

论 文 题 目： 昂立信息企业员工管理系统设计与实现
工 程 领 域： 软件工程
指 导 教 师： 张扬 教授
作 者 姓 名： 刘长青
班 学 号： 200890208025

分类号_____ 密级_____

UDC ^{注1} _____

学 位 论 文

昂立信息企业员工管理系统设计与实现

刘长青

指导教师姓名 张 扬 教 授

电子科技大学 成 都

齐洪星 高级工程师

沈阳昂立信息技术有限公司 沈 阳

(职务、职称、学位、单位名称及地址)

申请专业学位级别 硕士 专业学位类别 工 程 硕 士

工程领域名称 软 件 工 程

提交论文日期 2010.9 论文答辩日期 2010.11

学位授予单位和日期 电 子 科 技 大 学

答辩委员会主席 _____

评阅人 _____

2010 年 月 日

注 1：注明《国际十进分类法 UDC》的类号

独 创 性 声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，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，也不包含为获得电子科技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谢意。

签名：_____ 日期：_____ 年 _____ 月 _____ 日

论 文 使 用 授 权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后应遵守此规定）

签名：_____ 导师签名：_____ 日期：_____ 年 _____ 月 _____ 日

摘 要

摘 要

随着中国经济的飞速发展和现代技术的不断进步，计算机技术快速发展。信息化浪潮、网络革命在给社会带来巨大冲击的同时，也使企业管理备受考验，旧的企业管理方式已经不能适应现代信息化方式的需求。企业员工网络化管理对企业的发展起着至关重要的作用，在现代社会中越来越受到重视，很多公司都会采用企业员工管理系统来对他们的员工进行数字化、信息化管理。

在计算机飞速发展的今天，将计算机这一信息处理利器应用于企业管理已是势所必然，而且，将会给企业带来前所未有的改变，它可以带来意想不到的效益，同时也为企业的发展提供了无限的潜力。采用计算机管理企业是企业走向科学化和现代化的最明显标志，它给企业带来了明显的经济效益和社会效益。企业员工管理系统通过员工管理系统查看员工基本档案、员工工资信息、员工出勤信息，使企业的管理更加方便、快捷，节省了不少成本；还可以使企业实现网络化、信息化，可以方便管理人员更高效的管理员工的信息，这样既能弥补以前人工管理的不同，又能够给员工的管理带来更高效的发展，因此，建立一个完整的、安全的、稳定的企业员工管理系统是非常必要的。

笔者在调研中发现，企业的员工管理系统很多是从管理信息系统演变而来，从单一的人事行政管理或人力资源管理的业务需求角度出发设计，服务于某一具体业务的自动化操作需求；而且，虽然目前国内的企业员工管理软件已将模块功能扩展至企业人力资源管理或人才资本管理相关的整个业务领域，但系统在整体性、前瞻性和集成性方面仍有欠缺。因此，本文对企业员工管理系统采用面向对象的开发方法，以昂立信息企业为研究载体，对系统进行简单的需求分析以及相关技术分析、数据库设计、系统实施和系统测试。昂立信息企业员工管理系统实现了对员工档案的管理、员工档案的查询、员工工资的核算、员工考勤查询、月部门考勤、年度员工考勤、员工工资查询、工资的发放、管理员权限设置及部门设置等等。各模块之间采用了 c#语言、ASP.NET 技术、SQLSerever2005 数据库进行页面开发和数据库的设计，使模块之间能够实现相互链接。

关键词：企业员工管理系统，SQL Server2005，C#

ABSTRACT

ABSTRACT

With the rapid development of Chinese economy and modern technology progress, the rapid development of computer technology, network, information revolution wave brings to the society, also the huge impact test is that enterprise management. Enterprise management mode of old already can not adapt to the modern way of information needs. Enterprise employees to the development of enterprise network management plays an important role in the modern society, more and more attention, many companies will be adopted to enterprise staff management system for their employees on digital management.

With the rapid development of the computer, the computer information processing applications in enterprise management tool will bring an unprecedented changes. It can bring unexpected benefit, but also for the development of the enterprise offers unlimited potential. Using the computer on management enterprise is scientific and modern enterprises, it is bringing the obvious economic benefits and social benefits. Enterprise staff management system through the management system of the employees' view files, employee wages and information, the employee's attendance information, make enterprise management more convenient, quick, costs-saving. We can make the enterprise informatization can be conveniently more efficient management staff of information, so as to compensate the artificial management, before the management and employees can bring more efficient development. Therefore, to establish a complete, safety and stability of the enterprise staff management system is very necessary.

Based on investigation, the enterprise staff management system from the management information system for many evolved from a single, the personnel administration or human resources management of the business needs of Angle design, service in a particular business automation needs. And, although at present domestic enterprise staff management software module make a function to the human resources management or related to human capital management, but the whole business integration, and system integration are still lacking. Therefore, this article to enterprise staff management system used object-oriented method, with the development AngLi

ABSTRACT

information for the study of enterprise carrier. For example, the simple system requirements analysis and related technical analysis, database design, system implementation, and system test. Enterprise staff management system will realize the staff files management, Employees of archives, Salaries; Wage calculation; Monthly attendance sector; In personal attendance; Employee wages query, Attendance query, Permissions Settings: departments. Each module, using c # ASP.net page SQLServer2005 technology, database development and database design, makes each link between modules.

Key words: Enterprise Staff Management System, SQLServer2005, C#

目 录

第一章 绪论	1
1.1 系统的开发背景	2
1.2 本研究的目的和意义	3
1.3 国内外现状	4
1.4 本研究工作说明	6
第二章 系统开发的技术基础	8
2.1 B/S 体系结构	8
2.2 .NET 框架	8
2.3 ASP.NET 简介	9
2.4 C#语言简介	10
2.5 数据库及 SQL 语言简介	12
2.6 ADO.NET 简介	13
第三章 系统需求与可行性分析	15
3.1 系统的需求分析	15
3.1.1 公司员工管理系统设计定位	15
3.1.2 公司员工管理系统的预期设计	15
3.2 系统开发的可行性分析	16
3.2.1 技术可行性分析	16
3.2.2 经济可行性分析	17
3.2.3 管理可行性	17
3.2.4 开发环境可行性	18
3.3 系统用例模型与系统文档	18
3.3.1 公司员工管理系统的用例模型	18
3.3.2 公司员工管理系统的用例文档	19
3.4 用例的业务过程	23
第四章 员工管理系统的总体设计	25
4.1 系统设计架构	25
4.2 系统功能模块及组成	26

目 录

4.2.1 系统管理模块	26
4.2.2 考勤管理模块	26
4.3 数据库设计	27
4.3.1 数据库设计的原则	27
4.3.2 数据库设计的步骤	28
4.3.3 数据库逻辑结构设计	29
第五章 系统的详细设计与实现	36
5.1 系统描述	36
5.1.1 注册登录	36
5.2 开发工具的选择	38
5.3 页面设计和关键代码	39
5.3.1 系统页面设计	39
第六章 系统测试	42
6.1 系统测试基本原则	42
6.2 测试方法	42
6.3 系统测试过程	43
6.4 测试的结果	47
第七章 结束语	48
7.1 结论	48
7.2 进一步工作的方向	48
致 谢	50
参考文献	51
附录 1: 关键代码及部分编程实现	53

第一章 绪论

随着计算机和网络技术的发展，人类已经大踏步跨入信息化时代。信息技术和信息产业的飞速发展，使得信息化时代的影响已经遍及全球，目前信息技术的应用变得越来越广泛。这不但促进了整个社会的快速进步，同时也改变着人们的工作、学习、生活和娱乐的方式，以及人们的思想观念。在人类进入信息化社会时，随着计算机在各个行业的广泛应用，计算机软件的开发与应用也在不断深入，并且已经成为新兴信息产业的支柱。

计算机技术的快速发展，特别是网络技术的出现，标志着信息化时代已经来临。信息化浪潮、网络革命在给整个社会带来巨大冲击的同时，也使得企业的管理备受考验，陈旧的企业管理方式已经越来越不能适应现代信息化方式的需求。随着企业的不断壮大和人员的不断增加，员工管理就相应更为复杂，手工方式已经不能满足时代变换的需求，如何运用先进的信息技术管理手段，提高企业管理水平，是各个企业要面临的一个崭新的挑战。

一直以来，人们通过传统的方式管理员工资料，这种方式的利弊，随着时间的推移，日益显见。在计算机还没有在企业管理领域广泛使用之前，管理企业员工主要依靠人工方式来完成，这种管理方式的不足之处是显而易见的。查询和管理员工信息的效率很低，能够管理的员工数量也是十分的有限。利用计算机来处理这些数据无疑会最大程度地提高工作效率和处理能力。我们将会看到无论企业员工再多，资料再复杂，工作人员管理的效率都会越来越高。

随着行业之间竞争的不断加剧，新的管理方式不断涌现，对企业管理的影响也越来越深刻，这就要求企业员工管理系统必须要适应新的需求变化，为企业发展的需要提供了有效的拓展和革新途径。目前，个别企业已经实现了企业管理的数字化和信息化，经过实践检验证明成效显著。但由于竞争环境更新飞快，越来越激烈，旧的方式极易被淘汰，所以我们需要不断更新和优化管理系统，通过对目前的系统进行升级改造，建设一个新的企业员工管理系统，从而为企业的快速发展提供一个良好的技术支撑。

为了方便对企业员工的高效管理，特开发该管理系统以提高员工管理的效率。使用该系统之后，管理人员可以轻松的对员工档案、员工工资、员工出勤、管理员权限进行管理，以便全面进行员工的查询和管理。

1.1 系统的开发背景

人类文明的进步已经跨入信息时代，互联网的应用为人类生活提供了前所未有的信息资源，并为人类的生活提供了优质的服务，使得人类的生活与计算机密不可分。人们可以通过网络购物、收集信息、开展业务、休闲娱乐等。计算机的普及为网络带来了空前的发展空间。

计算机技术的普及和推广，网络需求的不断扩大，使其应用领域也越来越宽广。在将其引入到对现代企业、事业单位和学校等单位的管理后，显示出了明显的优势，并昭示出美好的发展前景。我国现阶段在应用计算机对企事业单位进行管理方面已经取得了巨大的成就。

随着网络的不断的发展，计算机作为知识经济时代的产物，已被广泛应用于人类生活的各个层面。目前，我国科学技术水平高速发展，计算机作为使用最广的现代化工具已深入到各个领域，并成为未来社会的重要支柱。在这样的大背景下，计算机管理已经成为一种必然趋势，因此，小型或中型的员工管理系统软件作为一个时代的产物，已经成为一个单位所必备的系统，人工管理逐渐被计算机所取代而去不复返，利用计算机实现企业人力资源的管理势在必行。对于企业来说，利用计算机支持企业高效率完成人力资源管理的日常事务，是适应现代企业制度要求、推动企业劳动人事管理走向科学化、规范化的必要条件；而人力资源管理是一项琐碎、复杂而又十分细致的工作，不允许出现差错，如果实行手工操作，每个月必须手工填制大量的表格，这种管理方式耗费了工作人员大量的时间和精力，而且还可能出现错误。而利用计算机软件对人力资源进行日常的管理工作，具有着手工管理所无法比拟的优点，它不仅能够保证准确无误、快速便捷，而且还可以利用计算机软件对相关的各种信息进行统计、分析。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、成本低等等。这些优点能够极大地提高企业人力资源管理的效率，这也正是企业真正实现科学化、正规化管理，与世界进行接轨的重要条件。因此，近年来，企业员工管理系统（即人力资源管理系统）便应运而生了^[1]。

现代企业的生存与发展受到了众多因素的影响和制约。其中，最重要、最根本的因素就是企业的劳动者——人力资源，他们是企业最宝贵的资源，因而进行有效管理变得尤为重要。但是在实际操作中，许多企业更关心的是资金和市场问题，人力资源管理常常被忽视。然而，恰恰是人力资源的瓶颈阻碍着企业的进一步发展，可见人力资源开发与管理在企业的发展过程中发挥着举足轻重的作用。

第一章 绪 论

一个企业要想在激烈的竞争中取胜，除了采用正确的策略、先进的技术，广纳人才之外，更重要的一点就是如何打造良好的团队模式。如人员的增减、调动，员工的职务、工资等问题，如何充分发挥员工的积极性和主动性，挖掘员工的创新精神，在实现人性化管理的同时创造最大的经济效益和社会效益。这就迫切需要一种有效的现代化的管理系统，以便实现对员工信息的充分掌握。

企业员工管理系统既能弥补以前人工管理的不足，又能够给员工的管理带来更高效的发展。我们可以通过员工管理系统查询员工基本档案、员工工资信息、员工的出勤情况，使企业的管理更加方便、快捷，同时降低企业运行成本；还可以使企业实现网络化、信息化，方便管理人员更高效地管理员工的信息。因此建立一个完整的、安全的、稳定的企业员工管理系统是非常必要的。

总部设在沈阳浑南高新区的昂立信息技术有限公司，成立于 1993 年，是沈阳一家专门从事信息技术服务的高科技股份制企业。公司现有员工 400 余人，资产近亿元，建有一期“昂立信息产业园”，占地 70 余亩，是沈阳市软件加工基地和数字娱乐产业基地。2007 年，又投资建设了沈阳国际软件园，将打造成为“东北亚信息中心”。该公司作为沈阳软件业的领军企业，已经初步实现了数字化、信息化管理，有效地促进了企业的发展，但由于新的竞争环境下，越来越不能适应新发展的需要。希望通过对目前系统进行改造，建设一个新的企业员工管理系统，提高对员工管理的效率，从而为企业的快速发展提供一个良好的支撑体系，使企业不断做大、做强。

1.2 本研究的目的和意义

21 世纪是信息化的新时代，信息作为社会最主要的资源，将成为企业发展重要的战略资源而备受广泛关注。那么，如何获取更加有效的信息呢？人类在过去的历史时间里，只能用自身的感觉去收集信息，用大脑存储和加工信息，用语言来交流信息。而当今社会正从工业社会走向信息化社会，各种信息浩如烟海。为了全面、深入、精确地认识和掌握这些信息所带来的巨大的行业潜质和发展机遇，必须使用计算机进行信息处理。目前，数据处理正在广泛地应用于办公自动化、企业管理、事务管理、情报检索等领域。数据处理已经成为计算机应用的一个重要方面，作为计算机应用的一部分，使用计算机对信息进行管理，具有很大的进步性，如检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命高、成本低等。这些优点能够极大地提高企业管理的效率，也是企业的信息化、科学化、

正规化管理，与世界接轨的重要条件。

计算机的应用已渗透到了社会的各个行业，正在迅速地改变着传统的工作、学习和生活方式，推动着社会发展。自从管理信息系统这一新兴的管理方法引入我国，企业管理就走向了信息化的方式，使企业得到了迅速的发展，而信息化的管理方式已经广泛应用于各行各业。虽然我国计算机信息管理系统建设在 20 世纪 80 年代就已经开始起步，但发展极不平衡，也就是说，大部分企业目前仍处于低水平开发和应用阶段。随着经济体制的改革和市场经济的发展，企业的运行机制有了很大的变化。企业的生产更多依赖于市场的竞争，信息从而成为企业的重要资源，信息系统的建设已成为企业走向现代化的重要标志，信息系统对于企业的运作以及市场中的竞争有重要作用。在当前形势下，信息化已经成为企业发展的必由之路。管理信息系统是先进的科学技术和现代管理相结合的产物，建立以计算机为主要手段的管理信息系统，已经成为现在企业和政府部门等各类单位提高自身素质、实现强大组织的战略目标。

企业管理在发展的过程当中，起着至关重要的作用。企业管理是企业发展的源头活水，能使企业的运作效率大大提高；能让企业有明确的发展方向；使每个员工都能充分发挥他们的潜能；使企业财务清晰，资本结构合理，投融资恰当；能向顾客提供更加满足的产品和服务；能很好地树立企业形象，为社会做出更大的贡献。随着信息技术的应用，企业领导对企业管理发挥作用的期望也越来越高，他们希望能够更全面、更快捷、更方便、更准备地获取员工的信息。在没有计算机、软件和网络通讯技术的时代，企业发布信息的手段通常采用开会、发文件等手段，除了信息发布的成本高之外，其传播速度、传播量，更新、查阅及查阅权限等等都受到了极大的制约，这已经不能满足现代企业管理的要求，更加阻碍了企业的快速发展。对于企业管理人员来说，面对日益增长的员工数量，手工运作的工作量太大，而且很容易出错。为了满足企业的需要，努力达到“领导满意”，使员工信息全面，管理人员以及个人查询方便，根据企业发展的需要，提高竞争力，我们决定利用信息化来提高企业的管理水平和工作效率。因此，开发一个界面友好，便于操作的企业员工管理系统软件进行自动化处理，这就是本系统开发的目的和意义^[2]。

1.3 国内外现状

国外人力资源管理系统起步得较早，完整性和成熟度都也较高，且伴随着管

第一章 绪 论

理理论的发展，其设计思路蕴涵了先进的管理理念，并据此开发出了适用于不同行业的解决方案，其优势主要体现在以下几个方面：

- (1) 具有雄厚的实力，在技术力量的培养、研发、市场推广等方面投入巨大；
- (2) 具有一定包括硬件厂商、数据库公司、咨询公司在内的合作伙伴，形成了强强联手的格局。

实际上，西方发达国家早已经广泛使用人力资源管理系统，其他一些企业管理系统，诸如 ERP(企业资源计划)、WFM(工作流管理)等在开发过程中就预留了与人力资源管理系统接口，使得人力资源管理系统今后的扩展难度降低。同时，国外人力资源管理系统对于基于互联网技术的研究和支持同样不遗余力，很多人力资源管理系统其最新版本是完全基于互联网架构的版本，使得其易用性和扩展性大大加强^[1]。

由于中国企业的管理传承于上世纪 80 年代工厂制管理的模式，所以派生出来的传统管理思想往往强调的是控制和效果，不怎么关心效能和效率；组织结构常常是金字塔式，人只是一个机械而单一性的工作者。因此，目前中国的企业员工管理系统情况是：员工管理软件系统大多源自管理信息系统(MIS)，从部门业务需求的角度出发设计。管理信息系统设计的初衷是服务于企业内绝大多数的业务操作人员，将业务操作人员的重复性劳动予以初步自动化，即从管理理论抽象出理想化的业务管理模式，在基于该业务模式的基础上实现低层次的数据处理或业务流程电子化，因此，管理信息系统的设计，是根据中小型企业业务单元的需求来编写的，一般无法满足多体制、多元化、多重组织结构的大型企业数据处理的需求。而现代化管理强调是授权、开放、培育能力和激情，人被视为一种人性化的知识工作者。因此，目前国内的员工管理软件存在如下的缺陷亟待改进：

第一，很多是从管理信息系统演变而来，从单一的人事行政管理或人力资源管理的业务需求角度出发设计，例如人事资料管理、员工考勤管理，或工资计算与发放管理等，服务于某一具体业务的自动化操作需求；

第二，虽然目前国内的企业员工管理系统已添加了企业人力资源管理或人才资本管理相关的模块，但系统在整体性、前瞻性和集成性方面仍然有所欠缺^[2]。

20 世纪 70 年代末，由于计算机技术的飞速发展，无论是计算机的普及，还是计算机系统工具和数据库技术的发展，都为企业员工管理系统的阶段性发展提供了可能。20 世纪 90 年代末，随着信息和知识经济时代的到来，企业对员工管理系统有了更高的需求；同时由于个人电脑普及，数据库技术，客户 / 服务器模式，特别是互联网技术的发展，使员工管理系统的出现成为必然。

1.4 本研究工作说明

在本研究中，设计开发一套基于 ASP.NET 技术的考勤和人员管理信息系统。本企业员工管理系统采用的是结构化系统设计方法，在系统需求分析的过程中，根据用户需求设计用例，采用用例驱动模型与用户进行交流，确定系统的总体功能模块。

在具体实施过程中，采用面向对象的程序设计和开发方法，系统采取 B/S 结构，使用 ASP.NET 开发框架，以统一建模语言(UML)对系统进行分析，采用 Rational 统一过程(RUP)作为系统开发方法，使用 Visio 制图工具生成图例，使用基于角色访问控制(RBAC)的机制；数据库采用 SERVER 2005，利用 ADO.NET 数据库访问技术实现对数据库的管理操作，实现了企业员工管理系统的各项重要功能^[3]。

该系统的基本生命周期划分为：系统规划、系统分析、系统设计、系统实施四个阶段。

系统规划：就是要在充分调研的基础上，认真撰写出论文的开题报告，熟悉相关的软件设计技术，配置好软、硬件开发所需的环境。

系统分析：就是要查阅各种相关的资料，并根据市场分析、导师的指导做出系统需求分析。

系统设计：就是要完成系统的概要设计以及各模块的详细设计等等。

系统实施：就是要进行编码、软件测试和论文设计，设计出能够实现的企业员工管理系统。

在管理系统设计上将采用统一建模语言 UML(Unified Modeling Language)。统一建模语言 UML 是一种定义方便、表达容易、功能强大且普遍适用的建模语言，它融合了多种优秀的面向对象的建模方法，以及多种得到认可和实践检验后的软件工程方法。通过统一的表示法，使不同知识背景领域的专家、系统分析和开发人员以及用户都可以方便地进行交流。因此，可以实现大型复杂系统的各种成分描述的可视化说明并构造系统模型，以及建立各种所需要的文档；因此本系统的设计开发中采用 UML 来建模。

本论文的内容共分为 6 章。

第一章 绪论。首先介绍系统的开发背景，然后介绍研究的目的和意义、国内外研究现状以及本研究的工作说明等等。

第二章 系统的开发技术基础。介绍 B/S 体系结构；.NET 框架；ASP.NET 简介；C#语言简介；数据库及 SQL 语言简介和 ADO.NET 简介。

第一章 绪 论

第三章 昂立企业员工管理系统需求与可行性分析。首先通过公司员工管理系统设计定位和公司员工管理系统的预期设计介绍系统的需求分析；然后通过技术可行性分析、管理可行性、开发环境可行性、经济可行性分析进行系统开发的可行性分析，通过公司员工管理系统的用例模型和公司员工管理系统的用例文档介绍了系统的用例模型与文档，最后介绍用例的业务过程。

第四章 员工管理系统总体设计。首先介绍本系统采用的典型的 3 层结构模式：Web 显示层、数据访问层和数据库层，并给出系统的总体设计架构图，然后介绍系统管理模块和考勤管理模块，数据库设计的原则和数据库设计的步骤以及数据库逻辑结构设计：体系结构和设计表结构。最后给出在员工管理系统中，涉及的数据表：操作用户表、工资发放表、公司部门表、员工档案表、考勤记录表。

第五章 系统的详细设计与实现。首先，要介绍注册登录和考勤系统的系统描述，然后再介绍本系统选择的开发工具、操作系统、编程环境及数据库服务器，最后给出系统界面设计和关键代码。

第六章 系统测试。首先，要介绍本系统采用的测试方法：这包括黑盒测试和白盒测试两种测试方法。然后通过链接测试、表单测试、Cookies 测试、设计语言测试、数据库测试等介绍，对员工管理系统进行的测试，以及发现在系统测试过程中存在的具体问题。最后，通过图形测试、内容测试、整体界面测试、性能测试介绍系统可用性测试等等。

结束语 对本论文的主要研究成果进行归纳总结和进一步展望。

第二章 系统开发的技术基础

本系统针对中小型企业事业单位对人事管理的需求，开发面向中小型企业的员工考勤和培训管理系统，力求方便快捷实现员工人事档案的现代化管理。

2.1 B/S 体系结构

B/S(Browser/Server)结构，即浏览器和服务器结构。当今 Internet 技术发展方兴未艾，技术革新进程不断加快。对 C/S 结构的一种变化或者改进的结构 B/S 三层体系结构在数据管理层(server)和用户界面层(Client)增加了一层结构，称为中间件(Middleware)，使整个体系结构成为三层。三层结构即是伴随着中间件技术的成熟而逐渐兴起的，核心概念是利用中间件将应用分为表示层、业务逻辑层和数据存储层三个不同的处理层次。三个层次是从逻辑上划分的，具体的物理分法可以有多种组合。作为构造三层结构应用系统的基础平台，中间件提供了以下主要功能：负责客户机与服务器、服务器与服务器之间的连接和通信；实现应用与数据库的高效连接；提供一个三层结构应用的开发、运行、部署和管理的平台。这种三层结构，层与层之间相互独立，任何一层的改变都不会影响其它层的功能^[4]。

2.2 .NET 框架

.NET 是微软近年推出的一种战略平台，是目前最为流行的商业企业级应用平台之一，它采用三层或多层结构，功能强大。在构建企业应用上，它吸取了 Java 技术的优点，并进行了合理的扩展。.NET 已经成为一个概念、平台和未来发展的重要方向，它由以下部分组成：

(1).NET 战略：所有的设备将通过 Internet 连接在一起，同时所有的软件将成为在该网络上提供的一种服务(XML)。

(2).NET Framework：它为 MICrosoft 公司的.NET 战略的建立提供了基础和渠道。该框架提供了具体的服务和技术，便于开发人员创建相应的应用程序用来支持连接到 Internet 上的用户的需要。

(3).NET 企业服务器：指.NETFramework 应用程序使用的服务器产品，例如 Microsoft SQL Server 2005。它们将使用.NET Framework 编写，例如使用 Visual Basic 或 C#语言编写存储过程。.NET 框架实现了语言开发、代码编译、组件配置、程序运行、对象交互等各个层面的功能，为 Web 服务及普通应用提供了一个托管、安全、高效环境。.NETFramework 从高到低包括四个层次：应用程序开发技术、类库、基类库、CLR(Conunon Language Runtime)。

(4) CLR 是在.NET 的运行时的一个软件引擎，用于执行和管理任何一种针对.NET 平台的语言编写的代码和异常处理工作，保证类型安全。• NET FrameWork 体系结构^[5]。

2.3 ASP.NET 简介

ASP 是 Active Server Pages 的简写，是由美国 Microsoft 公司推出的一种服务器端动态网页编写脚本，是基于公共语言运行库的程序构架，使用它可以在 IIS (Internet Information Server) 或 PWS (Personal Web Server) 等 Web 服务器端创建功能强大的 WEB 应用程序，它是新一代开发 Web 应用程序的平台，为开发人员提供了一个崭新的网络编程模型。随着 ASP 的广泛使用，软件开发人员发现“经典”的 ASP 模型中有许多问题，主要表现在：ASP 语法分析器不太可靠，性能比较低；在 ASP 中使用 Application 和 Session 对象来管理状态时，这些对象需要用户的 Web 浏览器打开 Cookie；ASP 页面混合了 HTML 代码和服务端脚本，所有的代码在整个文件中混合，十分杂乱。正是由于原有 ASP 模型存在许多的问题，而且与 Microsoft 的.NET 战略架构有许多冲突，因此，Microsoft 推出了基于.NET 平台的新一代 Web Forms 窗体和 XML Web Services 的开发平台——ASP.NET。

ASP.NET 是建立在公共语言运行库上的编程框架，ASP.NET 与 C#是当前非常流行的 Web 应用程序开发组合，具有灵活方便和易于扩充的特点，所以也越来越多的被应用于 Web 应用程序开发。ASP.NET 继承了.NET 框架中的编程模型，与以往的编程模型相比，这种新的编程模型提供了基于组件的架构和开发范例，因此提供了更易于编写、结构更清晰的代码，这些代码很容易进行再利用和实现共享；ASP.NET 使用编译后的语言，从而提升性能和伸缩性；ASP.NET 使用 Web 表单使开发更直观，利用面向对象技术促进组件的再利用。另外，ASP.NET 中还包括有 Web 控件、页面事件、缓冲技术以及服务器控件和对数据捆绑的改进。供 ASP.NET 使用的库以及在 Microsoft.NET 框架中允许通过 Web 使用客户商用函数，为程序

员提供了更多新的开发机会。ASP.NET 提供了下面一些超越以前 WEB 技术的重要特性^[6]：

(1) 优良的性能：ASP.NET 在服务器上运行的是经过编译的 CLR 代码。与以前的技术相比，ASP.NET 代码不再是解释型的脚本，而是运行于服务器端的编译好的公共语言运行库代码；同时引进了早期绑定、本地优化、实时编译、缓存服务等技术，大大提高了 ASP.NET 的执行效率。

(2) 大量工具箱的支持：ASP.NET 框架补充了 Visual Studio 集成开发环境中的大量工具箱和设计器，支持所见既所得、控件拖放、编译调试等功能，使得开发 ASP.NET 程序更加快速方便。在 Visual Studio 的开发环境中，ASP.NET 具有丰富的工具箱和设计器以及许多的第三方控件。

(3) 强大的功能和良好的机动性：因为 ASP.NET 是建立在 CLR 基础上的，所以整个平台的功能和适用性更加适合网络应用程序的开发。

(4) 崭新的配置管理方式：ASP.NET 使用的是一种基于文本格式，等级式的配置系统。

(5) 良好的可移植性和良好的适用性：当 ASP.NET 的一条线程出现异常情况，一个新的进程会在它的地方重新建立，继续担负原进程所处理的任务，这就使应用程序能不中断地处理客户的请求。

(6) 超强的可扩展性：ASP.NET 崭新的结构体系允许开发者开发自己的插件，且 ASP.NET 下任何一个组件都能够被用户自己开发的组件扩展或替换。

(7) 简易性：ASP.NET 提供了许多常用功能的控件，使诸如表单提交、表单验证、数据交互等常用操作变得更加简单。另外，公共语言运行库利用托管代码服务简化了开发过程。

ASP.NET 提供了很高的可重用性，ASP.NET 采用全新的编程环境，代表了技术发展的主流方向。所以，本系统采用 ASP.NET 作为开发工具，脚本语言采用 C# 语言来进行开发。

2.4 C#语言简介

C#(是 C Sharp 的简称)，是 MicroSoft 公司 2006 年 6 月发布的一种全新的简单、安全的新一代面向对象的高级语言产品，是可用于创建运行在 .NET CLR 上的应用程序的语言之一，C#看起来与 Java 相似，它包括了单一继承和界面，与 Java 的的语法几乎相同，C#与 Java 的不同之处在于：它与 COM（组件对

象模型)是直接集成的,它是从C和C++语言演化而来,是Microsoft专门为使用.NET平台而创建的,作为Microsoft公司.NET windows网络框架的主角。它是集成在Visual.Studio.NET中,体现了当前最新的程序设计技术的功能和精华。C#的面向对象技术已经成熟,而且效率很高,例如:能支持良好的类结构,并对继承性有一定的限制。正如Microsoft对C#的定义:“C#是一种类型安全的、现代的、简单的,由C和C++衍生出来的面向对象的编程语言,它是牢牢根植于C和C++语言之上的,并可立即被C和C++开发人员所熟悉。C#的目的就是综合BASIC语言的高生产率和C++的行动力”。

(1) C#的特点体现在^[7]:

- 可避免指针,语法更简单、易学;
- 语言的兼容、协作交互性;
- 可支持跨平台;
- 现代快速应用开发(RAD)功能;
- 与XML的天然融合;
- 面向对象且避免了多继承性;
- 对C++的继承且类型安全。

(2) C#与VB

C#和VB最明显的区别是:C#编译为MSIL,而VB编译为内部机器代码;C#运行时的优点是:它允许C#代码与其他语言编写的代码交互,允许C#代码使用.NET基类提供的丰富功能。

从特性上看,C#对面向对象的支持更加全面。从语法上看,C#比VB更加简洁,允许变量同时声明和初始化。

(3) C#与C++

C#只是C++另一种形式,可以将普通的C++代码内嵌到C#代码的“不安全”块中。

(4) C#和Java

Java和C#语法非常类似,甚至Java类库和.NET基类的结构也非常的相似,并且都依赖于一个中间的运行环境。

(5) C#语言和其他支持ASP.NET的语言相比较,其特点主要体现在:^[8]

✧ 简单性

C#语言首要的目标就是简单，去掉了许多 C++中容易引发程序错误的地方，如内存管理和指针等。同时也解决了存在于 C++中已经有些年头的多余东西。

✧ 现代性

C#作为企业级的编程语言，新增的金融数据类型专用于金融计算方面，很受欢迎。并且，C#提供了元数据语法，能胜任现代应用程序的安全性问题。

✧ 面向对象性

C#支持所有关键的面向对象的特性，如封装、继承和多态性等。在 C#语言中，对象模式不再是编程语言的一部分，而只是基础的一部分。

✧ 类型安全性

在 C#中，实施最严格的类型安全，以保护自己和垃圾收集器。C#要求遵守相关变量的规则，取消不安全的类型转换，禁止使用没有初始化的变量，采用了边界检查防止出现越界问题。

✧ 版本可控性

C#可支持版本控制，使程序员保证版本控制成为可能，避免了以往难以回避的“DLL 地狱”。

在过去的 20 多年中，C++一直是开发商品化软件使用最广泛的语言，给开发人员提供了大量灵活的进行底层控制的能力，但这种灵活性是以开发的效率为代价的。C#在保证强大的功能和灵活性的同时，给 C 和 C++带来了类似于 VB 的快速开发，并且它还针对 .NET 作了特别设计。C#不仅可以快速的构建从底层系统级到高层商业组件，而且使用 C#语言构建的这些组件，能够很容易的转换为 Web Services，从而可响应来自 Internet 的与平台和开发语言无关的访问。因此，在本研究中，笔者采用 C#与 ASP.NET 结合进行 B/S 结构编程。

2.5 数据库及 SQL 语言简介

SQL(Structured Query Language, 结构化查询语言)是一个功能强大的数据库语言。SQL Server 2005 是 MicroSoft 5 年来具有里程碑性质的企业级数据库产品。跟以往的数据解决方案相比，SQL Server 2005 具有非常丰富的新特性：

(1) 通过提供一个更安全、更可靠和更高效的数据管理平台，增强企业组织中用户的管理能力，大幅提升 IT 管理效率并降低运作风险和成本；

(2) 通过提供先进的商业智能平台满足众多客户对业务的实时统计分析、监控预测等多种复杂管理需求, 推动企业管理信息化建设和业务发展;

(3) SQLServer 2005 提供了一个极具灵活性和扩展性的开发设计平台, 不断拓展设计者的应用空间, 实现 Internet 数据业务互联。由于进行了严格的安全控制, 减少了应用程序的开发时间, 极大提高了系统的可伸缩性和性能。在支持现实工作中最苛刻的企业级系统需求的道路上, SQLServer 2005 迈出了极大的一步。

SQL 语言是所有关系数据库的公共语言。具有非过程化语言、统一的语言等十分突出的优点。同时, SQL 为许多任务提供了命令, 这其中包括^[9]:

- (1) 查询各类数据。
- (2) 在数据表中插入、修改和删除记录。
- (3) 建立、修改和删除数据对象。
- (4) 控制、存取数据和数据对象。
- (5) 能够保证数据库的一致性和完整性。

SQL 功能强大, 是一种完备的数据处理语言。它不仅能够用于数据库查询, 而且还能用于数据库中数据的修改和更新。概括起来, 它可以分成以下三组: 一组是 DML (Data Manipulation Language, 数据操作语言): 用于对数据进行检索或者修改, 包括 CREATE, ALTER 和 DROP 等语句; 二组是 DDL (Data Definition Language, 数据定义语言): 用于定义数据的结构, 比如对数据库对象进行创建、修改或者删除; 三组是 DCL (Data Control Language, 数据控制语言): 用于对数据库用户的权限进行定义^[4]。

总之, SQL Sewer 2005 的性能在安全性、可靠性、可管理性等方面都有了极大地提高, 不仅能够确保企业数据业务的实时稳定运行, 还能够大大提高管理效率, 降低操作复杂度和运作成本。因此, 本研究中采用 SQL Sewer 2005 作为数据库开发语言, 可谓是最佳的选择。

2.6 ADO.NET 简介

ADO.NET 是 ActiveX Data Objects 的简写, 是在 ADO 技术基础上发展起来的全新的面向对象模型, 比 ADO 更适应于分布式及 Internet 等大型应用程序环境, 是一种可以提供给 Web 页面开发者轻松存取 Internet 数据库, 并在用户端实现“网上实时更新”的强大 Web 数据库技术^[10]。ADO.NET 收集了所有的和数据存取有关的类, 这些类由一些具有典型数据库功能的数据容器对象组成, 这些对象具有一般

处理能力。ADO.NET 将 XML 和 ADO 对象模型集成在一个树状的类的集合中,使得把数据转换成 XML 以及通过网络传输的操作得到了简化和优化。ADO.NET 包含 System.Data 命名空间中的一组对象, System.Data 命名空间可以通过 .NET 数据提供者(provider)与数据库通信^[5]。

ADO.NET 对象允许我们连接到数据库,在数据库中检索数据、编辑数据、删除数据和插入数据,并在程序中处理数据。ADO.NET 由 Visual Studio.NET 系统提供,是 ADO 的增强版本,具有全部 ADO 的特点,同时又具有更多优点,特别是在可操作性、可编程性、可维护性、性能、可伸缩性等方面优点尤为突出。ADO.NET 既支持 C/S 结构程序开发模式,使得编程人员可以用窗体编程来访问网络数据库;又支持 B/S 结构,编程人员可以用 Windows 提供的 Internet Explore 或其他浏览器直接访问网络数据库。在 ADO.NET 内部包含了多个相互独立的组件和对象模型,如 DataSet(用于组件之间传输数据的一种分层对象模型),SqlConnection(主要完成 SQL 数据库的连接建立),SqlCommand(主要完成 SQL 数据库命令接口)。

ADO.NET 主要包含了 5 个组件,分别是: Connection、Command、DataAdapter、DataSet 和 DataReader。ADO.NET 程序设计模型是比较简单的由 DataSet 提供的高速缓存形式的数据,DataSet 对象可以说是由许多数据表、记录和字段组成的一个对象,主要用于内存中存放数据,可以一次读取整张数据表的内容。DataSet 对象也是一个集合对象,由许多 DataTable 对象组成,且没在与数据源联系的能力,其取得数据的方式主要有两种:一是通过读取 XML 文件取得;二是利用程序通过数据操作对象取得。

第三章 系统需求与可行性分析

3.1 系统的需求分析

利用计算机进行信息控制和数据处理，不仅可以提高工作效率，而且大大提高其安全性。要开发一个软件系统，首先要进行需求分析，为系统设计做好充分的准备，这样才能使得设计出的软件满足用户的各项功能需求。在调研过程中我们发现，目前市面上有许多管理信息系统软件，但并不是都能适合中小型企业事业单位对人事档案管理的要求，因此，我们着手开发面向中小型企业人力资源管理系统。为了充分利用计算机，使企业员工管理工作进一步规范化，满足员工管理及财务管理等方面的信息需求，使员工管理水平向标准化、现代化迈进一步，本研究在对国内现有的员工管理系统进行调研的基础上，重新设计开发方案，开发建立了一套完整的企业员工管理系统，来对员工基本信息、工资信息、出勤等进行添加、修改、删除、查询等日常管理工作，以简洁实用为基础，实现企业员工管理的计算机化，提高工作效率和信息化水平。

3.1.1 公司员工管理系统设计定位

公司员工管理系统的目的就是要采用当今最先进的信息技术手段，按照管理信息系统的管理和开发方法，支持并规范了员工信息的管理。公司员工管理系统应该具有添加、修改、删除、查询的全过程，并加上上述过程中对各种信息资源的管理和应用。提高各部门工作的工作效率，实现各部门信息的准确、及时处理和沟通及共享，为企业管理人员提供更有利的支持和反馈信息，实现将员工信息的存储做到最好的总体目标。

3.1.2 公司员工管理系统的预期设计

公司员工管理系统主要包括：员工基本档案管理和查询；月部门考勤、年度个人考勤、员工考勤信息查询；月工资核算、月工资发放、员工工资信息查询；用户信息和密码修改、用户权限分配、公司部门设置等等，可以分为以下几个主要的模块：

- 员工基本档案管理：档案管理员可以对所有员工的信息进行修改、增加和删除。
- 员工基本档案查询：档案管理员可以查询所有员工的基本信息。
- 月部门考勤：考勤管理员可以按照月份对各个部门的员工进行考勤管理，可以修改员工的考勤记录。
- 年度个人考勤：考勤管理员可以按照年份对所有员工进行考勤查询。
- 员工考勤信息查询：考勤管理员可以查询所有员工所有的考勤信息。
- 月工资核算：工资管理员可以查询所有员工每月的工资情况，然后修改每个员工对应的工资，还可以对员工发放工资。
- 月工资发放：工资管理员可以在这里查询到所有员工发放过的工资。
- 员工工资信息查询：工资管理员在这里可以查询到所有员工的工资信息。
- 用户信息和密码修改：登录的用户可以修改自身的个人信息和其登录的密码。
- 用户权限分配：系统管理员可以分配所有模块的查看权限，还可以新建、修改管理员的权限。
- 公司部门设置：系统管理员可以对部门进行修改、新增和删除。

3.2 系统开发的可行性分析

之所以要开发新系统，主要是因为对原有管理方式或系统的功能等方面不满意。原来的管理方式人工管理，原系统也可能是正在运行的信息系统。由于存在的问题可能充斥各个方面，内容较分散，这就要求系统分析人员针对用户提出的各种问题和初始要求，进行认真的识别、研究，提出是否可行的意见或报告。通过可行性分析，可以最后确定开发系统的必要性。可行性分析对系统的开发至关重要，可以大幅度的减少不必要的损失，保证系统开发顺利进行，从而也避免了盲目投资，减少不必要的损失。因此，要对系统进行技术可行性、经济可行性、管理可行性和开发环境可行性四方面的系统可行性分析，是十分必要的。

3.2.1 技术可行性分析

技术可行性分析是根据新系统的设计目标来考虑系统是否具有软硬件设备、计算机的联网能力、网络及数据安全保护设施、输入输出设备、大容量存储设备等等各方面的硬件条件。软件方面应重点考虑操作系统、数据库管理系统等

配置和功能的软件条件。在技术可行性分析时，还必须要充分考虑该公司具体从事该业务人员的文化素质、工作能力，经过短期培训后，是否能够承担使用和维护该管理系统的可能性。总体来看，近年来，计算机技术发展异常迅猛，高速度、大容量的电脑已经成为许多公司日常工作必不可少的设备，推动办公自动化的软件不断涌现，微机的普及为该系统的开发奠定了坚实的基础。

本系统是采用 B / S 架构的信息管理系统。系统采用流行 ASP. net 技术实现，同时采用 C# 作为开发语言，由于 SQL Server 2005 数据库是当前最流行的关系数据库系统之一，它对服务器的要求比较低，并包含了视图的灵活运用、触发器与存储全过程、用户自定义数据类型与自定义函数、维护数据的完整性等，有很强的实用性。因此拟订 SQL Server 2005 作为后台数据库服务器，用 Visual Studio 2005 作为开发工具^[6]。

3.2.2 经济可行性分析

经济可行性分析是指估算新的系统开发和运行所需要的费用，以及新系统能创造出的效益，并将投资和效益进行比较。在费用支出方面，不仅要考虑到主机的费用，而且要充分考虑到计算机外围设备的费用、软件开发费用、人员培训费用和将来系统投入运行后的管理维护费用和备件费用。经济效益应从以下两方面综合考虑：一方面是可以用钱衡量的效益，如加快流动资金周转，减少资金积压等；另一方面是难以用钱表示的，例如提供更多的更高质量的信息，提高取得信息的速度等等。

本研究经过较为详细、周密的调查研究，计算机硬件设备达到了相应的要求，采用功能强大的软件，借鉴成功的实例，及科学的开发方法，从而降低开发成本，达到经济高效的目的。一方面新系统的开发不需要额外增加设备购置费、软件开发费、管理和维护费，且人员培训过程简单不需要太多的费用；另一方面，新系统的开发可以较好地解决人工管理员工信息的繁琐，并能够及时更新员工的所有信息，从而提高经济效益，进一步减少成本投入、提高工作效率、提升公司的竞争力和影响力。

3.2.3 管理可行性

管理可行性就是要考虑当前系统的管理体制，是否给新系统提供所必需的各种数据，企业高层领导及各级管理人员对开发一个新系统来替代现有系统的需求

是否迫切，还要考虑当前系统的操作人员对新系统的适应能力等等。

3.2.4 开发环境可行性

企业领导层对开发新的员工管理系统的意见是否一致，有无相应的资金投入，能否抽调出骨干力量来参加新系统开发研制等。简单地说，就是企业能否为新的管理系统的开发建设提供一个比较长期的良好环境，这是可行性阶段必须要考虑的问题。计算机以其强大的数据处理能力作为人类脑力劳动的有力助手登上人类发展的历史舞台后，已经渗透到社会生活的各个领域，使企事业单位的信息处理能力越来越适应现在化管理的要求，并且，系统已经逐步从单项的事务信息处理系统迅速走向综合性的管理信息处理。本系统力求具有非常良好的用户界面，并能够做到容易进行操作。在当今的企业中，都拥有一批高学历、高素质的员工，只需要进行短期的系统培训，便可熟练操作和使用本管理系统。

3.3 系统用例模型与系统文档

软件研究开发人员在对用户进行需求分析的调研过程中，发现用户经常不能立即准确地描述出新的系统应该具备怎样的功能？因此，需要研发人员要充分理解和分析用户的需求，并将系统应该具有的功能通过用例图直观地描述出来，最终能够方便用户理解并根据示例做出评判，并根据用户的实际需要进行完善。研发人员可以根据用户的反馈意见不断调整用例模型，直至完全准确、充分地描述清楚系统功能。用例建模主要是分离出系统的活动参与者(Actor)和用例(Use Case)，用例是指对系统所提供功能的一种描述，而活动参与者是那些可能使用这些用例的人员或外部系统，通过用例图可以描述出系统外部的执行者、系统的用例，以及他们之间的联系。用例的执行必须对发起该用例的参与者或者其他参与者产生影响^[11]。

3.3.1. 公司员工管理系统的用例模型

本研究中采用如图 3-1 所示的公司员工管理系统的用例模型：

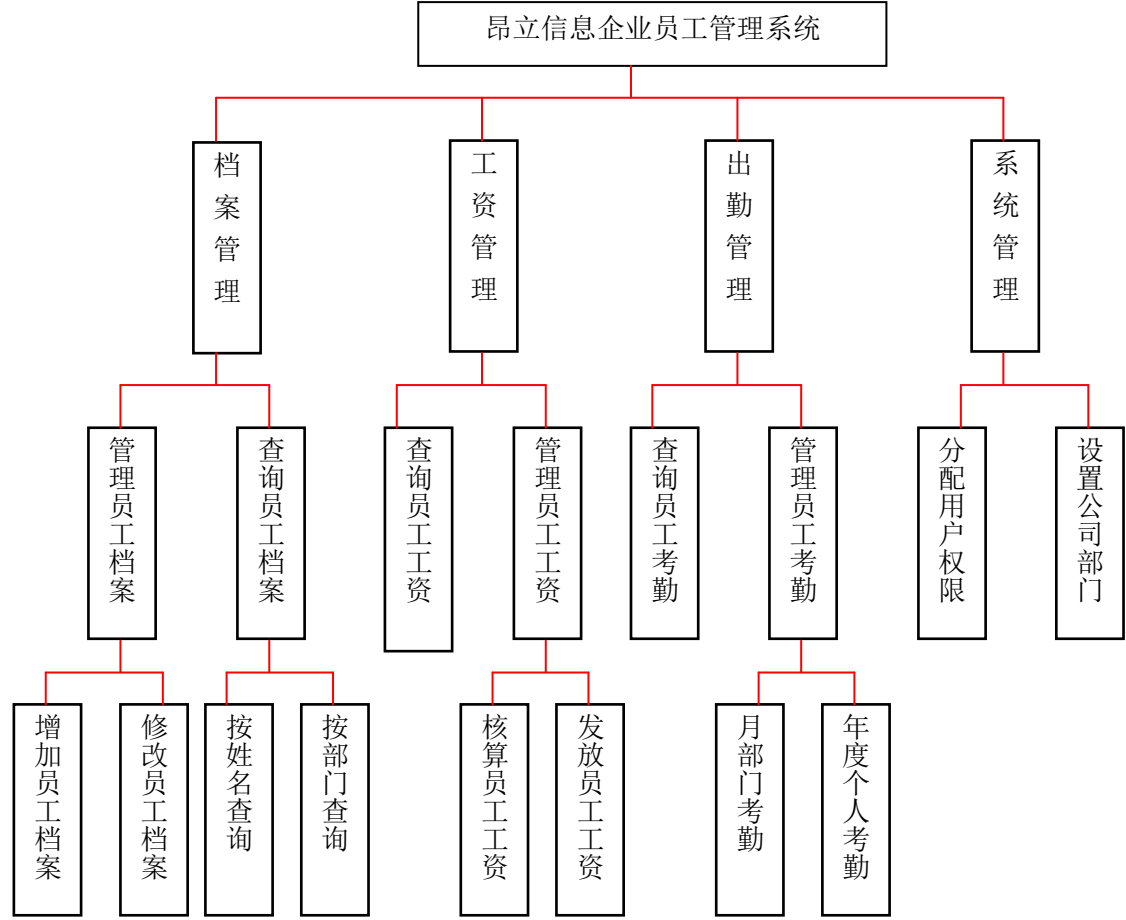


图 3-1 公司员工管理系统的用例模型

3.3.2 公司员工管理系统的用例文档

下面的表 3-1、3-2、3-3 分别列出这些用例文档。

表 3-1 档案管理员管理员工的档案用例

1	用例名称	档案管理员修改管理	员工档案
2	用例编号	UC1	
3	用例描述	档案管理员登录系统,对员工档案进行修改管理	
4	用例参与者	档案管理员	
5	前置条件	档案管理员进入档案管理界面	
6	后置条件	档案管理员对员工档案进行管理	
7	基本路径	(1)档案管理员登录到员工管理系统	
		(2)档案管理员进入员工档案管理界面	
		(3)档案管理员点击选择	
		(4)系统根据档案管理员选择的员工从数据库中找到员工的详细信息	
		(5)档案管理员可以修改员工的信息资料	
		(6)系统将改动后的数据保存在数据库中	
		(7)系统从新调用改后的数据显示	
8	扩展点	没有登录到员工管理系统	
9	补充说明	修改员工的内容包括员工编号、部门、姓名、出生日期等等	

第三章 系统需求与可行性分析

表 3-2 考勤管理员管理年度个人考勤用例

1	用例名称	考勤管理员修改管理	员工档案
2	用例编号	Uc2	
3	用例描述	考勤管理员要登录系统，对员工考勤进行管理	
4	用例参与者	考勤管理员	
5	前置条件	考勤管理员进入考勤管理界面	
6	后置条件	考勤管理员对员工考勤进行管理	
7	基本路径	(1) 考勤管理员要登录到员工管理系统	
		(2) 考勤管理员选择年度个人考勤管理	
		(3) 考勤管理员选择所在部门、员工姓名、年份	
		(4) 考勤管理员点击查询	
		(5) 系统将选择的数据从数据库找到所要查找的员工的年度个人考勤	
8	扩展点	没有登录到员工管理系统	
9	补充说明	年度个人考勤包括：考勤年份、考勤月份、考勤记录	

表 3-3 工资管理员管理月工资查询用例

1	用例名称	工资管理员	员工档案
2	用例编号	Uc2	
3	用例描述	工资管理员登录系统，对员工的工资进行管理	
4	用例参与者	工资管理员	
5	前置条件	工资管理员点击进入出勤管理界面	
6	后置条件	工资管理员对员工工资进行管理	
7	基本路径	(1) 工资管理员要登录到员工管理系统	
		(2) 工资管理员要选择员工工资查询界面	
		(3) 工资管理员选择年月，点击查询	
		(4) 系统将已经选择的员工数据从数据库中找到所要查询的员工数据	
		(5) 系统将选择的数据从数据库找到所要查找的员工的年度个人考勤	
8	扩展点	没有登录到员工管理系统	
9	补充说明	管理用户权限的方式：新增新用户、修改用户权限、删除用户权限	

3.4 用例的业务过程

1、档案管理员管理员工档案的业务过程，用例活动图表示如下：

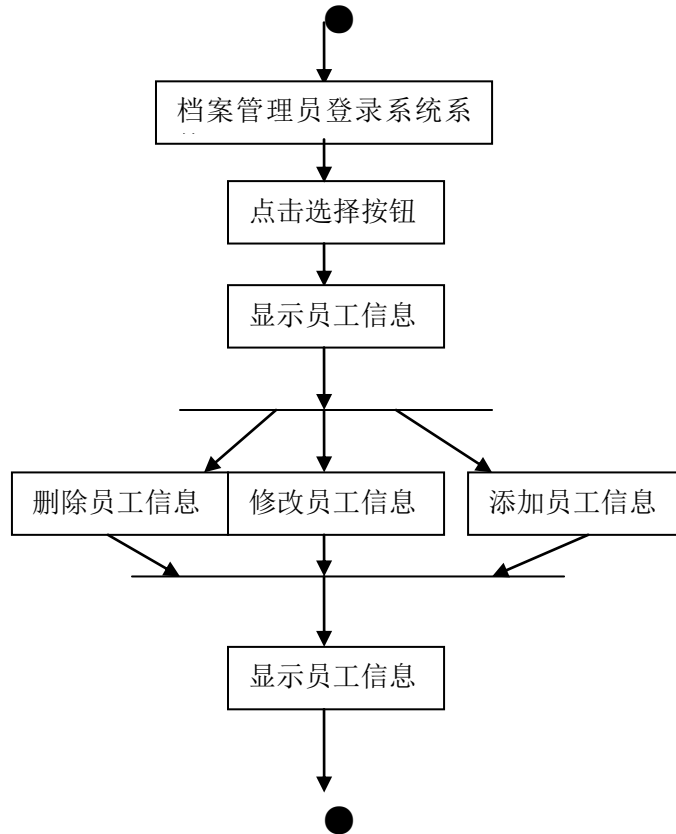


图 3-2 档案管理员管理员工档案活动图

2、工资管理员核算员工工资和发放工资的业务过程，用活动图表示如图 3-3 所示：

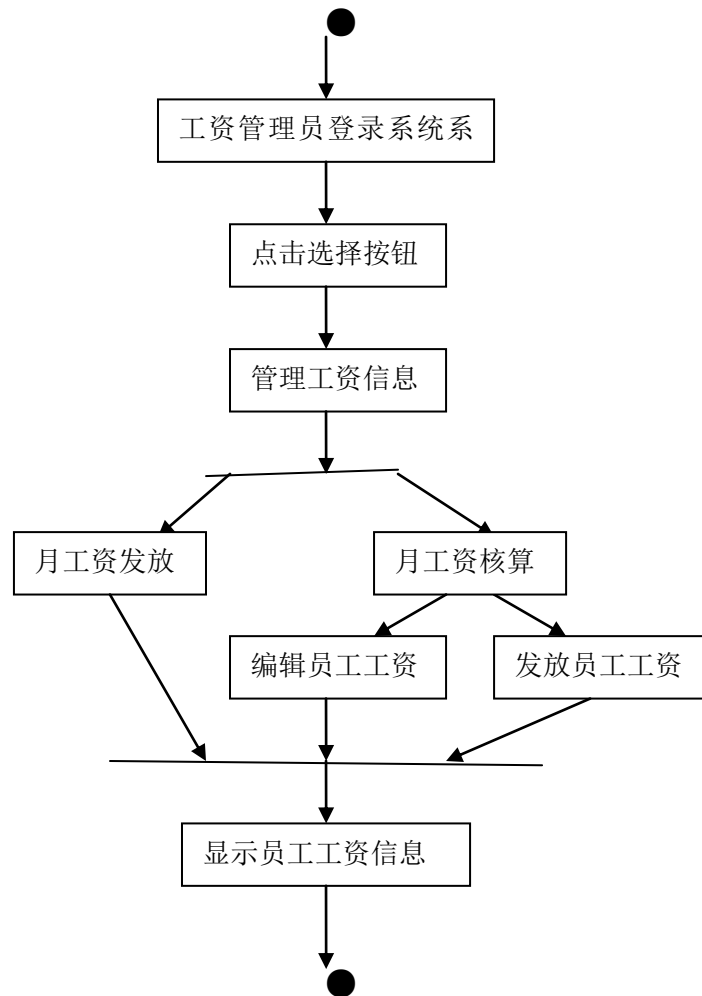


图 3-3 工资管理员核算员工工资用例活动图

第四章 员工管理系统的总体设计

4.1 系统设计架构

系统设计的目的是产生一个可用的、完整的解决方案，并且能够比较容易地将方案转换成程序代码。本系统采用典型的 3 层结构模式，三层结构是一种成熟、简单并得到普遍应用的程序架构，分别为 Web 显示层、数据访问层和数据库层，系统的总体设计架构如图 4-1 所示：

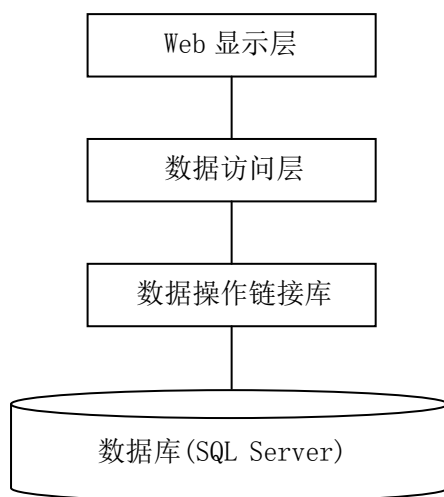


图 4-1 系统总体设计架构设计图

Web 显示层即为 ASP.NET 页面 (Web Pages) 层，为用户提供对应用程序的访问，该层通过数据访问层访问数据库。

数据访问层为 Web 显示层提供数据服务，一般封装操作数据库的选择、添加、修改、删除和查询等操作，同时还为 Web 显示层提供访问数据库的接口或方法等。

数据库操作动态链接库 (DLL)，该层直接访问系统的数据库^[7]。

该系统的设计严格遵循了系统设计的几个原则，即：灵活性、简洁性、系统性、可靠性的原则。

4.2 系统功能模块及组成

由于系统主要是完成整个企业员工管理系统，所以系统功能模块主要是由两大部分组成的：一是系统管理模块，二是考勤管理模块。在了解了系统的需求分析和系统功能以后，可以把系统划分为如下模块：系统管理模块、考勤管理模块。下面，就将对各模块的功能分别进行详细的介绍。

4.2.1 系统管理模块

在系统管理模块中，可进行的流程是：用户注册→用户登录→修改用户信息→用户退出系统。用户进入系统前，必须要先经过登录界面进行身份验证，用户输入用户名和密码后进行验证，若验证不能通过，系统会自动提示错误信息；当验证成功，就可以进入系统主界面。用户登录系统后用户的基本信息，如：姓名、所在部门及岗位等将写入 Session 对象，当用户退出系统时，将清除 Session 对象并返回到首页。此功能模块的组成描述如图 4-2 所示：

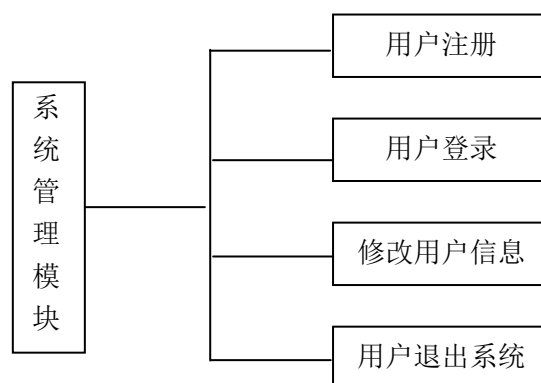


图 4-2 系统管理模块示意图

4.2.2 考勤管理模块

系统将记录包括请假、加班的日期等数据的考勤记录，记录每个职工的出勤统计情况，系统还可以进行考勤历史记录查询。在考勤管理模块中，主要包括：员工加班申请，对所有员工的加班历史记录进行查询；员工请假申请，对所有员工的请假历史记录进行查询，还有考勤记录单可以记录所有员工的加班和休假情况，以及可以根据考勤记录单查询员工的出勤情况。此功能模块的组成描述如图 4-3 所示：

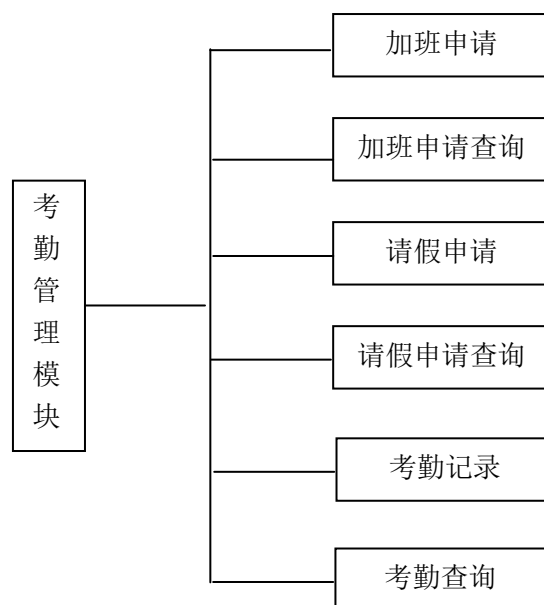


图 4-3 考勤管理模块功能示意图

4.3 数据库设计

计算机作为现代化的数据计算和信息处理工具具有精准高速的运算功能和强大的数据处理能力，被广泛应用于各个行业。数据库技术是信息时代计算机发展和信息处理的必然产物。数据库设计是系统开发中重要步骤之一。其设计的优劣直接决定了数据结构的优劣，最终影响到系统的功能。由于员工管理涉及大量信息的处理和各种操作，所以能否建立一个良好的数据组织关系与数据库，使整个系统的数据之间可以共享所拥有的信息资源，可迅速、方便、准确地调用和管理所需数据，是衡量系统开发工作好坏的重要指标，也是系统的基本要求。

4.3.1 数据库设计的原则

在设计数据库系统时，应该重点考虑以下几个因素^[12]：

1. 数据库必须要做到层次分明，布局合理；
2. 数据库必须要高度的结构化；
3. 必须要注意冗余度，一方面要尽可能地减小冗余度，以缩小存储空间的占用；另一方面，还要考虑要有适当的冗余，以提高运行速度和降低数据库的开发难度；

4. 必须要保证数据的正确性和一致性。
5. 必须要考虑设置相应的安全机制。

4.3.2 数据库设计的步骤

数据库的设计包括两个部分：一是数据库的逻辑设计，内容包括对应于概念级的概念模式，即系统要处理的数据库全局逻辑结构，同时包括对应于用户级的外模式；二是数据库的物理设计，这是在逻辑结构已经确定的前提下设计数据库的存储结构（即对应于物理级的内模式）。从软件的生命周期来看，这两个部分的设计过程又可分为六个步骤，如图 4-4 所示。设计一个完善的数据库应用系统是不可能一步到位的，它往往是这六个步骤不断反复、逐步完善的一个过程^[13]。

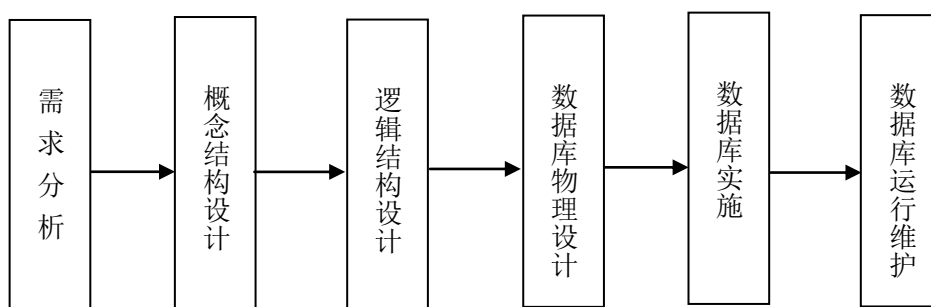


图 4-4 数据库的设计步骤

管理信息系统所涉及的数据库设计的六个步骤分别是：数据库需求分析、概念结构设计、逻辑结构设计、物理设计、数据库实施和数据库运行维护。

(1) 数据库需求分析

将业务管理单证流转化为数据流、划分主题之间的边界、并完成相应的数据字典，是数据库需求分析的任务。数据库需求分析的主要工作有以下几项工作：绘制数据流程图、数据分析、功能分析、确定功能处理模式和数据之间的关系。需求分析时要重点了解数据库中需要存储哪些数据，用户要完成什么处理功能，特别是要注意把握数据的安全性和完整性。需求分析阶段特别需要注意两点：①要充分考虑到可能的扩充和改变，使设计能够易于更改；②必须强调用户的参与。

(2) 概念结构设计

概念结构设计是整个数据库设计的关键所在。它通过对用户需求进行综合、归纳与抽象，最后形成一个独立于具体数据管理系统和硬件结构的概念模型。数

第四章 员工管理系统的总体设计

数据库概念结构设计方法主要有两种：一是集中式模式设计法，二是视图集成设计法。集中模式设计法是根据需求统一设计一个综合的全局模式，适用于小型或并不复杂的单位或者部门。视图集成设计法是将整体分解成若干个部分，对每个部分进行局部设计，建立各个部分的视图，然后以各视图为基础再进行集成设计。而对于比较大的系统一般采用 E-R 模型与视图集成法进行设计。设计步骤是：首先选择局部应用；然后进行局部视图设计；最后对局部视图进行集成得到概念模式。

（3）逻辑结构设计

逻辑设计的任务主要是从 E-R 图与对应的纲要表出发，把概念结构转换为与选用的数据库管理系统所支持的数据模型，从而确定各个实体及关系的表名属性。

（4）物理设计、数据库实施和数据库运行维护

物理设计的任务主要是确定所有属性的数据类型、宽度以及取值范围，设计出基本数据表的主键，实现物理数据库，完成数据库的物理设计字典。对数据库的物理设计初步完成后就可以开始建立数据库。在数据库实现阶段的主要任务是，设计人员要利用数据库管理系统提供的数据库定义语言和其他实用程序，将逻辑设计阶段和物理设计阶段的成果能够严格准确地描述出来，收集、存储数据库管理系统实际运行的数据。数据库的实施和数据库的运行维护统称为加载测试，这项工作贯穿于程序测试工作的全过程。在数据运行和维护阶段，可能需要对数据库结构进行修改或扩充，并不断地进行评价、调整直至最后完成设计。总之，对整个数据的录入、修改、查询、处理工作和后期的维护等工作都可以视为对数据库的加载测试。

4.3.3 数据库逻辑结构设计

4.3.3.1 体系结构

系统的解决方案是体系结构的一种表现方式，解决方案中包括界面设计和功能的代码以及与数据库的连接。

本系统的解决方案包括以下几点：

1. Check Manage 文件夹包括月部门考勤信息和年个人考勤信息。
2. File Mallage 文件夹包括员工档案信息。
3. Images 文件夹包括系统所需要的各种图片。
4. Query Manage 文件夹包括工资查询、员工档案查询、考勤查询等信息。

5. Salary Manage 文件夹包括工资核算以及工资发放等信息。

6. System Manage 文件夹包括登录信息、错误信息、权限设置、注销等信息。

4.3.3.2 设计表结构

用户表即 user，主要用来记录注册后员工的基本信息，包括：员工编号、用户名、密码、员工姓名、性别、所在部门、岗位及注册日期等信息，其中员工编号为主键，用户表的表结构如表 4-1 所示。

表 4-1 用户 user 表结构

字段名称	数据类型	长度	允许为空	是否为主键	说明
EmployeeID	Nvarchar	50	否	是	员工编号
Uesname	Nvarchar	20	否	否	登录系统用户名
Password	Nvarchar	50	否	否	登录系统的密码
EmployeeName	Nvarchar	20	否	否	员工姓名
Department	Nvarchar	20	否	否	员工所在部门
Post	Nvarchar	10	否	否	岗位
Date	Datetime	8	否	否	注册日期

4.3.3.3 后台数据库的设计

(1) 后台数据库配置

ASP.NET 提供了一个丰富而可行的配置系统，以帮助管理人员轻松快捷地建立自己的 Web 应用环境。ASP.NET 提供的是一个层次配置架构，可以帮助 Web 应用、站点、机器分别配置自己的扩展配置数据。ASP.NET 的配置系统具有以下优点：

ASP.NET 允许配置内容可以和静态内容、动态页面和商业对象放在同一应用的目录结构下。当管理人员需要安装新的 ASP.NET 应用时，只需要将应用目录拷贝到新的机器上即可。

ASP.NET 的配置内容以纯文本方式保存，可以以任意标准的文本编辑器、XML35 解析器和脚本语言解释、修改配置内容。

ASP.NET 提供了扩展配置内容的架构，以支持第三方开发者配置自己的内容。

ASP.NET 的配置文件的修改被系统自动监控，无须管理人员手工干预。为了保持系统良好的可移植性，系统采用对数据库配置进行统一管理的办法，将程序中用到的所有连接字符串信息统一放于 Web.Config 配置文件中，在程序中对该文档的这一配置进行调用，实现统一更改统一使用。对于数据库调用字符串，由于是数据库使用的是本地数据库，Data Source（数据源）设置为 localhost；UID（用

户 ID) 设置为默认的 sa, PWD(连接密码) 根据安装数据库时的密码设置为 qjqboy; DATABASE(数据库名称) 设置为 EHR^[8]。

(2) 后台 Web.Config 的配置文档

ASP.NET 的配置文件是基于 XML 格式的纯文本文件存在于应用的各个目录下, 统一命名为 “web. ocnfig”。它决定了所在目录及其子目录的配置信息, 并且子目录下的配置信息覆盖其根目录的配置。进入系统的用户必须通过身份验证, 而实现这验证功能我采用了 we. bconfig 文件。该文件的具体配置如下:

```
<?xml version="1.0"?>
<configuration>
<!--自定义用户数据库连接字段-->
<appSettings>
<add key="connectionString"
value="server=. ;database=EHR;uid=sa;pwd=qjqboy">
</add>
</appSettings>
<system.web>
<compilation
debug="true"
defaultlanguage="C#"
/><assemblies>
<add assembly="System.Windows.Forms, Version=2.0.0.0,
Culture=neutral, PublicKeyToken=B77A5C561934E089"/>
<add assembly="System.Design, Version=2.0.0.0, Culture=neutral,
PublicKeyToken=B03F5F7F11D50A3A"/>
</assemblies>
</compilation>
<authentication mode="Windows"/>
</system.web>
</configuration>
```

(3) 调用 Config.web 文件的代码

在程序中读取 Config.web 配置文件中的数据库设置信息, 只需使用如下语即可, 语句列举如下:

Stringstrconn=ConfigurationSettings.AppSettings["connectionString"]];我们只要在程序文件中加入该行代码,就可以轻松的将数据库连接字段读取到字符串 strconn 中,并且每个配置段处理器只需声明一次即可^{[9][10]}。

4.3.3.4 数据表的设计

一个成功的信息管理系统,是建立在许多先进的技术优势之上的,而数据库是其中一个重要的条件和关键技术。要设计出一个优质的信息管理系统数据库,除满足系统所要求的功能外,还必须遵守下列原则:

(1) 不应针对整个系统进行数据库设计,而应该根据系统架构中的组件划分,针对每个组件所处理的业务进行组件单元的数据库设计;不同组件间所对应的数据库表之间的关联应尽可能的减少,尽量不要创建外键关联,即使不同组件间的表需要外键关联,而只需要记录关联表的一个主键,确保组件对应的表之间的独立性,为系统或表结构的重构提供可能性。

(2) 主键的个数越少越好。主键是表与表之间连接的工具,为了能够使数据具有更好的独立性,途径就是减少主键,减少表与表之间的连接。这样数据的独立性就越好。采用领域模型驱动的方式和自顶向下的思路进行数据库设计,首先分析系统业务,根据职责定义对象。对象要符合封装的特性,确保与职责相关的数据项能够定义在一个对象之内,这些数据项就能够完整描述该职责,不会出现职责描述缺失。应该确保一个对象有且只有一项职责,如果一个对象要负责两个或两个以上的职责,应进行分拆。

(3) 字段的个数越少越好。

(4) 所有的基本表的设计均应尽量符合第三范式^[11]。

根据建立的领域模型进行数据库表的映射,此时应参考数据库设计第二范式:一个表中的所有非关键字属性都依赖于整个关键字。关键字可以是一个属性,也可以是多个属性的集合,不论那种方式,都应确保关键字能够保证唯一性。在确定关键字时,应保证关键字不会参与业务且不会出现更新异常,这时,最优解决方案为采用一个自增数值型属性或一个随机字符串作为表的关键字。

在员工管理系统中,涉及的表有:操作用户表、工资发放表、公司部门表、员工档案表、考勤记录表。

第四章 员工管理系统的总体设计

表 4-2 公司部门表

字段	类型	长度	主键	非空
部门名称	Varchar	50	是	是
部门负责人	Varchar	50		否
人事文员	Varchar	50		否
联系电话	Varchar	50		否
额定人数	Int	50		否
补充说明	Varchar			否

表 4-3 公司发放表

字段	类型	长度	主键	非空
自编号	int	50	是	是
员工编号	Varchar	50		是
员工姓名	Float	50		是
基本工资	Float	50		否
津贴	Float	50		否
奖金	Float	50		否
工龄工资	Float	50		否
岗位工资	Float			否
其他应增项	Float			否
应发合计	Float			否
所得税	Float			否
养老保险	Float			否
医疗保险	Float			否
住房公积金	Float			否
其他应减项	Float			否
应减合计	Float			否
实发金额	Float			否
银行账号	Varchar			否
养老保险号	Varchar			否
医疗保险号	Varchar			否
住房公积金号	Varchar			否
身份证号	Varchar			否
发放年份	Int			否
发放月份	Int			否
补充说明	Varchar			否

表 4-4 操作用户表

字段	类型	长度	主键	非空
用户名称	Varchar	50	是	是
用户姓名	Varchar	50		否
所在部门	Varchar	50		否
联系电话	Varchar	50		否
职责说明	Varchar	50		否
人事管理之员工基本档案				否
考勤管理之月度部门考勤				否
考勤管理之年度个人考勤				否
工资管理之月度工资核算				否
工资管理之月度工资发放				否
查询管理之查询个人档案				否
查询管理之查询部门档案				否
查询管理之查询考勤信息				否
查询管理之查询工资信息				否
系统管理之分配用户权限				否

表 4-5 考勤记录表

字段	类型	长度	主键	非空
自编号	int	是	是	
员工编号	varchar	50		
员工姓名	varchar	50		
C1	varchar	50		
C2	varchar	50		
.....	varchar	50		
C30	varchar	50		
C31	varchar	50		
考勤年份	int	否		
考勤月份	int	否		

第四章 员工管理系统的总体设计

表 4-6 员工档案表

字段	类型	长度	主键	非空
自编号	int	是	是	
员工编号	varchar	50	是	
员工姓名	varchar	50	是	
所属部门	varchar	50	否	
性别	varchar	50	否	
民族	varchar	50	否	
籍贯	varchar	50	否	
出生日期	smalldatetime	否	否	
身份证号码	varchar	50	否	
政治面貌	varchar	50	否	
婚姻状况	varchar	50	否	
文化程度	varchar	50	否	
职称	varchar	50	否	
专业	varchar	50	否	
毕业院校	varchar	50	否	
基本工资	float	否	否	
联系电话	varchar	50	否	
养老保险号	varchar	50	否	
医疗保险号	varchar	50	否	
住房公积金号	varchar	50	否	
手机号码	varchar	50	否	
家庭住址	varchar	50		
补充说明	varchar	50		

第五章 系统的详细设计与实现

5.1 系统描述

5.1.1 注册登录

注册：用户若要登录本系统，则必须先注册。没有注册的用户在首页单击“注册”按钮则会进入注册页面。在注册页面填写完注册信息，然后单击“提交”按钮，系统提示用户注册成功后会自动返回首页，此时用户可以根据刚才所填写的用户注册信息（用户名和密码等）登录系统。登录：用户根据自己注册过的信息（用户名和密码等），在首页中输入用户名和密码进行登录，登录时系统会到数据库中去查找是否存在用户名和密码匹配的记录，若存在则显示登录成功，否则显示提示信息：用户名或密码不正确，登录失败，注册流程图如图 5-1 所示。

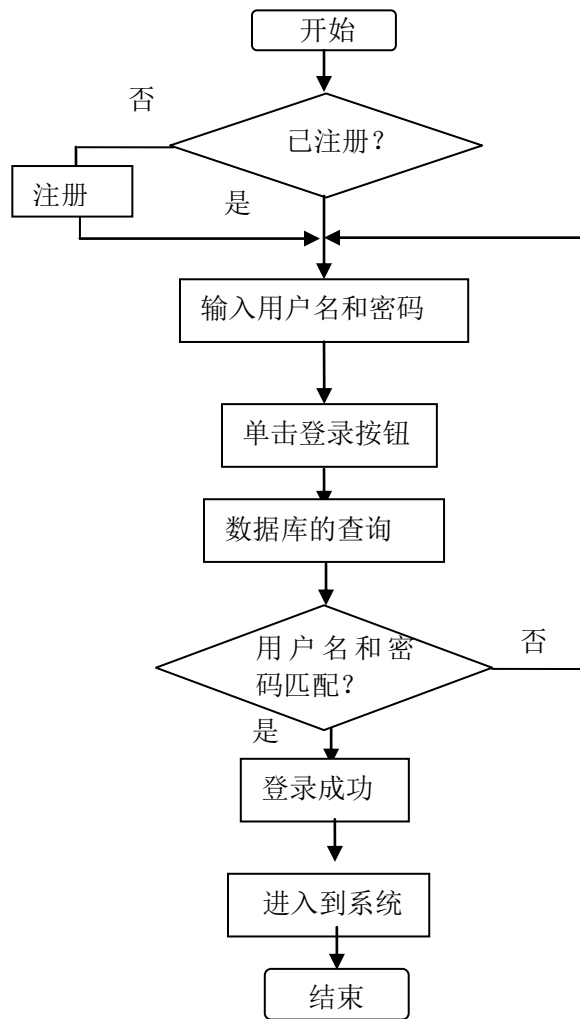


图 5-1 注册流程图

5.1.2 考勤系统

当用户登录成功后，就可以进入到整个系统。系统分为了三个大部分：加班管理、考勤管理及信息管理。这三个功能模块及各模块的子功能是通过“树”型结构组织起来的，而且“树”结构的每个节点都是超链接，单击这些超链接可以转到相应的页面。用户可以根据需要通过单击“树”结构的节点进入任意功能模块的任意子功能，来完成自己需要做的操作。系统示意图如图 5-2 所示：

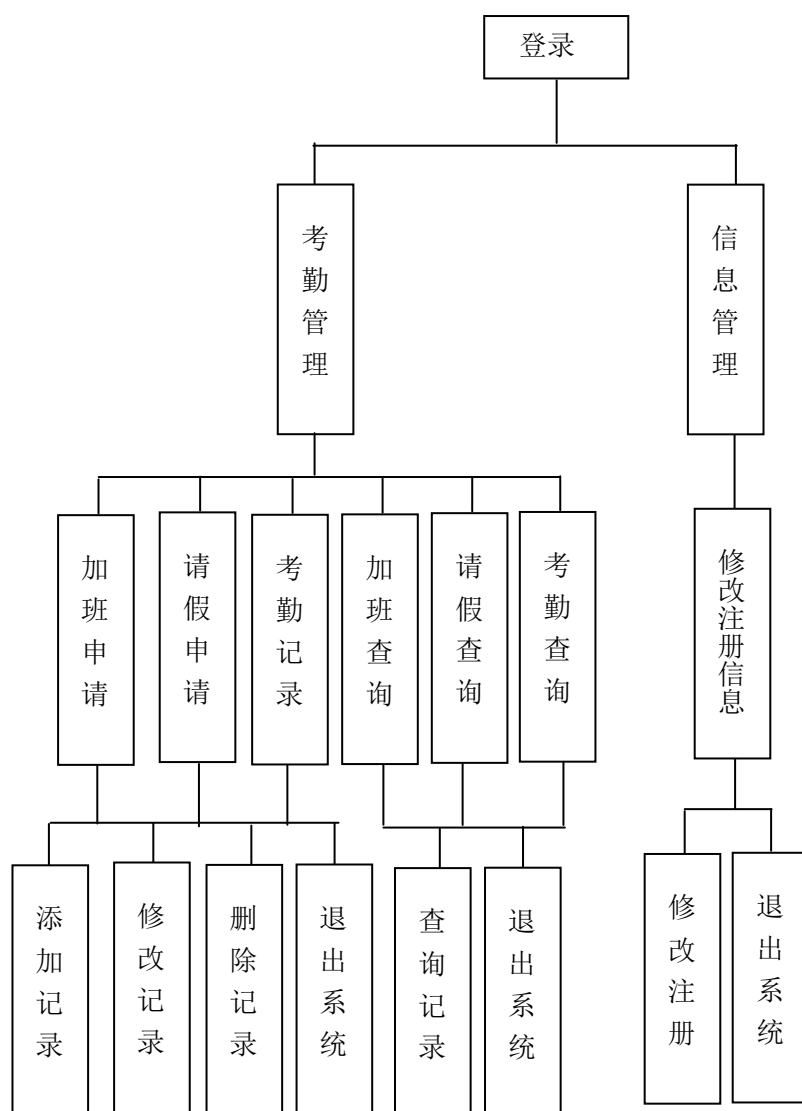


图 5-2 系统示意图

5.2 开发工具的选择

如果说硬件是一个系统的身体，那么软件就是个系统的灵魂，一个系统设计的好坏，不仅要有正确的软件开发思想，也应该有良好的软件工具，员工管理系统模块所采用的软件配置环境：

(1)操作系统：Microsoft Windows XP Professional

(2)编程环境：Microsoft Visual Studio. NET 2005

(3)数据库服务器：SQL server 2005

后台数据库配置中，本模块系统使用的是 SQL Server2005 数据库，用字符串


```
name="PersonnelDBConnectionString"  
onnectionString="server=NEUSOFT-2C9E0DB SQLEXPRESS; uid=sd; pwd=123456;  
database=MyPersonnelDB: "链接数据库，用 SqlConnection 和 SqlCommand 执行  
对数据库的增加、删除、修改的操作[12][13]。
```

5.3 页面设计和关键代码

5.3.1 系统页面设计

系统的界面设计是十分重要的，因为系统使用者的操作水平多层次化，所以整个系统在设计时应该注意：易导航、易操作性、容错性和友好的交互页面。首先在页面的风格上应保持一致，对所有界面的字体、字号、颜色以及超连接的样式作统一定义。界面的设计应该遵循快速、简洁、信息概括能力强、易于导航的原则，因此在设计该系统首页的时候主要考虑了以下几个问题：

(1) 内容匹配明晰，主题突出易寻：当用户登录到系统首页时，能很快知道系统包括哪些主要业务和主要功能，并能立刻找到自己所要操作的部分。

(2) 运行的速度要尽量快：尽量减轻系统运行的负担，缩短系统运行的时间，提高系统运行的效率，如页面上尽量不要放置图片等，否则就会延长系统运行的时间^[14]。

5.3.1.1 系统页面如图 5-3 所示：



图 5-3 系统首页

5.3.1.2 企业员工档案基本信息录入界面

界面如图 5-4 所示：

系统首页 人事管理 考勤管理 工资管理 查询管理 系统管理

请输入查询员工姓名:

员工编号: 所属部门: 姓名: 性别: 民族:

籍贯: 出生日期: 身份证号码:

政治面貌: 婚姻状况: 文化程度: 职称: 职务:

专业: 毕业院校: 进入公司时间:

合同服务年限: 基本工资: 银行帐号:

养老保险帐号: 医疗保险帐号: 住房基金帐号:

联系电话: 手机号码: 电子信箱: 自编号:

家庭住址: 邮政编码: 补充说明:

编号	员工编号	所属部门	姓名	性别	民族
1	A000001	财务部	张三	男	汉
2	A000002	财务部	吴多	男	土家
3	A000003	财务部	刘爽	女	汉族
4	A000004	财务部	王维	男	回族

图 5-4 员工档案基本信息

第五章 系统的详细设计与实现

其中，系统的关键代码及编程实现，请参看附录 1。

本章对本系统中所完成的信息管理和考勤管理的系统结构进行了设计与分析，完成对系统的详细描述，并设计了系统的主要界面，实现了企业员工管理信息系统的主要功能，并给出了这些功能模块的源代码。

第六章 系统测试

系统在开发设计的过程中，人为造成的错误或疏忽，很可能对整个系统的成功运行留下错误的根源。有时这些错误在开发过程中没有表现出来，但是在实际应用时，偶然的因素很可能将系统潜在的错误表现出来^[15]。因此，系统在投入运行之前，必须对其需求分析、编码、功能等进行最终复核，即系统测试。系统测试是软件生命周期中一项重要且复杂的环节，是软件质量保证的一个必不可少的关键步骤，通过系统测试可以极大地提高系统的可靠性，对软件可靠性保证具有非常重要的意义。

6.1 系统测试基本原则

系统测试应该遵循以下基本原则：

1. 定义好测试用例及“输入数据”、“预期的输出结果”。在执行程序之前要明确描述程序预期的输出结果，以便与程序测试后的输出结果对照。
- 2、系统测试必须贯穿整个系统开发过程。由于在系统开发的各阶段都可能会出现各种各样的问题，导致系统程序存在 Bug，因此，必须在开发的各阶段对系统进行测试，及时纠正错误，保证其功能达到用户预期要求。
- 3、系统测试时运行程序的所有部分，找出潜在的所有错误。测试通过也不能证明系统绝对没有问题，在系统的运行阶段还可能发现错误。
4. 既要选用合理的输入数据又要选择不合理甚至错误的输入数据进行测试。为了提高程序的可靠性，需要认真组织一些异常数据进行测试，仔细分析系统反应。
5. 既要检查程序是否做了必须做的工作，又要检查程序是否做了不需做的工作。

6.2 测试方法

根据软件工程的理论，按照测试时是否涉及程序的内部结构，测试方法可以分为白盒测试和黑盒测试两种方法。

白盒测试方法是把程序看成装在一个透明的白盒子里，也就是完全了解程序的结构和处理过程。这种方法按照程序内部的逻辑测试程序，检验程序中的每条通路是否都能按预定要求正常工作。白盒测试又称为结构测试。白盒测试的主要方法有逻辑覆盖、基本路径测试等^[16]。逻辑覆盖测试指以程序内部的逻辑结构为基础的测试用例设计技术。基本路径测试指对每一条独立执行的路径进行测试。

黑盒测试，黑盒测试又称为功能测试^[17-19]，是把程序看成一个黑盒子，不考虑程序的内部结构和处理过程。也就是说，黑盒测试是在程序接口进行的测试，它只检查程序功能是否能按照规格说明正常使用，程序是否能适当地接受输入数据产生正确的输出信息，并且保持外部信息的完整性。黑盒测试时，测试用例可以从等价类、边界值和错误预测三个方面进行考虑。等价类划分法是将输入数据的可能值分成若干个等价类，然后从每个等价类中选取数据作为测试用例。边界值分析法主要考虑的是各种输入、输出数据范围的边界值情况。错误预测法是测试人员依靠经验或直觉推测程序中可能存在的错误，从而编写检查这些错误的例子。在实际测试中，对功能模块测试、集成测试、确认测试时常采用黑盒测试法。如果发现错误，再采用白盒测试法进行调试。

黑盒测试，黑盒测试又称为功能测试^[17-19]，是把程序看成一个黑盒子，不考虑程序的内部结构和处理过程。也就是说，黑盒测试是在程序接口进行的测试，它只检查程序功能是否能按照规格说明正常使用，程序是否能适当地接受输入数据产生正确的输出信息，并且保持外部信息的完整性。黑盒测试时，测试用例可以从等价类、边界值和错误预测三个方面进行考虑。等价类划分法是将输入数据的可能值分成若干个等价类，然后从每个等价类中选取数据作为测试用例。边界值分析法主要考虑的是各种输入、输出数据范围的边界值情况。错误预测法是测试人员依靠经验或直觉推测程序中可能存在的错误，从而编写检查这些错误的例子。在实际测试中，对功能模块测试、集成测试、确认测试时常采用黑盒测试法。如果发现错误，再采用白盒测试法进行调试。

本系统主要采用黑盒测试的方法，对整个系统及其各功能模块进行严格的测试。在测试过程中，分别以系统管理员、档案管理员、工资管理员和考勤管理员的身份进入系统，首先测试相应用户的权限是否正确，是否只能进行相应权限的模块操作，测试各模块中业务流程是否正常流转。通过测试，尽可能的发现功能的错误和遗漏、界面错误、初始化错误等，进行及时的修改。

6.3 系统测试过程

系统测试是一个“测试—修改—调试—再测试”的不间断过程，只有这样才能使系统逐步趋于完善^[20]。本系统开发过程中根据软件工程中软件测试内容的各项进行了如下测试。

6.3.1 数据处理测试

主要包括数据导入导出测试、数据有效性测试、数据计算结构测试三个部分。

数据导入导出测试。各类数据导入导出后的一致性、各种数据表中的数据关系是否正确、能否识别输入的非法数据、汇总表及汇总查询结果是否正确、数据

打印是否正常。

数据有效性测试是指界面表单中的下拉框、单选框、复选框、数据输入框等数据输入单元的测试。在数据输入单元中输入正、负、零、小数进行测试，并检查是否能输入非正常符号等。

数据计算结果测试。对各类自动计算结果、汇总结果等功能进行手工计算结果来验证。测试同一数据在不同界面中是否保持一致，系统中数据的小数位数保留是否一致等。

6.3.2 按用户权限测试

(1) 系统管理员用户测试。系统管理员是系统中权限最高且最重要的用户，该用户的测试在系统测试中占有举足轻重的作用。以系统管理员的身份登录后，可以分配公司用户权限、设置公司部门等信息，并维护以上各类信息。系统管理员可对整个系统用户的信息、数据的安全性进行管理。本系统的测试中采用按模块进行逐一测试，出现问题及时修改代码，不断修改、不断完善直至满足设计要求。

(2) 档案管理员用户测试。档案管理员用户登录到系统后，可进行个人信息的维护，增加员工档案、修改员工档案、按姓名查询员工档案、按部门查询员工档案等功能的测试。

(3) 工资管理员用户测试。工资管理员用户登录到系统后，可进行个人信息的维护，查询员工工资、核算员工工资、发放员工工资等功能的测试。

(4) 考勤管理员用户测试。考勤管理员登录到系统后，可进行个人信息的维护，查询员工考勤、月部门考勤、年度个人考勤等功能的测试。

6.3.3 系统功能及内容测试

功能测试是测试系统各项功能的可用性，一般方法是从头到尾试用程序，查看程序的控制流程是否正确，各项功能是否可用；故意的错误操作，查看是否引起程序控制的混乱^[21-23]。内容测试用来检验 WEB 应用系统提供信息的正确性、准确性和相关性。信息的正确性是指信息是可靠的还是误传的；信息的准确性是指是否有语法或拼写错误。主要包括以下几个方面：

(1) 链接测试：

链接测试可分为三个方面进行。首先，测试所有链接是否按指示的那样确实链接到了该链接的页面；其次，测试所链接的页面是否存在；最后，保证系统上

没有孤立的页面。所谓孤立页面是指没有链接指向该页面，只有知道正确的 URL 地址才能访问。链接测试必须在集成测试阶段完成，也就是说，在整个系统的所有页面开发完成之后进行链接测试。

(2) 表单测试:

表单操作，就是当用户给系统的管理员在提交信息时进行的。例如用户注册、登录、信息提交等等。在这种情况下，我们必须测试提交操作是否完整，以校对、检验提交给服务器的信息是否正确。

(3) Cookies 测试:

当 WEB 应用系统使用了 Cookies 测试时，就必须检查 Cookies 是否能够正常的进行工作。测试的内容可包括 Cookies 是否起作用，是否按预定的时间进行保存，刷新对 Cookies 有什么影响等等^[24]。

(4) 设计语言测试

WEB 设计语言版本的差异可以引起客户端或服务器端严重的问题，例如使用哪种版本的 HTML 等。当在分布式环境中开发时，开发人员都不在一起，这个问题就显得尤为重要。

(5) 数据库测试

在使用了数据库的系统中，一般情况下，可能发生两种错误，分别是数据一致性错误和输出错误^[25]。数据一致性错误主要是由于用户提交的表单信息不正确而造成的，而输出错误主要是由于网络速度或程序设计问题等引起的，针对这两种情况，可分别进行测试。

以下仅列出测试过程中出现的三个错误:

1、数据库中的数据查询

操作：在 VS2005 中编译无误，系统运行不成功。

错误：系统异常退出。

提示：未处理的“System. Data. SqlClient. SqlException”类型的异常出现在 system. data. dll 中。

错误原因：说明数据库中没有相关表，或未引用 MySql. Data 文件。

2、未检查数据列表中是否选中数据记录的问题

操作：在 VS2005 中编译无误，系统运行，双击不包含数据的数据列表时。

错误：系统异常退出。

提示：Object reference not set to an instance of an Object

错误原因：在 Grid View 的双击事件中，一没有判断是否选中数据记录。

3、员工查询信息返回不完整

操作：在 VS2005 中编译无误，系统运行，查询信息。

错误：返回记录不完整，用户可浏览记录没有完全显示出来。

提示：无。

错误原因：SQL 语句构造不正确。

6.3.4 界面测试

测试系统操作界面图形文字布局是否一致，控件格式是否统一，控件是否对齐，相临控件之间的距离是否相同，是否存在错别字，文字表述是否合理；不同操作系统下字体显示是否正常；不同显示分辨率下界面显示是否正常。主要包括图形测试和整体界面测试两个部分。

1、图形测试

(1) 确保图形有明确的用途，图片或动画不要胡乱地堆放在一起，以免浪费传输时间。系统的图片尺寸要尽量地小，并且要能清楚地说明某件事情，一般都链接到某个具体的页面。

(2) 验证所有页面字体的风格是否一致。

(3) 背景颜色应该与字体颜色和前景颜色相搭配。

(4) 图片大小和质量也是一个很重要的因素，一般采用 JPG 或 GIF 压缩^[26]。

2、整体界面测试

整体界面是指整个系统的页面结构设计，是呈现给用户的一个整体感。对整体界面的测试过程，其实是一个对最终用户进行调查的过程。系统一般都是采取在主页上做一个调查问卷的形式，最终得到用户的反馈信息。对所有的可用性测试来说，都需要有外部人员(与系统开发没有联系或联系很少的人员)的参与，而且最好是最终用户的参与，这样可以提高测试的实用性能。

6.3.5 性能测试

(1) 连接速度测试：

用户连接到系统的速度根据用户上网方式的不同而变化，他们或许是电话拨号，或是宽带上网。当下载一个程序时，用户可以耐心的等待较长的时间，但是如果仅仅访问一个页面就不会有耐心的等待页面打开^[27-30]。如果系统响应时间太长(例如超过 5 秒钟)，用户就会因没有耐心等待而选择离开。另外，有些页面有超

时的限制，如果响应速度太慢，用户可能还没来得及浏览内容，就需要重新登录了。而且，连接速度太慢，还可能引起数据丢失，使用户得不到真实的页面。

（2）负载测试

负载测试主要是为了测量系统在某一负载级别上的性能，以此来保证系统在用户需求的范围内能正常、有效地工作。负载级别可以是某个时刻同时访问系统的用户数量，也可以是在线数据处理的数量。负载测试应该安排在系统发布以后，在实际的网络环境中进行测试。因为一个企业内部的员工，特别是项目组人员数量总是有限的，而一个系统能同时处理的请求数量将远远超出这个限度。所以，只有放在 Internet 上，接受负载测试，其结果才是正确可信的^[31-32]。

（3）压力测试

压力测试主要是测试系统的限制和故障恢复能力，也就是测试系统会不会崩溃，在什么情况下会崩溃。因为网络黑客常常通过提供错误的数据负载，直到导致系统崩溃，继而当系统重新启动时获得存取权。压力测试的区域范围包括表单、登录和其它信息传输页面等等。

6.4 测试的结果

通过对系统的各模块进行了详细全面的测试，得出以下测试结果：

系统已基本实现用户的需求，满足用户的设计要求，功能完善，操作界面友好，不同权限用户进入系统后，只能对自己权限内的工作进行操作，无关的工作也不出现在界面中。输入不合理甚至错误的数据，或是在不能为空的数据项没有输入数据，系统均会进行错误提示，并且能正确提示错误种类。在系统中输入真实数据，都能正常进行管理，都能正常完成数据库中信息的增加、删除、修改、查询等操作。系统连接速度能够保证目前大多上网方式的数据请求。系统中各作流程均能正常完成，并且当操作完成后，能够顺利流转到下一个过程。

第七章 结束语

7.1 结论

本系统的研究与设计就是为了提高企业人力资源管理水平而设计开发的企业员工信息管理系统。系统设计的目标是实现对企业人员的考勤进行统一管理，不断提高企业地工作效率、降低企业运行成本，充分满足用户及时沟通的需要，实现与其它系统协同工作。

在我的毕业设计的系统开发中，采取了 ASP.NET 技术平台架构，以统一建模语言(UML)对系统进行分析，本课题在实现上，论证了一个基于三层结构体系的 B/S 模式应用程序系统的实现过程，由于采用了优秀的 RAD 工具(快速开发工具)，降低了开发成本。采用了 B/S 模式，降低了系统的维护成本。而三层体系结构是当前最为流行的多层分布式结构体系中的一种典型结构，具有良好的可扩展性。

本篇论文主要阐述了企业员工管理系统的开发过程，从需求分析、数据库设计到系统详细设计与实现。虽然由于篇幅有限不能对整个系统作一个完整论述，但是基本能对本次毕业设计的核心技术有较完整的描述。本文主要描述了企业员工管理系统的模块分析、设计和实现过程，在考勤管理模块中，对加班申请、请假申请及考勤记录这些子功能模块的添加、删除、修改、退出系统及 Datagrid 控件的翻页等功能的实现，对加班查询、请假查询、考勤查询的查询功能的实现做了比较详细的描述。此外，还描述了企业员工管理系统的系统信息管理的用户注册、修改用户信息这两个功能的实现。

7.2 进一步工作的方向

本系统的设计开发使我受益颇多。在导师的悉心指导下，我真正负责了一个实际项目，从需求分析、设计到系统完成测试、撰写论文的整个软件开发过程，把理论应用到实践中去。毕业论文把我在学校中所学到的抽象理论融合到了实际的开发工具应用中，使我对软件设计，软件工程又有了一层新的领会，也锻炼了我的创新思维能力，同时使我对如何运用新的技术开发应用程序有了更深刻的了解。本系统还存在很多不足，需要进一步改善等。另外，由于本人经验方面的原

第七章 结束语

因，对于系统的功能设计方面还存在一些缺陷；还有由于 WEB 的开放性，WEB 服务器很容易遭到攻击，这要求在设计系统过程中还要充分考虑安全问题，这些都将在以后工作中逐步加以解决。

致 谢

在论文形成过程当中，我体会到了创作的艰辛与快乐，感受到了师长的关爱与鼓励。在此，我心怀感恩之情，诚挚的感谢曾给予我支持和帮助的师长和学友，尤其是我的指导老师张扬教授，从论文选题、设计、实施到写作，导师倾注了大量的时间和心血。张扬教授知识渊博、思维缜密、治学严谨、诲人不倦，给予我们很大的鼓励和支持。他渊博的学术知识，丰富、前卫的思维方式使我在知识方面受益匪浅。在我遇到问题和困难的时候，老师给予了我细心的解答和真诚的帮助，让我受益匪浅，使我能够顺利地圆满完成了企业员工管理系统的设计和开发工作。在这几年中我们相处的很融洽，我很高兴在张教授的指导下完成我的工程硕士毕业论文。在此，特向导师致以衷心的感谢！

在此，请允许我向指导和关心过我的老师们，说声：“老师，你们辛苦了，谢谢你们对我的谆谆教诲，是你们的帮助与教导，奠定了我完成此项目的基础，再次地感谢你们！

另外，我要感谢在学习和生活中给予我支持和帮助的同学和朋友们，在此，谨向他们表示我由衷的感谢！最后我要感谢我的家人，他们给予我精神上的支持和生活上的关照，使我能够安心的在学校完成学业，在此向他们表示深深的感谢！

参考文献

- [1]齐立波译.C#高级编程(第4版).清华大学出版社,2006.10.
- [2]汪星明.管理信息系统.北京:北京大学出版社,2004:20-21.
- [3]张友生.系统分析与设计技术[M].北京:清华大学出版社,2005:32-38.
- [4]Jason Price 等.Visual C#.NET 从入门到精通[M].北京:电子工业出版社,2003:93-107.
- [5]RichardABrianFrancIS 等 ASP.NET 高级编程[M] 北京清华大学出版社 2002.
- [6]刘永涛,肖中东.基于三层 C/S 和 B/S 集成的物流信息系统结构的研究[J].湖南科技大学学报(自然科学版).2005,20(3):85—89.
- [7]祈金佺.面向中小企业的员工考勤和培训管理信息系统的研究与设计[D].成都:电子科技大学.2008.10.
- [8]温涛,张永强,翟吉方.东软 Java 软件工程师实例参考手册,大连:东软电子出版社.
- [9]郑人杰.计算机软件测试技术[M].北京:清华大学出版社,2000:82-228.
- [10]李高健,孙瑛霖.ASP.NET 程序设计[M].北京:清华大学出版社,2002:212-304.
- [11]<http://blog.csdn.net/mmquit/archive/2007/03/28/1544447.ASPx>.
- [12]刘甫迎,刘光会,王蓉.C#程序设计教程[M].北京:电子工业出版社,2005.
- [13]刘军.电子商务系统的规划设计[M].人民邮电出版社 2003.4.
- [14]杨浩译.NET 软件测试指南[M].清华大学出版社,2007.9.
- [15]郭洪涛,刘丹妮,陈明华.ASP.NET(C#)大学实用教程[M].电子工业出版社.2003.9.
- [16]武新华等著.ASP.NET+SQL Server 典型网站建设[M].机械工业出版社,2007.
- [17]林成春等.SQL SERVER 2000 数据库实用技术[M].北京:中国铁道出版社,2005:113-224.
- [18]刘乃丽.精通 ASP.NET2.0+SQL Server 2005 项目开发[M].北京:人民邮电出版社,2007.
- [19]赛奎春,白伟明.信息系统开发实例精选[M].机械工业出版社,2006.
- [20]李虎,赵龙刚译 UML 基础案例于应用(第三版)[M].人民邮电出版社,2004.8.
- [21]赵克佳,沈志宇,赵慧.UNIX 程序设计教程[M].北京:清华大学出版社,2001.4.
- [22]陈绍英,戴金龙.软件测试案例分析[J].测试员.2005.8:23-25.
- [23]张海藩.软件工程导论[M].清华大学出版社 2007.8.
- [24]张龙祥,陶影等编.数据库原理与设计[M].西安交大出版社 2005.6.
- [25]刘润东.UML 对象设计与编程[M].广西:广西科技出版社,1998.
- [26]赵晓静.浅谈网站建设和网页制作的方法[J],电脑知识与技术,2005.9.

- [27]软件设计模式. [www. itisedu. com / 软件设计模式. htm](http://www.itisedu.com/软件设计模式.htm)[EB / OL]. 2004. 3.
- [28]许育诚. 软件测试与质量管理[D]. 上海: 海事大学, 2004. 6.
- [29]黄晓彤. Java 程序设计[M], 北京: 清华大学出版社, 2002.
- [30]Wendy Boggs, Michael Boggs. UML 与 Rational Rose 2002 从入门到精通[M]. 北京: 电子工业出版社, 2002. 7: 213-378.
- [31]夏邦贵, 郭胜. 数据库开发入门与范例解析[M]. 机械工业出版社, 2004. 5. 1.
- [32]郑人杰. 计算机软件测试技术[M]. 北京: 清华大学出版社, 2000:82-228.

附录 1：关键代码及部分编程实现

1. 首页登录代码

```
protected void btnRegister_Click(object sender, EventArgs e)
{
    Response.Redirect("Register.ASPx");
}

//登录
protected void btnLogin_Click(object sender, EventArgs e)
{
    string userName=txtUserName.Text.Trim();
    string Password=txtPassword.Text.Trim();
    ClassLibrary.UserRegister ur=new ClassLibrary.UserRegister();
    int value=ur.Login(userName, Password);
    if(value==1)
    {
        Session["userName"]=txtUserName.Text.Trim();
        string username=txtUserName.Text.Trim();
        DataTable dt=ur.getTureName(username);
        Session["turename"]=dt.Rows[0]["turename"].ToString();
        Session["department"]=dt.Rows[0]["department"].ToString();
        Session["post"]=dt.Rows[0]["post"].ToString();
        Response.Redirect("System.ASPx");
    }
    else
    {
        Response.Write("<script language=script>alert(' 用户名或密码错误!');</script>");
    }
}
```

```
}
```

2. 注册界面代码

```
protected void Page_Load(object sender, EventArgs)
{
    DateTime dt=DateTime.Now;
    txtDateTime.Text=dt.Year.ToString()+"-"+dt.Month.ToString()+"-"+
    dt.Day.ToString();
}
//提交
protected void btnSubmit_Click(object sender, EventArgs)
{
    if(txtPassword.Text.Trim()=="||txtCheckPassword.Text.Trim()=="
    ""||txtTureName.Text.Trim()=="||txtPost.Text.Trim()=="")
    {
        Response.Write("<script language=javascript>alert(' 姓名、密码和确认
        密码不能为空! ');</script>");
    }
    else
    {
        string userName=txtName.Text.Trim();
        ClassLibrary.UserRegister ur=new ClassLibrary.UserRegister();
        int value=ur.usernameCheck(userName);
        if(value==1)
        {
            Response.Write("<script language=javascript>alert(' 此用户名已存
            在! ');</script>");
        }
        else
        {

```


附 录

```
int returnValue=ur.userRegiser(model);
if(returnValue==1)
{
    MessageBox.Show("注册成功！");Response.Redirect("Login.ASPx");
}
}
}
}
//检测用户名
protected void btnCheck_Click(object sender, EventArgs e)
{
    string userName=txbName.Text.Trim();
    ClassLibrary.UserRegister ur=new ClassLibrary.UserRegister();
    int returnValue=ur.usernameCheck(userName);
    if(returnValue==1)
    {
        lblDisplay.Text="此用户名已存在！";
    }
    else
    {
        lblDisplay.Text="恭喜您，该用户名还未被注册，您可以使用这个用户名注册！";
    }
}
```

3. 修改信息代码实现

```
protected void btnModify_Click(object sender, Eve
{
    Model.UserRegisterModel model=new Model.U
    model.username=txbName.Text.Trim();
    model.password=txbNewPassword.Text.Trim();
```

```
model.department=txbDepartment.Text.Trim();
model.post=txbPost.Text.Trim();
ClassLibrary.UserRegister ur=new ClassLibrary.
int returnValue=ur.ModifyUserInformation(mode
if(returnValue==1)
{
    lblDisplay.Text="修改成功! ";
}
else
{
    lblDisplay.Text="修改失败! ";
}}
```

4. 员工管理系统核心代码及实现

4.1 员工档案基本信息

```
public partial class File Manage_PersonnelForm: System. Web.UI.Page
{
protected void Page_Load(object sender, EventArgs)
{
if(Session[ "MyForbid" ]!=null)
{
    string MyForbidString=Session[ "MyForbid" ]. ToString();
    if(MyForbidString. Indexof( "A1" )>1)
    {
        Server. Transfer( "/SystemManage / AllErrorHelp. ASPx" );
    }
}
else
{

```

附 录

```
Server. Transfer( “../logins. ASPx” );
    }
}
protected void Button2_Click(object sender, EventArgs)
{ // 新增员工基本档案信息
    this. SqlDataSource1. Insert();
}
protected void Button3_Click(object sender, EventArgs)
{ // 修改员工基本档案信息
    this. SqlDataSource1. Update();
}
protected void GridView1_RowCreated(object sender,
GridViewRowEventArgs e)
{ // 弹出删除操作确认对话框
    if(e.Row. RowType==DataControlRowType. DataRow)
    {
        Button MyButton=(Button)e.Row. FindControl( “Button1” );
        MyButton. OnClientClick= “return confirm( ‘是否确认删除当前选择的
记录’ )” ;
    }
}
protected void GridView1_SelectedIndexChanged(object sender, EventArgs)
{ // 在控件中显示选择的记录
    this. TextBox2. Text=this. GridView1. SelectedRow. Cells[3]. Text. ToS
tring();
    this. DropDownList 1. Text=
this. GridView1. selectedRow. Cells[4]. Text. ToString();
        this. TextBox3. Text=this. GridView1. SelectedRow. Cells[5]. Text. ToS
trong();
        this. DropDownList2. Text=
this. GridView1. SelectedRow. Cell[6]. Text. ToString();
        this. TextBox4. Text=this. GridView1. SelectedRow. Cells[7]. Text. ToS
```

```
tring()
    this.TextBox5.Text=this.GridView1.SelectedRow.Cells[8].Text.ToS
tring()
    this.TextBox6.Text=this.GridView1.SelectedRow.Cells[9].Text.ToS
tring()
    this.TextBox7.Text=this.GridView1.SelectedRow.Cells[10].Text.To
String()
    this.TextBox8.Text=this.GridView1.SelectedRow.Cells[11].Text.To
String()
    this.DropDownList3.Text=
this.GridView1.SelectedRow.Cell[12].Text.ToString();
    this.DropDownList4.Text=
this.GridView1.SelectedRow.Cell[13].Text.ToString();
    this.DropDownList5.Text=
this.GridView1.SelectedRow.Cell[14].Text.ToString();
    this.TextBox9.Text=this.GridView1.SelectedRow.Cells[15].Text.To
String()
    this.TextBox10.Text=
this.GridView1.SelectedRow.Cells[16].Text.ToString()
    this.TextBox11.Text=
this.GridView1.SelectedRow.Cells[17].Text.ToString()
    this.TextBox12.Text=
this.GridView1.SelectedRow.Cells[18].Text.ToString()
    this.DropDownList6.Text=
this.GridView1.SelectedRow.Cells[19].Text.ToString()
    this.TextBox13.Text=
this.GridView1.SelectedRow.Cells[20].Text.ToString()
    this.TextBox14.Text=
this.GridView1.SelectedRow.Cells[21].Text.ToString()
    this.TextBox15.Text=
this.GridView1.SelectedRow.Cells[22].Text.ToString()
    this.TextBox16.Text=
```

附 录

```
this.GridView1.SelectedRow.Cells[23].Text.ToString()
    this.TextBox17.Text=
this.GridView1.SelectedRow.Cells[24].Text.ToString()
    this.TextBox18.Text=
this.GridView1.SelectedRow.Cells[25].Text.ToString()
    this.TextBox19.Text=
this.GridView1.SelectedRow.Cells[26].Text.ToString()
    this.TextBox20.Text=
this.GridView1.SelectedRow.Cells[27].Text.ToString()
    this.TextBox21.Text=
this.GridView1.SelectedRow.Cells[28].Text.ToString()
    this.TextBox22.Text=
this.GridView1.SelectedRow.Cells[29].Text.ToString()
    this.TextBox23.Text=
this.GridView1.SelectedRow.Cells[30].Text.ToString()
    this.TextBox24.Text=this.GridView1.SelectedRow.Cells[2].Text.To
String()
    }
}
```

4.2 年度个人考勤:

```
public partial class CheckManage_YearCheckForm: System. Web. UI. Page
{
    protected void Page_Load(object sender, EventArgs)
    {
        if(Session[ "MyForbid" ]!=null)
        {
            string MyForbidstring=Session[ "MyForbid" ].ToString();
            if(MyForbidString.IndexOf( "B2" )>1)
            {
                Server. Transfer( " ~ / SystemManage / AllErrorHelp. ASPx" );
            }
        }
    }
}
```

```
    }  
}  
Else  
{  
    Server.Transfer( "../logins.ASPx" );  
}  
}  
Protected void Button2_Click(object sender, EventArgs)  
{  
    //打印年度个人考勤信息  
    Server.Transfer( "~/CheckManage/YearCheckPrint.ASPx" );  
}  
Public string MyPringtSQL  
{  
    //设置要传递到打印的数据  
    Get  
    {  
        Return "SELECT * FROM 考勤记录 WHERE(([员工编码]=' " +  
this.DropDownList2.SelectedValue.ToString()+ " ' ) AND([考勤年份])=' "  
+this.DropDownList3.SelectedValue.ToString()+ " ' ))" ;  
    }  
}  
Public String MyPringtDate  
{  
    //设置要传递到打印的数据  
    Get  
    {  
        Return "考勤日期:" +this.DropDownList3.SelectedValue.ToString()+  
"年" + "部门:" +this.DropDownlist1.SelectedValue.ToString()+ "姓名:"  
+this.DropDownList2.SelectedItem.Text.ToString();  
    }  
}  
Public String MyPrintTitele  
{  
    //设置要传递到打印页的数据  
    Get
```

附 录

```
{
    Return Session[ "MyCompanyName" ].ToString()+"年度个人考勤表";
}
}
```

4.3 查询考勤信息

```
Public partial class QueryMariage_QueryCheckForm: System. Web. UI. Page
{
    protected Void Page_Load(object sender, EventArgs)
    {
        if(session[MyForbid]!=null)
        {
            string MyForbidString=Session[MyForbid]. ToString();
            if(MyForbidString.IndexOf( "D3" )>1)
            {
                Server. Transfer( " ~/SystemManage/AllErrorHelp. ASPx" );
            }
        }
        else
        {
            Server. Transfer( "... /logins. ASPx" );
        }
    }

    protected Void Button2_Click(object sender, EventArgs)
    { //打印公司月度考勤信息
        Server. Transfer( " ~/ QueryManage/ QuerycheckPrint. ASPx" );
    }

    public string MyPrintSQL
    { //设置要传递到打印页的数据
```

```
Get
{
    Return “SELECT * FROM [全部考勤视图] WHERE(([考勤年份]=
“ ‘ +this.DropDownList1.SelectedValue.ToString()+’ ” AND([考勤月份]=
“ ‘ +this.DropDownList2.SelectedValue.ToString()+’ ” ))” ;
}
}

Public String MyPrintDate
{//设置要传递到打印页的数据
    Get
    {
        Retun “考勤日期: ” +this.DropDownList1.SelectedValue.ToString()+
“年” +this.DropDownList2.SelectedValue.ToString()+ “月”
    }
}

Public String Myprint Title
{//设置要传递到打印页的数据
    Get
    {
        Return Session[ “MyCompanyName” ].ToString()+ “员工年度考勤信息
表”;
    }
}
}
```

4.4 分配用户权限

```
public partial class System_OperatorForm: System. Web. UI. Page
{
    protected Void Page_Load(object sender, EventArgs)
    {
```


附 录

```
if(Session[ "MyForbid" ]!=null)
{
    string MyForbidString=Session[ "MyForbid" ]. ToString();
    if(MyForbidString. IndexOf( "E1" )>1)
    {
        Server. Transfer( "~ / SystemManage/AllErrorHelp. ASPx" );
    }
}
else
{
    Server. Transfer( "... /logins. ASPx" );
}

protected Void Button1_Click(object sender, EventArgs)
{
    //新增管理系统操作用户信息
    this. SqlDataSource2. Insert();
}

protected Void Button2_Click(object sender, EventArgs)
{
    //修改管理系统操作用户信息
    this. sqlDataSource2. update();
}

Protected Void GridView1_RowCreated(object sender, GridViewRowEventArgs)
{
    //弹出删除操作确认对话框
    if(e.Row. RowType==DataControlRowType. DataRow)
    {
        Button MyButton: (Button)e. Row. Find control( "Button1" );
        MyButton. OnClientClick= " return confirm( '是否确认删除当前选择的记录? ' )" ;
    }
}

protected Void GridView1_SelectedIndexChanged(object sender ,
EventArgs)
```

```

{ // 在控件中显示选择的记录
    This.DropDownList1.Text=This.GridView1.SelectedRow.Cells[2].Text.ToString();
    This.textBox1.Text=this.GridView1.SelectedRow.Cells[3].Text.ToString();
    This.DropDownList2.Text=this.GridView1.SelectedRow.Cells[4].Text.ToString();
    This.textBox3.Text=this.GridView1.SelectedRow.Cells[5].Text.ToString();
    This.textBox4.Text=this.GridView1.SelectedRow.Cells[6].Text.ToString();
    this.CheckBox1.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[7].Text.ToString())==1)?true: false;
    this.CheckBox2.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[8].Text.ToString())==1)?true: false;
    this.CheckBox3.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[9].Text.ToString())==1)?true: false;
    this.CheckBox4.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[10].Text.ToString())==1)?true: false;
    this.CheckBox5.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[11].Text.ToString())==1)?true: false;
    this.CheckBox6.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[12].Text.ToString())==1)?true: false;
    this.CheckBox7.Checked=
        (Convert.ToInt16(this.GridView1.selectedRow.Cells[13].Text.ToString())==1)?true: false;

```

附 录

```
this. CheckBox8. Checked=
(Convert.ToInt16(this.GridView1.selectedRow.Cells[14]. Text. ToString()
))==1)?true: false;
this. CheckBox9. Checked=
(Convert.ToInt16(this.GridView1.selectedRow.Cells[15]. Text. ToString()
))==1)?true: false;
this. CheckBox10. Checked=
(Convert.ToInt16(this.GridView1.selectedRow.Cells[16]. Text. ToString()
))==1)?true: false;
this. CheckBox11. Checked=
(Convert.ToInt16(this.GridView1.selectedRow.Cells[17]. Text. ToString()
))==1)?true: false;
this. CheckBox12. Checked=
(Convert.ToInt16(this.GridView1.selectedRow.Cells[18]. Text. ToString()
))==1)?true: false;
this. CheckBox13. Checked=
(Convert.ToInt16(this.GridView1.selectedRow.Cells[19]. Text. ToString()
))==1)?true: false;
this. CheckBox14. Checked=
(Convert.ToInt16(this.GridView1.SelectedRow.Cells[20]. Text. ToString()
))==1)?true: false;
this. CheckBox15. Checked=
(Convert.ToInt16(this.GridView1.SelectedRow.Cells[21]. Text. ToString()
))==1)?true: false;
this. CheckBox16. Checked=
(Convert.ToInt16(this.GridView1.SelectedRow.Cells[22]. Text. ToString()
))==1)?true: false;
this. CheckBox17. Checked=
(Convert.ToInt16(this.GridView1.SelectedRow.Cells[23]. Text. ToString()
))==1)?true: false;
this. CheckBox18. Checked=
(Convert.ToInt16(this.GridView1.SelectedRow.Cells[24]. Text. ToString()
```

```
ing()==1)?true: false;  
    }  
}
```

昂立信息企业员工管理系统设计与实现

作者: [刘长青](#)
学位授予单位: [电子科技大学](#)

引用本文格式: [刘长青](#) 昂立信息企业员工管理系统设计与实现[学位论文]硕士 2010