目录

【架构设计与数据库设计人员】项目系统设计与分析报告

“目录”字样小3号黑体居中，第一级标题：宋体小三号，缩2格；第二级标题：宋体小四号，缩1格；**需要手动调美观**

[1 引言 3](#_Toc404084082)

[1.1 编写目的 3](#_Toc404084083)

[1.2 读者对象 3](#_Toc404084084)

[1.3 术语解释 3](#_Toc404084085)

[1.4 执行标准与相关文档 3](#_Toc404084086)

[2 项目概述 3](#_Toc404084087)

[2.1 项目背景 4](#_Toc404084088)

[2.2 项目目标 4](#_Toc404084089)

[3 架构设计思想 4](#_Toc404084090)

[3.1 架构原则 4](#_Toc404084091)

[3.2 实现策略 4](#_Toc404084092)

[3.3 架构模式 4](#_Toc404084093)

[4 架构设计 4](#_Toc404084094)

[4.1 技术选型 4](#_Toc404084095)

[4.2 用例视图 5](#_Toc404084096)

[4.3 逻辑视图 5](#_Toc404084097)

[4.3.1 概述 5](#_Toc404084098)

[4.3.2逻辑架构 5](#_Toc404084099)

[4.4 开发视图 5](#_Toc404084100)

[4.5 物理视图 5](#_Toc404084101)

[4.6 进程视图 5](#_Toc404084102)

[4.7 数据存储设计 6](#_Toc404084103)

[4.8 性能设计 6](#_Toc404084104)

[5 数据库需求分析 6](#_Toc404084105)

[5.1 建立静态模型 6](#_Toc404084106)

[5.1.1 识别类 6](#_Toc404084107)

[5.1.2建立类图 6](#_Toc404084108)

[5.2 数据存储规划 6](#_Toc404084109)

[5.3 安全性和完整性要求 7](#_Toc404084110)

[5.4 数据库环境说明 7](#_Toc404084111)

[6 数据库的命名规则 7](#_Toc404084112)

[7 概念建模 7](#_Toc404084113)

[7.1 映射规则（映射策略） 7](#_Toc404084114)

[7.2 建模思路 7](#_Toc404084115)

[7.3 CDM图与ER图 8](#_Toc404084116)

[8 逻辑建模（可选） 8](#_Toc404084117)

[9 数据模型关系图 8](#_Toc404084118)

[10 物理设计 8](#_Toc404084119)

[10.1 物理数据模型 8](#_Toc404084120)

[10.2 实体表汇总 9](#_Toc404084121)

[10.2.1 表A 9](#_Toc404084122)

[10.2.2 表N 9](#_Toc404084123)

[10.3 存储过程设计 9](#_Toc404084124)

[10.4 视图设计 9](#_Toc404084125)

[10.5 安全性设计 10](#_Toc404084126)

[11 性能优化 10](#_Toc404084127)

[12 数据库管理与维护说明 11](#_Toc404084128)

[7总结与展望 11](#_Toc404084129)

[7.1 设计亮点总结 11](#_Toc404084134)

[7.2 设计缺陷总结 11](#_Toc404084135)

[7.3 心得与体会 11](#_Toc404084136)

[谢 辞 1](#_Toc404084137)

**书写格式：首行缩进2格**

**一级目录：1，宋体小三号，加粗，居左**

**二级目录：1.1，宋体四号。加粗，居左**

**三级目录：1.1.1，宋体小四，加粗，居左**

**四级目录：1.1.1.1，宋体五号，加粗，居左**

**五级始采用序号：（1），宋体小四**

**正文内容：宋体小四、行间距：22 Pt**

# 引言

## 编写目的

*提示：撰写本文档目的是什么？*

本文关于人员成长管理系统的系统设计与分析，主要内容包括

## 读者对象

本文档的读者有学生、教师、管理员、开发人员和测试人员。

*提示：阅读此文档的读者对象有哪些*。

## 术语解释

*提示：本小节应提供正确理解此文档所需的全部术语的定义、首字母缩写词和缩略语。*

|  |  |
| --- | --- |
| **缩写、术语** | **解 释** |
| MVC | 是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码 |
| SSH | Struts+Spring+Hibernate的一个集成框架，是目前较流行的一种Web应用程序开源框架 |
|  |  |
|  |  |
|  |  |

## 执行标准与相关文档

（1）《人员成长档案管理系统\_需求规格说明书》

（2）《毕业设计（工程文档）执行规范和要求V1.5》

*提示：编写些文档参考的标准与相关文档；*

# 项目概述

本系统是移动公司用于管理营销中心员工成长档案的一个信息管理系统。

# 项目背景

随着知识经济的到来，激烈的竞争将会使我国企业面临三方面的挑战：竞争的全球化挑战、满足利益相关者需要的挑战以及高绩效工作系统的挑战。企业越来越认识到人力资源是发展的第一资源，是企业获得并保持市场竞争优势的重要来源和途径，企业战略目标得以实现的重要保障。在人员成长管理与开发中，人员成长管理处于核心地位，建立科学的绩效管理体系已经成为企业培育核心竞争能力、获取持续竞争优势的必然选择。有效的人员管理是基于人本管理的思想，实现企业与员工发展的“双赢”。

随着计算机科学的发展与普及，人们越来越多的把现实中繁重、复杂的任务工作交给程序处理。人员成长信息的管理是企业管理的一个很重要的部分，而且员工成长信息量大，单纯的通过Excel等工具进行人工维护管理的话，不仅效率低，而且会影响到数据的准确性以及管理的质量。

目前，移动公司主要利用Office Excel等办公软件，进行人工式地录入、筛选、统计、核对员工的成长档案信息，从而大大降低了工作效率，增加了劳动成本。因此，对此现状，南京市移动公司希望借助于本系统替代人工作业方式，以达到减轻工作负担，提升工作效率的目的。

# 项目目标

本项目旨在开发一个界面友好，易于操作的员工成长信息管理系统，该系统能进行自动化处理员工成长信息。本系统能高效地实现员工信息管理系统的各项功能, 包括权限管理、人员综合信息管理、员工培训/考试管理、标准化和暗访管理、积分管理、创新管理、奖惩信息管理、绩效管理八大功能模块，通过这些功能模块，可以实现对人员的基本信息、考核成绩、进阶成绩、积分、创新信息、奖惩、绩效等信息的管理，包括录入、修改、删除、查询、批量导出等功能。

# 架构设计思想

本小节主要说明系统的架构原则和实施策略以及架构模式，通过这三方面来阐述本系统的架构设计思想。

## 架构原则

（1） 基于面对对象的程序设计思想，采用B/S三层架构思想设计

（2） 系统分成几个相对独立的模块，但这些模块都进行集中式管理

（3） 分层的模块化程序设计思想，整个系统采用模块化结构设计。作为应用程序有较强的可操作性和扩展性。

（4） 在应用系统设计中，相对独立的模块间以数据流相互连接，通过spring解耦使各模块间的耦合性较低，方便系统运行，提高系统安全性。

## 实现策略

1. 使用Struts2+Spring+Hibnerate集成开发。
2. 使用git版本控制和团队协同开发。
3. 分模块开发原则，组员负责各自模块开发实现，减少协同开发的冲突。

## 架构模式

本系统采用B/S服务架构，系统采用MVC（Model，Controller，View）的架构模式。

（1)视图（View）：视图层能够实现数据有目的的显示（理论上，这不是必需的）。在视图中一般没有程序上的逻辑。为了实现视图上的刷新功能，视图需要访问它监视的数据模型（Model），因此应该事先在被它监视的数据那里注册

（2)控制器（Controller）：控制器起到不同层面间的组织作用，用于控制应用程序的流程。它处理事件并作出响应。“事件”包括用户的行为和数据模型上的改变。

（3)模型层（Model）：“数据模型”（Model）用于封装与应用程序的业务逻辑相关的数据以及对数据的处理方法。“模型”有对数据直接访问的权力，例如对数据库的访问。“模型”不依赖“视图”和“控制器”，也就是说，模型不关心它会被如何显示或是如何被操作。但是模型中数据的变化一般会通过一种刷新机制被公布。为了实现这种机制，那些用于监视此模型的视图必须事先在此模型上注册，从而，视图可以了解在数据模型上发生的改变。

系统基于三层架构思想，系统分成五个模块：控制层(Action)、业务逻辑层(Service)、数据访问层(DAO)、数据模型层（Domain）、展现层(View)。数据访问层主要对数据库数据进行操作，包括增加数据，修改数据和删除数据。业务逻辑层基于数据访问层之上，它主要针对业务流程来实现系统的功能。控制层又基于业务逻辑层之上，它主要控制请求流转，并处理业务层通过数据访问层得到的数据在View表现层进行展现，以完成人员成长管理系统所要完成的功能。

实体类是领域模型的实体即数据模型，它与Database中的实体是一一对应的关系。Dao就是采用Model的形式来存取数据，采用Model或集合的形式来呈现或存储数据的。Service所处理的是业务流程中所要实现的功能，通过业务分析来定义，其根据可结合View来设计。

# 架构设计

技术的选择不仅要考虑到系统的功能性需求还要考虑团队当前的技术优势以及团队成员的技术发展要求。基于以上原因团队讨论确定以下技术进行系统开发。

## 技术选型

### 4.1.1 JavaEE技术规范（SSH）

JAVAEE是Java企业应用开发的技术规范，JavaEE是一个开放的、基于标准的平台，用以开发、部署和管理N层结构、面向Web的，以服务器为中心的企业级应用。SSH 为 Struts2+Spring+Hibernate的一个集成框架，是三个非常优秀的基于JAVAEE规范的开源框架，是目前较流行的一种Web应用程序开源框架。

集成SSH框架的系统从职责上分为四层：表示层、业务逻辑层、数据持久层和域模块层，以帮助开发人员在短期内搭建结构清晰、可复用性好、维护方便的Web应用程序。其中使用Struts作为系统的整体基础架构，负责MVC的分离，在Struts框架的模型部分，控制业务跳转，利用Hibernate框架对持久层提供支持，Spring做管理，管理Struts2和Hibernate。具体做法是：用面向对象的分析方法根据需求提出一些模型，将这些模型实现为基本的Java对象，然后编写基本的DAO(Data Access Objects)接口，并给出Hibernate的DAO实现，采用Hibernate架构实现的DAO类来实现Java类与数据库之间的转换和访问，最后由Spring做管理，管理Struts2和Hibernate。

### 4.1.2 ACE

ACE基于CSS框架Bootstrap，界面美观大方，开发的界面可实现三屏合一，是具有响应式布局的前台框架。

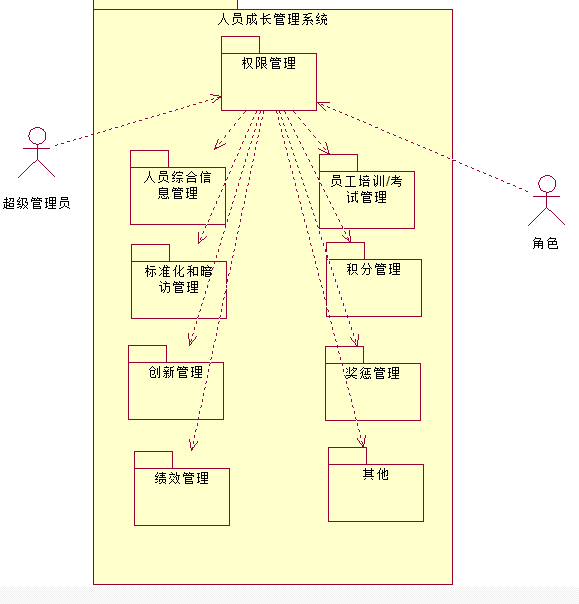
### 4.1.3 数据库和WEB容器

Oracle是是甲骨文公司的一款关系数据库管理系统。它是在数据库领域一直处于领先地位的产品。可以说Oracle数据库系统是目前世界上流行的关系数据库管理系统，系统可移植性好、使用方便、功能强，适用于各类大、中、小、微机环境。它是一种高效率、可靠性好的 适应高吞吐量的数据。库解决方案。考虑到成本和团队技术的熟练Web容器选择开源产品Tomcat7.0。

## 用例视图

用例视图（Use Case View）是被称为参与者的外部用户所能观察到的系统功能的模型图。用例是系统中的一个功能单元，可以被描述为参与者与系统之间的一次交互作用。所以该用例视图主要是从用户的角度来描述人员成长系统应该具备的功能，可以用户的角度和业务的角度来用用例视图描述整个系统的功能模块。因为本系统不存在后台系统或者其他子系统，本系统是由多个业务上的功能模块来划分，因此将通过业务的角度来分析用例视图。

本系统的用户的都是具有不同访问权限和不同功能操作的角色，其中超级管理员具有所有的权限，对所有的功能模块都有权限访问，角色通过权限管理模块对角色的权限进行控制，所有用户对系统的访问都得经过权限管理模块的验证才能访问相应的功能模块，用例视图如图4.1.



## 逻辑视图

本项目主要采用现在比较成熟的分层架构和MVC架构。系统架构模型如下图

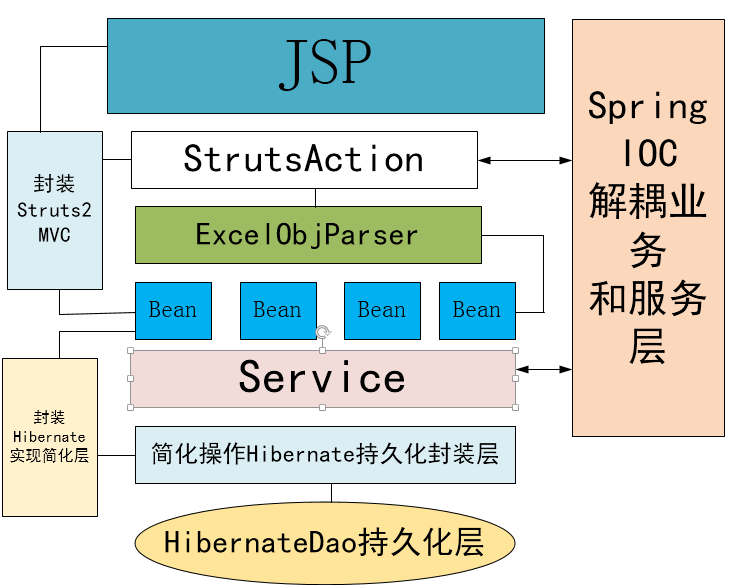


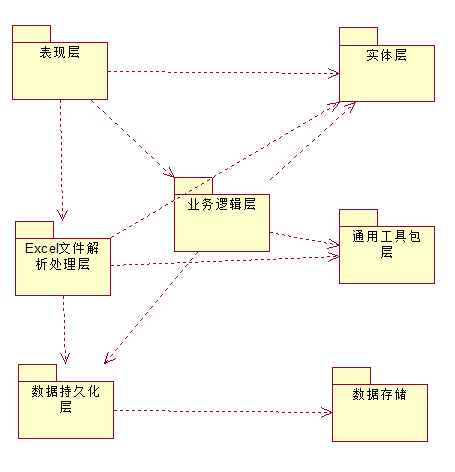
图3.1.1

### 4.3.1 概述

视图（JSP）接受用户输入的视图，用户参数，文件上传将会通过JSP页面收集。Struts2将表单数据类型转换和封装打包成对应的数据模型bean，充分利用Struts2MVC的特性。根据用户的请求，Struts2框架根据配置信息会找到对应的处理该JSP响应的action进行分发，实现程序和数据的流转。控制器（Struts2Actio）：根据表单请求分发数据处理模型处理。SpringIOC解耦，动态注入，实例化配置的对象，并实际操作Hibernate实现事务管理，ExcelObjParser:用户上传Excel文件时进入，实现Excel格式文件数据的类型转换和封装成bean对象并实现业务逻辑实现分发。Hibernate持久化简化层实现简化调用HibernateDao持久层的API，HibernateDao持久化层对数据库表的增删改查操作。

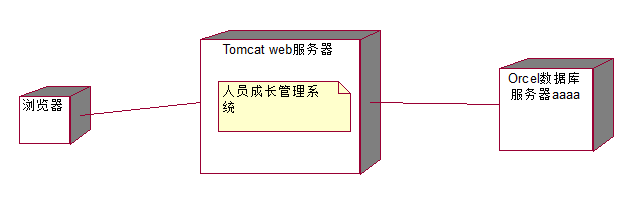
### 4.3.2逻辑架构

根据整体框架的分层思想，主要分为表现层、实体层、业务逻辑层、数据持久化层、通用工具层，因为本系统设计Excel文件上传下载，在涉及Excel文件操作时，业务数据将要通过Excel文件解析处理层处理，本系统还有一个Excel文件处理层，各层分别建立独立的包，通过包进行管理，以及描述它们之间的依赖关系。由此得到项目的逻辑视图，见总体逻辑视图图4.3。



## 开发视图

## 物理视图



*提示：物理视图：和部署相关的架构决策*

*软件最终要驻留、安装或部署到硬件才能运行，而软件架构的物理视图关注"目标程序及其依赖的运行库和系统软件"最终如何安装或部署到物理机器，以及 如何部署机器和网络来配合软件系统的可靠性、可伸缩性等要求。*

## 性能设计

1. 对于多次会使用到的数据使用自定义缓存和使用Hibernate缓存进行处理，
2. 对于耗时操作使用多线程技术解决
3. 对于Excel文件上传时多条数据要进行数据正确与完整性校验，使用二分算法加多线程解决

*提示：说明软件中会对构架产生影响的主要尺寸特征，以及目标性能约束。比如：1表示层性能设计考虑、应用层性能设计考虑、数据层性能设计考虑*

*在数据库设计方面，提供的架构级服务包括：*

*针对焦点表，进行表分割，按业务特性划分活动表和非活动表，提高OLTP（针对活动表的处理）效率；优化存取路径，检查索引效率；*

# 数据库需求分析

数据库设计是指对于一个给定的应用环境，构造（设计）优化的数据库逻辑模式和物理结构，并根据此建立数据库及应用系统，使之能够有效地存储和管理数据，满足各种用户的应用需求，包括信息管理要求和数据库操作要求[]。数据库设计首先要明确需求。数据库需求分析就是明确详细处理的对象，深入挖掘用户需求，并且根据用户需求制作出数据流图和数据字典，并在此基础上确定系统的整个功能，最后编辑出需求说明书[8]。数据库需求分析是分析客户在数据方面的信息需求、处理需求、安全性与完整性要求。应该注意一点，数据库设计要尽量考虑到系统以后的扩展。

*提示：通过详细调查现实世界要处理的对象（组织、部门、企业等），充分了解原系统(手工系统或计算机系统)的工作概况，明确用户需求，确定新系统的功能。*

## 建立静态模型

在功能性需求方面，本系统涵盖了权限管理、人员综合信息管理、员工培训/考试管理、标准化和暗访管理、积分管理、创新管理、奖惩信息管理、绩效管理八大功能模块。下面用面向对象的分析法来详细描述员工奖惩管理建立静态模型的过程。

员工奖惩信息管理是具有该功能权限授权的角色（部门管理员/部门经理/区域管理员）可以通过该功能模块对员工奖惩信息进行单条录入、批量录入、员工奖惩信息查询、员工奖惩信息删除，员工奖惩信息修改，员工信息批量导出操作。其功能性需求如下：

### 5.1.1员工奖惩信息单条录入

|  |  |
| --- | --- |
| 优先级 | 高 |
| 名称 | 员工奖惩信息单条录入 |
| 描述 | 员工奖惩信息单条录入 |
| 角色 | 部门管理员/部门经理/区域管理员 |
| 前提条件 | 操作者登录本系统中，并具有该功能的操作权限 |
| 主流程 | 1、用户登录后，进入员工奖惩信息单条录入界面  2、选择员工获取员工姓名、身份证、  3、输入：类型、奖惩原因、时间、级别等信息  4、信息保存到系统中 |
| 后置条件 | 员工奖惩信息保存到系统 |
| 相关需求 |  |
| 说明 | 1、提供弹出式画面，选择员工  2、必输项：身份证、类型、奖惩原因、时间、级别 |

### 5.1.2员工奖惩信息批量录入

|  |  |
| --- | --- |
| 优先级 | 高 |
| 名称 | 员工奖惩信息批量导入 |
| 描述 | 员工奖惩信息批量导入 |
| 角色 | 部门管理员/部门经理/区域管理员 |
| 前提条件 | 操作者登录本系统中，并具有该功能的操作权限 |
| 主流程 | 1、用户登录后，进入员工奖惩信息批量导入界面  2、选择导入的文件  3、员工奖惩信息保存到系统中 |
| 分支流程 |  |
| 后置条件 | 员工奖惩信息保存到系统数据库 |
| 相关需求 |  |
| 说明 | 1、录入信息：姓名、身份证、类型、奖惩原因、时间、级别  2、分值不需录入，显示时通过分值表计算表示  3、导入格式参照"员工奖惩信息批量导入模板.xls"文件 |

### 5.1.3员工奖惩信息查询

|  |  |
| --- | --- |
| 优先级 | 高 |
| 名称 | 员工奖惩信息查询 |
| 描述 | 员工奖惩信息查询 |
| 角色 | 所有用户，但根据权限的不同，显示的结果不同 |
| 前提条件 | 操作者登录本系统中，并具有该功能的操作权限 |
| 主流程 | 1、用户登录后， 进入员工奖惩信息查询界面  2、输入查询条件  3、显示总分列表画面  4、点击总分分数，进入奖惩详细列表画面 |
| 后置条件 | 显示员工奖惩信息。 |
| 相关需求 | 员工奖惩信息修改  员工奖惩信息删除  员工奖惩信息导出 |
| 说明 | 1. 输入查询条件：时间段、员工姓名（模糊型）、类型（惩罚/表彰/月度考试成绩/新员工考核成绩/营业厅表彰/营业厅惩罚）、级别 2. 总分列表展示的信息： 序号/身份证/员工姓名/年月/总分 3. 奖惩详细列表展示的信息： 序号/身份证/员工姓名/类型/奖惩原因/时间/级别/分值 |

### 5.1.4员工奖惩信息修改

|  |  |
| --- | --- |
| 优先级 | 高 |
| 名称 | 员工奖惩信息修改 |
| 描述 | 员工奖惩信息修改 |
| 角色 | 部门管理员/部门经理 |
| 前提条件 | 操作者登录本系统中，并具有该功能的操作权限 |
| 主流程 | 1. 用户登录后， 经过员工奖惩信息查询->查询结果列表->详细画面，进入员工奖惩信息修改画面   2、输入修改信息  3、员工奖惩信息保存到系统 |
| 后置条件 | 员工奖惩信息保存到系统数据库 |
| 相关需求 | 员工奖惩信息查询 |
| 说明 | 1. 可修改信息：类型、奖惩原因、时间，级别 2. 只能修改“惩罚”和“表彰”两种类型的奖惩信息 3. 不能修改员工姓名 |

### 5.1.5员工奖惩信息删除

|  |  |
| --- | --- |
| 优先级 | 高 |
| 名称 | 员工奖惩信息删除 |
| 描述 | 员工奖惩信息删除 |
| 角色 | 部门管理员/部门经理 |
| 前提条件 | 操作者登录本系统中，并具有该功能的操作权限 |
| 主流程 | 1. 用户登录后，经过员工奖惩信息查询->查询结果列表->详细画面 2. 选择要删除的数据 3. 将选中的数据删除 |
| 后置条件 |  |
| 相关需求 | 员工奖惩信息查询 |
| 说明 | 1. 删除为物理删除 2. 记录删除日志 3. 只能删除“惩罚”和“表彰”两种类型的奖惩信息 |

### 5.1.6员工奖惩信息批量导出

|  |  |
| --- | --- |
| 名称 | 员工奖惩信息导出 |
| 描述 | 员工奖惩信息导出 |
| 角色 | 所有用户，但根据权限的不同，导出的结果不同 |
| 前提条件 | 操作者登录本系统中，并具有该功能的操作权限 |
| 主流程 | 1、用户登录后，经过员工奖惩信息查询，进入查询结果列表画面  2、点击[导出]按钮  3、将结果导出到Excel |
| 后置条件 | 员工奖惩信息保存到Excel文件中 |
| 相关需求 | 员工奖惩信息查询 |
| 说明 | 1、导出格式：  按查询结果按详细信息形式输出，但不包括总分  排序：姓名、时间升序排序  2、导出格式参照"员工奖惩信息批量导出模板.xls"文件 |

*提示：指通过分析用例模型、领域模型等，建立起数据的静态模型；（以下小结可依据各自情况做调整）*

### 5.1.1 识别类

根据员工奖惩信息管理功能性需求的描述可知用户是以具有权限授权的角色访问本系统，用户也是系统所属公司的员工。由此可知系统中有一个角色类，其中超级管理员和部门管理员/部门经理等都属于角色类中的一种，而一些特定的角色又与用户的职位相关联，所以可以得到一个职位类和员工类和角色类。员工类要登录系统，可以退出需要有“登录密码”属性，角色类有一个特殊的角色具有所有访问权限，即超级管理员，所有角色类有一个“超级管理员标记”的属性，访问权限就是对一个系统功能的访问，可以得到一个系统功能类，因为角色和系统功能类是多对多的关系，所以建立一个角色功能类。奖惩信息管理是管理奖惩信息，所以奖惩信息类是必须的，根据以上的方法对需求进行分析并进行合理性添加属性，可以得到一下类及其属性

营业厅类（营业厅编码，营业厅名称，地址，营业厅类型）

用户类（工号，姓名，所属营业厅名称，职位名称，身份证，角色名称，登录密码）

职位类（职位编号，职位名称，）

角色类（角色编号，角色名称，超级管理标记）

角色功能类（角色编号，功能编号）

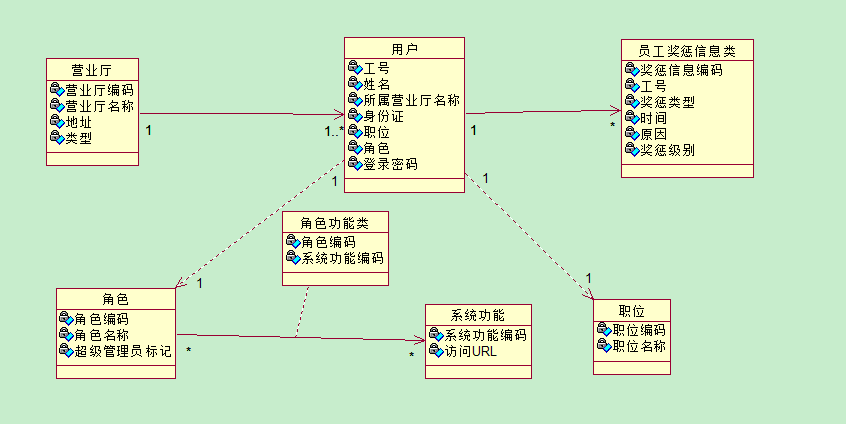
系统功能类（功能编号，访问链接）

员工奖惩信息类（奖惩信息编码，工号，奖惩类型，奖惩原因，时间，奖惩级别）

*提示：指通过分析用例模型、领域模型等，建立起数据的静态模型；（以下小结可依据各自情况做调整）*

### 5.1.2建立类图

在得到员工奖惩信息的基本类后考虑各类之间的关系，员工隶属于营业厅，所以员工和营业厅是多对一的关系。一个员工正常情况下，在本系统只考虑一个员工只有一个职位，不考虑身兼多职的情况，所以员工和职位是一对一，和角色也是一对一的关系。员工在工作期间可能多次获得奖励或者惩罚，所以员工与奖惩信息类是一对多的关系，角色和系统功能类是多对多的关系。由以上分析和画出类图如



*提示：指通过分析用例模型、领域模型等寻找类；*

*提示：通过提取用例规约中描述的业务规则及字段列表等信息，获取类属性；描述分析类与类之间的1:1,1:n及n:n的关系；并最终建立起类图*

## 数据存储规划

本系统涉及到将Excel表中的批量数据导入到数据库，所以需要存储Excel表中数据与数据库表的映射关系，而且上传的Excel文件时要符合公司的报表格式，所以有标准的Excel模板文件需要管理。

1. Excel表中数据与数据库表的映射存储在Xml文件中。
2. 模板文件存储在Web静态资源目录下，提供用户下载。

## 安全性和完整性要求

1. 主键约束，使用Auto触发器自增序列实现主键约束。
2. 用户密码加密。使用MD5加密
3. 基础数据，比如用户表实现假删除，设置状态属性。
4. 为防止数据丢失，多重备份，定时备份。

*提示：用户对于数据安全性和完整性的要求？*

## 数据库环境说明

因为Oracle数据在商用领域比较流行，为了增加对Oracle数据库的操作，所以选择使用Oracle作为数据库服务器，本系统采用Oracle11g作为后台数据库的开发环境。采用Power Designer进行数据库模型设计，使用SqlDeveloper进行包、存储过程、触发器的开发。

*提示：*

*（1）说明所采用的数据库系统及理由？*

*（2）设计工具，编程工具等并说明理由？*

*（3）详细配置*

# 数据库的命名规则

1. 数据库的命名：由英文小写字母组成，如：ppms；

2、数据库表的命名：tb\_表义名（可以使用缩写），如：tb\_Employee；

3、属性名的由其英文大小写单词组成，开头字母大写，单词之间用大小写区分，如：EnterTime；

4、主键由表头名去掉表义名，加Id组成，如：EmployeeId。

# 概念建模

根据数据库需求分析的结果，按照类与实体的映射规则，运用PowerDesigner建模工具来建立数据库的概念数据模型。

## 映射规则（映射策略）

1. 属性类型映射成域
2. 属性映射成字段
3. 类映射成表
4. 对象之间的关系也需要映射至数据库。对象之间的关系可分为：继承（Inheritance），关联（association），聚集（aggregation），组合（composition）。欲有效地映射关系，必须理解它们之间的不同点，如何实现一般的关系以及如何实现特定的多对多关系。

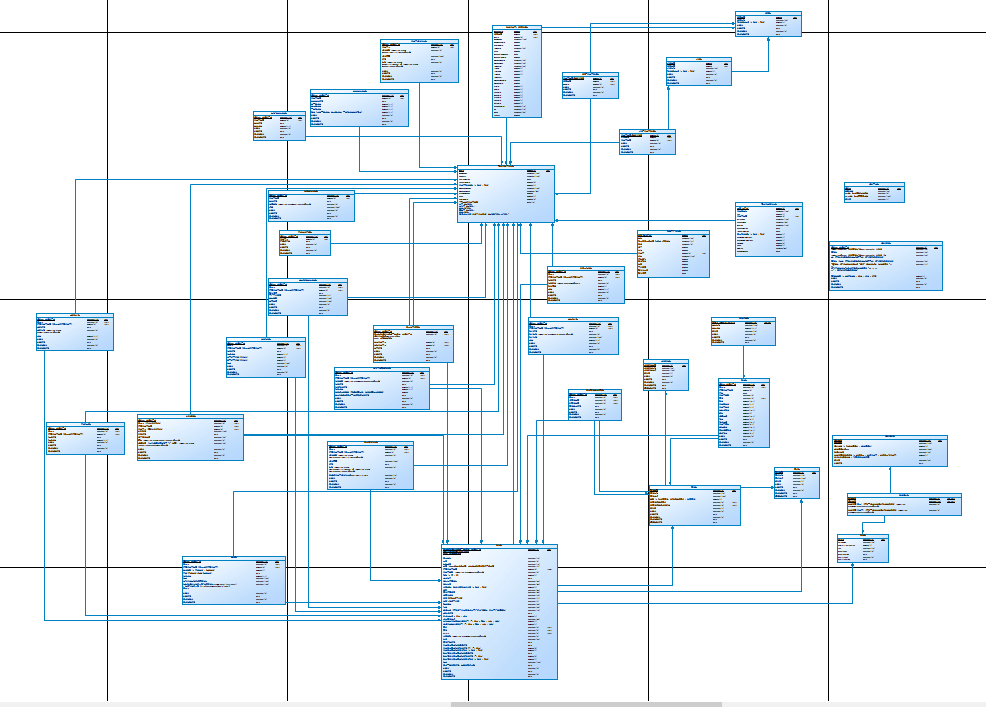
[]http://wenku.baidu.com/link?url=r6i8YYquQRA-3qjS9iY0oP0kHdvR9onaCVy9h-72vQjy18WV-\_pAeTzV\_El2K0iysUqQVHOjnPmjmIx4g\_dpJd0cu5rhd-mdRQepEOjLUZq

*提示：请描述根据静态模型进行数据库概念结构建模时所遵循的类到实体的映射规则或映射策略。*

## 建模思路

由类图根据映射规则得到实体关系图，实体关系图利用PowerDesigner工具建模得到CDM图。本项目采用以小见大的方法来阐述，通过一个单独描述员工奖惩信息模块来描述过程。

## CDM图



*提示：用CDM图请描述重要实体之间明细的对应关系。考虑CDM图信息太多，可辅助ER图说明数据的概念模型。*

# 数据模型关系图

*提示：请描述数据关系*

# 物理设计

*提示：*

*（1）主要是设计表结构。一般地，实体对应于表，实体的属性对应于表的列，实体之间的关系成为表的约束。逻辑设计中的实体大部分可以转换成物理设计中的表，但是它们并不一定是一一对应的。*

*（2）对表结构进行规范化处理（第三范式）。*

*数据库在物理设备上的存储结构与存储方法称为数据库的物理结构，它依赖于给定的计算机系统。为一个给定的逻辑数据模型选取一个最适合应用要求的物理结果的过程，就是数据库的物理设计。*

*数据库的物理结构设计通常分为两步：*

*（1）确定数据库的物理结构，在关系数据库中主要指存取方法和存储结构；*

*（2）对物理结构进行评价，评价的重点是时间和空间效率。*

*（3）根据各自所需，有选择地进行索引设计、约束设计等。*

## 物理数据模型

*提示：请描述生成物理模型的过程及PDM图的相关设计;*

## 实体表汇总

|  |  |
| --- | --- |
| 数据表名 | 用户信息表 |
| [operator\_nj](#RANGE!C2) | 同步营业员信息表 |
| [c\_user\_info](#RANGE!C2) | 本地用户信息表 |
| [organization\_nj](#RANGE!C2) | 同步营业厅信息表 |
| [c\_organization\_nj](#RANGE!C2) | 本地营业厅、区域关系表 |
| [tb\_Area](#RANGE!C2) | 区域表 |
| [tb\_AreaOrgRelation](#RANGE!C2) | 区域-营业厅关系表 |
| [tb\_SubArea](#RANGE!C2) | 片区表 |
| [tb\_SubAreaOrgRelation](#RANGE!C2) | 片区-营业厅关系表 |
| [tb\_Master](#RANGE!C2) | 系统字典表 |
| [tb\_SystemSettings](#RANGE!C2) | 系统设定表 |
| [tb\_SystemFunction](#RANGE!C2) | 系统功能表 |
| [tb\_Role](#RANGE!C2) | 角色表 |
| [tb\_RoleFunction](#RANGE!C2) | 角色功能表 |
| [tb\_Employee](#RANGE!C2) | 员工表 |
| [tb\_Stage](#RANGE!C2) | 成长阶段表 |
| [tb\_Post](#RANGE!C2) | 岗职表 |
| [tb\_Job](#RANGE!C2) | 岗位表 |
| [tb\_ChangeOrgHistory](#RANGE!C2) | 员工调厅记录表 |
| [tb\_ChangeJobHistory](#RANGE!C2) | 员工岗位变更记录表 |
| [tb\_MountGuardExam](#RANGE!C2) | 合作厅上岗证考核表 |
| [tb\_MonthExam](#RANGE!C2) | 月度考试表 |
| [tb\_OperationTraining](#RANGE!C2) | 业务培训表 |
| [tb\_PromotionTraining](#RANGE!C2) | 进阶培训表 |
| [tb\_FreshEmployeeExam](#RANGE!C2) | 新员工考核表 |
| [tb\_BecomeEmployeeExam](#RANGE!C2) | 转正考核表 |
| [tb\_StandardCheck](#RANGE!C2) | 标准化检查成绩表 |
| [tb\_MonitorCheck](#RANGE!C2) | 监控检查成绩表 |
| [tb\_VisitCheck](#RANGE!C2) | 暗访检查成绩表 |
| [tb\_OperationCheck](#RANGE!C2) | 业务差错检查成绩表 |
| [tb\_Point](#RANGE!C2) | 积分表 |
| [tb\_PointDetail](#RANGE!C2) | 积分明细表 |
| [tb\_Innovation](#RANGE!C2) | 创新提案表 |
| [tb\_EmployeePraiseCriticism](#RANGE!C2) | 员工奖惩信息表 |
| [tb\_OrgPraiseCriticism](#RANGE!C2) | 营业厅奖惩信息表 |
| [tb\_Performance](#RANGE!C2) | 绩效表 |
| [tb\_MonthPerformanceOpen](#RANGE!C2) | 月度绩效开通表 |

### 表A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表名 |  | | | |
| 字段名称 | 属性 | 非空 | 主外键 | 约束条件 |
| OPERATOR\_ID | NUMBER(8) | 是 | 是 |  |
| NAME | VARCHAR2(160) |  |  |  |
| SEX | NUMBER(1) |  |  |  |
| ADDR | VARCHAR2(256) |  |  |  |
| TELEPHONE | VARCHAR2(40) |  |  |  |
| OFFICE | VARCHAR2(256) |  |  |  |
| CREATE\_TIME | DATE |  |  |  |
| INVALID\_TIME | DATE |  |  |  |
| STATUS\_ID | NUMBER(1) |  |  |  |
| MAX\_LOGIN\_TIMES | NUMBER(2) |  |  |  |
| CURR\_LOGIN\_TIMES | NUMBER(2) |  |  |  |
| ORG\_ID | NUMBER(8) |  |  |  |
| EMP\_NO | NUMBER(8) |  |  |  |
| TYPE | NUMBER(1) |  |  |  |
| MOBILE | VARCHAR2(40) |  |  |  |
| CHANGE\_DATE | DATE |  |  |  |
| 补充说明 |  | | | |

### 表A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表名 |  | | | |
| 字段名称 | 属性 | 非空 | 主外键 | 约束条件 |
| OPER\_ID | NUMBER | 是 | 是 |  |
| OPER\_NAME | VARCHAR2(80) |  |  |  |
| OPER\_PASS | VARCHAR2(32) |  |  |  |
| ROLE\_ID | VARCHAR2(250) |  |  |  |
| DEPART\_ID | NUMBER |  |  |  |
| CENTER\_ID | NUMBER |  |  |  |
| ORG\_ID | NUMBER |  |  |  |
| ZW\_ID | VARCHAR2(250) |  |  |  |
| USER\_COUNTY | NUMBER |  |  |  |
| IS\_VALID | NUMBER |  |  |  |
| SHORT\_NUM | NUMBER |  |  |  |
| MOBILE\_NUM | NUMBER |  |  |  |
| OPERATING\_ID | NUMBER |  |  |  |
| OPERATING\_DATE | DATE |  |  |  |
| 补充说明 |  | | | |

### 表A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表名 |  | | | |
| 字段名称 | 属性 | 非空 | 主外键 | 约束条件 |
| ORG\_ID | NUMBER(8) | 是 | 是 |  |
| ORG\_NAME | VARCHAR2(160) |  |  |  |
| ADDRESS | VARCHAR2(256) |  |  |  |
| EXPIRE\_TIME | DATE |  |  |  |
| SUPERIOR\_ID | NUMBER(8) |  |  |  |
| STATUS | NUMBER(1) |  |  |  |
| CONTACT\_NAME | VARCHAR2(100) |  |  |  |
| CONTACT\_TEL | VARCHAR2(40) |  |  |  |
| USER\_COUNTY | NUMBER |  |  |  |
| TYPE | NUMBER(2) |  |  |  |
| REGION\_ID | NUMBER(8) |  |  |  |
| CHANNEL\_TYPE | CHAR(28) |  |  |  |
| 补充说明 |  | | | |
|  |  |  |  |  |

### 表A

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 属性 | 非空 | 字段说明 |
| OPERATOR\_ID | NUMBER(8) | ○ | 工号(14开头) |
| NAME | VARCHAR2(160) |  | 营业员姓名 |
| SEX | NUMBER(1) |  |  |
| ADDR | VARCHAR2(256) |  |  |
| TELEPHONE | VARCHAR2(40) |  |  |
| OFFICE | VARCHAR2(256) |  |  |
| CREATE\_TIME | DATE |  |  |
| INVALID\_TIME | DATE |  |  |
| STATUS\_ID | NUMBER(1) |  | 员工有效状态 (0:无效 1:有效) |
| MAX\_LOGIN\_TIMES | NUMBER(2) |  |  |
| CURR\_LOGIN\_TIMES | NUMBER(2) |  |  |
| ORG\_ID | NUMBER(8) |  | 营业厅编码 |
| EMP\_NO | NUMBER(8) |  |  |
| TYPE | NUMBER(1) |  |  |
| MOBILE | VARCHAR2(40) |  |  |
| CHANGE\_DATE | DATE |  |  |

### 表A

### 表A

### 表A

### v表A

## 存储过程设计

*提示：*

1. *是否进行存储过程设计？为什么？*
2. *如何设计存储过程？设计结果是？*

## 安全性设计

*提示：提高软件系统的安全性应当从“管理”和“设计”两方面着手。这里仅考虑数据库的安全性设计。以下章节可以增改；*

#### 防止用户直接操作数据库

*提示：用户只能用帐号登陆到应用软件，通过应用软件访问数据库，而没有其它途径操作数据库。*

#### 用户帐号密码加密

*提示：对用户帐号的密码进行加密处理，确保在任何地方都不会出现密码的明文。*

#### 角色与权限

*提示：确定每个角色对数据库表的操作权限，如创建、检索、更新、删除等。每个角色拥有刚好能够完成任务的权限，不多也不少。在应用时再为用户分配角色，则每个用户的权限等于他所兼角色的权限之和。*

|  |  |  |
| --- | --- | --- |
| 角色 | 可以访问的表与列 | 操作权限 |
| 角色A |  |  |
|  |  |
|  |  |
| 角色B |  |  |
|  |  |
|  |  |

# 性能优化

1. 在某些业务场景下，查询数据获取数据对象并不是需要操作到数据对象所关联的对象，因此开启Hibernate延迟加载，并提供一个API可以通过参数动态在当前操作下关闭延迟加载，而且可以指定要加载的关联对象。
2. 在获取角色具有访问权限的功能时，因为角色和功能权限是多对多的关系，当一个角色具有多个权限时，比如具有所有权限的超级管理员，将会要操作（查询和插入）多天记录，因此将角色具有的访问功能的id按一定格式拼接，存储到角色表中的一个字段，这样在操作角色权限是并不是操作多条记录，只是操作一条数据，对该数据存储功能ID拼接字符串进行操作就能操作当前角色具有访问权限的id。
3. 数据批量导入时，存在一个操作要执行多个插入语句，因此将这些插入操作做成一个原子事务，提高插入数据的速度。
4. 在本系统内数据批量导出都是在页面展示数据后操作，因此将页面展示的数据查询出来后，存储到一个缓存内，在一定时间内用户点击导出数据操作的话就不需要再操作数据库，而是从缓存取出数据生成Excel文件，如果超时，缓存自动清除。

*提示：分析并优化数据库的“时－空”效率，尽可能地“提高处理速度”并且“降低数据占用空间”。*

*（1）分析“时－空”效率的瓶颈，找出优化对象（目标），并确定优先级。*

*（2）当优化对象（目标）之间存在对抗时，给出折衷方案。*

*（3）给出优化的具体措施，例如优化数据库环境参数，对表格进行反规范化处理等。*

|  |  |  |
| --- | --- | --- |
| 优先级 | 优化对象（目标） | 措施 |
|  |  |  |
|  |  |  |
|  |  |  |

# 数据库管理与维护说明

*提示：在设计数据库的时候，及时给出管理与维护本数据库的方法，有助于将来撰写出正确完备的用户手册。*

*完成数据库的物理设计以后，设计人员就要用RDBMS提供的数据定义语言和其他使用程序将数据库逻辑设计和物理设计结果严格描述出来，成为DBMS可以接受的源代码，再经过调试产生目标模式，然后就可以组织数据入库了。*

# 7总结与展望



## 设计亮点总结

*提示：按以下格式详细描述设计成果有哪些亮点？*

亮点1：

亮点2：

## 设计缺陷总结

*提示：你认为本次分析设计还存在哪些方面的缺陷、产生原因是什么及改进措施？按以下格式进行描述*

缺陷1：

产生原因：

改进措施：

## 心得与体会

两个月的开发周期短暂而又漫长，和团队成员一起克服技术难度，解决错误的过程痛苦并快乐着。人员成长管理系统一个涉及一百三十多个功能需求的系统，对于大部分没有开发经验的团队成员来说确实是个很大的项目，但是团队成员的积极努力，不畏艰难深深的打动了我，在熊老师的帮助下系统也如期完成，这对于我们来说是一次巨大的成长。在开发的需求阶段是一个非常痛苦的过程，因为这只是一个实训项目，我们并不能真正地从客户那里获取需求，只能通过一些文档和网络资料运用自己的想法去做需求，在这个阶段真的要特别感谢熊老师给了我们极大的支持，也给我们很大的发挥空间，虽然需求做的并不是很专业，但最终我们还是顺利的通过了这个阶段，形成了初步的需求文档。

在整个项目开发的过程中我主要负责系统的架构设计，以及应用系统开发框架的搭建，在搭建SSH开发架构的时候，深切的感受到自己对于SSH的了解太少，不过通过看前辈的博客，我很迅速的了解了SSH框架的一些基本知识，也勉强的搭起了整个应用框架，虽然后期开发过程中因为框架的搭建并不是很好，也影响到了项目的进度，我感到非常的愧疚，不过这些问题在查找资料和老师的指导下最终都解决，我也深深感受到只有在不断解决问题的过程中，才能了解到更多的细节，获得更多的技能和知识，所以并不觉得解决问题的过程是痛苦的，反而能感受到不一样的成就感。

随着我前期系统分析与设计工作的结束，在项目编码阶段，我也参与到了具体的编码工作，我主要是负责开发一个统一的Excel文件批量上传和批量下载的API，在这个开发过程中我尝试了各种方式进行设计一个能完全通用的API，经过三次版本的修订最后终于完成了，在实现的过程中，我多次使用反射与注解和XML文件配置，这个过程让我加深了对框架设计的理解，也提高了我对Java反射，注解等搞基特性的理解和使用。

两个月的实战开发非常的有意义，不经提高了我的IT技能，也发现了自己的不足，更提高自己和团队的合作精神。在未来的学习或职业生涯，我一定会更加努力，学习更多的技能，努力充实和成长自己。

# 谢 辞

“致谢“标题居中，小三号黑体;内容：宋体小四号，固定行距22Pt

四年的大学生活，迷茫过，充实过，最终就业结束。非常幸运的在南昌航空大学度过这人生中最重要的青春年华，也非常荣幸能做微软技术中心度过这成长飞快的实训日子。

在微软技术中心，我学到的不仅仅只是一些专业知识，这边学习的和贴近社会的氛围也让我更知道自己想要的是什么，社会需要的是什么，所以在这种氛围下，我充分发挥自己的自学能力，学到很多书本上学不到的知识和技能，在中心老师的指导下，也接触到了许多流行，有用的技术，身边一群爱好技术的朋友也让我的眼界更加宽广，看到了IT世界的更多有意思的东西，也极大的提高我的兴趣和学习积极性。在两个月的实训过程和一年的学习经历里，各位老师给了我们极大的帮助，下面对这些老师给予的帮助表示感谢。

感谢我们的指导老师熊老师，在整个的项目开发过程中给予我们技术支持，在需求分析、架构设计、数据库设计和编码阶段对我们出现的错误给予纠正，在这过程中我们队项目的开发流程有了整体的认识以及学到了很多软件开发相关的知识。

感谢我们的娄老师，在编码规范和文档编写规范给予了很大的帮助，让我们在项目开发过程中减少了很多不必要的麻烦。

感谢我们的董老师，在项目测试和编写测试用例文档方面给予了很大的帮助，让我们对测试有了初步的了解，如期完成了项目相关测试。

感谢我们的王老师，在项目开发过程中对我们的项目进度有了合理的规划，让我们对项目开发周期有了初步的认识。

感谢班主任陆佳对我们的监督，让我们如期完成项目。

感谢我们学校的任课老师和中心任课老师给我们带来专业的知识，特别感谢要从学校老师风云无阻从学校来到中心给我们上课，答疑解惑。

最后感谢我们的组员，大家共同合作、共同探讨、互相学习使项目得以完成。

*提示：对给予本文研究的选题、构思、实验或撰写等方面给以指导、帮助或建议的人员致以谢意*

“谢辞“标题居中，小三号黑体;内容：宋体小四号，固定行距22Pt