编号：（ ）字 号

本科生毕业设计（论文）

基于SSL的研究生申请审核及交费系统的设计与实现

王晓蕊 08113522

计算机科学与技术11-6班

题目：

姓名： 学号：

班级：



二〇一五年六月

中 国 矿 业 大 学

本 科 生 毕 业 设 计

姓 名： **王晓蕊** 学 号**： 08113522**

学 院： **计算机科学与技术学院**

专 业： **计算机科学与技术**

论文题目： **基于SSL的研究生申请审核及交费**

**系统的设计与实现**

专 题：

指导教师： **夏阳** 职 称： **教授**

2015年 6月 徐州

中国矿业大学毕业设计任务书

学院 计算机学院 专业年级 计11-6班 学生姓名 王晓蕊

**任务下达日期： 2015年 1月19日**

**毕业设计日期： 2015年1月19日 至 2015年6月3日**

**毕业设计题目：基于SSL的研究生申请审核及交费系统的设计与实现**

**毕业设计专题题目：**

**毕业设计主要内容和要求：**

本课题要求学生有一定程序设计基础，了解研究生报名收费系统的特殊需求，基于B/S利用熟悉的开发技术和数据库技术开发设计（如目前流行的.Net平台J2EE平台LAMP平台等，可以自选）。具体要求：

1）本课题以矿大国际学院为实例，分析调研研究生报名收费系统。

2）熟悉和掌握SSL安全协议及其应用。

3）利用.Net技术和数据库技术为开发工具。

论文应有正确，清晰的图表说明，理论论述严谨，软件设计合理，功能设计齐全。

院长签字： 指导教师签字：

中国矿业大学毕业设计指导教师评阅书

指导教师评语（①基础理论及基本技能的掌握；②独立解决实际问题的能力；③研究内容的理论依据和技术方法；④取得的主要成果及创新点；⑤工作态度及工作量；⑥总体评价及建议成绩；⑦存在问题；⑧是否同意答辩等）：

成 绩： 指导教师签字：

年 月 日

中国矿业大学毕业设计评阅教师评阅书

评阅教师评语（①选题的意义；②基础理论及基本技能的掌握；③综合运用所学知识解决实际问题的能力；③工作量的大小；④取得的主要成果及创新点；⑤写作的规范程度；⑥总体评价及建议成绩；⑦存在问题；⑧是否同意答辩等）：

成 绩： 评阅教师签字：

年 月 日

中国矿业大学毕业设计答辩及综合成绩

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 答 辩 情 况 | | | | | |
| 提 出 问 题 | 回 答 问 题 | | | | |
| 正 确 | 基本正确 | 有一般性错误 | 有原则性错误 | 没有  回答 |
|  |  |  |  |  |  |
| 答辩委员会评语及建议成绩：  答辩委员会主任签字：  年 月 日 | | | | | |
| 学院领导小组综合评定成绩  学院领导小组负责人：  年 月 日 | | | | | |

摘 要

教育改革的不断深化，我国的研究生教育事业得到了极大的发展。随着研究生申请人数的日益增长，传统的申请报名模式显然已经不能适应如此庞大的申请大军。因此，网上申请成为招生管理信息化的必然发展趋势。

矿大国际学院需要招收各国的来华留学生，但由于时间和空间的限制，申请过程显得相当繁琐且难于管理。本课题专门为矿大国际学院设计了基于ssl协议的研究生申请审核和交费系统，降低了报名成本，减少了人为的干扰，动态地管理各种申请信息，并及时反馈，极大程度地提高了工作效率。

本系统建立在SSL安全协议的环境下，基于B/S实现研究生申请审核及收费的功能，采用C#语言，在asp.net平台进行开发设计。本系统分为以下几个模块：学生申请模块和教师审核模块。其中，学生申请模块包括学生注册、登录及修改密码、基本信息录入、奖学金申请、上传证明材料、反馈后信息修改和在线支付等功能。教师审核模块包括审核信息并给出反馈意见、修改学生信息、查看已通过学生信息、导出所需表格、下载相关文件等功能。

关键词：.NET 平台；网上支付；SSL

ABSTRACT

The continuous development of education reform, the graduate education in China has got great development. With the increasing number of graduate students, the application of the traditional application mode is clearly not adapted to such a huge application force. Therefore, the online application is the inevitable development trend of enrollment management information.

CUMT international schools need to recruit countries came to China with the students, but due to the limitation of time and space, the application process is rather cumbersome and difficult to manage. This paper specifically for the International School of Mines design based on SSL protocol graduate students to apply for the audit and payment system, reduce the registration cost, reduce man-made interference, dynamic management of various application information, and timely feedback, greatly improves the work efficiency.

This system is based on B/S to realize the function of the students applying for auditing and charging in the environment of SSL security protocol. Using C# language, the development and design of the asp.net platform is carried out.. The system is divided into the following modules: Student Application module and teacher audit module. Among them, the students application module including students registration, login and password, basic information input, scholarship application, upload proof material, feedback information and online payment functions. The teacher audit module includes audit information and give feedback, modify student information, view the information, export required form, download the relevant documents and other functions.

Keyword：.NET Framework；Online Payment；SSL

**目 录**

[1绪论 1](#_Toc421781169)

[1.1背景与意义 1](#_Toc421781170)

[1.2国内外应用现状 2](#_Toc421781171)

[1.3论文结构 2](#_Toc421781172)

[2相关技术 4](#_Toc421781173)

[2.1关键技术 4](#_Toc421781174)

[2.1.1 .net 框架 4](#_Toc421781175)

[2.1.2 ASP.NET 5](#_Toc421781176)

[2.1.3 网页前端技术（HTML+CSS+JavaScript） 6](#_Toc421781177)

[2.1.4 ADO.NET 7](#_Toc421781178)

[2.1.5 安全套接层协议（SSL协议） 7](#_Toc421781179)

[2.2开发工具 8](#_Toc421781180)

[2.2.1 Visual Studio 8](#_Toc421781181)

[2.2.2 SQL Server 8](#_Toc421781182)

[2.2.3 IIS 9](#_Toc421781183)

[3可行性分析与需求分析 10](#_Toc421781184)

[3.1可行性分析 10](#_Toc421781185)

[3.1.1 系统概述 10](#_Toc421781186)

[3.1.2 可行性分析 10](#_Toc421781187)

[3.1.3 结论意见 10](#_Toc421781188)

[3.2 需求分析 10](#_Toc421781189)

[3.2.1 概述 10](#_Toc421781190)

[3.2.2功能性需求 11](#_Toc421781191)

[3.2.3 非功能性需求 11](#_Toc421781192)

[3.3系统数据流图 12](#_Toc421781193)

[3.3.1顶层数据流图 12](#_Toc421781194)

[3.3.2 一层数据流图 12](#_Toc421781195)

[3.3.3 二层数据流图 13](#_Toc421781196)

[3.4 系统运行需求环境 14](#_Toc421781197)

[4总体设计 16](#_Toc421781198)

[4.1功能设计 16](#_Toc421781199)

[4.1.1系统功能模块图 16](#_Toc421781200)

[4.1.2功能划分 17](#_Toc421781201)

[4.2数据库设计 19](#_Toc421781202)

[4.2.1 概述 19](#_Toc421781203)

[4.2.2 数据库表的设计 19](#_Toc421781204)

[4.2.3 数据库的逻辑设计 29](#_Toc421781205)

[4.2.4 E-R图 30](#_Toc421781206)

[5详细设计与系统实现 32](#_Toc421781207)

[5.1系统流程图 32](#_Toc421781208)

[5.2与数据库的交互 37](#_Toc421781209)

[5.3系统实现 37](#_Toc421781210)

[5.3.1注册登录模块 37](#_Toc421781211)

[5.3.2基本信息录入模块 41](#_Toc421781212)

[5.3.3上传文件与导出数据模块 43](#_Toc421781213)

[5.3.4在线支付模块 51](#_Toc421781214)

[5.3.5 跨表查看学生信息 53](#_Toc421781215)

[5.3.6 审核信息并反馈意见 57](#_Toc421781216)

[5.3.7反馈后修改信息模块 58](#_Toc421781217)

[5.4系统测试 59](#_Toc421781218)

[5.4.1系统测试目标 59](#_Toc421781219)

[5.4.2系统测试环境 60](#_Toc421781220)

[5.4.3系统测试分析 60](#_Toc421781221)

[5.4.4测试结果分析 61](#_Toc421781222)

[6总结与展望 62](#_Toc421781223)

[6.1总结 62](#_Toc421781224)

[6.2展望 62](#_Toc421781225)

[参考文献 63](#_Toc421781226)

[翻 译 66](#_Toc421781227)

[英文原文 66](#_Toc421781228)

[中文译文 69](#_Toc421781229)

[致 谢 73](#_Toc421781230)

# 1绪论

## 1.1背景与意义

1981年，我国正式建立了学位制度，同时也就意味着我国的研究生教育事业从此拉开了帷幕。从1981年至今，30多年来我国总共培养了硕士研究生约426万人，近20年来，我国硕士研究生的申请人数呈现出爆发式的增长。当然，随着我国综合实力的提高，越来越多的外籍学生来到了我国，并申请了国内各大知名高校的研究生[26]。

2014年中，有172万人报名参加了全国硕士研究生招生考试。其中，接受学历教育的外国留学生总计164,394人，而非学历留学生也达到了212,660人[45]。

调查后发现，如今各大高校中，来华留学生先自行搜集并准备好大量的证明材料，来到需要申请的大学招生办报名[27]。在招生办公室，学生需要当面申请并填写申请信息。如果学生的条件符合，那么该学生就被录取了。再由招生老师将这个学生的个人详细信息录入到学校的来华留学生信息管理系统中，这样才能管理这位来华留学生的在校生活。这样传统的报名申请模式采用的是纸质登记、人工录入的方法，工作量大、效率低、还容易出现人为错误，已明显不适应现有的管理模式[28]。况且，如今的来华留学生人数大幅增加，传统的报名申请模式已经不能满足当前的工作需求了。无论对于来华留学生，还是对于招生老师，都造成了极大的不便[41]。

而如今，随着Internet网络技术的飞速发展，互联网正无时不刻地影响着我们生活的方方面面，将信息化、网络化、智能化加入到研究生教育事业中来已经成为新时代发展的必然趋势[9]。那么，我们可以开发一个基于互联网的研究生网上申请系统，自然既可以给海外的报名者带来便利，还可以减轻招生老师的负担。利用网上申请的模式，既打破了时间的限制，海外报名者可以24小时随时报名，也打破了空间的限制，再也不需要来到学校报名了[36]。有来华留学意愿的海外申请者，可以在世界上任何一个有互联网的计算机终端进行报名申请，自行填写申请材料，而招生老师只要在网上进行审核即可确认报名[34]。这样不仅大大方便了海外来华留学生报名，也减轻了学校招生老师的管理工作。由于网络应用不断扩大，现如今使得基于Web的网上报名系统成为现实[19]，降低了报名成本，减少了人为的干扰[35]。网上申请系统的最大优点是可以动态地管理各种申请信息，既减轻了招生工作人员的工作量，又降低了责任风险,还提高了招生工作的效率[37]。

## 1.2国内外应用现状

进入21世纪以来，Internet技术飞速发展，互联网已经进入了千家万户。在国内，基于互联网的报名申请系统成为了大家的研究热点之一[8]。随着网上报名申请系统的研究逐步深入，国内越来越多的大公司大企业招聘都采用了网上申请系统，如百度、腾讯、阿里巴巴、网易等等[20]。当然，公司招聘还包括面试等流程，这比研究生申请要复杂得多，但却为教育事业上网上申请审核提供了很好的模范带头作用[38]。

现如今，网上越来越多得涌现出了全国专业技术资格考试、执业资格考试、各类型水平考试、计算机等级考试等社会化的网上考试系统，而且都还是比较成功比较成熟的[28]，这些现成的系统都为本系统的设计提供了素材。目前的全国硕士研究生报名系统，逐步实现了网络化信息化，提高了工作效率，减少了人为误差，但是只是实现了报名面试的流程，不包括支付环节。然而，报名的过程中必不可少的就是支付费用，在这过程中，会出现各种问题，如货币的兑换等。因此，开发一个带有支付系统的研究生申请审核系统是非常必要的[29]。

在国外，采用计算机网上的考试管理系统起步比国内要早得多，技术相对也比较先进，而网上报名申请系统在各大高校中运用得也非常成熟。国外排名前十的综合性大学，都有自己成熟完善的研究生网上申请系统[33]。例如哈佛、耶鲁、斯坦福以及麻省理工学院等，这些知名高校的研究生网上申请系统的登录页面除了含有专业设置、导师介绍、招生简章、招生流程等信息外[30]，还有网上填写申请信息，网上提交报名申请登记表。这样对于留学生而言，不仅能看到报考学校的相关信息，招生的相关信息，招生的流程，而且还能在网上办理申请。网页上还可以进行交互，双向的信息交流，便于学生及时地了解报考学校的信息。这是实现网上招生的起点，也是实现网上申请系统的关键。这些高校的留学申请系统都为本系统提供了很好的借鉴[31]。

## 1.3论文结构

论文论述了Web开发的相关知识和本课题完成的分析、设计及开发过程。Web开发的相关知识包括.NET框架和C#的相关语法，HTML+CSS+JavaScript网页前端设计的技巧以及SQL Server数据库的交互。对本课题的研究包括该网站的功能模块划分，数据库设计，具体实现等设计和开发思路。

论文分以下几个部分：

第一部分简要介绍所选课题，分析研究生申请审核及收费系统的研究背景和意义；

第二部分是论文用到的相关技术的介绍，例如asp.net技术；

第三部分是系统可行性研究和需求分析，从不同角度探讨网站开发的可行性，并以数据流图的形式展示系统需求分析；

第四部分是系统总体设计，介绍了系统的功能模块和数据库设计；

第五部分是系统详细设计，这部分则演示系统流程，部分代码详情和网站测试效果；

第六部分则是系统总结与展望。

论文的最后部分是参考文献、英文文献翻译和致谢。

# 2相关技术

## 2.1关键技术

### 2.1.1 .net 框架

Microsoft公司发布的.NET Framework简称.NET[1]，是支持生成和运行下一代应用程序和Web服务的内部Windows组件。

.NET框架旨在实现的目标如表2.1所示：

表2.1 .NET Framework目标

|  |  |
| --- | --- |
| 序号 | 目标 |
| １ | 提供一个一致的面向对象的编程环境。无论开发的程序是在本地存储并执行的Windows窗体程序，还是基于B/S或者C/S架构的网络程序，其编程界面风格及控件都是相似的。 |
| ２ | 提供一个将软件部署和版本控制冲突最小化的代码执行环境。 |
| ３ | 提供一个可提高代码（包括由未知的或不完全受信任的第三方创建的代码）执行安全性的代码执行环境[43]。 |
| ４ | 提供一个可消除脚本环境或解释环境的性能问题的代码执行环境。 |
| ５ | 使开发人员的经验在面对类型大不相同的应用程序（如基于 Windows 的应用程序和基于 Web 的应用程序）时保持一致。 |
| ６ | 按照行业标准生成所有通信，以确保基于 .NET Framework 的代码可与任何其他代码集成。 |

.NET Framework主要有两个组件：公共语言运行库（CLR）和.NET Framework类库（FCL）[22]。

（1）公共语言运行库（CLR）

公共语言运行库是.NET Framework的基础，可以将运行库看作一个在执行时管理代码的代理，它提供内存管理、线程管理和远程处理等核心服务，并且还强制实施严格的类型安全以及可提高安全性和可靠性的其他形式的代码准确性。[39]以运行库为目标的代码称为托管代码，而不以运行库为目标的代码称为非托管代码。

（2）.NET Framework类库（FCL）

.NET Framework的另一个主要组件是类库，它是一个综合性的面向对象的可重用类型集合，可以使用它开发多种应用程序，这些应用程序既包括传统的命令行或图形用户界面（GUI）应用程序[32]，也包括基于ASP.NET所提供的最新创新的应用程序（如Web窗体、XML和Web Services）。.NET应用程序都可以使用这些类库进行开发。.NET安装在Windows之上，支持如C#、VB.NET、VC++.NET等开发语言，也就是所谓的跨语言开发。

### 2.1.2 ASP.NET

ASP.NET是Microsoft公司.NET Framework中一套用于生成WEB应用程序和XML WEB Services的技术[5]。它建立在公共语言运行库（CLR）之上[14]，利用.NET提供的强大的类库，用较少的代码实现功能非常强大的Web应用程序。ASP.NET可以使用多种开发语言，其中C#最为常用[10]。由于C#是.NET独有的语言，优化了很多的Web开发流程[21]，从而大大提高了程序开发的效率。ASP.NET页面中使用了一种已编译的、由事件驱动的编程模型，这种模型可以提高性能并支持应用程序逻辑同用户界面相隔离。这就是ASP.NET为开发者提供的一个全新而强大的服务器控件结构[16]。ASP.NET控件主要为服务器端控件，对控件的安全性和服务器端的运行策略都进行了优化[44]，可采用页面设计与代码分离的设计方案，更好地适应项目开发中的美工与程序员开发的并行工作，提倡组件与模块化设计，每一个页、对象、HTML元素都是一个运行的组件对象[17]。

图2.1是ASP.NET页面与Web服务器交互的过程[2]。

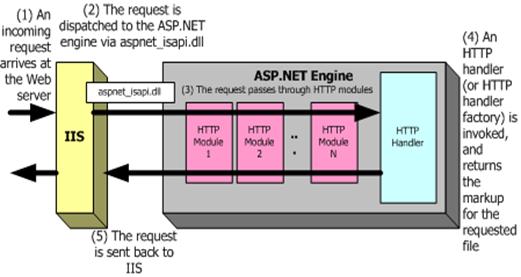


图2.1 ASP.NET与服务器交互过程

（1）用户请求页面。使用HTTP GET方法请求页面。

（2）aspnet\_isapi.dll处理这些请求。

（3）请求通过HTTP模块传送。页面发送到Web服务器。浏览器执行HTTP POST方法，页面发送回其自身。

（4）生成HTML流并返回到IIS。

（5）请求被传送回IIS。

### 2.1.3 网页前端技术（HTML+CSS+JavaScript）

在网页设计中，HTML、CSS与JavaScript并列为网页前端设计的三种基本语言。其中，HTML负责构建网页的基本结构，CSS负责设计网页的显示渲染，JavaScript负责开发网页的交互效果。

HTML

HTML是HyperText Markup Language的缩写，译为超文本标记语言。它是一种描述性的标记语言，用来描述网页文档的。它本身不能在浏览器中显示，但经过浏览器的解释和编译，能够正确得反映出HTML标记语言的内容。它通过标签来标记要显示的网页中的各个部分，如文字、图形、声音、动画、表格、视频、超链接等。其最基本的语法就是<标记符>内容</标记符>，且标记符成对使用。作为一种网页结构标识语言，HTML易学易懂，还可以制作内容丰富、结构复杂、美观大方的网页。因此，HTML是目前互联网上应用最为广泛的语言，也是构成网页文档的主要语言。

CSS

CSS 指层叠样式表 (Cascading Style Sheets)， 它可以与HTML或XHTML超文本标记语言配合来定义网页的外观。CSS是在HTML语言基础上发展而来的，是为了克服HTML网页布局标签的互相嵌套，造成网页代码臃肿杂乱的弊端。

CSS样式表可以将网页样式和内容分离，可以进行像素级得精确控制网页的布局，可以更迅速、更便捷地维护及更新大量网页，可以支持所有的字体字号样式，使浏览器成为更友好的界面。

CSS语言不需要编译，也不需要特殊的处理，只要把代码放在<style>和</style>标签之间，或者单独存储在一个text文件中，并保存为.css文件，再用<link>链接或导入到网页中即可。

JavaScript

JavaScript是一种解释型的脚本语言，可以使网页动态化和智能化。JavaScript既是一种基于对象和事件驱动并具有相对安全性的客户端脚本语言，也是一种广泛用于客户端Web开发的脚本语言，常用来给HTML网页增加动态效果。

在ASP.NET中，服务器端的程序需要一次浏览器与Web服务器的交互，一次页面的提交，需要来回传送大量的数据，而很多工作，如输入验证、删除确定、关闭窗口等可以使用JavaScript来实现。

JavaScript只需要嵌入到HTML文件中，它并不编译产生机器代码，只是由浏览器的解释器将其动态地处理成可执行的代码。所以，可以减小网页的规模，提高网页的浏览速度，还可以丰富网页的内容，如动画、声音等。

### 2.1.4 ADO.NET

ADO.NET提供了多种语言的数据库连接访问类，是一组用于与数据源进行交互的面向对象的类库[46]，主要用于访问数据库中的数据。如果连接的数据库不同（即数据源不同），那么ADO.NET采用的命名空间和类都不同[25]。

ADO.NET对象模型中有5个组件，包括Connection对象、Command对象、DataAdapter对象、DataSet对象以及DataReader对象[26]，其中DataSet对象是作为数据源之间桥梁的[42]，属于数据存储组件，其它4个组件负责建立连接和操作数据，成为数据操作组件。ADO.NET结构如图2.2：

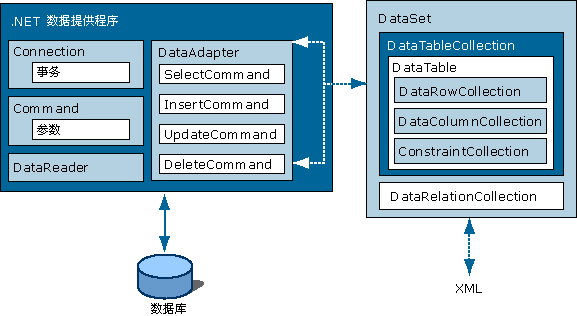


图2.2 ADO.NET结构

### 2.1.5 安全套接层协议（SSL协议）

SSL（Secure Socket Layer，安全套接层），是为网络通信提供安全及数据完整性的一种安全协议[3]。它由Netscape研发，用以保障在Internet上数据传输的安全，利用数据加密技术，确保数据在网络上的传输过程中不会被截取和窃听[6]。

SSL使用公钥加密系统[4]。

SSL协议位于TCP/IP协议模型的网络层和应用层之间。它使用TCP来提供一种可靠的端到端的安全服务，以使C/S应用之间的通信不会被攻击和窃听，并且始终会对服务器进行认证，还可以选择对客户进行认证[12]。SSL协议在应用层通过之前就已经完成了加密算法[13]，通信密钥的协商，以及服务器认证工作，在此之后，应用层协议所传送的数据都会被加密。

SSL协议可分为两层：SSL记录协议和SSL握手协议[11]，其详细结构如表2.2：

表2.2 SSL协议构成

|  |  |
| --- | --- |
| 协议 | 描述 |
| SSL记录协议（SSL Record Protocol） | 建立在可靠的传输协议（如TCP）之上，为高层协议提供对数据的封装、压缩、加密等基本功能的支持。 |
| SSL握手协议（SSL Handshake Protocol） | 建立在SSL记录协议之上，用于在实际的数据传输开始前，通信双方进行身份认证、协商加密算法、交换加密密钥等。 |

## 2.2开发工具

### 2.2.1 Visual Studio

Visual Studio是微软公司推出的一套集成开发环境，用于生成ASP.NET Web应用程序、桌面应用程序、移动应用程序和Windows应用商店等应用程序。该开发环境集成了Visual Basic、Visual C#、Visual F#、Visual C++和JavaScript等多种程序开发语言。这些语言都使用相同的集成开发环境（IDE），这样可以实现工具共享，并能够轻松地创建混合语言解决方案。而且这些语言都基于.NET 框架技术，从而可简化ASP.NET Web应用程序和XML Web服务的开发。

Visual Studio发展的版本如图2.3所示：



图2.3 Visual Studio版本

### 2.2.2 SQL Server

SQL Server 2008是微软公司推出的数据库管理系统[7]，相比之前的SQL Server版本，它有了很大的改进[49]，已不再是传统意义上的数据库管理系统，而是成为了整合数据库、商业智能、报表服务[50]、分析服务等多种技术的数据管理与分析平台[15]。

SQL Server与其他厂商的数据库产品在数据存储能力、并行访问能力、安全管理等关键性指标上并没有太大的差别[40]，但在多功能集成、操作速度、数据仓库构建、数据挖掘等方面具有较强的优势[23][51]。

### 2.2.3 IIS

IIS（Internet Information Server，互联网信息服务）是一种Web（网页）服务组件，其中包括Web服务器、FTP服务器、NNTP服务器和SMTP服务器，分别用于网页浏览、文件传输、新闻服务和邮件发送等方面，它使得在网络（包括互联网和局域网）上发布信息成了一件很容易的事[47]。

IIS支持一些有趣的东西，像有编辑环境的界面（FRONTPAGE）、有全文检索功能的（INDEX SERVER）、有多媒体功能的（NET SHOW）。其次,IIS是随Windows NT Server 4.0一起提供的文件和应用程序服务器，是在Windows NT Server上建立Internet服务器的基本组件[48]。它与Windows NT Server完全集成，允许使用Windows NT Server内置的安全性以及NTFS文件系统建立强大灵活的Internet/Intranet站点。

# 3可行性分析与需求分析

## 3.1可行性分析

### 3.1.1 系统概述

浏览了很多网上申请系统，借鉴了很多国外的留学生申请网站，以及国内的很多的考试中心组织的海外考试项目都采用了网上报名系统，比如雅思、托福、GRE等，使我获益匪浅。然而国内的研究生报名系统还不够完善，缺少网上支付模块，并且还没有完全性地投入使用。但我们可以借鉴这些尚未成熟的网上申请审核系统，学习模仿申请审核流程，为设计本系统打下坚实的基础。

### 3.1.2 可行性分析

经济上，本系统的开发所选择的工具：Visual Studio 2012及IIS服务器组件和SQL Server2008数据库都是免费的，大可不必担心系统的开发费用的问题。

技术上，.NET平台现有技术已经足够满足本系统的功能设计需求。根据平台特点，进行合理的架构设计，各个平台可以设计开发出功能完善、系统稳定的网站。

法律上，本系统只为本校国际学院设计，用于学生信息的采集和管理，不会牵扯到法律问题。

### 3.1.3 结论意见

基于ssl协议的研究生申请审核及交费系统可立即进行设计与实现。

## 3.2 需求分析

### 3.2.1 概述

本系统为研究生申请审核及交费系统，其中包括了研究生申请功能，网上交费的功能以及教师审核功能。除此之外还要将正确性、灵活性、安全性以及交互性等非功能性需求作为系统实现的一个考量。

### 3.2.2功能性需求

（1）注册登录功能

本系统中无论是学生申请人还是管理员都包含登录功能。申请人只有通过注册后才能使用本系统，继而才有基本信息录入、奖学金申请、上传证明材料、接收反馈信息并修改和在线支付的功能。教师也只有登录后台管理系统后才能查看申请学生信息，审核学生信息，导出或下载相关文件。

（2）基本信息录入功能

申请人申请就读研究生需要录入相关的学生信息，其中，包括学生信息、家庭信息、学习计划、受教育信息、工作经验、资金支持和语言能力等。

（3）上传及下载文件功能

申请就读研究生过程采集信息需要上传部分文件，如护照信息；在奖学金申请过程中需上传获奖证书等文件；而教师查看已通过学生时可导出相关文件，如申请表、录取通知书等。

（4）在线支付功能

申请人通过审核后，将进行在线支付相关费用，包括体检费、学费、住宿费等。

（5）审核学生信息并反馈功能

教师登录系统后，需对所有申请人的所有信息进行查看和审核，并反馈意见。

### 3.2.3 非功能性需求

（1）灵活性：系统中1级教师（即为管理员）具有最高权限，具有增删教师用户、修改权限、直接修改学生信息等功能，这使得本系统灵活可修改。

（2）正确性：系统能够正确运行。

（3）可操作性：网站的操作应简单且无争议，为用户提供友好的界面，使用户具有良好的体验。

（4）可用性：确保网站具有长期的使用能力；在日常操作中保持数据库的一致性，以维持系统的正常运行。

（5）安全性：作为一个具有申请报名及支付功能的系统，安全性是十分重要的。本系统对用户和管理员的登录进行必要的身份验证，尤其注重控制管理员权限及用户的相关权限，保证系统的正常运作不会受到人为因素的干扰。

（6）交互性：本系统的交互性体现在学生申请信息后，教师进行审核，可以提出反馈意见。学生可以再根据反馈意见进行信息的修改，使系统完善可靠，有不错的交互性。

## 3.3系统数据流图

Microsoft Office Visio是由微软公司推出的一款小巧但是功能齐全的绘图工具，它可以绘制Web图、流程图、数据流图等。接下来，采用Visio绘制数据流图对研究生申请审核及交费系统的结构进行描述。经过对系统的分析，合理建造三层数据模型来向大家展示系统的逻辑功能以及数据在系统内部的逻辑流向和逻辑变换过程。

### 3.3.1顶层数据流图

顶层数据流图是整个系统的一个概括，主要展示了系统的主体对象，以及它们与系统之间数据流的交互。系统顶层数据流图如图3.1所示：

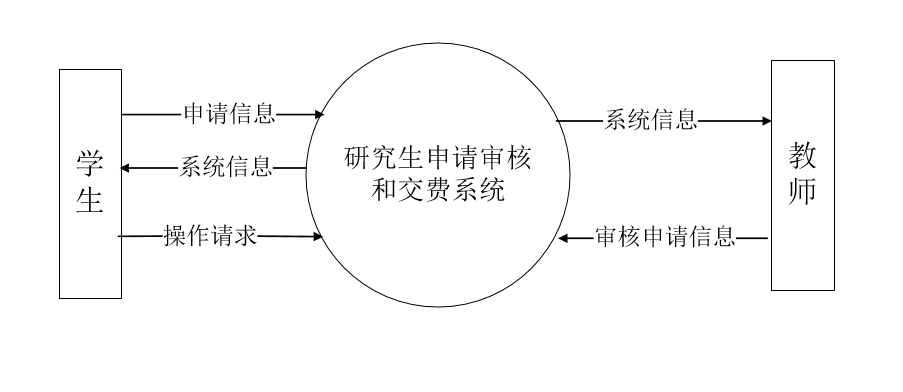


图3.1 系统顶层数据流图

### 3.3.2 一层数据流图

一层数据流图是在对顶层数据流图的各个模块进行细化和拓展的基础上而得到。各个模块都进行了细化，数据传递也比顶层数据流图显示地更加明确。系统一层数据流图如图3.2所示：

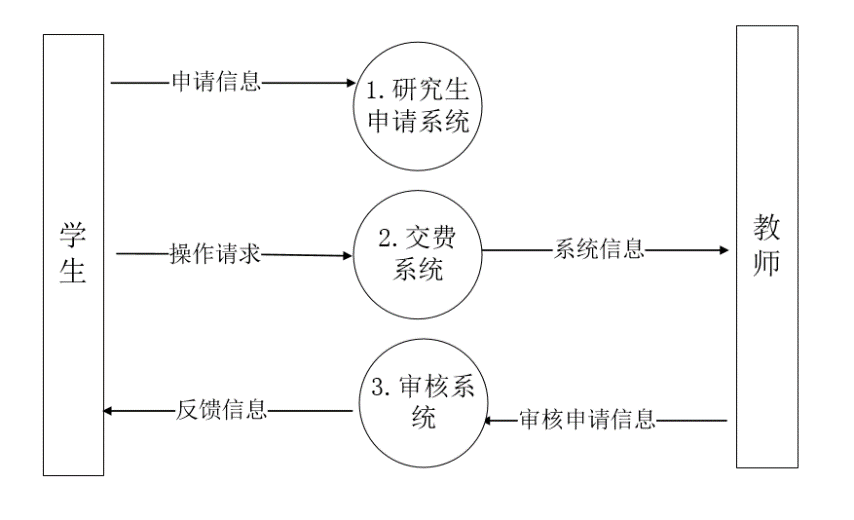


图3.2 系统一层数据流图

### 3.3.3 二层数据流图

二层数据流图是在一层数据流图各个流程的进一步细化。它将各个模块的各个流程间的数据传递都显示出来了，比一层数据流图更加仔细。

系统二层数据流图如图3.3，图3.4，图3.5所示：

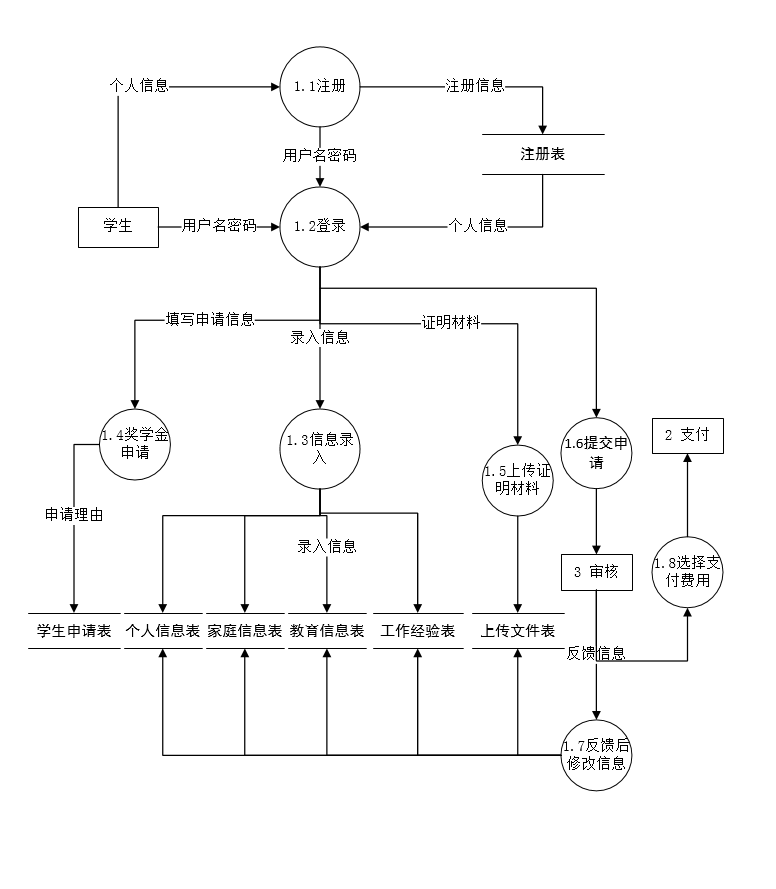


图3.3 系统申请模块二层数据流图

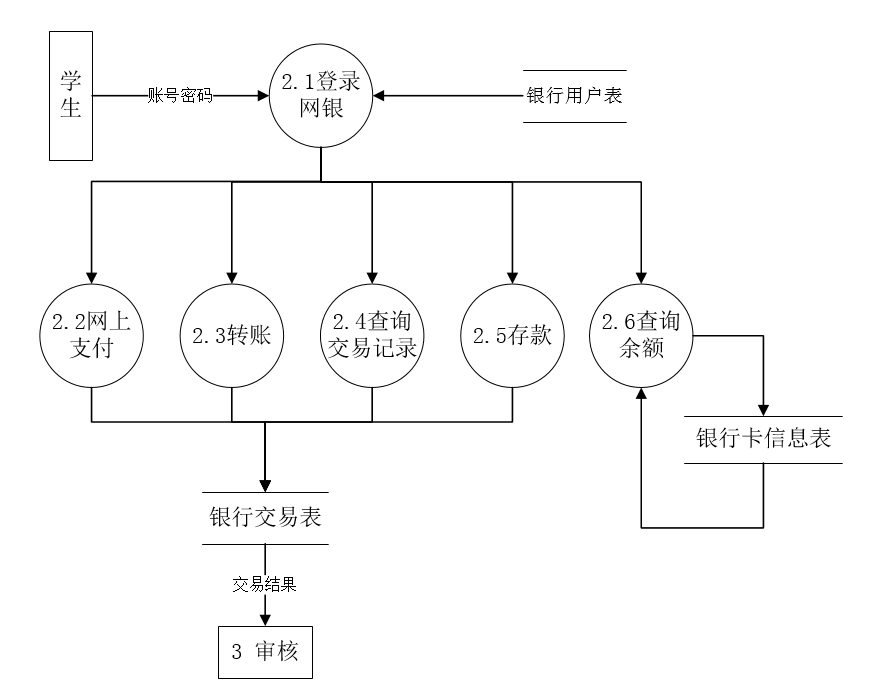


图3.4 系统支付模块二层数据流图

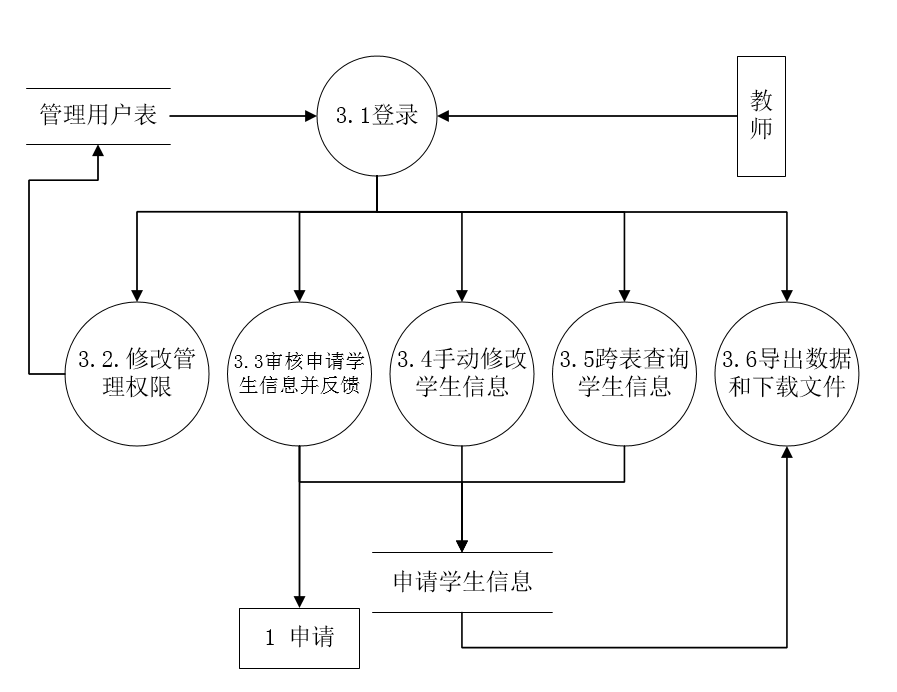


图3.5 系统审核模块二层数据流图

## 3.4 系统运行需求环境

服务器端：

硬件需求：要求内存1G以上，CPU Pentium Ⅳ 2Ghz以上。

软件需求：必须安装Windows Server服务器操作系统，IIS 8.0（或更高版本）服务器平台软件，.NET Framework 4.0，SQL Server 2008（或更高版本）数据库引擎。

客户端：

因为是B/S系统，所以无特别要求，只要通过浏览器联网状态下即可访问网站。本网站兼容IE7/8，Firefox，chrome，搜狗等浏览器。

# 4总体设计

## 4.1功能设计

### 4.1.1系统功能模块图

本系统主要划分为前台学生申请系统和后台教师管理系统。前台系统主要需实现学生的注册登录及申请研究生的功能；后台管理系统主要是教师操作系统，审核学生的申请信息。

系统功能模块图如图4.1所示：

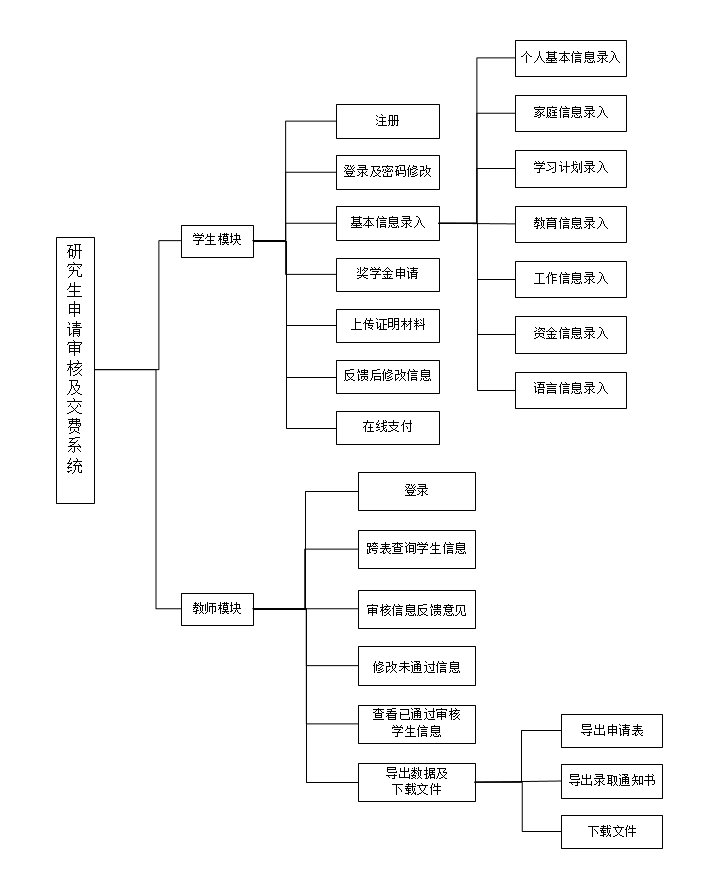


图4.1 系统功能模块图

### 4.1.2功能划分

本系统主要划分为学生模块和教师模块。学生模块主要是学生申请，包括了三类申请类型，毕业生申请，本科生申请，非学历学生申请。

学生模块的功能模块图如图4.2所示：

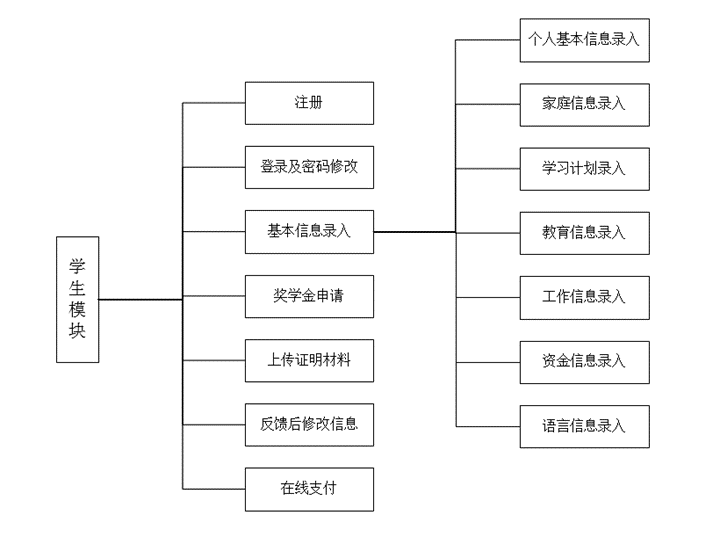


图4.2 学生模块功能模块图

（1）学生注册申请

研究生申请需填写相关信息，如姓名，电子邮箱，性别，密码，确认密码等。考虑到本系统的使用人群大多数为外国人，会有中间名的情况，注册信息中包含了中间名，但其并非是必填项。

（2）用户登录及忘记密码

对于已经注册过的申请人，可以直接进行登录操作。即填写注册时所用电子邮箱、密码及当前页刷新出的验证码，通过验证后即可进行申请研究生的下一步环节。

（3）基本信息录入

研究生申请需要录入大量信息，以供教师进行审核，其中，包括学生基础信息、家庭信息、学习计划、受教育信息、工作经验、资金支持和语言能力。

（4）奖学金申请

为鼓励学生刻苦学习，并帮助家庭困难学生顺利完成学业，我校为研究生设立了奖学金制度。申请就读研究生的同时，也可以提交申请奖学金。我校奖学金分为A、B两类，申请人可根据自己的情况填写相关信息。

（5）上传证明材料

所需上传的证明材料必须为pdf文档，且不能超过30MB。每个上传项只能选择一个文档上传，如有多个需先进行合并。上传材料包括体检表、高中或大学成绩单、学位证书、有效护照照片、汉语水平考试成绩单、有效资金支持表及其他有利于申请的相关材料。

（6）接收反馈意见并修改信息

申请人在收到教师审核的反馈信息后，需逐条对应修改信息，再次提交。

（7）在线支付相关费用

网上支付相关费用，包括体检费、学费和住宿费。采用A银行的网银交费，登录自己的银行账号进行支付所选费用。

教师模块中教师用户有权限设置，分为1级、2级和3级。1级相当于管理员，可以管理2级权限的教师，包括增加、删除2级权限的教师，还可以设置教师的权限。但是2级没有这个权限，系统会显示“您无法修改”的提示。

教师模块功能设计图如图4.3所示：

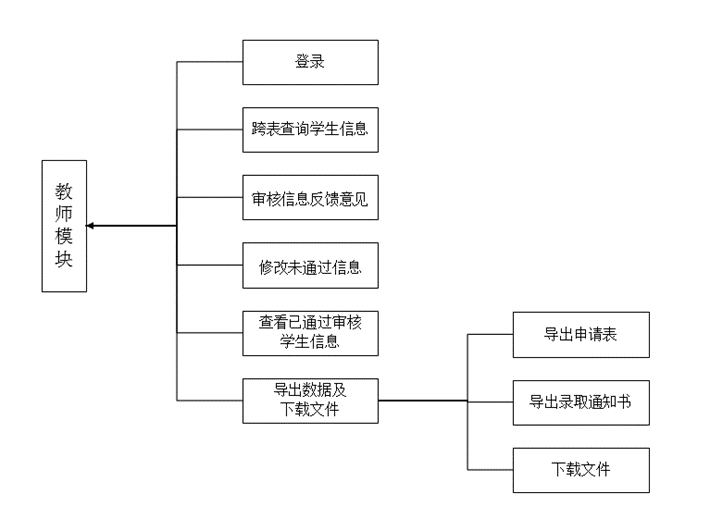


图4.3 教师模块功能关系图

（1）教师登录

教师根据自己的账号、密码及验证码通过验证登录后台管理系统。

（2）跨表查询学生信息

教师可以根据关键词分类查询学生信息，例如申请号、邮箱、性别、学生类别、学年、国籍，这样可以快速查询，方便快捷。

（3）审核申请人信息并给出反馈意见

教师登录后可查看到未审核学生信息，点击审核即可对该学生信息进行审核，查看到该学生的基本信息、家庭信息、学习计划、受教育信息、工作经验、资金支持和语言能力等，判断是否合格。针对该学生填写不恰当处，教师需在反馈意见中逐条列出，以便学生接收反馈信息后进行修改。

（4）直接修改申请人信息

学生信息有不恰当之处，存在问题较小，教师可以直接对其进行修改。不需填写反馈信息后，等学生自行修改，增强了系统的灵活性。需修改的学生名单会存放在未修改学生一列，便于教师进行修改，将学生信息进行了分类，大大增强了系统的可读性，易操作性。

（5）查看审核通过学生名单及相应信息

教师审核已通过的学生单独成一个名单。在这个名单中，教师可以进行下一步查看、导出、删除及下载文件等操作。

（6）导出审核通过学生的申请表及录取通知书

对于审核已通过的学生，本系统可以对其生成入学申请表和录取通知书。

（7）下载审核通过学生的相关文件

学生信息通过教师审核后，可以通过Excel表将其导出并打包下载，可以自行选择所需要的字段，便捷可操作。

## 4.2数据库设计

### 4.2.1 概述

系统采用Microsoft公司开发的SQL Server 2008数据库。SQL Server具有良好的用户操作界面，功能全面而强大，有很高的市场占有率和很好的发展前景。本系统采用的.NET平台，与SQL Server具有很好的兼容性，它们都是有Microsoft公司开发，通VB，C++、C#等语言结合得更好。

### 4.2.2 数据库表的设计

本系统新建的数据库名称为SACadmission和Bank。SACadmission是研究生申请审核系统的数据库，Bank是在线支付系统的数据库。

根据系统需求，需要在SACadmission数据库中创建16个表，前台学生界面13个表，后台教师管理界面3个表：

1. 学生注册表（studentRegister\_tb）
2. 学生申请信息表（studentApplyInfo\_tb）
3. 学生个人信息表（studentPersonalInfo\_tb）
4. 学生家庭信息表（studentFamily\_tb）
5. 学生高中信息表（studentHighSchool\_tb）
6. 学生大学信息表（studentCollege\_tb）
7. 学生联系表（studentCommunication\_tb）
8. 学生工作经验表（studentWorkExpericence\_tb）
9. 学生推荐人表（studentRecommender\_tb）
10. 上传文件表（uploadFile\_tb）
11. 学生管理表（studentManage\_tb）
12. 国家表（country\_tb）
13. 专业表（major\_tb）
14. 管理表（manage\_tb）
15. 管理用户表（manageUser\_tb）
16. 是否交费表（isPaid\_tb）

需要在Bank数据库创建4个表：

1. 用户信息表（User\_tb）
2. 交易记录表（Trade\_tb）
3. 银行卡信息表（Card\_tb）
4. 存款信息表（Saving\_tb）

如下展示系统所有的数据表：

表4.1 学生注册表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否  主键 |
| 标识号 | id | int | 否 | 是 |
| 邮箱 | email | nvarchar（250） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 性别 | gengder | nvarchar（10） | 否 | 否 |
| 密码 | password | nvarchar（250） | 否 | 否 |
| 姓 | familyName | nvarchar（50） | 否 | 否 |
| 中间名 | middleName | nvarchar（50） | 是 | 否 |
| 名字 | givenName | nvarchar（50） | 否 | 否 |
| 登记开始的时间 | startRegisteTime | nvarchar（50） | 否 | 否 |
| 登记结束的时间 | finishRegisteTime | nvarchar（50） | 是 | 否 |
| 学业类型 | programType | nvarchar（50） | 否 | 否 |
| 审核状态 | checkStatus | nvarchar（50） | 否 | 否 |
| 审核意见 | checkNotes | nvarchar（MAX） | 是 | 否 |

表4.2 国家表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 国家名 | name | nvarchar（MAX） | 是 | 否 |

表4.3专业表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 学院 | school | nvarchar（200） | 是 | 否 |
| 专业 | major | nvarchar（300） | 是 | 否 |
| 类型 | program | nvarchar（50） | 是 | 否 |

表4.4(1) 学生申请信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 学习计划 | studyPlan | nvarchar（50） | 否 | 否 |
| 申请专业 | applyMajor | nvarchar（150） | 否 | 否 |
| 资金支持 | fundSource | nvarchar（50） | 否 | 否 |
| 资金其他详细信息 | fundOtherDetail | nvarchar（500） | 是 | 否 |
| 是否参加汉语水平考试 | getHSK | nvarchar（50） | 是 | 否 |
| 汉语水平考试时间 | HSKTime | nvarchar（50） | 否 | 否 |
| 汉语水平考试成绩 | HSKScore | nvarchar（50） | 否 | 否 |
| 汉语水平考试阅读成绩 | HSKRead | nvarchar（50） | 是 | 否 |
| 汉语水平考试口语成绩 | HSKSpeak | nvarchar（50） | 是 | 否 |
| 汉语水平考试听力成绩 | HSKListen | nvarchar（50） | 是 | 否 |
| 汉语水平考试写作成绩 | HSKWrite | nvarchar（50） | 是 | 否 |
| 学习汉语时间 | studyChineseHour | nvarchar（50） | 是 | 否 |
| 申请学习汉语 | applyStudyChinese | nvarchar（50） | 是 | 否 |
| 英语优势 | englishProficiency | nvarchar（50） | 否 | 否 |
| 其他语言 | otherLanguage | nvarchar（50） | 是 | 否 |
| 申请CUMT奖学金 | scholarshipCUMT | nvarchar（10） | 否 | 否 |
| 奖学金类型 | scholarshipType | nvarchar（50） | 是 | 否 |
| 不申请CUMT奖学金 | scholarshipCUMTwithout | nvarchar（10） | 否 | 否 |
| 获悉CUMT | knowCUMT | nvarchar（50） | 否 | 否 |
| 机构 | agency | nvarchar（50） | 是 | 否 |
| 补充信息 | addInfo | nvarchar（MAX） | 是 | 否 |
| 是否住校 | campusIntend | nvarchar（10） | 否 | 否 |
| 是否有雅思或托福证书 | getTI | nvarchar（50） | 是 | 否 |
| 雅思还是托福 | tolefAndIelts | nvarchar（50） | 是 | 否 |
| 托福考试类型 | TOEFLType | nvarchar（50） | 是 | 否 |
| 托福考试时间 | TOEFLDate | nvarchar（50） | 是 | 否 |
| 托福成绩 | TOEFLScore | nvarchar（50） | 是 | 否 |

表4.4(2) 学生申请信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 托福阅读成绩 | TOEFLRead | nvarchar（50） | 是 | 否 |
| 托福写作成绩 | TOEFLWrite | nvarchar（50） | 是 | 否 |
| 托福听力成绩 | TOEFLListen | nvarchar（50） | 是 | 否 |
| 托福口语成绩 | TOEFLSpeak | nvarchar（50） | 是 | 否 |
| 雅思考试时间 | IELTSDate | nvarchar（50） | 是 | 否 |
| 雅思成绩 | IELTSScore | nvarchar（50） | 是 | 否 |
| 雅思阅读成绩 | IELTSRead | nvarchar（50） | 是 | 否 |
| 雅思写作成绩 | IELTSWrite | nvarchar（50） | 是 | 否 |
| 雅思听力成绩 | IELTSListen | nvarchar（50） | 是 | 否 |
| 雅思口语成绩 | IELTSSpeak | nvarchar（50） | 是 | 否 |
| 申请年份 | admissionYear | nvarchar（50） | 是 | 否 |
| 申请学制 | admissionSemester | nvarchar（50） | 是 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 申请学位 | applyDegree | nvarchar（50） | 是 | 否 |
| 奖学金个人说明 | scholarshipPersonalEssay | nvarchar（MAX） | 是 | 否 |
| 个人说明陈述 | personalEssayStatements | nvarchar（500） | 是 | 否 |
| 汉语类型 | chineseLanguageProgram | nvarchar（100） | 是 | 否 |
| 无学位类型 | nonDegreeAdvancedProgram | nvarchar（50） | 是 | 否 |
| 联系导师姓名 | contactedTutorName | nvarchar（100） | 是 | 否 |
| 联系导师邮箱 | contactedTutorEmail | nvarchar（250） | 是 | 否 |
| 联系导师 | contactedTutor | nvarchar（50） | 是 | 否 |

表4.5 工作经验表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 工作单位 | workEmployer | nvarchar（50） | 否 | 否 |
| 工作开始时间 | workStartDate | nvarchar（50） | 否 | 否 |
| 工作结束时间 | workStopDate | nvarchar（50） | 否 | 否 |
| 工作职称 | workJobTitle | nvarchar（50） | 否 | 否 |
| 工作国家 | workCountry | nvarchar（50） | 否 | 否 |
| 工作城市 | workCity | nvarchar（50） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 工作邮箱 | workEmail | nvarchar（250） | 是 | 否 |

表4.6 个人信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 中文名 | chineseName | varchar（50） | 是 | 否 |
| 国籍 | nationality | nvarchar（100） | 否 | 否 |
| 出生日期 | birthDate | nvarchar（50） | 否 | 否 |
| 出生地 | birthPlace | nvarchar（100） | 否 | 否 |
| 护照编号 | passportNO | nvarchar（50） | 是 | 否 |
| 签证到期时间 | expirationDate | nvarchar（100） | 是 | 否 |
| 母语 | nativeLanguage | nvarchar（50） | 否 | 否 |
| 宗教 | religion | nvarchar（50） | 是 | 否 |
| 中国居住地 | resideInChina | nvarchar（100） | 是 | 否 |
| 护照类型 | VISAstatus | nvarchar（10） | 是 | 否 |
| 婚姻状态 | maritalStatus | nvarchar（10） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 是否有孩子 | haveAnyChildren | nvarchar（50） | 是 | 否 |
| 孩子带来中国 | bringChildrenChina | nvarchar（50） | 是 | 否 |

表4.7 家庭信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 父母婚姻状态 | parentMaritalStatus | nvarchar（50） | 是 | 否 |
| 父母1类型 | parent1Type | nvarchar（50） | 是 | 否 |
| 父母1存活 | parent1Living | nvarchar（50） | 是 | 否 |
| 父母1称谓 | parent1Prefix | nvarchar（50） | 是 | 否 |
| 父母1全名 | parent1FullName | nvarchar（250） | 是 | 否 |
| 父母1职业 | parent1Occupation | nvarchar（250） | 是 | 否 |
| 父母1雇佣状态 | parent1EmploymentStatus | nvarchar（50） | 是 | 否 |
| 父母1职称 | parent1Position | nvarchar（100） | 是 | 否 |
| 父母2类型 | parent2Type | nvarchar（50） | 是 | 否 |
| 父母2存活 | parent2Living | nvarchar（50） | 是 | 否 |
| 父母2称谓 | parent2Prefix | nvarchar（50） | 是 | 否 |
| 父母2全名 | parent2FullName | nvarchar（250） | 是 | 否 |
| 父母2职业 | parent2Occupation | nvarchar（250） | 是 | 否 |
| 父母2雇佣状态 | parent2EmploymentStatus | nvarchar（50） | 是 | 否 |
| 父母2职称 | parent2Position | nvarchar（100） | 是 | 否 |
| 兄弟姊妹数量 | siblingNumber | nvarchar（50） | 是 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |

表4.8 高中信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 高中名 | highSchoolName | nvarchar（500） | 否 | 否 |
| 高中城市 | highSchoolCity | nvarchar（50） | 否 | 否 |
| 高中国家 | highSchoolCountry | nvarchar（50） | 否 | 否 |
| 高中开始时间 | highSchoolStartDate | nvarchar（50） | 否 | 否 |
| 高中毕业时间 | highSchoolGraduateDate | nvarchar（50） | 否 | 否 |
| 高中辍学 | highSchoolDisruption | nvarchar（10） | 否 | 否 |
| 高中辍学原因 | highSchoolDisruptionReason | nvarchar（MAX） | 是 | 否 |
| 高中辍学时间 | highSchoolDisruptionDate | nvarchar（50） | 是 | 否 |
| 高中时长 | highSchoolSchoolLength | nvarchar（10） | 否 | 否 |
| 高中成绩类型 | highSchoolScoreType | nvarchar（50） | 否 | 否 |
| 高中证书 | highSchoolCertificate | nvarchar（50） | 否 | 否 |
| 高中数学成绩 | highSchoolMathematicsScore | nvarchar（50） | 是 | 否 |
| 高中物理成绩 | highSchoolPhysicsScore | nvarchar（50） | 是 | 否 |
| 高中化学成绩 | highSchoolChemistryScore | nvarchar（50） | 是 | 否 |
| 高中语文成绩 | highSchoolLanguageScore | nvarchar（50） | 是 | 否 |
| 高中学校联系 | highSchoolSchoolContacts | nvarchar（500） | 是 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |

表4.9管理表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 审核时间 | checkTime | nvarchar（50） | 是 | 否 |
| 申请号 | applicationNO | nvarchar（100） | 否 | 否 |
| 表格名 | tableName | nvarchar（50） | 是 | 否 |
| 列名 | columnName | nvarchar（50） | 是 | 否 |
| 是否接受 | accepted | nvarchar（50） | 否 | 否 |
| 审核结论 | checkMark | nvarchar（MAX） | 否 | 否 |

表4.10 大学信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 大学名称 | collegeName | nvarchar（500） | 否 | 否 |
| 大学城市 | collegeCity | nvarchar（50） | 否 | 否 |
| 大学国家 | collegeCountry | nvarchar（50） | 否 | 否 |
| 大学联系方式 | collegeSchoolSchoolContacts | nvarchar（500） | 是 | 否 |
| 大学开始时间 | collegeStartDate | nvarchar（50） | 否 | 否 |
| 大学毕业时间 | collegeGraduateDate | nvarchar（50） | 否 | 否 |
| 有无辍学 | collegeDisruption | nvarchar（10） | 否 | 否 |
| 辍学原因 | collegeDisruptionReason | nvarchar（MAX） | 是 | 否 |
| 辍学日期 | collegeDisruptionDate | nvarchar（50） | 是 | 否 |
| 大学分数类型 | collegeScoreType | nvarchar（50） | 否 | 否 |
| 大学证书 | collegeCertificate | nvarchar（50） | 否 | 否 |
| 大学成绩排行 | collegeCollegeRank | nvarchar（50） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 大学年限 | collegeSchoolLength | nvarchar（50） | 否 | 否 |
| 学位名称 | degreeTitle | nvarchar（250） | 是 | 否 |
| 总平均分 | totalGPA | nvarchar（50） | 是 | 否 |
| 加权百分比 | weightedPercentages | nvarchar（50） | 是 | 否 |
| 本科专业 | collegeMajor | nvarchar（50） | 是 | 否 |

表4.11 交费状态表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 是否交费 | isPaid | nvarchar（10） | 是 | 否 |
| 交费类型 | paidType | nvarchar（50） | 是 | 否 |
| 交费金额 | paidMoney | nvarchar（50） | 是 | 否 |
| 交费时间 | paidDate | nvarchar（50） | 是 | 否 |

表4.12 管理用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 管理员姓名 | manageName | nvarchar（50） | 是 | 否 |
| 管理员密码 | managePassword | nvarchar（50） | 是 | 否 |
| 管理员级别 | manageLevel | nvarchar（50） | 是 | 否 |

表4.13 推荐人信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 推荐人姓名 | recomenderName | nvarchar（50） | 否 | 否 |
| 推荐人邮箱 | recomenderMail | nvarchar（50） | 否 | 否 |
| 推荐人工作 | recomenderEmployer | nvarchar（100） | 否 | 否 |
| 推荐人城市 | recomenderCity | nvarchar（50） | 否 | 否 |
| 推荐人国家 | recomenderCountry | nvarchar（50） | 否 | 否 |
| 推荐人来信邮箱 | recomenderMailTo | nvarchar（10） | 否 | 否 |
| 推荐人手机 | recomenderPhone | nvarchar（50） | 是 | 否 |
| 推荐人关系 | recomenderRelationship | nvarchar（50） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 推荐人职位 | recomenderPosition | nvarchar（50） | 是 | 否 |

表4.14 学生联系表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 住宅街道号 | homeStreetNO | nvarchar（500） | 否 | 否 |
| 住宅城市 | homeCity | nvarchar（50） | 否 | 否 |
| 住宅州 | homeState | nvarchar（50） | 否 | 否 |
| 住宅国家 | homeCountry | nvarchar（50） | 否 | 否 |
| 住宅邮政编码 | homeZipCode | nvarchar（50） | 否 | 否 |
| 宅电 | homeTel | nvarchar（50） | 是 | 否 |
| 住宅手机 | homePhone | nvarchar（50） | 是 | 否 |
| 邮件街道号 | mailStreetNO | nvarchar（500） | 否 | 否 |
| 邮件城市 | mailCity | nvarchar（50） | 否 | 否 |
| 邮件州 | mailState | nvarchar（50） | 否 | 否 |
| 邮件国家 | mailCountry | nvarchar（50） | 否 | 否 |
| 邮寄邮政编码 | mailZipCode | nvarchar（50） | 否 | 否 |
| 邮寄电话 | mailTel | nvarchar（50） | 是 | 否 |
| 邮寄手机 | mailPhone | nvarchar（50） | 是 | 否 |
| 紧急联系人 | emergencyPerson | nvarchar（50） | 否 | 否 |
| 紧急联系人关系 | emergencyRelationship | nvarchar（500） | 否 | 否 |
| 紧急联系人手机 | emergencyPhone | nvarchar（50） | 是 | 否 |
| 紧急联系人电话 | emergencyTel | nvarchar（50） | 是 | 否 |
| 紧急联系人地址 | emergencyAddress | nvarchar（500） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |

表4.15 学生管理表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 申请号 | applicationNO | nvarchar(50) | 否 | 否 |
| 是否完成注册 | isRegister | nvarchar(MAX) | 是 | 否 |
| 注册日期 | registerDate | nvarchar(MAX) | 是 | 否 |
| 房间号 | roomNum | nvarchar(MAX) | 是 | 否 |
| 保险 | insurance | nvarchar(MAX) | 是 | 否 |
| 是否毕业生 | isGraduate | nvarchar(MAX) | 是 | 否 |
| 毕业年份 | graduateYear | nvarchar(MAX) | 是 | 否 |
| 奖励 | rewards | nvarchar(MAX) | 是 | 否 |
| 惩罚 | punishment | nvarchar(MAX) | 是 | 否 |
| 奖学金 | tuition | nvarchar(MAX) | 是 | 否 |

表4.16 上传文件表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 文件目录 | fileCategory | nvarchar（100） | 否 | 否 |
| 文件名 | fileName | nvarchar（500） | 否 | 否 |
| 文件路径 | filePath | nvarchar（500） | 否 | 否 |
| 文件类型 | fileType | nvarchar（50） | 否 | 否 |
| 上传时间 | uploadTime | nvarchar（50） | 否 | 否 |
| 申请号 | applicationNO | nvarchar（50） | 否 | 否 |
| 类型 | programType | nvarchar（50） | 否 | 否 |
| 是否接受 | accept | nvarchar（50） | 否 | 否 |
| 文件大小 | fileSize | nvarchar（50） | 是 | 否 |
| 文件备注 | fileMark | nvarchar（MAX） | 是 | 否 |

表4.17 银行用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 卡号 | cardID | int | 否 | 否 |
| 真实姓名 | realName | nvarchar（50） | 是 | 否 |
| 密码 | password | nvarchar（MAX） | 否 | 否 |
| 性别 | gender | nvarchar（50） | 否 | 否 |
| 电话 | telephone | nvarchar（50） | 否 | 否 |
| 地址 | address | nvarchar（MAX） | 是 | 否 |

表4.18 存款信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 存款标识 | savingID | int | 否 | 是 |
| 存款名称 | savingName | nvarchar（MAX） | 否 | 否 |
| 存款描述 | description | text | 是 | 否 |

表4.19 银行交易表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 标识号 | id | int | 否 | 是 |
| 付款卡号 | paidCardID | int | 否 | 否 |
| 收款卡号 | receivedCardID | int | 否 | 否 |
| 交易日期 | tradeDate | nvarchar（50） | 否 | 否 |
| 交易金额 | tradeMoney | money | 否 | 否 |
| 交易类型 | tradeType | nchar（10） | 否 | 否 |
| 备注 | remark | text | 是 | 否 |

表4.20 银行卡信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 描述 | 字段名 | 数据类型 | 允许null值 | 是否主键 |
| 卡号 | cardID | int | 否 | 是 |
| 开户时间 | openDate | nvarchar（50） | 是 | 否 |
| 初始金额 | openMoney | nvarchar（MAX） | 否 | 否 |
| 余额 | Account | nvarchar（MAX） | 否 | 否 |
| 交易密码 | tradePwd | nvarchar（MAX） | 是 | 否 |
| 存款号 | savingID | int | 否 | 否 |

### 4.2.3 数据库的逻辑设计

数据表之间的关系图如图4.4，图4.5所示：

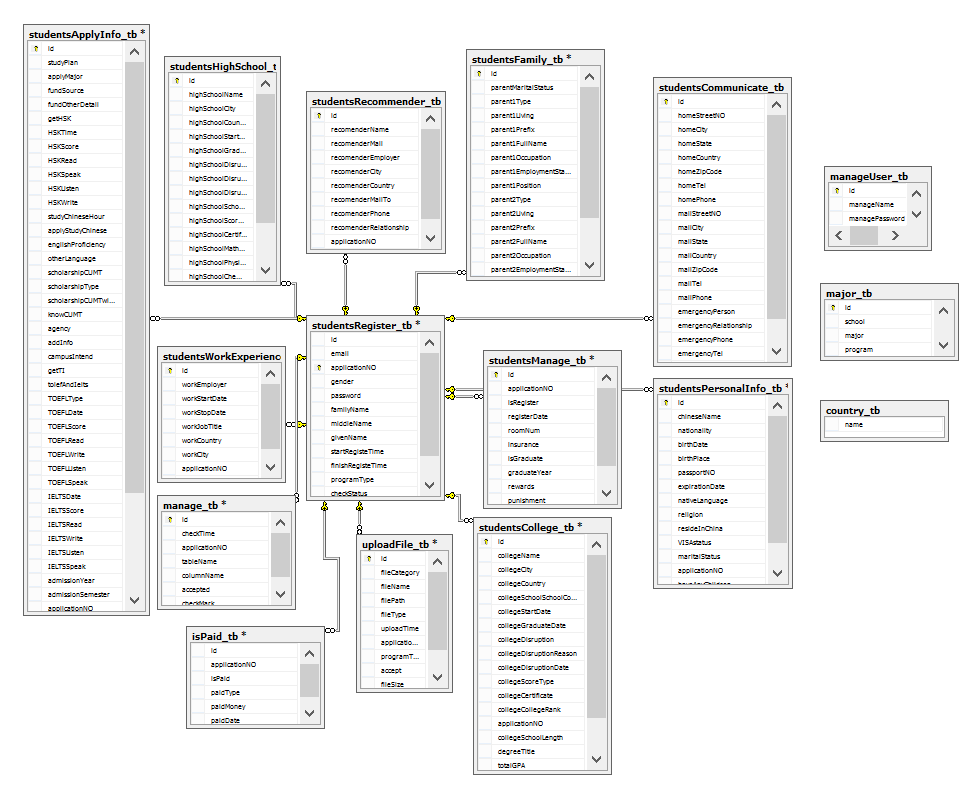


图4.4 SACADmission数据库关系图

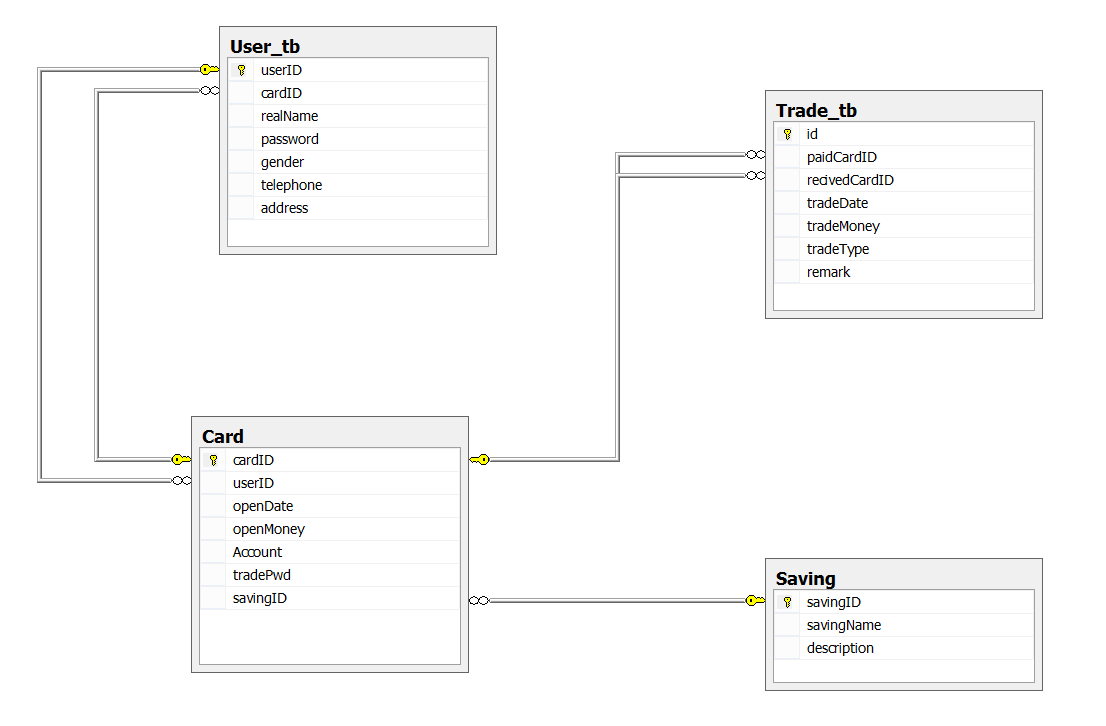


图4.5 Bank数据库关系图

### 4.2.4 E-R图

为了展示本系统的数据库设计，采用了E-R图进行描述。但考虑到本系统的复杂性，这里采用支付模块的虚拟银行系统作为代表。Bank数据库的E-R图如图4.6，图4.7，图4.8，图4.9，图4.10所示：

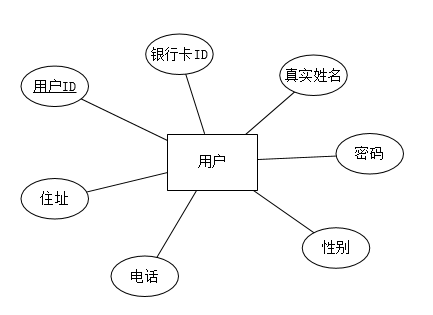


图4.6 用户实体图

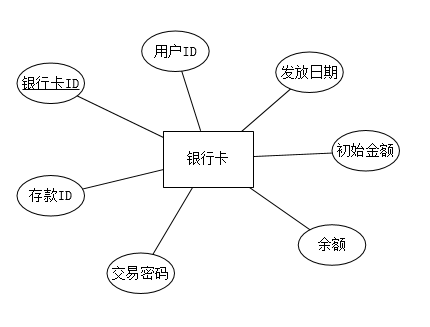


图4.7 银行卡实体图

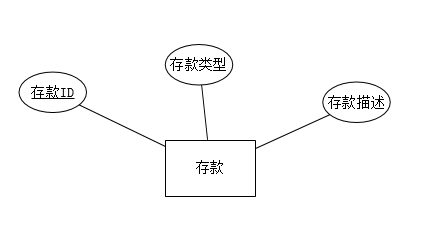


图4.8 存款实体图

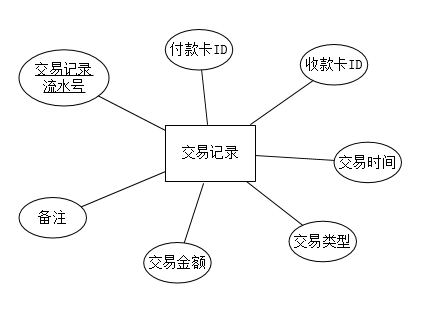


图4.9 交易信息实体图

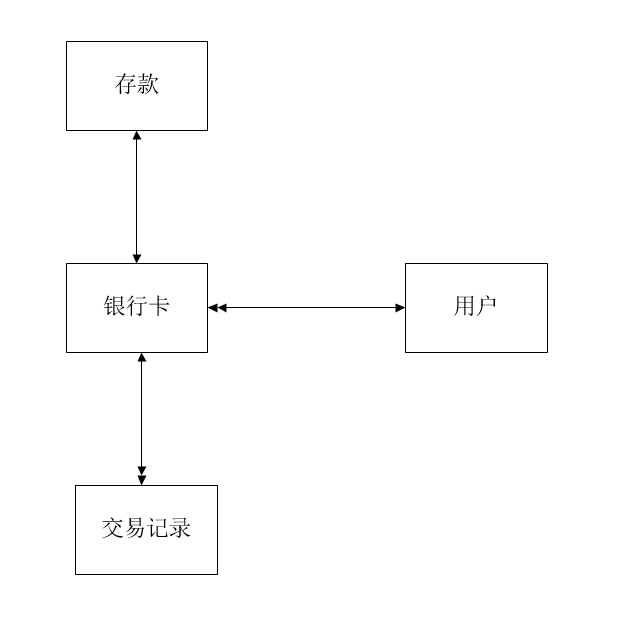


图4.10 Bank数据库E-R图

# 5详细设计与系统实现

## 5.1系统流程图

本系统用户主要分为两类，学生和教师，分别对应使用前台申请系统和后台管理系统。学生用户流程图如图5.1所示：

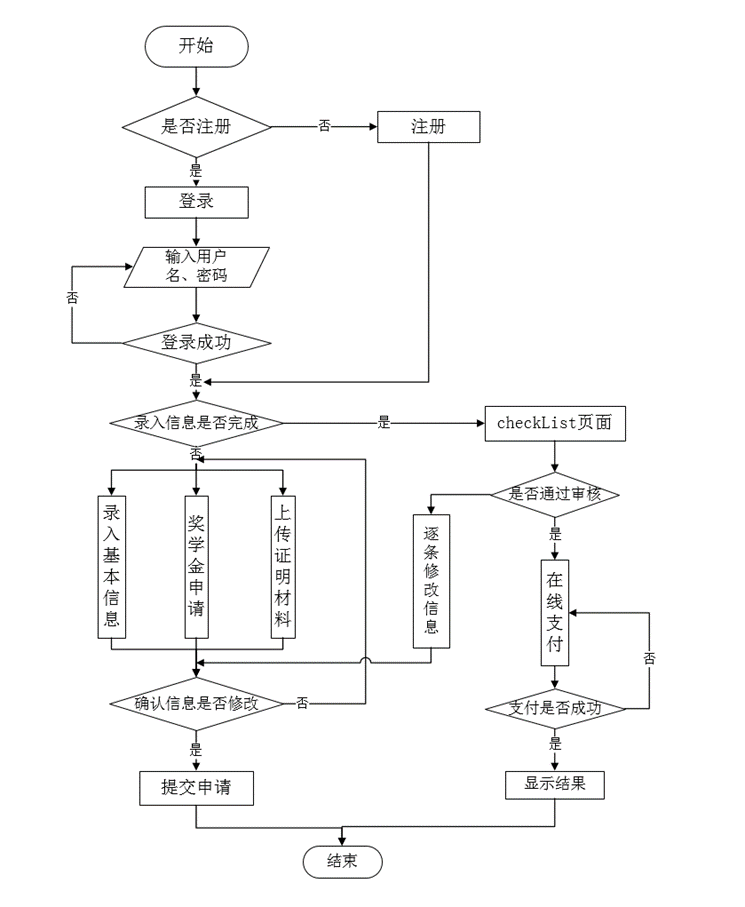


图5.1 学生用户系统流程图

初次使用本系统的学生用户需要进行注册，第一步选择自己的申请类型，如毕业生申请，接着需要填写学生的姓、名字、邮箱、性别、密码及密码确认。验证邮箱后即完成了本系统的注册，即可进入本系统。而对于已经注册过的学生，可以直接进行登录操作，即填写邮箱、密码和验证码，验证后进入本系统。本系统需要填写一些信息，其中包括个人信息、家庭信息、学习计划、奖学金申请、教育信息、工作经验、资金支持、语言能力等。所有信息填写完成后，点击注册提交申请。一旦确认提交，信息不得修改，而是上传后台，等待教师进行审核。经教师审核之后，本系统会反馈给学生用户一个checklist页面。如果该学生的申请信息完全符合要求，教师已经审核通过，则该学生可以进行下一步在线支付环节。如果该学生的申请信息不符合要求，教师审核拒绝，则该学生可以查看拒绝理由，申请失败。如果该学生的申请信息有小部分内容不符合要求，教师审核后给出了反馈意见，则该学生可以根据反馈意见修改申请信息，修改后再次提交审核。对于审核已通过的学生，先选择支付费用，再登录网上银行进行支付。返回后的支付结果，会显示在已通过审核学生的相关信息中。

教师用户流程图如图5.2所示：

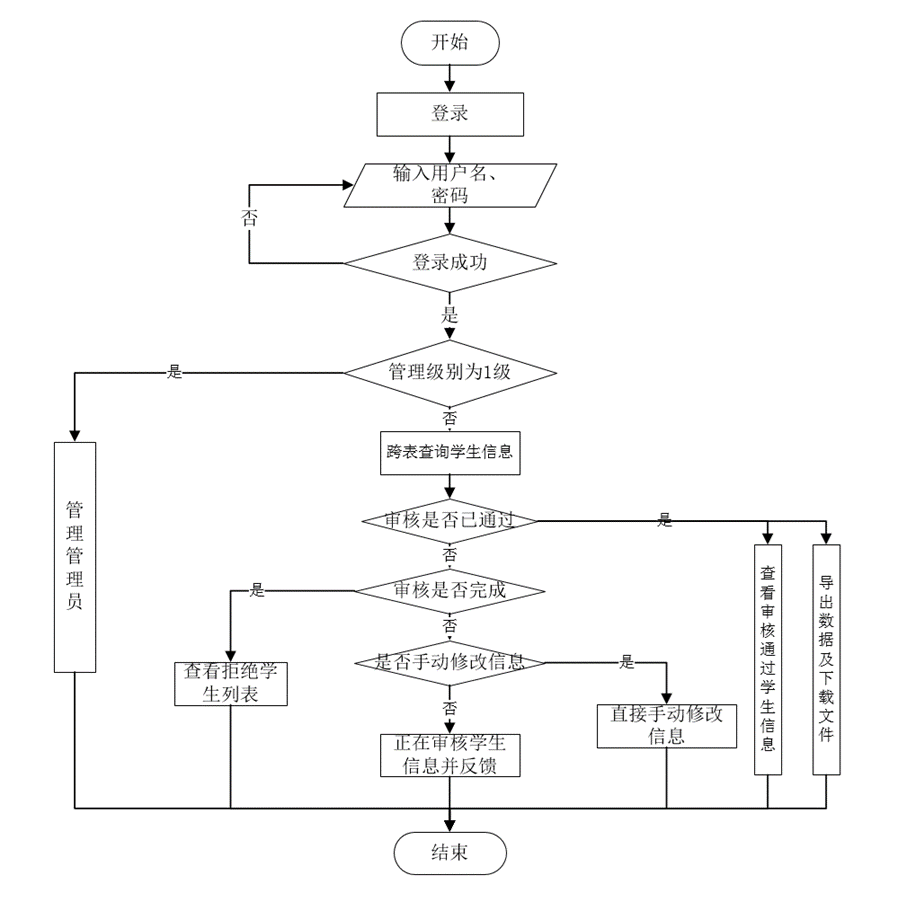


图5.2 教师用户系统流程图

后台管理界面即为教师操作界面，教师通过账号、密码、验证码登录到后台管理系统中。教师管理分为1、2、3级权限。1级权限即为管理员状态，可以管理其他教师信息，也可以修改自己的信息，但不能删除自己。2级和3级权限无法管理教师信息。用户管理功能包括，增加新的教师，删除已有的教师，以及修改教师的密码和权限。但所有教师均有权限进行跨表查询学生信息、审核学生信息并反馈、手动修改学生信息、查看审核已通过学生信息、查看拒绝学生信息以及导出数据和下载相关文件。教师登录到后台系统后，可以对未审核、等待修改、审核通过、学生列表、拒绝列表、导出数据六项中进行跨表查询，迅速便捷地定位学生信息。如果该学生信息正在审核，教师继续对其进行审核，逐条判断每项是否符合要求，并适当地提出反馈意见。如果该学生信息有小部分填写不妥且教师可以帮助修改的信息，教师可以直接手动对其修改。如果该学生申请信息不符合学校研究生申请要求，则教师将其列为拒绝学生，并给出拒绝理由；反之，若符合要求，教师将其列为审核已通过学生。教师可以查询拒绝学生列表和审核已通过学生列表的信息。特别地，查看审核已通过学生列表信息可以看到他支付费用的相关情况，包括是否已经交费和交纳了哪些费用等。对于审核已通过学生，教师可以生成其申请表、录取通知书并导出，还可以下载其相关文件。

子模块流程图如下图所示：

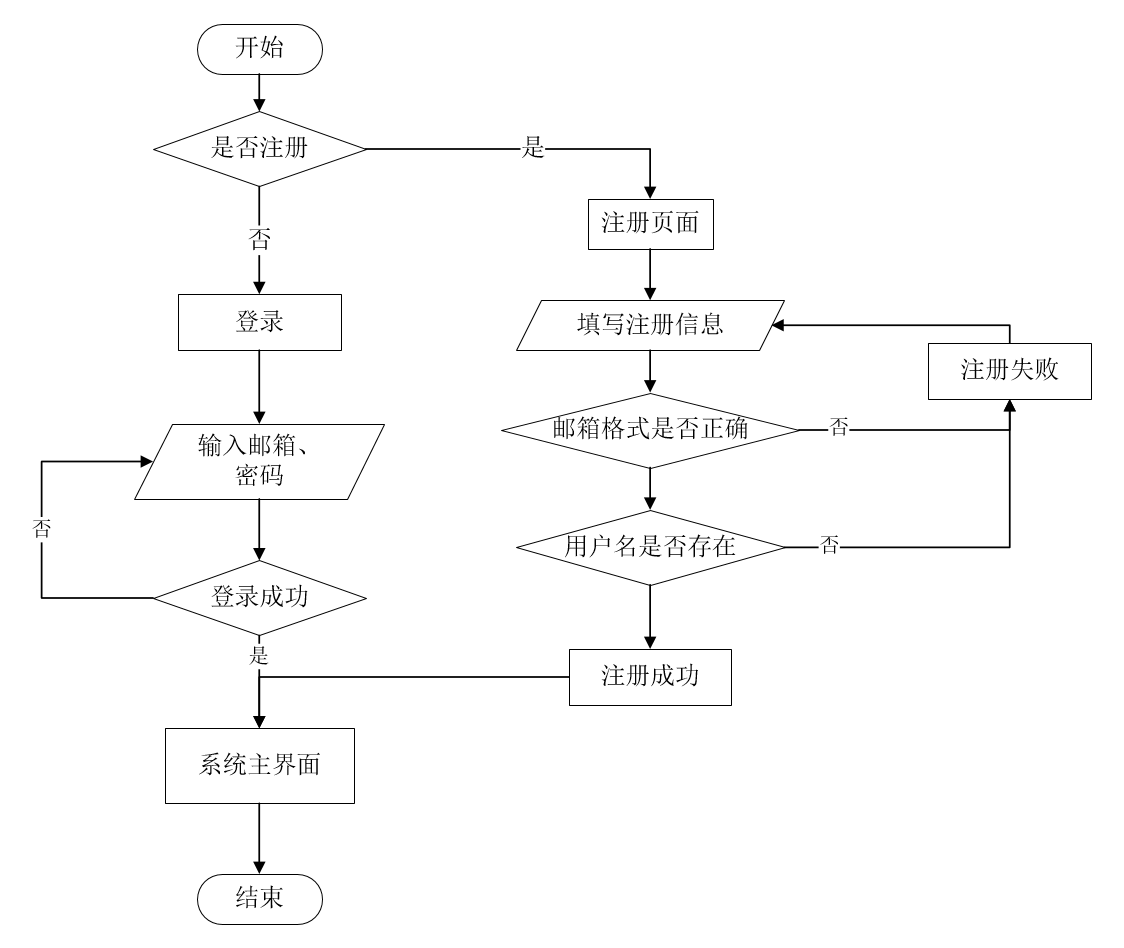


图5.3 学生注册登录模块流程图

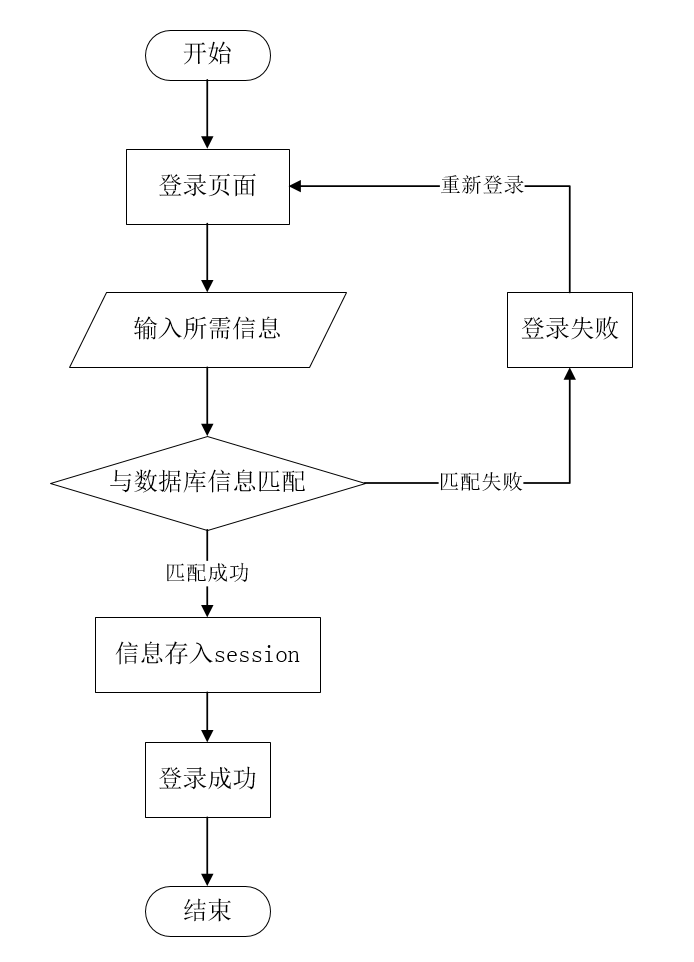


图5.4教师登录模块流程图

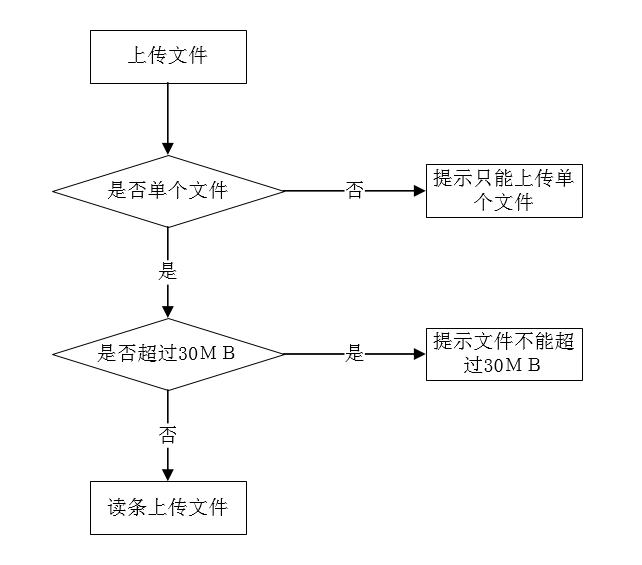


图5.5 上传文件模块流程图

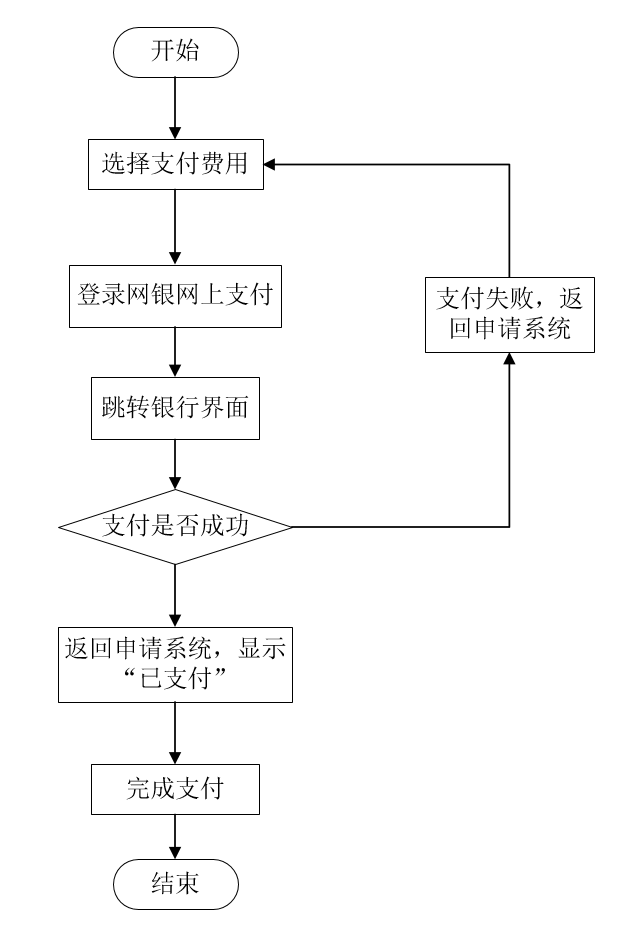


图5.6 支付模块流程图

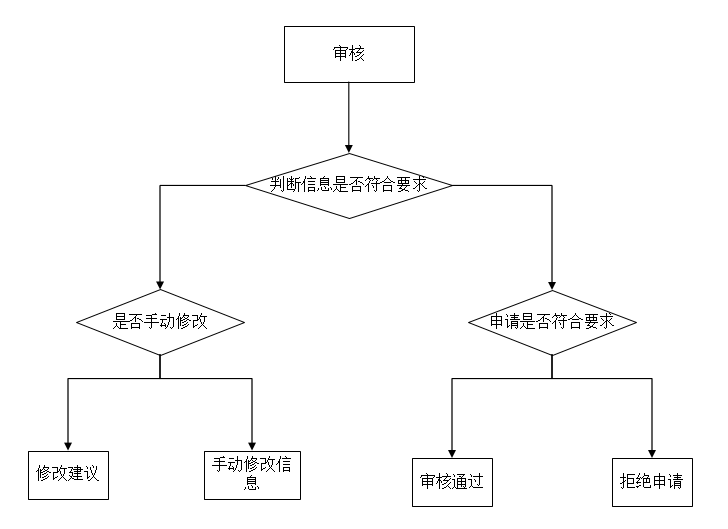


图5.7 审核模块流程图

## 5.2与数据库的交互

采用Microsoft公司开发的SQL Server 2008数据库软件。与数据库的交互，采用ADO.NET技术。ADO.NET是微软提供的用于应用程序和数据库操作的组件，ADO.NET提供平台互用性和可伸缩访问，通过ADO.NET就能在程序中执行SQL语句，完成对数据库的各种操作，如对数据的检索、修改、插入和删除等操作。ADO.NET将数据访问和数据处理分开，其由两部分组成：数据访问组件和数据存储组件。数据访问组件，用于访问数据库，由Connection、Command、DataReader和DataAdapter组成。数据存储组件，从数据库中提取数据并保存，由DataSet组件组成。

使用Connection连接数据库的步骤：

(1).添加命名空间 System.Data.SqlClient

(2)定义连接字符串。连接SQL Server 数据库时：server=服务器名；database=数据库名；uid=用户名；pwd=密码；

(3).创建Connection对象：

SqlConnection connection=new SqlConnection(connSting);

(4).打开数据库：

connection.Open();

(5).对数据库操作完毕后关闭数据库连接

connection.Close();

## 5.3系统实现

### 5.3.1注册登录模块

无论是前台页面，即学生的研究生申请模块，还是后台管理模块，甚至包括在线支付模块，想要进入系统，第一步首先要做的就是注册和登录。

学生注册的第一步先确定申请类型，毕业生申请，本科生申请还是非学历申请。选中三者之一的按钮，会从红色变为橙黄色。接下来就是填写注册信息，其中包括：姓、名字、邮箱、性别、密码和密码确认。考虑到外国人和中国人命名习惯的不同，本系统特别增加了中间名字段，但它是可以为空的。带\*号的字段信息不能为空，即对应数据库学生申请表中姓、名字、邮箱、性别、密码和密码确认字段不允许为空。邮箱作为账号的唯一标识，还会用于后面的密码找回服务，因此最好填写真实的常用邮箱。密码至少为8位以上，确认密码字段必须与密码一致。每注册一项，正在输入的输入框的外框边缘会有淡蓝色光影效果。完成一项的输入后，标签会由浅蓝色变为墨绿色。完成所有注册信息的填写，按Create Your User按钮进行提交，最后转入系统的信息录入界面。

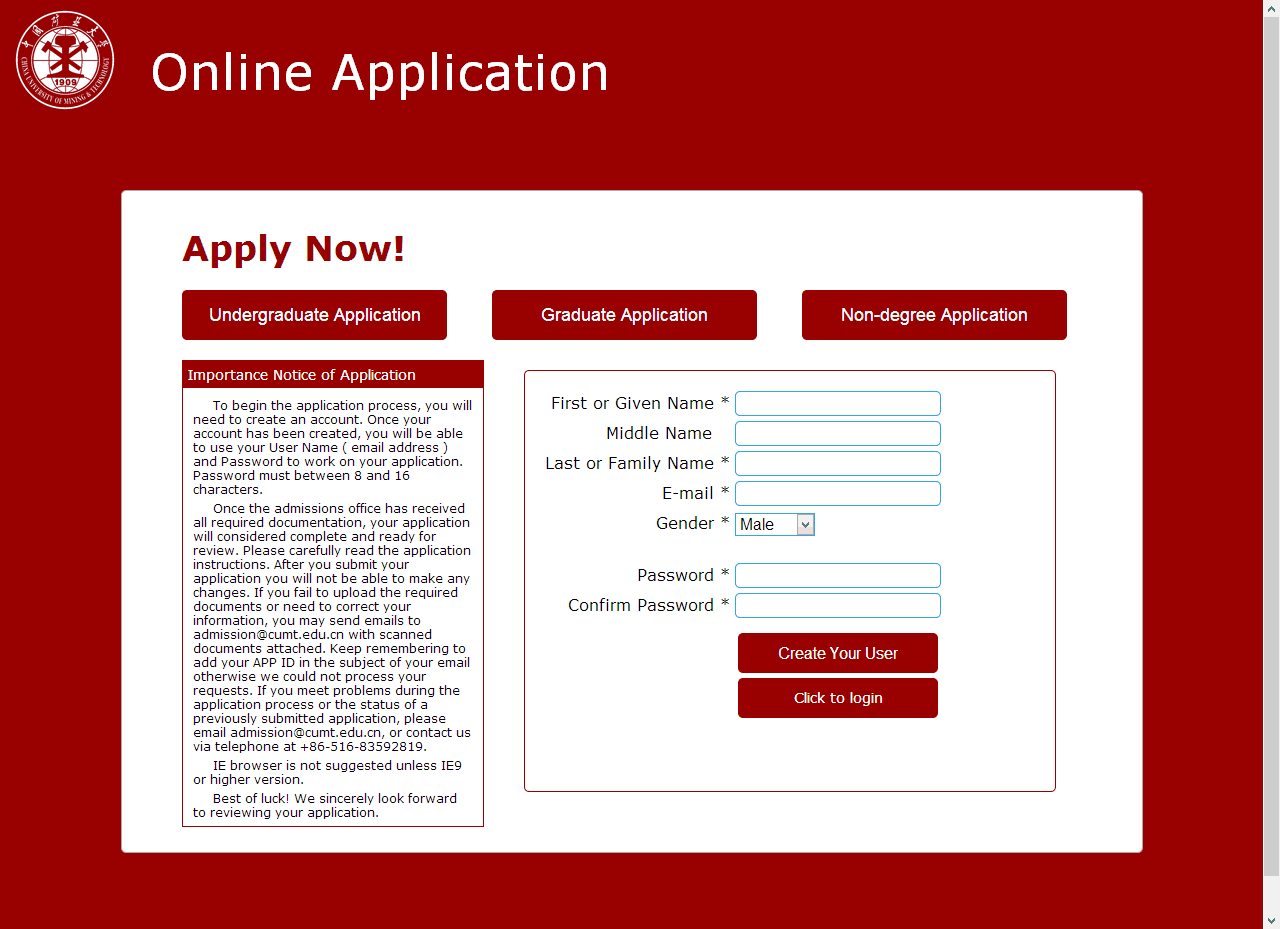


图5.8 学生注册实现图

对于已经完成注册的学生，直接点击Click to Login按钮转入登录页面。登录信息包括邮箱、密码和验证码。邮箱作为识别信息，一个邮箱对应一个账号，因此邮箱不可以重复。登录系统填写密码时，如果忘记了当初注册时设置的密码，本系统提供了重设密码的服务。点击Forget your password？超链接，进入找回密码页面。需要填写注册时所有的邮箱，点击send按钮，系统核对无误后会自动将重设密码的链接发送至该邮箱中。学生进入自己设置的邮箱后，点击链接，重设自己的密码。

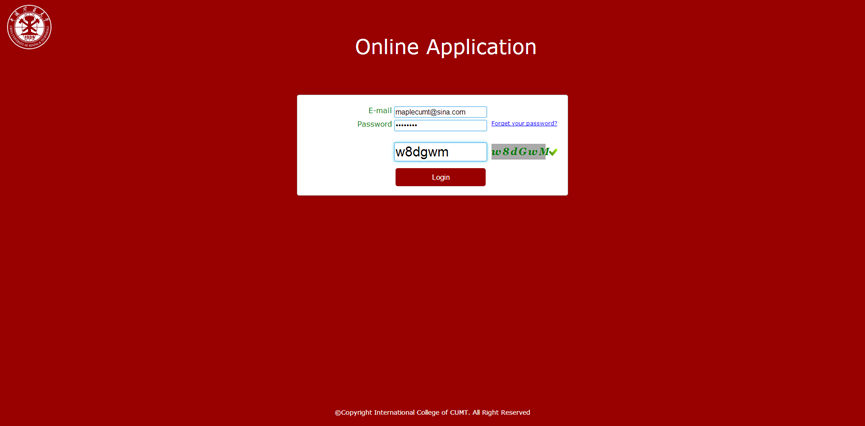


图5.9 学生登录实现图

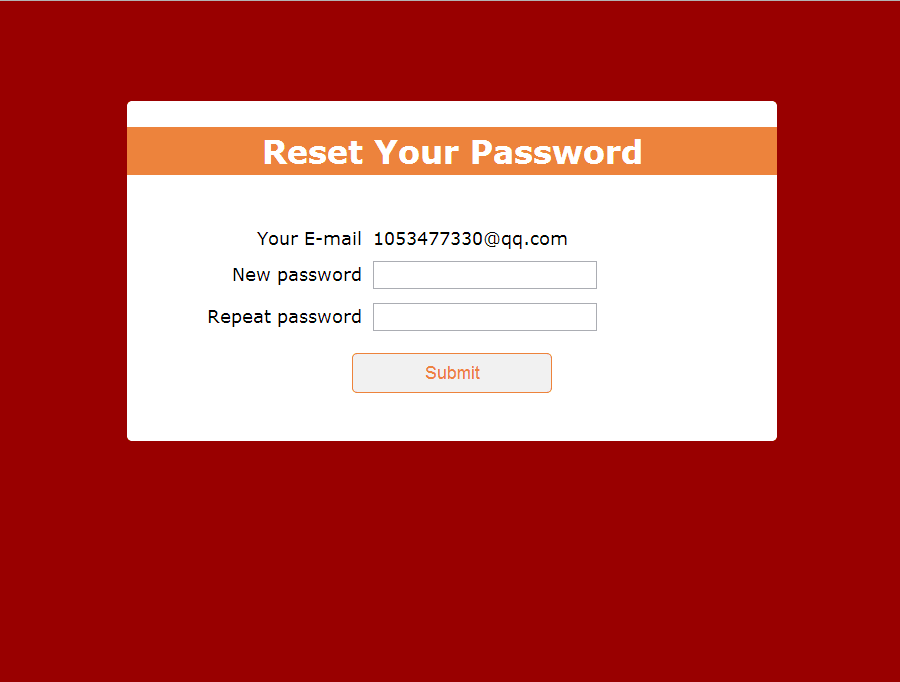


图5.10 学生忘记密码实现图

后台管理页面中，教师需要填写账号、密码和验证码才能实现登录。如果账号或者密码错误后，会弹出对话框，“提示：用户名或密码错误”。待验证码正确验证后，输入框后面会显示一个绿色的对号，显示完成验证；如果验证码错误，输入框后面会显示“验证码错误”的提示。



图5.11 教师登录后台管理系统实现图

在学生选择了所需支付的费用后，需要登录网上银行进行在线支付。填写自己的账号，即卡号，以及密码登录网上银行系统。

（1）MD5加密

为了提高密码的安全性，本系统中将密码存入数据库前使用了MD5加密。加密一般有两种，即双向加密和单向加密。双向加密最常用，它既能加密又能解密；单向加密只能对数据进行加密，不能对其解密[18]。MD5加密是根据指定的密码和哈希算法生成一个适合于存储在配置文件中的哈希密码。命名空间为System.Web.Security。

语法如下：

string passwordMD5;

passwordMD5=FormsAuthentication.HashPasswordForStoringInConfigFile(studentInputInfo.password, "MD5");

（2）验证码的设计（CAPTCHA）

2002年，卡内基·梅隆大学的Luis von Ahn、Manuel Blum、Nicholas J.Hopper以及IBM的John Langford联合提出了验证码的概念，它的中文意思就是“全自动区分计算机和人类的图灵测试”，缩写为CAPTCHA。

实际上，验证码就是将一串随机产生的数字或符号，生成一幅图片，图片里加上一些干扰元素，由用户肉眼识别其中的验证码信息，然后输入表单提交网站验证，验证成功后才能使用某项功能。验证码通常用在用户登录或留言的网页界面中，当用户在浏览器端输入验证码之后，将用户名、密码和验证码等信息提交到服务器，服务器端获取用户的提交之后，判断用户提交的验证码字符与服务器端保存的字符是否相同。如果相同，则通过对用户提交信息的验证；否则将提示没有通过验证的提示信息。验证码的作用就是防止机器恶意注册、破解密码、刷票、刷页以及论坛灌水等。

本系统中验证码设置为6位长度，是由2-9，a-z，A-Z等60个字符随机组合而成的字符串。实现代码如下：

var code;

function createCode() {

code = "";

var codeLength = 6; //验证码的长度

var checkCode = document.getElementById("checkCode");

var codeChars = new Array(2, 3, 4, 5, 6, 7, 8, 9,

'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'm', 'n', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',

'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'J', 'K', 'L', 'M', 'N', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z'); //所有候选组成验证码的字符，当然也可以用中文的

for (var i = 0; i < codeLength; i++) {

var charNum = Math.floor(Math.random() \* 52);

code += codeChars[charNum];

}

if (checkCode) {

checkCode.className = "code";

checkCode.innerHTML = code;

}

if (code != null && code != undefined) {

code = code.toLowerCase();

}

$("#hiddenCode").val(code);

return code;

}

function validateCode() {

var inputCode = document.getElementById("inputCode").value;

if (inputCode.length <= 0) {

return false;

}

else if (inputCode.toUpperCase() != code.toUpperCase()) {

//createCode();

return false;

} else if (inputCode.toUpperCase() != code.toUpperCase() && inputCode.length == code.length) {

createCode();

return false;

}

else {

return true;

}

}

### 5.3.2基本信息录入模块

学生在登录系统后，需要录入自己的详细信息，其中包括个人信息、家庭信息、学习计划、奖学金申请、教育信息、工作经验、资金支持、语言能力等。首先填写的是个人信息，包括私人信息、家庭住址、近期邮件地址和紧急联系人。考虑到家庭住址可能与近期邮件地址一致，那么直接勾选“家庭住址和邮件地址一致”选项，可以收起近期邮件地址，系统默认为家庭住址。每一个小分类都有自己的必填项，如私人信息中，国籍、母语、出生地、出生年月均为必填。如果在提交个人信息时，必填项未填写完整，则无法提交，系统提示“This field is required.”完成填写后，点击“Save and Continue”按钮，保存并提交。与此同时，系统跳转填写家庭信息，而左边的导航栏会显示个人信息后有一个绿色的对号，表示个人信息已经填写完成并提交。从家庭信息以后，提交按钮的左边有一个“preview”按钮，用于对前面信息的修改，或者，点击左侧的导航栏，选择需要修改的信息。在填写教育信息时，学生可以根据自己的情况选择几个大学或学院进行填写，选择值为0-5，相应会出现0-5个大学或学院信息进行填写。在填写语言能力信息时，学生根据第一个问题的答案，会有不同的第二个问题，呈现出树状结构，递进式的问题。例如，考察学生的英语能力时，以TOEFL和IELTS的成绩为标准，那么当学生选择TOEFL时，就会出现有关TOEFL的问题，如TOEFL的类型，考试时间，考试总成绩，考试的阅读、听力、写作、口语各项的分数；而当学生选择IELTS时，相应地就会出现有关IELTS的问题。在上传文件时，会显示上传速度及进度并读条。全部信息填写完成后，点击“Continue”按钮进入提交页面。“Back”按钮返回填写的信息，可以进行修改；“Review”按钮可以生成预览；“Submit”按钮完成提交，跳转到申请结果界面，显示申请的基础信息，反馈，以及添加推荐人信息。

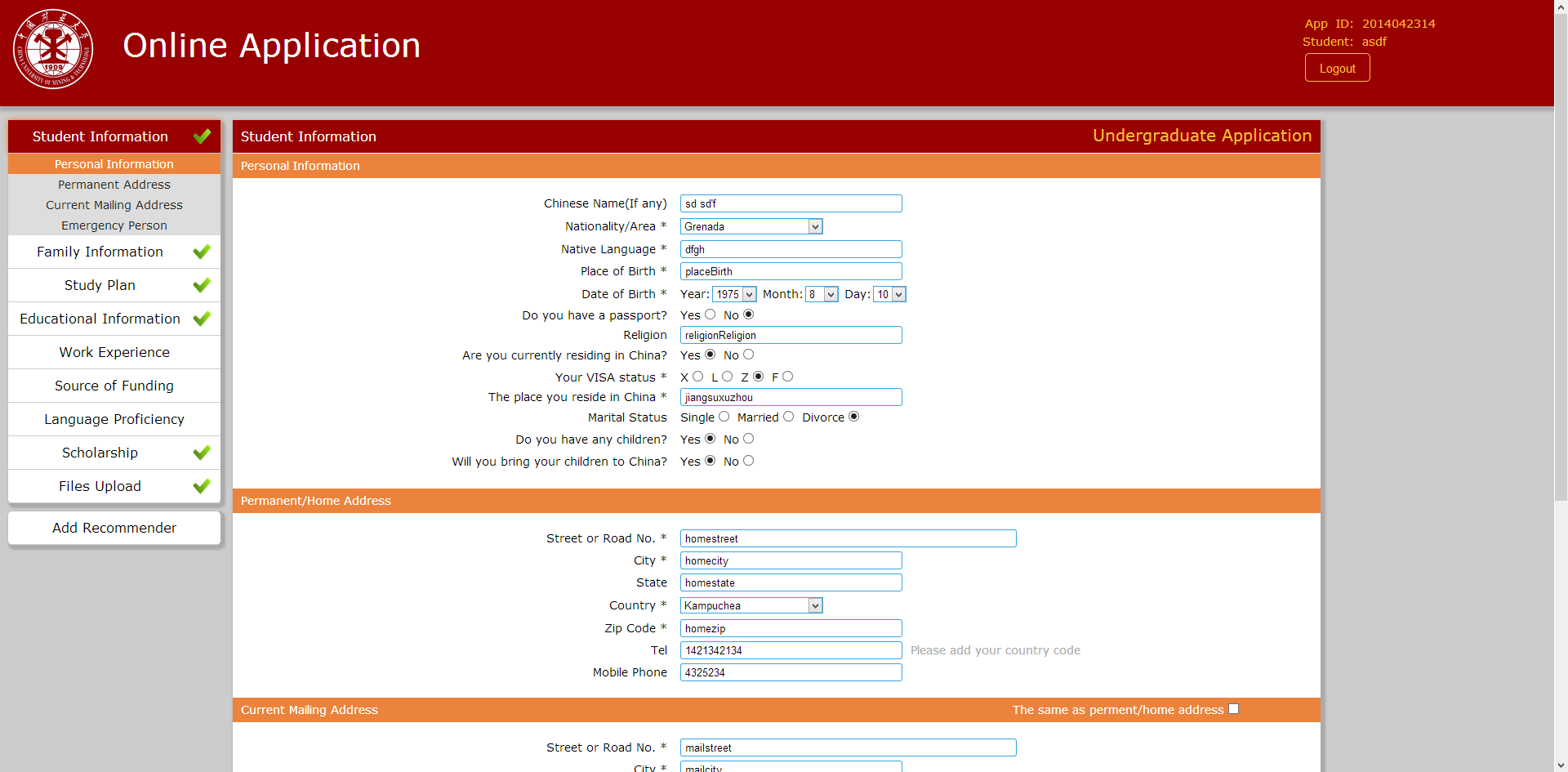


图5.12 学生个人信息录入实现图

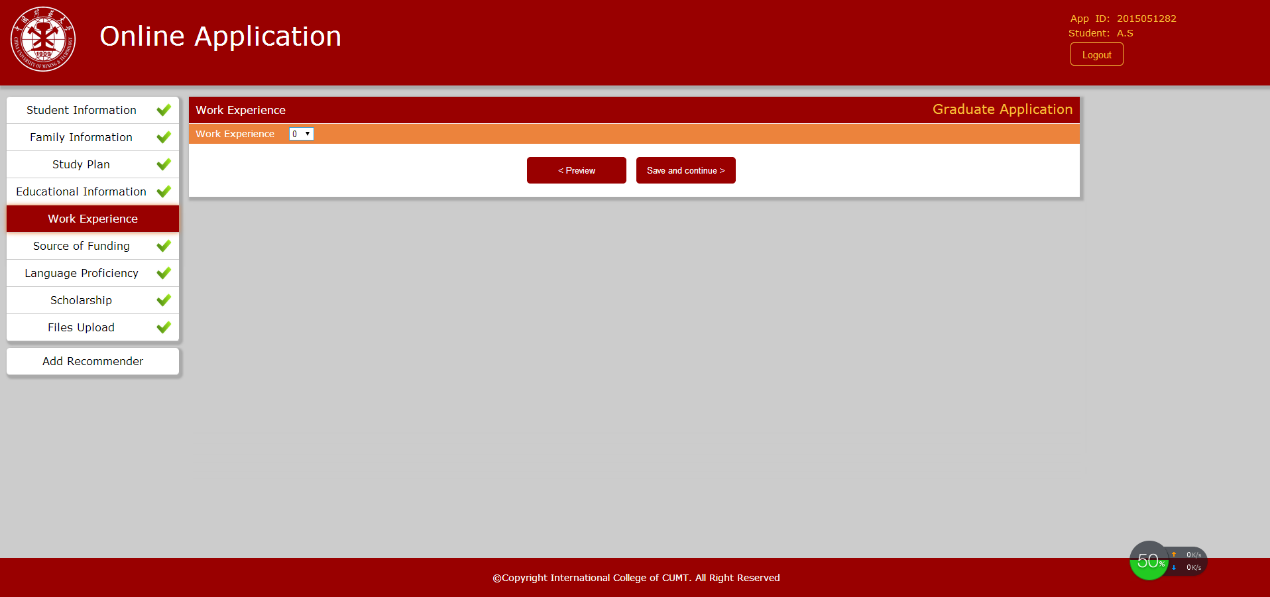


图5.13 学生工作经验录入实现图

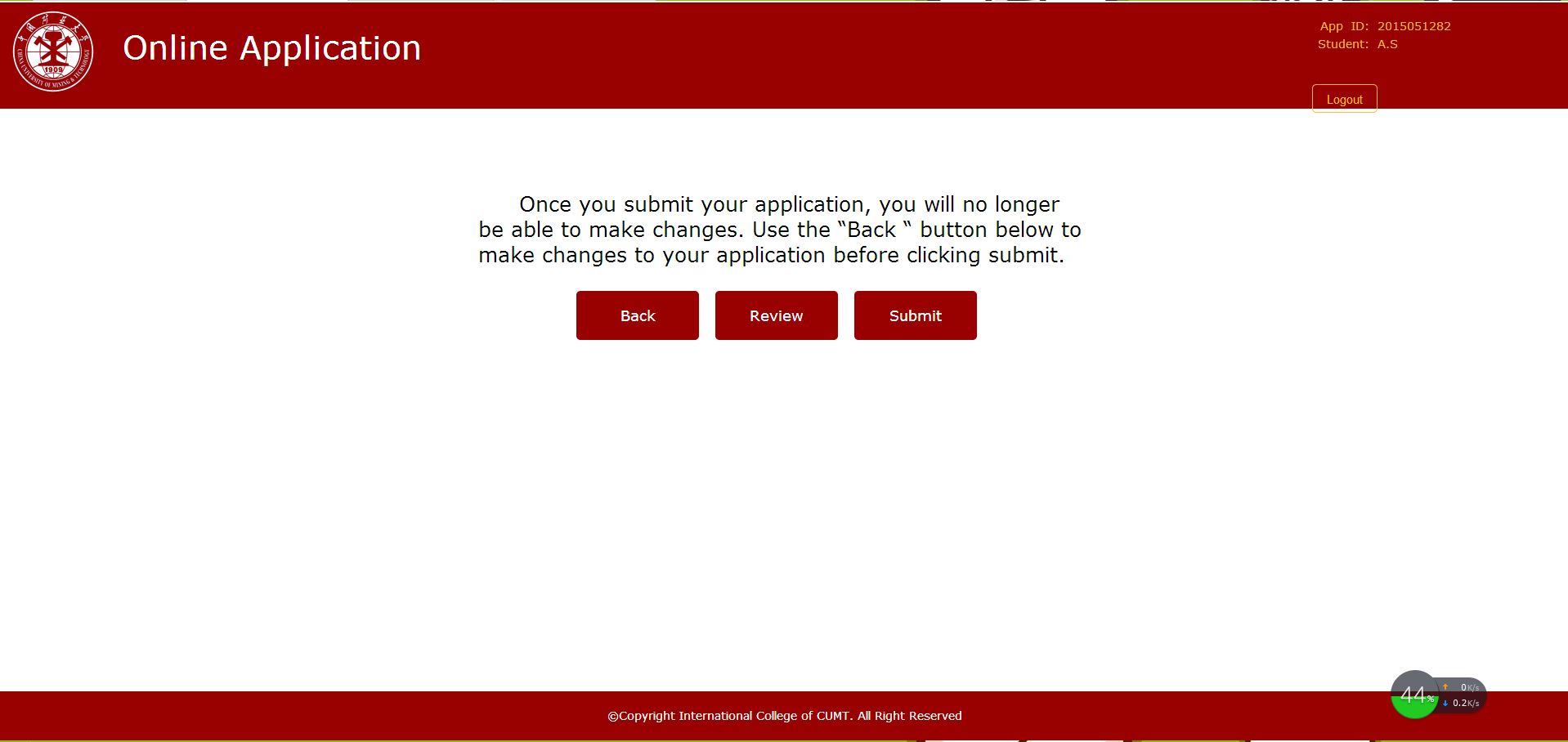


图5.14 学生信息提交实现图

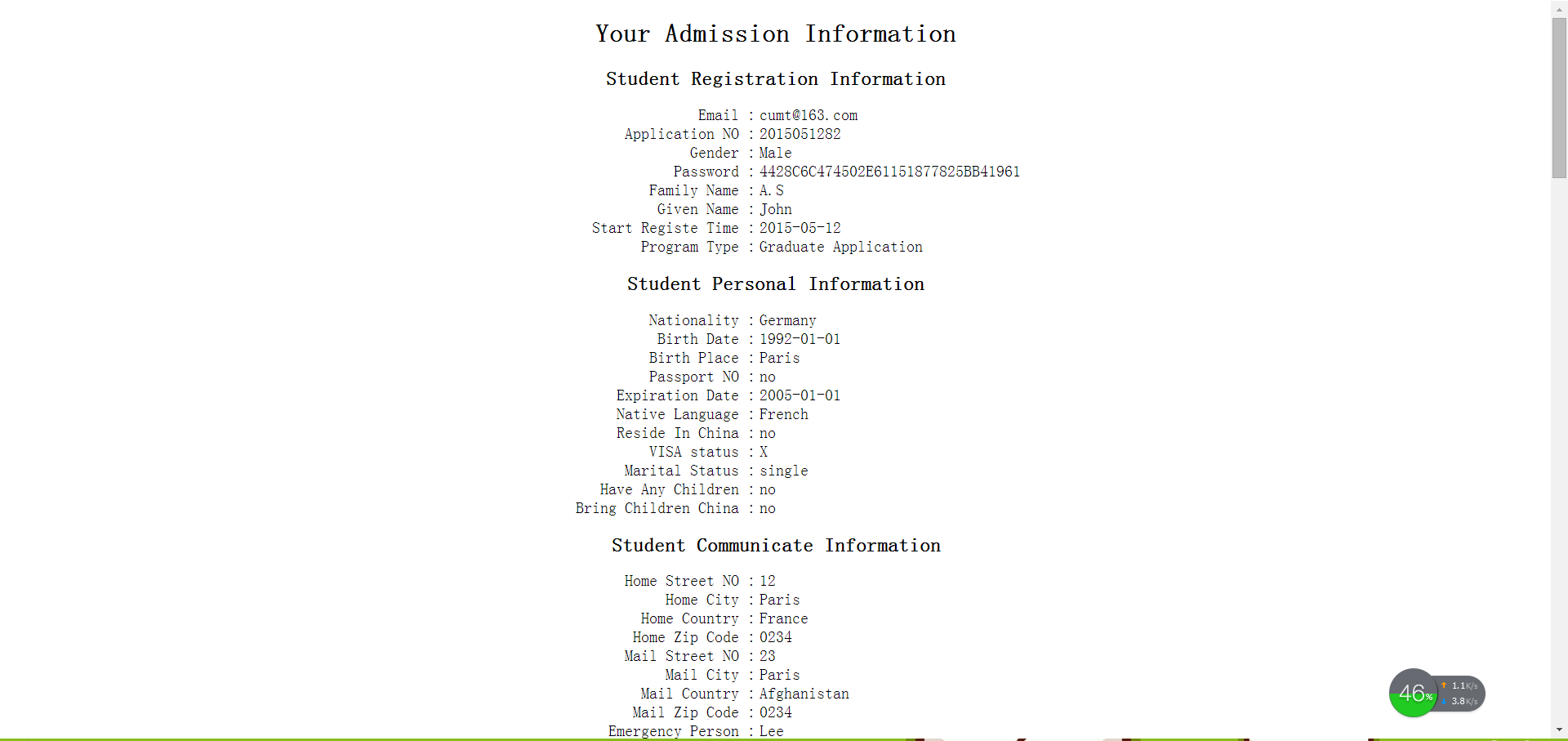


图5.15 学生信息预览图



图5.16 学生信息提交后结果图

### 5.3.3上传文件与导出数据模块

学生申请需要上传相应的证明材料，包括体检表、高中或大学成绩单、学位证书、有效护照照片等。学生通过审核后，教师可以导出学生的相应信息，可以导出申请表、导出录取通知书，或者下载学生填写的任何材料。

1. 上传文件模块

所需上传的证明材料必须为pdf文档，且不能超过30MB。每个上传项只能选择一个文档上传，如有多个需先进行合并。上传材料包括体检表、高中或大学成绩单、学位证书、有效护照照片、汉语水平考试成绩单、有效资金支持表及其他有利于申请的相关材料。

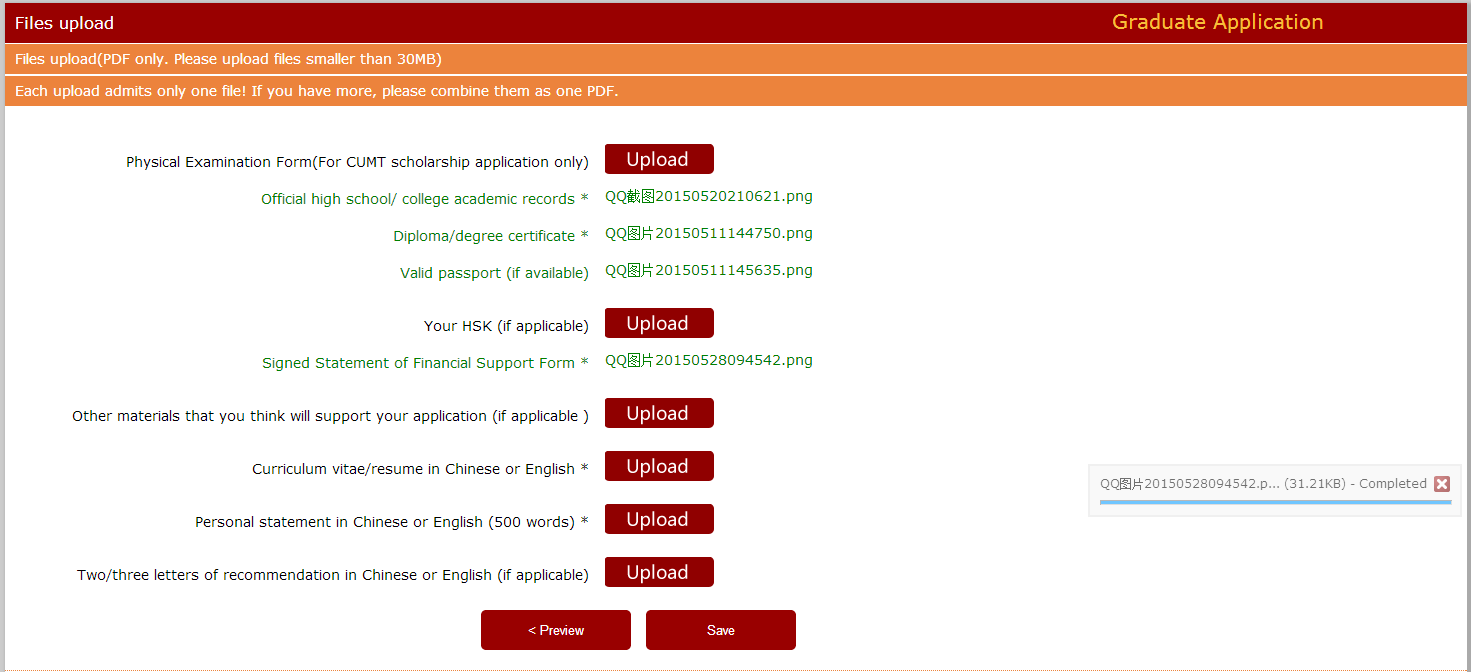


图5.17 上传文件图

上传模块是通过jQuery框架的uploadify插件实现的，uploadify插件是一个基于Flash技术的文件上传jQuery插件。jQuery，顾名思议，也就是JavaScript和查询（Query），即是辅助JavaScript开发的库。Uploadify插件的功能特色有：支持单文件或多文件上传，可控制并发上传的文件数；在服务器端支持各种语言与之配合使用，诸如PHP,.NET,Java等；通过参数可配置上传文件类型及大小限制；通过参数可配置是否选择文件后自动上传；易于扩展，可控制每一步骤的回调函数；通过接口参数和CSS控制外观。

第一步需要做的就是下载uploadify zip包。解压缩后，将其拷贝到项目文件夹下，如图所示：

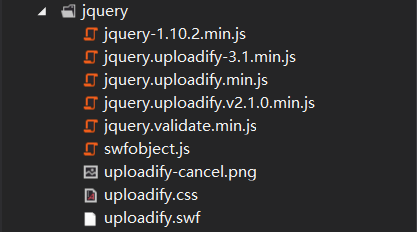


图5.18 uploadify下载内容图

可以看到里面的几个主要文件：jquery.uploadify.js是完成上传功能的脚本文件，需要在调用页面引用；uploadify.css是外观样式表；uploader.swf是上传控件的主体文件，flash控件；upload.php是一个官府提供的PHP文件，用于服务器处理上传文件。而在ASP.NET中可以使用“一般处理程序（ashx）”文件作为服务器端处理上传的文件。

1. 导出数据模块

在后台管理系统中，对于审核已通过的学生，教师可以生成其申请表和录取通知书并用word导出相应文件，可以下载学生的所有上传文件，还可以选择需要的字段生成Excel文件。导出数据模块采用了COM组件技术，利用微软提供的程序集Microsoft.Office.Interop.Excel,dll和Office.dll，但使用的前提是必须安装Office软件。接下来，以导出Excel文件为例，介绍一下COM组件技术。

1） ASP.NET引用COM组件的原理

.NET Framework的核心是公共语言运行库。在.NET公用语言框架内运行的程序为受管代码。受管代码在程序中所有类型都受到严格检查，没有指针，对内存的管理完全由运行系统控制。受控状态下，编写 程序更为容易，且更少出错，我们可以花更多的时间在解决实际问题上而不是在计算机语言问题上。相对而言，那些在.NET框架外运行的程序为非受管代码。比 如：COM组件、ActiveX组件、Win32 API函数、指针运算等。C#编程中在某些特定情况下，需要运用非受管代码，例如，要利用一个成熟的COM组件，或者调用一个API函数，或者用指针去编 写实时/高效程序等。公共语言运行库通过名为运行库可调用包装(RCW)的代理来公开COM对象，如图1所示。虽然RCW在.NET客户端看来是普通的对象，但它的主要功能是封送在.NET客户端和COM对象之间传递的调用。同时. NET提供Interop程序集，它用作托管和非托管代码之间的桥梁，将COM对象成员映射为等价的.NET托管成员。

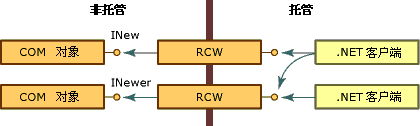


图5.19 通过运行库可调用包装来访问COM对象

操作Excel表格最直接的方法就是利用Excel提供的Excel Object Library COM组件，并将包装后的程序集叫做“互操作程序集”（Primary Interop Assembly, PIA）。

下面介绍ASP.NET引用COM组件，设置权限和处理Excel表格的各种典型操作，如编辑表格、保存表格等，以及从二维数组中向Excel表格快速导入数据的实现方法。

2 ）ASP.NET引用COM组件的方法

建立ASP.NET工程，添加引用COM组件“Microsoft Excel 15.0 Object Library”，如图2所示。前提是计算机上安装了Microsoft Office的Excel软件，该文例子所使用的是Office2013。引用过程就是RCW的打包过程，.NET自动创建PIA，也可以通过.NET提供的工具Tlbimp.exe手动创建PIA。

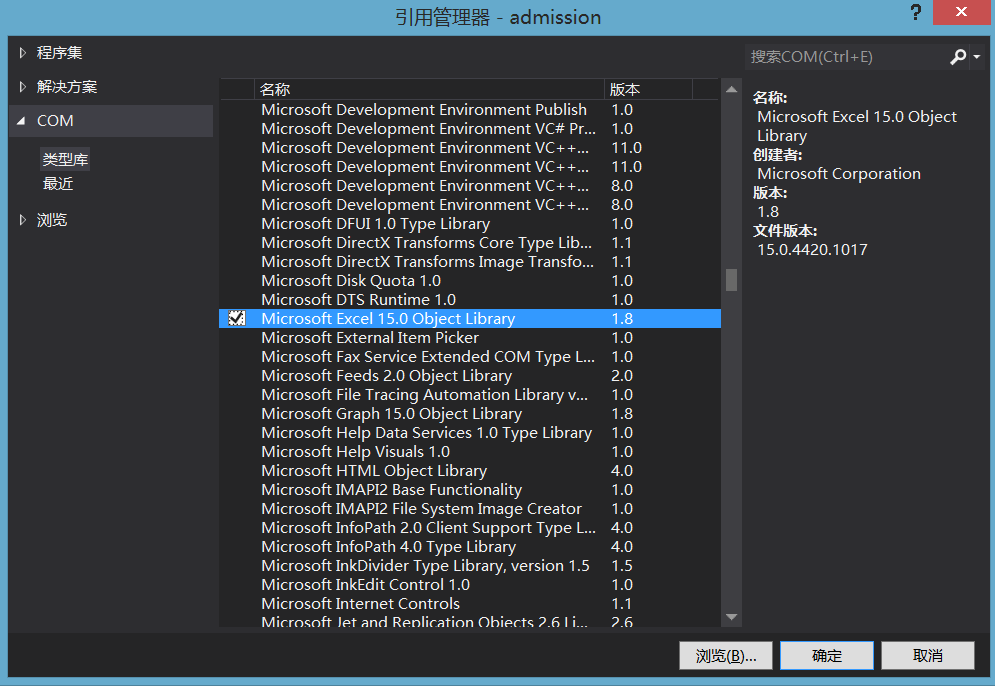


图5.20 引用COM组件图

3） 具体实现代码

$(function() {

generateTableTitle();//生成表格抬头

generateSelectOption();//筛选表格

outPutTableData();//导出表格数据

//DownLoadZip();//导出压缩文件

logOut("dataExport.aspx");

});

var generateArray = new Array();

//生成表格抬头

function generateTableTitle() {

$(document).on("click", ".chooseTitle", function () {//添加列

var theName = $(this).attr("name");

var theTitle = $(this).html();

$(".fileDownload").slideUp();//隐藏文件下载

$(this).fadeOut(function () {

//console.log($(".exportTable th").html())

if ($(".exportTable th").html() == undefined) {

if (theName != "theNum") {//点击非序号标题添加序号

$("li[name='theNum']").hide();

$(".exportTable").append("<th title='删除' name='theNum' class='exportTableTh'>序号</th>");

generateArray.push("theNum");

}

if ($(".exportDiv").css("display") == "none") {

$(".exportDiv").slideDown();

$(".exportTable").append("<th title='删除' name='" + theName + "' class='exportTableTh'>" + theTitle + "</th>");

} else {

$(".exportTable").prepend("<th title='删除' name='" + theName + "' class='exportTableTh'>" + theTitle + "</th>");

}

} else {

$(".exportTable th:last").after("<th title='删除' name='" + theName + "' class='exportTableTh'>" + theTitle + "</th>");

}

generateArray.push(theName);

var dataStr = generateArray.join(","); //把数组里的元素连接成字符串

var searchData = JSON.stringify(generateSearchJson("#searchBarForm")); //查询表单

$.ajax({

type: "post",

url: "dataExport.aspx/gengerateTableData",

//data: '{"data":"' + dataStr + '"}',

data: JSON.stringify({ data: dataStr, theSearchData: searchData }),

datatype: "json",

contentType: "application/json; chaset=utf-8",

success: function(data) {

$(".exportTable tr.manageStudentTrOdd").remove();

$(".exportTable tr.manageStudentTrEven").remove();

$(".exportTable").append(data.d[0]);

$(".downloadFile").val("点击下载" + data.d[1] + "个学生的文件");

$(".dataExportStudentNumber").html("符合条件的学生有" + data.d[1] + "名。");

$(".exportDiv").show();

},

error: function(xmlReq, err, c) {

alert("error:" + err + c);

}

});

});

});

$(document).on("click", ".exportTableTh", function (e) {//删除列

//console.log($(e.currentTarget).attr("name"));

if ($(e.currentTarget).attr("name") != "theNum") {

$(".fileDownload").slideUp();//隐藏文件下载

$(".exportDiv").slideDown();

//console.log($(this).html());

$(this).fadeOut("slow", function () {

$("li[name='" + $(this).attr("name") + "']").fadeIn("slow");

$(this).remove();

for (var i = 0; i < generateArray.length; i++) {

if ($(this).attr("name") == generateArray[i]) {

generateArray.splice(i, 1);

break;

}

}

var dataStr = generateArray.join(",");//把数组里的元素连接成字符串

var searchData = JSON.stringify(generateSearchJson("#searchBarForm"));//查询表单

$.ajax({

type: "post",

url: "dataExport.aspx/gengerateTableData",

data: JSON.stringify({ data: dataStr, theSearchData: searchData }),

datatype: "json",

contentType: "application/json; chaset=utf-8",

success: function (data) {

$(".exportTable tr.manageStudentTrOdd").remove();

$(".exportTable tr.manageStudentTrEven").remove();

$(".exportTable").append(data.d[0]);

$(".downloadFile").val("点击下载" + data.d[1] + "个学生的文件");

$(".dataExportStudentNumber").html("符合条件的学生有" + data.d[1] + "名。");

},

error: function (xmlReq, err, c) {

alert("error:" + err + c);

}

});

//console.log(generateArray);

});

}

});

}

//筛选表格

function generateSelectOption() {

$(".searchButton").click(function () {

$(".fileDownload").slideUp();//隐藏文件下载

$(".exportDiv").slideDown();

var theTitleObj = $(".exportTableTh");

var theTitleArray = [];

for (var i = 0; i <theTitleObj.length ; i++) {

theTitleArray[i] = $(theTitleObj[i]).attr("name");

}

//console.log(theTitleArray);

var titleStrings = "";

if (theTitleArray.length > 0) {

titleStrings = theTitleArray.join(",");

} else {

titleStrings = "theNum";

$("li[name='theNum']").hide();

$(".exportTable").append("<th title='删除' name='theNum' class='exportTableTh'>序号</th>");

}

var searchData = JSON.stringify(generateSearchJson("#searchBarForm"));

//console.log(searchData);

$.ajax({

type: "post",

url: "dataExport.aspx/generateSelectOption",

data: JSON.stringify({ myData: searchData, titleArray: titleStrings }),

datatype: "json",

traditional: true,

contentType: "application/json; chaset=utf-8",

success: function (data) {

//console.log(data.d);

$(".exportTable tr.manageStudentTrOdd").remove();

$(".exportTable tr.manageStudentTrEven").remove();

$(".exportTable").append(data.d[0]);

$(".downloadFile").val("点击下载" + data.d[1] + "个学生的文件");

$(".dataExportStudentNumber").html("符合条件的学生有"+data.d[1]+"名。");

},

error: function (xmlReq, err, c) {

alert("error:" + err + c);

}

});

});

}

//导出表格数据

function outPutTableData() {

$(".generateTableButton").click(function () {

var theTitleObj = $(".exportTableTh");

var theTitleArray = [];

for (var i = 0; i < theTitleObj.length ; i++) {

theTitleArray[i] = $(theTitleObj[i]).attr("name");

}

var titleStrings = "";

if (theTitleArray.length > 0) {

titleStrings = theTitleArray.join(",");

}

var searchData = JSON.stringify(generateSearchJson("#searchBarForm"));

var theTableContent = $(".exportDiv table").html().trim();

//alert("1"+theTableContent+"1");

//theTableContent = theTableContent + "";

if (theTableContent != "") {

$(".exportDiv").slideUp();

$(".dataWaiting").show();

var dataStr = generateArray.join(",");//把数组里的元素连接成字符串

$.ajax({

type: "post",

url: "dataExport.aspx/outPutTableData",

data: JSON.stringify({ myData: searchData, titleArray: titleStrings }),

//data: '{"data":"' + dataStr + '"}',

datatype: "json",

contentType: "application/json; chaset=utf-8",

success: function (data) {

if (data.d != "") {

//console.log(data.d);

//self.location = data.d;

$(".dataWaiting").hide();

$(".fileDownload").slideDown();

}

},

error: function (xmlReq, err, c) {

alert("error:" + err + c);

}

});

} else {

alert("请选择需要导出的项。");

}

});

}

### 5.3.4在线支付模块

学生通过教师审核之后，可以进行在线支付环节，支付费用包含体检费、学费、住宿费等。学生根据自己的情况选择支付费用。如果学生需要在学校进行体检，则勾选体检费一栏。如果学生需要在学校住宿，则需交纳住宿费。确定好需要支付的费用后，点击“Pay Now”按钮跳转网上银行界面。首先填写账号、密码等信息登录网银系统，然后进行网上支付。支付结果返回给学生的同时，也写入数据库的Ispaid数据表。该表记录了学生的支付状态。学生是否支付成功，支付了哪些费用，可以在学生信息的支付状态中看到。教师也可以在审核系统中查看学生的支付状态。

（1）模拟网上银行系统

学生的支付环节，本系统采用了模拟网上银行技术。随着电子商务技术大潮的，网上支付已经普及。当然，系统中支付学杂费的环节也可以突破时间和地点的限制，进行网上支付。

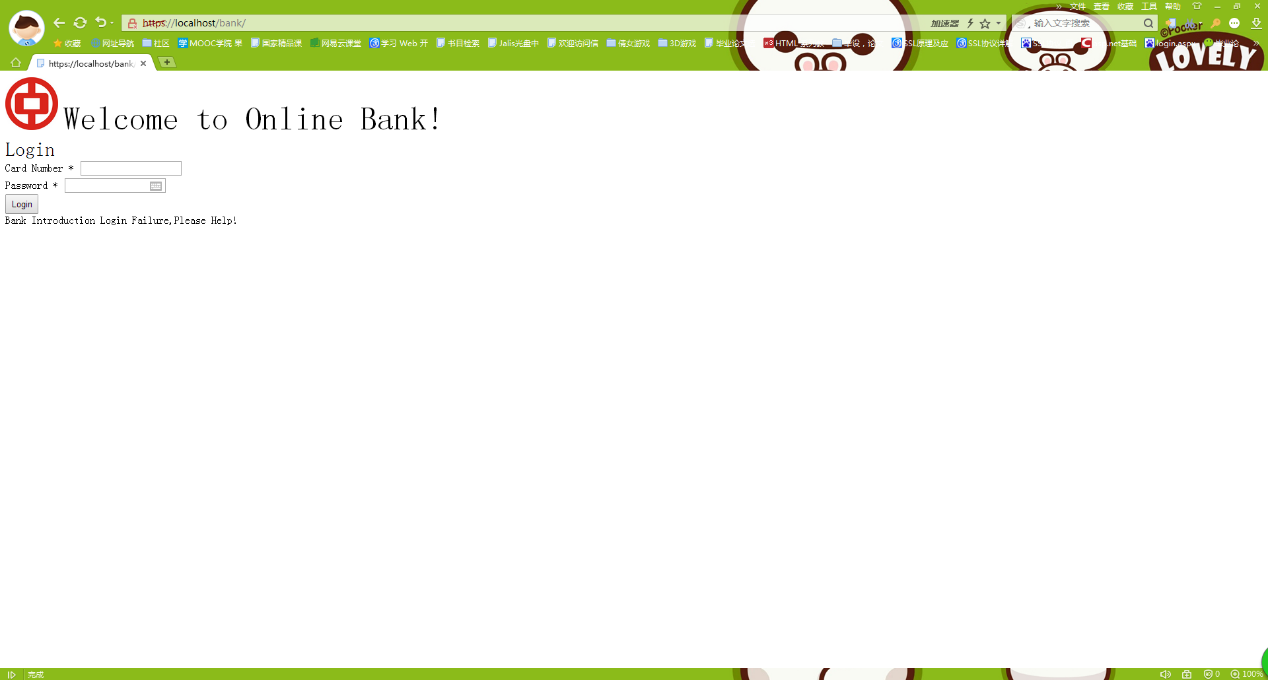


图5.21 网上银行登录图

（2）SSL协议

支付需要考虑安全性问题，本系统使用了SSL协议。下面介绍一下IIS服务器的SSL配置。

在系统已经实现以后，第一步可以先配置IIS服务器。点击运行Internet信息服务（IIS）管理器。展开左侧的节点，可以看见默认的站点“Default Web Site”，本系统就把网站绑定在默认站点上，并填写本系统的物理路径。

接下来配置站点的IP地址和TCP端口。点击默认站点，保持选中状态，点击右侧操作下，编辑网站的“绑定”选项,为其绑定端口。选择类型为“http”，IP地址“全部未分配”，默认端口为“80”。

这样，点击右侧浏览网站下的“浏览\*：80（http）”，就可以在默认浏览器下浏览本系统的主页网站。

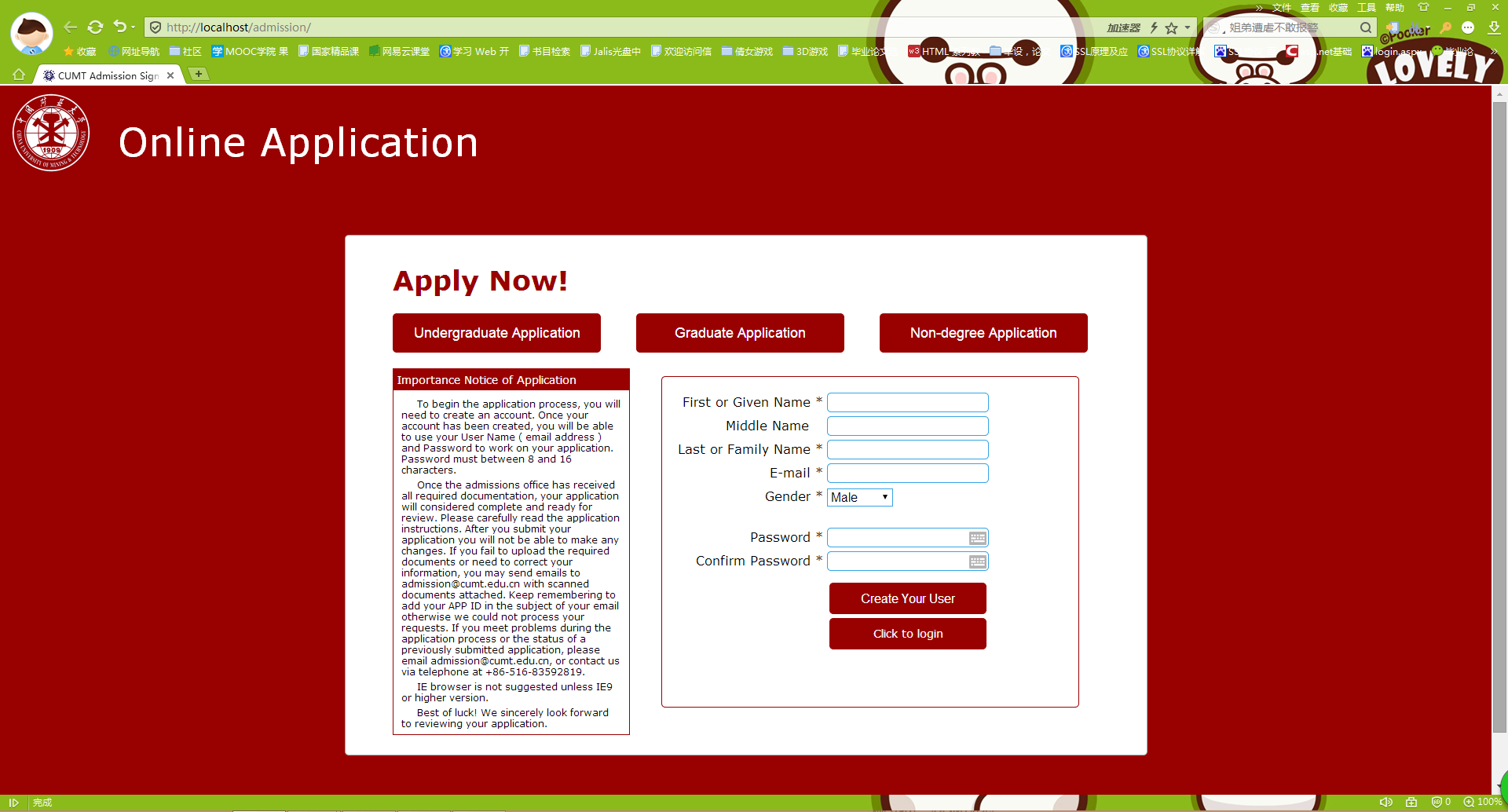


图5.22 网站绑定实现图

接着，继续进行SSL的配置。首先，先需要引入和建立一个安全证书来用于SSL的绑定。点击左侧树形视图管理器里的根机器节点，然后在中间的特性视窗里选择"服务器证书(ServerCertificates)"图案。

双击进入后，点击右侧管理工具操作下的“创建自签名证书”选项。为证书填写一个友好的名字，如“test”，确认完成自签名证书的创建。



图5.23 自签名服务器证书

有了自签名证书后，下面要给网站进行SSL绑定。点击左侧默认站点，再选择右侧操作栏下“绑定”选项，并为其添加。类型为“https”，IP地址“全部未分配”，端口为“443”，SSL证书为“test”，点击确认完成。

如此，点击默认站点后，右侧的浏览网站下会出现“浏览\*：443（https）”。点击此链接，会在默认浏览器下查看经SSL证书保护下的系统主页。

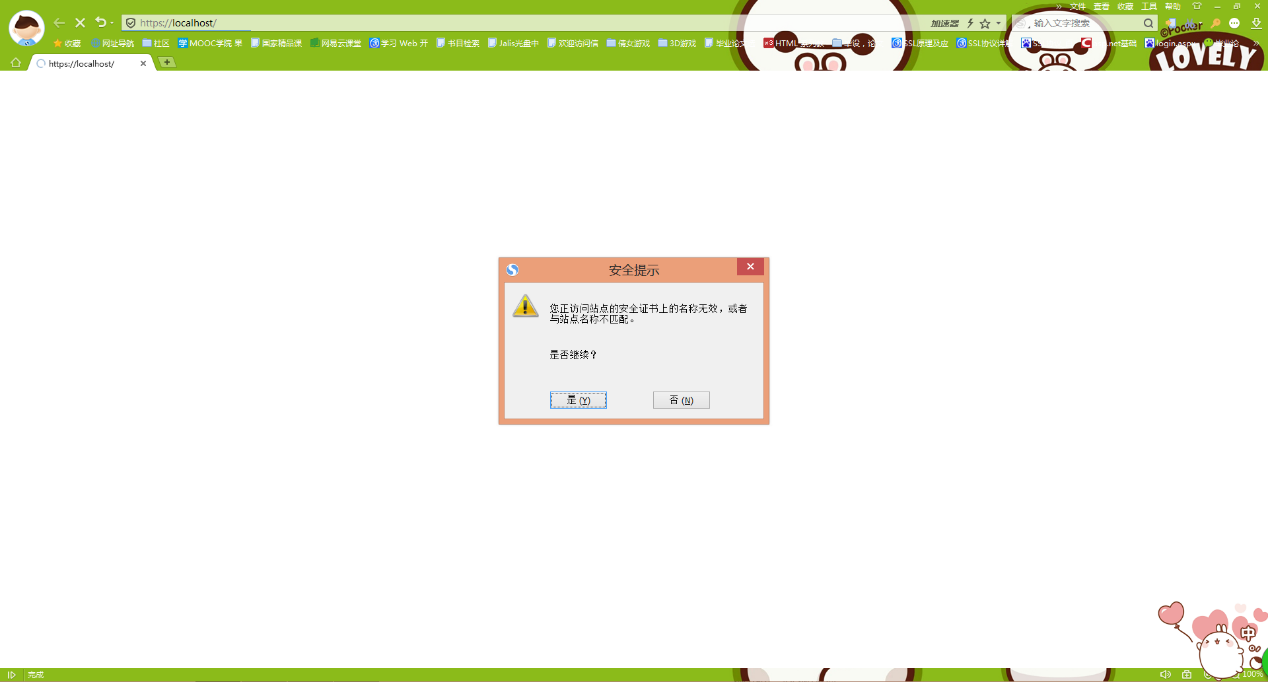


图5.24 绑定SSL协议（1）

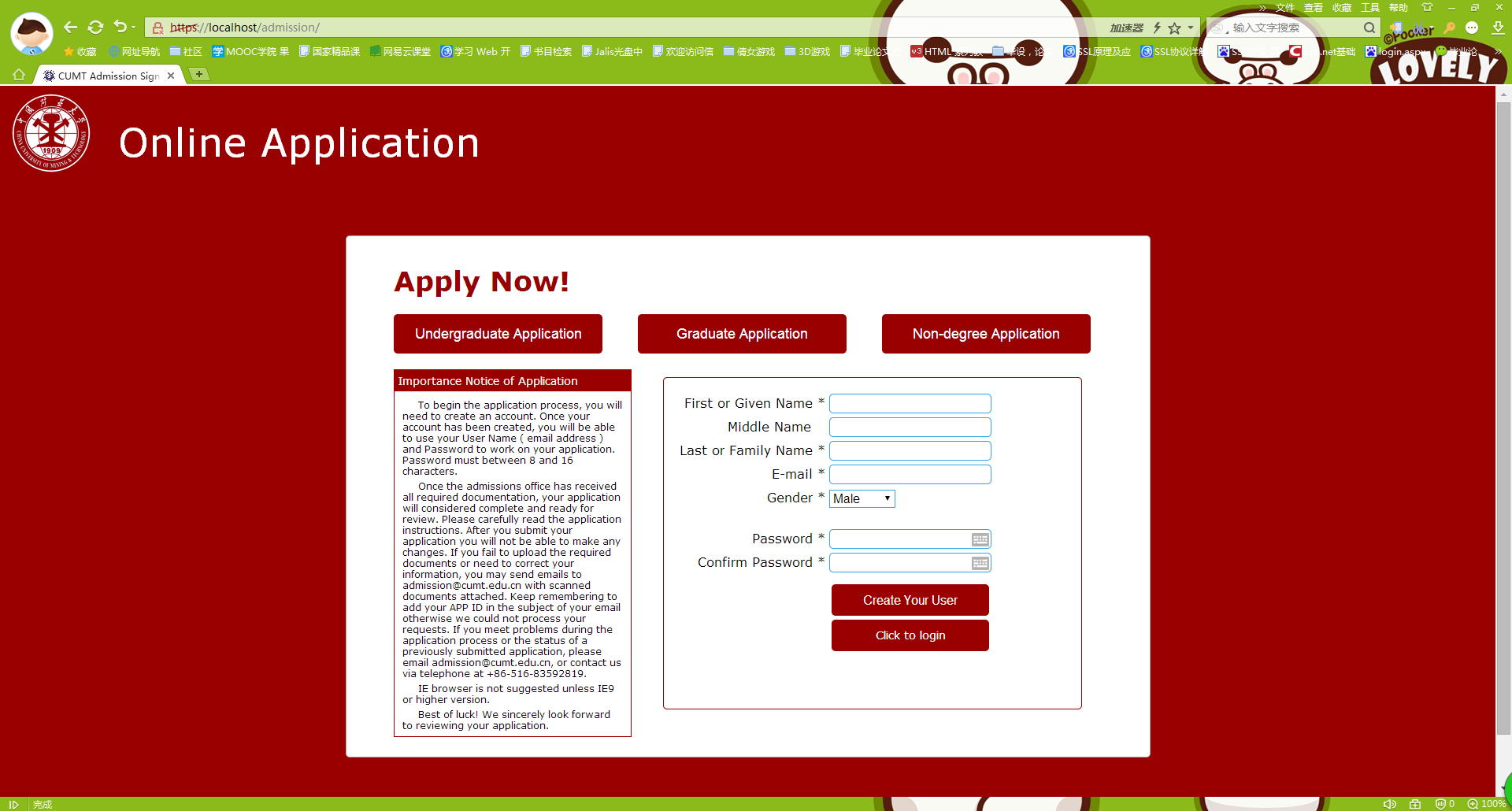


图5.25 绑定SSL协议（2）

需要注意的事，默认浏览器下需要添加“test”自签名证书为可信任证书，即在Internet选项中进行设置，添加可信任证书。

### 5.3.5 跨表查看学生信息

教师登录后台管理系统后，为便于查找某个或某些学生信息，可以通过关键词进行模糊查询，快速搜索到相关学生信息，缩小了搜索范围，迅速查找。

代码如下：

/// <summary>

/// 新的生成查询sql语句

/// </summary>

/// <param name="dataDictionary">查询条件</param>

/// <param name="pageName">页面名称</param>

/// <param name="startPage">开始页码</param>

/// <param name="endPage">结束页码</param>

/// <returns></returns>

public string[] generateSqlStrNew(Dictionary<string, Dictionary<string, string>> dataDictionary, string pageName,string startPage,string endPage)

{

string joinKey = "applicationNO";

string sqlCount = "",selectEmail="select email ",checkStatusOrder="";

if (pageName == "studentManage"||pageName=="studentList") checkStatusOrder = " DESC ";

string theSqlStr = " from ( "+

"select ROW\_NUMBER() over(order by checkStatus "+checkStatusOrder+", finishRegisteTime DESC,startRegisteTime DESC) as rownumber, "+

"email,studentsRegister\_tb.applicationNO,gender,givenName,middleName,familyName,nationality,admissionYear, " +

"startRegisteTime,finishRegisteTime, studentsRegister\_tb.programType,checkTime,birthDate,checkStatus " +

"from studentsRegister\_tb "+

"full join studentsPersonalInfo\_tb "+

"on studentsRegister\_tb.applicationNO=studentsPersonalInfo\_tb.applicationNO"+

" full join studentsApplyInfo\_tb on studentsRegister\_tb.applicationNO=studentsApplyInfo\_tb.applicationNO " +

"left join ( select max(checkTime)as checkTime,applicationNO from manage\_tb group by applicationNO )"+

"manage\_tb on studentsRegister\_tb.applicationNO=manage\_tb.applicationNO where ";

string pagePart = "";

string requirementPart = "";

switch (pageName)

{

case "studentManage":

pagePart = "(checkStatus='' or checkStatus='checking')";

break;

case "approvedList":

pagePart = "checkStatus='checked'";

break;

case "feedbackList":

pagePart = "(checkStatus='edit' or checkStatus='back')";

break;

case "studentList":

pagePart = "(checkStatus='edit' or checkStatus='back' or checkStatus='' or checkStatus='checking' " +

"or checkStatus='checked' or checkStatus='refuse')";//1=1会出现重复数据的bug

break;

case "refuseList":

pagePart = "(checkStatus='refuse')";

break;

default:

pagePart = "(checkStatus='edit' or checkStatus='back' or checkStatus='' or checkStatus='checking' " +

"or checkStatus='checked' or checkStatus='refuse')";//1=1会出现重复数据的bug

break;

}

if (dataDictionary != null)

{

//针对studentlist和导出数据的审核状态查询，对等待修改做处理

if (dataDictionary.ContainsKey("checkStatus"))

{

if (dataDictionary["checkStatus"]["columnValue"] == "edit")

{

dataDictionary["checkStatus"]["columnValue"] = "edit' or studentsRegister\_tb.checkStatus='back";

}

}

foreach (string columnName in dataDictionary.Keys)

{//生成查询条件sql语句

if (columnName != dataDictionary.Last().Key)

{

if (!String.IsNullOrEmpty(dataDictionary[columnName]["columnValue"]) &&

dataDictionary[columnName]["columnValue"] != "All")

{

requirementPart += dataDictionary[columnName]["tableName"]+"." + columnName + "='"

+ dataDictionary[columnName]["columnValue"] + "' and ";

}

else

{

requirementPart += " 1=1 and ";

}

}

else

{

if (!String.IsNullOrEmpty(dataDictionary[columnName]["columnValue"]) &&

dataDictionary[columnName]["columnValue"] != "All")

{

requirementPart += dataDictionary[columnName]["tableName"]+"." +columnName + "='"

+ dataDictionary[columnName]["columnValue"] + "'";

}

else

{

requirementPart += " 1=1 ";

}

}

}

if (String.IsNullOrEmpty(requirementPart))

{

requirementPart = " 1=1 ";

}

sqlCount = "select COUNT(rownumber) " +theSqlStr+ "(" + pagePart + " and " + requirementPart + " ))as template";

selectEmail = selectEmail + theSqlStr + "(" + pagePart + " and " + requirementPart + " ))as template";

theSqlStr += "(" + pagePart + " and " + requirementPart + " ))as template where (rownumber between ";

}

else

{

sqlCount = "select COUNT(rownumber) " +theSqlStr + "(" + pagePart + " ) )as template";

selectEmail = selectEmail + theSqlStr + "(" + pagePart + " ) )as template";

theSqlStr += "(" + pagePart +" ) )as template where (rownumber between ";

}

theSqlStr ="select \* "+theSqlStr+ startPage+" and "+endPage+")";

string[] bacStrings=new string[3];

bacStrings[0] = theSqlStr;

bacStrings[1] = sqlCount;

bacStrings[2] = selectEmail;

return bacStrings;

}

### 5.3.6 审核信息并反馈意见

教师登录后台管理系统后，针对学生的四种不同的状态有相应的不同的四种操作。如果学生信息处于未审核或者正在审核阶段，教师选择未审核选项卡，选择学生，点击审核，进入审核页面。

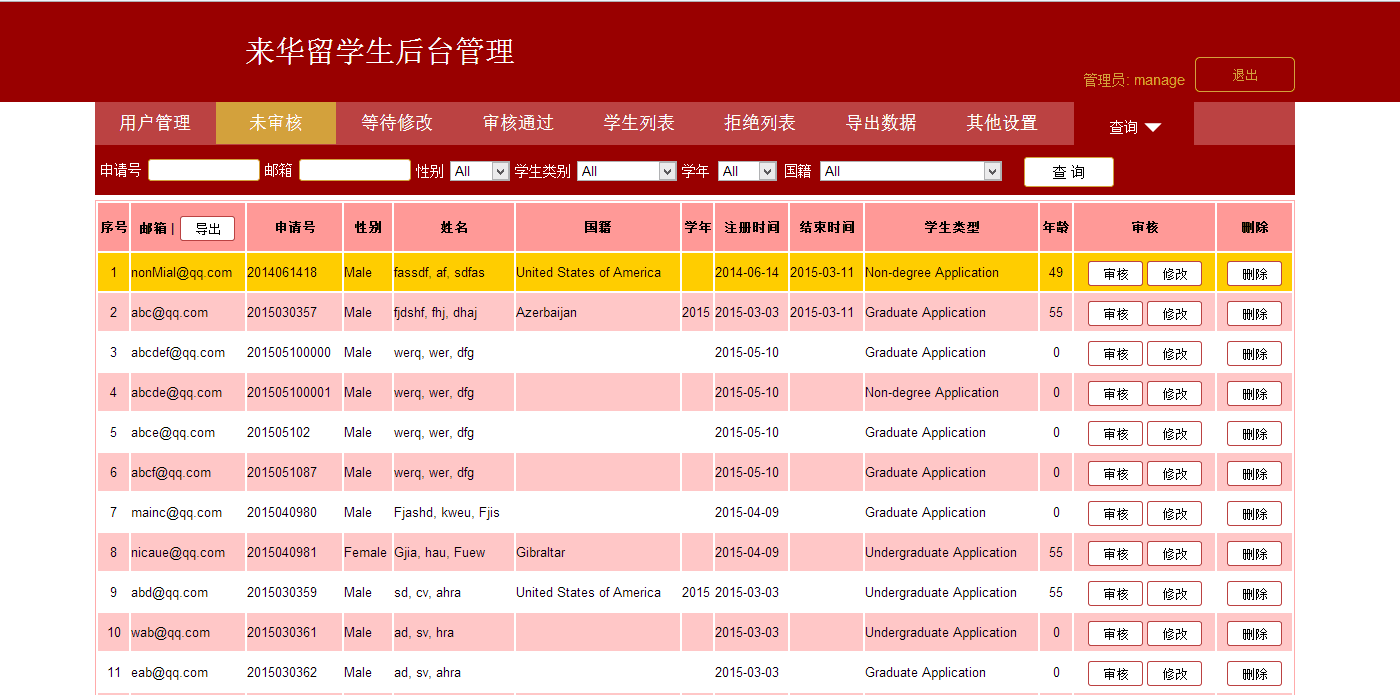


图5.26 准备审核图

审核页面的左侧依然是导航栏，可以直接搜素到学生填写的每一项申请信息，如个人信息、家庭信息、教育信息等。审核页面的主体是对于每一项信息的详细列表，一共四列，列，值，是否合格，反馈。这样教师可以选择每一条信息进行审核，判断其是否合格，如果不合格，则需要在反馈一列中写出反馈意见。值得注意的是，在左侧的导航栏下面有一个审核备注的文本输入框。这里主要填写对学生拒绝申请的理由，点击拒绝申请，则这个学生的研究生申请被拒，并列在了拒绝名单中。

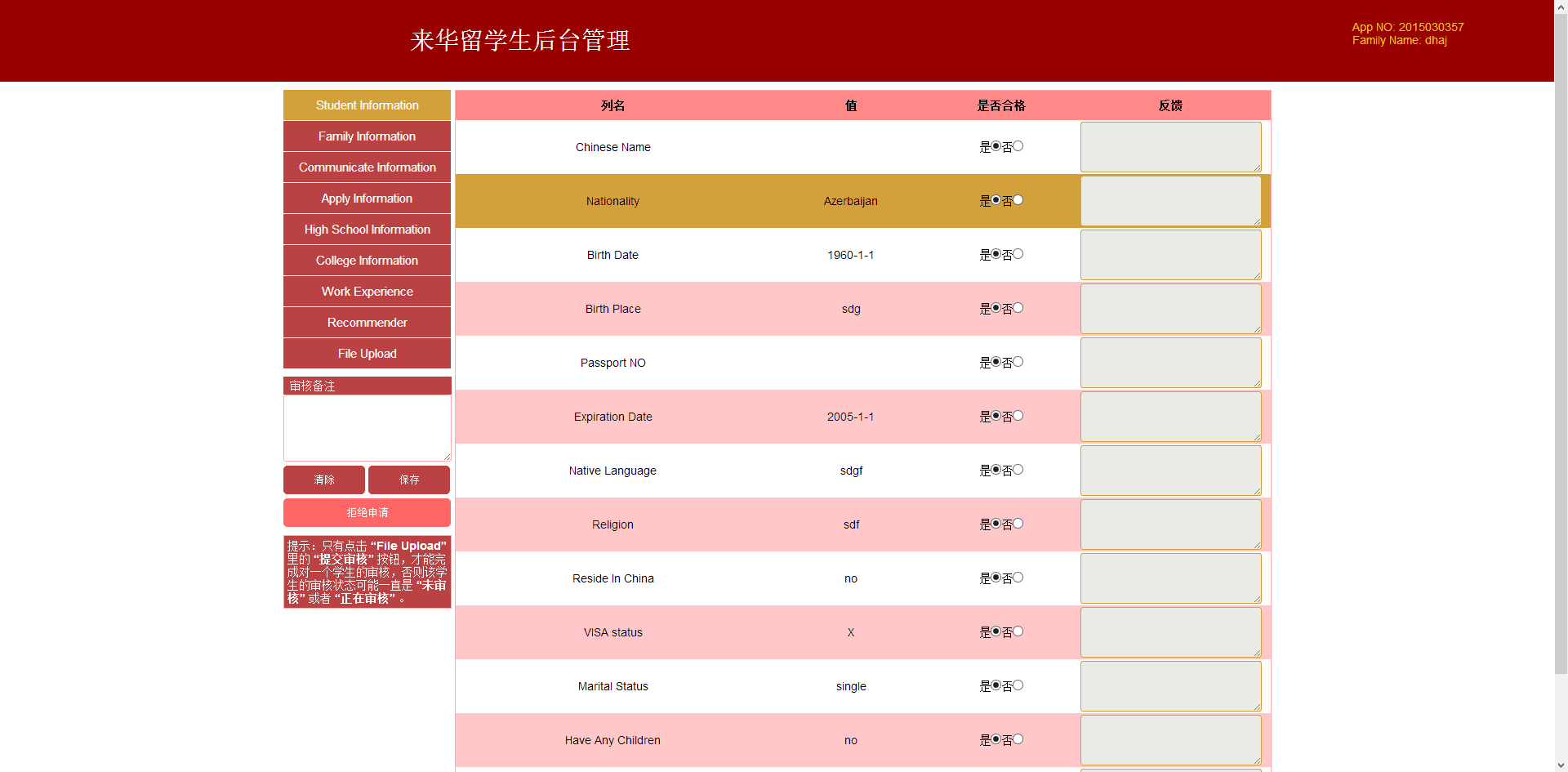


图5.27 正在审核图

### 5.3.7反馈后修改信息模块

学生申请信息较多较为繁杂，但小部分的中英文转换或格式问题，教师用户可以直接进行手动修改。在未审核选项卡下，选择学生，点击修改。或者在等待修改选项卡下，选择学生完成修改操作。进入修改页面后，即进入学生填写信息的页面，并对学生的信息手动修改，节省了与学生交互的时间，方便快捷。学生信息的导航栏下，已经添加了“Your Checklist”一项，这是教师审核后的结果及反馈意见。点击“Your Checklist”，可以看到“Feed Back”一项，这是一个列表，显示了，哪个字段，哪个值，需要怎样修改。点击“Back to Edit”按钮，即可返回填写信息页面，学生就可以有针对性的进行修改了。

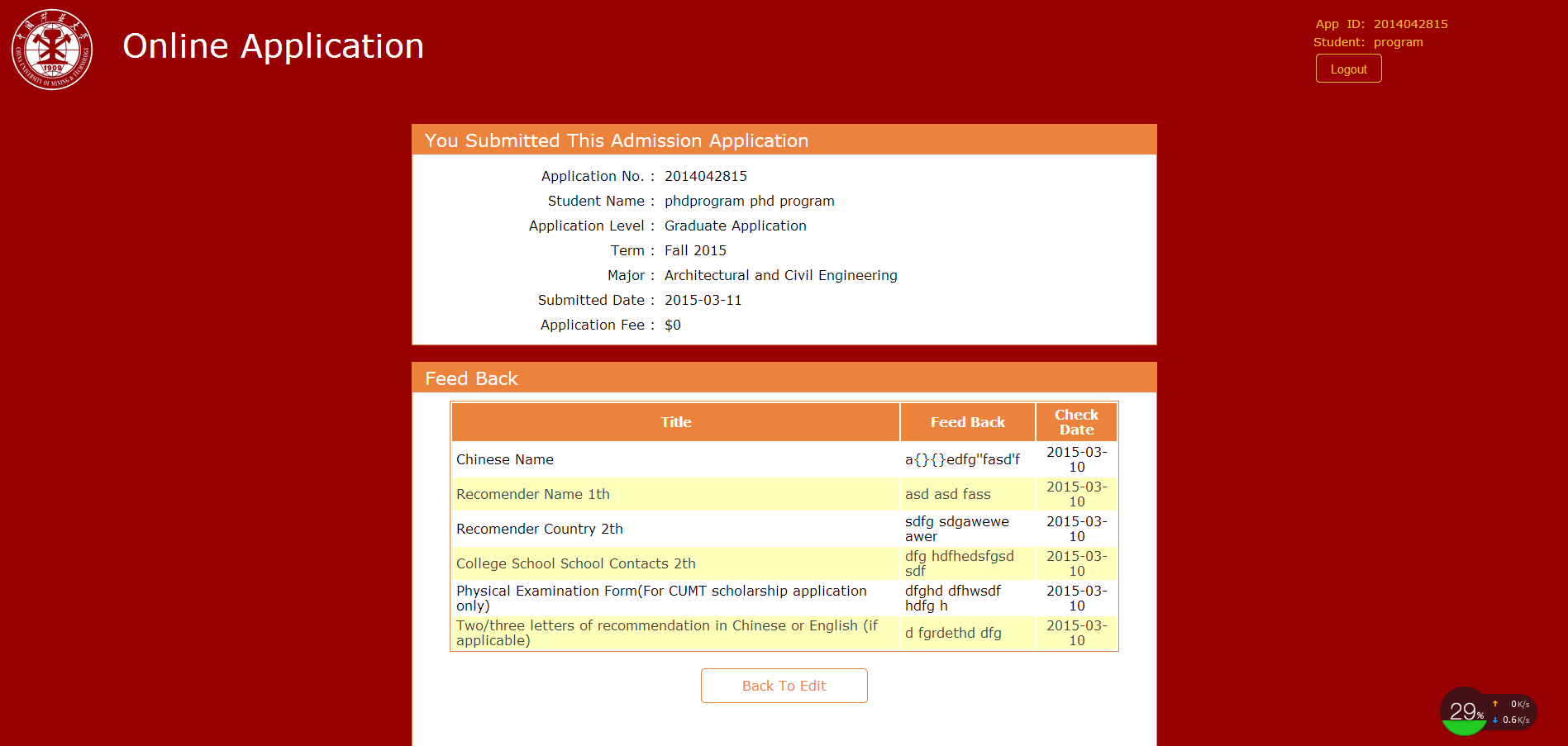


图5.28 审核结果反馈图

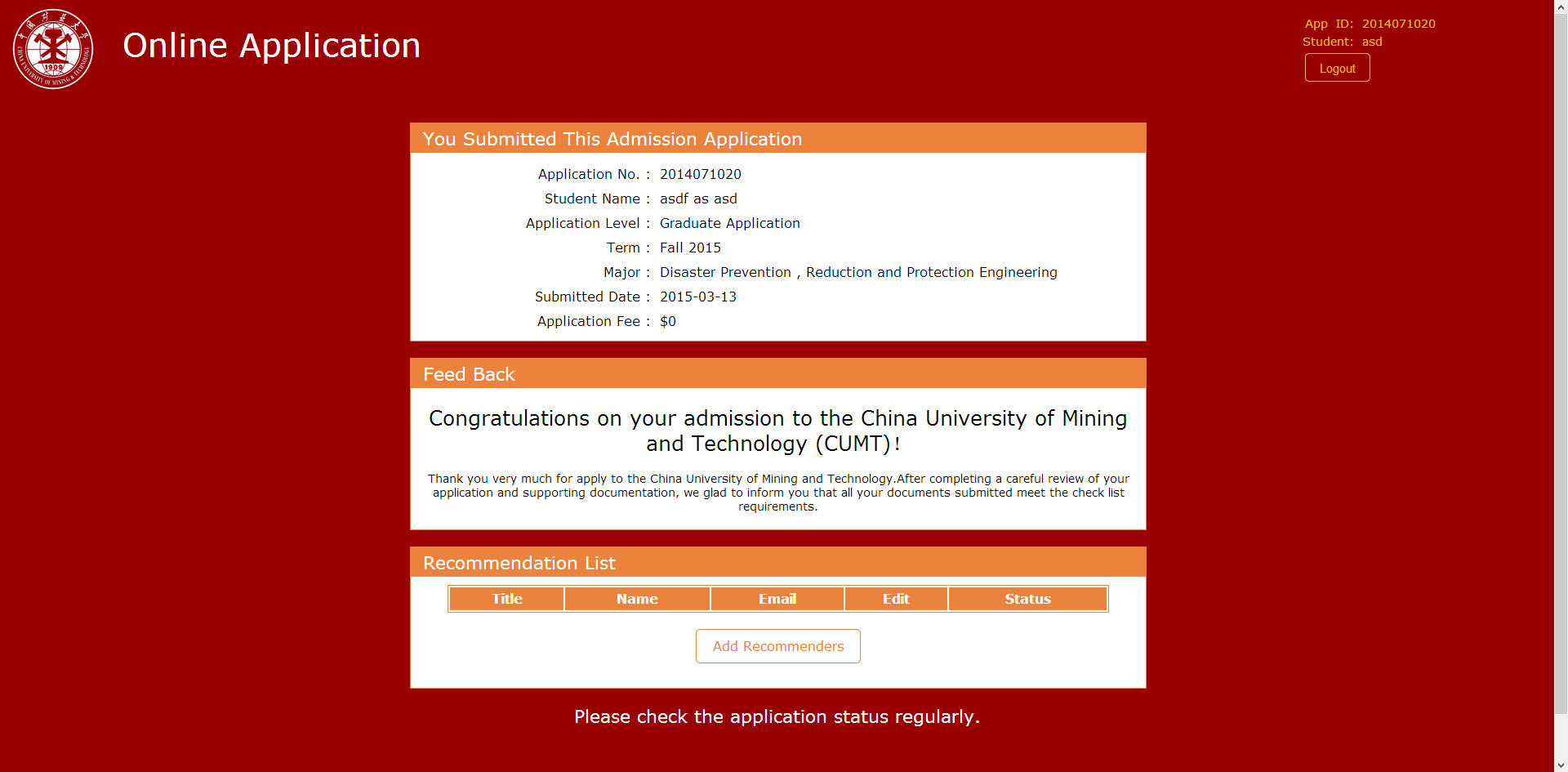


图5.29 审核通过图

## 5.4系统测试

作为系统开发的重要环节，测试越来越受到广大的开发者的重视。在整个系统完成之后，我们必须对其进行系统测试。测试是为了发现程序中的错误而执行程序的过程，尽可能多地找出程序中的错误，才能保证系统的开发质量，延长系统的生命周期。系统测试分为以下几个阶段，即代码审查、单元测试、集成测试和系统测试。代码会审是由一组人通过阅读、讨论和争议，对程序进行静态分析的过程。单元测试集中检查系统设计的最小单位——模块，通过测试发现该模块的实际功能与定义该模块的功能说明不符合的情况，以此发现编码的错误。集成测试是将模块按照设计要求组装起来，在组装的同时进行测试。系统测试时为了向未来的用户表明系统能够像预定要求的那样工作，达到未来的预期效果。本文针对单元测试进行了详细的介绍。

### 5.4.1系统测试目标

系统设计完成后，需要对已完成的系统进行测试，检验系统功能是否满足毕业设计的课题要求，以及各个功能模块能否按照需求正常运行。系统测试的具体目标如下：

（1）学生注册：当学生填写姓、中间名、名字、性别、邮箱、密码等注册信息后，点击注册按钮，是否能够注册成功，将信息写入数据库中。而当某些必填输入框为空或某些输入框格式不正确时，是否有系统提示消息。

（2）用户登录：本系统中涉及3个登录页面，其中包括了学生登录系统页面、教师登录后台管理系统和网上银行登录页面。学生登录系统页面中，当学生输入邮箱、密码和验证码等信息以后，点击登录按钮，是否能够登录成功。教师登录页面中，当教师填入账号、密码和验证码后，是否登录成功。网上银行登录页面中，当学生填写了卡号和密码后，是否登录成功。当输入相应信息为空或与数据库匹配不正确时，是否有相应的提示消息。

（3）基本信息录入：学生用户成功登录本系统后，学生需要根据相应的需求，填写基本信息，其中有个人信息、家庭信息、学习计划、教育信息、工作信息、资金信息和语言能力信息、推荐人信息，是否能够成功录入、仔细查看、认真修改并保存信息上传到数据库中。

（4）奖学金申请：学生用户登录系统，申请就读研究生的同时，可以进行奖学金申请，是否能够填入所需信息、修改并保存信息。

（5）上传文件与下载：学生用户登录系统后，点击上传按钮，选择所需上传的相应的文件或图片进行上传，是否能够正常上传，是否能够显示上传进度，是否能够完成上传，是否能够限制上传文件大小。教师用户点击导出数据后，是否能够导出生成的研究生申请表和研究生录取通知书；教师用户点击下载文件选项后，是否能够选择性地下载学生上传的申请证明文件信息并完成打包下载。

（6）在线支付：学生用户在选择所需支付的费用后，填写账号和密码登录网上银行界面，是否能够完成支付过程，是否能够实现转账、存款等操作。

（7）教师用户管理：教师用户成功登录本系统后，1级教师用户可以对2级和3级教师用户进行管理，是否能够增加、修改、删除2级和3级教师用户。

（8）跨表查询学生信息：教师用户登录本系统后，教师用户是否能够根据申请号、邮箱、性别、学生类别、学年、国籍进行跨表查询，快速便捷地搜索到所需的申请学生信息。

（9）教师用户审核信息并反馈：教师用户进行信息审核，是否能够逐条判断学生申请信息填写有无问题，如果有问题，则提出反馈意见，如何修改。是否成功提交反馈意见。

（10）教师用户自行手动修改学生的申请信息：当学生信息填写小有不妥，可以直接修改时，为了系统的灵活性，教师用户进行手段信息修改申请学生的信息，是否正确修改学生信息并保存。

（11）学生用户接受反馈意见并修改申请信息：学生登录系统后，是否能够查询教师给出的反馈意见，是否能够根据反馈意见逐条修改相应信息并保存。

### 5.4.2系统测试环境

系统测试使用的环境配置：

服务器端：

硬件需求：要求内存2G以上，CPU Pentium Ⅳ 2Ghz以上。

软件需求：必须安装Windows Server服务器操作系统，IIS 8.0（或更高版本）服务器平台软件，.NET Framework 4.0，SQL Server 2008（或更高版本）数据库引擎。

客户端：

因为是B/S系统，所以无特别要求，只要通过浏览器联网状态下即可访问网站。本网站兼容IE7/8，Firefox，chrome，搜狗等浏览器。

### 5.4.3系统测试分析

（1）学生注册：当学生填写姓、中间名、名字、性别、邮箱、密码等注册信息后，点击注册按钮，不仅能够注册成功，还将信息写入数据库中。而当某些必填输入框为空或某些输入框格式不正确时，系统提示了错误消息。

（2）用户登录：学生登录系统页面中，当学生输入邮箱、密码和验证码等信息以后，点击登录按钮，登录成功。教师登录页面中，当教师填入账号、密码和验证码后，也登录成功了。网上银行登录页面中，当学生填写了卡号和密码后，登录成功。当输入相应信息为空或与数据库匹配不正确时，系统有相应的提示消息。

（3）基本信息录入：学生用户成功登录本系统后，学生需要根据相应的需求，填写基本信息，其中有个人信息、家庭信息、学习计划、教育信息、工作信息、资金信息和语言能力信息、推荐人信息，能够修改、查看并保存信息。

（4）奖学金申请：学生用户登录系统，申请就读研究生的同时，可以进行奖学金申请，能够填入所需信息、修改并保存信息。

（5）上传文件与下载：学生用户登录系统后，点击上传按钮，选择所需上传的相应的文件或图片进行上传，能够正常上传，显示上传进度，还能够限制上传文件大小。教师用户点击导出数据后，能够导出生成的研究生申请表和研究生录取通知书；教师用户点击下载文件选项后，能够有选择性地下载学生上传的申请证明文件信息并完成打包下载。

（6）在线支付：学生用户在选择所需支付的费用后，填写账号和密码登录网上银行界面，能够完成支付过程，能够实现转账、存款等操作。

（7）教师用户管理：教师用户成功登录本系统后，1级教师用户可以对2级教师用户和3级教师用户进行管理，能够增加、修改、删除2级和3级教师用户。

（8）跨表查询学生信息：教师用户登录本系统后，教师用户能够根据申请号、邮箱、性别、学生类别、学年、国籍进行跨表查询，快速便捷地搜索到所需的申请学生信息。

（9）教师用户审核信息并反馈：教师用户进行信息审核，能够逐条判断学生申请信息填写有无问题，如果有问题，则提出反馈意见，如何修改；并能够成功提交反馈意见。

（10）教师用户自行手动修改学生的申请信息：当学生信息填写小有不妥，可以直接修改时，为了系统的灵活性，教师用户进行手段信息修改申请学生的信息，正确修改学生的申请信息并完成保存。

（11）学生用户接受反馈意见并修改申请信息：学生登录系统后，能够查询教师给出的反馈意见，并根据反馈意见逐条修改相应信息并保存。

### 5.4.4测试结果分析

根据以上的测试结果表明，本系统已经实现了设计的所有主体功能。系统可以正确无误地运行，操作简单，具有友好的用户界面。

# 6总结与展望

## 6.1总结

本系统为基于SSL的研究生申请审核及交费系统，已经实现了研究生申请、教师审核、学生交费等功能，并使用了SSL协议，保障了系统的安全性。本系统分为前台学生申请子模块、后台教师审核子模块以及支付子模块。这样，学生通过前台提交申请信息到后台，经教师审核通过后，再登录网上银行进行支付。

本系统采用B/S模式，在.net平台，使用C#语言，数据库为SQL Server2008，开发环境是Visual Studio 2012+IIS8.0。在此设计下的系统具有以下特点：

（1）可操作性：系统的操作应简单且无争议，且为用户提供了友好的界面，使用户具有良好的体验。

（2）实际性：根据我校国际学院的实际需求进行开发，针对性强，具有实际意义。

（3）安全性：作为一个具有申请报名及支付功能的系统，安全性是十分重要的。本系统对用户和管理员的登录进行必要的身份验证，尤其注重控制管理员权限及用户的相关权限，保证系统的正常运作不会受到人为因素的干扰。还有支付子模块建立在SSL安全协议下，保障支付的安全性。

（4）交互性：本系统的交互性体现在学生申请信息后，教师进行审核，可以提出反馈意见。学生可以再根据反馈意见进行信息的修改，使系统完善可靠，有不错的交互性。

## 6.2展望

由于能力和时间有限，本次毕业设计的研究生申请及交费系统中只是实现了大体框架内的基础功能，一些细节问题还没有落实。再者，系统功能上还可以逐步地充实和完善。比如支付子模块中，只限制使用网上银行支付，还可以引入第三方支付，如支付宝。系统中SQL语句的运用，只是实现了基本的功能，还可以进一步地优化。至于界面的美化工作，也还可以提升。

# 参考文献

[1] 王小科，刘莉莉.ASP.NET典型模块开发全程实录[M].北京：清华大学出版社，2013，1-31

[2] 明日科技，孙晓彤.ASP.NET典型模块与项目实战大全[M].北京：电子工业出版社，2012，261-278

[3] 吕玉珠.基于ssl协议的电子商务支付系统的实现和安全性研究[J].煤炭技术.2011（9）：119-121

[4] 陈秀琼. 基于ssl协议的网上支付模型改进研究[D].辽宁：东北大学.2012

[5] Stephen Webber,Kevin Hoffman.ASP.NET 4揭秘(卷1)[M]. 北京：北京人民邮电出版社，2011，29-134

[6] Moran SHI,Yongfei HAN,Yingjia LI. The Research of Improvement on SSL Protocol by Using PMI[A]. IEEE.Proceedings of 2011 4th IEEE International Conference on Computer Science and Information Technology(ICCSIT 2011) VOL01[C].IEEE,2011,4

[7] Araby Greene. Managing subject guides with SQL Server and ASP.Net[A]. Library Hi Tech, Vol. 26 Iss: 2:213-231

[8] Chun Yan Zhang. The Design of Campus Online Supermarket Management System [J].Advanced Materials Research,2014(25): 3316-3318

[9] Jiang Hua, Yang Jing. Security technology support system on the electronic commerce online payment [J]. WSEAS Transactions on Computers, 2007,6(5):813-820

[10] Adam Freeman,Matthew MacDonald,Mario Szpuszta. Pro ASP.NET4.5 in C#(Fifth Edition) [M].北京：人民邮电出版社,2014:100-207

[11] 邓淼磊.几类安全协议的研究与设计[D].西安：西安电子科技大学，2010

[12] 李继良.基于ssl协议的网银系统设计与实现[J].计算机工程与设计.2010（23）

[13] 张海燕.煤炭网电子支付中ssl协议的改进与实现[J].煤炭技术.2013（2）：251-252

[14] 赛奎春，顾彦玲.ASP.NET项目开发全程实录[M].北京：清华大学出版社，2013，72-125

[15] 明日科技.SQL Server从入门到精通[M].北京：清华大学出版社，2012，50-68

[16] 房大伟，吕双.ASP.NET开发实战1200例[M]. 北京：清华大学出版社，2013,120-124

[17] 余丽娜.基于ASP.NET的中小型商务网站设计与实现[J].信息通信.2014，5（1）：94-94

[18] 张洪星, 褚建立. 基于ASP.NET的SQL注入攻击及防范解决方案[J].电脑知识与技术.2006（35）

[19] Siddiqui Ahmad Tasnim,Singh Arun Kumar . Secure E-business transactions by securing web services [J]. Management of e-Commerce and e-Government (ICMeCG), 2012 International Conference on, 10.1109/ICMeCG.2012.11:79-84

[20] Wang Anbao, Pan Hailan. The teaching in ASP.NET programming and the development in E-commerce project [J]. Education Technology and Computer Science (ETCS), 2010 Second International Workshop on (Volume:1 ), 10.1109/ETCS.2010.111:386-389

[21] Nachila Catalin.Extending the ASP.NET membership schema for a SQL server survey database through IIS 7[A]. Knowledge Management and Innovation in Advancing Economies,2009,3:975-984

[22] Chen Li-fang. Achieve the system of the study of technical Chinese base on .NET and SQL[J]. Microcomputer Information,2009,9:183-193

[23] Haiyun Lin,Yujiao Wang. Implementation of college physics experiment Management system based on ASP+SQL [J]. Proceedings of the 2010 International Conference on Educational and Information Technology,2010:290-294

[24] Xia Zhou, Zhihong Zhuo.Research and Realization of ADO.NET Database Access Technology[J]. Applied Mechanics and Materials.2013: 1748-1751

[25] Jiang Wei, Lai Qing-gui, Qin Ling. Research and Application of Data Access Technology Based on ADO.NET[J]. Microcomputer Information.2010（30）:141-143

[26] 刁文广，郑卫东，舒云星. ASP.NET连接SQL Server数据库的实现[J]. 洛阳工业高等专科学校学报 .2007（02）

[27] 王婷婷. 基于ASP.NET和SQL技术的研究生调剂系统开发研究[D].云南：昆明理工大学，2011

[28] 王淼. 移动公司网上支付系统的设计与实现[D].黑龙江：东北石油大学.2013

[29] 宋少伟，刘方爱. 基于ASP的网上报名系统研究与设计[J]. 信息技术与信息化.2006（04）

[30] 罗惠平. 网上报名系统技术实现框架研究[J].科技创业月刊.2010（04）

[31] 陆国平，张晓涛，江莹. 硕士研究生招生网上报名的思考和设想[J]. 学位与研究生教育.2004（02）

[32] 朱伟民. 基于.NET的学位申请考试管理系统的设计与实现[D].辽宁：东北大学，2008

[33] 林莉萍. 学历硕士研究生网上报名系统的改进和建议[J]. 北京航空航天大学学报(社会科学版).2007（S1）

[34] 田晓芳. 基于数字化校园的留学生网上报名系统设计与实现[J].中国教育信息化.2011（03）

[35] 王曙光，张阳武，张晓庆. 全日制硕士研究生招生管理系统设计与开发[J]. 教育信息技术.2014（02）

[36] 陈林.网上报名系统设计与实现[D].辽宁：大连海事大学，2010

[37] 许春艳.网上考试报名系统的设计与实现[D].辽宁：大连理工大学，2009

[38] 石蕴伟.基于.NET的网络考试系统的设计与实现[D].北京：北京邮电大学，2008

[39] 马玉龙，熊巧荣. .NET应用开发概述.[A]，第六届全国计算机应用联合学术会议[C]2002年

[40] 王小伟.基于.NET B/S结构的数据库系统研发[D].甘肃：兰州大学，2007

[41] 白兆庆.基于B/S模式的选课系统的设计与实现[D].山东：中国海洋大学，2009

[42] 王慧，薄涛.浅析ADO.NET数据库编程[J].电脑知识与技术.2008（24）

[43] 伍红兵. .NET应用程序执行机制[J].微型机与应用.2003（09）

[44] 尹堃.浅谈ASP.NET技术的应用[J].硅谷.2009（07）

[45] 曾陈萍.基于.NET的高校设备管理信息系统的设计与实现[D].四川：四川大学，2005

[46]Wang Lei. Designation and Realization of the Pharmaceutical Management System Based on ADO.NET[J]. Microcomputer Information.2010（12）:51-53

[47] Zhang Ping, Zhang Yan, Wu Zhengjiang. IIS Security Mechanisms [J]. Proceedings of the Third International Symposium on Intelligent Information Technology and Security Informatics (IITSI 2010).2010:619-622

[48] Khalid, S., Abbas, H.,Pasha, M. Securing Internet Information Services (IIS) configuration files[A], 2012 7th International Conference for Internet Technology and Secured Transactions (ICITST)[C].2012年

[49] 钱哨，张继红，陈小全. SQL Server2008数据库应用案例教程[M]. 北京：北京大学出版社,2012:296

[50] 张咪.SQL Server2008性能优化实践探讨[J]. 计算机光盘软件与应用 .2014（05）:310-312

[51] 刘俊强.SQL Server2008入门与提高[M]. 北京：清华大学出版社.2014,444

# 翻 译

## 英文原文

The Analysis of the Safety Defects Based on ASP.NET

Zhou Jing

Abstract.

This paper analyzes the study of the ASP.NET application process, three elements of security (authentication, authorization, fake) their own characteristics; Secondly, in detail ASP.NET environment based on common security vulnerabilities and security architecture asp.net environment; Finally, the study of the IIS and ASP.NET authentication mode application, and authorization mechanisms, and their intranet and internet environment to achieve the security policy configuration. As Internet application developers, we should recognize that ASP.NET applications use a lot of users, a number of potential uses and a variety of different security requirements, so the application process to ensure the appropriate use of defense mechanisms, it also To ensure efficiency.

Keywords: Architecture, defects, security, process, strategy.

1 Introduction

As Web applications become increasingly widespread, more and more systems and services using B/S structure. At the same time high performance and reliability of the application process has been the goal pursued by the developer, how to make better use of share some resources to better improve application performance and reliability of their developers have become key issues for consideration. With the intranet and internet applications and popular in all walks of life, security is a growing web application developer concerns, but to protect the safety of web applications become more complex, developers using ASP.NET technology, on the one hand by the windows 2000 and IIS of the security features of security[1], on the other hand, ASP.NET technology itself provides authentication, authorization and other security features.

2 ASP.NET's Security Architecture

ASP.NET and IIS, .NET Framework and provides the basis for the operating system used in conjunction with the security services, together provide a range of authentication and authorization mechanism, known by the architecture, ASP.NET, security involves secure communication, authentication and authorization, secure communication technology to ensure the HTTP request, the security of sensitive data, and users to access limited resources must be authenticated and authorized, including the IIS and ASP.NET authentication and authorization, IIS and ASP.NET each contain between, common to good ASP.NET application security[2].

Authentication is the identification from the client to obtain credentials, such as user name and password, and verify those credentials process. Authorization is the process of determining user access privileges, that is to determine the authenticated user access to the requested resource type. The resources required to authenticate as limited resources, does not require authentication is called non-constrained resources of resources, can be anonymous called.

3 ASP. NET Security Process

ASP.NET and the Microsoft .NET Framework and IIS is to work which together to provide the security of web applications. First is the Windows operating system and IIS on the user's identity and permissions requests verification, and security is the ASP.NET validation framework[5].

There are three elements are the basis for security implementation:

1. Authentication

Authentication is the user credentials from the client to obtain and verify the credentials of the process. The purpose of authentication is to confirm the identity of the client, the identity is verified, the authorization process will determine the identity can access a given resource. Windows authentication is completed through IIS, all the web clients through IIS with ASP.NET application, communication, all requests need to go through us and then sent to ASP.NET application to verify the request. If the requirement resourced to allow anonymous access that is not authenticated. If the requirement resourced to requires the user has certain privileges while customers have been verified and authorized, then IIS will send the request to the ASP.NET process to deal with, if the status is not authorized by or are not denied access. IIS always maps to the windows user account credentials and use it to authenticate users. In IIS, there are three different types of authentication: basic, digest and integrated Windows authentication[6].

2. Licensing

The purpose is to determine whether authorization should be granted to a user on a given type of resource access request. There are two basic ways to grant access to a given resource: file authorization and URL authorization.

After authentication the user to determine the identity, but also the decision by the authorized processing module whether this user has access to this resource. Authority is divided into "file under" and "URL authorization", and each authorization method is divided into user-based authorization and role-based authorization, role-based authorization can greatly simplify the licensing workload.

Document authorized by file authorization module executed by windows operating system access control list (ACL) [7] to control user or group on the file or folder access rights to achieve, windows authentication mode of user or role (group) is authorized by file authorization to implement, and document authentication, authorization forms and passpon (passport) does not work even under authentication. Authorization implementation file, the file system must be configured to NTFS format, FAT32 format can not be achieved in the ACL.

3. Fake

Fake identification in the context of other users to execute code of the process, disabled by default fake, Win2000 environment, all ASP.NET code is in the Domain / ASP.NET runs under the user account can be set in the web.config <dentity impersonate = "tme" /> to enable fake. The main purpose is to simplify the fake work authorization rather than authentication. Note that only windows authentication mode can use the fake, and so on in the form authentication mode is not fake at all.

4 Implementation ASP.NET Security Policy

1. The security program of intranet application

When developing an application is running in the intranet environment, and access to the users’ windows account, the application authentication on the IIS to perform, and if the client is using Microsoft Intranet Explorer, then use the integrated windows identity verification.

Internet's security configuration is as follows:

Authentication: (1)ASP.NET Web application configured to use windows authentication, no simulation: the virtual path editing application directory, such as web.config. <authentication> element is set to <authentication mode="windows/"> which to ensure that analog is turned off: < identity impersonate = "false"/> (2) by IIS using integrated windows authentication against Web application's virtual root directory for anonymous access[8].

Authorized to:(1)ASP.NET Web application is configured to use the "file under"; (2)to configure the web server user license the right to limit the use of resources, in order to simplify the management of the user to the windows group and the use of groups in the ACL; (3) ASP.NET web application according to documents related to permissions that using the client's identity implementation of the access check.

The implementation of the IIS integrated windows authentication to use internet explorer. In a mixed browser environment, you can use other authentication scheme, such as basic authentication and SSL, client certificates, forms authentication, in these programs, the authentication must use encrypted communications: SSL (Secure Sockets Layer, Secure Sockets Layer).

2. The security program of internet application

Most Web applications are based on intemet users, the application have the following characteristics: ①users have many different browser types; ②anonymous users can browse the unrestricted application page;③users must register or log in (through HTML form) to have access to restricted pages;④SQL server database according to validate the user credentials[9].

Based on these features, common safety measures are:

Authentication:(1)HS configured to allow anonymous access;(2)ASP.NET web applications configured for forms authentication:Edit the application's virtual root directory web.config.<Authentication> element is set to: <authentication mode="windows/">;(3)use the database to store user name and password, but password is not stored directly, but to convert the value of the password hash with a salt value, in order to reduce the threat associated with dictionary attacks to verify.

Authority: the ASP.NET Web application is configured to "URL authorization" .

Forms Authentication, you must use SSL to protect the initial login credentials, the same time, the resulting forms authentication ticket must be protected, you can use SSL for all pages to protect the votes, it can be web.config surface element protection attribute is configured to All or Encrypt, to encryption. Web server URL authorization allows unauthenticated users to view web pages unrestricted, and restricted the page to force authentication.

5 ASP.NET Application, Common Security Flaw

ASP.NET programmers to bring a vast new program of space, but because it contains many new technologies, in terms of each There are still a lot of programmers need to learn, programming, they also inevitably some flaws .

1. Excessive trust of user input

ASP.NET that page as an intermediary with the user interaction, the user will enter the website legal information, and therefore the user's input does not carry out verification and inspection, in fact, now many hackers will attack with a lot of methods and tools to the input box, enter the specific information with a bad purpose, to achieve its other objectives. This vulnerability is easy to be hackers to SQL injection attacks and XSS attacks, and thus defrauding the trust and access to confidential data on servers and other hazards. This vulnerability is the low level of security loopholes.

2. The use of information leakage caused by Cookie储存在用户本地终端上的数据

To use the Cookie, do not need to programmatically create and read them explicitly. If you use session state and forms authentication implementation will implicitly use the Cookie[3]. Of course, ASP.NET support without Cookie session state, but if ID can be embedded in the URL, so more vulnerable.

3. Pass parameters from the URL security vulnerability

Using a variety of development languages used by Web programmers almost all URL parameters passed by value, with the ASP.NET programming, too, it can easily get to the next page on the page associated with a value, simplify code writing, however, it vulnerable to hacker attack vulnerability.

4. The page directly Script Language

Many programmers are accustomed to a web page (that is .Aspx file) [4] to add language to deal with Script pages the user insertion of information (such as form input), without any network transfer data back and forth, so when the user enters a items of information, it need not pass through the server (Server) address, and transmits it back to the process, the client directly can be handled by the application. This can speed up access to a good speed.

Since it is generally only the client implementation, and its source code client can directly see, with the server client is not necessary connection, but it can be in the client offline, so hackers can easily attacked, so from security considerations should be avoided.

6 Conclusion

This paper discusses the asp.net environment based on common security flaws and security architecture asp.net environment. Also studied the application IIS and ASP.NET authentication mode and authorization mechanisms, and their intranet and internet environment, security configuration and did not be related to the background database security issues. But as the internet application developers, we should recognize that ASP.NET applications use a large number of users, many potential uses and a variety of different security requirements, so the application process to ensure the appropriate use of defense mechanisms, while must also ensure efficient.

## 中文译文

基于ASP.NET的安全缺陷分析

摘要

本文分析了ASP.NET应用程序的研究，安全三要素（身份验证，授权，伪造）各自的特点；其次，详细介绍ASP.NET环境下基于常见的安全漏洞和安全架构ASP.NET环境；最后，介绍了IIS和ASP.NET身份验证模式的应用研究和授权机制，以及其局域网和互联网环境下实现安全策略配置。作为互联网应用的开发者，我们应该认识到ASP.NET应用程序使用了大量的用户，许多潜在的用途和各种不同的安全要求，所以应用程序中确保防御机制的适当使用，并确保效率。

关键词：结构，缺陷，安全，程序，策略。

1引言

随着Web应用变得越来越广泛，越来越多的系统和服务采用B/ S结构。同时，高性能和高可靠性的应用程序已成为开发者追求的目标，如何更好地利用共享一些资源更好地提高应用程序的性能和可靠性已经成为开发者考虑的关键问题。随着各行各业的局域网和互联网的应用和普及，安全是逐渐变成了Web应用程序开发人员的关注焦点，但保护Web应用的安全性变得越来越复杂，开发者使用ASP.NET技术开发，一方面考虑到Windows 2000和IIS的安全功能[1]，另一方面ASP.NET技术本身，提供身份验证，授权和其他安全功能。

2 ASP.NET的安全体系结构

ASP.NET和IIS，.NET框架通过已知的ASP.NET架构，提供了对操作系统的使用与安全服务相结合的基础上，提供一系列的身份验证和授权机制，安全服务包括安全通信，身份验证和授权等安全通信技术确保HTTP请求，敏感数据的安全，以及用户访问有限的资源，必须通过身份验证和授权，包括IIS和ASP.NET身份验证和授权，IIS和ASP.NET彼此之间都包含了不错的ASP.NET应用程序的安全性[2]。

身份验证是从客户端获取证书的标识，如用户名和密码，并验证这些证书的过程。授权是确定用户的访问权限的过程，这是确定身份验证的用户访问请求的资源类型。如果请求的资源是有限制的资源需要进行身份验证，如果为无限制资源则不需要身份验证，可以匿名调用。

3 ASP.NET安全过程

ASP.NET和Microsoft.NET Framework和IIS的一起合作，提供web应用的安全。首先是Windows操作系统和IIS的用户的身份和权限请求验证，和ASP.NET验证框架[ 5 ]。

有三个要素是安全实施的基础：

1.身份验证

身份验证从客户端的获取用户证书并验证它们的过程。身份验证的目的是确认客户的身份，待身份验证后，授权过程将决定该权限是否可以访问给定的资源。通过IIS，Windows身份验证完成后，所有Web客户端通过IIS与ASP.NET应用程序通信，所有的请求都需要经过我们，然后发送到ASP.NET验证请求。如果请求的资源允许匿名访问，则不需要进行身份认证。如果要求的资源需要用户有一定的权限，而用户进行验证和授权，然后，IIS将请求发送到ASP.NET程序来处理，如果状态是没有授权则拒绝访问。IIS总是映射到Windows用户一个帐户的证书，并使用它来验证用户的身份。在IIS中，有三种不同的类型：基本身份验证，摘要验证和混合Windows身份验证[ 6 ]。

2.许可

目的是确定是否授权给请求访问给定资源的用户。有授予访问给定资源的两种基本方式：文件授权和URL授权。

验证用户的身份确定后，而且被授权处理模块决定该用户是否能访问该资源。权限分为“文件”和“URL授权”，每个授权方法分为基于用户和基于角色的授权，基于角色的授权可以大大简化许可工作量。

授权由操作系统访问控制列表（ACL）的窗口中执行文件授权模块 [7]，以控制用户或组上实现文件或文件夹的访问权限，用户或角色（组）Windows身份验证模式是由文件授权、授权实施和文档认证，授权的形式不在身份验证工作中。授权执行文件时，文件系统必须配置为NTFS格式，FAT32格式不能在ACL实现。

3.伪造

伪造识别在其他用户的上下文中执行代码的过程，默认情况下禁用伪造，WIN2000环境，所有的ASP.NET代码在域/ ASP.NET运行在用户帐户可以在Web.config < dentity impersonate = "tme" />识别伪造。主要目的是简化伪造授权而不是伪造身份验证。注意：只有Windows身份验证模式可以伪造，在form验证等验证方式中都不会出现伪造的情况。

4执行ASP.NET安全策略  
1.局域网应用的安全程序

当开发一个在局域网环境下运行的应用程序，并获得用户的Windows帐户，在IIS服务器进行应用验证，如果客户端使用微软的局域网浏览器，然后使用混合Windows身份验证。

局域网的安全配置如下：

身份验证：（1）ASP.NET Web应用程序配置为使用Windows身份验证，没有模拟的虚拟路径编辑应用程序目录，如web.config. < authentication >元素设置为< authentication mode="windows/">要确保模拟是关闭的：< identity impersonate = "false"/>（2）由IIS使用针对于Web应用程序的虚拟根目录的集成Windows身份验证匿名访问[8]。

授权：（1）ASP.NET Web应用程序配置为在“使用”文件；（2）配置Web服务器用户权限限制资源的使用，以简化的Windows组的用户和组在ACL的使用管理；（3）ASP.NET Web应用程序相关的权限以访问检查客户的身份执行文件。

对IIS混合Windows身份验证使用局域网浏览器中的实现。在一个混合的浏览器环境，你可以使用其他的认证方案，如基本身份验证和SSL，客户端证书，form身份验证，在这些方案中，认证必须使用加密通讯：SSL（安全套接层协议）

2.互联网应用的安全程序

大多数Web应用程序都是基于互联网的用户，应用程序具有以下特点：①用户有许多不同类型的浏览器；②匿名用户可以浏览无限制的应用程序页面；③用户必须注册或登录（通过HTML表单）来访问SQL Server数据库④SQL Server数据库根据用户证书进行身份验证[ 9 ]。

基于这些特点，常见的安全措施：

身份验证：（1）HS配置为允许匿名访问；（2）ASP.NET Web应用程序配置为Forms身份验证：编辑应用程序的虚拟根目录下的web.config.<Authentication>元素设置为：<authentication mode="windows/">；（3）使用数据库来存储用户的名称和密码，但密码不直接存储，而是用固定值的密码哈希值进行转换，以减少与词典攻击相关验证的威胁。

权限：ASP.NET Web应用程序配置为“URL授权”。

Forms身份验证，您必须使用SSL保护初始登录证书，同时，由此产生的form身份验证凭据必须受到保护，你可以使用SSL对所有页面保护凭据，也可以是web.config中表面元素的保护属性配置为加密或所有加密。Web服务器的URL授权允许未经身份验证的用户查看不受限制的网页，而限制页面强制进行身份验证。

5 ASP.NET应用中常见的安全漏洞

ASP.NET程序员为新程序开发带来了广阔的空间，但因为它包含了许多新技术，在各方面仍有很多程序员需要学习，编程中他们也不可避免地存在一些缺陷。

1. 用户输入过多的信任

ASP.NET的页面作为与用户交互的中介，用户将进入该网站的合法信息，因此用户的输入不进行验证和检查，事实上，现在很多黑客会用很多的方法和工具攻击到输入框，以一些坏的目的输入特定的信息，来实现其他控制。这个漏洞很容易被黑客SQL注入攻击和跨站脚本攻击，从而骗取信任并访问服务器上的机密数据或造成其他危害。这个漏洞是安全漏洞的较低水平。

1. 通过Cookie造成信息泄漏的使用

使用cookie，不需要以编程方式明确地创建和读取它们。如果你使用session状态和form验证的实施将隐式地使用Cookie [ 3 ]。当然，ASP.NET支持无Cookie的session状态，但如果把ID可以嵌入在URL中，将更容易受到攻击。

1. 从URL的安全漏洞中传递参数

采用多种开发语言的Web程序员几乎所有URL参数按值传递， ASP.NET编程也一样，它可以很容易地进入一个值相关页面的下一个页面，简化代码的编写，但是，它是个容易受到黑客攻击的漏洞。

1. 直接的网页脚本语言

很多程序员习惯于网页（即.aspx文件）[ 4 ]添加语言处理用户插入的脚本页面信息（如表单的输入），无需任何网络来回传输数据，所以当用户输入一项的信息，它不需要通过服务器（服务器）的地址，并将其发送回的过程中，客户端直接可以由应用程序进行处理。这样可以加快访问速度。

因为通常只有客户端的实现，及其源代码客户端可以直接看到，且客户端与服务器无需连接，但它可以在客户端离线，所以很容易被黑客攻击，所以从安全考虑，应尽量避免。

6结论

本文讨论了基于常见的安全漏洞的ASP.NET环境以及ASP.NET环境下的安全体系架构。研究应用IIS和ASP.NET身份验证模式和授权机制，以及他们的Intranet和Internet环境，安全配置并没有对后台数据库的安全问题。但作为互联网应用的开发者，我们应该认识到ASP.NET应用程序使用了大量的用户，许多潜在的用途和各种不同的安全要求，所以应用过程中确保防御机制的适当运用，同时也必须确保高效。

# 致 谢

在此毕业论文结束，毕业设计完成之际，我要特别感谢我的指导老师——夏阳老师。他认真负责的指导和建议使我受益匪浅，他每时每刻的督促使我们不敢有丝毫的懈怠。每一次的毕业设计小组会，老师总是细心耐心地为我们解答疑惑，还指出毕业设计中我们需要注意的问题。这些都让我们非常感动。另外，我还要感谢叶枫学长。在我遇到问题时，学长总是放下手中的工作，耐心地对我进行指导，帮我解决了好多的问题。感谢学校老师在我四年的大学生活当中对我的教育与培养，没有你们的辛勤劳动，就没有我们今日的满载而归，感谢大学四年中曾经帮助过我的所有同学。最后，再次郑重感谢夏老师对我的耐心教导。