

华东交通大学

毕业设计（论文）

题目：基于 Java 的移动营业厅应用系统的设计与实现

学 院：	<u>软件学院</u>		
	<u>软件工程（杰普）</u>		
专 业：	<u>软件工程</u>	班 级：	<u>2016-1</u>
学生姓名：	<u>王文钊</u>	学 号：	<u>20162111001000222</u>
指导教师：	<u>胡林峰、于炎</u>	完成日期：	<u>2020 年 5 月 8 日</u>

毕业设计（论文）诚信声明

本人郑重声明：所呈交的毕业设计（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。就我所知，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和撰写的研究成果，也不包含为获得华东交通大学或其他教育机构的学位或证书所使用过的材料。

如在文中涉及抄袭或剽窃行为，本人愿承担由此而造成的一切后果及责任。

本人签名

导师签名

2020 年 5 月 8 日

华东交通大学毕业设计（论文）任务书

姓名	王文钊	学号	2016211001000222	毕业届别	2020	专业	软件工程
毕业设计（论文）题目		基于 Java 的移动营业厅应用系统的设计与实现					
指导教师	胡林峰、于炎	学历	研究生、本科	职称	讲师、高级软件工程师		

具体要求：

1、设计目的与内容

 设计目的：

 由于移动营业厅后台开发业务时没有一个很好的归结开发人员所接业务的管理，而且开发人员对各个项目上线情况不能够实现实时查询、告警。因此，设计一个基于开发人员的移动营业厅很有必要。

 移动营业厅应用系统面向后台开发人员，便于他们更好的管理自己手中的业务，以及对开发好的业务进行视察以便维护。

 设计内容：

 本网站的主要功能模块分为五块：

 系统设置管理、用户信息管理、部门信息管理、业务信息管理、日志信息管理。

2、设计要求和成果

 设计工具：

 eclipse, chrome, tomcat

 技术要求：

 采用 Java 作为后台开发语言,jsp 作为前台开发语言,tomcat 作为服务器,MySQL 作为数据库，进行开发。

 设计步骤：

- 1) 语言框架选用
- 2) 页面设计
- 3) 接口开发
- 4) 系统运行
- 5) 系统测试

 设计成果：

 一个面向开发人员和开发管理人员对用户的信息，部门信息，业务信息进行更改，新增的系统平台，以及毕业论文，源码等。

3、应收集的主要参考文献

[1]王智勇. 基于 Struts 的 J2EE 构架的移动营业厅应用系统设计与实现[J]. 中国管理信息化, 2013, 14(13):37-39.

[2]赵建华, 李楠, 王国双. 移动营业厅应用系统实用性[J]. 中小企业管理与技
术, 2013, 21(15):259-260.

[3]韩友君. 关于移动营业厅应用系统的应用方案分析[J]. 科技创新与应用, 2013, 28
(25):79-79.

[4]李建东. 移动营业厅应用系统可靠性设计[J]. 计算机安全, 2013, 7(4):79-82.

[5]许平格. 数据库管理系统中查询优化的设计与实现[D]. 浙江：浙江大学, 2005.

进度安排：

	设计各阶段内容	时间	%
1	选题和资料收集阶段	第 7 学期 18~20 周	5
2	分析计划阶段	第 8 学期 1~2 周	10
3	设计阶段	第 8 学期 3~5 周	25
4	实现和测试阶段	第 8 学期 6~9 周	40
5	毕业论文写作、查重、答辩	第 8 学期 10~14 周	20
6	英文资料翻译	第 8 学期第 4 周之前	

指导教师签字：

2020 年 1 月 2 日

题目发出日期	2020.1.3	设计（论文）起止时间	2020.1.6-2020.6.7
--------	----------	------------	-------------------

学院意见：

同意发布题目

毕业设计领导小组组长签章

华东交通大学毕业设计（论文）开题报告书

课题名称	基于 Java 的移动营业厅应用系统的设计与实现				
课题来源	导师拟定	课题类型	BY	导师	胡林峰、于炎
学生姓名	王文钊	学号	2016211001000222	专业	软件工程

一、开题报告内容

1、目的和意义

选题意义：

如今的时代是信息技术和互联网的时代，在这过去的几十年里，互联网进行着飞速的发展和不断的进步，在互联网高速公路建设的同时，人们的生活也发生了巨大的变化，自习近平主席提出互联网+的发展战略以来，越来越多的年轻人投身到互联网行业的发展和自主创业当中，为这个奔腾起飞的国家贡献着自己微薄的力量。

在互联网技术高速发展的同时，对于员工、业务等的管理成了重中之重，由于在公司员工的业务和信息方面难以管理，企业的管理工作通常需要依靠各部门之间的配合才能顺利完成，有时在沟通业务方面有很大的工时消耗，导致开发业务的时候效率低下，这之间所造成的时间上的损失就很巨大。

就拿我现在实习工作的这家公司举例：当局方提出一个需求的时候，很多时候不是直接给到开发人员手中，而是经过组长，经过导师，一层层到达我的手中，如果有这么一个平台，将需求描述、需求的开发人、提出人这些信息提供给平台，平台通知到个人，开发人员可以在页面上清楚的看到需求的提出人，最晚要完成需求的时间，就可以知道这个需求的紧急情况和需要开发的内容，而需求的提出人作为管理员可以在平台上对需求的后续进展进行跟踪和修改，按时检查项目的进展，管理员也可以对用户的所有信息，以及部门上司等进行设置，这大大方便了公司管理员上层对公司的员工的一个监管。

这时候就需要一款面向开发人员和管理人员的工具，便于开发人员和管理人员对业务进行查看、编辑和修改。

设计目的：

在我所在的实习企业中，很多时候开发人员的需求查看，并不方便，而且管理人员如果要对开发人员的开发工时进行统计也没有很好地解决方法，很多时候都是口头上评估，然后粗略的进行登记，但是作为新时代的开发人员和具有创新严谨精神的程序员职业，我们要对这些信息进行使人满意的一个应用，能够更准确的记录、修改和删除这些信息。

我开发的这款移动营业厅管理系统，主要是面向开发人员和开发管理人员，进入应用之后，可以很方便的管理用户信息和业务信息。

对以上提出的互联网公司都具有的难点，我本文所研究的一套移动营业厅管理系统，面对项目开发人员和管理人员，能够使他们对自己部门小组内的项目进展，需求状态，以及员工的信息进行更大更方便的了解和管理。

对促进小组之间的交流，项目开发的流畅度，以及可靠度有很大的提升，节省项目开发的时间，对公司的效益也有更远大的影响。

2、研究现状和发展趋势

国外目前的移动营业厅管理系统，由于互联网的应用早而且比之国内发展的更加广泛，所以国外的的营业厅管理系统功能更加的完善，且使用起来更加的得心应手，并且他们有很多的应用平台可以借鉴开发，在很多程度上都领先于国内的移动营业厅管理系统。

但随着国内的开发业务管理系统越来越多，且技术上的问题交流更加方便，有许多技术上的问题正在迫近于国外的技术，甚至有赶超的势头，但是在很多时候，时间上的差距是不可避免的，所有国内目前做的很好的移动营业厅管理系统凤毛麟角。

3、设计方案

设计工具：Eclipse，jsp，spring JDK1.8+，MySQL5.6+

移动营业厅系统两种不同角色类型的用户提供了其所需要的功能，分别是管理员和普通用户，针对不同的角色，移动营业厅系统分别构建了功能使用模块，分别是管理员的五个模块：用户信息管理模块、系统用户管理模块、部门信息模块、业务信息管理模块、系统日志管理模块，普通用户的：用户信息管理模块和业务信息查看模块。

4、进度安排

	设计各阶段内容	时间	%
1	选题和资料收集阶段	第 7 学期 18~20 周	5
2	分析计划阶段	第 8 学期 1~2 周	10
3	设计阶段	第 8 学期 3~5 周	25
4	实现和测试阶段	第 8 学期 6~9 周	40
5	毕业论文写作、查重、答辩	第 8 学期 10~15 周	20
6	英文资料翻译	自定	

5、主要参考文献

- [1]王智勇. 基于 Struts 的 J2EE 构架的移动营业厅应用系统设计与实现[J]. 中国管理信息化, 2013, 14(13):37-39.
- [2]赵建华, 李楠, 王国双. 移动营业厅应用系统实用性[J]. 中小企业管理与技术, 2013, 21(15):259-260.
- [3]韩友君. 关于移动营业厅应用系统的应用方案分析[J]. 科技创新与应用, 2013, 28(25):79-79.
- [4]李建东. 移动营业厅应用系统可靠性设计[J]. 计算机安全, 2013, 7(4):79-82.
- [5]许平格. 数据库管理系统中查询优化的设计与实现[D]. 浙江: 浙江大学, 2005.
- [6]刘姗. 一种改进的 MySQL 数据库访问控制的设计与实现[D]. 武汉: 华中科技大学, 2011.
- [7]王晓东. J2ee 架构的研究与应用[J]. 科技信息, 2013, 23(23):439-439.
- [8]廖胜军. 基于 SSH 整合架构的应用研究[D]. 武汉: 武汉理工大学, 2008.
- [9]胡元发. 基于 J2EE 架构的 SSH 组合框架的设计与应用[D]. 苏州: 苏州大学, 2006.
- [10]Ou Feng, ZOU Min and LI Xiao-Zhen. Survey on Java Technology Framework[J]. Computer Systems& Applications, 2012, 21(8):236-239.
- [11]Farias G, Cervin A, Dormido S, Esquembre F. Java Simulations of Embedded Control Systems[J]. Sensors, 2010, 10(9):8585-8603.

- [12]陈明志. 基于 SSH 的移动营业厅应用系统的设计[J]. 电子技术与软件工程,2013, 14(6):22-24.
- [13] 王晓东. J2EE 架构的研究与应用[J]. 科技信息, 2013, 23 (23) :439-439.
- [14] Ou Feng , ZOU Min and LI Xiao-Zhen. Survey on Java Technology Framework[J]. Computer Systems& Applications, 2012, 21(8):236-239.

二、方法及预期目的

1、研究方法（手段）：

本系统初步使用 ssm 作为后台框架，jsp 作为前端框架进行开发，mysql5.0 作为数据库。

数据通过自己制作和当前实习单位上所接触到的一些基本数据。

2、预期目的：

用户或者管理员能够使用账号密码登录进应用系统，并可以在应用系统中对自身的信息，管理员则可以对系统中所有用户的信息进行增加修改删除查询，同时对公司部门的信息，业务管理的信息，日志信息等进行管理。

三、指导老师意见

同意开题

指导教师签名：

日期：

2020.3.13

课题类型：（1）A—工程设计；B—技术开发；C—软件工程；D—理论研究；

（2）X—真实课题；Y—模拟课题；Z—虚拟课题

（1）、（2）均要填，如 AY、BX 等。此部分可以附页

华东交通大学毕业设计(论文)评阅书 (1)

姓名	王文钊	学号	2016211001000222	专业班级	软件工程	
毕业设计(论文)题目		基于 Java 的移动营业厅应用系统的设计与实现				
指导教师评分表:						
课程目标	毕业要求指标点	权重系数	评判标准及分值		得分	
目标 1	3.4 能够从社会、健康、安全、法律、文化及环境等制约因素角度论证软件系统实现方案的可行性。	0.5	理解研究的目的是和意义, 明确研究内容, 论点正确, 研究方案合理。	20		
	6.2 能够分析评价软件工程实践和软件工程领域中复杂工程问题的解决方案对社会、健康、安全、法律以及文化的影响, 以及这些制约因素对项目实施的影响, 并理解应承担的责任。	0.5				
目标 2	3.2 针对软件系统或模块, 能够运用软件工程方法, 对其进行需求分析、概要设计、详细设计、编码与测试。	0.5	系统分析、设计、实现方法合理, 能实现预期功能, 有一定难度, 工作量满足要求。	30		
	3.3 能够在软件系统设计或开发过程中具有优选和追求创新设计方案的态度和意识。	0.2				
	7.2 能够理解复杂软件工程问题的实践活动对环境和社会可持续发展的影响, 评价软件产品周期中可能对人类和环境造成的损害和隐患。	0.3				
目标 3	10.1 熟练掌握一门外语, 能就专业问题, 在跨文化背景下进行沟通和交流。	0.5	论文内容正确, 结构严谨, 格式规范, 文献翻译数量质量符合要求	30		
	10.3 具备良好的口头沟通和书面表达的能力, 能准确陈述和表达自己的观点, 能够撰写专业文档, 与业界同行及社会公众进行沟通和交流。	0.5				
目标 4	11.2 理解多学科复杂工程问题的知识融合理念, 能够将管理学、工程经济学方法运用到软件工程项目管理活动。	1	能选用合适的软件技术和工具解决应用领域实际问题, 对前人工作有改进或有独特见解, 有一定应用价值。	20		
合计:				100		
指导教师评语:						
指导教师签字: 2020 年 月 日						

华东交通大学毕业设计(论文)评阅书 (2)

姓名	王文钊	学号	2016211001000222	专业班级	软件工程
毕业设计(论文)题目		基于 Java 的移动营业厅应用系统的设计与实现			
评阅人评分表:					
课程目标	毕业要求指标点	权重系数	评判标准及分值		得分
目标 1	3.4 能够从社会、健康、安全、法律、文化及环境等制约因素角度论证软件系统实现方案的可行性。	0.5	理解研究的目的是意义, 明确研究内容, 论点正确, 研究方案合理。	20	
	6.2 能够分析评价软件工程实践和软件工程领域中复杂工程问题的解决方案对社会、健康、安全、法律以及文化的影响, 以及这些制约因素对项目实施的影响, 并理解应承担的责任。	0.5			
目标 2	3.2 针对软件系统或模块, 能够运用软件工程方法, 对其进行需求分析、概要设计、详细设计、编码与测试。	0.5	系统分析、设计、实现方法合理, 能实现预期功能, 有一定难度, 工作量满足要求。	30	
	3.3 能够在软件系统设计或开发过程中具有优选和追求创新设计方案的态度和意识。	0.2			
	7.2 能够理解复杂软件工程问题的实践活动对环境和社会可持续发展的影响, 评价软件产品周期中可能对人类和环境造成的损害和隐患。	0.3			
目标 3	10.1 熟练掌握一门外语, 能就专业问题, 在跨文化背景下进行沟通和交流。	0.5	论文内容正确, 结构严谨, 格式规范, 文献翻译数量质量符合要求	30	
	10.3 具备良好的口头沟通和书面表达的能力, 能准确陈述和表达自己的观点, 能够撰写专业文档, 与业界同行及社会公众进行沟通和交流。	0.5			
目标 4	11.2 理解多学科复杂工程问题的知识融合理念, 能够将管理学、工程经济学方法运用到软件工程项目管理活动。	1	能选用合适的软件技术和工具解决应用领域实际问题, 对前人工作有改进或有独特见解, 有一定应用价值。	20	
合计:				100	
<p>评阅人签字:</p> <p>2020 年 月 日</p>					

华东交通大学毕业设计(论文)评阅书 (3)

姓名	王文钊	学号	2016211001000222	专业班级	软件工程
毕业设计(论文)题目		基于 Java 的移动营业厅系统的设计与实现			
答辩小组评分表:					
课程目标	毕业要求指标点	权重系数	评判标准及分值		得分
目标 1	3.4 能够从社会、健康、安全、法律、文化及环境等制约因素角度论证软件系统实现方案的可行性。	0.5	论点正确, 研究方案合理	20	
	6.2 能够分析评价软件工程实践和软件工程领域中复杂工程问题的解决方案对社会、健康、安全、法律以及文化的影响, 以及这些制约因素对项目实施的影响, 并理解应承担的责任。	0.5			
目标 2	3.2 针对软件系统或模块, 能够运用软件工程方法, 对其进行需求分析、概要设计、详细设计、编码与测试。	0.5	设计合理, 难易程度、工作量满足要求, 能实现预期功能	30	
	3.3 能够在软件系统设计或开发过程中具有优选和追求创新设计方案的态度和意识。	0.2			
	7.2 能够理解复杂软件工程问题的实践活动对环境和社会可持续发展的影响, 评价软件产品周期中可能对人类和环境造成的损害和隐患。	0.3			
目标 3	10.1 熟练掌握一门外语, 能就专业问题, 在跨文化背景下进行沟通和交流。	0.5	论文内容正确, 结构严谨, 格式规范, 文献翻译数量质量符合要求。答辩准备充分, 时间把握合理, 回答问题有理论依据, 思路清晰, 表达准确	30	
	10.3 具备良好的口头沟通和书面表达的能力, 能准确陈述和表达自己的观点, 能够撰写专业文档, 与业界同行及社会公众进行沟通和交流。	0.5			
目标 4	11.2 理解多学科复杂工程问题的知识融合理念, 能够将管理学、工程经济学方法运用到软件工程项目管理活动。	1	能选用合适的软件技术和工具解决应用领域实际问题, 对前人工作有改进或有独特见解, 有一定应用价值	20	
合计:				100	
答辩小组组长(签字): 2020 年 月 日					

华东交通大学毕业设计（论文）答辩记录

姓名	王文钊	学号	2016211001000222	专业班级	软件工程
毕业设计(论文)题目	基于 Java 的移动营业厅应用系统的设计与实现				
答辩组成员（签字）：					
答辩记录：					
记录人（签字）： 2020 年 月 日					

华东交通大学毕业设计(论文)成绩

姓名	王文钊	学号	2016211001000222	专业班级	软件工程		
毕业设计(论文)题目	基于 Java 的移动营业厅应用系统的设计与实现						
答辩委员会意见:							
同意以上评定, 根据前面三项得分, 按 25:15:60 的比例评定总成绩为							
<table border="1"><tr><td>等级</td><td></td></tr></table>						等级	
等级							
进入二次答辩 (毕业设计成绩由二次答辩的最终成绩决定)							
<input type="checkbox"/> 评优组							
<input type="checkbox"/> 合格组							
答辩委员会主任签字:							
2020 年 月 日 (学院公章)							

注: “等级”用优秀、良好、中等、及格、不及格五级制。优秀: 90~100 分; 良好: 80~89 分; 中等: 70~79 分; 及格: 60~69 分; 不及格: 60 分以下。

基于 java 的移动营业厅应用系统的设计与实现

摘 要

在二十一世纪二十年代，互联网科技已经发生了令人动容的改变，许多互联网计算机公司丛生而起的今天，就算在不同的互联网计算机有巨大影响力的公司中，管理也成了让人头疼的、需要解决的问题，不论是对人员的管理，对业务的管理，对部门的管理，对系统日志的管理，都需要一个移动营业厅的管理平台去统一管理这些信息，让高层管理人员，类似于经理、组长，业务提起人等等，这些角色能够更好的对公司的员工和信息进行快捷，有效的全方面管理。本系统结合了上述的需求，给管理人员和业务开发人员提供了一个分为五个模块、十一个小模块的人员业务管理平台——基于 java 的移动营业厅应用系统。

本系统采用 Java 作为后台开发语言，Tomcat 作为 web 服务器，Jsp 作为前台开发语言，Ssm 作为后台开发框架，Mysql 作为数据库进行开发。

目前，已经完成的主要信息管理模块和主要有这五个：用户信息管理模块,系统信息管理模块,部门信息管理模块,业务信息管理模块,系统日志信息管理模块。公司的系统管理员通过这些信息管理模块,使得部门的用户和系统管理员可以对项目开发人员信息、部门信息、日志信息进行实时有效的管理。在本篇文章中结合我所学习的编程技术，以及查阅许多管理平台方面的资料，简洁且清晰的介绍了移动营业厅如何实现需求管理功能，以及用户管理功能。

关键词:移动营业厅;Tomcat;SSM; Mysql;Tomcat

Design and Implementation of Mobile Business Hall Application System Based on Java

Abstract

In the twenties of the 21st century, Internet technology has changed a lot. Today, many Internet computer companies are growing up. Even in different Internet computer companies with great influence, management has become a headache and need to be solved. No matter how to manage personnel, business, Department, system, etc Log management needs a management platform of mobile business hall to manage these information in a unified way, so that senior managers, like managers, team leaders, business promoters, etc., can better manage the company's employees and information quickly and effectively. Combined with the above requirements, this system provides a personnel business management platform, which is divided into five modules and eleven small modules, which is based on Java mobile business hall application system.

This system uses Java as the background development language, Tomcat as the web server, JSP as the foreground development language, SSM as the background development framework, MySQL as the database for development.

At present, the main information management modules that have been completed include user information management module, system information management module, department information management module, business information management module and system log information management module. Through these information management modules, the system administrator of the company can manage the information of project developers, departments and logs in real time and effectively. In this article, combined with the programming technology I have learned, as well as consulting many information about the management platform, it briefly and clearly introduces how the mobile business hall realizes the demand management function and user management function.

Key words: Mobile Business Hall;SSM;Mysql;Tomcat

目 录

1 绪论	1
1.1 研究的背景和意义	1
1.1.1 选题的背景	1
1.1.2 国内外研究现状	1
1.1.3 课题研究目的及意义	1
1.2 系统目标	1
2 需求分析	3
2.1 功能需求	3
2.1.1 功能模块	3
2.1.2 功能需求描述	4
2.2 非功能需求	4
2.2.1 外部接口需求	4
2.2.2 其它需求	4
3 系统概要设计	5
3.1 系统主要开发流程	5
3.2 主要营业厅功能模块设计	5
4 数据库设计	8
4.1 概念结构设计	8
4.1.1 实体图	8
4.1.2 移动营业厅系统 E-R 图	9
4.2 逻辑结构设计	10
5 系统详细设计	12
5.1 密码信息设置模块实现	12
5.1.1 密码信息设置模块详细算法描述	12
5.1.2 密码信息设置模块程序流程图	12
5.1.3 密码信息设置模块关键类说明	12
5.1.4 密码信息设置模块界面	12
5.2 账号信息设置模块实现	13
5.2.1 账号信息设置模块详细算法描述	13
5.2.2 账号信息设置模块程序流程图	13
5.2.3 账号信息设置模块关键类说明	13
5.2.4 账号信息设置模块界面	13
5.3 用户信息管理模块实现	14
5.3.1 用户信息管理模块详细算法描述	14
5.3.2 用户信息管理模块程序流程图	14
5.3.3 用户信息管理模块关键类说明	15
5.3.4 用户信息管理模块界面	16
5.4 部门信息管理模块实现	17
5.4.1 部门信息管理模块详细算法描述	17
5.4.2 部门信息管理模块程序流程图	17
5.4.3 部门信息管理模块关键类说明	18
5.4.4 部门信息管理模块界面	18

5.5 业务信息管理模块实现.....	18
5.5.1 业务信息管理模块详细算法描述.....	18
5.5.2 业务信息管理流程图.....	18
5.5.3 业务信息管理模块关键类说明.....	19
5.5.4 业务信息管理模块界面.....	20
5.6 日志信息删除模块实现.....	21
5.6.1 日志信息删除模块详细算法描述.....	21
5.6.2 日志信息删除模块程序流程图.....	21
5.6.3 日志信息删除模块关键类说明.....	21
5.6.4 日志信息删除模块界面.....	21
6 程序测试与评价.....	23
7 总结和展望.....	30
7.1 总结.....	30
7.2 展望.....	30
8 谢 辞.....	31
参考文献.....	32
附录 A 外文翻译—原文部分.....	34
附录 B 外文翻译—译文部分.....	37
附录 C 软件使用说明书.....	40

1 绪论

1.1 研究的背景和意义

1.1.1 选题的背景

在大型的互联网计算机企业里，由于在公司员工的业务和信息方面难以管理，企业的管理工作通常需要依靠各部门之间的配合才能顺利完成，有时在沟通业务方面有很大的工时消耗，导致开发业务的时候效率低下，这之间所造成的时间上的损失就很巨大。

就拿我现在实习工作的这家公司举例：当局方提出一个需求的时候，很多时候不是直接给到开发人员手中，而是经过组长，经过导师，一层层到达我的手中，如果有这么一个平台，将需求描述、需求的开发人、提出人这些信息提供给平台，平台通知到个人，开发人员可以在页面上清楚的看到需求的提出人，最晚要完成需求的时间，就可以知道这个需求的紧急情况和需要开发的内容，而需求的提出人作为管理员可以在平台上对需求的后续进展进行跟踪和修改，按时检查项目的进展，管理员也可以对用户的所有信息，以及部门上司等进行设置，这大大方便了公司管理员上层对公司的员工的监管。

这时候就需要一款面向开发人员和管理人员的工具，便于开发人员和管理人员对业务进行检查、编辑和修改。

本营业厅管理系统，是基于 Java 语言，apache-tomcat 服务器、SSM 框架、Jsp 框架开发的一套 web 移动营业厅开发管理系统,提供业务开发进展,它能够使项目开发人员和项目管理人员对人员的信息、业务的信息、部门的信息进行有效快速的修改和查看。促进需求和开发人员以及需求提出人员的交流，提高工作效率。

1.1.2 国内外研究现状

国外目前的移动营业厅管理系统，由于互联网的应用早而且比之国内发展的更加广泛，所以国外的的营业厅管理系统功能更加的完善，且使用起来更加的得心应手，并且他们有很多的应用平台可以借鉴开发，在很多程度上都领先于国内的移动营业厅管理系统。

但随着国内的开发业务管理系统越来越多，且技术上的问题交流更加方便，所有许多技术上的问题正在迫近于国外的技术，甚至有赶超的势头，但是在很多时候，时间上的差距是不可避免的，所以国内目前做的很好的移动营业厅管理系统凤毛麟角。

1.1.3 课题研究目的及意义

在我所在的实习企业中，很多时候开发人员的需求查看，并不方便，而且管理人员如果要对开发人员的开发工时进行统计也没有很好的解决方法，很多时候都是口头上评估，然后粗略的进行登记，但是作为新时代的开发人员和具有创新严谨精神的程序员职业，能够更准确的记录、修改和删除这些信息。

本文所研究的一套移动营业厅管理系统，面对项目开发人员和管理人员，能够使他们对对自己部门小组内的项目进展，需求状态，以及员工的信息进行更大更方便的了解和管理。

对促进小组之间的交流，项目开发的流畅度，以及可靠度有很大的提升。

1.2 系统目标

设计一个基于 JAVA+JSP 的移动营业厅系统，主要面向后台开发人员，拥有普通用户和管理员用户两大角色，业务流程包括：业务查看、业务新增、业务修改、用户信息修改、日志信息查看等

本系统采用 Java 作为后台开发语言，Tomcat 作为 web 服务器，Jsp 作为前台开发语言，Ssm 作为后台开发框架，Mysql 作为数据库进行开发。

2 需求分析

随着信息技术的飞速发展，使用信息管理系统来管理学生信息已成为一种必然趋势。在开发一个系统前，首先要做的是充分了解系统需求，包括系统使用对象、需要的功能、再依据系统需求选择明确的开发方向。据了解，我们公司目前使用的管理系统中，并不具有需求管理，和日志管理功能。因此，为了跟上时代发展的脚步，同时提升公司员工之间的信息交流，和项目开发流畅度，本系统结合公司当前业务需求以及将来的规划，打造一个高质量，安全，规范的移动营业厅的信息需求管理系统。

通过系统分析，可以得出本系统主要使用人员为公司管理人员和开发人员。

管理人员作为管理用户，开发人员作为普通用户。本系统主要分为前台和后台，要求系统能整体满足日常所需功能、前台页面简介美观、系统易于操作，后台能够安全稳定的存储大量数据。

更具需求最终选用 Java+Eclipse+Mysql 等工具和技术进行本系统的开发。

2.1 功能需求

在大体了解开发本系统的基本需求后，在具体设计本系统时，应尽可能详细了解系统每个模块的功能需求，开发一个便于操作的系统。系统具有的功能如下。

- 1) 登录系统时需要输入正确的账号、密码、用户类型。
- 2) 管理员用户具有管理权限，可以根据需要实时修改、增加、删除用户，部门等信息。并可以赋予普通用户管理员权限。
- 3) 应满足对用户信息的录入。
- 4) 应满足部门信息的录入修改等功能。
- 5) 应满足系统日志的查看。
- 6) 应满足业务信息的查看和修改增加。

2.1.1 功能模块

本系统具有的主要模块包括：系统设置管理模块、用户信息管理模块、部门信息管理模块、业务信息管理模块、日志信息管理模块。

- 1) 系统设置管理：对系统用户的自身信息包括账号和密码进行管理。
- 2) 用户信息管理：对所有本平台用户的信息信息，进行修正，改错，增添，删除操作，主要面向管理员。
- 3) 部门信息管理：对部门信息的管理，如果有新增的部门或者部门进行更改姓名变动时，可以在部门信息管理模块中更改。
- 4) 业务信息管理：普通用户登录进入主页后只能看到业务信息查看模块，而管理员登录系统进入时,不仅能够看到所有业务管理员的信息,还拥有权限可以对相应的信息做一些处理。
- 5) 日志信息管理：对用户或者管理员操作 YiDong 页面时，记录下来的一些操作的保存。

2.1.2 功能需求描述

1) 模糊查询功能

用户在有搜索框的界面，例如用户信息管理页面，能够对展示区域的数据进行模糊匹配查询功能，在对数据查询时讲关键字家带到请求数据中。后台解析数据向队形数据库表获取数据后，对相应的用户 id、姓名、或者真实姓名进行模糊查询。后台在进行模糊查询后，将查询到的数据返回给前台页面进行展示。

2) 分页功能

用户在对数据进行查询时可能会查询到过多的数据，于是设计一个分页功能可以更好的展示数据，优化用户的体验。分页功能能够根据当前页面的页码与当前页的数据容纳量去数据库获取相应数据，并渲染系统前台界面。

2.2 非功能需求

2.2.1 外部接口需求

本系统暂时未使用外部接口。

2.2.2 其它需求

稳定性：系统在运行过程中面对各式各样的操作人员，如果在用户操作的过程中经常出现问题会给用户带来较大的困扰，所以必须提升系统的稳定性。系统应当对一些基本错误做出自适应的处理，如用户输入数据出现错误，系统应该给用户反馈相应的提示，保证部分功能出现问题导致系统崩溃；

安全性：由于系统中存储着大量的用户个人信息以及相关业务操作的数据，所以系统必须拥有完善的安全机制，保证系统根据用户的权限信息来渲染操作界面，避免越权操作的情况出现。同时系统必须对病毒攻击和网络攻击做出有针对性的措施来保护系统安全。

3 系统概要设计

3.1 系统主要开发流程

营业厅系统主要开发流程设计：

系统开发从对需求的收集开始，通过公司管理员和普通员工的业务方面的需求收集，管理员需要对开发人员和下级管理人员的信息进行有效管理，普通员工需要对自己项目的进展进行查看。

本系统所使用的数据，大致来自我目前实习的公司中所接触到的项目以及自创的用户信息。

应用测试：在测试方面，选用的黑盒测试，对每一个小模块，进行一个个单元测试，举出一些常见的测试用例，测试系统的基本功能是否完善。

本系统主要是用于收集管理电脑和移动终端营业厅的应用的信息。改系统功能支持已注册营业厅的用户对各项信息的收集和查询,系统开发人员对自己手中的应用需求的掌握和了解,可以看到需求的开发时间，还有多少天结束，需要多少工时完成，最大程度上减轻了管理者对需求管理的负担。

3.2 主要营业厅功能模块设计

本网站的主要功能模块分为五块：

- 1)系统设置管理：对系统用户的自身信息包括账号和密码进行管理。
- 2)用户信息管理：对所有本平台用户的信息信息，进行修正、改错、增添、删除操作，主要面向管理员。
- 3)部门信息管理：对部门信息的管理，如果有新增的部门或者部门进行更改姓名变动时，可以在部门信息管理模块中更改。
- 4)业务信息管理：普通用户登录进入主页后只能看到业务信息查看模块，而管理员登录系统进入时，不仅能够看到所有业务管理员的信息，还拥有权限可以对相应的信息做一些处理。
- 5)日志信息管理：对用户或者管理员操作 YiDong 页面时，记录下来的一些操作的保存。

营业厅系统主要功能模块结构：

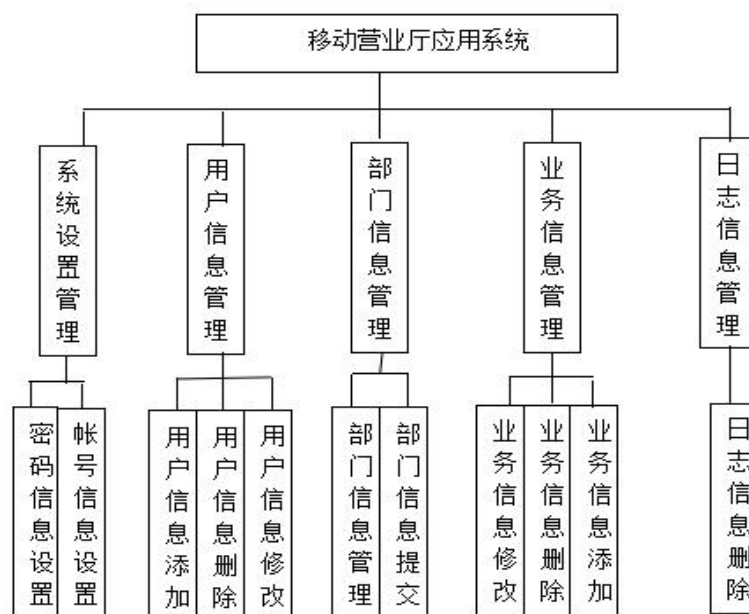


图 3-1 系统功能结构图

各个模块基本功能：

1) 密码信息设置模块：用户登录之后，在密码信息设置模块中，通过输入当前密码，再输入需要修改的新设置密码，再次输入一个上一个输入框中相同的密码，这一步是防止用户一次输入密码错误，往后登录失败，修改自己的一个登录帐号和密码。除此之外，经常有规律地管理和设置登录密码和帐号，能够使系统和用户个人信息的保密性和安全性得到保障。

2) 账号信息设置模块：用户登陆之后，在账号信息设置模块中，可以随时修改自己的基本身份信息，例如您的姓名、年龄、手机号码等。

3) 用户信息添加模块：用户进入用户信息管理模块中，通过点击新增按钮，新增用户，管理员可以通过这个模块对新开发人员和领导的账号进行新增，相当于注册功能。

4) 用户信息删除模块：对其他用户注册信息进行点击自动勾选删除个人信息模块按钮，可以对已经自动储存到现在的所有其他用户注册个人信息全部进行自动勾选删除，也就是用户可以通过多次勾选，实现批量删除，当一些员工离职之后可以轻松地管理他们的信息。

5) 用户信息修改模块：用户姓名输入一个需要姓名信息修改的用户姓名，年龄，部门，电话等信息，随后用户可以点击下方的修改信息按钮，可以对已经注册存在的平台用户所提供的信息是否进行了修改，达到对信息的完全掌握。

6) 部门信息管理模块：营业厅管理员进入部门的信息管理模块，就能够对部门信息进行一些基本操作，比如说增添部门，删除修正部门，例如目前有业务部、技术部、项目部等，后续根据生产上业务的增加，陆续增加部门信息。

7) 部门信息提交模块：管理员输入需要添加的部门信息，在部门信息提交模块中，可以对部门信息进行新增，以达到对部门新增的目的。

8) 业务信息修改模块：在业务信息管理模块，用户可以修改现有业务信息，例如可以修改项目工时、项目提出人、项目开始结束时间等等。

9) 业务信息删除模块：在业务信息删除模块，用户可以删除部门信息，当这个业务已经挤压很久了，或者早已完成，就可以对他们单个删除或者批量删除。

10) 业务信息添加模块：用户在业务信息添加模块中，可以新增业务信息，实现业务的实时更新。

11) 日志信息删除模块：用户所有的正确操作和错误操作都可以在这里面记录到并反馈给管理人员，用户已经进入了日志信息的删除管理模块,可以删除冗余的日志信息。

4 数据库设计

4.1 概念结构设计

本系统的开发中主要设置了一下几个实体：

用户实体：主要包含了用户 id，账号、密码、真实名、性别、年龄、电话、部门、加入时间和身份（管理员和普通用户）。

部门实体：部门 id，部门名。

日志实体：日志 id、操作用户、操作、操作时间。

业务实体：业务 id、部门、提出人、开发人、开发时间、工时、预计工时、实际工时、状态。

4.1.1 实体图

1) 用户实体图

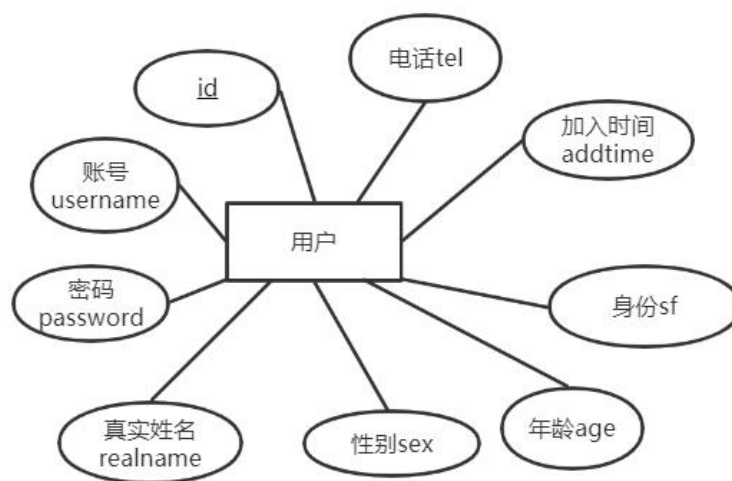


图 4-1 用户实体图

2) 部门实体图



图 4-2 部门实体图

3) 业务实体图

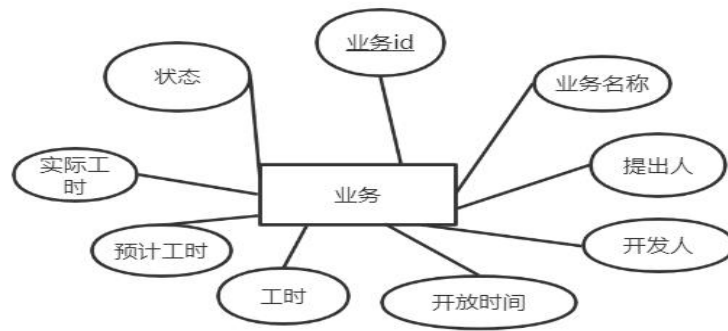


图 4-3 业务实体图

4) 日志实体图

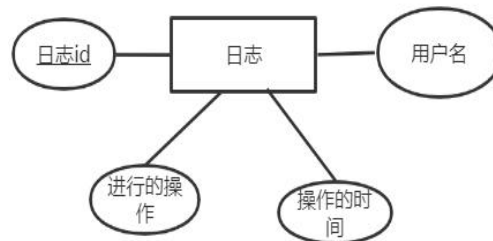


图 4-4 日志实体图

4.1.2 移动营业厅系统 E-R 图

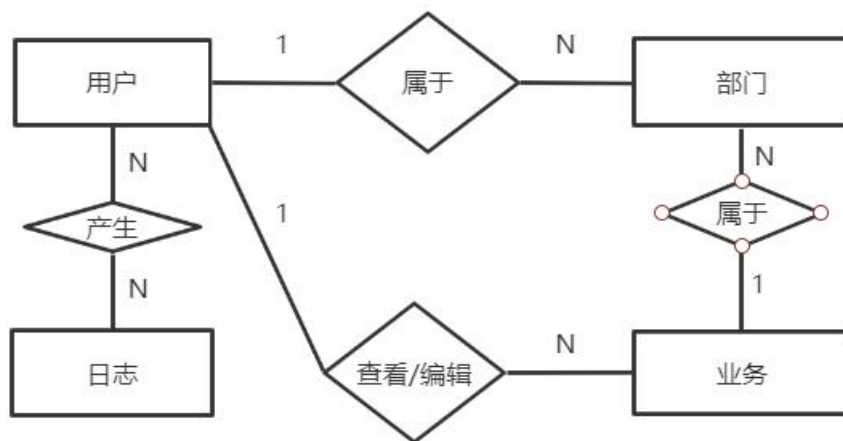


图 4-5 整体 E-R 图

1) 普通用户和管理员用户和部门关系 E-R 图:



图 4-6 普通用户和管理员用户和部门关系 E-R 图

2) 用户和日志部门 E-R 图:

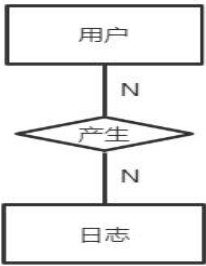


图 4-7 用户和日志部门 E-R 图

3) 用户和业务 E-R 图:



图 4-8 用户和业务 E-R 图

4.2 逻辑结构设计

通过对本系统的基本功能需求分析，可以看出本系统的结构相对来说比较简洁明了，按照我所设定好的移动营业厅应用系统的所有功能要求，主要设计了以下四个数据库表：

1) 用户信息表

表 4-1 用户信息表

列 名	数据类型	长度	说 明
id	int	4	编号
name	varchar	50	管理员 ID
password	varchar	50	管理员密码 password
Realname	varchar	50	真实名称 realname
Sex	varchar	50	用户性别 sex
age	varchar	50	用户年龄 age
address	varchar	50	用户部门 address
tel	varchar	50	用户电话 tel
addtime	varchar	50	添加时间 addtime

2) 部门信息表

表 4-2 部门信息表

列 名	数据类型	长度	说 明
id	int	4	编号 id
Mc	varchar	50	名称 mc

3) 业务信息表

表 4-3 业务信息表

列 名	数据类型	长度	说 明
id	int	4	编号 id
Mc	varchar	50	名称 mc
Tcr	varchar	50	提出人 tcr
Kfr	varchar	50	开发人 kfr
Ksj	int	4	开始时间 ksj
Gs	varchar	50	工时 gs
Esj	varchar	50	结束时间 esj
Sjgs	varchar	50	实际工时 sjgs
zt	varchar	50	状态 zt

4) 日志信息表

表 4-4 日志信息表

列 名	数据类型	长度	说 明
id	int	4	编号 id
Yh	varchar	50	用户 yh
Cz	varchar	50	操作 cz
Jg	varchar	50	时间 jg

5 系统详细设计

5.1 密码信息设置模块实现

5.1.1 密码信息设置模块详细算法描述

在现实使用中，往往有时因为安全等种种原因需要对用户自身的账号密码进行修改，在本模块中，通过用户输入原始密码，新密码，旧密码，调用接口、/admin/adminuapPwd 传入参数，与数据库信息进行匹配，匹配失败则返回原始密码错误，如果两次输入密码相同且新密码不为空，则返回修改成功，否则返回操作失败。

5.1.2 密码信息设置模块程序流程图

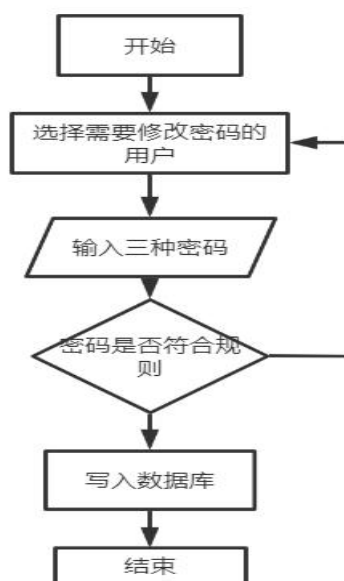


图 5-1 密码信息设置模块流程图

5.1.3 密码信息设置模块关键类说明

Admin admin 创建用户实体类，通过/admin/adminupPwd 传入参数，调

admin.setUsername(username)

admin.setPassword(oldpwd)

List<Admin> list = adminService.admincheckPwd(admin)

调用 adminService.admincheckPwd 方法得到列表 list

通过判断 list 大小是否为空判断旧密码是否存在数据库中，如果不存在则调用 Admin.setPassword(newpwd) 修改密码并写入数据库

5.1.4 密码信息设置模块界面

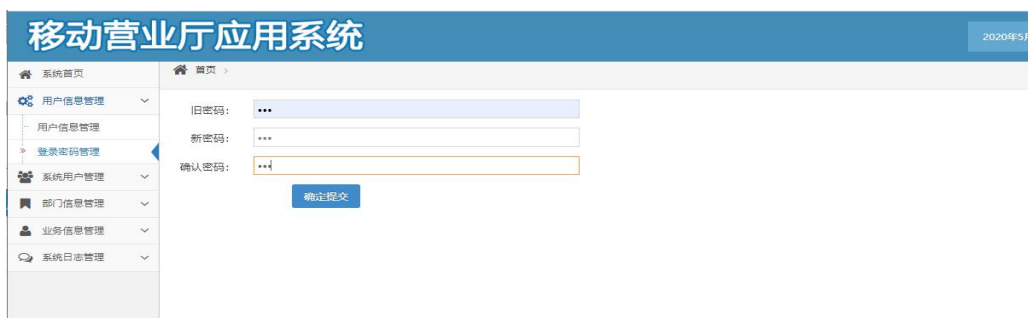


图 5-2 密码信息设置模块界面图

5.2 账号信息设置模块实现

5.2.1 账号信息设置模块详细算法描述

管理员要修改自己的, 真实姓名 `realname`, 年龄 `age`, 电话号码 `tel`, 部门 `address` 等信息, 通过调用 `/admin/adminupdateByUsername` 接口传入这些参数, 先获取到用户 `username`, 通过 `username` 定位到其他信息, 进行修改。

5.2.2 账号信息设置模块程序流程图

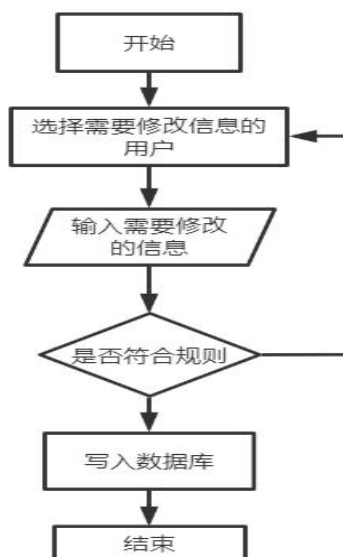


图 5-3 账号信息设置模块流程图

5.2.3 账号信息设置模块关键类说明

`Admin admin` 创建实体类

`/admin/adminupPwd` 传入参数

`admin.setUsername(username);` 获得用户名

`adminService.adminupdateByUsername(admin);`

再进入 `dao` 层 调用 `adminupdaeByUsername(reord(参数))`

动态传入, `sql` 对用户的账号信息需要修改的部分进行修改。

通过用户名修改用户信息

5.2.4 账号信息设置模块界面



图 5-4 账号信息设置模块界面图

5.3 用户信息管理模块实现

5.3.1 用户信息管理模块详细算法描述

新增：管理员进入用户添加模块（可以添加普通用户和管理员的消息），通过在页面上输入想要添加的用户信息，用户名 `username` 、密码 `password` 、年龄 `age` 、电话 `tel`、部门 `address` 等等信息，点击完成按钮，调用 `/admin/addAdmin` 接口，传入参数，调用方法对数据库用户表 `insert` 操作，达到新增用户的目的。

删除：有时用户的信息有错误或者用户离职过期，我们可以调用删除模块。来实现对用户信息的删除，通过调用 `/admin/del/{id}` 单个删除 和 `/admin/batchDel` 批量删除。

修改：用户通过点击编辑按钮，输入需要修改的用户信息，将参数通过 `/admin/updateAdmin` 传入后台，后台调用方法，将用户的信息更新进数据库中。

5.3.2 用户信息管理模块程序流程图

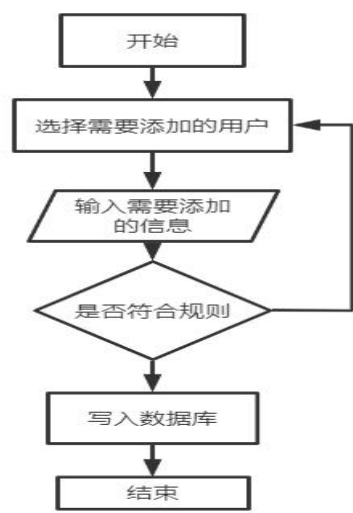


图 5-5 用户信息添加模块流程图

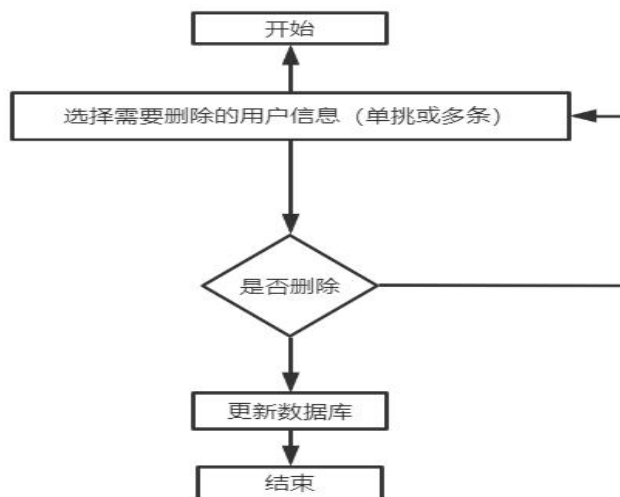


图 5-6 用户信息删除模块流程图

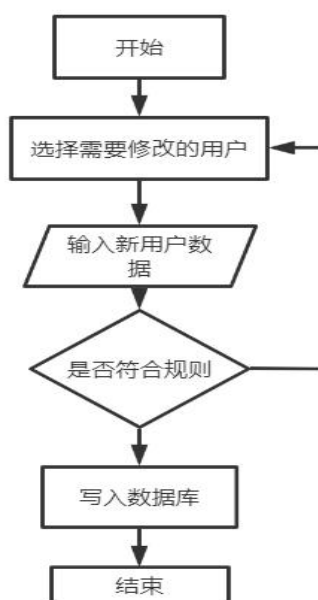


图 5-7 用户信息修改模块流程图

5.3.3 用户信息管理模块关键类说明

新增:

Admin admin 获取用户实体类, /admin/addAdmin 接口获得前台传入参数 String username=admin.getUsername(), String str = adminService.getSame(username)获取用户名并获取用户 id, adminService.insertSelective(admin);对用户表进行新增, 进入 dao 层, 调用 int insertSelective(Admin record); 将用户名传入数据库配置文件, 调用 jdbc 对数据库中的用户信息进行添加。

删除:

单个删除: /admin/del/{id} 前台传入需要删除的用户 id

adminService.deleteByPrimaryKey(id)后台调用 deleteByPrimaryKey 方法对用户全部信息删除。

批量删除：/admin/batchDel 前台传入需要删除的用户 id 组成的列表、

```
for(int i=0;i<chk_list.length;i++){
    adminService.deleteByPrimaryKey(Integer.parseInt(chk_list[i]));
}
```

后台通过遍历列表，调用 deleteByPrimaryKey 方法，达到批量删除的目的。
修改：

Admin admin 创建用户类
前台通过/admin/updateAdmin 接口，传入需要修改的参数值。
后台先通过调用 adminService.upgetSame(admin)方法，获取到用户 id
再判断是否存在，如果不存在就调 adminService.updateByPrimaryKey(admin)方法，
进行数据新增 update。

5.3.4 用户信息管理模块界面



图 5-8 用户信息添加模块界面图

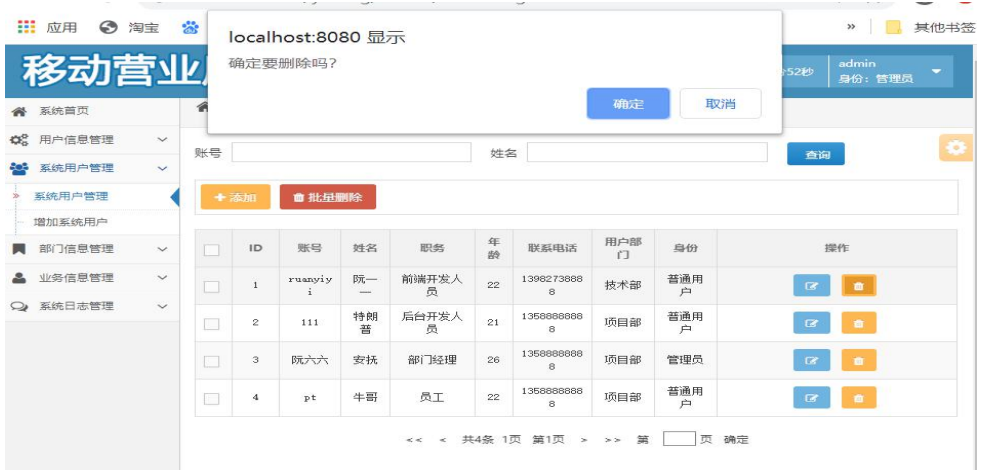


图 5-9 用户信息单个删除模块界面图



图 5-10 用户信息批量删除模块界面图



图 5-11 用户信息修改模块界面图

5.4 部门信息管理模块实现

5.4.1 部门信息管理模块详细算法描述

新增：用户进入部门管理模块新增部门子页面，输入需要新增的部门信息，点击提交，后台判断新输入部门名称是否重复，没有则写入数据库，有则返回选择页面。

5.4.2 部门信息管理模块程序流程图

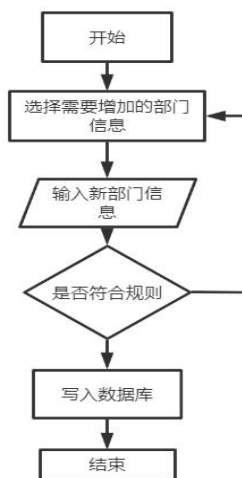


图 5-12 部门信息修改模块流程图

5.4.3 部门信息管理模块关键类说明

FL fl 部门实体类创建

前台通过调用： /fl/addMethod 将需要增加修改的部门数据传入。

后台通过调用： String mc=fl.getMc();

String str = flService.getSame(mc);方法对用户 id 进行 select

再调用 flService.insertSelective 方法将数据插入部门表。

5.4.4 部门信息管理模块界面



图 5-13 部门信息修改模块界面图

5.5 业务信息管理模块实现

5.5.1 业务信息管理模块详细算法描述

修改：业务信息包括 id、业务名称、提出人、开发人员、开始的时间、结束的时间、实际的工时、预计工时、业务状态、剩余天数。管理员可以通过调用修改接口 /yw/addMethod 接口对这些字段进行修改，以便管理员管理平台目前的业务。

删除：当有些业务挤压过久被取消或者业务已经完成了，可以通过调用单个删除 /yw/del/{id} 或者批量删除接口 /yw/batchDel 对业务信息进行删除，达到刊业务信息的目的。

添加：管理员进入添加业务信息页面，通过添加业务名称、提出人、开发的施工时间、开始周期施工时间、预计工时、结束周期施工时间、实际工时、业务状态，点击确定提交按钮，调用 /yw/addMethod 接口，调用 jdbc 对数据库业务表进行新增。

5.5.2 业务信息管理流程图

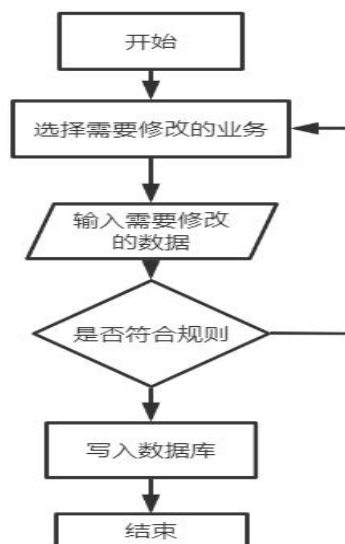


图 5-14 业务信息修改流程图

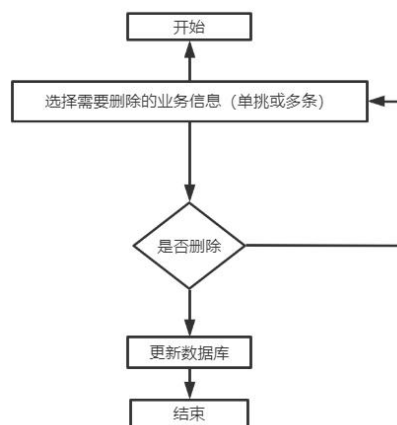


图 5-15 业务信息删除模块流程图

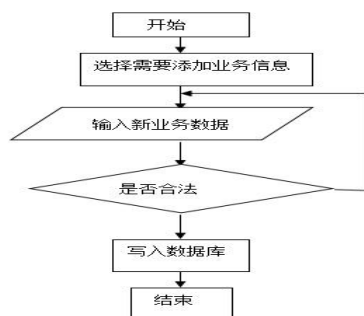


图 5-16 业务信息添加模块流程图

5.5.3 业务信息管理模块关键类说明

1) 修改:

Yw yw 声明业务实体类

```
String user = (String)session.getAttribute("user");
```

```
String method = request.getParameter("method");
```

获取到 method 中前台传入的"addyw"字段,发现没有 addyw 字段,则进入修

改操作中，调用 `ywService.updateByPrimaryKeySelective(yw)`接口，对业务表中字段进行修改。

2) 删除:

单个删除:

接口: `/yw/del/{id}` 前台将要删除的业务数据 `id` 传入后台
`@PathVariable("id") int id` 通过 `pathvariable` 方法获取 `url` 中的 `id`
`ywService.deleteByPrimaryKey(id);`调用 `deleteByPrimaryKey` 方法，根据 `id` 将业务数据删除。
通过 `new ComBean().comUp("insert into rz(yh,cz,sj) values('"+user+"','删除业务','"+date2+"')");`在日志表中写入记录

批量删除:

接口: `/yw/batchDel` 前台将管理员选中的业务 `id` 存入列表传入后台。
后台通过遍历列表 `chk_list`，将列表中的 `id` 遍历调用 `ywService.deleteByPrimaryKey(Integer.parseInt(chk_list[i]));`方法，达到批量删除的目的。

3) 增加:

`Yw yw` 声明业务实体类
`String user = (String)session.getAttribute("user");String method = request.getParameter("method");`
获取到 `method` 中前台传入的“`addyw`”字段，发现有 `addyw` 字段，则进入新增操作中，调用 `ywService.insertSelective(yw)`接口，对业务表中字段进行新增。

5.5.4 业务信息管理模块界面



图 5-17 业务信息修改模块界面图



图 5-18 业务信息删除模块界面图

图 5-19 业务信息添加模块界面图

5.6 日志信息删除模块实现

5.6.1 日志信息删除模块详细算法描述

当有许多积压的日志信息的时候，管理员可以通过调用单个删除`/rz/del/{id}`接口或`/rz/batchDel`批量删除接口对日志信息进行删除。

5.6.2 日志信息删除模块程序流程图

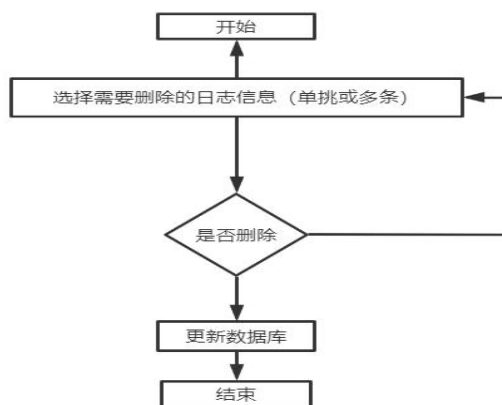


图 5-20 日志信息删除模块流程图

5.6.3 日志信息删除模块关键类说明

单个删除：

接口：`/rz/del/{id}` 前台将要删除的业务数据 `id` 传入后台
`@PathVariable("id") int id` 通过 `pathvariable` 方法获取 `url` 中的 `id`
`ywService.deleteByPrimaryKey(id);`调用 `deleteByPrimaryKey` 方法，根据 `id` 将业务数据删除。

批量删除：

接口：

`/rz/batchDel` 前台将管理员选中的业务 `id` 存入列表传入后台。

后台通过遍历列表 `chk_list`，将列表中的 `id` 遍历调用

`ywService.deleteByPrimaryKey(Integer.parseInt(chk_list[i]));`方法，达到批量删除的目的。

5.6.4 日志信息删除模块界面

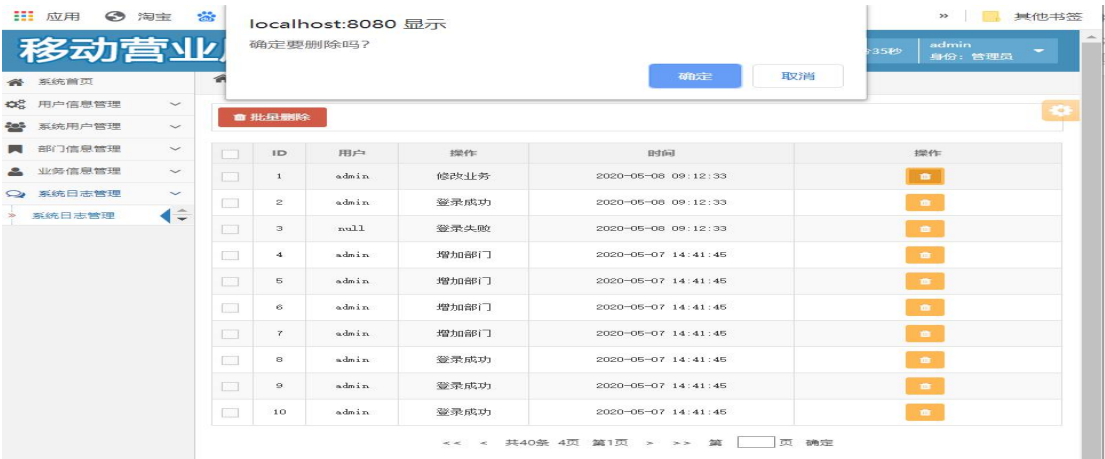


图 5-21 日志信息单个删除模块界面图

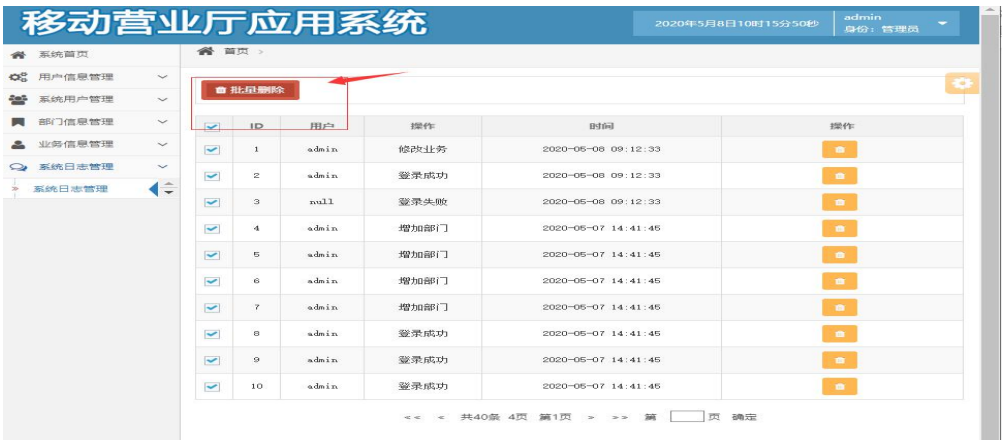


图 5-22 日志信息批量删除模块界面图

6 程序测试与评价

测试环境：电脑型号：华硕飞行堡垒。

操作系统：windows 10 ×64 家庭版。

英特尔处理器：inter core i5-6300HQ CPU @2.30 GHz 2.30 Ghz。

已安装的内存：12GB。

测试浏览器：谷歌浏览器 Google Chrome。

测试 java 环境：JDK1.7。

服务器：tomcat-7.0。

(1) 用户管理员/普通用户登录测试

为了确定登录模块不具有漏洞和错误，通过输入不同类型的账号、密码、用户身份对登录模块进行测试，对不同用户输入的场景进行全方面测试。

表 6-1 用户登录测试表

测试目的	测试操作程序流程	测试简单用例	预测结果	测试结果
用户登录	填写正确的用户名密码,选择管理员或者普通用户,点击首页的登录按钮进行登录	用户名: Schwarz 密码: schwarz 身份: 管理员	登录失败	弹出登录信息错误,如图 6-1 所示
用户登录	填写正确的用户名密码,但没有选择登录者的身份,再次点击登录按钮	用户名: schwarz 密码: iLOVEYOU1998829 身份: 无	登陆失败	弹出提示,请在身份列表中选择一行,如图 6-2 所示



图 6-1 登录失败账号密码错误图



图 6-2 未输入身份测试图

(2) 用户信息模块测试

为了检查用户进行用户信息模块测试的时候不会出现错误漏洞，所有通过输入各种情况下可能输入的用户信息进行测试，以确保模块的完整无误。

表 6-2 系统的用户信息管理测试表

测试项目	测试操作流程	测试用例	预测结果	测试结果
用户登录密码设置	用户管理员登陆密码设置登陆后，只需点击密码设置中的登陆密码设置按钮即可进入用户管理人员页面,输入旧的登陆密码,新密码,确认密码。	用户密码: Iloveyou1998 旧密码: iloveyou1998 新密码: Iwillbetheking 确认密码: Iwillbetheking	修改失败,弹出提示按钮	修改失败,原始密码错误。如图 6-3
用户登录密码设置	输入旧的登陆密码,新密码,确认密码。	用户密码: Iloveyou1998 旧密码: Iloveyou1998 新密码: Iwillbetheking 确认密码: iwillbetheking	修改失败	弹出两次输入的密码不匹配,如图 6-4 所示
用户登录密码设置	输入旧的登陆密码,新密码,确认密码。	用户密码: Iloveyou1998 旧密码: Iloveyou1998 新密码: Iwillbetheking 确认密码: Iwillbetheking	修改成功	修改成功



6-3 修改密码原始密码错误测试结果图



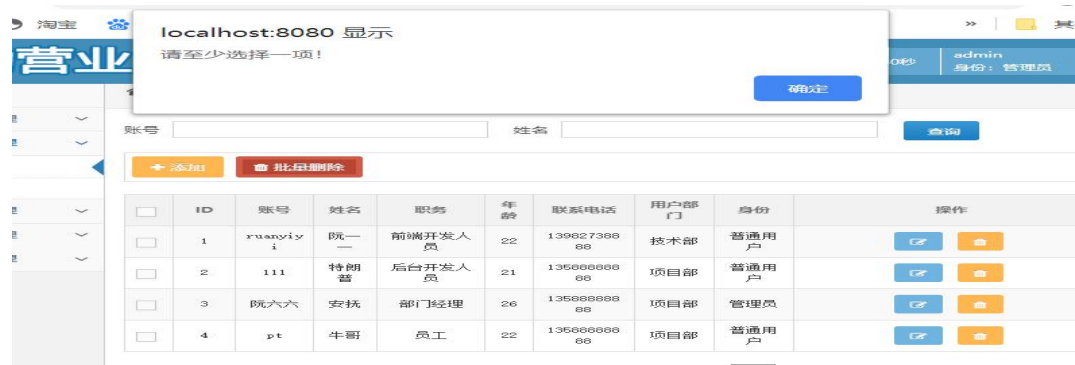
6-4 修改密码两次输入密码不一致测试结果图

(3) 系统用户信息模块测试

为了检查系统用户信息模块的完成度，对删除、新增这两个功能模块进行测试，通过输入各种用户可能输入的数据情况，对这两个模块进行测试。

表 6-3 系统的系统用户信息管理测试表

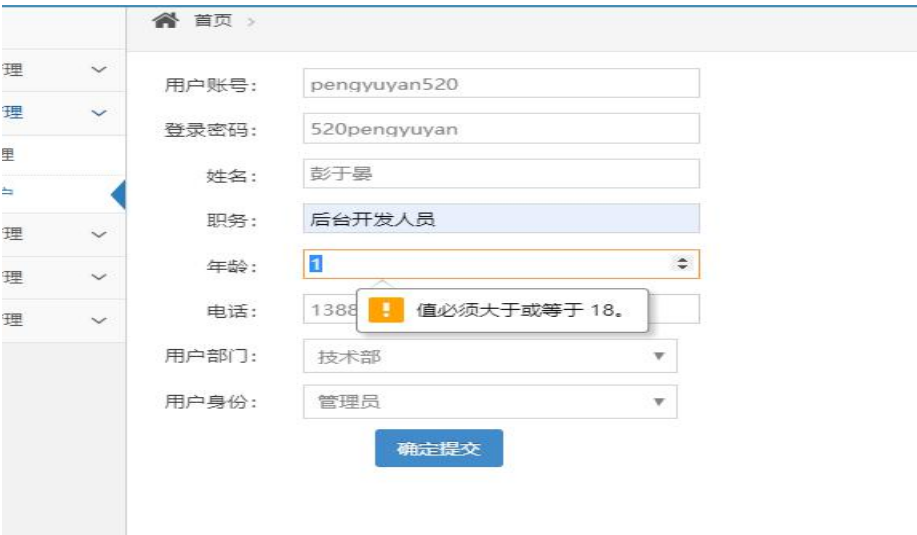
测试项目	测试操作流程	测试用例	预测结果	测试结果
系统用户管理删除模块	用户或者管理员进入这个模块中，确认需要删除的列名	未选中任何选项，点击批量删除	查询失败	删除失败，提示请填写内容，如图 6-5
系统用户管理新增模块	点击增加系统用户模块，进入修改页面。	空	新增失败，弹出提示按钮	修改失败，原始密码错误。如图 6-6
系统用户管理新增模块	点击增加系统用户模块，进入修改页面	用户账号：pengyuyan520 登陆密码：520pengyuyan 姓名：彭于晏 职务：需求开发人员 年龄：1 电话：13888888888 用户部门：业务部 用户身份：管理员 g	修改失败	修改失败，提示年龄必须大于 18 岁，如图 6-7



6-5 批量删除未选择测试结果图



6-6 新增系统用户失败图



6-7 新增系统用户：彭于晏 测试失败结果图

(4) 部门信息模块测试

为了测试部门信息模块的完成度，通过输入用户可能键入的信息，对模块进行全量的测试。

表 6-4 系统的部门信息管理测试表

测试项目	测试操作流程	测试用例	预测结果	测试结果
部门信息管理删除模块	用户或者管理员进入这个模块中,确认需要删除的列名	未选中任何选项,点击批量删除	查询失败	删除失败,提示请填写内容,如图 6-8
增加部门信息模块	点击增加增加部门信息模块,进入增加页面。	部门名称:信息技术部	新增成功	图新增部门:信息技术部成功 6-9



6-8 未选中信息删除失败测试图



6-9 新新增信息技术部成功图

(5) 业务信息管理模块测试

为了测试业务信息管理模块的完成度,最大程度提高用户的使用体验,通过模拟数据对删除、新增这两个模块进行具体全面的测试。

表 6-5 系统的业务信息管理测试表

测试项目	测试操作流程	测试用例	预测结果	测试结果
业务信息管理 删除模块	确认需要删除 的列名	选中 id=1, 业务名称名称 =kafka 集群搭建和数据 筛选 提出人：盛未见 开发人：陈朝花 开始时间：2020-05-01 结束时间：2020-05-12 实际工时：0 业务状态：正常 剩余天数：2	删除成功	提示您已 删除成功， 如图 6-10
增加业务信息 模块	用户管理员登 陆密码设置登 陆后，点击增加 增加业务信息 模块，进入增加 页面。	空	新增失败，弹 出提示按钮	新增失败， 如图 6-11
增加业务信息 模块	点击增加增加 业务信息模块， 进入增加页面。	业务名称：短信熔断接口 开发 提出人：高春发 开发人：王文钊 开始时间：2020-05-11 预计工时：14 结束时间：2020-5-16 实际工时：0 业务状态：正常	新增成功	图新增业 务：短信熔 断接口开 发 6-12



6-10 选中 kafka 和 cboss 业务删除成功测试图

业务应用系统

首页 >

业务名称:

提出人: ! 请填写此字段。

开发人:

开始时间:

预计工时:

结束时间:

实际工时:

业务状态:

6-11 新增业务信息 未输入内容测试图

业务名称

查询

+ 添加

批量删除

<input type="checkbox"/>	ID	业务名称	提出人	开发人	开始时间	预计工时	结束时间	实际工时	业务状态	剩余天数	操作
<input type="checkbox"/>	1	短信熔断接口开发	高春发	王文钊	2020-05-11	14	2020-05-28	0	正常	14	<div><div></div><div></div></div>

6-12 新增业务短信熔断接口开发测试图

7 总结和展望

7.1 总结

在设计基于 Java 的移动营业厅系统中，通过设计各个功能模块：登录模块、用户信息管理模块、系统用户管理模块、业务管理模块，我学习到了很多平时学习不到的知识和经验。特别是业务管理模块，因为很多代码实现和别的模块相关联，更加考验了我对系统架构的理解，了解了 SSM 框架的实现方式和他如何能够减少程序员开发的难度，还有在编码方面学会了如何更加简洁明了的编写代码，一些以前没有注意的编码规范，例如驼峰显示，字母大小写等，总的来说，这次的毕设项目编写过程让我学会了很多，讲专业知识转化为实践，在以后的工作学习中，我会充分利用这些经验，做一名更好的程序开发员。

7.2 展望

目前虽然应用基本功能都以实现，但是还是略显粗糙，有些功能模块设计的不严谨。

- ①登录模块，后续加入验证码登录模块，增加系统安全和严谨
- ②系统用户管理模块：有一些固定的字段写死，不可以随意修改。
- ③业务信息管理模块：后续快要到结束时间没有完成的业务，增加短信告警功能，达到强提醒的效果。
- ④界面优化：界面设计的不够简洁，后期要继续美化。
- ⑤密码修改模块，后期加入加密解密功能，增加系统的安全性。
- ⑥业务管理模块：界面设计的还是很粗糙，有些字段设置的不够全面具体。后期要加入更严谨的考虑，例如增加完成任务按钮，增加任务具体描述等。

8 谢 辞

论文搁笔之际，我首先要衷心的感谢恩师胡林峰老师的谆谆教诲。从最初的选题到最后的修改，胡老师都给了我很大的帮助和意见，让我在选题和论文的修改上并没有遇到很大的问题。导师的言传身教，让我不仅仅学习到了专业知识，还得到了严肃的学风刻苦的钻研精神，严谨的治学态度。

同时，感谢大学四年来陪伴着我成长的所有老师、同学们，你们的影响和教导将使我受益终生。

在往后的我将永远饱含这份感激之情，并以此激励自己，在以后的学习工作中，继续奋进，勇攀高峰。

参考文献

- [1] 王智勇. 基于 Struts 的 J2EE 构架的移动营业厅应用系统设计与实现[J]. 中国管理信息化, 2013, 14(13):37-39.
- [2] 赵建华, 李楠, 王国双. 移动营业厅应用系统实用性[J]. 中小企业管理与技术, 2013, 21(15):259-260.
- [3] 韩友君. 关于移动营业厅应用系统的应用方案分析[J]. 科技创新与应用, 2013, 28(25):79-79.
- [4] 李建东. 移动营业厅应用系统可靠性设计[J]. 计算机安全, 2013, 7(4):79-82.
- [5] 许平格. 数据库管理系统中查询优化的设计与实现[D]. 浙江: 浙江大学, 2005.
- [6] 刘嫻. 一种改进的 MySQL 数据库访问控制的设计与实现[D]. 武汉: 华中科技大学, 2011.
- [7] 王晓东. J2ee 架构的研究与应用[J]. 科技信息, 2013, 23(23):439-439.
- [8] 廖胜军. 基于 SSH 整合架构的应用研究[D]. 武汉: 武汉理工大学, 2008.
- [9] 胡元发. 基于 J2EE 架构的 SSH 组合框架的设计与应用[D]. 苏州: 苏州大学, 2006.
- [10] Ou Feng, ZOU Min and LI Xiao-Zhen. Survey on Java Technology Framework[J]. Computer Systems& Applications, 2012, 21(8):236-239.
- [11] Farias G, Cervin A, Dormido S, Esquembre F. Java Simulations of Embedded Control Systems[J]. Sensors, 2010, 10(9):8585-8603.
- [12] 陈明志. 基于 SSH 的移动营业厅应用系统的设计[J]. 电子技术与软件工程, 2013, 14(6):22-24.
- [13] 王晓东. J2EE 架构的研究与应用[J]. 科技信息, 2013, 23(23):439-439.
- [14] Ou Feng, ZOU Min and LI Xiao-Zhen. Survey on Java Technology Framework[J]. Computer Systems& Applications, 2012, 21(8):236-239.
- [15] Farias G, Cervin A, Dormido S, Esquembre F. Java Simulations of Embedded Control Systems[J]. Sensors, 2010, 10(9):8585-8603.
- [16] 王庆华. 快学易用 JSP. 北京: 北京邮电大学出版社, 2001 年.
- [17] 邓文渊. JSP 与网络数据库技术. 北京: 中国铁道出版社, 2003.
- [18] 李存斌. JSP 高级编程及其项目应用开发. 北京: 中国水利水电出版社, 2004.
- [19] FLANAGAN. JAVA 技术手册. 中国电力出版社, 2012 年 6 月:1-465
- [20] Paco Hope. Web Security Testing Cookbook[M]. New York: EDITORIAL MANAGER, 2010: 24-26.
- [21] Akgiray et al. An effective way to do network marketing [J]. UNIONS., 2009(9):84-95.
- [22] Loped. Discuss online book sales website design and implementation [J]. J union expo, 2010(11):18-22.
- [23] Solt, Swanson. The sales class website design[J]. Technological era, 2010(8):18-23.
- [24] LEE ANNE PHILLIPS. 巧学活用 HTML4. 电子工业出版社, 2010 年 8 月:1-319
- [25] Rosemary Stockdale. Managing customer relationships in the self-service environment of e-tourism[J]. Journal of Vacation Marketing. Jul 2007;13:。
- [26] (美) 高斯林 (Gosling, J.) 等编著, 陈宗斌, 沈金河译. Java 编程规范 (第三版). 中国电力

出版社. 2005-7-1.

[27] 杨志茹, 温沿书编著, Java 入门与实战教程。人民邮电出版社, 2010-3-1.

[28] David M Geary. Graphic Java 2 Mastering the JFC Volume 2:Swing (3rd Edition) [M]. Prentice Hall/Pearson, 2000.

[29] Joshua Marinacci, Chris Adamson. SWING HACKS [M]. O'reilly, 2007.

[30] James Elliott. Java Swing (Second Edition) [M]. O' Reilly & Associates, 2002. [12].

附录 A 外文翻译—原文部分

—来自 Spring 官方参考文献

Introduction to Spring Web MVC framework

The spring Web model-view-controller (MVC) framework is designed around a `DispatcherServlet` that dispatches requests to handlers, with configurable handler mappings, view resolution, locale, time zone and theme resolution as well as support for uploading files. The default handler is based on the `@Controller` and `@RequestMapping` annotations, offering a wide range of flexible handling methods. With the introduction of Spring 3.0, the `@Controller` mechanism also allows you to create RESTful Web sites and applications, through the `@PathVariable` annotation and other features. "Open for extension..." A key design principle in Spring Web MVC and in Spring in general is the "Open for extension, closed for modification" principle.

Some methods in the core classes of Spring Web MVC are marked `final`. As a developer you cannot override these methods to supply your own behavior. This has not been done arbitrarily, but specifically with this principle in mind.

For an explanation of this principle, refer to *Expert Spring Web MVC and Web Flow* by Seth Ladd and others; specifically see the section "A Look At Design," on page 117 of the first edition. Alternatively, see Bob Martin, *The Open-Closed Principle* (PDF) cannot add advice to the `AbstractController`. You cannot add advice to final methods when you use Spring MVC. For example, you cannot call `controller.setSynchronizeOnSession()` method. Refer to Section 10.6.1, "Understanding AOP proxies" for more information on AOP proxies and why you cannot add advice to final methods. In Spring Web MVC you can use any object as a command or form-backing object; you do not need to implement a framework-specific interface or base class. Spring's data binding is highly flexible: for example, it treats type mismatches as validation errors that can be evaluated by the application, not as system errors. Thus you need not duplicate your business objects' properties as simple, untyped strings in your form objects simply to handle invalid submissions, or to convert the Strings properly. Instead, it is often preferable to bind directly to your business objects.

Spring's view resolution is extremely flexible. A Controller is typically responsible for preparing a model Map with data and selecting a view name but it can also write directly to the response stream and complete the request. View name resolution is highly configurable through file extension or Accept header content type negotiation, through bean names, a properties file, or even a custom `ViewResolver` implementation. The model (the M in MVC) is a `Map` interface, which allows for the complete abstraction of the view technology. You can integrate directly with template based rendering technologies such as JSP, Velocity and Freemarker, or directly generate

XML, JSON, Atom, and many other types of content. The model Map is simply transformed into an appropriate format, such as JSP request attributes, a Velocity template model.

SWF integrates with existing frameworks like Spring MVC and JSF, in both Servlet and Portlet environments. If you have a business process (or processes) that would benefit from a conversational model as opposed to a purely request model, then SWF may be the solution. SWF allows you to capture logical page flows as self-contained modules that are reusable in different situations, and as such is ideal for building web application modules that guide the user through controlled navigations that drive business processes.

Clear separation of roles. Each role — controller, validator, command object, form object, model object, DispatcherServlet, handler mapping, view resolver, and so on — can be fulfilled by a specialized object.

Powerful and straightforward configuration of both framework and application classes as JavaBeans. This configuration capability includes easy referencing across contexts, such as from web controllers to business objects and validators.

Adaptability, non-intrusiveness, and flexibility. Define any controller method signature you need, possibly using one of the parameter annotations (such as `@RequestParam`, `@RequestHeader`, `@PathVariable`, and more) for a given scenario.

Reusable business code, no need for duplication. Use existing business objects as command or form objects instead of mirroring them to extend a particular framework base class.

Customizable binding and validation. Type mismatches as application-level validation errors that keep the offending value, localized date and number binding, and so on instead of String-only form objects with manual parsing and conversion to business objects.

Customizable handler mapping and view resolution. Handler mapping and view resolution strategies range from simple URL-based configuration, to sophisticated, purpose-built resolution strategies. Spring is more flexible than web MVC frameworks that mandate a particular technique.

Flexible model transfer. Model transfer with a name/value Map supports easy integration with any view technology.

Customizable locale, time zone and theme resolution, support for JSPs with or without Spring tag library, support for JSTL, support for Velocity without the need for extra bridges, and so on.

A simple yet powerful JSP tag library known as the Spring tag library that provides support for features such as data binding and themes. The custom tags allow for maximum flexibility in terms of markup code. For information on the tag library descriptor, see the appendix entitled Chapter 42, spring JSP Tag Library

A JSP form tag library, introduced in Spring 2.0, that makes writing forms in JSP pages

much easier. For information on the tag library descriptor, see the appendix entitled Chapter 43, spring-form JSP Tag Library

Beans whose lifecycle is scoped to the current HTTP request or HTTP Session. This is not a specific feature of Spring MVC itself, but rather of the `WebApplicationContext` container(s) that Spring MVC uses. These bean scopes are described in Section 6.5.4, “Request, session, and global session scopes”

Pluggability of other MVC implementations

Non-Spring MVC implementations are preferable for some projects. Many teams expect to leverage their existing investment in skills and tools, for example with JSF.

If you do not want to use Spring’s Web MVC, but intend to leverage other solutions that Spring offers, you can integrate the web MVC framework of your choice with Spring easily. Simply start up a Spring root application context through its `ContextLoaderListener`, and access it through its `ServletContext` attribute (or Spring’s respective helper method) from within any action object. No “plug-ins” are involved, so no dedicated integration is necessary. From the web layer’s point of view, you simply use Spring as a library, with the root application context instance as the entry point.

Your registered beans and Spring’s services can be at your fingertips even without Spring’s Web MVC. Spring does not compete with other web frameworks in this scenario. It simply addresses the many areas that the pure web MVC frameworks do not, from bean configuration to data access and transaction handling. So you can enrich your application with a Spring middle tier and/or data access tier, even if you just want to use, for example, the transaction abstraction with JDBC or Hibernate. maintains a list of default beans to use if you don’t configure any. More on that in the next section. First see the table below listing the special bean types the `DispatcherServlet` relies on.

附录 B 外文翻译—译文部分

Spring Web MVC 框架简介

Spring 的模型-视图-控制器 (MVC) 框架是围绕一个 `DispatcherServlet` 来设计的, 这个 `Servlet` 会把请求分发给各个处理器, 并支持可配置的处理器映射、视图渲染、本地化、时区与主题渲染等, 甚至还能支持文件上传。处理器是你的应用中注解了 `@Controller` 和 `@RequestMapping` 的类和方法, Spring 为处理器方法提供了极其多样灵活的配置。Spring 3.0 以后提供了 `@Controller` 注解机制、`@PathVariable` 注解以及一些其他的特性, 你可以使用它们来进行 RESTful web 站点和应用的开发。

“对扩展开放”是 Spring Web MVC 框架一个重要的设计原则, 而对于 Spring 的整个完整框架来说, 其设计原则则是“对扩展开放, 对修改闭合”。

Spring Web MVC 核心类库中的一些方法被定义为 `final` 方法。作为开发人员, 你不能覆写这些方法以定制其行为。当然, 不是说绝对不行, 但请记住这条原则, 绝大多数情况下不是好的实践。

关于该原则的详细解释, 你可以参考 Seth Ladd 等人所著的“深入解析 Spring Web MVC 与 Web Flow”一书。相关信息在第 117 页, “设计初探 (A Look At Design)”一节。或者, 你可以参考:

Bob Martin 所写的“开闭原则 (The Open-Closed Principle)” (PDF)

你无法增强 Spring MVC 中的 `final` 方法, 比如 `AbstractController.setSynchronizeOnSession()` 方法等。请参考 10.6.1 理解 AOP 代理一节, 其中解释了 AOP 代理的相关知识, 论述了为什么你不能对 `final` 方法进行增强。

在 Spring Web MVC 中, 你可以使用任何对象来作为命令对象或表单返回对象, 而无须实现一个框架相关的接口或基类。Spring 的数据绑定非常灵活: 比如, 它会把数据类型不匹配当成可由应用自行处理的运行时验证错误, 而非系统错误。你可能会为了避免非法的类型转换在表单对象中使用字符串来存储数据, 但无类型的字符串无法描述业务数据的真正含义, 并且你还需要把它们转换成对应的业务对象类型。有了 Spring 的验证机制, 意味着你再也不需这么做了, 并且直接将业务对象绑定到表单对象上通常是更好的选择。

Spring 的视图解析也是设计得异常灵活。控制器一般负责准备一个 `Map` 模型、填充数据、返回一个合适的视图名等, 同时它也可以直接将数据写到响应流中。视图名的解析高度灵活, 支持多种配置, 包括通过文件扩展名、`Accept` 内容头、`bean`、配置文件等的配置, 甚至你还可以自己实现一个视图解析器 `ViewResolver`。模型 (MVC 中的 M, `model`) 其实是一个 `Map` 类型的接口, 彻底地把数据从视图技术中抽象分离了出来。你可以与基于模板的渲染技术直接整合, 如 JSP、Velocity 和 Freemarker 等, 或者你还可以直接生成 XML、JSON、Atom 以及其他多种类型的内容。`Map` 模型会简单地被转换成合适的格式, 比如 JSP 的请求属性

(attribute)，一个 Velocity 模板的模型等。Spring mvc 新特性 Spring Web Flow

Spring Web Flow (SWF) 意在成为 web 应用中的页面流(page flow)管理中最好的解决方案。

SWF 在 Servlet 环境和 Portlet 环境下集成了现有的框架，如 Spring MVC 和 JSF 等。如果你的业务流程有一个贯穿始终的模型，而非单纯分立的请求，那么 SWF 可能是适合你的解决方案。

SWF 允许你将逻辑上的页面流抽取成独立可复用的模块，这对于构建一个 web 应用的多个模块是有益的。that guide the user through controlled navigations that drive business processes.

Spring 的 web 模块支持许多 web 相关的特性：

清晰的职责分离。每个角色——控制器，验证器，命令对象，表单对象，模型对象，DispatcherServlet，处理器映射，视图解析器，等等许多——的工作，都可以由相应的对象来完成。强大、直观的框架和应用 bean 的配置。这种配置能力包括能够从不同的上下文中进行简单的引用，比如在 web 控制器中引用业务对象、验证器等。

强大的适配能力、非侵入性和灵活性。Spring MVC 支持你定义任意的控制器方法签名，在特定的场景下你还可以添加适合的注解（比如 @RequestParam、@RequestHeader、@PathVariable 等）

可复用的业务代码，使你远离重复代码。你可以使用已有的业务对象作为命令对象或表单对象，而不需让它们去继承一个框架提供的什么基类。

可定制的数据绑定和验证。类型不匹配仅被认为是应用级别的验证错误，错误值、本地化日期、数字绑定等会被保存。你不需要再在表单对象使用全 String 字段，然后再手动将它们转换成业务对象。

可定制的处理映射和视图解析。处理器映射和视图解析策略从简单的基于 URL 配置，到精细专用的解析策略，Spring 全都支持。在这一点上，Spring 比一些依赖于特定技术的 web 框架要更加灵活。

灵活的模型传递。Spring 使用一个名称/值对的 Map 来做模型，这使得模型很容易集成、传递给任何类型的视图技术。

可定制的本地化信息、时区和主题解析。支持用/不用 Spring 标签库的 JSP 技术，支持 JSTL，支持无需额外配置的 Velocity 模板，等等。

一个简单但功能强大的 JSP 标签库，通常称为 Spring 标签库，它提供了诸如数据绑定、主题支持等一些特性的支持。这些定制的标签为标记 (markup) 你的代码提供了最大程度的灵活性。关于标签库描述符 (descriptor) 的更多信息，请参考附录第 42 章 Spring JSP 标签库一个 Spring 2.0 开始引入的 JSP 表单标签库。它让你在 JSP 页面中编写表单简单许多。关于标签库描述符 (descriptor) 的更多信息，请参考附录第 43 章 Spring 表单的 JSP 标签库

新增生命周期仅绑定到当前 HTTP 请求或 HTTP 会话的 Bean 类型。严格来说，这不是 Spring MVC 自身的特性，而是 Spring MVC 使用的上下文容器 `WebApplicationContext` 所提供的特性。这些 bean 的 scope 在 6.5.4 请求、会话及全局会话 scope 一节有详细描述。

允许其他 mvc 实现

有些项目可能更倾向于使用非 Spring 的 MVC 框架。许多团队希望仍然使用现有的技术栈，比如 JSF 等，这样他们掌握的技能 and 工具依然能发挥作用。

如果你确实不想使用 Spring 的 Web MVC，但又希望能从 Spring 提供的一些解决方案中受益，那么将你所使用的框架和 Spring 进行集成也很容易。只需要在 `ContextLoaderListener` 中启动一个 Spring 的根应用上下文（root application context），然后你就可以在任何 action 对象中通过其 `ServletContext` 属性（或通过 Spring 对应的 helper 方法）取得。不需要任何侵入性的插件，因此不需要复杂的集成。从应用层的视角来看，你只是将 Spring 当成依赖库使用，并且将它的根应用上下文实例作为应用进入点。

即使不用 Spring 的 Web MVC 框架，你配置的其他 Spring 的 bean 和服务也都能很方便地取得。在这种场景下，Spring 与其他 web 框架的使用不冲突。Spring 只是在许多问题上提出了其他纯 web MVC 框架未曾提出过的解决方案，比如 bean 的配置、数据存取、事务处理等，仅此而已。因此，如果你只是想使用 Spring 的一部分特性来增强你的应用，比如 Spring 提供的 JDBC/Hibernate 事务抽象等，那么你可以将 Spring 作为一个中间层和/或数据存取层来使用。

附录 C 软件使用说明书

1. 前期准备

- (1) 下载 Eclipse 和 MYSQL 5.1.5
- (2) 浏览器 Chrome
- (3) JDK1.8 版本及以上, Java 相关环境等。
- (4) Tomcat 服务器配置
- (5) Jar 包的导入

2. 使用说明

- (1) 首先将 Eclipse 根据 db 配置文件 与 MYSQL 数据库进行连接。
- (2) 在 Mysql 数据库中, 创建一个名为 yidong 的数据库, 并加载 yidong.sql 数据库文件
- (3) 在数据库中先添加一个管理员的用户。
- (4) 登录系统, 在登录时输入一开始录入的管理员的用户名、密码。
- (5) 密码信息设置模块: 用户登录之后, 在密码信息设置模块中, 通过输入当前密码, 再输入需要修改的新设置密码, 再次输入一个上一个输入框中相同的密码, 达到修改密码的目的。
- (6) 账号信息设置模块: 在账号信息设置模块中, 可以随时修改自己的基本身份信息,例如您的姓名、年龄、手机号码等。
- (7) 用户信息添加模块: 用户进入用户信息管理模块中, 通过点击新增按钮, 新增用户, 管理员可以通过这个模块对新开发人员和领导的账号进行新增, 相当于注册功能。
- (8) 用户信息删除模块: 对其他用户注册信息进行点击自动勾选删除个人信息模块按钮,可以对已经自动储存到现在的所有其他用户注册个人信息全部进行自动勾选删除,也就是用户可以通过多次勾选,实现批量删除, 当一些员工离职之后可以轻松地管理他们的信息。
- (9) 用户信息修改模块: 用户姓名输入一个需要姓名信息修改的用户姓名,年龄,部门,电话等信息,随后用户可以点击下方的修改信息按钮,可以对已经注册存在的平台用户所提供的信息是否进行了修改,达到对信息的完全掌握。
- (10) 部门信息管理模块: 管理员进入部门的信息管理模块, 就能够对部门信息进行一些基本操作, 比如说增添部门, 删除修正部门, 例如目前有业务部、技术部、项目部等, 后续根据生产上业务的增加, 陆续增加部门信息。
- (11) 部门信息提交模块: 管理员输入需要添加的部门信息, 在部门信息提交模块中, 可以对部门信息进行新增, 以达到对部门新增的目的。
- (12) 业务信息修改模块: 用户可以修改现有业务信息, 例如可以修改项目工时、项目提出人、项目开始结束时间等等。
- (13) 业务信息删除模块: 用户可以删除部门信息, 当这个业务已经挤压很久了, 或者早已完成, 就可以对他们单个删除或者批量删除。

(14) 业务信息添加模块：可以新增业务信息，实现业务的实时更新。

(15) 日志信息删除模块：用户所有的正确操作和错误操作都可以在这里面记录到并反馈给管理人员，用户已经进入了日志信息的删除管理模块,可以删除冗余的日志信息。