

新闻管理系统的具体实现及其优点

徐景忠, 赵迪

(厦门大学 自动化系, 福建 厦门 371005)

摘要: 该系统结构合理, 功能完善, 实现了新闻的动态管理, 采用前台浏览和后台管理双层架构, 前台部分包括用户登陆、新闻浏览及发布功能, 后台管理则包括新闻添加、修改和删除功能。该系统用户分为普通游客和注册用户。普通游客可以方便地浏览新闻, 注册用户可以方便、快捷地发布和管理新闻。新闻发布及后台管理页面包含身份验证的内容, 未登陆的用户无法访问。在系统中设计了一个通用类, 方便后台页面对数据库所作的各种操作, 比如读取、写入或更新数据库。在各个页面的上方都添加了其它页面的链接, 以便于在不同页面之间快速切换。在首页和新闻列表页面显示新闻列表的地方使用了 GridView 控件, 以便于格式化地显示新闻信息。

关键词: 新闻管理; ASP.NET; ADO.NET; Browser/Server

中图分类号: TP311.52 **文献标识码:** A **文章编号:** 1009-3044(2010)01-93-03

Concrete Realization and its Benefits of News Management System

XU Jing-zhong, ZHAO Di

(Department of Automation, Xiamen University, Xiamen 371005, China)

Abstract: This system has realized dynamic management of the news, including foreground view and background management parts, having general function as a news administrative system. Foreground part includes a user login, check and publish news features, the background management includes the news editing and news deleting functions. Registered users can quickly publish and manage news, and guests could read news easily. News issuing and background management pages contain authentication, logged-off users can not access these pages. I have written a foundation class and it's useful when manipulating database, for instance, reading, writing or updating database. There is hyperlinks directing to other pages on each page, and it will be convenient to switch between different pages. GridView Control is added to home page and news list page, news will be orderly displayed in the GridView.

Key words: news administrative; asp.net; ado.net; Browser/Server

1 需求分析

该系统用于新闻发布, 需要具有一个新闻管理系统的基本功能, 对其做需求分析, 可得一个新闻管理系统应具备的基本功能如下:

- 1) 前台新闻查看
- 2) 后台新闻管理
- ① 新闻发布
- ② 新闻修改
- ③ 新闻删除

根据以上需求分析, 该系统要实现的主要功能为:

- 1) 前台新闻查看部分
- ① 用户登陆
- ② 显示最新的几条新闻
- ③ 显示所有新闻
- ④ 新闻详细信息阅读
- 2) 后台管理部分
- ① 登录验证
- ② 根据权限显示相应的管理目录
- ③ 添加、修改和删除新闻
- ④ 管理发布新闻

1.1 新闻管理系统的模型结构

该系统的模型结构如图 1 所示。

该系统结构分为三个逻辑层:

Web 层: Web 层为客户端浏览器提供对应用程序的访问, 这一层是作为解决方案文件中的 Web 项目实现的。Web 层由 ASP.NET Web 窗体和代码隐藏文件组成,

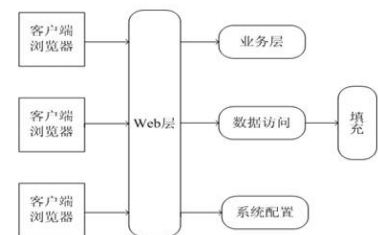


图 1 新闻管理系统的模型结构图示

收稿日期: 2009-11-05

作者简介: 徐景忠 (1985-), 男, 浙江龙游人, 厦门大学自动化系在读硕士, 从事软件开发应用, 硬件开发; 赵迪 (1986-), 女, 河北辛集人, 在读硕士, 从事软件开发应用。

Web 窗体只是用 HTML 提供用户操作,而代码隐藏文件实现各种控件的事件处理程序。
 业务层:包含各种业务规则和逻辑的实现,业务规则完成如客户帐户的验证和字符串处理这样的任务。
 数据访问层:数据访问层为业务层提供数据服务,这一层是作为解决方案文件中的 Conn 类(数据访问类,实现公共操作的一个通用类)实现的。

1.2 新闻管理系统的功能模块简介

工作流程为:

打开该系统网站,普通浏览者能浏览和阅读新闻,能进行注册,注册后为新闻输入人员,登录进入新闻后台管理区,具有添加新闻和管理新闻的权限。

在添加新闻时,具有常规的添加新闻标题、新闻内容的选项。

管理新闻功能包括对任一条新闻内容可作相应修改,也可删除任何一条不再需要保留的新闻。

2 数据库的设计

该系统使用 SQL Server 作为应用程序的数据库,建立了有 2 张数据表:

User 表用于存储帐户的信息,News 表用于存储新闻内容。具体细节如下:

News 表设计如 1。

User 表设计如表 2。

3 Web.config 文件配置

ASP.NET 应用程序的配置功能放在 Web.config 文件中,它包含每一个具体 Web 应用程序的配置信息。在程序运行时,ASP.NET 使用分层虚拟目录结构的 Web.config 文件提供的配置信息为每个唯一的 URL 资源计算一组配置设置,然后缓存结果配置设置,以供所有页面对资源的请求使用。

Web.config 文件对于访问站点的用户来说是不可见的,而且也是不可访问的,它基于 XML,每个配置文件都包含 XML 标记和子标记的嵌套层次结构。

新闻管理系统的 Web.config 文件中的配置:

```
<connectionStrings>
  <add name="database" connectionString="Data Source=.\SQLEXPRESS; AttachDbFilename=|DataDirectory|\database.mdf;uid=sa;pwd=sa"
    providerName="System.Data.SqlClient" />
</connectionStrings>
```

通过连接字符串“connectionStrings”配置数据库连接的相关信息,将 SQL 数据库文件 database.mdf 用相对路径的方式设置为字符串“database”在程序代码中进行连接引用。

4 配置发布主页

首先要创建虚拟目录,新建虚拟目录的步骤如下:

- 1) 打开 IIS(Internet 信息服务),在默认网站上右击,在快捷菜单中选择新建—>虚拟目录;
- 2) 在弹出的虚拟目录创建向导对话框里点击“下一步”;
- 3) 输入虚拟目录别名;
- 4) 选择网站内容所在目录路径;
- 5) 设置“访问”权限。

在所建虚拟目录上右击,在弹出菜单中选择浏览,或在浏览器里输入相应网址,都可运行浏览程序。虚拟目录不在目录列表中显示(也称为 WWW 服务的“目录浏览”),要访问虚拟目录,用户必须知道虚拟目录的别名并在浏览器中键入 URL。对于 WWW 服务,也可以在 HTML 页面中创建链接。发布网站的时候,把程序首页的名字(index.aspx 或 default.aspx)写在 IIS 当中就可以访问了。

5 系统各项功能及其关键代码

5.1 通用类代码

在所有相关页面显示文件和 Web 用户控件的代码隐藏文件.aspx.cs 或.ascx.cs 中,通过对数据访问类的引用(以创建类实例方式),调用数据访问类里的相应函数完成相关功能,如建立数据库连接、打开和关闭数据库、对数据库执行查询、插入、删除和更新操作等。

此处列举 AdoSql 类(数据访问类)中关键函数的代码:

```
public class AdoSql
{ //执行 SQL 读取语句
  public DataTable ExecuteSqlRead(string SqlString)
  { //建立数据库连接字符串
    using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["database"].ConnectionString))
    { //打开数据库连接并执行查询命令,查询结果返回 DataTable 类型
      conn.Open();
      SqlCommand command = new SqlCommand(SqlString, conn);
```

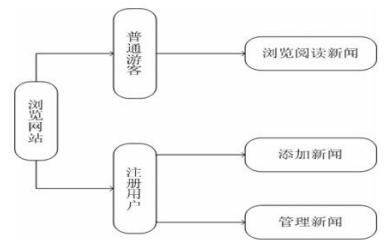


图 2 系统功能模块图示

表 1 新闻信息表

列名	数据类型	注释
newsid	int	新闻 id
newstitle	nvarchar (50)	新闻标题
newsauthor	nvarchar (20)	新闻作者
newscontent	text	新闻正文
newstime	datetime	发表时间
newscount	int	阅读次数

表 2 用户数据表

列名	数据类型	注释
username	varchar (20)	用户名
password	varchar (20)	用户密码

```

        SqlDataAdapter adapter = new SqlDataAdapter(command);
        DataSet dataSet = new DataSet();
        //通过 SqlDataAdapter 将数据填充到 DataSet 中
        adapter.Fill(dataSet);
        return dataSet.Tables[0];
    }
}
//执行 SQL 写入语句
public static Boolean ExecuteSqlWrite(string SqlString)
{ //建立数据库连接字符串
    using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["database"].ConnectionString))
    { //打开数据库连接并执行查询
        conn.Open();
        SqlCommand command = new SqlCommand(SqlString, conn);
        command.ExecuteNonQuery();
    } return true;
}
}
}

```

5.2 用户注册代码

该系统的内容是需要靠众多用户来丰富充实的,而用户注册功能提供了给用户注册和发布新闻的作用。在首页里只需点击注册按钮即可注册,注册页面里注册按钮的单击事件处理程序如下:

```

string user = "";
string pwd = "";
user = txtUserName.Text.Trim();//获取文本框用户名
pwd = txtPassword.Text.Trim();//获取文本框用户密码
string SqlString = "select username from [user] where username='" + user.Replace("'", "") + "'";
using (SqlConnection conn = new SqlConnection(ConfigurationManager.ConnectionStrings["database"].ConnectionString))
{ //建立连接并执行查询
    conn.Open();
    SqlCommand command = new SqlCommand(SqlString, conn);
    //查找当前数据库中是否存在此用户名
    if (command.ExecuteScalar() == null) //用户名未注册
    { //向数据库中插入此条记录
        string SqlString2 = "insert into [user] values ('" + user + "','" + pwd + "')";
        SqlCommand command2 = new SqlCommand(SqlString2, conn);
        int i = command2.ExecuteNonQuery();
        Session["UserSession"] = user;
        lblMsg.Text = "注册成功"; //提示用户注册成功
    } else //用户名已被注册
    { lblMsg.Text = "用户名已存在,请重新注册";
    }
} //清空输入文本框
txtUserName.Text = "";
txtPassword.Text = "";
txtConfirm.Text = "";
}
}

```

在注册页面中用到了 RequiredFieldValidator(非空)和 CompareValidator(比较)验证控件,分别用于文本输入框的非空验证和验证两个输入框的值是否相等。

服务器端验证控件是 ASP.NET 控件中新产生的一种验证控件,可以在客户端直接验证用户的输入,但控件必须包含“runat=server”属性。正常情况下验证控件是不可见的,当用户输入数据错误时它们才可见。

6 结论

1) 操作简单、界面友好:完全控件式的页面布局,使得新闻的录入工作更简便,许多选项只需要点击鼠标就可以完成。对常见网站的新闻管理的各个方面:新闻录入、浏览、删除、修改等方面都大体实现,实现了网站对即时新闻的管理要求;

2) 即时可见:对新闻的处理(包括录入、修改、删除)将立即在主页的对应栏目显示出来,达到“即时发布、即时见效”的功能;

3) 系统运行应该快速、稳定、高效和可靠;

4) 在结构上应具有很好的可扩展性,便于将来的功能扩展和维护。

参考文献:

- [1] 天极网新技术研究室.ASP.NET 完全入门[M].北京:人民邮电出版社,2001:15-57.
- [2] Rohilla S.ADO.NET 专业项目实例开发[M].陈君,译.北京:中国水利水电出版社,2003:50-67.
- [3] 王华杰,李律松,孙一波.C# 数据库开发技术[M].北京:清华大学出版社,2003:44-70.
- [4] 攀建.ASP.NET+ADO.NET 项目开发实例[M].北京:清华大学出版社,2004:143-220.
- [5] Allen S.C# 数据库入门[M].郝启堂,张哲峰,译.北京:清华大学出版社,2003:10-29.
- [6] 微软公司. Microsoft .NET Framework 1.1 Class Library Reference[M].北京:清华大学出版社,2004:99-103.