分类号：TP315UDC：D10621-408-(2018)2284-0

密　级：公开编号：**2014053030**

成都信息工程大学

学位论文

百货公司报价系统的设计与实现

|  |  |
| --- | --- |
| 论文作者姓名： | 徐雯俊 |
| 申请学位专业： | 数字媒体技术 |
| 申请学位类别： | 工学学士 |
| 指导教师姓名（职称）： | 廖德钦（副教授） |
| 论文提交日期： | 2018年5月10日 |

百货公司报价系统的设计与实现

**摘　要**

本系统研究的目的是了解企业ERP的管理需求，结合线上线下的售货服务的沟通，在零售方式上探索一种新的方法。本系统使用了Asp.Net技术作为支持，B/S架构为模式，并且使用了MySql作为后台的数据库，设计百货公司报价系统，系统主要包含产品管理系统，用户管理系统，报价管理系统，消息通知系统，权限管理系统五个子系统；整个系统界面布局十分友好易懂，用户能够很容易的使用该系统，只要有一个浏览器即可完成各项工作，十分容易操作。最终，本项目实现了通过线上便捷的操作，来取代平时繁琐的线下操作，实现便捷的报价询价工作。系统的实现成果已经能够很好的取代传统的报价操作，极大了方便了供货商和代理商之间的业务沟通交流，能更好的适应“新零售”时代的业务需求。

**关键词：**B/S架构；数据库；Asp.Net；新零售

Design and Implementation of Department Store Quote System

**Abstract**

The purpose of this system research is to understand the management requirements of enterprise resource planning, to improve the communication between online and offline users, and also to explore a new way of retailing. This system uses Asp.Net as technical support, uses browser and server schema, and uses MySql database. The system has five subsystems, including the product management system, user management system, offer management system, message notification system, and rights management system. The layout of the whole system interface is very friendly and easy to understand and use. Users only need to spend little time studying the operation of this system. It’s very easy for Users to complete all the work by using the browser. Finally, this project has realized the convenient operation on the Internet, it can replace the traditional cumbersome offline operation. And it also has realized the convenient quotation operation. This system can replace the traditional quotaion operation, and it’s convenient for suppliers and distributors to have more communication. And it can also adapt the business requirements of a new way of retailing.

**Key words:** b/s model;database;asp.net; the “new retail” era

**目 录**

论文总页数：31页

[1 引言 1](#_Toc514418198)

[1.1 课题背景 1](#_Toc514418199)

[1.2 国内外研究现状 1](#_Toc514418200)

[1.3 本课题研究的意义 1](#_Toc514418201)

[1.4 本课题的研究方法 2](#_Toc514418202)

[2 需求分析及项目方案 2](#_Toc514418203)

[2.1 需求分析 2](#_Toc514418204)

[2.1.1 功能需求 2](#_Toc514418205)

[2.1.2 系统约束条件 5](#_Toc514418206)

[2.2 项目方案 5](#_Toc514418207)

[2.2.1 项目特色 5](#_Toc514418208)

[2.2.2 ASP.NET技术简介 6](#_Toc514418209)

[2.2.3 Entity Framework6技术简介 6](#_Toc514418210)

[2.2.4 开发工具和运行环境 6](#_Toc514418211)

[2.2.5 项目实施计划 7](#_Toc514418212)

[3 项目设计及实现 8](#_Toc514418213)

[3.1 数据库设计 8](#_Toc514418214)

[3.1.1 总模型图 8](#_Toc514418215)

[3.1.2 主要的表结构 8](#_Toc514418216)

[3.2 功能模块的设计与实现 13](#_Toc514418217)

[3.2.1 系统整体框图 13](#_Toc514418218)

[3.2.2 登录管理模块的设计与实现 14](#_Toc514418219)

[3.2.3 用户管理模块的设计与实现 15](#_Toc514418220)

[3.2.4 产品管理模块的设计与实现 17](#_Toc514418221)

[3.2.5 报价管理模块的设计与实现 18](#_Toc514418222)

[3.2.6 权限管理模块的设计与实现 20](#_Toc514418223)

[3.2.7 信息管理模块的设计与实现 21](#_Toc514418224)

[4 特殊问题及解决方案 23](#_Toc514418225)

[4.1 报价单问题 23](#_Toc514418226)

[4.1.1 问题描述 23](#_Toc514418227)

[4.1.2 解决方案 23](#_Toc514418228)

[4.1.3 结果 23](#_Toc514418229)

[4.2 图片上传问题 23](#_Toc514418230)

[4.2.1 问题描述 23](#_Toc514418231)

[4.2.2 解决方案 23](#_Toc514418232)

[4.2.3 结果 23](#_Toc514418233)

[4.3 Excel文件问题 24](#_Toc514418234)

[4.3.1 问题描述 24](#_Toc514418235)

[4.3.2 解决方案 24](#_Toc514418236)

[4.3.3 结果 24](#_Toc514418237)

[5 结果测试及性能分析 24](#_Toc514418238)

[5.1 测试概要 24](#_Toc514418239)

[5.1.1 测试环境 24](#_Toc514418240)

[5.1.2 测试范围 25](#_Toc514418241)

[5.2 性能分析 25](#_Toc514418242)

[项目总结 26](#_Toc514418243)

[结 束 语 27](#_Toc514418244)

[参考文献 29](#_Toc514418245)

[致　　谢 30](#_Toc514418246)

[声　　明 31](#_Toc514418247)

# 引言

## 课题背景

在信息化时代到来的今天，各个企业都开始将传统的商业模式向信息化转型，ERP技术受到了越来越多的企业的重视。ERP即企业资源计划，通常是指建立在信息技术的基础之上，集合信息技术和先进的管理思想于一体，以系统化的管理思想，为企业员工及决策层提供决策手段的管理平台，ERP的作用是能够让企业对企业的各个项目数据运行的更加的全面，更加的准确，更加的及时，能够让企业更加科学化，更加合法合理化，更佳制度化，所有企业如果想走向信息化的道路，ERP系统是其中关键的一步。ERP系统是一个十分庞大的覆盖很全面的系统，主要包括了财务管理，生产控制管理，物流管理，采购管理，分销管理，库存管理，人力资源管理等等子系统，而报价系统是生产资料管理的重要一环，是值得重点研究的对象。现在各个电商开始转型，着重于线上线下的结合，逐步开始了了“新零售”的经营模式。“新零售”最开始是马云提出的，主要的意思是以互联网作为后台技术支持，通过使用先进的大数据等新兴技术，对传统的商品的生产过程，销售流转过程进行转型，结合线上线下，回归到零售的本质，所有零售这一块的工作会越来越受到各个企业的重视，会涌现出各种与零售相关的ERP功能模块，代理商与供货商之间的联系会更加紧密，而联系双方的报价系统也会被更加的重视。

## 国内外研究现状

从在网络搜索和查看与之相关的各项文献来看，国内的报价系统多是作为企业ERP中的一个功能模块存在的，和其他的各个子功能模块联系紧密，这十分方便大型企业的工作和流程的管理，但是对于小型企业和类似于代理商和供货商这一类小型的企业来说，整个ERP系统太过于庞大，有许多他们并不需要的功能模块的存在，加重了他们的使用负担；当前，现在也存在一些单独的报价系统，但是多是机加工艺，五金产业之类的报价系统，涉及到一般的日用产品，百货公司之类的产品的报价系统十分稀少，完全满足不了现有的市场需求，这是一个巨大的市场空缺。

## 本课题研究的意义

信息化是各个企业未来必然要走的发展方向，不管是大型企业还是小型企业，甚至是“夫妻店”，市场上对于大型企业的ERP系统功能已经十分的完善，能满足他们的需求，但是对于小型企业，和零售行业的整个技术支持却不是十分的完善，这是一个巨大的市场空缺，研究这一块的内容，能够让我们知道市场需求的走向，以及信息化时代的今天，各个企业的发展方向，能够让我们更好使用现有的技术去实现有价值的工作，弥补市场的空缺，建立一个完整的信息化市场，形成一个完整的销售走向的闭环；同时能降低传统的方式对资源和时间带来的巨大的消耗，能够为资源节约型社会和可持续发展社会尽一份力。

## 本课题的研究方法

本报价系统是基于Asp.Net技术开发的，使用MVC（即model-view-control）模式，采用当前十分流行的B/S(browser/server)架构进行统一开发。

前端的界面使用的是html5,css,razor,jquery,easyui,javascript等技术搭建界面，以及完成与用户的交互逻辑功能。

后台服务器使用IIS（Internet Information Services）,使用C#语言进行开发，使用MVC模式，后台数据库使用开源的免费数据库Mysql进行数据的存储，使用EF6（entity framework 6）框架进行数据库中数据的处理。

# 需求分析及项目方案

## 需求分析

### 功能需求

本报价管理系统的主要针对供应商和代理商之间的报价操作，主要的角色有管理员，内部人员，外部人员；外部人员可以查看和修改自己的用户信息，并且还可以查看产品信息，以及根据需要生成报价单，以及修改和删除报价单，也可以下载已经审核通过的报价单。而内部人员可以查看和修改自己的用户信息，还可以编辑用户信息，添加编辑删除用户，并且还可以查看产品信息，修改编辑产品信息，以及上传产品信息，以及可以审核外部人员添加的报价单，也可以下载已经审核通过的报价单。而管理员有所有的功能权限，同时管理员还可以添加编辑系统的角色，编辑系统用户信息，以及调整系统的预售价数量的数据，管理员还拥有权限功能，可以赋予和修改各个角色不同的权限功能；此外所有的用户都可以查看自己的所接收到的消息，以及管理个人的信息系统，比如一键已读和删除已读的功能。

本系统分模块实现，系统主要包括用户管理模块，产品管理模块，报价管理模块，权限模块，信息模块五大模块。

1. 用户管理模块

在这个模块中，管理员或者其他拥有权限的人，可以进行角色的管理，包括添加系统角色、修改角色信息、删除已经废弃的系统角色，以及查看已有角色的详细信息；也可以进行使用系统的用户的管理，包括添加、修改、删除用户，以及查看用户的详细信息，可以自定义该系统的用户信息；除此之外，所有的用户都可以查看和修改用户自己的资料信息，以及修改自己的登录密码等。本模块的用例图如图2- 1所示。



图2- 1 用户管理模块用例图

1. 产品管理模块

在这个模块中，管理员或者其他拥有权限的人，可以进行产品的管理，包括产品的创建、修改、删除、批量上传，以及产品分类的增添、删除、修改操作等功能。而其他没有权限的人，或者是系统的外部人员只能够查看各个产品的详细信息，以及各个产品所对应用户的售价情况和分类的信息，无权改动。本模块的用例图如图2- 2所示。



图2- 2 产品管理模块用例图

1. 报价管理模块

在这个模块中，内部人员可以查看和审核外部人员提交订单，亦可以下载审核通过的订单；外部人员可以添加订单，以及对为审核通过的订单进行重新编辑以及删除操作，也可以下载已经通过审核的订单。管理员或者是其他具有权限的用户可以设置外部用户的售价的自动升级和降级的判断规则。所有用户可以查看订单详情，也可以查看报价单按月统计的结构信息。本模块的用例图如图2- 3所示。



图2- 3报价管理模块用例图

1. 权限模块

在这个模块中，管理员或者其他具有权限的用户可以查看不同等级角色所具有的不同模块的具体的操作权限，以及设置不同等级的角色的权限。本模块的用例图如图2- 4所示。



图2- 4权限模块用例图

1. 信息模块

在这个模块中，所有用户可以查看自己收到的信息，可以根据信息的标题或者是信息的未读或者已读状态筛选信息进行查看，当用户选择某一条消息时，可以查看这一条信息的详情信息，除此之外用户可以删除不需要的信息；并且也可以批量的操作信息，包括了一键已读，删除已读等批量操作。本模块的用例图如图2- 5所示。



图2- 5信息模块用例图

### 系统约束条件

本系统的使用范围是针对销售日用百货的供应商和代理商之间，用于替代供应商和代理商之间的传统的繁琐的线下的报价询价操作，以及线下的业务的交流等。

## 项目方案

### 项目特色

1) 因为项目采用的是B/S架构，通过部署在IIS上，只需要依赖一个浏览器即可访问，所以可以兼容各个操作平台。

2) 系统采用MVC模式，将显示和逻辑控制分离，使得系统整个的框架功能划分十分清除，十分方便后续更多功能的添加。

3) 页面大量使用了Razor标签，使得很多界面可以重复使用，方便了CSHTML页面和Web程序的开发。

4) 本项目在需要修改大量数据的地方提供了上传文件功能，简化了繁琐的依次录入功能。

5) 通过统计功能，能够方便的看每月订单统计结果，无需人工手动通过，方便快捷。

6) 本项目使用的第三方功能库均是开源免费的，不仅在功能上能满足要求，并且极大的降低了整个系统的开发成本，且这些软件使用人群非常广泛，所以在遇到问题时，能方便的找到答案。

7) 本项目能够对产品图片进行放大展示，当小图看不清是的时候，能够通过放大功能让用户看到图片的细节功能。

8) 本项目前端界面布局合理，操作简单，学习成本极低，很轻松就能够上手。

9) 本项目可以很方便的通过Visual Studio部署到本地的IIS服务器上，并且如果后续有功能改动，只需复制相关文件到服务器即可，整个过程十分的方便快捷。

### ASP.NET技术简介

ASP.NET 是一种已编译的、基于 .NET 的环境，只要编程语言兼容于.NET，就可以用其编写应用程序。ASP.NET有许多优点，让它成为了到目前为止最为流行的Web开发技术之一。ASP.NET十分易于管理，它的网站的配置信息的存储方式是存文本的，所以可以直接使用文本工具即可更改新的设置，而不需要启动本地的管理工具；此外它也是十分简单易学的，ASP.NET原生支持很多对网站的处理的库，比如网站的用户身份的验证，授权管理的库，使得开发人员能够很容易的使用，学习成本很低；并且，ASP.NET的部署也是十分方便的，结合Visual Studio开发工具，只需要简单几步即可将网站发布出去，当数据有改动时，只需要将相应的文件复制到服务器上即可；对于开发人员来说，ASP.NET也是非常容易调试的，结合Visual Studio工具，能轻易的使用系统功能的调试和页面数据的跟踪。

### Entity Framework6技术简介

Entity Framework6是一种由微软开发的，是一种域对象和数据存储对象映射关系的处理的框架。它使得开发人员编写和访问数据库变得十分容易。EF框架摆脱了SQL,使用了一种形式上与SQL十分相似的结构，名叫Linq，通过Linq，程序员能够完成对数据库的各种操作。

EF有两种常用对控制数据库的方式，一种是Code First，另一种是Database First，Code First是目前比较流行的一种方式，当使用Code First时，系统会有一套自己默认的约定规则，能够根据数据模型生成相应的数据表和表之间的关系；而Database First则是先使用SQL语句建立数据库和相关的表结构以及表与表之间的关系，然后在通过EF选择相应的数据库，生成对应的数据模式。

EF相比与传统的SQL语句或者ADO.NET，最大的优点就是极大的加快了开发者的开发效率，只要开发者熟悉了EF框架的使用规则，只需要极少量的代码，就能做完成之前模式下需要很多代码才能完成的相同的功能的数据交互。但是另一方面，EF框架也不是完美的，因为EF的原理是通过代码来生成SQL语句，然后通过SQL语句再来执行相应的操作，所以相比于原生的SQL语句来说，性能肯定是比不上的，所以在使用EF框架开发时，需要格外的注意性能的问题才行。

### 开发工具和运行环境

系统开发工具和开发环境如表2- 1所示。

表2- 1开发工表

|  |  |
| --- | --- |
| 开发工具名称 | 软件用途 |
| Asp.net MVC5 | 系统环境 |
| VS2015 | 开发平台 |
| IIS | 服务器 |
| Microsoft Visio 2013 | 系统设计，创建流程图等 |
| Mysql | 系统后台数据库 |

### 项目实施计划

本系统从3月8号正式开始启动，到5月1号为止，项目确定的所有需求功能全部完成。

首先是项目的前期准备，包括项目的主体的前端框架和数据库的设计和建立，这个过程大概花费了8天的时间。然后开始分模块逐步完善系统的功能，首先就是用户登录模块的完成，这个过程大概花费了2天时间，在系统能够获得所需要的用户数据后，继续进行下一步操作；下一步开始完成用户管理模块，包括了用户角色，用户资料的管理以及个人信息管理，这个过程大概花费5天时间；再下一步就是产品模块的管理，包括了产品的分类和信息管理，这个过程大概花费5天；然后就是报价模块的管理，主要包括了报价单的生成、审核以及下载，还有报价单的统计操作，这个过程大概花费了6天；然后就是权限模块，主要是控制不同角色的权限，这个过程大概花费了3天时间；最后就是信息模块，能够查看和管理自己的信息，这个过程花费了3天；当主体功能完成后，就对项目进行调优工作，并逐步清理系统Bug，最后把系统部署到本地IIS上，这个过程花费了5天时间。实施计划的甘特图如图2- 6所示。

图2- 6项目实施计划

# 项目设计及实现

## 数据库设计

### 总模型图

本系统根据需要，在数据库一共设计了15张表，表所有的表的总模型图如图3- 1所示。

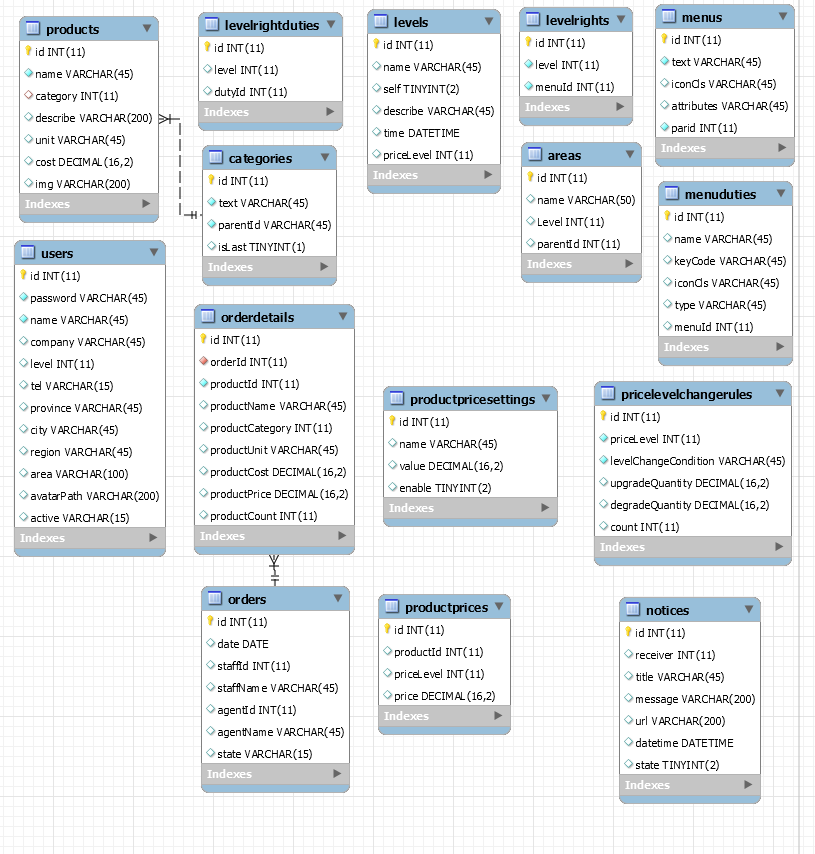


图3- 1总模型图

### 主要的表结构

Level表的功能是存储系统角色信息。Level表的表结构如表3- 1所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 1 系统角色表 | | | | | | | |
| 中文名称 | | 系统角色表 | | | | | |
| 物理表名 | | Level | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 名字 | | PriceLevel | Varchar(45) |  | 是 |  |
| 3 | 是否是内部人员 | | Self | TINYINT(2) |  | 是 |  |
| 4 | 详细描述 | | Describe | Varchar(45) |  |  |  |
| 5 | 时间 | | Time | DATETIME |  |  |  |
| 6 | 售价等级 | | PriceLevel | Int |  |  |  |

User表主要的作用是存储用户的账号信息，主要包括了账号密码，地址信息等。User表结构如所表3- 2示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 2用户表 | | | | | | | |
| 中文名称 | | 用户表 | | | | | |
| 物理表名 | | users | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 序列号 | | Id | int | 是 | 是 |  |
| 2 | 密码 | | Password | Varchar(45) |  | 是 |  |
| 3 | 姓名 | | Name | Varchar(45) |  | 是 |  |
| 4 | 公司 | | Company | Varchar(45) |  |  |  |
| 5 | 等级 | | Level | Varchar(15) |  |  |  |
| 6 | 电话 | | Tel | Varchar(15) |  |  |  |
| 7 | 省 | | Province | Varchar(45) |  |  |  |
| 8 | 市 | | City | Varchar(45) |  |  |  |
| 9 | 区 | | Region | Varchar(45) |  |  |  |
| 10 | 详细地址 | | Area | Varchar(200) |  |  |  |
| 11 | 头像地址 | | AvatarPath | Varchar(200) |  |  |  |
| 12 | 状态 | | State | Varchar(15) |  |  |  |

Categories表的主要作用是存储产品的分类信息。Categories的表结构如表3- 3所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 3 分类表 | | | | | | | |
| 中文名称 | | 分类表 | | | | | |
| 物理表名 | | categories | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 名字 | | text | Varchar(45) |  | 是 |  |
| 3 | 父节点 | | ParentId | Int |  | 是 |  |
| 4 | 是否是最后节点 | | Islast | tinyint(1) |  |  |  |

Orders表的主要作用是存储报价单总体情况。Orders表的结构如表3- 4所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 4 报价单表 | | | | | | | |
| 中文名称 | | 报价单表 | | | | | |
| 物理表名 | | Orders | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 时间 | | Date | date |  | 是 |  |
| 3 | 处理人索引 | | Staffid | Int |  |  |  |
| 4 | 处理人名字 | | Staffname | Varchar(45) |  |  |  |
| 5 | 代理商索引 | | Agentid | Int |  |  |  |
| 6 | 状态 | | State | Varchar(15) |  |  |  |

Orderdetails表的主要作用是存储报价单每一单的产品详情。Orderdetails的表结构如表3- 5所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 5订单详情表 | | | | | | | |
| 中文名称 | | 订单详情表 | | | | | |
| 物理表名 | | Orderdetails | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 订单编号 | | Orderid | Int |  | 是 | 是 |
| 3 | 产品编号 | | ProductId | Int |  | 是 | 是 |
| 4 | 产品名称 | | Productname | Varchar(45) |  |  |  |
| 5 | 产品分类 | | Productcategory | Int |  |  |  |
| 6 | 产品单位 | | productunit | Varchar(45) |  |  |  |
| 7 | 产品成本 | | Productcost | Decimal(16,2) |  |  |  |
| 8 | 产品售价 | | Productprice | Decimal(16,2) |  |  |  |
| 9 | 产品数量 | | Productcount | Int |  |  |  |

Product表的主要作用是存储产品的基础信息包括产品单位，分类等等，以及产品的成本信息，但是不包括产品的售价信息。Product的表结构如表3- 6所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 6 产品表 | | | | | | | |
| 中文名称 | | 产品表 | | | | | |
| 物理表名 | | Products | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 名字 | | Name | Varchar(20) |  | 是 |  |
| 3 | 分类 | | Category | Int |  |  | 是 |
| 4 | 备注 | | Describe | Varchar(200) |  |  |  |
| 5 | 单位 | | Unit | Varchar(45) |  |  |  |
| 6 | 成本 | | Cost | Decimal(16,2) |  |  |  |
| 7 | 图片 | | Img | Varchar(200) |  |  |  |

PriceLevelChangeRule表的主要功能是存储代理商所对应产品价格等级的自主升级和降级的判断规则的信息。PriceLevelChangeRule的表结构如表3- 7所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 7价格等级改变规则表 | | | | | | | |
| 中文名称 | | 价格等级改变规则表 | | | | | |
| 物理表名 | | PriceLevelChangeRule | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 价格等级 | | PriceLevel | Int |  | 是 |  |
| 3 | 改变条件 | | LevelChangeCondition | Varchar(45) |  | 是 |  |
| 4 | 升级数值 | | Productname | Decimal(16,2) |  |  |  |
| 5 | 降价数值 | | Productcategory | Decimal(16,2) |  |  |  |
| 6 | 生效次数 | | productunit | Int |  |  |  |

ProductPrices表的主要的功能是存储产品的各个等级的售价信息。ProductPrices的表结构如表3- 8所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 8产品价格详情表 | | | | | | | |
| 中文名称 | | 产品价格表 | | | | | |
| 物理表名 | | ProductPrices | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 产品编号 | | ProductId | Int |  | 是 | 是 |
| 3 | 价格等级 | | PriceLevel | Int |  | 是 |  |

ProductPriceSetting表的主要功能是存储产品价格规则的设置信息。ProductPriceSetting的表结构如表3- 9所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 9产品价格设置表 | | | | | | | |
| 中文名称 | | 产品价格设置表 | | | | | |
| 物理表名 | | ProductPriceSetting | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 名字 | | Name | Varchar(45) |  | 是 |  |
| 3 | 值 | | Value | Decimal(16,2) |  | 是 |  |

LevelRight表的功能是存储权限信息。LevelRight表结构如表3- 10所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 10角色权限表 | | | | | | | |
| 中文名称 | | 角色权限表 | | | | | |
| 物理表名 | | LevelRight | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 角色索引 | | Level | Int |  | 是 | 是 |
| 3 | 菜单索引 | | MenuId | Int |  | 是 | 是 |

Notices表主要功能是存储用户的消息。Notices的表结构如表3- 11所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表3- 11消息表 | | | | | | | |
| 中文名称 | | 消息表 | | | | | |
| 物理表名 | | Notices | | | | | |
| 主键 | | id | | | | | |
| 业务主键 | | 无 | | | | | |
| 所属表空间 | | Quotation\_system | | | | | |
| 索引 | | IND\_TBL\_USER\_PK(主键索引) | | | | | |
| 字段列表 | | | | | | | |
| 序号 | 中文名称 | | 列名 | 数据类型 | 主键 | 非空 | 外键 |
| 1 | 索引 | | Id | Int | 是 | 是 |  |
| 2 | 接收者 | | Receiver | Int |  | 是 |  |
| 3 | 标题 | | Title | Varchar(45) |  | 是 |  |
| 4 | 内容 | | Message | Varchar(200) |  |  |  |
| 5 | 跳转链接 | | Url | Varchar(200) |  |  |  |
| 6 | 时间 | | Datetime | DATETIME |  |  |  |
| 7 | 状态 | | State | TINYINT(2) |  |  |  |

## 功能模块的设计与实现

### 系统整体框图

系统整体框图结构如图3- 2所示。

图3- 2 系统整体框架图

### 登录管理模块的设计与实现

A）功能描述

登录模块的主要功能是实现用户登录时验证用户是否是系统的合法用户。

B）流程设计

仅以用户登录为例，流程设计如图3- 3所示。



图3- 3 用户登录流程图

C）页面截图

仅以用户登录界面为例，界面如图3- 4所示。



图3- 4 用户登录界面

D）主要代码

仅以用户登录账号密码查询为例，代码如下。

public static User AccountLogInResult(int id, string password){

string encrypt = Md5Encrypt.Encrypt(password);

DbOperator dbOperator = new DbOperator();

User user = dbOperator.users.SingleOrDefault(model => model.id == id && model.password == encrypt);

return user;

}

### 用户管理模块的设计与实现

A）功能描述

用户管理模块主要的功能包括了角色的管理、用户信息管理、以及用户自己信息的管理功能等功能。角色管理主要的功能是管理员或者其他有权限的人可以根据本系统的实际情况，来创建和修改符合本系统的角色。用户信息管理主要是管理各个用户的信息，能够根据需要调整用户的信息，并为新建用户分配角色等级等，在这个模块中，具有权限的用户还具有更改用户的账号状态，重置密码等等功能。用户个人信息管理针对本用户的信息，用户可以查看或更改自身的信息。

B）流程设计

仅以角色创建为例，流程设计如图3- 5所示。



图3- 5 角色创建流程图

C）界面截图

仅以角色创建界面为例，界面如图3- 6所示。

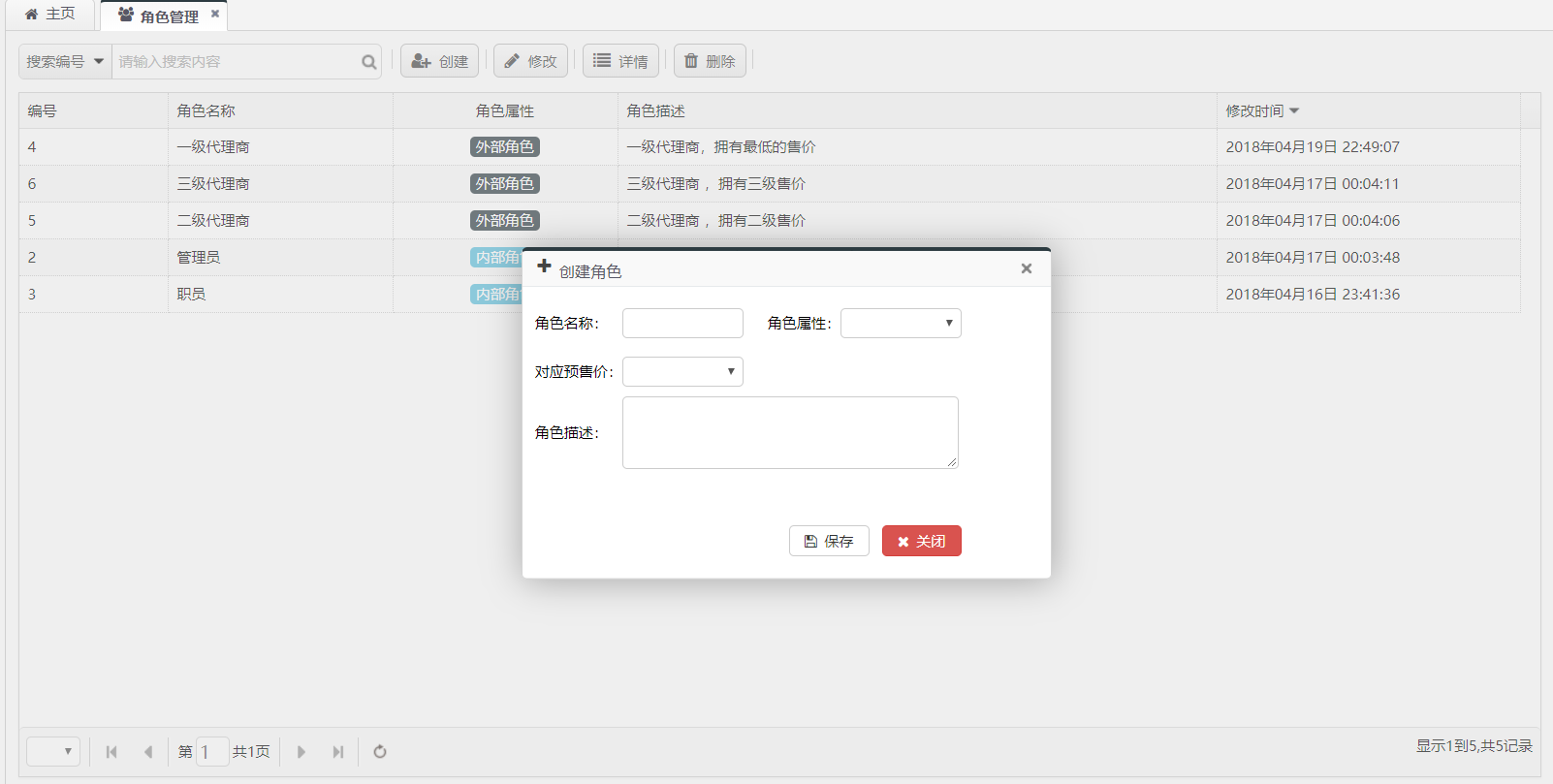


图3- 6 角色管理界面

D）主要代码

仅以系统角色创建为例，代码如下。

$("#save").click(function () {

if (emptyFocus("#name")) return

if (emptyFocus("#self")) return

if ($("#self").val() == "false") {

if (emptyFocus("#price"))

return

}

$.post("/SystemUser/EditLevel", {id: $("#id").val(), name: $("#name").val(), describe: $("#describe").val(), self: $("#self").val() ,priceLevel:$("#price").val()}, function (data) {

if (data.result) {

$("#modalwindow").window("close");

showTip("修改成功", 2000)

datagridReload("#levelList");;

}

})

})

### 产品管理模块的设计与实现

A）功能描述

产品管理模块的主要的功能包括了产品管理，以及产品的管理，和产品的售价管理，产品售价管理只针对有权限的人可以查看。产品管理主要包括了产品的分类管理，能够自由的调整产品的分类；还有产品的信息管理，具有权限的人可以添加或修改产品的信息，或者批量删除产品，以及批量的修改产品的分类等等，当点击产品图片时，有放大显示的效果能让用户更清楚的看到图片信息，可以通过Excel上传，批量的操作。对于外部人员，只拥有查看的权限，不能更改信息。

B）流程设计

仅以产品分类更改为例，流程设计如图3- 7所示。



图3- 7 产品分类更改流程图

C）界面截图

仅以产品管理界面为例，界面如图3- 8所示。

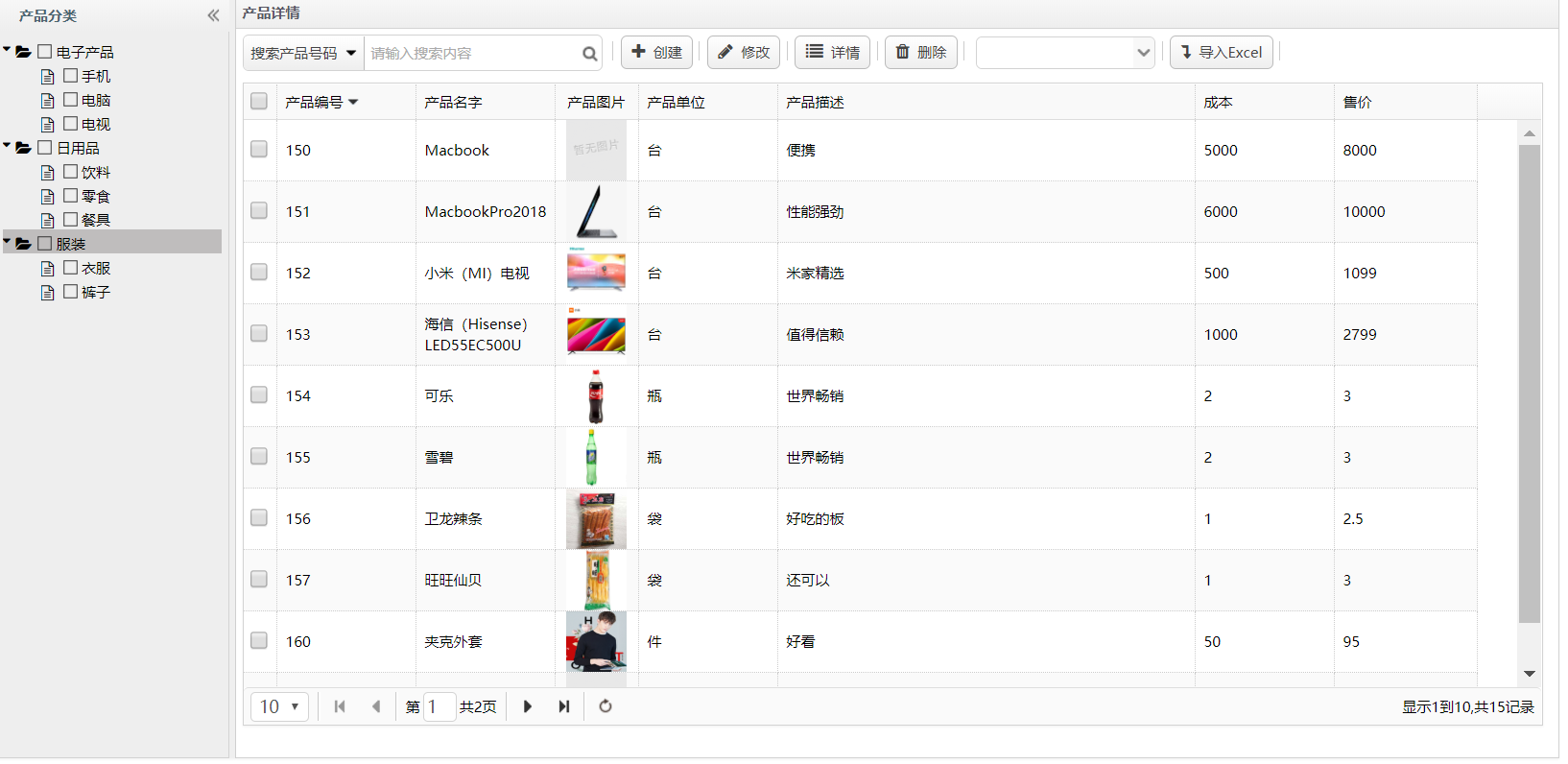


图3- 8 产品管理界面

D）主要代码

仅以产品分类移动为例，代码如下。

onAfterEdit: function (node) {

var preNode = findAndDeleteNode(node.id)

if (preNode.text != node.text)

{

$.post("/SystemProduct/EditCategoryText", { id: node.id, text: node.text }, function (data) {

showTip(data.result? "修改成功":"修改失败", 500);

});

}

},

### 报价管理模块的设计与实现

A）功能描述

报价管理模块主要包括了报价单操作，根据已经通过审核的报价单的统计结果查看操作，以及代理商售价升降级操作，该操作只有具有权限的用户或者是管理员可以查看和修改。报价单操作对应不同的用户显示不同的功能，对于内部人员，用户可以审核外部人员提交的订单；对于外部人员，用户可以生成订单，以及可以编辑未审核通过的订单重新提交，也可以删除未审核通过的订单；此外所有的用户均可以查看订单的详情，也可以下载已经审核通过的订单，下载的订单也会根据内部外部人员而显示出不同的报价单数据，内部人员会显示利润。报价单统计是显示已经通过审核的订单汇总情况。代理商售价升降级操作主要是通过制定不同售价等级的代理商的升降级规则，每月初系统会统计生效次数月份的销售情况，然后判断该代理商是否满足升级降级操作，如果满足就修改相应的等级。

B）流程设计

仅以查看每月报价单报表为例，流程设计如图3- 9所示。



图3- 9 查看每月报价单报表流程图

C）界面截图

仅以报价单统计界面为例，界面如图3- 10所示。



图3- 10 报价单统计界面

D）主要代码

仅以按分类统计当月报价单为例，代码如下。

public static GridRows<Order> GetOrderPassList(string date){

DbOperator dbOperator = new DbOperator();

GridRows<Order> result = new GridRows<Order>(); IEnumerable<Order> list = dbOperator.orders.Where(model => model.state == "审

核通过");

if (!string.IsNullOrWhiteSpace(date)){

DateTime time = DateTime.Parse(date);

list = list.Where(model => model.date.Year == time.Year && model.date.Month == time.Month);

}

result.total = list.Count();

}

result.rows = list.OrderByDescending(model => model.date).ToList();

return result;

}

### 权限管理模块的设计与实现

A）功能描述

权限管理模块只有一个核心的功能是查看和调整不同等级角色的权限，如何设定相应角色的用户具有该权限，则所有对应是该角色的用户都有权去操作该功能，如果取消相应角色的某一个权限，则对应角色的用户无权操作该功能，如果取消角色某一个模块的所有功能，则对应角色的用户不能够访问该模块。管理员也可以将该权限赋予其他角色，让其他角色也有权可以操作该模块。

B）流程设计

仅以更改用户权限为例，流程设计如图3- 11所示。



图3- 11 更改用户权限流程图

C）界面截图

仅以权限管理界面为例，界面如图3- 12所示。



图3- 12 权限管理界面

D）主要代码

仅此更改用户权限为例，代码如下。

$("#Edit").click(function () {

var updateRows = 0;

var rows = $("#dutyList").datagrid("getRows");

rows.forEach((value, index) => {

var checkState=$("td[field='havePermission']").find("input[type='checkbox']").eq(index).prop('checked');

if (checkState != value.havePermission) {

value.havePermission = checkState;

updateRows++;

}})

if (updateRows != 0) {

$.post("/SystemRight/EditUserDuty", { level: level, dutyList: rows }, function (data) {

$("#dutyList").datagrid('reload');})}

})

### 信息管理模块的设计与实现

A）功能描述

信息管理模块主要功能查看自己的收到的信息。所有的用户可以在这个模块管理自己的信息。

B）流程设计

仅以查看信息详情为例，流程设计如图3- 13所示。



图3- 13 查看信息详情流程图

C）界面截图

仅以信息管理界面为例，界面如图3- 4所示。

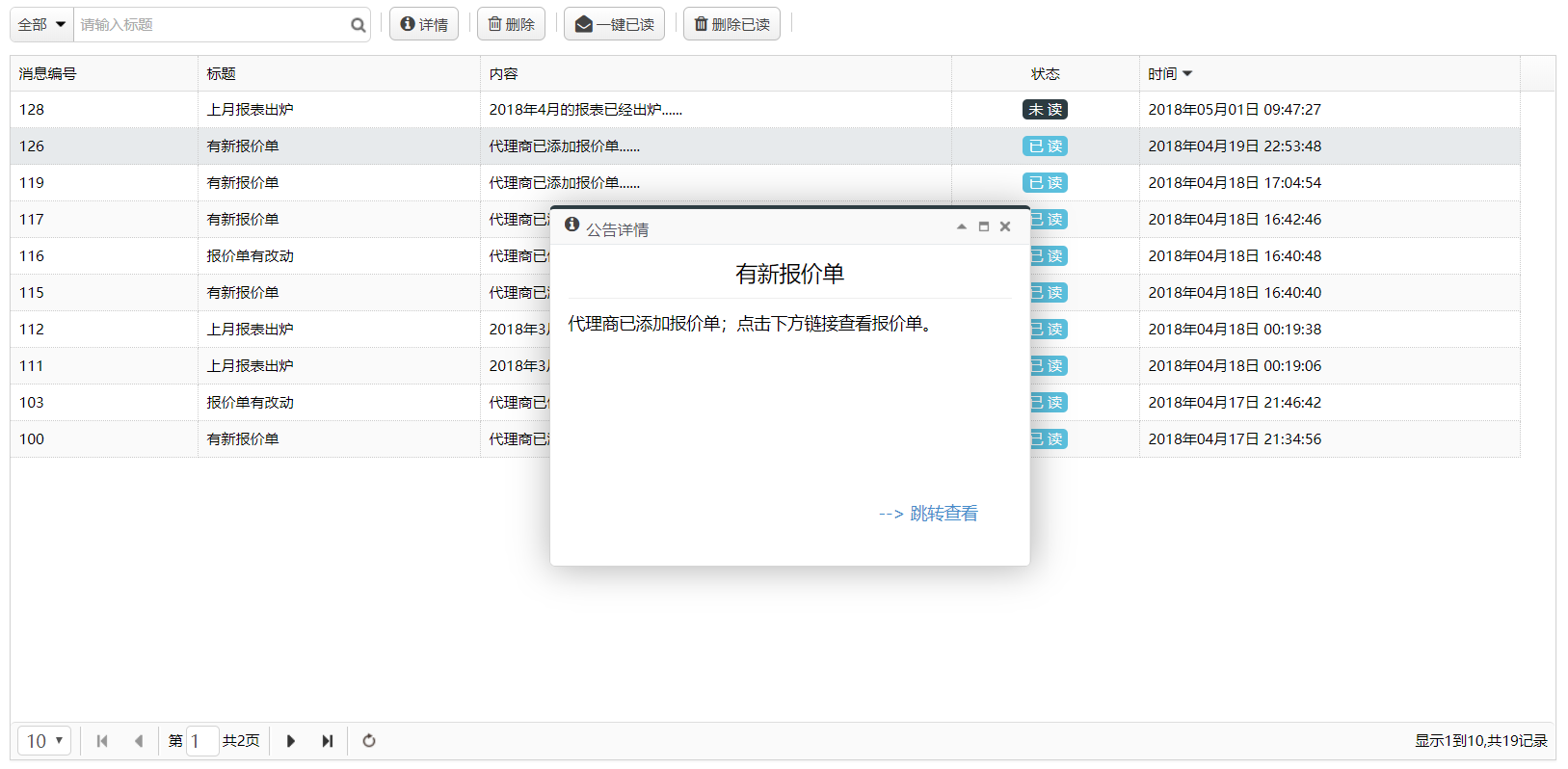


图3- 14 信息管理界面

D）主要代码

仅以查看信息为例，代码如下。

$("#Detail").click(function () {

var row = getSelectRow("#noticeList");

if (!row.state)

$.post("/Notice/ReadNoticeById", { id: row.id }, function (data) {

if (data.result) {

datagridReload("#noticeList");

window.parent.updateNotReadNotices();

}

})

openNoticeWindow("#modalwindow", 400, 300, 'fa fa-info-circle', row)

})

# 特殊问题及解决方案

## 报价单问题

### 问题描述

当用户下载已经通过审核的报价单时，报价单已经在服务器上生成了一个word文件，但是无法直接通过浏览器下载一个word文件。

### 解决方案

在点击下载报价但是的时候，把生成的word文件保存在内存中，而不是保存到服务器的硬盘里，然后通过Response对象，把word文件的内容写入Response对象，然后设置Response对象的ContentType属性，设置成为word文件，最后将Response对象的数据发回到用户操作的客户端中。

### 结果

当用户点击下载报价单按钮的时候，浏览器自动捕捉到服务器返回的数据，然后当用户确认下载后，用户就能够得到了一份对应的word格式的报价单文件了。

## 图片上传问题

### 问题描述

在用户上传头像图片的时候，传统的控件格式太过于固定，更改样式比较复杂，也不能够自由的添加事件进行操作，比如说在上传完成后给出提示结果，以及让用户可以预览上传图片的缩略图的功能

### 解决方案

使用由百度提供第三方的文件上传工具WebUploader，有效的解决了这个问题，能够在方便的更改控件的样式，也能够自由的在上传过程中添加事件来控制整个系统的上传的流程。

### 结果

上传控件很好的兼容本系统的前端风格，也能够在上传结束后添加事件，达到了上传完成了，可以预览缩略图。

## Excel文件问题

### 问题描述

当用户上传包含产品信息的Excel文件后，系统无法对Excel文件数据进行解析。

### 解决方案

使用开源的第三方库LinqToExcel，让数据模块的各个字段对应好Excel文件的各个列。在对应关系建立好了之后，逐行对数据进行读取，然后对所得到的数据进行校验和解析，筛选出合法的数据结果，最后在所有数据读取完毕后，将上传的Excel文件删除。

### 结果

系统能够顺利的解析上传的Excel中的数据，并且也能够在数据全部解析完成后，将上传的Excel文件从服务器硬盘中删除掉，保证了服务器系统资源的节约。

# 结果测试及性能分析

## 测试概要

### 测试环境

数据库服务器配置如表5- 1所示。

表5- 1 数据库服务器配置

|  |  |
| --- | --- |
| CPU | I7-4710HQ |
| 内存 | 8G |
| 硬盘（可用空间大小） | 500G |
| 操作系统 | Win10 |
| 应用软件 | Mysql |
| 机器网络名 | DESKTOP-QM61D9B |
| 局域网地址 | 127.0.0.1:3306 |

应用服务器配置如表5- 2所示。

表5- 2 应用服务器配置

|  |  |
| --- | --- |
| CPU | I7-4710HQ |
| 内存 | 8G |
| 硬盘（可用空间大小） | 500G |
| 操作系统 | Win10 |
| 应用软件 | IIS |
| 机器网络名 | DESKTOP-QM61D9B |
| 局域网地址 | 192.168.3.1:80 |

客户端配置如表5- 3所示。

表5- 3 客户端配置

|  |  |
| --- | --- |
| CPU | I7-4710HQ |
| 内存 | 8G |
| 硬盘（可用空间大小） | 500G |
| 操作系统 | Win10 |
| 应用软件 | Chrome |
| 机器网络名 | DESKTOP-QM61D9B |
| 局域网地址 | 192.168.1.1 |

### 测试范围

测试主要是针对与各个界面的主要功能进行测试，通过输入正确和错误的结果分别查看系统给出的反馈结果是否正确。

1) 登录测试：输入正确的登录账号和密码，验证是否能正确的跳转到系统的主页面；输入错误的密码，验证系统是否能够给出提示说明账号或密码错误；输入被禁用的账号，验证系统是否能提示说明该账号已被禁用。

2) 用户信息注册测试：测试创建用户时，能够得到一个新的用户ID号；上传图片时能否以新ID号命名图片；当取消注册时，系统能否自动清除之前操作上传的图片；当确认注册的时候，系统能够保留已经上传的图片信息。

3) 用户信息修改测试：测试点击修改用户信息时，系统能否读取用户原本的数据填充到编辑框中；测试系统能够通过用户原本的地址信息，填充地址栏下拉框数据。

4) 产品分类数据读取测试：在数据库中直接建立树形分类结构表，测试系统能够正确的通过深度遍历读取出正确的产品分类树形结构图。

5) 产品批量上传测试：从系统中下载一份模版，然后填入一些合法的数据，以及一些不合法的数据，然后上传后，测试系统能否正确的解析出格式正确且合法的数据。

6) 报价单下载测试：审核通过一个报价单，然后点击下载按钮，测试是否能够下载一个word格式的报价单文件；以及分别使用内部和外部人员进行下载测试，测试下载的word格式文件是否会根据用户身份而显示和隐藏利润数据项。

7) 权限测试：修改某一个角色的权限，然后使用一个对应角色的用户，查看该用户的权限是否已经被修改；关闭某一角色的某一个模块的全部权限，查看该角色对应的用户是否菜单中会取消这一模块的操作菜单的显示。

## 性能分析

综合来看，如果是页面请求时的操作，本系统大概会在两秒左右请求并返回和显示页面的结果；如果是其他少量数据请求的操作，本系统都会立刻给出响应的结果，总体的反应时间较快。除此之外，当用户进行一些错误的操作的时候，系统会进行容错的处理，并且会给出用户反馈提示操作错误，使用整个系统的容错性较强。当用户上传文件的，或者批量修改数据的时候，系统能够及时清理掉旧文件，更新新文件，保证了整个系统没有多余的无用的垃圾文件的残留，保证了系统的资源都能够很好的被利用起来，不会造成资源的浪费。

总体来说，本系统的具有一定程度的容错力能，较快的响应速度，以及很好的垃圾文件清理机制，是一个性能良好的稳定的事务处理系统。

项目总结

本报价系统是采用Asp.net框架，并且使用了MVC模式，将页面显示和后台逻辑控制分离开，保证了系统不杂糅。并且依托与Asp.net的技术支持，通过使用其系统自带的验证和授权管理，很好的提高了系统整体的安全性，保证了系统所有的系统操作必须是用户经过登录验证后，才能够有权限能够操作，避免了匿名用户的不安全操作。

本系统的后台数据库使用了开源的最受欢迎的Mysql数据库，能够有效的管理用户和系统的资料数据，并且降低了系统使用成本。后台与数据库的交互操作，使用了经由微软开发的Entity Framework 6框架，避免了使用繁琐的Sql语句来交互数据库，而是使用Linq这种类似于Sql的语言结构代替，极大了降低了开发者为实现某一些功能而需要编写大量代码的繁琐低效的工作，使得系统在数据库交互方面的代码大大的减少了，提高了系统开发的效率。

系统前端模块主要是使用了EasyUI、Bootstrap、Jquery等框架以及一些第三方的开源的库，本系统由于是后台管理系统，所有会存在大量的CRUD表格来保证系统更好的展示数据和管理数据，EasyUI在CRUD表格的应用上有其得天独厚的优势，相对于原生了Html来说，更加适合作为后台管理系统的开发框架，通过使用EasyUI，极大的降低了系统前端设计的代码量。并且搭配上Bootstrap的Css样式表，使系统整体的界面风格统一并且美观；而在Dom元素的控制方面相比原生的Js关于Dom的支持，Jquery不仅更加简洁高效，而且很多了处理了浏览器的兼容性问题。能够对开发者屏蔽掉底层的浏览器兼容性适配问题，让开发者能够更加专注于功能开发，除此之外，像是EasyUI，Bootstrap之类的框架都是就架构在Jquery之上的框架，所以不管从哪一方面来说，Jquery都是项目前端的核心，能够简化工作，加快效率。

而在Word文件的导出和Excel文件的导入方面，都避免使用了Office的Com库，而是用开源的第三方库来实现功能，这样做的好处有很多，如果使用Office自带的库，那么系统所在服务器必须安全有Office才能进行操作，否则无法使用库，再者，Office自带的库对于不同的Office版本不能很好的兼容，常常会出现兼容性问题，而是用第三方库，既可以无法让系统安装Office，也解决了Office版本兼容的问题。

整个系统不管是数据库还是一些第三方的功能库，都保持了开源免费的原则，不仅迎合了现在开源的大趋势，另一方面免费开发的工具和库极大的降低整个系统整体的开发成本；也使得整个系统更加的简洁高效。

结 束 语

本系统顺应现在“新零售”的趋势和信息化办公的潮流。将系统的使用用户定位到日用百货的代理商和供应商之间。在Asp.net的框架下开发设计而现实的。系统前端界面美观友好，而且布局合理，将系统需要操作的功能明显且集中的提供给用户，使得用户能够轻易找到。让用户从传统的办公模式中解放出来，进入到信息化办公的时代。降低了工作的繁琐性，提高了工作的效率。并且新系统学习成本极低，使得用户转化工作模式无需花费大量的学习时间，能够在极短的时间内轻易上手操作。

本系统主要的功能包括了内部人员可以上传本公司的售卖的商品信息，以及管理商品的各个信息，并且调整产品的各个等级售价等等；而外部人员则可以通过本系统查看供应商的各个产品的情况和针对自己的售价，也能够根据自身需要生成或者是修改报价单提交给内部人员审核，也可以删除作废的报价单；而内部人员可以查看外部人员提交的报价单，也可以审核报价单是否被批准；除此之外，不管是外部人员还是内部人员都可以下载已经审核通过的报价单作为依据。在每一个月的开始，系统会自动将上一个月的销售情况汇总，形成统计表呈现给用户，而不需要用户自己手动统计。系统也提供了自主升级降级的操作，当代理商的每月的销售情况出炉后，会自动根据每月销售情况对代理商进行等级处理，而不再需要供应商来一个一个排查处理，降低了人员的工作量。同时系统还有权限系统，能够很好的对不同等级的用户开放或者是关闭某一些系统功能，让系统更具灵活易用。

本系统在业务处理方面还有一些不足。因为对代理商和供应商之间的询价报价操作流程没有细致的了解过，所以对一些流程的细节把控的不是很到位，所以相比传统的报价管理，本系统在业务处理的流程上会有一些细节上的差异和不足之处。

除此之外，本系统由于使用了大量的第三方的库，相比于原生的语言写出来的项目，在执行的速度上会慢一些，虽然在系统上做了一定的优化工作，在一定程度上加快了速度，但是仍然无法和原生的系统的运行速度相比，所消耗的资源相比于原生系统也会更多一些。

通过本系统，供应商可以科学地，系统地并且高效地管理公司的产品和报价单，代理商可以方便地进行询价操作。加之系统提供了很好的统计功能，极大的降低时间成本，充分的提高了工作的效率，使得公司进入了信息化办公的新时代，顺应了时代的发展趋势。

参考文献

1. 陈静,李红灵. 基于B/S模式的管理信息系统开发研究[J]. 云南民族学院学报(自然科学版), 2010.
2. 刘甫迎,刘光会,王蓉.C#程序设计教程（第2版）[M].北京:电子工业出版社,2013.
3. 王慧芳, 毕建权. 软件工程[M]. 浙江: 浙江大学出版社, 2005.
4. 吉根林, 崔海源. Web程序设计(第二版)[M]. 北京: 电子工业出版社, 2006.
5. 王珊, 萨师煊. 数据库系统概论[M]. 浙江: 浙江大学出版社, 2005.
6. W3Cschool ASP.NET教程. <https://www.w3cschool.cn/aspnet/>
7. 胡新强. 基于ASP.NET的物流管理系统的设计及车辆调度优化[D].东华大学,2017.
8. 朱帮明. 基于MVC与ASP.NET的学生信息管理系统的设计与实现[D].吉林大学,2015.
9. 吴清芳. 基于ASP.NET的网页设计精品课程网络教学系统的设计与实现[D].中国海洋大学,2011.
10. 宣帅. .NET平台下MVC框架应用研究[D].南京航空航天大学,2010.
11. 董贺. 基于Entity Framework的网络劳资考勤管理系统设计与实现[D].河北师范大学,2016.
12. 刘坤. 基于LINQ的数据查询服务设计及应用研究[D].中国海洋大学,2015.
13. 曾原. 基于EasyUI的高职院校科研管理系统的设计与开发[D].吉林大学,2014.
14. 熊惠敏. 基于SSH+EasyUI的信用卡资金监管系统的设计与实现[D].厦门大学,2014.
15. 毛平. Bootstrap方法及其应用[D].湘潭大学,2013.
16. 黄競. 基于jQuery框架的Web前端系统构建方法的研究与应用[D].北京邮电大学,2013.

致　　谢

在论文完成过程中，本人还得到了廖德钦老师和徐三多同学的热心帮助，在我陷入困难中，都是因为有老师的提醒，和同学们的帮助，我才能够解决一个又一个的问题，及时完成了项目的功能！

最后向在百忙之中评审本文的各位专家、老师表示衷心的感谢！

**作者简介：**

|  |  |  |  |
| --- | --- | --- | --- |
| 姓 名： | 徐雯俊 | 性别： | **男** |
| 出生年月： | 1996年3月29日 | 民族： | **汉** |
| E-mail： | 906626481@qq.com |  |  |

声　　明

本论文的工作是2017年10月至2018年5月在成都信息工程大学计算机学院完成的。文中除了特别加以标注地方外，不包含他人已经发表或撰写过的研究成果，也不包含为获得成都信息工程大学或其他教学机构的学位或证书而使用过的材料。

关于学位论文使用权和研究成果知识产权的说明：

本人完全了解成都信息工程大学有关保管使用学位论文的规定，其中包括：

（1）学校有权保管并向有关部门递交学位论文的原件与复印件。

（2）学校可以采用影印、缩印或其他复制方式保存学位论文。

（3）学校可以学术交流为目的复制、赠送和交换学位论文。

（4）学校可允许学位论文被查阅或借阅。

（5）学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

除非另有科研合同和其他法律文书的制约，本论文的科研成果属于成都信息工程大学。

特此声明！

作者签名：徐雯俊

2018 年 5 月 9 日