



湘南學院 畢業設計（論文）

基于.NET 的本科毕业论文

管理系统的设计与实现

目 录

摘 要.....	I
Abstract	II
1 概述.....	1
1.1 国内外研究动态.....	1
1.2 课题研究意义.....	1
2 开发使用到的技术和工具.....	1
2.1 Microsoft .NET 技术.....	1
2.2 ASP.NET 简介.....	2
2.3 Visual Studio 2008 简介.....	3
3 需求分析.....	3
3.1 角色和 workflow.....	4
3.2 数据库表设计.....	5
4 系统主要问题的解决方法.....	10
4.1 InsusIOUtility.....	10
4.2 fckeditor HTML 编辑器.....	11
4.3 MD5 密码加密.....	12
4.4 用户登录.....	12
4.5 DropDownList 控件选择项绑定.....	15
4.6 论文上传.....	15
4.7 随机生成教师答辩组和学生答辩组.....	16
5 系统软件的运行界面.....	20
5.1 登录.....	20
5.2 学生模块.....	21
5.3 教师模块.....	24
5.4 管理员模块.....	28
6 系统分析总结与改进意见.....	29
参考文献.....	30
致 谢.....	31

摘 要

高校本科毕业论文工作持续时间长、涉及人员多、形成的数据材料量大，使用传统的管理方式，既容易在管理人员处形成瓶颈，又不利于学生、教师、管理人员之间沟通，而且存在纸制论文查询不便，以及对毕业论文工作过程中的监控力度不够等诸多问题。设计和开发一个本科毕业论文工作全程管理系统，以实现毕业论文工作过程的网络化跟踪，毕业论文的电子化管理，并通过网络搭建起学生、教师间的联系渠道，改变教学管理人员的信息中转站地位，对提高毕业论文工作管理效率，减轻工作人员负担极具意义，并且有利于毕业论文后期的评估和教学研究。

本文以湘南学院计算机系毕业论文工作过程为基础，首先用.NET 技术设计实现了一个本科毕业论文全程管理系统，该系统基于三层结构，实现了指导教师安排、选题提交、选题审核、在线辅导、论文提交、论文评分审核、论文查询、用户管理、公告管理、报表打印等功能。

关键字：ASP.NET; SQL Server 2005; 三层结构; 毕业论文管理系统;

Abstract

Undergraduates' graduation thesis management in colleges and universities is a hard work with long time duration, involving large numbers of staff and students, and forming a great deal of data and documents. The traditional mode of thesis management brings about a series of problems such as causing "bottleneck" for administrative personnel; communicating barriers among students, teachers and administrative personnel; paper thesis being inconvenient for search; lack of effective supervision. Thus, the development of undergraduates' thesis full management system working on net, through which we can track the whole process and realize electronic management of graduation thesis, and through which a platform for managers, teachers and students is bridged, and also the role of administrative personnel as medium between teachers and students is changed, is necessary for improving work efficiency for administrative personnel and relieving their heavy workload, as well as for later thesis evaluation and teaching research.

This thesis is written on the basis of the undergraduates' graduation thesis management work of the computer department of Xiangnan University. Firstly, the thesis gives implementation of undergraduates' graduation thesis full management system on .NET, which is based on three-layer structures, implementing the function of teacher arrangement, thesis title submitting, thesis title checking and approval, online tutorship, thesis submitting, thesis grading and marking, thesis search, users management, bulletin management and form printing.

Keywords: ASP.NET; SQL Server 2005; Three-layer Structures; Graduation thesis management system

1 概述

本课题以计算机系为依托,开发一个基于.NET 的本科毕业设计论文全程管理系统。分为学生、教师、管理员三大模块。涵盖了学生毕业论文设计流程中的大部分过程。

1.1 国内外研究动态

全世界都在以飞快的速度向前发展,教育改革也势在必行。国内高等学校的规模及办学条件都在朝着更大和更强的方向不断发展。随着网络应用的不断扩展,很多高校在管理上也日趋网络化。在教学管理方面,很多高校实现了学生档案管理和成绩管理,而对于毕业设计以及论文答辩方面,绝大多数高校的网络化管理还没有实现。但随着高等学校改革不断深入,对于毕业设计管理的要求也越来越高,为了提高办学质量,有效地加强高等学校的管理水平,这一方面的改革工作已刻不容缓,成为很多高校今后改革工作的重点。

1.2 课题研究意义

目前,许多高校的本科毕业论文管理工作相对滞后,基本采用多年来一贯的操作模式:教师命题、学生选题,经过开题、论文撰写和论文答辩确定成绩,最后由教学管理人员将论文归档保存,整个过程基本靠人工来完成。在目前各高校网络教学环境成熟的条件下,传统模式无疑显露出诸多不足。首先,教学管理人员成为毕业设计管理流程的核心,在该管理模式下,教学管理人员逐渐成了教师、学生交流信息的中转站,使得整个管理流程环节繁杂、效率底下。另外,纸制论文查询不便,需占用大量空间保存,以及对毕业设计过程的监控力度不够等都是传统管理模式所固有的弊端。

2 开发使用到的技术和工具

2.1 Microsoft .NET 技术

2.1.1 Microsoft .NET 技术概述

Microsoft .NET 是 Microsoft 公司推出的新一代基于互联网平台的软件开发构想,它提供了对开发基于服务软件系统的强大支持。它将.NET 框架技术和面向服务计算充分结合起来,以此来满足对复杂的异构网络环境中信息共享和数据交换的需要。

.NET 将使开发人员能够比任何 Microsoft 早期平台更为有效地利用各种技术。具体来讲，.NET 讲真正实现代码的重复利用、代码专用化、资源管理、多语言开发、安全性、部署以及管理方面的一些好处。

.NET 系列开发工具常用的有 Visual Studio 2003、Visual Studio 2005、Visual Studio 2008、Visual Studio 2010。

2.1.2 Microsoft .NET 框架

.NET 框架包括用于创建和操作新一代服务的.NET 基础结构和工具，用于建立新一代高度分布的数以百万计的.NET 组建服务，以及启用新一代智能互联网设备的.NET 设备软件。其完整的结构如图 2-1 所示：

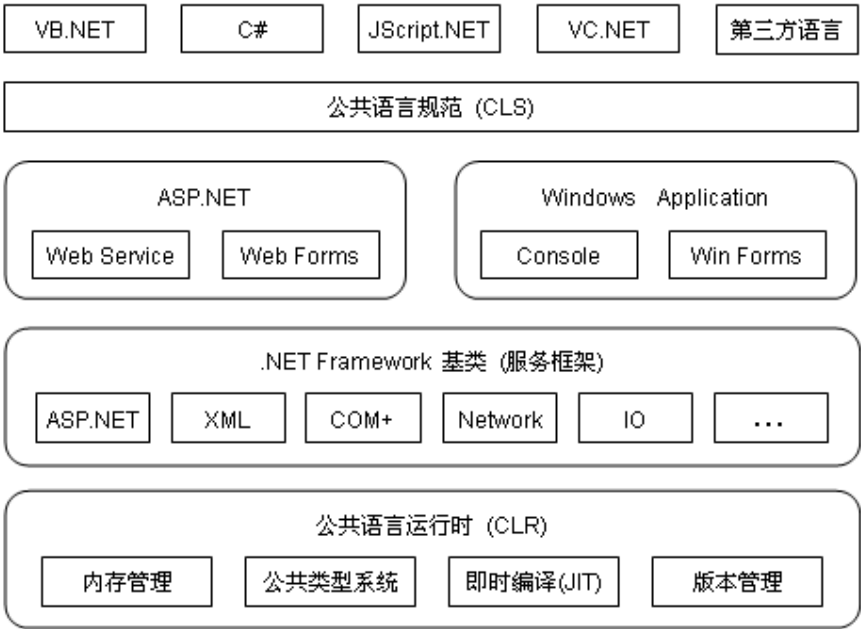


图 2-1 Microsoft .NET 框架

2.2 ASP.NET 简介

ASP.NET 提供了一个 Web 应用程序模型，该模型由一组控件和一个基本结构组成。有了它，Web 应用程序的构建变得非常容易。开发人员可以直接使用 ASP.NET 控件集（如数据源控件、数据显示控件等），这些控件运行在 Web 服务端上，它们将用户界面转换成 HTML 格式化后再发送给浏览器。在服务器上，控件负责将面向对象的编程模型呈现给 Web 开发人员，这种编程模型能够提供面向对象的编程技术拥有的丰富功能。ASP.NET 还提供一些基本结构服务（如会话状态管理

和进程循环)，这些服务进一步减少了开发人员要编写的代码量，并使用程序的可靠性得到了大幅度提高。ASP.NET 还允许开发人员将软件作为一项服务进行传送。通过使用 ASP.NET Web 服务功能，ASP.NET 开发人员只需进行简单的业务逻辑编程，而有 ASP.NET 基本结构负责通过 SOAP 传送服务。

.NET 开发平台为开发 WEB 应用程序提供了强大的技术支持，使开发变得非常便捷，高效。在微软的 .NET 框架中，ASP.NET 的 Web Forms 为在 VS.NET 集成开发环境下，创建能够利用 WWW 强大功能的 Web 应用程序带来了一种更加直接和易于使用的开发模式，ASP.NET 具有下面三方面的优势。

1.支持编译型语言

ASP.NET 提供 VB.NET 和 C# .NET 等编译型语言用于 Web 服务端开发。通过这些开发语言，程序员可以像开发普通的 Windows 程序一样来开发 Web 程序，在传统 Windows 程序中用于开发 GUI 界面的各种控件，在 Web 程序开发中也有它们相应的 Web 版本。

2.代码分离和高效可视化开发环境

ASP.NET Web 开发技术为程序员提供了一种非常好的开发模式，即 Code Behind 技术，它通过 Web 控件将程序代码与页面内容成功分离，从而使 ASP.NET 的程序结构异常清晰，开发和维护的效率也得到了很大的提高。微软提供了 Visual Studio .NET 集成环境开发 ASP.NET 应用程序，该环境不仅提供了强大的调试能力，而且还集成了“所见即所得”的 HTML 编辑器，为开发人员开发 Web 页提供了方便的图形化支持。

3.引入服务器端控件

在 VS.NET 中提供了丰富的 Web 控件，利用这些 WEB 控件，可以非常容易的开发 ASP.NET Web 程序。

2.3 Visual Studio 2008 简介

Microsoft Visual Studio 2008 是面向 Windows Vista、Office 2007、Web 2.0 的下一代开发工具，代号“Orcas”，是对 Visual Studio 2005 一次及时、全面的升级。

Visual Studio 2008 引入了 250 多个新特性，整合了对象、关系型数据、XML 的访问方式，语言更加简洁。使用 Visual Studio 2008 可以高效开发 Windows 应用。设计器中可以实时反映变更，XAML 中智能感知功能可以提高开发效率。同时 Visual Studio 2008 支持项目模板、调试器和部署程序。Visual Studio 2008 可以高效开发 Web 应用，集成了 AJAX 1.0，包含 AJAX 项目模板，它还可以高效开发 Office 应用和 Mobile 应用。

3 需求分析

3.1 角色和工作流程

论文管理系统的用户有：学生、教师、系管理员，不同用户在系统中担任不同角色，具有不同权限，要完成不同操作。不同用户的操作又有着相互依赖管理。按毕业论文工作的时间顺序，及其不同角色在不同时间阶段要参与的活动综合分析后，得出毕业论文工作过程流程图，如下所示：

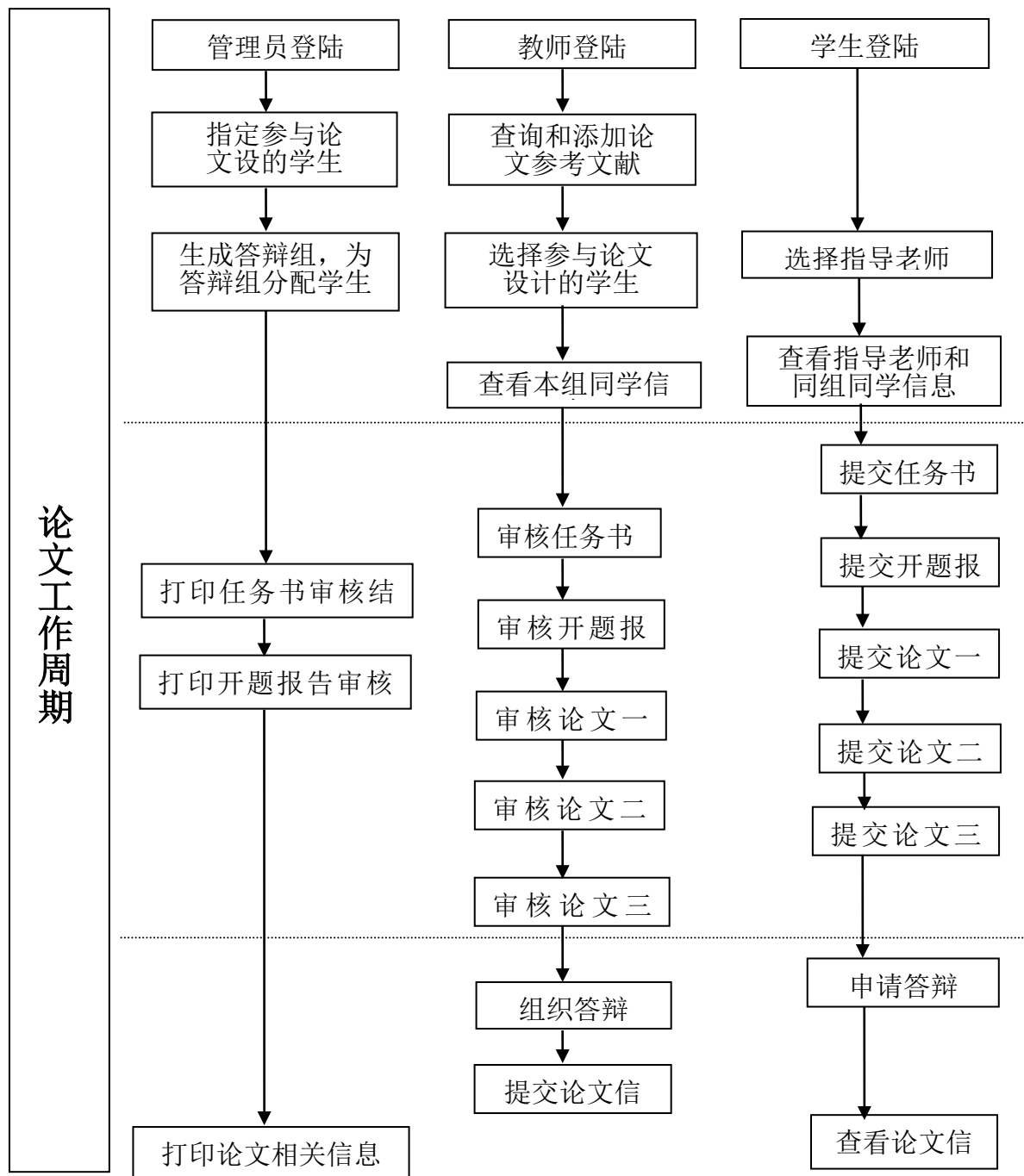


图 3-1 论文工作周期图

3.2 数据库表设计

3.2.1 数据库表设计

本系统后台数据库使用 SQL 2005 数据库，数据库名称为“xnxyjsjx”,数据库表主要包括 10 张表，如图 3-1 所示：

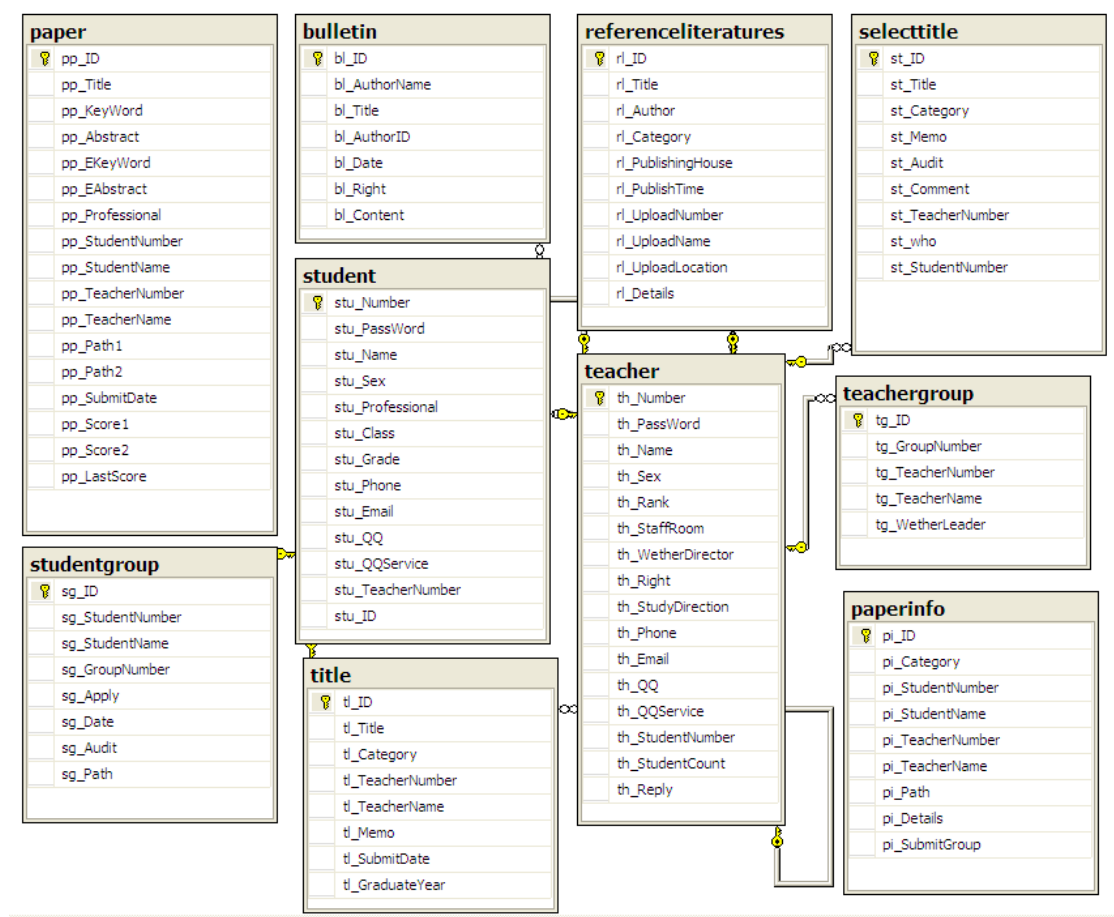


图 3-2 数据库表结构

1 系统公告表

系统公告表（bulletin）用来存放整个系统运行过程中各个关系之间发布的公告信息，其结构如下表 3-1 所示：

表 3-1 bulletin 表

列 名	数据类型	长度	说 明
bl_ID	int		主键，信息编号
bl_AuthorName	nvarchar	50	发布公告的人姓名
bl_Title	nvarchar	100	公告标题
bl_AuthorID	nvarchar	10	发布公告的人的编号

bl_Date	datetime		公告发布的日期
bl_Right	int		公告发布人的权限
bl_Content	text		公告内容

2 学生表

学生表（student）用于存放学生相关信息，其结构如下表 3-2 所示：

表 3-2 student 表

列 名	数据类型	长度	说 明
stu_Number	nvarchar	14	学生学号
stu_PassWord	nvarchar	50	学生账号
stu_Name	nvarchar	50	学生姓名
stu_Sex	nvarchar	5	学生性别
stu_Professiona l	nvarchar	20	学生专业
stu_Class	nvarchar	5	学生班级
stu_Grade	nvarchar	10	学生年级
stu_Phone	nvarchar	15	学生的手机号码
stu_Email	nvarchar	100	学生 E-mail
stu_QQ	nvarchar	20	学生 QQ 号
stu_QQService	text		QQ 聊天服务代码
stu_TeacherNumb er	nvarchar	10	外键，学生指导老师编号
stu_ID	int		外键，存放学生的论文题目编号

3 教师表

教师表（teacher）用来存放教师相关信息，其结构如表 3-3 所示：

表 3-3 teacher 表

列 名	数据类型	长度	说 明
th_Number	nvarchar	10	教师编号
th_PassWord	nvarchar	50	密码
th_Name	nvarchar	50	姓名
th_Sex	nvarchar	5	性别
th_rank	nvarchar	10	职称
th_StaffRoom	nvarchar	20	教研室

th_WetherDirect ion	bit		是否为教研室主任
th_Right	char	1	管理权限
th_StudyDirecti on	nvarchar	100	研究方向
th_Phone	nvarchar	15	电话号码
th_Email	nvarchar	100	Email
th_QQ	nvarchar	20	教师 QQ
th_StudentCount	int		带学生的人数，默认为 8
th_Reply	int		是否能够参与毕设（0，否，1，是）

4 参考文献表

参考文献表（referenceliteratures）用于存放教师在平常教学中见到的有价值的文献，其结构如表 3-4 所示：

表 3-4 referenceliterature 表

列 名	数据类型	长度	说 明
rl_ID	int		主键，参考文献编号
rl_Title	nvarchar	100	参考文献文献名
rl_Author	nvarchar	50	作者
rl_Category	nvarchar	50	参考文献类别
rl_PublishingHose	nvarchar	50	出版社
rl_PublishTime	datetime		出版时间
rl_UploadNumber	nvarchar	10	上传者编号
rl_UploadName	nvarchar	50	上传者姓名
rl_uploadLocation	text		上传位置
rl_Details	text		参考文献介绍

5 教师答辩组表

教师答辩组表（teachergroup）存放着参与答辩组的教师以及教师所在的答辩组组号，其结构如 3 表-5 所示：

表 3-5 teachergroup 表

列 名	数据类型	长度	说 明
tg_ID	int		主键，编号（无实意）

tg_GroupNumber	int		答辩组组长号
tg_TeacherNumber	nvarchar	10	教师编号
tg_TeacherName	nvarchar	50	教师姓名
tg_WetherLeader	bit		是否为答辩组组长

6 学生答辩表

学生答辩表（studentgroup）存放着学生答辩的相关信息，其结构如表 3-6 所示：

表 3-6 studentgroup 表

列 名	数据类型	长度	说 明
sg_ID	int		编号
sg_StudentNumber	nvarchar	14	学生编号
sg_StudentName	nvarchar	50	学生姓名
sg_GroupNumber	int		答辩组编号
sg_Apply	char	1	申请答辩
sg_Date	datetime		答辩时间
sg_Audit	char	1	是否允许答辩
sg_Path	text		答辩申请表存放位置

7 论文题目表

论文题目表（title）存放着学生论文选题的相关信息，其结构如表 3-7 所示

表 3-7 title 表

列 名	数据类型	长度	说 明
tl_ID	int		编号
tl_Title	nvarchar	100	论文题目
tl_Category	nvarchar	50	论文类别
tl_TeacherNumber	nvarchar	10	教师编号
tl_TeacherName	nvarchar	50	教师姓名
tl_Memo	text		论文题目说明
tl_SubmitDate	datetime		题目提交的日期
tl_GraduateYear	nvarchar	5	题目源于那届毕业生

8 论文题目副表

论文题目副表（selecttitle）用于存放学生在选择论文题目但是论文题目还未确定时的信息，其结构如表 3-8 所示：

表 3-8 selecttitle 表

列 名	数据类型	长度	说 明
st_ID	int		编号
st_Title	nvarchar	100	论文标题
st_Category	nvarchar	50	论文类别
st_Memo	text		题目相关说明
st_Audit	nvarchar	5	是否通过
st_Comment	text		不通过的原因
st_TeacherNumber	nvarchar	10	教师编号
st_Who	int		题目谁发布（0，学生，1，教师）
st_StudentNumber	nvarchar	14	学生编号

9 论文信息表

论文信息表（paper）存放着历届最终论文的相关信息，其结构如表 3-9 所示：

表 3-9 paper 表

列 名	数据类型	长度	说 明
pp_ID	int		编号
pp_Title	nvarchar	100	论文题目
pp_KeyWord	nvarchar	1000	中文关键字
pp_Abstract	text		中文摘要
pp_EKeyWord	nvarchar	1000	英文关键字
pp_EAbstract	text		英文摘要
pp_StudentNumber	nvarchar	14	学生学号
pp_StudentName	nvarchar	50	学生姓名
pp_TeacherNumber	nvarchar	10	教师编号
pp_TeacherName	nvarchar	50	教师姓名
pp_Professional	nvarchar	50	学生专业
pp_Path1	text		存储路劲一
pp_Path2	text		存储路径二
pp_SubmitDate	nvarchar	10	论文提交日期
pp_Score1	nvarchar	10	答辩分一

pp_Score2	nvarchar	10	答辩分二
-----------	----------	----	------

pp_LastScore	nvarchar	5	最终分数
--------------	----------	---	------

10 论文中途信息表

论文中途信息表（paperinfo）存放着论文设计中途中如开题报告、任务书等教师与学生交流的论文的相关信息，其结构如表 3-10 所示：

表 3-10 paperinfo 表

列 名	数据类型	长度	说 明
Pi_ID	int		编号
Pi_Category	int		类型（1，任务书，2，开题报告）
Pi_StudentNumber	nvarchar	14	学生学号
Pi_StudentName	nvarchar	50	学生姓名
Pi_TeacherNumber	nvarchar	10	教师编号
Pi_TeacherName	nvarchar	50	教师姓名
Pi_Path	text		文档存放路径
Pi_Details	text		相关说明
Pi_SubmitGroup	int		答辩组是否同意通过

4 系统主要问题的解决方法

4.1 InsusIOUtility

InsusIOUtility 是一个 dll 组件，有常用的目录操作，文件操作，上传文件，下载文件等功能。其中的 Insus.NET 是多个行业网站上经常应用的一个 IO 组件。本系统使用该工具的文件下载功能。

4.1.1 InsusIOUtility 的使用方法

首先下载 InsusIOUtility.dll 文件，然后通过“添加引用”操作将该组件添加到项目里面，最后再调用相关类库和方法来实现功能，相关代码如下：

```
using Insus.NET;

public class DownloadHelp
{
    public static void Download(string path,string filename)
    {
        string extension=Path.GetExtension(path);
        filename += extension;
        InsusIOUtility obj = new InsusIOUtility();
        if (obj.IsExistFile(path))
```

```

    {
        obj.Download(path);
    }
}
}
}

```

4.2 fckeditor HTML 编辑器

fckeditor 是一个专门使用在网页上属于开放源代码的所见即所得文字编辑器。它志于轻量化，不需要太复杂的安装步骤即可使用。它可和 PHP、JavaScript、ASP、ASP.NET、ColdFusion、Java、以及 ABAP 等不同的编程语言相结合。“FCKeditor”名称中的“FCK”是这个编辑器的作者的名字 Frederico Caldeira Knabben 的缩写。

4.2.1 fckeditor 使用

fckeditor 图片显示如图 4-1 所示：

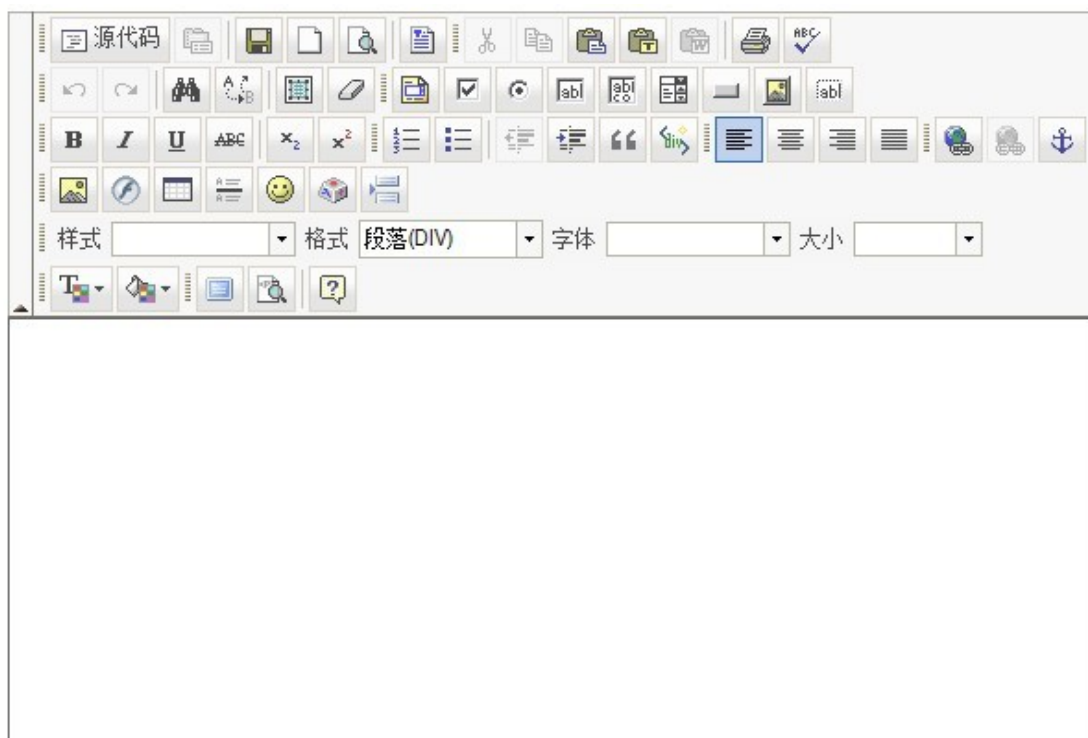


图 4-1 fckeditor 编辑器显示

4.2.2 使用方法

注册控件

```

<%@ Register Assembly="FredCK.FCKeditorV2"
Namespace="FredCK.FCKeditorV2" TagPrefix="FCKeditorV2" %>

```

添加 fckeditor 控件

```
<FCKeditorV2:FCKeditor ID="content" runat="server"
BasePath="~/editors/fckeditor/" ToolbarSet="Default"
Width="590px" Height="400px"/>
```

4.3 MD5 密码加密

MD5 加密算法是将用户输入的明文密码转换为一条长为 32 个字节的字符串保存在数据库中，但用户输入明文密码时，可以通过将改密码通过 MD5 算法加密后再同数据库中的加密后的密码匹配，若相同，则表明密码正确，反之，则用户输入的密码不正确。

本论文给 MD5 算法定义了一个 MD5 类，MD5 算法的方法采用静态函数实现，这样就可以直接调用“类名+方法名”来转换密码了，相关代码如下：

```
public class MD5
{
    public static string GetMD5Hash(string input)
    {
        System.Security.Cryptography.MD5CryptoServiceProvider
md5Hasher = new System.Security.Cryptography.MD5CryptoServiceProvider();
        byte[] data =
md5Hasher.ComputeHash(Encoding.Default.GetBytes(input));
        StringBuilder sBuilder = new StringBuilder();
        for (int i = 0; i < data.Length; i++)
        {
            sBuilder.Append(data[i].ToString("x2"));
        }
        return sBuilder.ToString();
    }
}
```

4.4 用户登录

用户登录分为三个角色，即管理员、教师、学生，通过不同的选择，登录用户可以根据自己的权限登录到相应的页面，相关代码如下：

```
protected void submit_Click(object sender, EventArgs e)
{
    string jiaose = ddl.SelectedValue.ToString();
```

```

        string str_Name = tx_name.Text.ToString().Replace("'", "''");
        string str_Password
=MD5.GetMD5Hash(tx_password.Text.ToString().Replace("'", "''"));

        DataSet ds = DAL.Login.LoginOn(returntable(jiaose, str_Name,
str_Password));

        if (ds.Tables[0] != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                if (jiaose.Equals("0") || jiaose.Equals("1"))
                {
                    Session["teachernumber"] =
ds.Tables[0].Rows[0]["th_Number"].ToString();
                    Session["teachername"] = ds.Tables[0].Rows[0]["th_Name"].ToString();
                }
                else if (jiaose.Equals("2"))
                {
                    Session["studentnumber"] =
ds.Tables[0].Rows[0]["stu_Number"].ToString();
                    Session["studentname"] = ds.Tables[0].Rows[0]["stu_Name"].ToString();
                    Session["teachernumber"] = ds.Tables[0].Rows[0]["stu_TeacherNumber"].ToS
tring();
                }

                Response.Redirect(returntable(jiaose));
            }
            else
            {
                JScript.AlertAndPageJump("登录失败，密码或用户名有误，请重新登录！",
"Default.aspx");
            }
        }
    }
}

```

```

else
{
    JScript.AlertAndPageJump("登录失败，密码或用户名有误，请重新登录！",
"Default.aspx");
}

}

```

```

private string returntable(string jiaose,string str_Name,string
str_PassWord)
{
    switch (jiaose)
    {
        case "0": return "select * from teacher where th_Number='" + str_Name
+ "' and th_PassWord='" + str_PassWord + "' and th_Right='1'";
        case "1": return "select * from teacher where th_Number='" + str_Name
+ "' and th_PassWord='" + str_PassWord + "' ";
        case "2": return "select * from student where stu_Number='" + str_Name
+ "' and stu_PassWord='" + str_PassWord + "' ";
        default: return " ";
    }
}

```

```

private string returntable(string jiaose)
{
    switch (jiaose)
    {
        case "0": return "Admin/index.aspx";
        case "1": return "Teacher/index.aspx";
        case "2": return "Student/index.aspx";
        default: return " ";
    }
}

```

4.5 DropDownList 控件选择项绑定

论文中用了许多 DropDownList 控件，主要用途是用于某些项的选择，这些选项都是绑定到数据库，动态加载，这样做的好处是，选项可以根据数据库的变化而动态变化，这样当更改某一选项时可以直接更改数据库，而不需要更改论文源码。

DropDownList 跟数据库绑定需要绑定几个属性：

DataSource: 数据源绑定，用于绑定到从数据库中查询出来的数据表。

DataTextField: 绑定用户在选择时所能看见的选项。

DataValueField: 绑定用户在选择时所能看见的选项的各个选项的选项值，程序可以通过该值很容易的从数据库中查询用户选择的选项的相关信息。

DataBind () 方法: 调用该方法才能完成 DropDownList 控件的最终绑定。

相关代码如下：

```
DataSet ds = teacher.selectOneInfo("th_StudyDirection");
DataTable table1 = ds.Tables[0];
DataRow row = table1.NewRow();
row["th_StudyDirection"] = "----请选择----";
table1.Rows.InsertAt(row, 0);
direction.DataSource = table1;
direction.DataTextField = "th_StudyDirection";
direction.DataValueField = "th_StudyDirection";
direction.DataBind();
```

4.6 论文上传

论文上传是通过 FileUpload 控件来实现的，用户通过选择自己需要上传到服务器的论文相关文档，通过触发按钮的点击事件来实现文论上传，相关代码如下：

```
public static string UploadMissionBook(FileUpload upload,string
directpath,string number,string uploadname)
{
    try
    {
        string fileextension =
Path.GetExtension(upload.PostedFile.FileName);
        if (fileextension == ".doc")
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/638054035022006120>