

摘 要

随着计算机网络技术的发展和学校办学规模的不断扩大,教务管理成为学校教育管理的重中之重,如何提高管理的效率,是实现教务管理现代化的必然要求。

本文首先简要地介绍了本系统的开发背景,接着介绍了系统使用的相关技术,在此基础上对系统的需求和总体设计进行了分析,对系统的实现步骤和功能进行了详细论述,最后对系统尚存在的问题进行了总结。

本文所设计的中学教务管理系统是针对天津市华杰中学教务管理的需求编制的,用于完成对日常的教务工作的管理。前台开发工具使用 Visual Basic 6.0 编程语言,后台数据库使用 SQL Server,建立在 Windows 系统之上,适用于 Windows2000/ XP/Vista 等版本。本系统根据使用者的特点来简化操作,进行布局;根据数据之间的关联性合理建表,以达到最佳的效果。

本系统采用流行的 ADO 数据库访问技术,并将每个数据库表的字段和操作封装到类中,从而成功地将面向对象的程序设计思想应用到数据库应用程序设计中,这也是本系统的特色和优势。

关键词: 教务管理系统; VB6.0; SQL Server

Abstract

Along with the development of the computer and web technology and the expansion of the school scale, the educational management system is gaining increasing importance and the modernization of educational management requires to enhance the administrative efficiency.

The thesis begins with a brief introduction of the developing background of the system. After introducing the techniques associated with the system, it analyzes the system requirements and the overall design. Then the operating process and functions of the system are expounded and finally the remaining problems are discussed.

The system discussed in this thesis is designed to meet the demands of daily educational administration of the Tianjin Huajie middle school. The front desk developing instrument of this system is Visual Basic 6.0 program language. The backstage supporter's database is SQL Server. Set up in Windows operating system, the management system is suitable for the editions such as Windows 2000/NT/XP/vista, etc. This system simplifies the operations and designs the operating interface for user's convenience. Tables are created in terms of the relationship between the data to optimize the functions.

Adopting the prevailing ADO database visit technology at present, the system puts the word and operation of database form into section, thus succeeds in applying the target-oriented design philosophy to the database and its practical programs, which constitutes characteristic and advantage of this system too.

Key words: Educational Management System; VB6.0; SQL Server

独创性声明

本人声明所提交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得天津师范大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名：寇薇薇 日 期：2008.5.20

学位论文授权使用授权书

本人完全了解天津师范大学有关保留、使用学位论文的规定，即：学校有权将学位论文的全部或部分内 容编入有关数据库进行检索，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。

(保密的论文在解密后应遵守此规定)

签 名：寇薇薇 导师签名：毕乃志 日 期：2008.5.20

第 1 章 教务管理信息系统概述

1.1 信息系统概述

1.1.1 信息系统的定义

信息系统是指计算机化的处理系统,它既能对数据进行处理,又能进行决策处理^[1]。它是由计算机硬件、网络和通讯设备、计算机软件、信息资源、信息用户和规章制度组成的,以处理信息流为目的的人机一体化系统。一般可以将信息系统的功能划分为输入、存贮、处理、输出和控制五个基本功能^[2]。下面对这五个基本功能分别略加说明:

输入功能: 决定于系统所要达到的目的及系统的能力和信息的许可。一般情况下,信息系统的输入内容包括信息资源的采集,控制指令的输入,信息检索条件的输入等等。

存贮功能: 指的是系统存贮各种信息资料和数据的能力。从实际出发尽可能少增加投资,扩大整体的存贮量,保证所存贮的数据资料充分有效,便于用户利用。

处理功能: 是信息系统内部的生产过程。一般将分散的处理业务集中统一进行处理。信息系统处理功能的大小,取决于系统内部的专业技术力量和信息处理技术设备的现代化水平。

输出功能: 包括经过信息系统加工处理后的资料信息,信息系统运行过程中状态的反馈信息,以及需要人工干预的提示信息。

控制功能: 体现在两个方面:其一是对构成系统的各种信息处理设备,如计算机,通信网,人员等进行控制和管理;其二则是对整个信息加工处理,输入输出等环节通过各种程序进行控制。时刻掌握系统预期要达到的状态和实际状态,不断使实际状态与程序规定的状态保持一致。

1.1.2 信息系统的发展

最初的信息系统,主要是信息处理系统,如图书情报中心等基于文档的信息检索系统。自 20 世纪 40 年代电子计算机问世以来,信息系统进入现代化系统阶段——计算机信息系统,并随着信息技术的飞跃以及现代社会对信息需求的增长,迅猛发展。美国哈佛大学教授查德·诺兰于 1979 年将计算机信息系统的发展道路划分为初始阶段,扩展阶段,控制阶段,统一阶段,数据管理阶段,成就阶段六个阶段^[2]。

1.2 管理信息系统概述

管理信息系统就是我们常说的 MIS(Management Information System),是由人和计算机等组成的,能进行信息收集、加工、保存、维护和使用的系统。它能利用历史数据预测未来,从全局出发辅助管理决策,利用信息控制企业行为,帮助实现规划目标。

管理信息系统有各种不同的分类方法:从层次上可以分为业务信息系统、管

理信息系统和决策支持系统；从系统的功能和服务对象，可分为国家经济管理信息系统、企业管理信息系统、事务型管理信息系统、行政机关办公型管理信息系统和专业型管理信息系统等；从计算机技术的应用程度，可分为人工管理信息系统、单机管理信息系统、基于网络的管理信息系统。

管理信息系统是现代管理方法和计算机技术相结合的系统。它不仅是对手工操作的简单模拟，而且需要融入现代的管理方法和手段。管理信息系统建设中常用的管理方法有：MRPII（制造资源规划）、ERP（企业资源规划）、JIT（准时制生产）、OPT（最优化生产技术）、Agile 制造和 BPR（业务过程重组）^[3]。

1.3 我国管理信息系统的发展

随着计算机技术的发展和应用，各种信息的计算机管理也日渐普及。计算机在信息处理方面的强大功能，为信息管理自动化提供了有效的技术手段。在过去的二十多年，我国有关单位通过实践和探索，先后推出了一个个符合实用所需、功能完备的管理信息系统。这些系统的管理运行，推动了现代企、事业单位的管理工作进程、完善了管理制度、提高了管理效率。

回顾我国管理信息系统发展进程，从宏观上看大体经历了三个阶段。第一阶段是单机检索。20 世纪 80 年代初到 90 年代，一些企、事业单位开始利用计算机管理本部门的信息，建立了一个个以单机为主要处理工具的管理信息检索系统，并取得了初步的管理成效和管理经验。在应用系统的开发中，大都采用 DBASE、BASIC、C、FOXPRO 等语言作为编程工具，由 DOS 操作系统支持。这一时期的应用特点是：信息录入数据简单，没有统一的标准格式；检索内容单一，数据处理能力有限，另外由于各单位和部门所采用的开发软、硬件环境不尽相同，因此，应用程序的适用性不够广泛。尽管如此，单机检索系统开掘了我国管理信息系统的先河，为全面推进管理信息系统软件的普及应用积累了许多宝贵经验。第二阶段是上世纪的最后一些年，是单机与局域网相结合的管理系统。此间，管理信息系统作为企、事业单位的计算机管理系统的一部分推出，并得到广泛的利用。系统开发主要有可视化开发工具 VisualFoxpro、PowerBuilder 和大型数据库管理系统 Oracle、Sybase、DB2、Informix 等，系统平台为 Windows、Unix、Linux，并建立了统一的数据格式和其它技术标准，使信息数据交换和管理软件共享成为现实。另外局域网技术普及应用发展迅速，推动了管理信息系统服务范围和服务水平的提高。第三发展阶段是上世纪末到本世纪初，广域网的应用范围以惊人的速度扩展，管理信息系统也正在搭乘 Internet 快车驶向未来。这一阶段信息管理单位正在纷纷利用 XHM、HTML 网页脚本语言和服务器端编程环境 ASP、PHP、JSP、ASP.NET 等在互联网上建立站点，通过 Access、SQL Server、MySql 数据库管理系统的支持，提供远程信息服务，让千里之外的用户轻松点击鼠标即可获得所需信息和电子文件。随着宽带互联网的开发应用，以声音、图像为主体的多媒体信息可以在网上快速传递^[4]。管理信息系统既是教育信息化的重要组成部分，也是教育信息化的基础。在推进教育信息化的过程中管理信息系统建设要先行一步，把管理信息化的基础打好。只有管理信息系统的根深，教育信息化这棵大树才能叶茂。可以预见，新的世纪管理信息系统将进入广泛普及应用、全面提升系统水平的辉煌时期。

1.4 开发教务管理信息系统的背景

知识经济作为影响当今世界发展的一个重要趋势，正在蓬勃兴起。知识经济

就是信息化的经济,是人类精神财富长期积累的产物,是当代高科技对社会日益深化的产物。充分地开发利用社会所有信息资源,真正实现社会中各种信息资源、信息技术资源的充分共享是信息社会的基本特征。这正是计算机被广泛应用于信息管理系统的的环境。计算机的最大好处在于利用它能够进行信息管理。使用计算机进行信息控制,不仅提高了工作效率,而且大大提高了其安全性。

随着知识经济时代的到来,社会信息化程度越来越高,知识总量剧增,知识更新的速度也越来越快,这使得教育的负担越来越重,如何评价学生在有限的时间内掌握的知识,这是摆在所有教育工作者面前的一大难题。在学校的日常管理工作中,教务管理是一个学校必不可少的部分,学校作为一个信息资源的集散地,资料繁多,包含很多的信息数据的管理。它将教学资源与老师,学生,课程,学生与老师,老师与课程,课程与学生相互间联系起来,而且联系错综复杂,数据信息处理工作量大,容易出错。由于数据繁多,资料易丢失,且不易查找。它的内容对于学校管理者来说至关重要,所以教务管理系统应该能够为我们提供充足的信息和快捷的查询手段。但一直以来人们使用纸张记录的形式来管理学生的学籍信息,成绩信息,档案信息,教师的资料信息,课程的信息等,这种管理方式效率低、保密性差,此外时间一长,将产生大量的文件和数据,很不易于查找、更新和维护。随着计算机的普及,人们迫切希望能在计算机上能够实现教务管理操作,把人们从繁琐的教务管理工作中解放出来。因此,教务管理系统应运而生。

1.5 开发教务管理信息系统的意义

教务管理信息系统是教学领域中的一个重要的系统。教务管理信息系统软件是用计算机控制教务工作由自动化系统运行而完成,它即具有智能化辅助教务人员管理教务工作的功能,又发挥了计算机能够存贮大量信息数据的优势。使用教务管理系统软件来进行教务管理工作,可以让教务管理人员从繁重的工作中解脱出来,并且能够准确无误,轻松自如地对所有教务工作进行管理。而且使用计算机对教务信息进行管理,具有手工管理所无法比拟的好处。例如:检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这不仅节省了人力,还提高了工作效率,而且增强了信息的流通。

因此,建立一个教务信息管理系统是时代发展的必然要求,它可以使教务管理工作规范化,系统化,程序化,避免管理的随意性,提高信息处理的速度和准确性,能够及时、准确、有效的查询和修改各种信息。

1.6 开发本系统的现实意义

目前,各高校为保证教学秩序的稳定,教学成绩的提高和学校的自身发展,纷纷建立自己的教务管理信息系统,使各项管理工作快速、方便、安全、准确地得以实现。然而,现在的很多中学至今仍采用传统的教务管理方法,教务工作过于依赖职工的经验 and 精力的付出,存在管理环节不紧凑,纸张满天飞,工作效率低下等问题。

我所工作的中学,以前的教务管理没有一个完整的系统,所有的学生档案信息,学籍信息,教师资料信息,成绩信息都是以纸张记录的形式来管理。随着近年来我国教育改革的深化,素质教育的全面推进,我校的办学规模不断扩大,教学资源日益紧张。对教务管理部门来说,不仅增加了工作量,更增加了工作难度,管理手段落后将直接影响教学质量和办学水平。如何充分利用紧缺的教学资源,提高资源的使用率,处理成倍增长的信息量,已成为管理职能部门关心的焦点,

对新的教务管理信息系统的需求也越来越迫切。

目前市场上也有不少商品化的中学教务管理信息系统，例如易友、北京世纪考联和长沙强智等软件公司开发的教务管理系统软件，经过仔细调研，发现这些软件公司开发的产品冗余信息多，有的所需功能不完备，无法体现学校的特色，也给教务管理工作带来很大的不便。因此，开发一个适合我校的教务管理软件是必然的要求。

第 2 章 开发教务管理信息系统工具简介

2.1 本应用程序的开发语言

如今流行的界面开发工具软件像 VB、JAVA, POWERBUILD、DELPHI、VC 等都能够实现强大的功能, 并且拥有良好的人机交互界面, 还可以与数据库进行连接, 成为许多软件开发者的首选前台开发软件。本系统最终决定采用 VB 作为前台开发工具主要有以下几个原因:

(1) 可视化的用户界面设计功能, 把程序设计人员从繁琐复杂的界面设计中解脱出来。传统的程序设计语言设计程序时, 都是通过编写程序代码来设计用户界面, 在设计过程中看不到界面的实际显示效果, 必须编译后运行程序才能观察。如果对界面的效果不满意, 还要回到程序中去修改, 这就大大影响了软件的开发效率。Visual Basic 提供了可视化设计工具, 把 Windows 界面设计的复杂性“封装”起来, 开发人员不必为界面设计而编写大量的程序代码。只需要按设计要求的屏幕布局, 用系统提供的工具, 在屏幕上画出各种控件, 并设置这些对象的属性。Visual Basic 自动产生界面设计代码, 程序人员只需要编写实现程序功能的那部分代码, 从而大大提高了程序的效率。

(2) 基于面向对象的程序设计思想。Visual Basic 把程序和数据封装起来成为一个个对象, 并且为每一个对象赋予各种属性, 使对象成为实在的东西。编程时, 不必编写建立和描述每个对象的程序代码, 而是用工具画出来, 系统自动生成对象的程序代码并封装起来。每个对象以图形方式显示在界面上, 都是可视的。

(3) 事件驱动的编程机制。在 Visual Basic 程序设计语言中, 对象与程序的交互是通过事件来完成的。即针对每一对象, 可能会有多种事件产生, 每一事件都可以驱动一段程序运行, 完成相应的工作。

(4) 结构化的程序设计语言。Visual Basic 是在 BASIC 语言的基础上发展起来的, 具有高级程序设计语句结构, 接近于自然语言和人类的逻辑思维方式。语句简单易懂, 其编辑器支持彩色代码, 可自动进行语法错误检查, 同时具有功能强大且灵活使用的调试器和编译器, 并且可以随时运行程序。而在整个应用程序设计好之后, 可以编译生成可执行文件(.EXE), 脱离 Visual Basic 环境, 直接在 Windows 环境下运行。

(5) 基于构件的程序开发。Visual Basic 程序设计中, 可以使用已经设计好的程序构件, 通过构件的重用和共享, 迅速建立应用程序系统, 完成应用程序的开发。

此外, 数据管理器(DataManager)、数据控件(DataController)以及数据访问对象(DAO)、ODBC API、远程数据对象(RDO)都具有极强的功能。

1、数据管理器

数据管理器是一个可以完全独立于 VB 之外的数据管理工具, 它具有 Jet 的数据库引擎, 既可以建立和修改数据库, 也可以实现对数据库的维护。它可以操作的数据库包括:Microsoft Access, dBase, FoxPro, Excel 和各种 ODBC 数据库。恰当的使用数据库管理器可以方便数据库数据的利用, 对系统的开发有很大帮助。

2、数据控件

所谓数据控件, 就是用来在数据库数据和数据感知控件之间提供连接方式的程序接口。而数据感知控件, 就是对数据有感觉, 通过建立与数据库的连接, 能

对数据的变化做出及时反应的控件。VB 提供的数据库感知控件包括 DBList, DBCombo, DBGrid, MSFlexgrid, Picture, Label, TextBox, CheckBox, Image 和 ListBox 等。与数据库控件建立了连接的数据感知控件, 能够自动的显示来自当前纪录的一个或多个字段内容。但是, 在建立数据库、数据库控件、数据库感知控件的相互关系时, 只需要少量的 VB 语言程序, 甚至只需要通过相应控件的属性设置就可以实现这一功能。

3、数据库访问对象

VB 所提供的数据库访问对象(DAO: Data Access Object)给予开发者较大的发挥空间。数据库访问对象是一种集合对象(Collection), 它所描述的对象是一个对象组, 其中的对象是层层包含关系。另外 DAO 还可以通过编程语言来实现数据的添加(AddNew)、更新(Update)、删除>Delete)等几乎所有的数据库操作, 并且使得对数据库的操作, 简化到了对特定集合对象操作的程度。

4、开放式数据库链接编程接口(ODBC)

开放式数据库链接编程接口作为一种接口, 可以提供大量的动态库函数, 这些函数存储在运行动态库 ODBC.DLL(16 位)或 ODBC.DLL(32 位)中。使用时, 只需在 VB 模块中加入 ODBC API 声明语句, 然后在 VB 编程时, 直接调用 ODBC API 函数就可以访问 ODBC 数据库。

此外, 已完成的应用程序是使用 Visual Basic 虚拟机的真正.exe 文件, 可以自由发布。

综上所述, VB 完全能够实现对数据库的复杂操作, 而且编程简单、方便实用, 因此开发中学教务信息管理系统时采用 VB 作为前端开发语言。

2.2 数据库管理系统的选择

虽然创建数据库可在多种环境(如 Access、Excel、Paradox、SQL Server、Oracle 等)下进行, 但从学校不断发展扩大方面的考虑, 本系统选择使用 Microsoft SQL Server 2000 中文版, 创建关系型数据库。

SQL Server 是一种关系型数据库(Relational Database), 它除了支持传统关系型数据库对象(如数据库、表)和特性(如表的 join)外, 也支持现今关系型数据库常用的对象如存储过程(Stored Procedure)、视图(View)等。另外, 它还支持目前关系数据库必须要支持的标准查询语言 SQL(Structured Query Language)。SQL Server 另外一个重要的特点是它支持数据库复制的功能, 也就是当在一个数据库上执行更新时, 可以将其结果传到远程 SQL Server 相同的数据库上, 让两边数据库的数据保持同步。

SQL Server 的重要特点是:

- (1) 真正的 C/S 体系结构。
- (2) 简单、友好的操作方式。
- (3) 丰富的编程接口工具, 为用户进行程序设计提供了更大的选择余地。
- (4) SQL Server 与 Windows NT 完全集成, SQL Server 还可以与 Microsoft Office 完美结合。
- (5) 具有很好的伸缩性, 可跨越从运行 Windows 95/98 的膝上型电脑到运行 Windows 2000/XP 的大型多处理器等多种平台使用。
- (6) 对 Web 技术的支持, 使用户能够很容易地将数据库中的数据发布到 Web 页面上。
- (7) SQL Server 提供数据仓库功能。

(8) 数据库类型转换支持功能, 包括 mdb, xls, txt 等。

综上所述, SQL Server 数据库功能强大, 安全性能高, 运行稳定, 易于维护, 因此开发的教务信息管理系统采用 Microsoft SQL Server 2000 为后台数据库。

2.3 程序与数据库接口

使用 Visual Basic 作为数据库的前端开发语言, 其访问数据库的方法很多, 一般情况下分成两大类: 接口对象法 API 和数据控件法, 而每一类又有很多方法。

接口对象法需要程序员编程控制数据库的建立连接、打开数据表、修改数据、关闭连接等所有数据库操作, 对程序员要求有相当高的数据库和编程技术。数据控件法是指编程工具已经将大部分数据库操作集成在控件中, 只需对控件进行简单的编程就可以满足所有的要求, 大大降低了编程难度、减少了编程所需时间。

VB 与数据库之间的接口常用的有 5 种方法, 即: ADO, ODBC API, DAO, ODBC Direct, DAO。ADO 是微软最新的数据访问技术, 它实际是一种提供访问各种数据类型的连接机制。它被设计用来同新的数据访问层 OLEDB Provider 一起协同工作, 通过其内部的属性和方法

提供通用数据访问(Universal Data Access)。它以简单的对象模型为特征, 能够更好地与 Microsoft 和非 Microsoft 的各种技术结合。ADO 具有较高的访问速度, 较低的内存占用, 熟悉的界面, 还提供了对 OLEDB 的 Automation 封装接口。

对于数据库程序设计人员而言, 最需要得到的就是一个简单、一致的应用程序编程接口(API), 从而使应用程序能够访问和修改众多类型的数据源。数据源可能像文本文件一样简单, 也可能像 SQL Server 2000 数据库那样复杂。此外, API 不应该预先设定访问和操作数据源的方式。典型的数据源依然需要支持开放式数据库连接性(ODBC)标准的关系型数据库, 并可通过结构化查询语言((SQL)编写的命令对它进行操作。

Microsoft 对这个问题所提供的解决方案是 OLEDB, 即一组“组件对象模型”(COM)接口, 以统一的方式访问存储在不同信息源中的数据。但是, OLEDB 应用程序编程接口的目的是为各种应用程序提供最佳的功能, 它并不符合简单化的要求。而 Microsoft Activex Data Objects (ADO)正是我们所需要的 API, 它是一座连接应用程序和 OLEDB 的桥梁。

ADO 就是为微软最新和最强大的数据访问范例 OLEDB 专门设计的, 是一个便于使用的应用程序层接口。OLEDB 为任何数据源提供了高性能的访问, 这些数据源包括关系和非关系数据库、电子邮件和文件系统、文本和图形、自定义业务对象等等。ADO 在关键的 INTERNET 方案中使用最少的网络流量, 并且在前端和数据源之间使用最少的层数, 所有这些都是为了提供轻量、高性能的接口。同时 ADO 使用了与 DAO 和 RDO 相似的约定和特性, 简化的语义使它更易于学习。

ADO 提供执行以下操作的方式:

(1) 连接到数据源。同时, 可确定对数据源的所有更改是否已经成功或没有发生。

(2) 指定访问数据源的命令, 同时可带变量参数, 或优化执行。通过涉及 ADO 的 Command 对象。

(3) 执行命令, 例如一个 Select 脚本。

(4) 如果这个命令使数据按表中的行的形式返回, 则将这些都地存储在易于检查、操作或更改的缓存中。

(5) 适当情况下, 可使用缓存行的更改内容来更新数据源。

(6) 提供常规方法检测错误(通常由建立连接或执行命令造成), 涉及 ADO 的 Error 对象。

在典型情况下, 需要在编程模型中采用所有这些步骤。但是, 由于 ADO 有很强的灵活性, 所以最后只需执行部分模块就能做一些有用的工作。例如: 将数据从文件直接存储到缓存行, 然后仅用 ADO 资源对数据进行检查。

以程序员的视角来看, 使用 ADO 对象要比 DAO 和 RDO 简单得多。最主要的一点在于, 程序员不用像在使用 DAO 和 RDO 那样要从对象模型的顶层开始一步步地创建子对象。因此 ADO 提供了一种更灵活的编程方式。

综上所述, 最终确定本系统调用数据库方法: 利用 ADO 数据库控件, 在 VB 中通过数据源(ODBC)实现对 SQL Server 2000 数据库的访问。

2.4 创建到 SQL Server 的新数据源

点击桌面左下角“开始”, 在菜单栏选择“控制面板”, 在“控制面板”里面选择“管理工具”, 左键双击打开后可见“数据源(ODBC)”, ODBC 是 Open Database Connectivity(开放式数据库链接性)的简称, 用于安装链接数据库的驱动程序和设置数据源的名称及位置。双击“数据源(ODBC)”, 打开“数据源”对话框。选择“系统 DSN”选项卡[DSN 是数据源名称(Data Source Name)的简称, 其数据源信息将保存于系统注册文件中], 单击“添加”, 按钮, 选择数据库“SQL Server”, 填写数据源名称“StudentSys”, 描述项填“教务管理数据表”并选择数据库的路径, 单击“确定”按钮, 建立数据源。建立后, 进行测试, 测试成功表明数据库已成功链接。

2.5 SQL 技术

2.5.1 SQL 简介

结构化查询语言(SQL)是工业标准数据库操作语言, 是 Visual Basic 与数据库引擎之间进行通讯的基本方法, 在大多数关系型数据库管理系统中, 都需要用到 SQL。

SQL 是一种数据库编程语言, 它的起源与关系数据库紧密相联。现代的 SQL 已经发展为关系数据库所广泛使用的标准, 并且被 ANSI 标准所定义。现在的标准是一个 SQL 查询至少要包括下面 3 个元素: 一个动词, 例如 SELECT, 它决定了操作的类型; 一个谓词宾语, 由它来指定一个或多个字段名, 或者指定一个或多个对象; 一个介词短语, 由它来决定动词在数据库中哪个对象上动作。当一个 SQL 语句被传送给一个基于 SQL 的查询引擎, 产生结果数据集合。结果集合以行(记录)和列(字段)的形式给出。

2.5.2 查询的类型

目前的 ANSI 标准认可 4 种执行 SQL 语句的方法:

(1) 交互式 SQL: 交互式 SQL 允许在一个类似于 Dbase 的点命令提示符下

输入 SQL 语句。交互式 Dbase 命令的使用与交互式 SQL 语句的使用非常类似。交互式 SQL 称为一个批过程。如果在 SQL 提示符下输入一个查询，查询结果将在计算机显示器上显示。

(2) 嵌入式 SQL: 大多数 SQL 语言都支持把 SQL 语句插入到普通的应用程序代码中。这样, 不一定使用固定的用户接口工具就可以通过 SQL 访问数据库。嵌入式 SQL 允许用户通过在 SQL 语句之前加入一个关键字来执行 SQL 语句。嵌入式 SQL 的优点在于进一步为单一变量赋予属性值, 其缺点是必须逐行地处理查询结果集合。

(3) 存储过程或模块 SQL: 存储过程由一些 SQL 代码组成, 当应用程序通过 Call 命令调用它时, 它会实现许多动作。利用 Visua Basic6.0 提供的数据库环境设计器可以方便地使用存储过程来执行各种操作。

(4) 动态 SQL: 动态 SQL 可以创建在编写语句时无法预测其内容的 SQL 语句。对数据库应用程序设计而言, 静态 SQL 和动态 SQL 都称为约束 SQL 语句方法。约束是指如何把 SQL 语句组合到源代码和目标代码中, 如何向 SQL 语句传送值以及如何处理查询结果集合。

2.5.3 SQL 语句

SQL 语句由命令、子句、运算符和合计函数构成, 这些元素结合起来组成语句, 用来创建、更新和操作数据库。

任何 SQL 命令都是以下面几种命令开头: SELECT、CREATE、DROP、ALTER、INSERT、DELETE 或 UPDATE。主要使用这些命令来指定所要进行操作的类型。

(1) SELECT 命令: 于在数据库中查找满足特定条件的记录。它是所有 SQL 命令中最常用的一个命令。SELECT 命令可以生成一个数据库中的一个或多个表的某些字段的结果集合。

(2) CREATE, DROP 和 ALTER 命令: 用于操纵整个表。其中 CREATE 命令用于创建新的表、字段和索引, DROP 命令用来删除数据库中的表和索引, ALTER 命令通过添加字段或改变字段定义来修改表。

(3) INSERT, DELETE 和 UPDATE 命令: 主要适用于单个记录。其中 INSERT 命令用于在数据库中用单一的操作添加一个记录, DELETE 命令用于删除数据库中已经存在的一个记录, UPDATE 命令用来修改特定记录或字段的值。

SQL 子句是用来指定条件的, 这些条件被用来定义要选定或要操作的数据。

SQL 语句中常用的几个子句 FORM、WHERE、GROUP BY、HAVING、ORDER BY 的功能分别为指定从其中选定纪录的表的名称、指定所选纪录必须满足的条件、把选定的纪录分成特定的组、说明每个组需要满足的条件、按特定的次序将纪录排序。

本系统数据库的操作是通过 SQL 语言来实现的。举一个通过从数据库的教师信息录入表中的“姓名”来查询某位教师信息的例子: "SELECT * FROM teachers WHERE Teacher_name = " _+ "" + query_tch + ""

第3章 系统分析

3.1 本中学教务管理信息系统简介

本系统是针对我校教务管理工作编制的。本系统通过对我校教务工作涉及的学生、教师、科研课题、课程、成绩信息等数据的采集、加工和处理, 为我校教务工作的实施和管理提供了有力帮助, 管理人员可以随时对相关信息进行查询、修改、添加和删除的操作。同时使学生、教师、课程、成绩等实体的管理机制规范化、系统化、科学化。在每一项的管理中, 本系统根据学生、教师、课程、成绩等实体的共性和个性将相关信息统一规范, 进而有效地管理各个实体信息。每个管理功能可以互不干扰的进行相关信息的查询, 但相互之间也可以实现信息的共享。

我校是一所初级中学, 日常的教务工作包括对学生档案的管理、学籍变动的管理、课程的管理、考试成绩的管理、教师教学工作的管理和教师承担的科研课题的管理。因此, 本系统分为班级管理、学生管理、课程管理和教师管理四个主要功能。

3.2 总体功能需求

需求分析是对系统必须符合的条件或具备的功能进行分析和阐述。需求分析的主要目的是对系统应该具有的功能和所涉及到的人员情况进行分析, 对系统进行定义, 对各个用例进行分析, 在软件要实现的功能上让用户满意。

通过在学校一线的大量调研工作, 与我校教务管理人员进行多次深入交流探讨, 确定了我校教务信息管理系统具备的条件如下:

(1) 建立客户机和服务器系统, 学校管理部门使用客户机系统, 进行日常办公管理工作, 服务器系统实现用户监控和接收客户端请求等操作。

(2) 为每一位用户设置密码, 提高系统的安全性, 防止学校重要信息资料外泄。

(3) 用户可以通过客户端系统主窗口菜单或工具栏完成系统所有功能操作。

(4) 录入的各种数据必须保证完整性与正确性。

(5) 数据库设计尽可能避免数据冗余。

(6) 能够根据学生或教师的姓名对数据库进行查询。

(7) 能够对数据库进行维护。

(8) 系统要求在 Windows 2000/XP 等不同操作系统下都可以正常运行。

(9) 系统运行要求有良好的可靠性, 不能出现运行错误自动退出的情况。

(10) 系统有良好的接口, 传输效率高、误代码低。

本系统力求界面简单、美观, 可操作性强, 操作员无需掌握过多的计算机理论知识, 只需对教务管理工作流程比较熟悉, 就可以比较轻松的掌握本系统的操作流程。

3.3 具体功能需求描述

1、学生基本信息管理功能:

每年学校都会有新的学生到来, 同时又有一些毕业的学生离开, 这涉及到对新生信息的录入、毕业生信息的更改以及其他年级学生相关信息的更改。因此, 本系统能实现针对学生基本信息(包括学号、姓名、性别、生日、所在班级等信

息)的录入、修改、删除、查询的操作和管理。

2、学生照片管理功能:

在学生的档案信息中,照片是不可或缺的资料,每年教务管理人员在为新生创建档案时,都需要为他们添加相应的照片,为此本系统具有对学生照片信息进行管理的功能。包括学生照片的录入、删除和查询的功能。

3、学生主要教育经历信息管理功能:

为了全面了解每一个学生的情况,以便在实际教学中更好地实现管理,同时更充分地发挥每一位学生的特长,学校需要知道学生详细的受教育经历,因此本系统具备对学生主要教育经历信息(包括开始日期、截止日期、所在学校、职务等信息)进行管理的功能,包括学生主要教育经历信息的录入、修改和删除的功能。

4、学生奖惩管理功能:

学生成长过程中所受的各种奖励,对于学生今后的发展至关重要,因此作为教务管理人员,需要全面掌握本校学生在校期间的各种受表彰和奖励的情况。为此本系统具有对学生奖惩信息(包括奖惩日期、奖惩内容等信息)进行管理的功能。包括学生奖惩信息的录入、修改和删除的功能。

5、学生学籍变动管理功能:

在学校实际工作中。某些学生可能会由于某种变故,需要在学习期间进行学籍的变更。为此本系统应具有对学生的学籍变动的管理功能,包括学生学籍变动信息的录入、修改和删除的功能。

6、课程设置管理功能:

学校课程设置的是否合理、高效,对于学校正常教学的开展至关重要,而对于具体操作该项工作的管理人员来说,这是一项非常繁琐的工作。因此本系统能完成对学校课程设置信息的全面管理,包括课程设置信息的录入、修改和删除的功能。

7、学生成绩管理功能:

学习成绩是全面掌握学生阶段信息情况、检验教师阶段性教学成果的有力参考,而对于实际操作该项工作的管理人员来说,需要处理的数据量太大。而且难保不出错。因此本系统具有对学校学生成绩管理的功能,包括学生成绩信息的录入、修改和删除的功能。

8、班级管理功能:

作为一个中学,涉及到的班级数量一般都较大,如何全面掌握各个班级的教师、学生情况,对于学校的日常管理尤为重要。因此本系统具有对学校的班级信息的全面管理的功能,包括对班级信息的添加、修改和删除的功能。

9、教师管理功能:

为了记录每位教师的情况,以能更好地对教师的教学和教科研工作实现管理,本系统具有对每位教师基本信息和教学信息的全面管理功能,包括对教师信息的添加、修改、删除和查询的功能。

10、系统用户管理功能:

在实际应用中,由于学籍管理信息系统都是由专人管理,所以用户登录系统,首先必须通过系统的身份认证,因此本系统能够实现对用户信息的管理功能。

3.4 系统的环境

3.4.1 软件环境

- (1) 操作系统:服务器端使用 Windows 2000 Server/XP
客户端使用 Windows 2000 Professional/XP
- (2) 数据库系统:Microsoft SQL Server 2000
- (3) 编程语言:Visual Basic 6.0

3.4.2 硬件环境

- (1) 处理器: Pentium II 166MHz 或更高的微处理器
- (2) 硬盘: 4GB 及以上
- (3) 内存: 建议至少 256MB
- (4) 显示器: VAG 或更高, 最佳分辨率: 800×600, 最好使用 1024×768
- (5) 其它: CD—ROM 驱动器、移动硬盘、鼠标等

3.5 系统的体系结构

3.5.1 局域网

随着计算机网络技术的发展, 计算机资源的共享已经成为现实。在一个企业中, 使用局域网, 采用客户机/服务器(Client/Server, C/S)架构的数据库应用系统, 不仅可以实现对数据库的共享, 同时也可以提高数据库的安全。

C/S 架构的应用程序是基于企业内部网络(局域网)的。在客户机/服务器架构的数据库应用程序中, 数据库和应用程序是分离的, 数据库存放在数据库服务器中, 应用程序则主要运行于客户端的工作站上。

局域网在使用网络技术前和网络技术发展的早期阶段, 大多数企业都拥有多台计算机、打印机和其他的附加设备, 不同的部门分别拥有这些设备。由于设备不能够共享, 所以必须配置多台打印机和其它设备, 造成设备的重复投资和较低的使用效率。在没有网络环境时, 文件资源是不能共享的。所有需要使用同一文件的人都必须在自己的计算机上拥有该文件的一个拷贝, 如果企业中有人对文件进行了更新, 其他人不能马上得到更新的版本。同时, 文件资源不能集中管理也造成文件和数据库安全方面的许多问题。

解决计算机资源(包括硬件设备和文件)共享的最佳解决方式就是使用网络。在网络上, 可以实现打印机、硬盘、文件和应用程序等资源的共享。

局域网是相对于广域网(也叫万维网)来说的, 是指一个企业或组织内部有限范围内的网络, 它由服务器和多个工作站组成。一个典型的企业内部网采用的总线拓扑结构, 有文件服务器、打印服务器、通信服务器、应用程序服务器、多台工作站微机和其他共享硬件资源。

3.5.2 C/S 架构

C/S 架构的数据库应用程序基于局域网络，包括连接在一个网络中的多台计算机。在 C/S 架构中，应用程序主要运行于客户端，它向数据库服务器提出请求、返回数据，并在本机上操作，然后将操作的结果发送到数据库服务器中；数据库服务器提供大容量的存储设备，管理数据资源，处理客户机的请求，也运行应用系统的服务器应用程序。这些处理应用程序，并请求另一台计算机响应其请求的计算机称为客户机，而处理数据库的计算机称为服务器。所有的用户都拥有自己的计算机来处理应用程序。在现今的 C/S 架构中，C/S 的应用程序一般不再处理有关网络协议、通信指令等问题，只要建立了与数据源的连接，就如同操作本地数据库一样方便。这些工作都是由中间的软件来完成的。

3.5.3 C/S 架构的特点

C/S 架构具有如下特点：运行性能高、便于最终用户使用、具有多功能的应用程序开发接口、开放的体系结构、良好的可扩充性。

鉴于 C/S 架构具有以上优点，在安全性要求高、交互性强、处理数据量大、数据查询灵活、地点固定的小范围内使用 C/S 架构模式比较方便。因此本系统采用 C/S 架构模式。

第4章 系统的总体设计

4.1 设计原则

本系统在设计时，遵循以下原则：

1、实用性原则

首先符合实际需要，教务管理工作是繁多而复杂的，一般的程序人员对这些业务的认知程度较低，为此，要做到实用性并不简单，对于软件开发来讲，没有学校方面的工作经验是很难做到的。因此，教务信息管理系统的实用性不仅仅是一个设计的问题，还要靠软件的不断升级来实现。

2、可维护性原则

系统运行后，维护工作将是一个长期的工作。因此，本系统设计充分考虑维护工作的需求：设计采用面向对象、模块化的技术，提高软件的复用性、界面的一致性、代码的重用性，尽量降低维护工作量及困难。

3、经济性原则

由于不同的系统设计会使计算机设备的应用配置和数量发生变化，为此，在系统设计阶段，对系统规模和功能范围必须有适度的界定。找到一种合理的折衷，达到适度，使系统能实现良性发展。规模适度是达到较好的性价比的有效方法。要考虑在满足学校现在及未来几年需求的前提下，如何减少学校的设备投入，如何解决将来的系统升级、扩充，同时还要考虑减少耗材的消耗等。

4、先进性原则

系统所有的组成要素均应充分地考虑其先进性。只有将当今最先进的技术和实际应用相结合，才能获得最大的系统性能和效益。因此，应尽量采用目前先进而成熟的管理技术及设备，系统软硬件均采用模块化结构，界面清楚。同时，配以先进和高效实用的系统软件和应用软件，使整个系统协调一致地运行。

5、系统兼容性和数据共享

要实现教务信息管理系统子系统间互相连接、相互制约、快速、准确地随时提供教务工作所需要的各种数据，支持教务工作中的各项基本活动，在设计时必须从系统论的观点出发，以实现系统总体功能为目的，构建子系统结构，解决各子系统之间信息高度共享。

6、可扩展性

系统应具有较好的可扩展性。系统的可扩展性应包括能接纳已有的系统和在今后系统软硬件扩展时，能有效地保护已有的资源。特别是在应用需求变化时(应用与系统的需求往往会变化)，有一个较好的应用平台，能容易的加以调整。系统易于扩充升级，既能满足当前的业务需求，又为今后的扩充留有空间。

4.2 系统的功能设计

学生基本信息管理功能——实现对学生的学号，姓名，性别，生日，所在班级等信息的录入、修改、删除和查询的功能。

学生照片管理功能——包括学生照片信息的录入、修改和删除的功能。

学生主要教育经历信息管理功能——包括对学生主要教育经历的开始日期，截止日期，所在学校，职务等信息的录入、修改和删除的功能。

学生奖惩的管理——包括对学生所受奖惩的日期，内容等信息的录入、修改和删除的功能。

学生学籍变动管理功能——包括学生学籍变动信息的录入、修改和删除的功能。

课程设置管理功能——包括课程设置信息的录入、修改和删除的功能。

学生成绩管理功能——包括学生成绩信息的录入、修改和删除的功能。

班级管理功能——包括对班级信息的添加、修改和删除的功能。

教师信息管理功能——实现对教师所教课程，任课班级，教科研等信息的录入、修改、删除和查询的功能。

系统用户管理功能——实现对用户信息的管理功能。

4.3 系统的功能模块设计

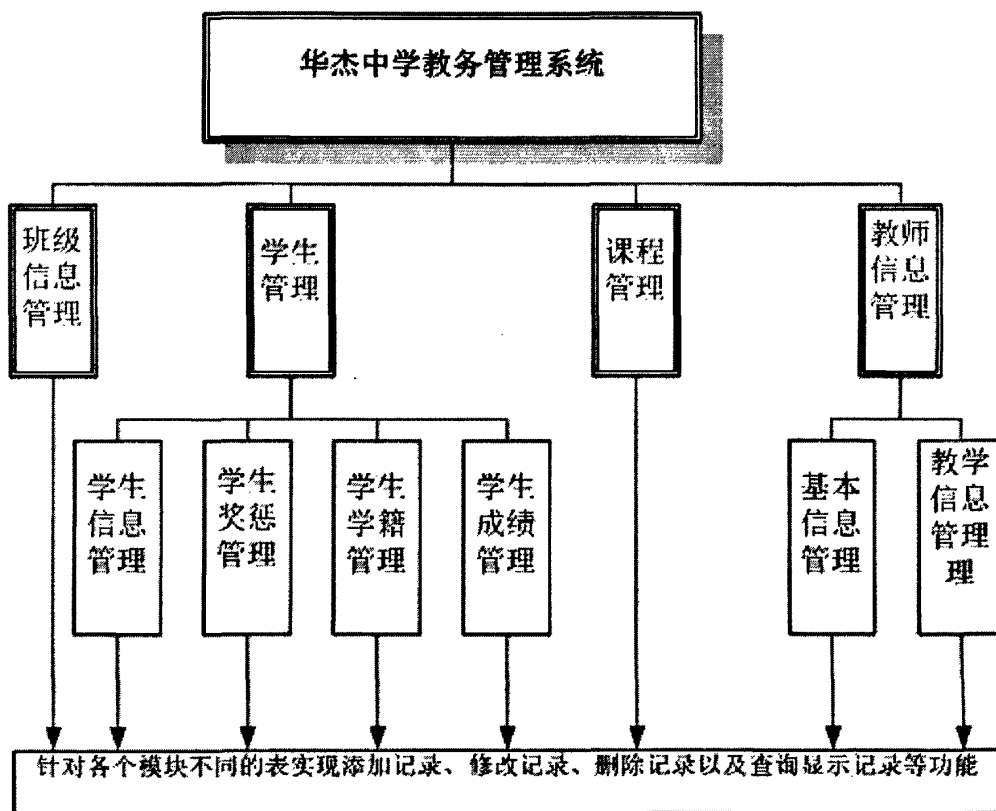


图 4.1 系统模块总体设计图

4.4 数据库设计

4.4.1 数据库表的建立

根据教务信息管理系统的功能要求，选取 SQL Server 作为后台数据库，数据库前台控制界面使用 Visual Basic6.0+ODBC+ADO 来实现。

通过第二部分的分析，大家应该对本系统所要求的数据库结构有了一个初步

的印象，并建立了一个数据库结构的框架，现在就在 SQL Server 2000 数据库系统中来实现需要实现的逻辑结构。

先介绍一下进入 SQL Server 里面建立数据库以及数据表的过程。首先“开始—所有程序—Microsoft SQL Server—企业管理器”，打开 SQL Server 2000 的企业管理器后，展开“控制台根目录”的“数据库”项，在“数据库”项上右击鼠标选择“新建数据库”，弹出“数据库属性”，在“名称”栏内填写“StudentSys”，其他项可选系统自动默认，点击“确定”，会在看到一新建名称为“StudentSys”文件，数据库创建成功。如图 4.2 所示：

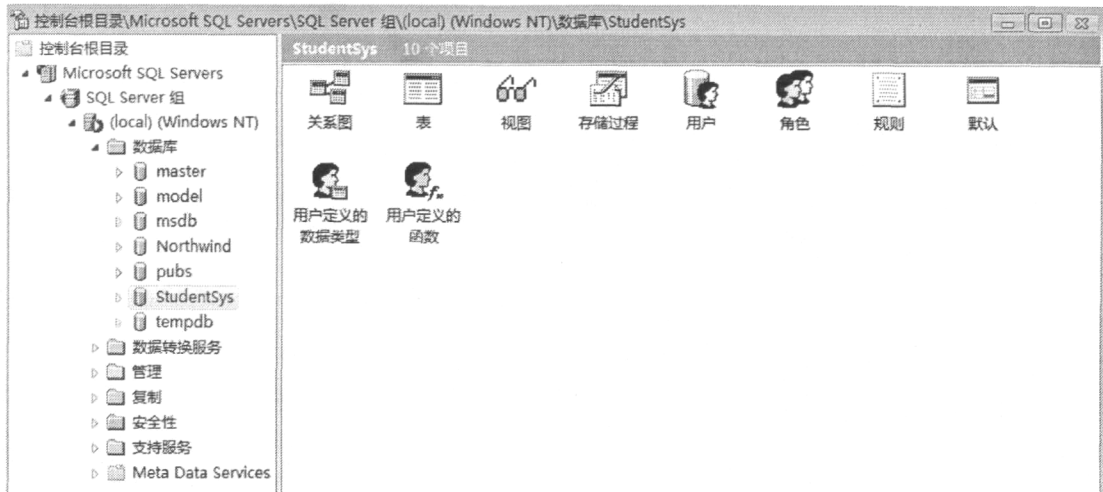


图 4.2 在 SQL Server 企业管理器建立数据库“StudentSys”后的图

下面简述一下数据表的建立。左键单击上图 4.2 里面教务信息管理系统缩写“StudentSys”，右键单击“表”。在弹出的菜单中选择“新建表”，填入相应信息并进行设置，点击“保存”，在框内填入学生管理的英文“Students”，点击“确定”即可。至此，表格设计基本完成，所有表格均遵循这一步骤创立，不再一一赘述。

值得一提的是在建立数据库过程中，应尽量使用英文字母为表以及表内各字段命名，这样能减少系统运行出错的机会。在系统测试过程中曾出现该错误，后将数据库中的字段改为英文，系统运行恢复正常。

在当今软件开发环境日益发达的情况下，数据库设计成为管理信息系统设计的中心内容。在制定了优秀的数据库设计方案的数据库中，数据检索和修改就很容易实现。同样，从用户的角度，数据库的真正设计过程对使用应用软件的最后用户来说也是透明的。从数据库的管理的角度，数据库的维护工作也变得简单并且容易完成。数据库本身也就确保数据完整性以及保持与业务规则的一致性。

4.4.2 系统数据库表设计的原则

数据库设计是管理信息系统开发的关键，不同的设计方案将影响数据库的访问方式及编程的效率。本系统的设计目标是：在学校工作环境下，建立一个性能良好的，能满足用户使用要求的数据库模式。按照该数据库模式建立的数据库，应当能够完整地反映现实世界中的信息及信息之间的联系；能够有效地进行数据存储；能够方便地执行各种数据检索和处理操作；并且有利于进行数据维护和数据控制管理的工作。为了达到这个目标，在数据库的设计过程中遵循以下原则：

1、一致性

对数据进行系统、统一的分析与设计，协调好各数据源，做到“数出一门”、

“算法统一”、“度量一致”，保证系统数据的一致性和有效性。

2、完整性原则

数据库的完整性是指数据的正确性和相容性。要防止合法用户使用数据库时向数据库加入不合语义的数据。对输入到数据库中的数据要有审核和约束机制。

3、安全性原则

数据库的安全性是指保护数据，防止非法用户使用数据库或合法用户非法使用数据库造成数据泄漏、更改或破坏，要有认证和授权机制。

4、可伸缩性原则

数据库结构的设计应充分考虑发展的需要、移植的需要，有良好的扩展性、伸缩性和适度冗余。

5、规范化原则

数据库的设计应遵循规范化的理论。规范化程度过低，可能会存在插入或删除异常、修改复杂、数据冗余等问题，解决的方法就是对关系模式进行分解或合并(规范化)，转换成高级范式。

4.4.3 系统数据库的概念结构设计

数据库的概念结构设计采用构建实体模型(E-R)的方法。E-R(Entity-Relation)即实体—联系的方法，用以构造数据库的概念模型，是设计数据库的一种工具。E-R图有三种基本成分：实体、联系和属性。本教务管理信息系统涉及的实体较多，联系复杂，要建立起整个的E-R有很大困难，处理方法是先把整个系统划分为几块，先进行几个局部设计，画出几个局部E-R图，然后将这些E-R图进行汇总，综合成一个整个的E-R图。依据系统分析，系统的部分主要实体有：学生实体，班级实体，教育经历实体，奖惩实体，学籍变动实体，成绩实体，课程设置实体和教师实体等。它们的E-R图如下：

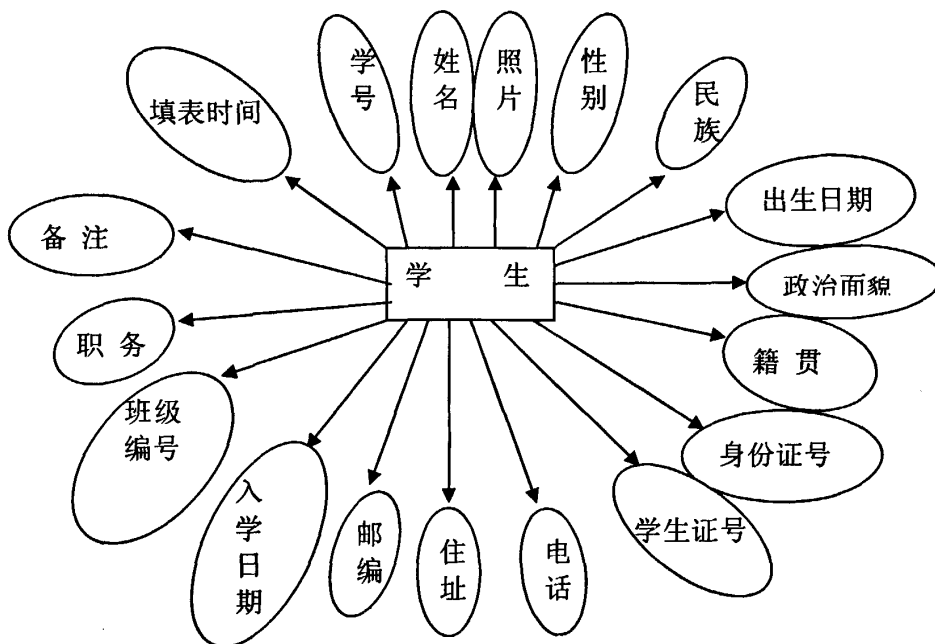


图 4.3 学生的 E-R 图

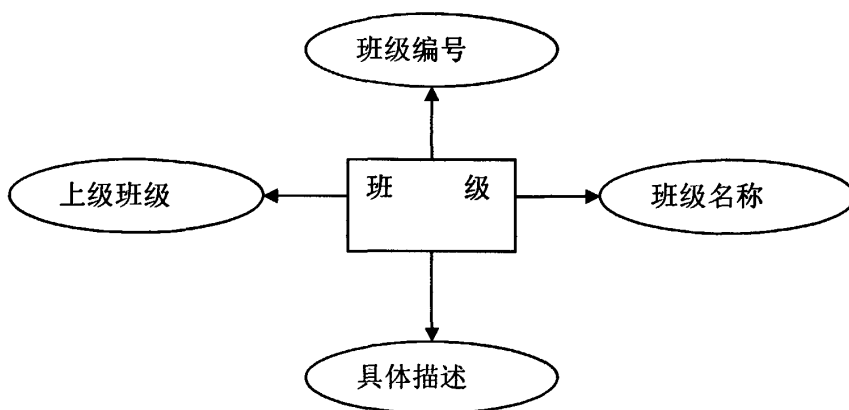


图 4.4 班级的 E-R 图

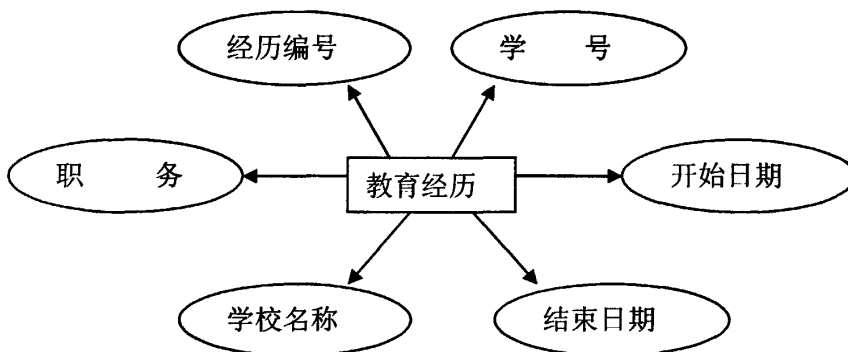


图 4.5 教育经历的 E-R 图

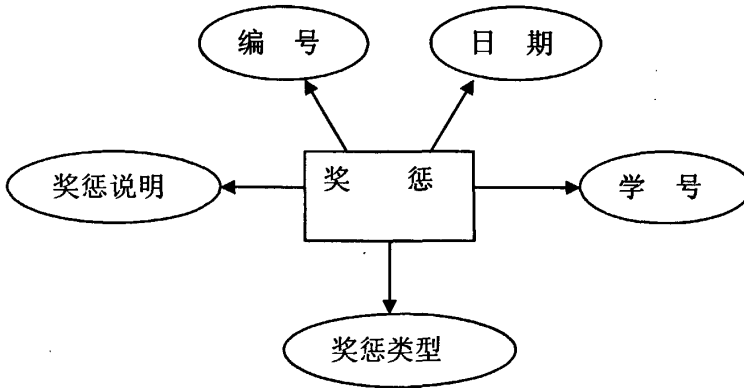


图 4.6 奖惩的 E-R 图

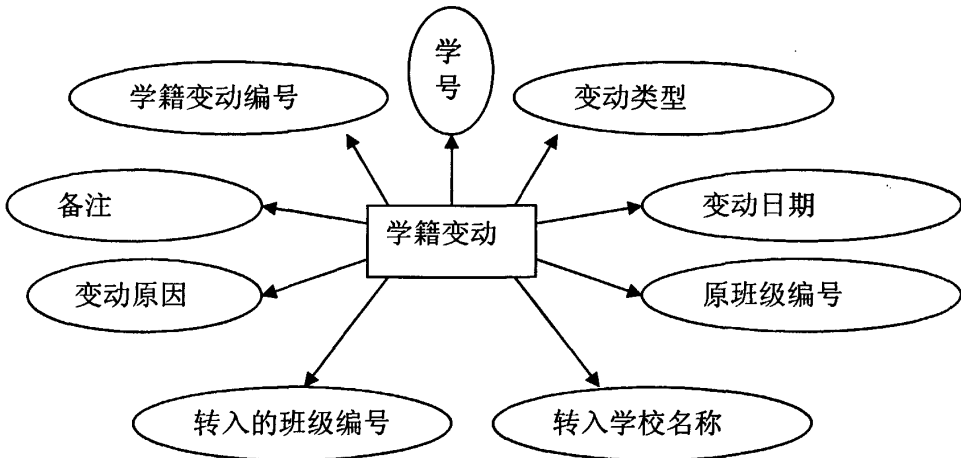


图 4.7 学籍变动的 E-R 图

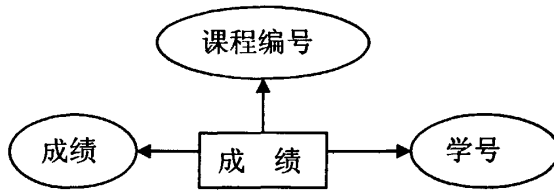


图 4.8 成绩的 E-R 图

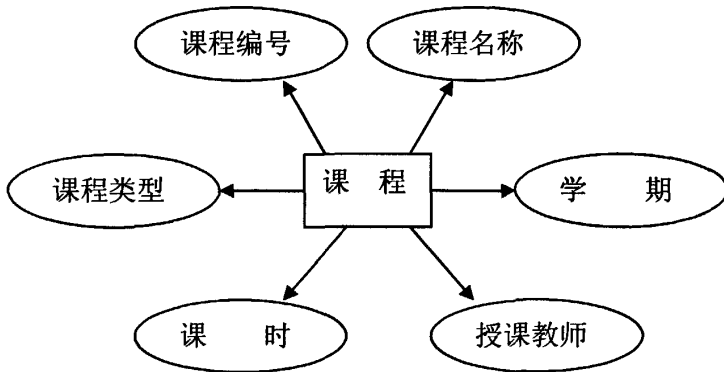


图 4.9 课程的 E-R 图

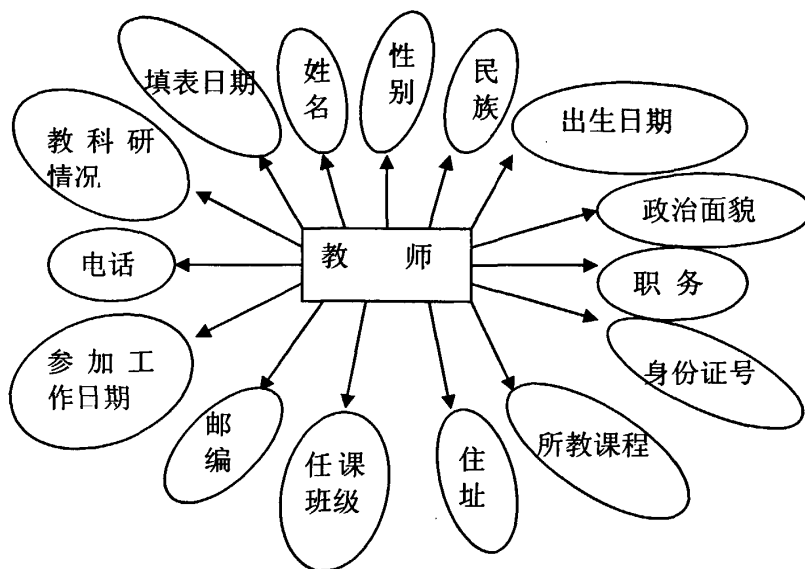


图 4.10 教师的 E-R 图

4.5 系统数据库的逻辑结构设计

E-R 图是实现数据库逻辑结构设计的基础, 把 E-R 图转换成系统数据库所能接受的形式, 生成数据库的表。

本中学教务信息管理系统数据库主要由 10 个主报表组成。这 10 个报表包括: 学生基本信息表 (Students)、学生教育经历表 (Evaluation)、学生奖惩情况表 (Experience)、学籍变动情况表 (Change)、课程设置表 (Lessons)、考试成绩表 (Scores)、班级信息表 (Classes)、教师信息表 (Teachers)、教师任教学科表 (Xuekes)、用户信息表 (Users)。各表分别包含相应的数据库内容, 并按索引建立关系。

4.5.1 主表的结构设计

1、学生基本信息表 (Students) 是对所有学生的基本信息的管理。包括: 学号、姓名、照片、性别、民族、生日、政治面貌、籍贯、身份证号、学生证号、电话、地址、邮编、入学日期、班级编号、职务、备注、填表时间等信息。学生基本信息表 (Students) 的结构见表 4.1 所示:

表 4.1 学生基本信息表

| 编号 | 字段名 | 数据类型 | 备注 |
|----|-----------------|------------|------|
| 1 | Student_Id | Int | 学号 |
| 2 | Student_Name | Varchar 50 | 姓名 |
| 3 | Photo | Image | 照片 |
| 4 | Sex | Char 2 | 性别 |
| 5 | Nationalty | Varchar 40 | 民族 |
| 6 | Birth | Varchar 20 | 生日 |
| 7 | Political_Party | Varchar 40 | 政治面貌 |

| | | | |
|----|--------------|-------------|-------------|
| 8 | Family_Place | Varchar 60 | 籍贯 |
| 9 | Id_Card | Varchar 20 | 身份证号 |
| 10 | BadgeID | Varchar 40 | 学生证号 |
| 11 | Home_phone | Varchar 30 | 电话 |
| 12 | Residence | Varchar 100 | 地址 |
| 13 | Postcode | Varchar 10 | 邮编 |
| 14 | InDate | Varchar 20 | 入学日期 |
| 15 | Class_Id | Smallint | 班级编号 |
| 16 | Title | Varchar 20 | 职务 |
| 17 | Memo | Varchar 200 | 备注 |
| 18 | Fillin_Time | Varchar 20 | 填表时间（日期+时间） |

2、学生教育经历表（Evaluation）对学生教育经历基本信息的管理。包括：经历编号、学号、开始日期、结束日期、学校名称、职务等信息。学生教育经历表（Evaluation）的结构见表 4.2 所示：

表 4.2 学生教育经历表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|------------|------|
| 1 | ExpId | Int | 经历编号 |
| 2 | Student_Id | Int | 学号 |
| 3 | Start_Date | Char 10 | 开始日期 |
| 4 | End_Date | Char 10 | 结束日期 |
| 5 | School | Varchar 50 | 学校名称 |
| 6 | Title | Varchar 20 | 职务 |

3、学生奖惩情况表（Experience）是对所有学生的奖惩情况的管理。包括：奖惩编号、奖惩日期、学号、奖惩类型、奖惩说明等信息。学生奖惩表（Experience）的结构见表 4.3 所示：

表 4.3 学生奖惩表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|-------------|-----------------|
| 1 | EvaId | Int | 奖惩编号 |
| 2 | EvaDate | Char 10 | 奖惩日期 |
| 3 | Student_Id | Int | 学号 |
| 4 | EvaType | Bit | 奖惩类型（0 奖励，1 处罚） |
| 5 | Memo | Varchar 200 | 奖惩说明 |

4、学籍变动情况表（Change）是对所有学生的变动情况的管理。包括学籍变动编号、学号、变动类型、变动日期、原班级编号、转入学校的名称、转入的班级编号、变动原因、备注等信息。学籍变动表（Change）的结构见表 4.4 所示：

表 4.4 学籍变动表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|---------|---------------------------|
| 1 | ChangeId | Int | 学籍变动编号 |
| 2 | Student_Id | Int | 学号 |
| 3 | ChangeType | Tinyint | 变动类型（1 退学，2 休学，3 转学，4 调班） |
| 4 | ChangeDate | Char 10 | 变动日期 |

| | | | |
|---|-----------|-------------|---------|
| 5 | OriClass | Int | 原班级编号 |
| 6 | NewSchool | Varchar 50 | 转入学校名称 |
| 7 | NewClass | Int | 转入的班级编号 |
| 8 | Reason | Varchar 200 | 变动原因 |
| 9 | Memo | Varchar 200 | 备注 |

5、课程设置表 (Lessons) 是对学校各年级的课程设置信息的管理。包括：课程编号、课程名称、学期、授课教师、课时、课程类型等信息。课程设置表 (Lessons) 的结构见表 4.5 所示：

表 4.5 课程设置表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|-------------|------|
| 1 | LessonId | Int | 课程编号 |
| 2 | LessonName | Varchar 100 | 课程名称 |
| 3 | Term | Varchar 20 | 学期 |
| 4 | Teacher | Varchar 500 | 授课教师 |
| 5 | Hours | Tinyint | 课时 |
| 6 | LessonType | Varchar 20 | 课程类型 |

6、考试成绩表 (Scores) 是对所有学生的各门功课的成绩信息的管理。包括：课程编号、学号和成绩。成绩表 (Scores) 的结构见表 4.6 所示：

表 4.6 成绩表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|---------|------|
| 1 | LessonId | Int | 课程编号 |
| 2 | Student_Id | Int | 学号 |
| 3 | Score | Tinyint | 成绩 |

7、班级信息表 (Classes) 是对学校所有班级的详细信息的管理。包括：班级编号、班级名称、具体描述、上级班级的信息。班级信息表 (Classes) 的结构见表 4.7 所示：

表 4.7 班级信息表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|-------------|------|
| 1 | Class_Id | Int | 班级编号 |
| 2 | Class_Name | Varchar 40 | 班级名称 |
| 3 | Describe | Varchar 400 | 具体描述 |
| 4 | UpperId | SmallInt | 上级班级 |

8、教师信息表 (Teachers) 是对所有教师的基本信息的管理。包括：教师编号、姓名、照片、性别、民族、生日、政治面貌、职务、身份证号、所教课程、地址、任课班级、邮编、参加工作日期、电话、教科研情况、学科编号、填表日期等信息。教师信息表 (Teachers) 的结构见表 4.8 所示：

表 4.8 教师信息表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|--------------|------------|------|
| 1 | Teacher_Id | Int | 教师编号 |
| 2 | Teacher_Name | Varchar 50 | 姓名 |
| 3 | Photo | Image | 照片 |
| 4 | Sex | Char 2 | 性别 |
| 5 | Nationality | Varchar 40 | 民族 |

| | | | |
|----|-----------------|--------------|--------|
| 6 | Birth | Varchar 20 | 生日 |
| 7 | Political_Party | Varchar 40 | 政治面貌 |
| 8 | Title | Varchar 20 | 职务 |
| 9 | Id_Card | Varchar 20 | 身份证号 |
| 10 | subject | Varchar 60 | 所教课程 |
| 11 | Residence | Varchar 500 | 地址 |
| 12 | Renkeclass | Varchar 40 | 任课班级 |
| 13 | Postcode | Varchar 10 | 邮编 |
| 14 | InDate | Varchar 20 | 参加工作日期 |
| 15 | Home_phone | Varchar 40 | 电话 |
| 16 | study | Varchar 2000 | 教科研情况 |
| 17 | Xueke_Id | SmallInt | 学科编号 |
| 18 | Fillin_Time | Varchar(20 | 填表日期 |

9、教师任教学科表 (Xuekes) 是对所有教师任教学科信息的管理。包括：学科编号和学科名称。教师任教学科表 (Xuekes) 的结构见表 4.9 所示：

表 4.9 教师任教学科表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|------------|------------|------|
| 1 | Xueke_Id | Int | 学科编号 |
| 2 | Xueke_Name | Varchar 40 | 学科名称 |

10、用户信息表 (Users) 是对系统的用户信息 (管理员和普通用户) 的管理。包括：用户名和密码。用户信息表 (Users) 的结构见表 4.10 所示：

表 4.10 用户信息表

| 编号 | 字段名称 | 数据类型 | 备注 |
|----|----------|------------|-----|
| 1 | UserName | Varchar 40 | 用户名 |
| 2 | Pwd | Varchar 40 | 密码 |

4.5.2 主表间的关系设计

本软件建立了 10 个主报表，这些报表之间存在着一定的关联关系，有些甚至是密不可分的。在学校教务的实际管理工作中，一些数据是共享的，比如“考试成绩表 (Scores)”里面的学号必须是“学生基本信息管理 (Students) 表”里面已经录入的，如果没有录入该生资料，那么在成绩管理里面将查询不到该生的成绩。同理，教育经历、奖惩情况、学籍变动里面的学生信息都是以“学生基本信息管理表 (Students)”里面存在该生学号为前提的。所以各表之间存在一定的关联关系。各表内学号均以学生基本信息里面的学号为主键，其他表内的学号为外键。

第 5 章 教务信息管理系统实现

5.1 窗体的实现

5.1.1 窗体实现的步骤

窗体实现的步骤如下：

- 1、生成窗体。
- 2、设置窗体的属性。
- 3、引入控件。
- 4、设置控件属性。
- 5、添加代码。

5.1.2 窗体的事件

Visual Basic 采用事件驱动编程机制的语言，传统编程使用的是面向过程、按顺序进行的机制，编程人员必须关心编程的顺序，而不必考虑按精度次序执行的每个步骤。所谓事件，是由 Visual Basic 预先设置好的、能够被对象识别的动作。窗体的事件较多，本中学教务信息管理系统窗体的事件主要有 Click（单击）事件和 Load（装入）事件。

5.1.3 窗体的方法

方法就是 Visual Basic 在界面控件中内置的代码，可以根据用户要求来调用。本系统中用到的窗体方法有：Load 方法、Unload 方法、Hide 方法和 Show 方法。

5.2 系统的功能模块

本系统包括登录管理模块、学生管理模块、奖惩管理模块、学籍变动管理模块、成绩管理模块、课程管理模块和教师管理模块等几个模块。下面分别对这几个功能模块进行介绍。

5.2.1 登录功能

本系统采用用户登陆验证方式，登陆页面如图 5.1 所示：

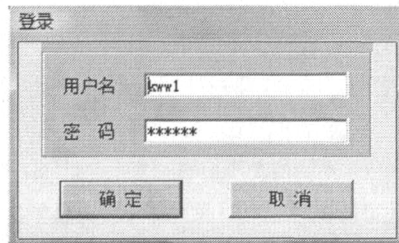


图 5.1 系统登陆界面

为了方便用户登陆，用户可以输入或者从系统用户列表选择用户名，口令要用户输入，用户可以输入事先由向服务器注册得到的口令，登录以后，用户即可进入系统主界面。

5.2.2 主界面

用户登陆后，显示控制界面运行的主界面。如图 5.2 所示。

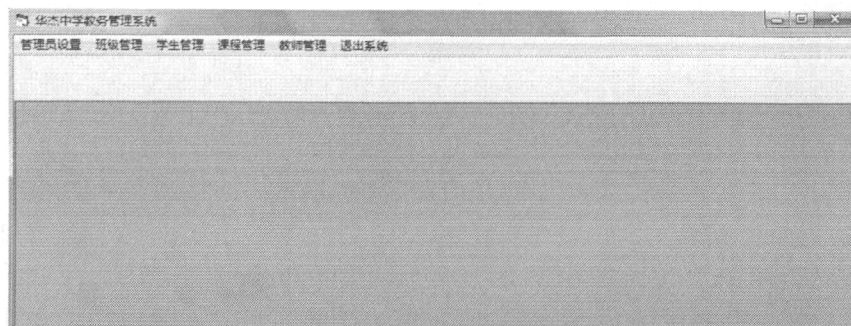


图 5.2 主界面

界面的可视化编程中采用事件触发机制，即程序的功能通过点击菜单、按钮等触发方式组织在一起，这就决定了控制界面的结构是把各个功能分类组织在相应的菜单项下，用户根据需要对系统进行设置，执行相应的操作。

主界面的功能从图 5.2 可以看出，包括：管理员设置、班级管理、学生管理、课程管理、教师管理和退出系统，其中的学生管理菜单下又包含学生信息管理、奖惩管理、学籍变动管理和成绩管理四个子菜单。点击任何一个菜单，都会弹出相应的子窗口，而其它窗口将会关闭。

5.2.3 “班级管理”功能模块

在此功能中，实现了对班级信息添加、修改和删除的管理，在窗体中的 TreeView 控件中选中年级，然后用鼠标点击添加或删除按钮后，会弹出编辑班级详细信息的窗口就可以进行添加和修改班级的操作操作。其界面如图 5.3 所示：

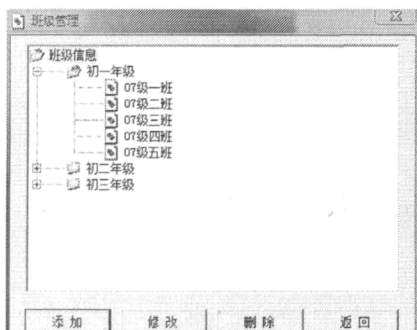


图 5.3a 班级管理功能界面

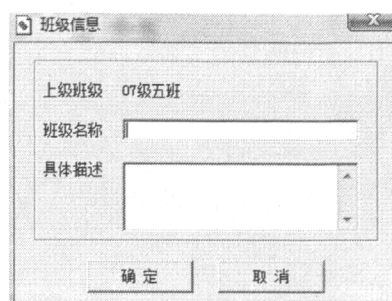


图 5.3b 编辑班级详细信息功能界面

5.2.4“学生信息管理”功能模块

此功能实现了对学生个人情况、家庭情况和教育经历等信息的添加、查找、

修改和删除等功能。其页面如图 5.4 示：

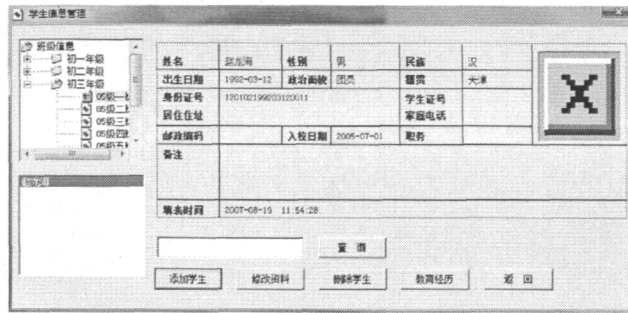


图 5.4 学生信息管理界面

在该窗体中的右下侧设置了一个 **CommandButton** 控件来连接教育经历窗体。当单击该按钮后会弹出教育经历窗口，可以对学生的主要教育经历进行添加、修改和删除等操作。其界面如图 5.5 所示：

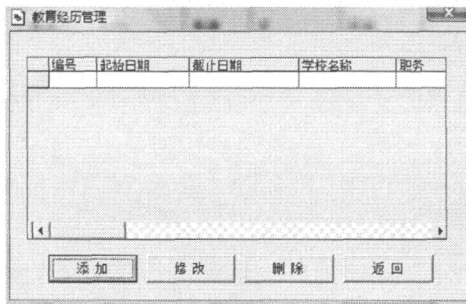


图 5.5 学生教育经历界面

5.2.5“学生奖惩管理”功能模块

在此功能中,实现了对学生获得奖励和处罚的记录和管理,并可以根据年份和奖惩类型分别显示这些信息,并根据情况进行添加、修改和删除等操作。在窗体中先在上面的两个 **ComoboBox** 控件中选中日期和奖惩类型,然后在中间的 **DataGrid** 控件中会显示在此期间,对应类型奖惩的学生信息的列表,并根据情况进行添加、修改和删除等操作。其界面如图 5.6 所示：

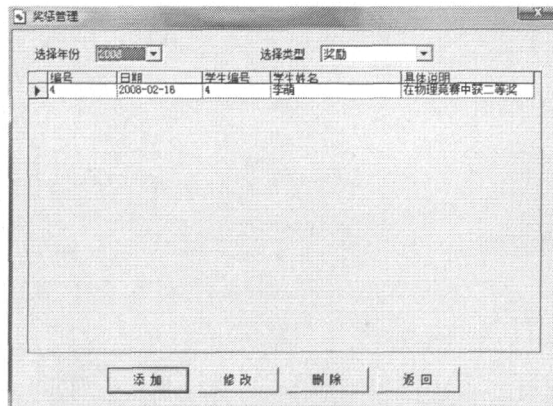


图 5.6 奖惩信息功能界面

5.2.6“学籍变动管理”功能模块

在此功能中,对学生退学,休学,转学和调班情况的管理,在窗体中先在上方的两个 ComboBox 控件中选择日期和变动类型,然后在中间的 DataGrid 控件中会显示在此期间,对应类型学生学籍变动信息的列表,并根据情况进行添加、修改和删除等操作。其界面如图 5.7 所示:

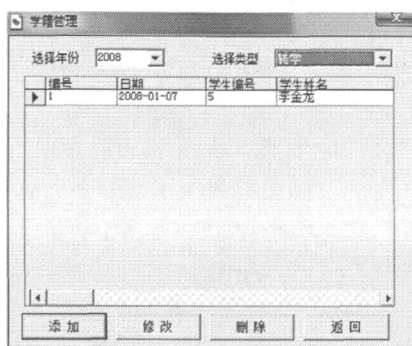


图 5.7 学籍变动管理界面

5.2.7 “成绩管理”功能模块

在此功能中,实现了按学科,分班级的成绩管理,其界面如图 5.8 所示:

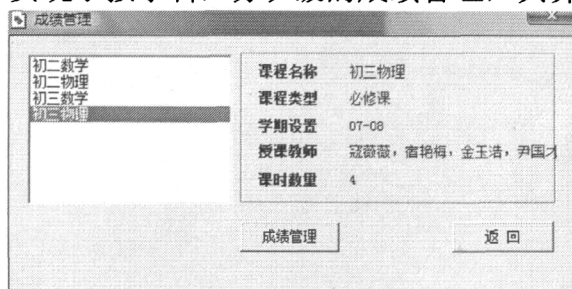


图 5.8 成绩信息功能的界面

单击该界面中的“成绩管理”按钮,就会进入具体成绩信息管理的界面,可以进行成绩的添加、修改和删除的操作,其界面如图 5.9 所示:

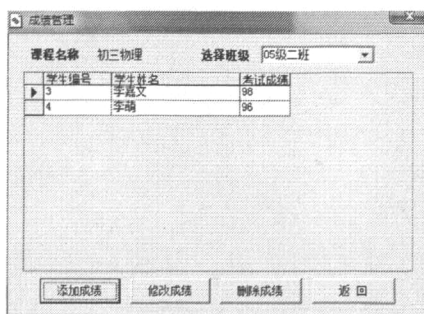


图 5.9 具体成绩信息功能的界面

5.2.8 “课程管理”功能模块

在此功能中，实现了对不同学科的学期设置，任课教师，周课时量的管理，单击“添加课程”、“修改课程”和“删除课程”按钮来实现对课程信息的添加，修改和删除操作。单击“添加课程”或“修改课程”按钮，就会进入编辑课程管理的界面，其界面如图 5.10 所示

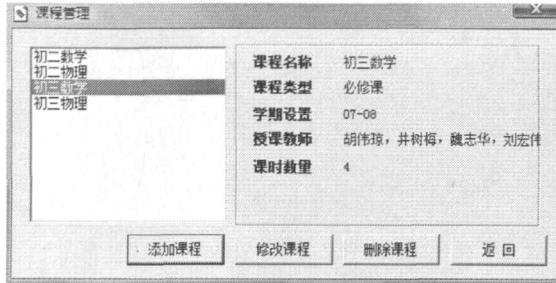


图 5.10a 课程管理功能界面

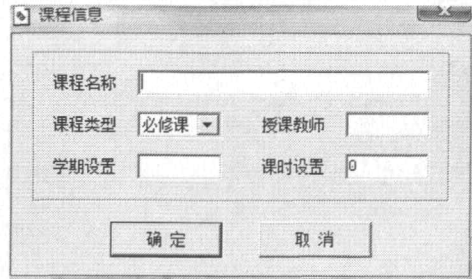


图 5.10b 编辑课程信息的界面

5.2.9“教师管理”功能模块

在此功能中，实现了对教师基本信息和教学信息的管理。在左上部，用 TreeView 控件显示学科名称；在右下部，用 DataList 控件显示教师姓名；在右侧，用 SSTab 控件显示每位教师的基本信息和教学信息；在下面，用了五个 CommandButton 控件实现对教师信息的添加，修改，删除和查询的操作，并用一个 TextBox 控件来输入要查询的教师的姓名。对这些信息可以进行添加，修改，删除和查询的操作。其界面如图 5.11 所示：

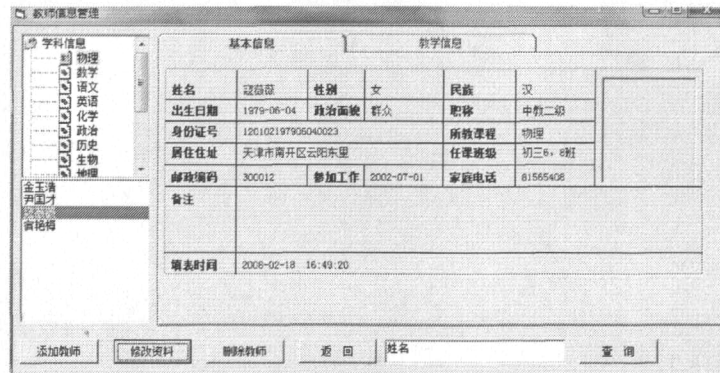


图 5.11 教师信息管理功能的界面

5.3 各模块添加、修改、删除及查询功能的实现

各项数据的添加功能主要是通过各模块中的 Insert() 自定义函数来实现的。在创建数据库表并录入数据后，修改或更新表中的数据就成为对数据库进行维护的日常过程。系统编程过程中是使用 UPDATE 语句来修改和更新数据库表中的信息的。通过使用 DELETE 语句从数据库表中删除一条或多条记录的。DELETE 语句能从指定表中删除满足 WHERE 子句条件的所有记录。如果省略 WHERE 子句，则删除表中的全部记录，但表的定义仍然在数据字典中。例如，学生信息管理模块中删除一条学生记录主要是通过“Delete FROM Students WHERE Student_Id=001”这条语句来完成的。查询功能是通过使用 SELECT 语

句来完成的。例如通过教师姓名查询教师个体信息的 SELECT 语句为“SELECT * FROM teachers WHERE Teacher_name = '张红'”

在代码的编写时各模块中的添加、修改、删除及查询功能主要是通过系统中编写的类模块实现的。在通常情况下，类的成员变量与对应的表中的字段名相同。由于绝大多数成员函数的编码格式都是非常相似的，只是所使用的 SQL 语句不同，所以在此将只说明每个类中成员函数的功能，以并不对所有的成员函数进行具体的代码分析。

1. Classes 类模块

Classes 类模块用来管理班级信息表 Classes 的数据库操作，它保存为 Classes.cls。其成员函数如下表 5.1 所示：

表 5.1 Classes 类的成员函数

| 函数名 | 备注 |
|----------------------|--|
| Init | 初始化成员变量 |
| Delete | 删除指定的班级记录。参数 TmpId 表示要删除的班级编号 |
| GetIngo | 读取指定的班级记录。参数 TmpId 表示要读取的班级编号 |
| GetName | 根据指定的班级名称读取班级编号 |
| GetNewId | 生成新的班级编号 |
| HaveStudent | 判断指定班级中是否存在学生。如返回 True，则表示存在学生，否则表示没有学生。此函数将在删除班级数据之前调用，如返回 True，则不能删除指定班级 |
| In_DB | 判断班级是否已经在数据库中。参数 ClassName 表示指定的班级名称，ClassUpper 表示上一级班级的编号 |
| Insert | 插入新的班级记录。插入前应调用 GetNewId 函数生成新的编号 |
| Load_Classes_ByUpper | 根据指定的上级班级编号，读取所有下一级班级的名称 |
| Update | 修改指定的班级记录 |

对其中几个函数的说明如下：

(1) Init 函数

```
Public Sub Init()
    Class_Id = -1
    Class_Name = ""
    Describe = ""
    UpperId = -1
End Sub
```

(2) GetNewId 函数

```
Public Function GetNewId() As Long
    Dim TmpId As Long
    DB_Connect '连接数据库
    '设置要执行的 SQL 语句
    SqlStmt = "SELECT Class_Id FROM Classes ORDER BY Class_Id"
    OdbcExt (SqlStmt) '执行 SQL 语句
    i = 1 '读取 SELECT 语句返回的查询结果
```

```

Do While SQLFetch(Hstmt) <> SQL_NO_DATA_FOUND
    ColVal = String(40, 0)
    Rc = SQLGetData(Hstmt, 1, 1, ColVal, Len(ColVal), pcblen)
    TmpId = Val(ColVal)      '把读取的班级编号与自然数序列比较
    If TmpId = i Then
        i = i + 1
    Else                    '如果班级编号不连续, 则用新编号来填充
        GetNewId = i
        Rc = SQLFreeStmt(Hstmt, SQL_DROP)
        DB_Disconnect
        Exit Function
    End If
Loop
Rc = SQLFreeStmt(Hstmt, SQL_DROP)
DB_Disconnect      '断开与数据库的连接
GetNewId = I      '将得到的新编号作为函数的返回值
End Function
    
```

此函数的功能是生成新的班级编号, 在 Insert()过程中将调用此函数。生成新的班级编号的算法是: 把读取的班级编号与自然数序列比较, 如果班级编号是不连续的, 则用新编号来填充; 如果班级编号是连续的, 则用最大的班级编号加 1 作为新编号。

(3) HaveStudent 函数

```

Public Function HaveStudent(ByVal TmpClassId As Long) As Boolean
    DB_Connect      '连接数据库
    '设置要执行的 SQL 语句
    SqlStmt = "SELECT * FROM Students WHERE Class_id=" +
Trim(Str(TmpClassId))
    OdbcExt (SqlStmt)      '执行 SQL 语句
    '如果 SQLFetch()函数返回 SQL_NO_DATA_FOUND, 则表示没有满足条件
    查询结果
    If SQLFetch(Hstmt) = SQL_NO_DATA_FOUND Then
        HaveStudent = False
    Else
        HaveStudent = True
    End If
    Rc = SQLFreeStmt(Hstmt, SQL_DROP)      '断开与数据库的连接
    DB_Disconnect
End Function
    
```

2. Students 类模块

Students 类模块用来管理学生基本信息表 Students 的数据库操作, 它保存为 Students.cls。其成员函数如下表 5.2 所示:

表 5.2 Students 类的成员函数

| 函数名 | 备注 |
|------|---------|
| Init | 初始化成员变量 |

| | |
|-----------------|---|
| Delete | 删除指定的学生记录。参数 TmpStudentId 表示要删除的学号 |
| GetInfo | 读取指定的学生记录。参数 TmpStudentId 表示要读取的学号 |
| GetName | 根据指定的学号读取学生姓名。参数 TmpStudentId 表示要读取的学号 |
| GetNameId | 学生的学号 |
| GetNewId | 生成新的学号 |
| Insert | 插入新的学生记录。插入前应调用 GetNewId 函数生成新的学号 |
| Load_ByClass | 读取指定班级的所有学生学号到数组 Arr_StuId() 和 Arr_StuName()中 |
| Update | 修改指定的学生记录。参数 OriStudentId 表示要修改的学号 |
| Update_Transfer | 修改指定学生的班级信息 Class_Id。此过程在学生班级调转时使用 |

Load_By_Class 过程如下:

```
Public Sub Load_By_Class(ByVal TmpClassId As Integer)
    Dim i As Integer          '初始化班级数组
    Erase Arr_StuId
    ReDim Arr_StuId(0)
    Erase Arr_StuName
    ReDim Arr_StuName(0)
    DB_Connect              '连接数据库
    '要执行的 SQL 语句
    SqlStmt = "SELECT Student_Id, Student_Name FROM Students WHERE
Class_Id=" + Trim(Str(TmpClassId)) + " ORDER BY Student_Id"
    OdbcExt (SqlStmt)      '执行 SQL 语句
    i = 0
    '依次读取满足条件的学生编号读取到数组中
    Do Until SQLFetch(Hstmt) = SQL_NO_DATA_FOUND
        ColVal = String(40, 0)
        Rc = SQLGetData(Hstmt, 1, 1, ColVal, Len(ColVal), pcblen)
        ReDim Preserve Arr_StuId(i + 1)
        Arr_StuId(i) = Val(ColVal)
        ColVal = String(40, 0)
        Rc = SQLGetData(Hstmt, 2, 1, ColVal, Len(ColVal), pcblen)
        ReDim Preserve Arr_StuName(i + 1)
        Arr_StuName(i) = TrimStr(ColVal)
        i = i + 1
    Loop
    Rc = SQLFreeStmt(Hstmt, SQL_DROP)
    DB_Disconnect          '断开与数据库的连接
End Sub
```

3. Experience 类模块

Experience 类模块用来管理学生教育经历表 Experience 的数据库操作, 它保存为 Experience.cls。其成员函数如下表 5.3 所示:

表 5.3 Experience 类的成员函数

| 函数名 | 备注 |
|------------------|---|
| Init | 初始化成员变量 |
| DeleteById | 删除指定的学生记录。参数 TmpId 表示记录编号 |
| DeletByStudentId | 删除指定的学生的所有记录。参数 TmpStudentId 表示学号 |
| GetInfo | 读取指定的记录。参数 TmpStudentId 表示学号，TmpId 表示记录号，由这两个字段唯一确定一条记录 |
| Insert | 插入新的记录。参数 TmpStudentId 表示学号 |
| Update | 修改指定的记录。参数 TmpStudentId 表示学号，TmpId 表示记录号，由这两个字段唯一确定一条记录 |

(1) DeleteById 过程

'删除单个数据

```
Public Sub DeleteById(ByVal TmpId As Integer)
```

```
    DB_Connect '连接数据库
```

```
    SqlStmt="DELETE FROM Experience WHERE ExpId=" + Trim(Str(TmpId))
```

'SQL 语句

```
    OdbcExt (SqlStmt) '执行 SQL 语句
```

```
    Rc = SQLFreeStmt(Hstmt, SQL_DROP)
```

```
    DB_Disconnect '断开与数据库的连接
```

```
End Sub
```

(2) DeleteByStudent 过程

'删除一个学生的所有教育经历数据

```
Public Sub DeleteByStudent(ByVal TmpStudentId As Long)
```

```
    DB_Connect '连接数据库
```

```
    SqlStmt = "DELETE FROM Experience WHERE Student_Id=" + Trim(Str(TmpStudentId)) 'SQL 语句
```

```
    OdbcExt (SqlStmt) '执行 SQL 语句
```

```
    Rc = SQLFreeStmt(Hstmt, SQL_DROP)
```

```
    DB_Disconnect '断开与数据库的连接
```

```
End Sub
```

4. Evaluation 类模块

Evaluation 类模块用来管理学生奖惩表 Evaluation 的数据库操作，它保存为 Evaluation.cls。其成员函数如下表 5.4 所示：

表 5.4 Evaluation 类的成员函数

| 函数名 | 备注 |
|-------------------|-----------------------------------|
| Init | 初始化成员变量 |
| DeleteById | 删除指定的奖惩记录。参数 TmpId 表示记录编号 |
| DeleteByStudentId | 删除指定的学生的所有记录。参数 TmpStudentId 表示学号 |
| GetInfo | 读取指定的记录。参数 TmpId 表示记录编号 |
| Insert | 插入新的奖惩记录 |
| Update | 修改指定的奖惩记录。参数 OriId 表示记录编号 |

GetInfo 过程如下。

```
Public Function GetInfo(ByVal TmpId As Long) As Boolean
```

```

EvalId = TmpId
DB_Connect    '连接数据库
'SQL 语句
SqlStmt = "SELECT * FROM Evaluation WHERE EvalId=" + Trim(TmpId)
OdbcExt (SqlStmt)    '执行 SQL 语句
'使用 SQLFetch()过程取得结果集
'如果返回 SQL_NO_DATA_FOUND 则表示结果集为空
If SQLFetch(Hstmt) = SQL_NO_DATA_FOUND Then
    GetInfo = False
    Init
    Exit Function
Else
    '使用 SQLGetData()函数从结果集中按列读取数据
    '奖惩日期
    ColVal = String(400, 0)
    Rc = SQLGetData(Hstmt, 2, 1, ColVal, Lench(ColVal), pcblench)
    EvaDate = TrimStr(ColVal)
    '学生编号
    ColVal = String(400, 0)
    Rc = SQLGetData(Hstmt, 3, 1, ColVal, Lench(ColVal), pcblench)
    Student_Id = Val(ColVal)
    '奖惩类型
    ColVal = String(400, 0)
    Rc = SQLGetData(Hstmt, 4, 1, ColVal, Lench(ColVal), pcblench)
    EvaType = Val(ColVal)
    '具体说明
    ColVal = String(400, 0)
    Rc = SQLGetData(Hstmt, 5, 1, ColVal, Lench(ColVal), pcblench)
    Memo = TrimStr(ColVal)
End If
Rc = SQLFreeStmt(Hstmt, SQL_DROP)
GetInfo = True    '返回 True, 表示已成功的取得数据
DB_Disconnect    '断开与数据库的连接
End Function

```

5. Change 类模块

Change 类模块用来管理学籍变动表 Change 的数据库操作，它保存为 Change.cls。其成员函数如下表 5.5 所示：

表 5.5 Change 类的成员函数

| 函数名 | 备注 |
|-------------------|---------------------------------------|
| Init | 初始化成员变量 |
| DeleteById | 删除指定的学籍变动记录。参数 TmpId 表示记录编号 |
| DeleteByStudentId | 删除指定的学生的所有学籍变动记录。参数 TmpStudentId 表示学号 |
| GetInfo | 读取指定的记录。参数 TmpId 表示学籍变动记录编号 |

| | |
|--------|---------------------------------|
| Insert | 插入新的学籍变动记录 |
| Update | 修改指定的学籍变动记录。参数 TmpId 表示学籍变动记录编号 |

(1) Insert 过程

```
Public Sub Insert()
    DB_Connect      '连接数据库
    'INSERT 语句
    SqlStmt = "INSERT INTO Change (Student_Id, ChangeType, ChangeDate,
OriClass," _
        + " NewSchool, NewClass, Reason, Memo) Values(" + Trim(Student_Id) _
        + "," + Trim(ChangeType) + "," + Trim(ChangeDate) + "," +
Trim(OriClass) _
        + "," + Trim(NewSchool) + "," + Trim(Str(NewClass)) + "," +
Trim(Reason) _
        + "," + Trim(Memo) + ")"
    OdbcExt (SqlStmt)      '执行 INSERT 语句
    Rc = SQLFreeStmt(Hstmt, SQL_DROP)
    DB_Disconnect      '断开与数据库的连接
End Sub
```

(2) Update 过程

```
Public Sub Update(ByVal TmpId As Long)
    DB_Connect      '连接数据库
    'UPDATE 语句
    SqlStmt = "Update Change Set Student_Id=" + Trim(Student_Id) + ",
ChangeType=" _
        + Trim(ChangeType) + ", ChangeDate=" + Trim(ChangeDate) + ",
NewSchool=" _
        + Trim(NewSchool) + ",NewClass=" + Trim(NewClass) +
",Reason=" + Trim(Reason) _
        + ",Memo=" + Trim(Memo) + " WHERE ChangeId=" +
Trim(TmpId)
    OdbcExt (SqlStmt)      '执行 UPDATE 语句
    Rc = SQLFreeStmt(Hstmt, SQL_DROP)
    DB_Disconnect      '断开与数据库的连接
End Sub
```

6. Lesson 类模块

Lesson 类模块用来管理课程表 Lesson 的数据库操作，它保存为 Lesson.cls。其成员函数如下表 5.6 所示：

表 5.6 Lesson 类的成员函数

| 函数名 | 备注 |
|---------|-----------------------------|
| Init | 初始化成员变量 |
| Delet | 删除指定的课程记录。参数 TmpId 表示课程记录编号 |
| GetInfo | 读取指定的记录。参数 TmpId 表示课程记录编号 |
| Insert | 插入新的课程记录 |

| | |
|--------|---------------------------------|
| Update | 修改指定的课程数据。参数 OriId 表示要修改的课程记录编号 |
|--------|---------------------------------|

7. Scores 类模块

Scores 类模块用来管理成绩表 Scores 的数据库操作，它保存为 Scores.cls。其成员函数如下表 5.7 所示：

表 5.7 Scores 类的成员函数

| 函数名 | 备注 |
|----------|--|
| Init | 初始化成员变量 |
| Delet | 删除指定的成绩记录。参数 TmpLid 表示指定的课程编号月份，参数 TmpSid 表示指定的学号 |
| GetScore | 返回指定学生，指定课程的成绩。参数 TmpLid 表示指定的课程编号月份，参数 TmpSid 表示指定的学号 |
| In_DB | 判断成绩是否在库中。参数 TmpLid 表示指定的课程编号月份，参数 TmpSid 表示指定的学号 |
| Insert | 插入新的成绩记录 |
| Update | 修改指定的成绩记录 |

8. Teachers 类模块

Teachers 类模块用来管理教师信息表 Teachers 的数据库操作，它保存为 Teachers.cls。其成员函数如下表 5.8 所示：

表 5.8 Teachers 类的成员函数

| 函数名 | 备注 |
|-----------------|--|
| Init | 初始化成员变量 |
| Delete | 删除指定的教师记录。参数 TmpTeacherId 表示要删除的编号 |
| GetInfo | 读取指定的教师记录。参数 TmpTeacherId 表示要读取的编号 |
| GetName | 根据指定的编号读取教师姓名。参数 TmpTeacherId 表示要读取的编号 |
| GetNameId | 教师的编号 |
| GetNewId | 生成新的编号 |
| Insert | 插入新的教师记录。插入前应调用 GetNewId 函数生成新的编号 |
| Load_ByClass | 读取指定学科的所有教师编号到数组 Arr_TchId() 和 Arr_TchName() 中 |
| Update | 修改指定的教师记录。参数 OriTeacherId 表示要修改的编号 |
| Update_Transfer | 修改指定教师的任教学科信息 Xueke_Id。 |

9. Users 类模块

Users 类模块用来管理表 Users 的数据库操作，它保存为 Users.cls。其成员函数如下表 5.9 所示：

表 5.9 Users 类的成员函数

| 函数名 | 备注 |
|---------|---------------------------------|
| Init | 初始化成员变量 |
| Delet | 删除指定的用户记录。参数 TmpUser 表示要删除的用户名 |
| GetInfo | 读取指定的记录。参数 TmpUser 表示要读取的用户名 |
| In_DB | 判断用户名是否在库中。参数 TmpUser 表示要读取的用户名 |
| Insert | 插入新的用户记录 |

| | |
|--------|--------------------------------|
| Update | 修改指定的用户记录。参数 TmpUser 表示要修改的用户名 |
|--------|--------------------------------|

5.4 系统功能模块实现实例

下面以学生信息管理模块为实例，附上代码来说明模块实现功能的方式，因为功能实现的相似性，在此就不一一详细赘述其他模块的具体制作过程了，详细过程见软件。

当用户在学生信息管理窗口中选择了相应的班级后，单击“添加学生”按钮，会打开编辑学生信息的窗口该过程的主要功能代码如下：

```

TmpOk = False
If CurClass.Class_Id <= 0 Then
    MsgBox "请选择所属班级", 48, "警告"
Exit Sub
End If
StuEdit.Modify = False
StuEdit.Show 1
If TmpOk = True Then
    TreeView1_Click
End If

```

在该窗口中，当用户添加完所有必填信息后，单击“确定”按钮，会将所填的信息添加到窗体中，其主要功能代码如下：

‘根据变量 Modify 判断是插入新数据，还是修改已有数据

```

If Modify = False Then
    .Insert
Else
    .Update (CurStu.Student_Id)
End If
End With
StuMan.TmpOk = True
Unload Me

```

‘设置学生管理窗口中的状态标识

当用户在该窗口中选择了相应的班级，相应的学生信息后，单击“修改资料”按钮，也会打开编辑学生信息的窗口，其主要功能代码如下：

‘检查是否选择要修改的学生记录

```

If CurStu.Student_Id <= 0 Then
    MsgBox "请选择要修改的学生", 48, "警告"
Exit Sub
End If
StuEdit.Modify = True
StuEdit.Show 1
DataList1_Click

```

‘通过单击 DataList 控件刷新学生信息的显示

当用户在该窗口中选择了相应的班级和学生信息后，单击“删除学生”按钮，会将所选的信息删除，其主要功能代码如下：

‘检查是否选择要删除的学生记录

```

If CurStu.Student_Id <= 0 Then
    MsgBox "请选择要删除的学生", 48, "警告"

```



```

Exit Sub
End If
‘确定是否删除
If MsgBox("姓名:" + Lbl_Stu_Name + Chr(13), vbYesNo, "是否删除") = vbNo
Then
Exit Sub
End If
CurStu.Delete (CurStu.Student_Id) ‘调用 Delete 方法删除选择的学生信息
CurStu.Student_Id = 0
TreeView1_Click
    当要想查询某位学生的信息,在文本框中输入该为学生的姓名,单击“查询”
    按钮,就能在右侧显示出该学生的信息,其主要功能代码如下:
query_xhxm = Trim(txt_xhxm.Text)
DB_Connect
sql_query1 = "SELECT * FROM Students WHERE Student_Name = " _
    + "" + query_xhxm + ""
OdbcExt (sql_query1)
If SQLFetch(Hstmt) = SQL_NO_DATA_FOUND Then
HaveStudent = False
Else
HaveStudent = True
End If
Rc = SQLFreeStmt(Hstmt, SQL_DROP)
DB_Disconnect
If HaveStudent = True Then
AdoStuDetail.RecordSource = "SELECT * FROM Students WHERE
Student_Name = " _
    + "" + query_xhxm + ""
AdoStuDetail.Refresh
AdoStuNameList.RecordSource = "SELECT Student_Id, " + _
"Student_Name FROM Students WHERE Student_Name = " + "" +
query_xhxm + ""
AdoStuNameList.Refresh
DataList1.ReFill
CurStu.GetNameid (query_xhxm)
DataList1.BoundText = CurStu.Student_Id
Call ShowImage(Image1, AdoStuDetail)
CurStu.GetInfo (DataList1.BoundText)
Else
AdoStuNameList.RecordSource = "SELECT * FROM Students WHERE
Class_Id<0"
AdoStuNameList.Refresh
CurStu.Init
Image1.Picture = LoadPicture("")

```

```

AdoStuDetail.RecordSource = "SELECT * FROM Students WHERE
Student_Id<0"
AdoStuDetail.Refresh
DataList1.ReFill
CurStu.Student_Id = -1
MsgBox "无此学生", 48, "警告"
End If

```

5.5 安装程序的制作

应用程序开发完成后，必须要制作成安装程序，才能提供给用户使用。VB 的打包和展开向导可以创建应用系统的.cab 文件，将这些文件组合成一个包含安装所需的所有信息的单元或软件包，并将这些软件包交付给最终用户。但用 VB 制作的安装程序体积庞大，且文件琐碎，丢失一个文件就会导致无法安装程序。因此，本系统采用专业软件 Install Shield6.0 制作安装程序。

使用 Install Shield6.0 开发的安装工程还可以很方便的解决升级以及更新的问题，只需在 File Groups 页中以新版本的文件去替换旧版本的相应文件，并重新编译、发布即可，而不必为每一个版本的软件去开发一套与之相应的安装程序，并且可以形成单一的安装文件。

Install Shield 6.0 的所有功能可以通过七个选项卡实现：

- 1、Scripts 页：该选项卡页面用来管理安装程序的脚本文件。
- 2、Components 页：管理安装程序的各组件项。缺省的组件项包括 ProgramFiles(程序文件)、ExampleFiles(示例文件)、HelpFiles(帮助文件)和 SharedDLLs(共享 DLL 文件)等。
- 3、Setup Types 页：用来管理提供给用户的安装类型。缺省设置为 Typical(典型)、Compact(紧凑)和 Custom(自定义)等三种安装类型。
- 4、SetupFiles 页：该页面用来管理在安装过程中将要用到的安装文件。通常包含 Splash Screen, Language Independent, English 和 Advanced Files 等内容。Splash screen 对启动画面进行管理，可以为不同的语系设置不同的启动位图。Language Independent 允许用户为各种操作系统指定相应的安装文件。而 English(或其他语系)项则要求安装程序根据相应的语系及语系下的操作系统来选取相应的文件。Advanced Files 将在使用具体的媒介发布时，允许用户放置一些非压缩文件。
- 5、FileGroups 页：负责管理安装工程所需要的文件组。缺省设置为 ExampleFiles(示例文件组)、HelpFiles(帮助文件组)、ProgramDLLs(应用程序所需的 DLL 文件组)、ProgramExecutable Files(应用程序文件组)以及 Shared DLLs(共享 DLL 文件组)共五个文件组。
- 6、Resources 页：用来管理安装工程所需要的安装资源。缺省设置为 String Table、Registry Entries 和 Shell Objects 共三种资源。String Table 中定义了工程中用到的一些字符串常量。RegistryEntries 设置了注册表入口。Shellobjects 资源则是经常用到的，通过该资源可以在指定位置(桌面、开始菜单等)为安装的应用程序创建文件夹或快捷方式。
- 7、Media 页：用来管理安装程序的发布媒介。该页面包含有 MediaBuildWizard 向导、缺省的媒介以及用户新创建的媒介等项。

使用 Install Shield 制作教务信息管理系统的安装程序非常简单，其制作过程

可分为以下几个步骤:

1)在 Components 页添加程序文件、数据库文件和控件文件。

2)在 Setup Types 页设定自定义安装方式: 用户在安装时需要选择自己是服务器端还是客户端, 服务器端安装所有文件, 客户端则只安装程序文件和控件文件。

3)在 File Groups 页创建程序文件组、数据库文件组和控件文件组, 其中控件文件组设置为安装自动注册。

4)在 Setup Files 页中, 将安装需要的所有文件分类添加到上一步建好的文件组。。

5)在 Resources 页修改安装显示画面、建立桌面快捷方式与设定程序组名称。

6)其他页中保持默认设置, 在 Media 页中建立安装程序。

第6章 系统测试及分析

6.1 软件测试概述

在早期的软件开发过程中,往往不重视软件的测试,软件在开发完成时急于投入使用,结果在使用中会发现有很多问题,使得用户对软件质量的可靠性产生了怀疑,使开发者也处于被动的地位。随着软件工程的引入,人们逐步认识到了软件在投入使用前应进行必要的软件测试,使大部分潜在的错误被排除。因此,软件测试在软件生存期中作为一个独立的阶段受到了广泛的重视。

软件测试是在软件投入运行之前,对软件需求分析、设计规格说明书和编码进行的最终复审,是保证软件质量的关键步骤。

软件测试是为了发现错误而执行程序的过程。测试时根据软件开发各阶段的规格说明和程序的内部结构设计一批有代表性的测试用例,利用这些测试用例去运行程序,以便发现程序中的错误。

软件测试不只是一个独立的阶段。编程人员在编写出每一个模块代码之后即可对它做必要的测试,这一测试过程也称为单元测试,此时模块的测试者与编写者是同一个人,编码与单元测试属于软件生存期中的同一个阶段。在这个阶段结束之后,还要对软件系统进行各种综合测试,这属于软件生产期的另一个独立的阶段,即测试阶段,通常由专门的测试组成员承担这项工作。

6.2 软件测试的目的和原则

6.2.1 软件测试的目的

软件测试是保证软件质量的一个重要环节,从不同的角度考虑软件测试的目的,结果是不相同的。从用户的角度,希望通过软件测试发现软件中的错误和缺陷,决定是否接受该产品。从开发者的角度,则希望测试表明该产品中不存在错误,验证软件实现了用户的要求。因此,软件测试要从整体利益出发,综合考虑。软件测试应当尽量发现程序中存在的错误。

Grenford J. Myers 就软件测试目的提出了以下观点:测试是程序的执行过程,目的在于发现错误;一个好的测试用例在于能发现至今未发现的错误;一个成功的测试是发现了至今未发现的错误的测试。^[48]

测试是想以最少的人力、最短的时间系统地找出软件中潜在的各种错误和缺陷。如果测试是成功的,就能够发现软件中的错误,同时能够证明软件的功能和性能与需求说明相符合。另外,在测试过程中收集到的测试结果数据也为可靠性分析提供了依据。测试不能表明软件中不存在错误,它只能说明软件中存在错误。

6.2.2 软件测试的原则

基于以上的软件测试目的,随着软件测试理论和实践方法的不断发展,软件测试原则规定了测试过程中应该遵循以下的几条原则:

- 1、应尽早规划和不断地进行软件测试。软件测试不仅仅是软件开发的一个独立阶段,它贯穿于软件开发的各个阶段中。在软件开发的各个阶段进行技术评

审，能够在开发过程中尽早发现和预防错误。

2、测试用例设计应包括测试输入数据和与之对应的预期输出结果两部分。测试以前应根据测试的要求设计、选择测试用例，用来检验程序员编写的程序。测试用例不仅需要测试的输入数据，而且要有相应的预期输出结果，以便进行比较。

3、程序员应避免检查自己的程序。调试由程序员自己来做更有效，而测试过程中程序员应尽可能避免测试自己编写的程序，程序开发小组也应尽可能避免测试本小组开发的程序。应根据实现情况，尽量建立独立的软件测试小组或测试机构。

4、在设计测试用例时，应当包括合理的输入条件和不合理的输入条件。合理的输入条件是指能验证程序正确的输入条件，不合理的输入条件是指异常的、临界的、可能引起问题变异的输入条件。用不合理的输入条件测试程序时，往往比不用合理的输入条件进行测试能发现更多的错误。

5、严格执行测试计划，排除测试的随意性。测试之前应仔细考虑测试的项目，对每一项测试做出周密的计划。对于测试计划，要明确规定，不要随意解释。

6、应对于每一个测试结果做全面检查。有些错误可通过实测结果的一些征兆显示出来，如果不仔细全面地检查测试结果，则会使这些错误被遗漏掉。所以必须明确定义预期的输出结果，对实测的结果仔细分析检查，抓住征兆，暴露错误。

7、妥善保存测试计划，测试用例、出错统计和最终分析报告，为维护提供方便。

8、测试应有一个终止的标准。

由于受到时间、费用等多方面的限制，测试不可能永远进行下去。要在各方面的因素中找到一个平衡点。

6.3 软件测试的方法和步骤

6.3.1 软件测试的方法

在软件测试过程中，常用的方法大体分为两类：黑盒测试和白盒测试。黑盒测试就是依据软件的需求规格说明书，检查程序的功能是否符合用户要求。这种测试方法不需要考虑程序内部的逻辑结构和内部特性，完全把测试对象看作一个黑盒子。黑盒测试通常在软件开发和测试过程的后期进行。软件的白盒测试把测试对象看成一个透明的盒子，它允许测试人员利用程序内部的逻辑结构及有关信息，设计或选择测试用例，对程序所有逻辑路径进行测试。通过在不同点检查程序的状态，确定实际的状态是否与预期的状态一致。^[48]

6.3.2 软件测试的步骤

随着软件工程化技术的发展，软件开发过程与软件测试工作结合得越来越紧密。在软件开发过程的多个重要阶段都需要进行不同类型的软件测试。同时，软件测试作为一个独立的阶段，有一定的目的，要完成相关的任务。这一阶段与开发过程一样，要按步骤进行，这样才能保证测试的完整性和可靠性。测试过程中

每一个步骤在逻辑上是前一个步骤的继续。软件的测试过程一般按 4 个步骤进行,即单元测试、集成测试、确认测试和系统测试。

单元测试集中对用源代码实现的每一个程序单元进行测试,检查各个程序模块是否正确地实现了规定的功能。然后,进行集成测试,根据设计规定的软件体系结构,把已测试过的模块组装起来,在组装过程中,检查程序结构和正确性。确认测试则是要检查已实现的软件是否满足了需求规格说明中确定的各种需求,以及软件配置是否完全、正确。最后是系统测试,把已经过确认的软件纳入实际运行环境中,与其他系统成分组合在一起进行测试。

6.4 本系统典型的测试用例及测试结果

对班级管理信息模块的测试见表 6.1 所示:

表 6.1 班级管理信息模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|----------------------------|--------|------------------------|-------|
| 以教员“jw1”登录系统,点击“班级管理”菜单 | 显示班级列表 | 显示班级列表 | 结果相符合 |
| 点击“添加”按钮,给初一年级增加一个“07 级六班” | 可以添加 | 在初一年级列表下增加了“07 级六班” | 结果相符合 |
| 点击“修改”按钮,将初二二班的班主任改为“李建军” | 可以修改 | 在初二二班的班级描述中,班主任改为“李建军” | 结果相符合 |
| 点击“删除”按钮,将初一六班删除 | 可以删除 | 可以删除 | 结果相符合 |

对学生信息管理模块的测试见表 6.2 所示:

表 6.2 学生信息管理模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|------------------------------------|------------|----------------------|-------|
| 以教员“jw1”登录系统,点击“学生管理”菜单下的子菜单“信息管理” | 显示学生信息管理窗口 | 显示学生信息管理窗口 | 结果相符合 |
| 点击“添加学生”按钮,给初一一班添加学生“张红”的基本信息 | 可以添加 | 点击初一一班后,显示出学生“张红”的信息 | 结果相符合 |
| 点击“修改资料”按钮,将初一一班张红的职务改为“班长” | 可以修改 | 将张红的职务由“体委”改为“班长” | 结果相符合 |
| 点击“删除学生”按钮,将初一一班学生张红删除 | 可以删除 | 可以删除 | 结果相符合 |
| 点击“教育经历”按 | 显示教育经历窗 | 显示教育经历窗 | 结果相符合 |

| | | | |
|----------------------|-------------|-------------|-------|
| 钮 | 口 | 口 | |
| 输入要查询的学生姓名, 点击“查询”按钮 | 显示所要查询的学生信息 | 显示所要查询的学生信息 | 结果相符合 |

对学生奖惩信息管理模块的测试见表 6.3 所示:

表 6.3 学生奖惩信息管理模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|-------------------------------------|----------|----------|-------|
| 以教员“jw1”登录系统, 点击“学生管理”菜单下的子菜单“奖惩管理” | 显示奖惩管理窗口 | 显示奖惩管理窗口 | 结果相符合 |
| 点击“添加”按钮, 添加初一一班李丽在 07 年获奖情况 | 可以添加 | 可以添加 | 结果相符合 |
| 点击“修改”按钮, 更改初一一班李丽在 07 年获奖情况 | 可以修改 | 可以修改 | 结果相符合 |
| 点击“删除”按钮, 将初一一班李丽在 07 年获奖情况删除 | 可以删除 | 可以删除 | 结果相符合 |

对学籍变动管理模块的测试见表 6.4 所示:

表 6.4 学籍变动管理模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|---------------------------------------|----------|---------------|-------|
| 以教员“jw1”登录系统, 点击“学生管理”菜单下的子菜单“学籍变动管理” | 显示学籍变动窗口 | 显示学籍变动窗口 | 结果相符合 |
| 点击“添加”按钮, 添加初三五班杜文在 07.10.9 办理休学 | 可以添加 | 可以添加 | 结果相符合 |
| 点击“修改”按钮, 将初三五班杜文在 07.10.9 办理休学改为“退学” | 可以修改 | 杜文的“休学”改为“退休” | 结果相符合 |
| 点击“删除”按钮, 将初三五班杜文的学籍情况删除 | 可以删除 | 可以删除 | 结果相符合 |

对学生成绩管理模块的测试见表 6.5 所示:

表 6.5 学生成绩管理模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|-------------------------------------|----------|-------------------|-------|
| 以教务员“jw1”登录系统,点击“学生管理”菜单下的子菜单“成绩管理” | 显示成绩管理窗口 | 显示成绩管理窗口 | 结果相符合 |
| 点击“添加成绩”按钮,给初三六班添加物理成绩 | 可以添加 | 可以添加 | 结果相符合 |
| 点击“修改成绩”按钮,修改初三六班张同的成绩 | 可以修改 | 将张同的成绩由“85”改为“94” | 结果相符合 |
| 点击“删除成绩”按钮将初三六班张同的成绩删除 | 可以删除 | 可以删除 | 结果相符合 |

对课程管理模块的测试见表 6.6 所示:

表 6.6 课程管理模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|-----------------------------|----------|-----------------|-------|
| 以教务员“jw1”登录系统,点击“课程管理”菜单 | 显示课程管理窗口 | 显示课程管理窗口 | 结果相符合 |
| 点击“添加课程”按钮,加入“初中美术”,并设置相关信息 | 可以添加 | 在学科列表中显示出“初中美术” | 结果相符合 |
| 点击“修改课程”按钮,修改初三物理课时量 | 可以修改 | 将课时量由“4”改为“5” | 结果相符合 |
| 点击“删除课程”按钮,删除“初中几何” | 可以删除 | 可以删除 | 结果相符合 |

对教师管理模块的测试见表 6.7 所示:

表 6.7 教师管理模块的测试

| 操作 | 期望结果 | 测试结果 | 结论 |
|--------------------------|----------|----------|-------|
| 以教务员“jw1”登录系统,点击“教师管理”菜单 | 显示教师管理窗口 | 显示教师管理窗口 | 结果相符合 |
| 点击“添加教师”按钮,添加“物理教师蔡景玥” | 可以添加 | 可以添加 | 结果相符合 |
| 点击“修改资料”,按钮,修改“尹国才”的教学信息 | 可以修改 | 可以修改 | 结果相符合 |
| 点击“删除教师”按钮,将物理教师“袁 | 可以删除 | 可以删除 | 结果相符合 |

| | | | |
|---------------------|------|-------------|-------|
| 长茂”删除 | | | |
| 输入要查询的教师姓名，点击“查询”按钮 | 可以查询 | 显示出所查询教师的信息 | 结果相符合 |

第7章 讨论

7.1 中学教务管理系统的实施

本中学教务信息管理系统的终极目标是使开发的系统能够为我校提供教务管理服务,系统开发完成以后,问题并没有终结,系统还要接受实际运用过程的检验,这对系统的完善是非常必要的。在系统实施的初级阶段,可能会出现一些问题,总的来说系统出现的问题无外乎三个方面:网络配置的问题、程序本身的问题和系统用户操作不当的问题。所以,在系统实施前要做好计划,实施计划一般分三个阶段进行:

(1) 试运行。系统在正式投入使用之前,试运行一个月,将系统在不同的操作系统下的安装光盘在低、中、高档机器上分别进行安装试运行,在这期间,使用者和系统开发人员就程序的功能性、实用性等方面进行交流,程序员可以就用户的合理意见对程序进行修改和完善。

(2) 培训用户。在试运行的同时,可以对用户进行使用培训,让用户充分了解中学教务信息管理系统的功能和使用方法,并给用户编写详细的使用手册和帮助文档。

(3) 验收正式投入使用。在试运行结束后,用户如果对程序认可验收后,就可正式投入使用。

(4) 做好日常维护工作。专门负责系统的日常维护工作,包括给用户分配用户名、权限等。

开发者在系统的开发过程中付出了很大的心血,但由于时间的仓促和自身水平的限制以及学校也在不断的发展变化,系统还有很多需要完善的地方,这就需要不断的加以更新和补充新的内容。同时加强日常维护工作,尤其是对 SQL Server 数据库的管理。在以后的时间里,将不断推出新的系统更新版本,以满足实际需要。

7.2 中学教务管理系统的特点

从学校的角度来说,学校管理层关心的热点问题莫过于学校的管理方式能最大限度的降低人力管理成本,提高经济效益和社会效益。本中学教务信息管理系统在对本中学教务管理需求分析的基础上,充分考虑学校教务各项工作的需要,建立了包括:学生信息管理,学籍管理,成绩管理,课程管理和教师管理在内的5个子系统。当然,要想熟练使用本系统,还需要软硬件的大力支持,尤其是需要专业人员的专职操作管理。

从技术的角度考虑,本系统的开发采用基于 C/S 的主流系统开发模式,实现了高水平的信息共享。使用数据库管理系统 SQL Server 作为后台管理工具,易于操作和维护。以当前比较流行的 Visual Basic 6.0 企业版作为前端开发工具,并且预留接口,使软件具备较强的开放性和可移植性,能够适应不同的软硬件平台。模块化结构设计使系统可以根据教务工作的需要进行不断扩充,并且还具有易学易用,人机界面友好的特点。

7.3 系统开发过程中应注意的问题

(1) 学校管理者要对教务管理模式要有清晰的认识和思考。由于学校教务管理模式取决于管理信息系统建设的水平,所以,学校管理者必须对教务管理信

息系统中将采用的管理模式有认真的思考和研究,得出明晰的结论,并指导教务管理信息系统的建设。学校管理者的积极参与对于教务管理信息系统建设的成败至关重要。

(2) 软件开发人员要对教务工作有清楚的了解。很多软件开发人员不是领域内的专家,往往过多地把注意力集中在具体的技术细节上,如新语言的使用、编程技巧、新的软件工具的掌握等,而忽略了一个好的应用软件系统要想真正投入使用,首先必须满足用户的需求。考虑到这方面的因素,作为长期工作在教学第一线的教师,充分了解教务的各项工作流程,力求开发出来的系统能够满足学校的实际需要。

(3) 要充分发挥教务处计算机人员的纽带和桥梁作用。教务处计算机人员的优势在于对教务工作的流程比较熟悉,对教务处的人员比较了解,同时对计算机关键技术也比较熟悉。他要承担教务处使用人员的培训、对校领导决策提供参谋意见、对管理信息系统数据和文档进行管理任务。

(4) 采用软件工程的方法以保证项目的稳定性和长期维护。在有些管理信息系统的应用中,常有一个关键人员因故离开,而造成整个系统瘫痪的先例。这种情况最好的防范方法是以现代化软件工程的方法进行项目实施,使所有的系统分析结果和软件的系统规格说明书明确,任何高水平的计算机专业人员都可以根据系统规格说明书来分析系统的特性和所存在的问题,从而确保学校长期利益的维护,解除系统对个人的依赖性。

(5) 程序设计或修改前要高度重视系统分析过程。系统分析是指计算机人员与教务处相关人员共同对工作流程和方式进行透彻的分析,形成信息管理系统的总体框架。系统分析是系统开发不可跳跃的阶段,系统分析的成败在项目实施中举足轻重,必须给予足够的重视,本系统严格按照程序化的设计流程进行设计和开发。

7.4 数据库的安全性

对于数据库而言,由于其中保存了大量的数据,提高其安全性是非常重要的。在本系统中,综合运用以下两种方式增强数据库的安全性:

1、设置数据库密码

对数据库设置密码是最简单的方法,就是为打开的数据库设置密码。设置密码后,再次打开数据库时,将显示输入密码的对话框,要求输入正确的密码后用户才可以打开该数据库。这种方法比较安全。数据库打开以后,数据库中的所有对象对用户本身将是可用的。对于在某个用户组中,共享的数据库或是单机上的数据库,设置密码一般就足够了,但它不能防止使用工具程序读取其中的数据或用其它软件打开该数据库。

2、设置用户访问权限

在本系统中,系统初始化时有一个默认的“系统管理员”用户 Admin,该用户可以创建用户、修改用户信息以及删除用户;普通用户则只能修改自己的用户名和密码,确保未授权的用户不能进行相应的操作,不能存取相应的数据。

7.5 建立一些公共函数或过程

本系统很多模块都使用了相同的函数或过程,为了避免代码显得有些乱,影响可读性,进一步影响系统运行速度。在公用模块建立一些函数或过程,供各窗口模块调用,从而达到精简代码的目的。增加程序可读性,加快系统运行速度。

7.6 丰富和完善查询功能

因为时间关系，没来得及设计模糊查询。所以本系统不支持模糊查询，不能够进行多字段的组合查询。为了提高查询工作效率，经常需要进行多个字段的模糊查询。下一版中，将引进 SQL 语言中的 LIKE 模式匹配关键字和“%”、“_”等通配符实现模糊查询。

第 8 章 总 结

1、通过各种技术方法之间的比较，确定了系统的最优化开发方案：C/S 结构作为开发模式、SQL Server 作为后台数据库、Visual Basic6.0 作为前端开发工具。

2、结合目前我校的特点，通过对教务业务流程分析和业务需求分析，确定中学教务信息管理系统应实现的功能。主要包括学生信息管理、奖惩管理、学籍变动管理、成绩管理、课程管理、教师管理 6 个方面的管理内容。

3、本系统实现了对教务信息的一体化、信息化、智能化管理。教务管理人员可以通过对相关模块的操作输入或查询相关的信息，从而提高工作效率。

4、在本系统开发的过程中充分考虑到了系统的实用性和易操作性，大部分信息的录入、查询、修改、删除等多项功能都是集中在一个窗体中的，这样就减少了用户在使用过程中的窗体切换，提高了软件的使用效率。并且面向对象编程方法的运用使得人机交互界面简单、一致、友好。

目前,本系统的设计和开发工作已经完成，系统测试工作已经结束，现正处在试运行阶段，我校的教务管理人员正在进行部分基础数据的录入工作。试运行阶段结束后，会对发现的问题集中修改，直到系统稳定运行。

在系统实现中，较全面的学习了 VB 编程语言和 SQL Server 数据库的使用。由于时间和能力的限制，这次毕业设计还有很多不足之处，在本系统中还没实现对诸如排课管理、毕业管理、教材管理等模块的设计与实现。这些都有待进一步研究和实现。

参 考 文 献

- [1] 王宪磊. 信息管理论[M]. 北京: 社会科学文献出版社, 2004. 54
- [2] 马费成, 胡翠华, 陈亮. 信息管理学基础[M]. 武汉: 武汉大学出版社, 2002. 253-255
- [3] 柯平, 高洁. 信息管理概论[M]. 北京: 科学出版社, 2002. 56-62
- [4] 朱玉媛. 现代人事档案管理[M]. 北京: 中国档案出版社, 2002. 190-191
- [5] 李勇帆. Visual Basic 6.0 程序设计与应用[M]. 北京: 人民邮电出版社, 2002.
- [6] 姚巍. Visual Basic 数据库开发从入门到精通[M]. 北京: 人民邮电出版社, 2006.
- [7] 郜焕平. Visual Basic 6.0 程序设计教程[M]. 北京: 学苑出版社, 2001.
- [8] 宋琦凡, 付敬平. 使用 Visual Basic 开发数据库应用软件[M]. 北京: 电子工业出版社, 1999.
- [9] 丁彩红, 杨向萍. 在 VB 中通过 ADO 对象实例化来远程访问 SQL Server[J]. 计算机应用研究, 2003. (5): 155-156
- [10] 杜军平, 黄杰. SQL Server 2000 数据库开发[M]. 北京: 机械工业出版社, 2001.
- [11] 郭琦. Visual Basic 数据库系统开发技术[M]. 北京: 人民邮电出版社, 2003.
- [12] 郭瑞军, 唐邦民等. Visual Basic 数据库开发实例精粹[M]. 北京: 电子工业出版社, 2005.
- [13] 何玉洁. 数据库基础及应用技术[M]. 北京: 清华大学出版社, 2002. 1-250
- [14] 胡荣根. Visual Basic 6.0 中文版数据库和 Internet 编程[M]. 北京: 清华大学出版社, 1999.
- [15] 敬铮. Visual Basic 6.0 数据库开发与专业应用[M]. 北京: 国防工业出版社, 2002.
- [16] 李晓黎, 张巍. Visual Basic+SQL Server 数据库应用系统开发与实例[M]. 北京: 人民邮电出版社, 2003.
- [17] 梁海华. Install Shield 安装程序设计与制作[M]. 北京: 人民邮电出版社, 2002
- [18] 陆春友, 贾利荣. VB 应用程序访问 SQL Server 方法探讨[J]. 煤炭技术, 22(7):100-101.
- [19] 陆伟民. 面向对象程序设计的专家系统研究[J]. 计算机科学, 1990.(5):67-70
- [20] 马绍良, 李凤保, 谢阅. 用 ODBC 实现 SQL Server 2000 在 VB 中的应用[J]. 计算机应用, 2003.(1):115-118
- [21] 求是科技. Visual Basic 信息管理系统开发实例导航[M]. 北京: 人民邮电出版社, 2005.
- [22] 赛奎春等. Visual Basic 工程应用与项目实践[M]. 北京: 机械工业出版社, 2005.
- [23] 王行言, 俞盘祥, 汤荷美, 黄维通编著. 计算机信息管理基础[M]. 北京: 高等教育出版社, 1999. 1-185
- [24] 温贤发. Visual Basic 6.0 数据库程序设计高手[M]. 北京: 科学出版社, 2001.
- [25] 西尼奥雷. 开放式数据库互连 ODBC 方案集粹[M]. 北京: 电子工业出版社, 1995.1-296
- [26] 易金聪, 张秀萍, 宁正元. 基于 C/S 与 B/S 模式的管理信息系统的设计与实现[J]. 微型电脑应用, 2003.19(4):46-48
- [27] 尹为民, 李石君等. 现代数据库系统及应用教程[M]. 武汉: 武汉大学出版社, 2005.
- [28] 雍兰利, 张士欢. 信息系统主要开发方法创新的比较分析[J]. 河北师范大学学报, 2005.28(6): 150-154
- [29] 余晨, 李文炬. SQL Server 培训教程[M]. 北京: 人民邮电出版社 2001. 3-5
- [30] 曾伟民, 邓勇刚. Visual Basic 6.0 高级实用教程[M]. 北京: 电子工业出版社, 1999.
- [31] 张莉, 王强等. SQL Server 数据库原理及应用教程[M]. 北京: 清华大学出版社, 2006.
- [32] 张龙详. 数据库原理与设计[M]. 北京: 人民邮电出版社, 2002.
- [33] 张树兵等编著. Visual Basic 6.0 中文版入门与提高[M]. 北京: 清华大学出版社, 1999.

- [34] 郑广成. VB 访问 SQL Server 数据库的几种方案[J].河北科技大学学报, 2004.25(3): 66-69
- [35] 周绪等. SQL Serve 2000 中文版入门与提高[J]. 北京: 清华大学出版社, 2001.
- [36] (美)Mike Gunderloy. Visual Basic 开发指南-ADO 篇[M]. 北京: 人民邮电出版社, 2000.
- [37] A.Wood. Predicting Client/Server Availability[J]. Computer, 1995.28(4)41-48
- [38] Abelson M. Data loss prevention and useful utilities[J]. Am J Orthod Dentofacial Orthop. Mar, 2000.117(3): 367-369
- [39] D.Clark. Visua lBasic and the Future of Web—APPLieation Development[J]. Computer, 2001.34(8):16-18
- [40] Elmasri R A, Navathe S B. Fundamentals of Database Systems[J]. 2nd ed.Benjamin Commings, 1997.56(126):36-37
- [41] Francesco Balena(美). Visual Basic 6.0 编程技术大全[M]. 北京: 机械工业出版社,2000.
- [42] Guttag J. Abstract Data Type and the Development of Data tructures[J]. Communication of the ACM, 1997.20(6):34-41
- [43] Ohn, V.osoPh, M.Thatte. Object~Oriented Database:Design and Implementation[J]. IEEE, 1993.79(1): 25-68
- [44] Roger.Jennings. Visual Basic 6.0 数据库开发人员指南[M].北京: 机械工业出版社, 1999.
- [45] Vaskeviteh. Client/Server Strategies[M]. IDG Books, 2001.45-46
- [46] Microsoft Corporation. Desktop applications with Microsoft Visual Basic 6.0[M]. Beijing: Beijing University Publication,2000.1-489
- [47] Thomas Connolly,Carolyn Begg.Database systems[M]. Beijing: Publishing House of Electronics Industry,2003.
- [48] 肖梦强等. 软件工程: 原理、方法与应用[M]. 北京: 中国水利水电出版社, 2005. 151-156

谢 辞

首先，我要真挚的感谢我的指导老师毕广吉教授。在设计期间，由于我要一边做设计一边在单位工作，时间非常紧张，但是毕老师对我严格要求，对我的问题给予悉心的指导和帮助，使得整个设计过程得以顺利地进行，并最终顺利完成。

再有，感谢天津师范大学所有教过我的老师们对我的培养，让我圆满地完成了学业！

由于本人水平有限，论文中的缺点、错误在所难免，敬请前辈专家们给予批评指正。

寇薇薇

2008年3月