

郑州轻工业学院

实训报告

实践环节： [1326119]软件开发综合实训（jsp）

项目名称： 基于 MVC 模式的电信资费管理系统

院（系）： 软件学院

专业班级： 软件工程（Java 技术）15-02

学 号： 541513440224

学生姓名： 门甜甜

指导教师： 刘育熙 陈明

学年学期： 2016-2017 学年第 2 学期

成 绩： _____

郑州轻工业学院软件学院

学生实训任务书

一、题目

基于 MVC 模式的电信资费管理系统

二、学生信息

专业班级软件工程(Java技术)15-02 学号 541513440224 姓名门甜甜

三、用户需求及约束条件

1、功能需求：实现前台访客页面与后台管理，电信 IP 业务计费总体需求模块如下：

- ①登陆管理：实现登陆后方可使用该应用程序。
- ②用户管理：增、删、改、查用户信息。
- ③管理员管理：增、删、改、查管理员信息。
- ④资费管理：增、删、改、查资费信息。
- ⑤用户自服务管理：用户修改密码，查询账单。
- ⑥数据采集模块：实现数据采集。
- ⑦账单管理：按月生成用户计费账单。
- ⑧帐务管理：按月生成帐务月、年统计报表。并生成统计图形和打印报表。

2、性能需求：

- ①站点中网页的外观设计和内容应相互独立。
- ②站点要有统一的风格和布局：使用母版页、内容页、主题。
- ③站点要为用户提供方便的站点导航。

3、网站开发要求：

选用技术： Java、JSP、HTML； DBMS： Microsoft SQL server2000、MY SQL 或 Oracle； 软件集成开发环境（IDE）： NetBeans 或 Eclipse； 采用 MVC 模式。

注：以上 8 个功能模块，至少完成 5 个以上，根据完成模块的多少给作品进行评分，对完成模块多者，可以根据情况对本次试训成绩进行适当加分；技术、数据库、IDE 可根据自己情况进行选定。

四、设计要求、技术参数及设计规格

- 1、前台实现：电信计费系统 IP 主页面。
- 2、后台实现：对用户信息、管理员信息、资费管理。

注：根据个人知识掌握情况，可继续扩充该系统的功能。

五、工作量

天/日期	任务描述	通过标准
2017年6月12日	完成“需求分析说明书”、“概要设计说明书”	需求分析说明书、概要设计说明书提交.
2017年6月13日	完成数据库设计	表及表关系合理.
2017年6月14日	登陆管理	实现登陆后方可使用.
2017年6月15日	管理员模块的实现	实现管理员管理功能.
2017年6月16日	用户管理模块的实现	实现用户管理
2017年6月19日	用户管理模块的实现	实现用户管理
2017年6月20日	资费管理模块功能的实现	增、删、改、查资费信息.
2017年6月21日	资费管理模块功能的实现	增、删、改、查资费信息.
2017年6月22日	帐单管理	按月生成用户计费账单.
2017年6月23日	帐单管理	按年生成用户计费账单.
2017年6月26日	帐务管理	生成可打印报表
2017年6月27日	帐务管理，并生成统计图形和报表	生成可打印报表
2017年6月28日	用户自服务模块的实现	实现用户自服务，修改密码
2017年6月29日	用户自服务模块的实现	实现用户自服务，查询账单.
2017年6月30日	用户数据采集功能模块的实现以及实训综合项目验收	实现用户自服务、数据采集模块的功能以及实训综合项目验收.

六、考核形式、考核时间、考核指标及成绩评定

实训结束后，上交作品（电子版）、实训报告（打印版、电子版）。

成绩评定：考勤（20%）+作品（50%）+实训报告（30%）

注：作品、实训报告上传到指定ftp上后，请检查本人上传情况。

完 成 期 限：2017年6月30日

指导教师签章：_____

专业负责人签章：_____

教学院长签章：_____

2017年6月12日

实训进度日历

实践环节		[1326119]软件开发综合实训(jsp)				
项目名称		基于 MVC 模式的电信资费管理系统				
学年学期		2016-2017 学年第二学期	专业班级	软件工程（Java 技术）15-02	指导教师姓名	刘育熙、陈明
天	上午/下午	实训任务详细描述	教师实训任务的指导（解决方案、实现步骤、技术路线、难点提示）		通过标准	
第一天	上午	完成“需求分析说明书”	①查阅其他相关资料，了解系统的基本工作原理；②书写需求分析说明书描述电信资费管理系统的功能需求、性能需求、运行需求、数据描述等。			
	下午	完成“概要设计说明书”；	①书写概要设计说明书描述电信资费管理系统的总体设计、接口设计、数据结构设计、运行设计等。		概要设计说明书提交	
第二天	上午	完成数据库设计	设计表，用于记录用户基本信息，并录入数据；设计用户表、管理员表，用于存储相关信息；设计资费信息表，用于存储资费信息。			
	下午	完成数据库设计	设计表，用于记录用户基本信息，并录入数据；设计用户表、管理员表，用于存储相关信息；设计资费信息表，用于存储资费信息。		数据库和表建成	
第三天	上午	登陆管理	实现登陆功能，通过用户名密码进行登陆。			
	下午	注册管理	不能登陆时注册用户名密码。		输入正确用户名密码实现登陆和注册。	
第四天	上午	管理员模块的实现	在系统添加业务类，该类包括访问数据库经常使用的方法。 管理员管理相关业务： ①设计管理账号业务。 ②设计管理管理员基本信息的表。			
	下午	管理员管理模块的实现	③设计管理员的各类权限。		实现管理员管理模块	

第五天	上午	用户管理模块的实现	用户管理相关业务： ①开通帐号。	
	下午	用户管理模块的实现	②查询帐号。	
第六天	上午	用户管理模块的实现	③修改、暂停、删除帐号。	
	下午	用户管理模块的实现	③修改、暂停、删除帐号。	实现用户管理相关功能
第七天	上午	资费管理模块功能的实现	资费的相关业务： 电信产品(Products): ① 在线实验室出租业务。	
	下午	资费管理模块功能的实现	② 163/169 拨号上网。	
第八天	上午	资费管理模块功能的实现	移动/固定电话服务。	
	下午	资费管理模块功能的实现	③ 在线电影下载。 设计相应表，对数据进行管理。	实现资费管理的功能
第九天	上午	帐单管理	帐单管理： 整合系统按月生成用户帐单。 账单信息（可参考移动或联通的账单管理模块）： <ul style="list-style-type: none"> ● 用户标识信息：账单 ID、姓名、账号、状态、联系电话、开通日起、Email。 ● 账务信息：账号、日期、登陆时间长度、本月费用。 	
	下午	帐单管理	整合系统按月生成用户帐单。	实现帐单管理管理按月
第十天	上午	帐单管理	整合系统按年生成用户帐单。	
	下午	帐单管理	整合系统按年生成用户帐单。	实现帐单管理管理按年
第十一天	上午	帐务管理	帐务管理： 按月、年分别生成《月帐务信息统计报表》和《年帐务信息统计表》。 月帐务信息(可参考移动或联通的帐务模块): <ul style="list-style-type: none"> ● 生成一个计费月周期中， 	

			每一天的帐务信息。 ● 月帐务信息：日期、时长、费用。	
	下午	帐务管理	月帐务信息(可参考移动或联通的帐务模块): ● 生成一个计费月周期中, 每一天的帐务信息。 ● 月帐务信息：日期、时长、费用。	实现月账务信息
第十二天	上午	帐务管理	年帐务信息(可参考移动或联通的帐务模块): ● 生成一个计费年周期中, 每一月的帐务信息。 年帐务信息：月期、时长、费用。	
	下午	帐务管理	年帐务信息(可参考移动或联通的帐务模块): ● 生成一个计费年周期中, 每一月的帐务信息。 年帐务信息：月期、时长、费用。	实现年账务信息
第十三天	上午	用户自服务模块的实现	用户自服务需求: 需求: 用户通过 Internet 自查询自己的当前或历史账单, 并能修改自己的密码或变更相关业务。	
	下午	用户自服务模块的实现	用户信息修改。	
第十四天	上午	用户自服务模块的实现	账单明细查询：登陆时间、退出时间、时长。	
	下午	用户自服务模块的实现	账单明细查询：登陆时间、退出时间、时长。	实现用户自服务管理
第十五天	上午	用户数据采集功能模块的实现以及实训综合项目验收	数据采集需求: 数据采集中心通过 Socket、TCP/IP 适时监听并收集采集信息, 保存到数据库中, 完成数据的采集。	实现系统通信
	下午	用户数据采集功能模块的实现以及实训综合项目验收	项目的综合验收以及讲解撰写实训报告的注意事项。学生撰写实训报告。	项目综合验收

制订教师：刘育熙、陈明

目 录

1 系统需求分析	1
1.1 项目背景	1
1.2 功能模块需求分析	1
1.2.1 登录管理需求	2
1.2.2 用户管理需求	2
1.2.3 管理员管理需求	2
1.2.4 资费管理需求	3
1.2.5 账单管理需求	3
1.2.6 账务管理需求	3
1.2.7 用户自服务管理需求	4
1.3 其他需求	4
2 系统设计	5
2.1 系统模块设计	5
2.1.1 登录模块设计	6
2.1.2 用户管理设计	6
2.1.3 管理员管理设计	6
2.1.4 资费管理设计	7
2.1.5 账单管理设计	7
2.1.6 账务管理设计	7
2.1.7 用户自服务管理设计	8

2.2 数据库设计	8
2.3 MVC 设计模式	10
2.4 系统模型设计	12
2.5 设计原则	12
3 系统实现.....	14
3.1 登录界面的实现	14
3.2 主界面的实现	15
3.3 用户管理界面的实现	16
3.4 管理员管理界面的实现	17
3.5 资费管理界面的实现	17
3.6 账单管理界面的实现	18
3.7 账务管理界面的实现	18
3.8 用户自服务管理界面的实现	19
3.9 退出系统界面的实现	19
3.10 数据库的实现	20
4 项目总结.....	21
5 实训感想.....	23

1 系统需求分析

1.1 项目背景

随着网络的普及以及科技的飞速发展，信息产业得到了空前的发展，信息服务也已经深入到国民生活的方方面面。与之相关的，是通信技术与电信服务行业的高速发展。近年来，电信行业经历了很大的变化，从固定到移动，从语音到数据，从电路交换到分组交换，从拨号上网到光线宽带。随着提速降费政策的提出和民间资本的准入，电信行业将迎来更多的挑战和机遇。

近年来，国内各运营商面临着严峻的市场竞争形势，市场主体的多样化和用户需求的个性化已经使传统的电信业务受到了严重的冲击。如何提高运营效率、降低运维成本、改善服务质量、提高服务水平，是运营商迫切需要解决的问题。对运营商来说，重心要从“面向网络、面向设备”向“面向业务、面向客户”进行转换，而后端运维则要打破专业、条线分割，向综合化集中维护发展，最大程度地利用现有网络资源，提高后端对业务变化的响应速度，改善运营商对客户的服务质量。

在这种大背景下，对电信运营商的计费和其他业务支持系统的要求有了大的变化，如何能够达到尽量为用户提供优质方便的服务，以吸引更多的用户，同时又能提高企业的运行效率，降低人力资源成本，使企业能够高效运行和更好的发展，成为亟待解决的问题。电信计费系统成为了电信行业运行体系中最重要支撑系统，为电信企业的信息综合管理提供了一套可行而又方便的解决方案。从为提高信息管理效率和为客户提供高效便捷的服务的角度来看，一个好的计费系统应具有以下特点：

- (1) 综合性：能够满足同时从不同的数据源中采集数据，并能够对数据进行适当的处理，并送到数据处理中心去。
- (2) 灵活性：计费系统是业务规则驱动以及对用户有好。
- (3) 集成性：系统具备高内聚低耦合性，系统具有良好的接口和可扩展性。
- (4) 可靠性：对计费业务，要求保证数据的完整性。

1.2 功能模块需求分析

作为一款商用的大数据管理系统，在多用户操作方面要有很好的兼容性。对于不

同用户要有身份级别设计，实现多种身份操作，不仅商家可以操作，用户也可以操作。对于一个资费管理系统而言，其根本作用是对电信用户的资费管理，所以该功能一定要完善和详细。对于系统的使用者之一电信运营商而言，可以实现多用户操作，查看月账单和报表等功能。对于电信用户而言，可以登录该系统，并且可以实现一些功能的自助服务。比如基本信息的查询，个人信息的修改等。根据业务模型和电信业务的需要，该系统的功能需求模块分为以下几种类型。

1.2.1 登录管理需求

登录系统需要具有一定的保密性，因此在登录界面需设置账号和密码，只有输入正确的信息后，才能使用该应用程序，否则将无法使用。同时为了适应多方面的需求，并对系统进行有效的管理，可将本系统的登录模块分为管理员登录和用户登录两种，即采用管理员与用户同界面不同功能的模式进行开发。

1.2.2 用户管理需求

用户管理模块包括开通账号、用户账号查询、用户列表、用户修改、用户删除，其中开通账号查询可以更方便用户的查询，用户可以通过开通账号查询来得到一些相关的信息，包括用户的状态、姓名、电话、邮箱、开户日期。用户列表中具有增加、暂停、修改、和删除等功能。

该模块实现对用户的账务账号和业务账号进行管理，具有用户管理权限的管理员登录成功后，即可进行开通账号操作。账务账号必须和用户的一些真实信息关联起来，如用户的真实姓名、邮箱、联系电话等。开通时，由用户提供这些信息，然后由管理员将信息录入到系统中并保存下来。对于开通好的账务账号也是由具有用户管理权限的管理员来管理，如暂停账务账号、恢复账务账号、删除账务账号、修改用户真实信息等。由于用户的一个账务账号会对应多个业务账号，所以对账务账号的操作会级联影响关联的所有业务账号。

1.2.3 管理员管理需求

管理员管理模块的需求主要有：管理用户账号，包括账号开通、暂停、恢复、删

除等；管理员资料管理，包括姓名、身份证号、地址等；管理员信息管理，包括 ID、姓名、账号、密码、状态（正常/暂停/关闭）、联系电话、Email、开通日起、停止日起、权限（查询/修改/开户/ALL）等。

1.2.4 资费管理需求

为了给客户差异化的服务，电信运营商可能会提供多种不同的套餐和资费方案。而多种的资费和服务套餐也给人工管理增加了难度，因此需要增加资费管理模块对资费进行管理，实现对资费信息的插入、修改、删除和查看等功能。资费管理模块是用来管理资费的，具有资费管理权限的管理员登录成功后，可以添加新的资费，查询所有的资费，可以修改现有的资费信息，还可以删除资费。特别注意的是，由于对资费的修改会影响选择了这种资费的用户要缴纳的费用，所以在修改资费信息前要事先通知所有的已经选择了这种资费的用户，只有当没有任何用户选择某个资费了，这个资费才能被删除掉。

1.2.5 账单管理需求

为了方便对账单进行管理，系统需要对账单进行记录、分类和统计等管理，并且能够将分析后的信息提供给相应的管理员和用户查看，因此帐单管理模块需要整合系统按月生成用户帐单。具有账单管理权限的管理员可以利用此子系统对所有用户的月账单进行查询。此子系统不仅提供对某个账务账号上产生的总的费用进行查询，还提供对某个账务账号上产生的每个费用明细进行查询。该模块包括用户标识信息，如账单 ID、姓名、账号、状态、联系电话、开通日起、Email；账务信息，如账号、日期、登陆时间长度、本月费用。账单管理是通过报表的方式来设计需求的，需要使用报表从数据库里面取出来数据，从而来让数据以报表的形式呈现出来，这样也方便了管理者的管理。

1.2.6 账务管理需求

帐务管理是要把用户的费用信息按月、年分别生成《月帐务信息统计报表》和《年帐务信息统计表》。生成的计费年周期中包括每个月的费用信息，如月期、时长、费

用等；生成的计费月周期中包括每天的账务信息，如日期、时长、费用等。

1.2.7 用户自服务管理需求

为了方便用户查询自己的账户信息，和减少企业投入的人力成本，一些问题可以从线下转移到线上处理。通过网络的自助式服务可以方便用户，所以需要有用户自服务模块。用户可以通过 Internet 自查询自己的当前或历史账单，并能修改自己的密码或变更相关业务。

1.3 其他需求

对于一款系统而言，首先要满足用户的基本要求，其次还要求界面简洁、功能明了、设计合理和操作简单；安全性高，适合多用户操作。大致可以分为以下几个方面：

（1）操作需求：对于用户而言，我们要做到以最少的做工换来最大效果，竟可能让用户以最少的输入量达到效果。在满足实用性的条件下，界面设计做到美观大方，大小适中，排列整齐，各种控制键的中文名字简单明了，不同的业务甚至不同的功能，有不同的界面。美观的界面配色柔和，减轻视觉疲劳，给用户一个良好的计算机工作环境。

（2）性能需求：本系统在性能上尽量做到实时性强、数据容量小、响应速度快、稳定性高、出错率低、容错性好等优点。

（3）可维护性：电信资费管理系统的可维护性包括对数据的维护和对系统的维护。在设计时，主要从以下几个方面进行考虑：对于正常的数据库维护，管理人员可以通过电信资费管理系统本身来实现。数据库维护包括权限设置、高级地址的集合运算、无用数据的删除、数据库的清理、数据备份与恢复等功能。全面按照软件工程的要求来开发电信资费管理系统，做到严格管理、严格测试。每个工作阶段，都具备相应的经过严格审查的文档，为将来系统服务提供技术上的保证。

2 系统设计

通过前面的需求分析及对系统逻辑功能的要求分析，从实际出发，进行各种具体设计，确定系统的实施方案，最终给出详细设计方案，为系统实施阶段制定详细计划。接下来将从功能模块设计、设计模式、数据库设计等方面对系统进行设计。

2.1 系统模块设计

本系统具体可分为以下几个模块，分别是登录模块、用户管理模块、管理员管理模块、资费管理模块、账单管理模块、账务管理模块、用户自服务管理模块。使用管理员账号登录该系统后菜单栏会出现用户管理、资费管理、账单管理、账务管理、管理员管理、退出系统这些选项；而使用用户账号登录后，菜单栏会出现用户自服务、退出系统两个选项。这样可以有效的对系统进行管理，总体运行设计如图 2-1 所示。

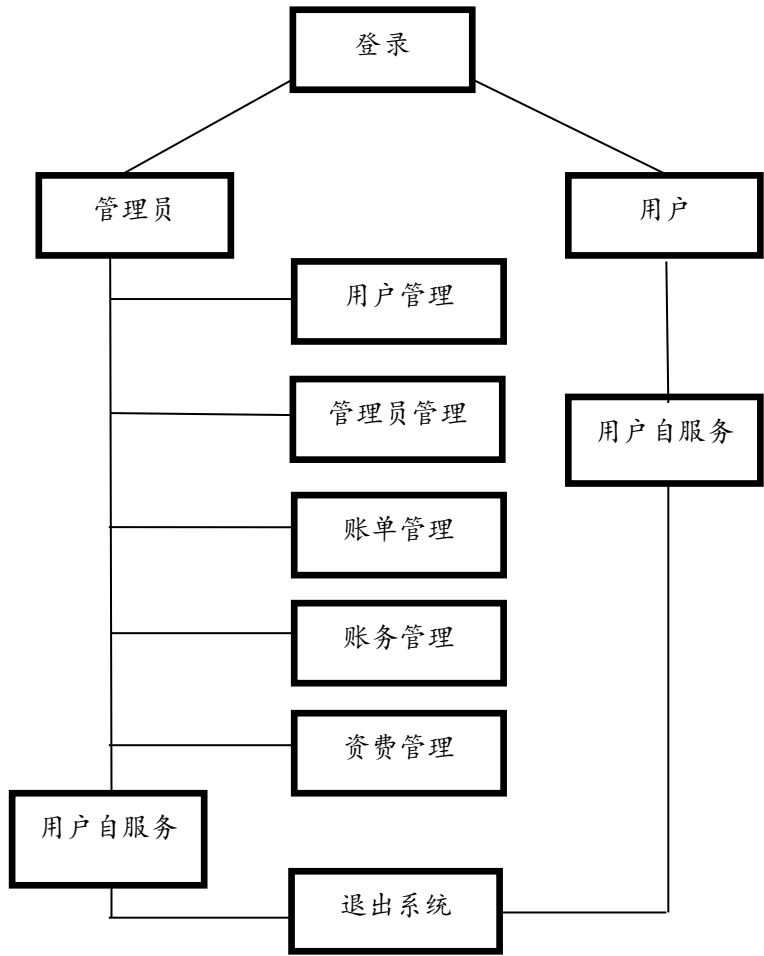


图 2-1 总体运行设计

2.1.1 登录模块设计

登录界面是整个系统较为独立存在的页面，该页面包括用户进入系统所需的用户名与密码。整个界面只有一个对话框，因此需要将对话框的大小、对话框在页面布局中所占用的比例、背景图的设计、背景色的调和进行详细设计。只有输入正确的用户名和密码才能使用本系统，所以需要在数据库中设计用户名与密码的用户表，设计完毕后，在表格中增加一些数据作为原始的登录信息。在代码的编写上，需要完成两个代码的构建。一个是视图层，它是一个静态界面的显示结果，包括用户的登录名与密码、用户名与密码所占有的文本框及登录和清除按钮。另一个就是数据页面的处理，将静态页面与数据库的内容有机的结合在一起才能算是登录页面真正的完成。二者相辅相成，缺一不可。

2.1.2 用户管理设计

用户管理模块是对整个用户的一个管理界面，该模块主要包括用户列表与开通用户。用户列表显示的是用户的个人基本信息，包括用户的状态、账务账号、姓名、电话、邮箱以及用户的删除与修改。用户列表的信息是动态的从数据库中获取来的，通过数据处理呈现在系统的页面中，可以让用户直观的了解到自己的信息，倘若用户的个人信息有误，可以通过表格后面的链接进行修改，也可以删除用户的基本信息。

开通用户界面采用文本框、单选框、复选框结合的形式，以确保用户的个人信息能够完整有效的被录入到数据库中。所填信息包括必填选项和选填选项，一切的选项都会在数据库中进行备份，当在开通用户界面进行填写后，点击确定按钮进行提交，用户所开通的信息就会在后台数据库中有新一条的数据存储下来，方便以后的查询。开通后数据库中的数据就会在用户列表中显示出来。用户开通的信息量大，数据多，如若完全显示在列表中会出现溢出情况，因此可以在功能实现时挑拣一些重要的信息在列表中呈现。

2.1.3 管理员管理设计

管理员界面是用来管理管理员自己的界面，有管理员管理权限的管理人员可以在此界面上对管理员进行增、删、改、查等操作。该模块可以设置管理员的权限问题，

即添加或删除管理员权限。

2.1.4 资费管理设计

资费管理模块是对资费的管理，包括资费列表、添加资费和资费修改三个部分。资费列表中含有资费名称、月租费用、每小时的费用以及资费说明。与之相对应的，在数据库中需要对上述内容进行相应的设计，包括表名与数据类型，以及表格中主键的设置。若资费的费用有变更，可以通过列表中的资费管理进行修改，在点击“修改”按钮后就会进入资费修改的页面。该页面中有要修改的项目列表，但是资费的名称仍然保留，这是由于要修改的不是资费名称，而是费用与资费说明的变更。在添加资费页面，管理员可以添加新的资费政策，包括资费名称、月租费用、每小时费用、资费描述等。在填写完成之后单击“提交”按钮即可将新的资费政策添加到资费列表中供用户选择使用。

2.1.5 账单管理设计

账单管理模块是管理人员在权限允许的情况下，查询个人用户月账单的使用情况，包括登录时长、费用、状态等。管理人员在输入用户的账务账号，选定年份和月份后便可查询到用户的资费使用情况。通过“明细”按钮可以进入用户资费明细页面，在该页面中可以查看到用户的账单明细，包括总计、服务器、总费用、登录时间、退出时间和时长等。通过这些数据，管理人员可以统计出不同人群的资费使用情况，以便对下一阶段的工作做好准备。

2.1.6 账务管理设计

在账务管理模块中需要按照月、年分别生成《月账务信息统计报表》和《年账务信息统计表》。该模块包括服务器月账务查询和服务器年账务查询这两个界面，在服务器月账务查询中，管理人员可以查询到系统中每个服务器每月的使用情况，并且在账务明细中可以查询到该月每天的服务器使用情况。在服务器年账务查询中，管理人员可以查看到每一年的服务器使用情况，需要选定需要查询的年份，单击年账务查询即可查找到系统所有服务器的使用情况。并且在清单后面的详细清单里可以看到这一

年里每个月服务器的使用情况，方便管理人员对服务器的管理和对服务器的调试运行。

2.1.7 用户自服务管理设计

用户自服务管理模块是为用户而设计的，所有的用户都有权限来操作自己的信息。在该模块用户可以进行账单的查询和个人信息的修改。在个人信息修改中，用户可以自己修改自己的个人账户信息，包括密码、姓名、联系方式、电子邮箱、支付方式、职业、性别、公司、公司邮箱、邮编等。需要修改个人信息的用户需要在自己原始的信息上面直接修改，修改完成后单击“修改”按钮即可完成对自己信息的修改和完善。同时这些信息会被同步到管理员页面中的用户列表中。

在账单查询中，用户可以查询到具体日期的账单情况，其中年份和月份是采用下拉框的形式供用户选择的，这样可以方便用户的操作。选定好查询的条件后点击“账单查询”按钮便可以查询到当前用户当月资费的使用情况，包括登录时长和费用。在数据的后面还有一个“明细”按钮，里面是用户具体使用情况，包括业务账号、服务器、总计和总费用。用户能够在这里清楚的看到自己月账单的使用情况以便合理安排资费的使用。

2.2 数据库设计

数据库由用户应用程序要访问的数据源和自身参数组成，可以通过数据库来完成数据的增加、查询、删除的功能，所以有了数据库可以更加方便数据的保存和更改。如果不使用数据库，则需要使用数组来保存数据，相对比较麻烦，而且工作量比较大。因此数据库是电信资费管理系统中非常重要的环节之一，数据库是按照数据结构来组织、存贮的仓库，在项目中有着重大的作用。一个结构良好的数据库可以很好的实现系统的查询和管理，并且能够极大的提高系统的运行效率，以及一个确定数据库存储结构与存取方法的物理设计，建立起既能够反映现实世界信息间的联系，满足用户数据要求和加工要求，又能被某个数据库管理系统所接受，同时能实现系统目标，并有效存取数据的数据库。

本系统使用 MySQL 数据库进行连接，用户表(user)用于管理登录页面中用户登录的信息，具体表设计如表 2-1 所示。

表 2-1 用户表(user)

字段名称	字段类型	字段长度	字段说明
userName	varchar	10	用户名
password	varchar	10	用户密码

用户列表(userlist)用于管理用户的基本信息，具体表设计如表 2-2 所示。

表 2-2 用户列表(userlist)

字段名称	字段类型	字段长度	字段说明
id	varchar	6	账务账号
password	varchar	50	账务密码
name	varchar	50	姓名
sex	varchar	50	性别
methord	varchar	50	付款方式
job	varchar	50	职业
company	varchar	50	公司
province	varchar	50	省份
telephone	varchar	50	电话
mail	varchar	50	邮箱
mailaddress	varchar	50	公司邮箱
post_code	varchar	50	邮政编码
date	varchar	50	开户日期
state	varchar	50	状态

账单信息表(tab)用于管理账单信息，具体表设计如表 2-3 所示。

表 2-3 账单信息表(tab)

字段名称	字段类型	字段长度	字段说明
id	varchar	50	账号
time	int	50	登录时长（小时）
spent	int	50	费用（元）
state	varchar	50	状态
date	varchar	50	开户日期

资费表(pricinglist)用于管理资费信息，具体表设计如表 2-4 所示。

表 2-4 资费表(pricinglist)

字段名称	字段类型	字段长度	字段说明
name	varchar	50	资费名称
month_rent	varchar	50	月租费用
hour_rent	varchar	50	小时费用
introduce	varchar	50	资费描述

账务信息表(unit)用于管理账务信息，具体表设计如表 2-5 所示。

表 2-5 账务信息表(unit)

字段名称	字段类型	字段长度	字段说明
id	int	6	账务账号
time	int	6	时长
dengtime	date	50	登录时间
tuitime	date	50	退出时间
name	varchar	50	服务器名

2.3 MVC 设计模式

本系统是基于 MVC 模式的系统开发，MVC 设计模式是目前用得比较多的一种设计模式，把应用程序分成三层：模式 Model（模型）表示业务逻辑层，View（视图）代表表示层，Controller（控制器）代表控制层。

（1）View

在 Java Web 应用程序中，View 部分一般用 JSP 和 HTML 构建。客户在 View 部分提交请求，在业务逻辑层处理后，把处理结果又返回给 View 部分显示出来。因此，View 部分也是 Web 应用程序的用户界面。

（2）Controller

Controller 部分一般由 Servlet 组成。当用户请求从 View 部分传过来时，Controller 把该请求发给适当的业务逻辑组件处理；请求处理完成后，又返回给 Controller。后者再把处理结果转发给适当的 View 组件显示。因此，Controller 在视图层与业务逻辑层之间起到了桥梁作用，控制了它们两者之间的数据流向。

(3) Model

Model 部分包括业务逻辑层和数据库访问层。在 Java Web 应用程序中，业务逻辑层一般由 JavaBeans 或 EJB 构建。EJB 是 J2EE 的核心组件，可以构建分布式应用系统。与普通 JavaBeans 不同，它由两个接口和一个实现类组成，并且包含一些固有的用于控制容器生命周期的方法。

按照 MVC 设计模式，用 JSP, Servlet 及 JavaBean 构建一个登录系统。构建步骤如下：

- (1) 数据库设计：使用 MySQL 作为数据库，构建数据库。
- (2) 构建视图组件：登录页面、主页面、以及注册页面。
- (3) 构建控制层组件：一个 Servlet，取名为 LoginServlet.java。
- (4) 构建业务逻辑层组件 (Model 组件)：一个 JavaBean，取名为 LoginBean.java。
- (5) 构建数据访问层组件。
- (6) 编译、打包程序。
- (7) 部署该程序到 Web 服务器 Tomcat 中，然后运行。该系统的工作流程图如图 2-2 工作流程图所示。

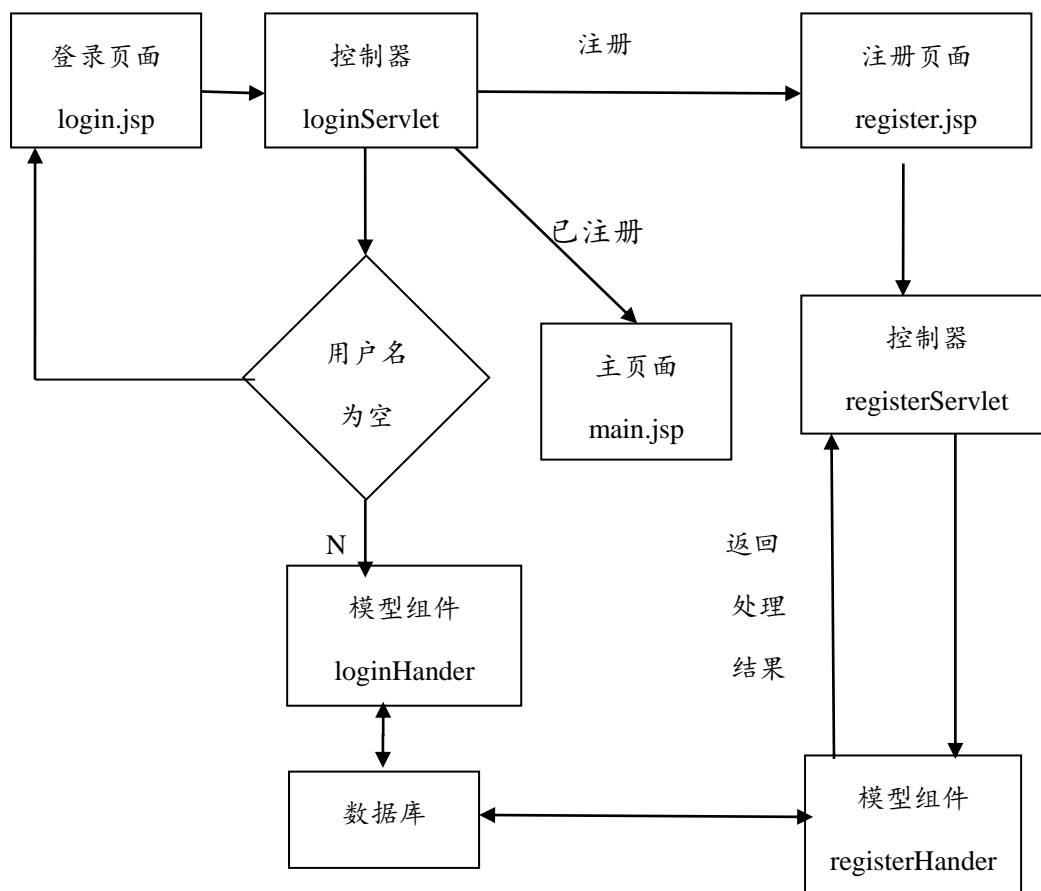


图 2-2 工作流程图

2.4 系统模型设计

本系统通过 MVC 设计模式，前端采用 JSP 技术，通过 xml 配置连接后台 Servlet 接口，后台 Servlet 通过调用相应的程序来完成操作与控制。系统模型图如图 2-3 所示。

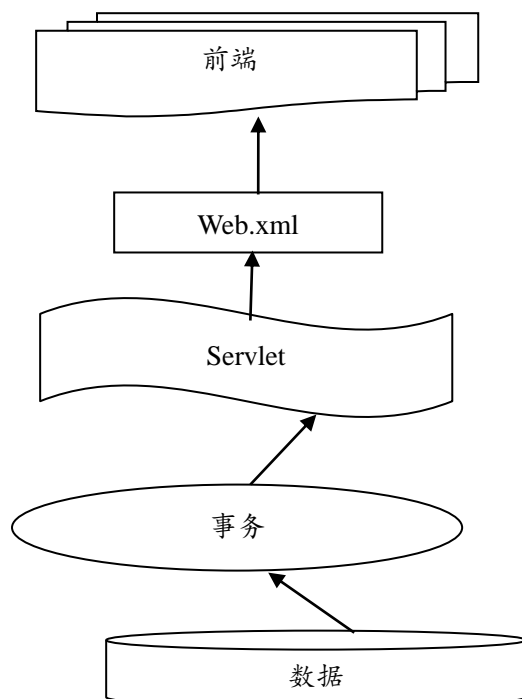


图 2-3 系统模型示意图

2.5 设计原则

系统设计所依据的原则如下：

(1) 合法性原则：依据电信资费管理系统的工作规定以及要求，参照核算实际的工作情况，进行用户管理、资费管理、账单管理、账务管理、用户自助服务等工作。

(2) 实用性原则：适合电信资费管理工作的实际需求，并能够处理一些特殊情况的要求，此外，尽可能预留空间，以便扩充功能。

(3) 准确性原则：对输入的相关资料建立检错机制，及时报错，使用户能够及时准确的输入合法资料（如类型匹配，长度超限等）。

(4) 易操作原则：要求设计的系统功能齐全，接口友好，操作方便，必要的地方进行提示。

(5) 优化原则：为了达到优化的目的，合理的运用窗口、菜单、对象等的继承，

自定义用户对象、事件、函数、减少不必要的重复性代码，使程序简洁明了，也方便了将来的维护。

(6) 安全性原则：该系统的管理员可管用户，并具有可设置登录密码，修改密码的权限。用户不能管理管理员，不具备任何对管理员操作的权限。

(7) 输入量最小原则：在保证信息量充分性条件下，让用户以最小的数据输入完成业务操作，从而让用户感到本系统的方便。

(8) 数据库字段可空原则：具有默认数值和状态的数据库字段，采用可空假定。对于可伸缩的数据库，可空字段在没有数据输入的情况下是不占空间的。

(9) 美观化设计原则：在满足实用性的条件下，界面设计做到美观大方，大小适中，排列整齐，各种控制键的中文名字简单明了，不同的业务甚至不同的功能，有不同的界面。美观的界面配色柔和，减轻视觉疲劳，给用户一个良好的计算机工作环境。

(10) 数据可维护性原则：对于正常的数据维护，管理人员可以通过电信资费管理系统本身来实现。数据维护包括权限设置、无用数据的删除、数据库的清理、数据备份与恢复等功能。

根据 MVC 设计模式和软件设计原则，电信资费管理系统设计如下：

(1) 电信资费管理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。

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

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

(4) 社会可行性分析，随着经济体制的快速发展，对于电信的业务来说，需要一个功能强大、便于管理并且能够满足客户的一般查询需求，也能满足电信公司对业务的一般管理。这种软件符合企业的实际情况，在功能操作方面容易入手。

(5) 技术可行性分析，软件要求易学，界面友好，容易掌握，可以很简单方便的对电信业务进行管理。硬件的配置要求不能太高，这样可以很好的适应电信公司的管理，也可以使用户更容易的访问系统。本系统将会用到 HTML、CSS、JavaScript、JDBC、MySQL、XML、JAVA、JSP、JavaBean、Servlet。这些都是现今比较主流的技术，利用这些技术、框架和工具来完成整个系统。

3 系统实现

对系统进行详细的需求分析与设计后，接下来要进行的是系统的实现。系统的实现需要根据设计逐步进行实现，条理清晰，有条不紊。本系统采用的是 MVC 设计模式，大部分模块都是由单独的前端与后端组合而成。系统实现前需要建立好系统的软件与硬件平台，以便系统顺利的实现。

3.1 登录界面的实现

登录界面主要是具有一定的保密性，在登录界面设有账号和密码，只有输入正确的信息后，才能使用本系统，否则也就进入不了主界面。登录界面是由一个文本框，一个密码框，两个按钮组成。文本框可输入用户的账号，密码框可输入用户密码。登录页面实现登录的数据处理功能，当用户点击“登录”按钮后，后台 Servlet 接口会先判断用户所输入的值是否正确，如果正确，则向后台发送账号和密码；如果不正确，则会再次退回到登录界面。

后台 Servlet 接口的关键代码如下。

```
if(currentUser==null){
    request.setAttribute("error", "用户名或密码错误！");
    // 服务器跳转
    request.getRequestDispatcher("/index.jsp").forward(request, response);
}else{
    // 客户端跳转
    response.sendRedirect("/frame/main.jsp");
}
}
```

若用户名或密码为空，也会再次退回到登录界面。关键代码如下。

```
if(StringUtil.isEmpty(userName) || StringUtil.isEmpty(password)){
    request.setAttribute("error", "用户名或密码为空！");
    request.getRequestDispatcher("index.jsp").forward(request, response);
    return;
}
```

本系统登录模块主要分为两部分，一部分是管理员登录，一部分是用户登录。在向后台发送数据之后，后台会根据账户编码规则分辨账户身份，之后从相应的数据库中调取相应的数据进行校验，如果校验通过则将页面跳转到相应的功能模块，如果校验不通过则向页面返回错误信息。登录页面如图 3-1 所示。



图 3-1 系统登录页面

3.2 主界面的实现

在登录界面输入正确的用户名和密码后，便进入到系统的主页面，主页面显示了系统的主要功能与各个管理模块。由框架 `frameset` 将该界面分为顶部(`top`)、左(`left`)、右(`right`)和左上(`topleft`)四个部分。该界面起引导作用，通过点击上面的选项卡按钮来打开不同的界面。点击管理模块后，对应的信息会在右部显示，这一功能是通过 `target` 实现的。当点击退出系统时，按照上面的提示就可以退出系统，退出到登录界面。在退出系统后，如果想再次进入该系统，可以进行再次登录。

与登录模块相对应，主界面有两种形式。当使用管理员账号登录时，主页面会显示管理员那部分的功能，如图 3-2 所示。



图 3-2 管理员登录页面

当用户账号登录时，那么系统就会显示用户那部分功能，如图 3-3 所示。

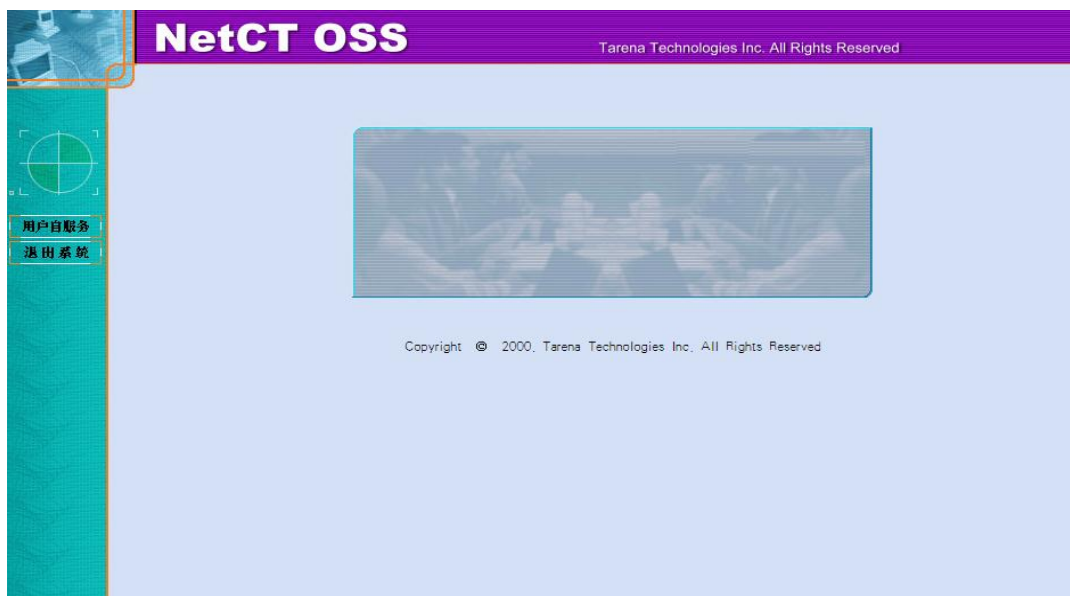


图 3-3 用户登录页面

3.3 用户管理界面的实现

用户管理界面可以实现添加新用户、查找用户信息、删除用户和修改用户信息这些功能。在添加新用户页面，采用了文本框、单选框、复选框结合的形式，以确保用户的个人信息能够完整有效的被录入到数据库中。该页面的基本信息分必填信息和选填信息两项。必填项是用户必须要填写完整的，否则的话将无法进行提交。在选填信息中用户可以自由选择是否填写该部分的信息，该部分的内容并不会影响提交的效果。

系统检查用户是否填写完必填项，关键代码如下。

```
if(userName.length()==0||password1.length()==0||password2.length()==0||name.length()==0||sex.length()==0||methord.length()==0||state.length()==0||mail.length()==0){ //判断必填项是否为空
wrong1();
response.sendRedirect("http://localhost:8084/mtt/userManage/adduser.jsp");
}else if(!(password1.equals(password2))){ //判断输入的两次密码是否相同
wrong2();
response.sendRedirect("http://localhost:8084/mtt/userManage/adduser.jsp");
}else{ //信息正确，允许登录
}
```

在提交时还会检查用户两次输入的密码是否一致，只有一致才会提交成功，否则无法完成提交。

删除用户和查找用户这两个模块都是通过代码对数据库进行操作的。删除用户通

过数据库语句直接将数据库中相关用户的信息直接删除掉，这样在页面上就不会显示被删除用户的信息。查找用户也是通过数据库语句查找到相关的用户的信息，然后将个人信息显示到页面上。

3.4 管理员管理界面的实现

管理员管理模块是用来管理管理员自己的界面，有管理员管理权限的管理人员可以在此界面上对管理员进行增、删、改、查等操作。该模块利用 `Servlet` 来连接数据库，从而实现相关的增、删、改、查的功能。在添加新的管理员时，管理员需要输入账号、登录密码、重复密码、真实姓名、管理员邮箱、联系电话这些信息，同时需要赋予新管理账号一定的权限，在权限在选择上用的是复选框，将所有会用到的管理员权限都放在复选框中，在添加新账号时选取相应的权限来赋给新账号。填写完基本信息后点击“提交”按钮，即可将新的管理员信息添加到管理员列表。另外，管理人员在选定好权限后提交到数据库时还会检查操作人员是否具有管理员管理权限，只有具有相关的权限的管理人员才能添加新的账号。相关权限的设置，确保了系统信息的安全。关键代码如下：

```
if (login == null) {
    wrong3();
    response.sendRedirect("http://localhost:8084/dinxinguanli/adminManage/admin.jsp");
} else {
    for (int i=login.size()-1;i>=0;i--) { //检查是否有管理员管理权限
        LoginBean nn = (LoginBean) login.get(i);
        manage = nn.getTariffManage();
        if (manage.indexOf("管理员管理")>=0) { if(!(password.equals(password2))){
            wrong1();
            response.sendRedirect("http://localhost:8084/dinxinguanli/adminManage/addadmin.jsp");
        }else{
            //添加成功
        }
    }
}
```

在管理员列表界面可以看到管理员的一些基本信息，如账号、姓名、电话、邮箱、开户日期、权限等。在该界面还可以删除和修改管理员信息，删除是通过单选框实现的；单击“修改”按钮，便进入修改信息界面，修改完成后点击提交即可将修改后的信息添加到管理员列表。

3.5 资费管理界面的实现

资费管理模块是对资费的管理，包括资费列表、添加资费与资费修改。资费列表

中含有资费名称、月租费用、每小时的费用以及资费说明。添加资费中也有资费名称、月租费用、每小时费用以及资费说明，不过两者的功能不同，在资费列表页面中只具有查看的功能，在添加资费页面中可以进行对资费的管理。

在修改资费页面上，管理员可以看到原始的资费信息，在原始资费信息上作出修改。这样做的优点是管理人员不用查询原始信息后再进行修改，这样就方便了对数据的操作，提升了用户体验。

将原始信息放到页面上供管理员修改的关键代码如下。

```
// 将原始资费信息显示在修改页面
<%      ArrayList pricinglist = (ArrayList) session.getAttribute("pricinglist");
if (pricinglist == null || pricinglist.size() == 0) {
} else {
for (int i = pricinglist.size() - 1; i >= 0; i--) {
PricingListBean nn = (PricingListBean) pricinglist.get(i);      %>
```

通过代码可以看出该功能的实现是通过将资费原始信息保存在相关的表里，然后保存在系统 Session 对象中，然后在修改页面调用 Session 对象中相应的表即可将原始信息显示出来。在管理员作出修改工作后就会更新已保存在表里的信息，这样在下次进入更改页面时就会显示新的资费信息。

添加资费是将相关的资费信息输入到对应的文本框中，确认无误后单击“提交”按钮即可添加成功。添加成功的资费信息会在资费列表中显示出来。

3.6 账单管理界面的实现

账单管理界面主要是通过数据库的查询实现的。在账单查询中，管理员需要提供查询用户的账务账号，并且选择好需要查询的时间，然后点击查询即可查找到该用户在选定时间里的账单。还可以通过“明细”按钮进入详细清单页面，这里可以看到用户每天的使用时间。

3.7 账务管理界面的实现

账务管理界面实现了按照月、年分别生成《月账务信息统计报表》和《年账务信息统计表》。该管理模块包括服务器月账务查询和服务器年账务查询这两个界面。

在服务器月账务查询中，管理人员可以查询到系统中每个服务器每月的使用情况，并且在账务明细中可以查询到该月每天的服务器使用情况。

在服务器年账务查询中，管理人员可以查看到每一年的服务器使用情况，需要选定需要查询的年份，单击年账务查询即可查找到系统所有服务器的使用情况。并且在清单后面的详细清单里可以看到这一年里每个月服务器的使用情况，方便管理人员对服务器的管理和对服务器的调试运行。

3.8 用户自服务管理界面的实现

用户自服务管理模块是为用户而设计的，所有的用户都有权限来操作自己的信息。在该模块用户可以进行账单的查询和个人信息的修改。

在个人信息修改中，单击“修改个人信息”就可以进入到信息修改页面，在该界面用户可以修改自己的个人账户信息，包括密码、姓名、联系方式、电子邮箱、支付方式、职业、性别、公司、公司邮箱、邮编等。需要修改个人信息的用户需要在自己原始的信息上面直接修改，这样可以有效的防止用户忘记原始信息而做出错误的修改，方便了用户的操作体验，使该系统具有更好的用户体验。修改完成后单击“修改”按钮即可完成对自己信息的修改和完善，同时这些信息会被同步到管理员页面中的用户列表中。

在账单查询中，用户可以查询到具体日期的账单情况，其中年份和月份是采用的下拉框的形式供用户选择的，这样可以方便用户的操作。选定好查询的条件后点击“账单查询”按钮便可以查询到当前用户当月资费的使用情况，包括登录时长和费用。在数据的后面还有一个“明细”按钮，里面是用户具体使用情况，包括业务账号、服务器、总计和总费用。用户能够在这里清楚的看到自己月账单的使用情况以便合理安排资费的使用。

3.9 退出系统界面的实现

退出界面是由一个超链接组成的，这个超链接直接连接到了登录界面，这个界面的实现相对简单。在这个超链接直接连接到了登录页面，也就实现了相应的退出系统的功能。退出系统的关键代码如下。

```
<a href="../index.jsp" target="_parent" onMouseOver="msover(7) ; return true"
onMouseOut="msout(7)">
 </a>
```

3.10 数据库的实现

在数据库连接时，为了提高代码的利用率，采用了统一的数据库连接类统一连接数据库。关键代码如下。

```
Class.forName("com.sql.jdbc.Driver");  
    String  
url="jdbc:mysql://localhost:3306/dianxin?useUnicode=true&characterEncoding=gbk";  
    con=DriverManager.getConnection(url,"root","123456");  
    st=con.createStatement();
```

4 项目总结

本次项目名称为基于 MVC 模式的电信资费管理系统，其产生的背景为：随着电信的重组，国内各运营商面临着严峻的市场竞争形势，市场主体的多样化和用户需求的个性化已经使传统的电信业务受到了严重的冲击。如何提高运营效率、降低运维成本、改善服务质量、提高服务水平，是运营商迫切需要解决的问题。

MVC 设计模式是一种目前广泛流行的软件设计模式，它把一个应用的输入、处理、输出流程按照 Model、View、Controller 的方式进行分离。MVC 模式对于 Web 应用的开发是一种非常先进的设计思路，无论选择哪种语言，无论应用多复杂，MVC 为构造产品提供清晰地设计框架，为软件工程提供规范的依据。

本系统通过 MVC 设计模式，前端采用 JSP 技术，通过 xml 配置连接后台 Servlet 接口，后台 Servlet 通过调用相应的程序来完成操作与控制。

本系统的需求包括功能需求和性能需求。其功能需求有：用户管理需求、管理员管理需求、资费管理需求、账单管理需求、账务管理需求和用户自服务管理需求。在性能上尽量做到实时性强、数据容量小、响应速度快、稳定性高、出错率低、容错性好。实现以下几点需求：站点中网页的外观设计和内容应相互独立；站点要有统一的风格和布局：使用母版页、内容页、主题；站点要为用户提供方便的站点导航。

本系统的主要功能包括登录、用户管理、管理员管理、资费管理、账单管理、账务管理、用户自服务管理和退出系统。下面对这些主要功能进行详细介绍。

(1) 登录管理：实现登陆后方可使用该应用程序。用户在登录界面输入正确的用户名和密码后，便进入到系统的主页面，主页面显示了系统的主要功能与各个管理模块。

(2) 用户管理模块：用户管理模块可以实现添加新用户、查找用户信息、删除用户和修改用户信息这些功能。

(3) 管理员管理模块：管理员界面是用来管理管理员自己的界面，有管理员管理权限的管理人员可以在此界面上对管理员进行增、删、改、查等操作。

(4) 资费管理模块：资费管理模块可以查看资费信息、添加资费和修改资费。

(5) 账单管理模块：账单管理模块可以查询个人用户月账单的使用情况，包括登录时长、费用、状态等。

(6) 账务管理模块：账务管理模块可以查询到系统中每个服务器的使用情况。

(7) 用户自服务管理模块：用户自服务管理模块是为用户而设计的，所有的用户都有权限来操作自己的信息，在该模块用户可以进行账单的查询和个人信息的修改。

(8) 退出系统模块：退出系统模块直接连接到了登录页面，实现退出系统的功能。

另外，数据库是电信资费管理系统中非常重要的环节之一，一个结构良好的数据库可以很好的实现系统的查询和管理，并且能够极大的提高系统的运行效率。本系统使用 MySQL 数据库进行连接，通过数据库来完成数据的增加、查询、删除的功能。

5 实训感想

本次实训历时三周，在这三周中我感慨颇多。从最开始的茫然无措到最后的一点收获，这个过程是很奇妙的。最先接触项目无从下手，有些焦虑，每次实训都是一个大的考验，也是对自己所学知识的一个巩固，在过程中也会有收获新知的喜悦。在实训的这几周也对未来的职业有了更深一步的了解。

最开始接触项目的时候，说实话对我来说是有些难的，因为这是一个完整的系统，不再是单独的一个个的小程序，这就需要你进行整合，使系统具有连贯性，同时还不能过于单调。因此在实训的第一天几乎就没怎么进行下去，所幸书上有相似的例子，可以借助课本完成初步的页面设计。慢慢发现代码的功能实现也不是想象中的那么难，只要熟练掌握了一个页面的设计，接下来的工作就会减轻很大的难度。

对于项目的实现难的不是页面的呈现，而是怎样使静态的页面与数据库连接起来使之成为动态的。当页面编写好时，它只是那么一个页面，不具备其他的功能，它需要从数据库中获取数据来呈现到页面上，这样才算是一个基本页面所具有的功能。

经过这次实训确实学到了不少，也从中得到了一些教训，那就是一定要细心，不能急躁，一定要缓下心，慢慢来。因为这些程序的实现是由一个个的字母拼接成的，所以中一点点的差错都会导致整个程序无法实现。

在实训中的得到了许许多多的感悟。很多时候一个人是很难成就大事的，需要大家齐心协力来合作完成，一个人的知识与学识总是有限的，想要很好的完成任务就得需要去不断的学习新的知识来充实自己，同周围的人交流探讨，一个人的思路总是不如大家合力讨论得来的结果。同时学习是没有尽头的，单单靠课本上的内容是不足以支撑整个项目的完结，需要不断的查阅资料，询问老师与同学，一次次的演变，一次次的修改才会找到最为合适的代码，设计出更为优化的界面。

在实训中体会到了程序员的艰辛与不易，一个标点符号的出错就会导致整个程序无法正常运行，因此在敲代码的过程中一定要细心，认真检查，不慌不忙，不能操之过急，急于求成，正所谓欲速则不达。在项目中一定要细心。同时也需要耐心，在出现错误的时候一定要耐下心来找错，不能急躁，保持头脑清晰，才会如期完成任务。