实现 (工程文件为 Index.dpr)

ISAPI 具有占用资源小、速度快的特点, 更适合制作网站, 其在线编辑器实现方式与 StandAlone 方式编译成可执行文件略有不同, 具体如下: (发布时一定要建立一个虚拟目录, 访问该页面的地址要类似: http://ip/虚拟目录/index.dll)

(1) 修改上面的相应代码

```

//ASession.Response.Content := BuildPage(jsScript, '');
//StandAlone 应用程序实现时使用

```

(下转第 58 页)




```

object=2;
end
%端点 1 右边和端点 2 上面均有连接点
if Point(1).right+Point(2).upper==2
object=7;
end
end
if EndPointNumber>=3    %3,4,5
%端点 1 右、上均无连接点
if Point(1).right+Point(1).upper==0
object=5;
else %端点 1 右、上均有连接点
object=3;
end
%最后一个端点右、下无连接点
if Point(EndPointNumber).right+
Point(EndPointNumber).down==0
object=4;
end
end
if objectEuler==2 %两个对象,没有孔:5
object=5;
end PostcodeNO=strcat(PostcodeNO,String(object+1));

```

```
%objectEuler
%EndPointNumber
end
disp(PostcodeNO); %显示识别结果
```

4 结语

对邮政编码手写体进行分析,提出了依据数字号码的拓扑结构特征和端点特征的识别方法,该识别方法较简单可行,对多个样本图像进行测试(将程序中的 Sample01.jpg 替换为待测试的样本图像文件名),均能快速、准确识别。但对于书写不规范的邮政编码识别,软件算法有待优化。

参考文献

- [1] 吴谨, 邱亚. 基于空间分布特征的手写体数字识别 [J]. 武汉科技大学学报 (自然科学版), 2004, 27 (2): 176-178.
 - [2] 苏金明, 王永利. MATLAB7.0 使用指南. 电子工业出版社, 2004.
 - [3] 吕凤军. 数字图像处理编程入门—做一个自己的 Photoshop. 北京: 清华大学出版社, 1999.
- (收稿日期: 2010-10-15)

(上接第 51 页)

```
VNewForm:=TfrmUpload.Create(ASession);  
//显示返回页面接收参数 ISAPI 的实现时使用
```

(2) 返回页面

新建一个 IWForm，改名为 frmUpload

在 onCreate 加入代码：

```

procedure TfrmUpload.IWAppFormCreate (Sender: TObject);
var jsScript:string;
begin
{ -----将上传的文件地址返回
给在线编辑器----- }
jsScript:='window.parent.CKEDITOR.tools.callFunction (' +
WebApplication.RunParams.Values ['funNum'] + ', "' + WebApplication.RunParams.Values['backUrl'] + '", ' +
'"'+WebApplication.RunParams.Values['messageNotice'] + '");';
AddToInitProc(jsScript);
end;

```

在 `onAfterRender` 加入代码，这一点很重要，否则如果上传过文件后，界面的其他按钮无效。

```
procedure TfrmUpload.IWAppFormAfterRender (Sender:
TObject);
begin
```

```
Self.Free;  
end;
```

(3) CKEditor 的升级

先将 ckeditor 文件夹下的 skins 文件夹和 config.js 文件备份，然后在 <http://ckeditor.com/> 下载 ckeditor 的最新版本，将压缩包解压至 ckeditor 目录下覆盖现有文件，再将备份的文件拷贝回去覆盖即可，请严格按照此步骤执行。

最终在线编辑器实现界面如图3所示。

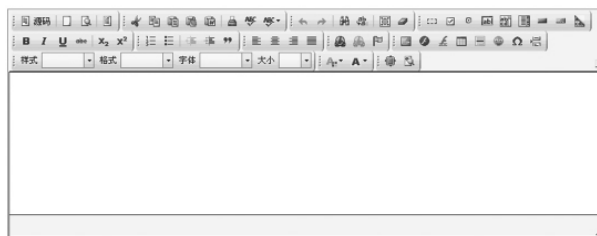


图 3 编辑器实现界面

3 结语

以上代码在 Delphi7+IntraWeb 10.21 中编译通过，编辑器在 IE8 运行正常，相信大家能在这个基础上开发出相应的文件管理或其他插件。

(收稿日期: 2010-10-20)