**华东交通大学**

**毕业设计（论文）**

**题目： 基于DUBBO的“珍宝馆珠宝商城”的设计和实现**

**Design and Implementation of "Gems Mall"**

**Based on DUBBO**

|  |  |  |  |
| --- | --- | --- | --- |
| **学 院:** | **软件学院** | | |
| **专 业:** | **软件工程** | **班 级:** | **杰普一班-2014** |
| **学生姓名:** | **胡劲松** | **学 号:** | **20142110010529** |
| **指导教师:** | **叶云青** | **完成日期:** | **2018.05.06** |

|  |
| --- |
| **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  **2016 年 5 月 23 日** |

**华东交通大学毕业设计（论文）任务书**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 胡劲松 | | | 学号 | | 20142110010529 | | | | 毕业届别 | 2014 | | 专业 | | 软件工程 |
| 毕业设计（论文）题目 | | | | | 基于DUBBO的“珍宝馆珠宝商城”的设计和实现 | | | | | | | | | | |
| 指导教师 | | 叶云青 | | | | | 学历 | | 研究生 | | | 职称 | | 讲师 | |
| **具体要求：**  1、设计目的与内容  选题意义：现在人们的生活质量渐渐提高，对日常首饰、玉石、水晶等珠宝的需求逐渐增高，为满足人们需求，开发关于珠宝的商城。  设计目的：通过需求分析、详细设计及功能实现以及数据库设计，设计出一个珠宝商城。  设计内容：系统主要实现用户注册、修改用户信息、首页各类商品的推荐展示、商品分类、商品按条件筛选、商品详情页、购物车页面、付款流程以及订单查询，并可编辑个人信息等。  2、设计要求与成果  技术要求：使用Spring、SpringMVC、Mybatis作为基本框架，使用Redis进行持久化缓存，使用Dubbo中间件，安全框架使用Shiro进行实现，前端使用Vue.js完成动态展示效果。  设计步骤：按照软件开发包括需求、设计、编码和测试等阶段进行设计；  英文翻译：要求两千个实词以上的英文相关技术文献翻译为中文；  设计成果：毕业论文一篇+程序源代码。  3、应收集的主要参考文献  [1] 郝玉龙.JavaEE编程技术[M].清华大学出版社,2008.  [2] 李智慧.大型网站技术架构[M].北京:电子工业出版社,2013.  [3] 李磊,李娟.Dubbo服务框架技术在学习系统开发中的应用与实践[J].计算机系统应用,2017,计算机系统应用.  **进度安排：**   |  |  |  |  | | --- | --- | --- | --- | |  | 设计各阶段内容 | 时间 | % | | 1 | 按题目要求进行资料收集、调研，阅读相关技术资料和文献，熟悉开发设计工具。 | 第7学期18～20周 | 5 | | 2 | 确定系统的结构、功能、模块和数据，制定具体的工作计划和方案，完成需求分析、开题报告。 | 第8学期1～2周 | 10 | | 3 | 按任务书要求进行系统的概要设计、详细设计和数据库设计。 | 第8学期3～5周 | 25 | | 4 | 撰写中期检查报告。根据系统设计的成果，完成代码调试，并进行系统测试。 | 第8学期6～9周 | 40 | | 5 | 修改和批改后完成终稿。对正文部分进行查重。全部审核合格后，参加答辩。 | 第8学期10～15周 | 20 | | 6 | 英文资料翻译 | 第8学期1～2周 |  |   指导教师签字：  2015年12月27日 | | | | | | | | | | | | | | | |
| 题目发出日期 | | | 2015.12.28 | | | | | 设计（论文）起止时间 | | | 2016.12.29-2016.6.3 | | | | |
| 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DUBBO的“珍宝馆珠宝商城”的设计和实现 | | | | |
| 课题来源 | 自主选题 | 课题类型 | CY | 导师 | 叶云青 |
| 学生姓名 | 胡劲松 | 学号 | 20142110010529 | 专业 | 软件工程 |
| **一、开题报告内容**  1、目的和意义  随着互联网技术的快速的发展，人们不只通过网络去接收资讯，而且还通过网络去满足生活中的需要，网上购物早已普及大众，随着生活水平的提高，人们步入小康水平，跨入中产阶级，这时在网上购物不单单是满足生活的需求，还向往着更高的生活品质，这时珠宝正好满足这一需求，再与互联网结合，应运而生的珠宝商城，给人们带来便利，同时降低了人力物力等。   1. 研究现状和发展趋势   随着生活水平的提高，人们对生活的需求不只是柴米油盐酱醋，而是要活得精致，珠宝便是一种使人精致的物品，珠宝是可以增加女性气质的最佳产品。 时间让你变老，但珠宝可以永远让你美丽而性感。 珠宝往往是人们感情和情绪的寄托。 无论价值如何，具有特殊意义的纪念品在人们心中都是不可替代的。在这种市场下，网上珠宝商场方兴未艾，开发此商城是非常有必要的。   1. 设计方案   本系统后台是基于MVC三层模式架构，使用Spring、SpringMVC、Mybatis框架，后台使用maven管理，设计成一个聚合工程，使用Dubbo中作为服务中间件，从而来发布服务和通知调用服务。将项目分成两层：服务层和表示层。 服务层包含业务逻辑，它只需要提供外部服务。 表示层只需要处理与页面的交互，通过调用服务层的服务来实现业务逻辑。使用Redis作为缓存，数据库使用mysql。  4、进度安排   |  |  |  |  | | --- | --- | --- | --- | |  | 设计各阶段内容 | 时间 | % | | 1 | 按题目要求进行资料收集、调研，阅读相关技术资料和文献，熟悉开发设计工具。 | 第7学期18～20周 | 5 | | 2 | 确定系统的结构、功能、模块和数据，制定具体的工作计划和方案，完成需求分析、开题报告。 | 第8学期1～2周 | 10 | | 3 | 按任务书要求进行系统的概要设计、详细设计和数据库设计。 | 第8学期3～5周 | 25 | | 4 | 撰写中期检查报告。根据系统设计的成果，完成代码调试，并进行系统测试。 | 第8学期6～9周 | 40 | | 5 | 修改和批改后完成终稿。对正文部分进行查重。全部审核合格后，参加答辩。 | 第8学期10～15周 | 20 | | 6 | 英文资料翻译 | 第8学期1～2周 |  | | | | | | |

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）开题报告书（续）**

|  |
| --- |
| 5、主要参考文献  [1] 郝玉龙.JavaEE编程技术[M].清华大学出版社,2008.  [2] 李智慧.大型网站技术架构[M].北京:电子工业出版社,2013.  [3] 李磊,李娟.Dubbo服务框架技术在学习系统开发中的应用与实践[J].计算机系统应用,2017,计算机系统应用.  [4] 毕建信.基于MVC设计模式WEB应用研究与实现[D].武汉:武汉理工大学,2006.  [5] 李国娟.基于JavaEE架构的网上购物系统的设计与实现[D].河北:河北工业大   1. **方法及预期目的** 2. 研究方法（手段）   这个系统主要基于J2EE开发，使用Idea作为开发工具和MYSQL作为数据库。  后台系统采用JAVA语言开发，前台使用Vue进行开发，前后台分离，后台只需提供前台需要的接口，这种设计可以更好的开发系统以及维护系统，同时系统界面简单易操作。  使用Redis对经常需要查询的大数据进行缓存。   1. 预期目的   (1) 界面简洁，操作简单；  (2) 实现商品分类展示，以及条件筛选商品；  (3) 实现购物车功能，以及购买一系列操作；  (4) 实现查询订单功能，以及编辑个人信息等  。  **三、指导老师意见**  同意开题  指导教师签名： 日期：2016.3.14 |

**华东交通大学毕业设计(论文)评阅书(1)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **姓名** | 胡劲松 | **学号** | 20142110010529 | **专业** | 软件工程 |
| **毕业设计(论文)题目** | | 基于DUBBO的“珍宝馆珠宝商城”的设计和实现 | | | |
| **指导教师评语：**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 具 体 要 求 | 优 | 良 | 中 | 一般 | 差 | | 出勤及工作态度（20%） |  |  |  |  |  | | 方法合理，设计工作量饱满（20%） |  |  |  |  |  | | 论点正确，论文内容有一定难度（20%） |  |  |  |  |  | | 结构严谨，论文有一定应用价值（20%） |  |  |  |  |  | | 对前人工作有改进或有独特见解。（10%） |  |  |  |  |  | | 论文格式正确，撰写规范（10%） |  |  |  |  |  |  |  |  | | --- | --- | | **得分** |  |   **指导教师签字：**  **2016年 5 月 日**  **评阅人评语：**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 具 体 要 求 | 优 | 良 | 中 | 一般 | 差 | | 选题合理，设计有应用价值（20%） |  |  |  |  |  | | 方法合理，设计工作量饱满（20%） |  |  |  |  |  | | 论点正确，论文内容有一定难度（20%） |  |  |  |  |  | | 对前人工作有改进或有独特见解。（20%） |  |  |  |  |  | | 论文格式正确，撰写规范（20%） |  |  |  |  |  |  |  |  | | --- | --- | | **得分** |  |   **评阅人签字：**  **2016 年5 月 日** | | | | | |

**华东交通大学毕业设计(论文)评阅书(2)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **姓名** | 胡劲松 | **学号** | 20142110010529 | **专业** | 软件工程 |
| **毕业设计(论文)题目** | | 基于DUBBO的“珍宝馆珠宝商城”的设计和实现 | | | |
| **答辩小组评语：**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 具 体 要 求 | 优 | 良 | 中 | 一般 | 差 | | 符合要求 |  |  |  |  |  | | 答辩准备充分，论文题目与内容相符 |  |  |  |  |  | | 语言精练能突出重点，思路清晰能准确表达 |  |  |  |  |  | | 论点正确，论文内容有一定难度 |  |  |  |  |  | | 方法合理，论文内容工作量饱满 |  |  |  |  |  | | 结构严谨，论文有一定应用价值 |  |  |  |  |  | | 对前人工作有改进或有独特见解 |  |  |  |  |  | | 正面回答问题 |  |  |  |  |  | | 回答问题有理论依据，基本概念清楚 |  |  |  |  |  | | 主要问题回答准确，深入 |  |  |  |  |  |  |  |  | | --- | --- | | **得分** |  |   **组长签字：**  **2016 年5 月 日** | | | | | |
| **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**   |  |  | | --- | --- | | **等级** |  |   **进入二次答辩 □ 评优组 最终成绩由二次答辩决定**  **□ 合格组**  **答辩委员会主任签字：**  **2016 年 6 月 日（学院公章）** | | | | | |

注：答辩小组根据评阅人的评阅签署意见、初步评定成绩，交答辩委员会审定，盖学院公章。

“等级”用优、良、中、及、不及五级制（可按学院制定的毕业设计(论文)成绩评定办法评定最后成绩）。

**华东交通大学毕业设计（论文）答辩记录**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 胡劲松 | 学号 | 20142110010529 | 毕业届别 | 2018 | 专业 | | 软件工程 |
| 题目 | 基于DUBBO的“珍宝馆珠宝商城”的设计和实现 | | | | 答辩时间 | | 2018年 5 月23 日 | |
| 答辩组成员（签字）： | | | | | | | | |
| 答辩记录：  记录人（签字）：    答辩小组组长（签字）： | | | | | | | | |

基于DUBBO的“珍宝馆珠宝商城”的设计和实现

摘 要

随着互联网技术的快速的发展，人们不只通过网络去接收资讯，而且还通过网络去满足生活中的需要，网上购物早已普及大众，随着生活水平的提高，装饰品必不可少，而珠宝是一使人精致的物品，珠宝是可以增加女性气质的最佳产品。 时间让你变老，但珠宝可以永远让你美丽而性感。 珠宝往往是人们感情和情绪的寄托。 无论价值如何，具有特殊意义的纪念品在人们心中都是不可替代的。

本文首先对珠宝商城的发展现状进行研究，明确了项目的研究思路和方向。然后，按照实际项目的要求，进行详细分析月设计，后台是基于SOA的架构，也就是把工程拆分成服务层、表现层两个工程。使用Spring、SpringMVC、Mybatis框架，后台使用maven管理，使用Dubbo中作为服务中间件，设计成一个聚合工程，从而来发布服务和通知调用服务。服务层包含业务逻辑， 它只需要提供外部服务。 表示层只需要处理与页面的交互，通过调用服务层的服务来实现业务逻辑。使用Redis作为查询大数据的缓存，使用Mysql作为数据库。

关键词：SOA、Redis、Spring

**Design and Implementation of "Gems Mall"**

**Based on DUBBO**

**Abstract**

With the rapid development of Internet technology, people not only receive information through the Internet, but also through the Internet to meet the needs of life, online shopping has long been popular, with the improvement of living standards, decorations are essential, and jewelry is An exquisite item, jewelry is the best product that can increase femininity. Time makes you old, but jewelry can always make you beautiful and sexy. Jewelry is often the sustenance of people's feelings and emotions. Regardless of value, souvenirs with special significance are irreplaceable in people's hearts.

This article first studied the development status of the jewelry mall, and clarified the research ideas and direction of the project. Then, in accordance with the requirements of the actual project, a detailed analysis of the monthly design, the background is based on the SOA architecture, which is to split the project into the service layer, the presentation layer of the two projects. Use Spring, SpringMVC, Mybatis framework, use maven management in the background, use Dubbo as service middleware, and design an aggregation project to publish services and notification calling services. The service layer contains business logic. It only needs to provide external services. The presentation layer only needs to handle the interaction with the page and implement the business logic by calling the services of the service layer. Use Redis as a cache for querying big data and Mysql as a database.

Keywords: SOA、Redis、Spring

**目 录**

[**1 绪 论 1**](#_Toc4107)

[1.1 研究的背景及意义 1](#_Toc23381)

[1.1.1 选题的背景 1](#_Toc28985)

[1.1.2 国内外研究现状 1](#_Toc22578)

[1.1.3 研究的意义 2](#_Toc14725)

[1.2 系统目标 2](#_Toc25158)

[**2 需求分析 3**](#_Toc5001)

[2.1 功能需求 3](#_Toc2077)

[2.1.1 子模块说明 3](#_Toc12408)

[2.1.2 功能需求描述 3](#_Toc1940)

[2.2 非功能需求 3](#_Toc10853)

[2.2.1 性能需求 3](#_Toc16244)

[2.2. 2其它需求 4](#_Toc26098)

[**3 总体设计 5**](#_Toc26344)

[3.1 运行环境 5](#_Toc22118)

[3.2 基本处理流程 5](#_Toc445)

[3.2.1总功能模块介绍 7](#_Toc24436)

[**4 数据库设计 9**](#_Toc32652)

[4.1系统和数据库的配置 9](#_Toc10812)

[4.2概念结构设计 9](#_Toc9529)

[4.3逻辑结构设计 12](#_Toc12105)

[4.3.1 设计思路 12](#_Toc7026)

[4.3.2 逻辑模型 12](#_Toc1439)

[4.4物理结构设计 14](#_Toc4620)

[4.4.1 存取方式 15](#_Toc18197)

[4.4.2 存储结构 15](#_Toc7304)

[**5 界面设计 16**](#_Toc31903)

[5.1 界面关系图 16](#_Toc22705)

[5.2 界面设计成果 17](#_Toc24810)

[5.2.1 主界面 17](#_Toc32418)

[5.2.2 子界面 17](#_Toc8003)

[**6 详细设计 22**](#_Toc14646)

[6.1 系统主要功能模块介绍 22](#_Toc31816)

[6.2 用户登录模块设计 22](#_Toc13051)

[6.2.1 用户登录算法描述 22](#_Toc14103)

[6.2.2 登录程序流程图 22](#_Toc31320)

[6.2.3 登录模块关键类说明 23](#_Toc10570)

[6.3 学生信息管理模块设计 23](#_Toc15360)

[6.3.1 学生信息管理算法描述 23](#_Toc18475)

[6.3.2 学生信息管理程序流程图 24](#_Toc11766)

[6.3.3 学生信息管理关键类说明 24](#_Toc6178)

[6.4 课程信息管理模块设计 24](#_Toc6056)

[6.4.1 课程信息管理算法描述 24](#_Toc27877)

[6.4.2 课程信息管理程序流程图 24](#_Toc2030)

[6.4.3 课程信息管理关键类说明 25](#_Toc25601)

[6.5 选课管理模块设计 25](#_Toc4079)

[6.5.1 选课管理算法描述 25](#_Toc13476)

[6.5.2 选课管理程序流程图 25](#_Toc21046)

[6.5.3 选课管理关键类说明 26](#_Toc1756)

[6.6 任课教师信息管理模块设计 26](#_Toc28550)

[6.6.1 任课教师信息管理算法描述 26](#_Toc21924)

[6.6.2 任课教师教师信息管理程序流程图 26](#_Toc26409)

[6.6.3 任课教师信息管理关键类说明 27](#_Toc14960)

[**7 编码 28**](#_Toc28361)

[7.1 代码实现与核心算法 28](#_Toc13858)

[7.2 代码优化分析 28](#_Toc9758)

[**8 测试 30**](#_Toc27268)

[8.1 测试方案设计 30](#_Toc27889)

[8.1.1 测试策略 30](#_Toc4288)

[8.1.2 测试进度安排 30](#_Toc5695)

[8.1.3 测试资源 30](#_Toc7278)

[8.1.4 关键测试点 31](#_Toc18827)

[8.2 测试用例构建 31](#_Toc14378)

[8.2.1 测试用例编写约定 31](#_Toc11161)

[8.2.2 测试用例设计 31](#_Toc28632)

[8.2.3 关键测试用例 31](#_Toc29225)

[8.2.4 测试用例维护 33](#_Toc15866)

[**9 总结与展望 34**](#_Toc25199)

[9.1 设计工作总结 34](#_Toc7364)

[9.2 未来工作展望 34](#_Toc30807)

[**谢 辞 35**](#_Toc217)

[**参考文献 36**](#_Toc4814)

[**附录A 外文翻译—原文部分 37**](#_Toc26087)

[**附录B 外文翻译—译文部分 43**](#_Toc11897)

[**附录C 软件使用说明书 49**](#_Toc7601)

[**附录D 主要源代码 50**](#_Toc13554)

1 绪 论

## 1.1 研究的背景及意义

### 1.1.1 选题的背景

近年来，为了顺应“互联网+”时代的发展，各行业开始从事电子商务寻求突破。珠宝电子商务也迅速增长。国内许多珠宝品牌在电子商务领域取得了令人瞩目的成绩。目前，电子商务正促使珠宝公司随着互联网的流行而改变经营方式和销售方式。珠宝行业采用电子商务促进可持续发展。这不是一种选择，已成为必需品。

早期珠宝电子商务的发展主要是创建网上购物中心和使用第三方平台。随着电子商务的发展，这些模式已成为过去。 “互联网+珠宝”不是简单地把珠宝放在线上。为了在珠宝市场占有一席之地，珠宝电子商务可能会创造出新的模式。目前，国内珠宝电子商务已进入合理发展时期，珠宝电子商务市场也呈现出新的气氛。在这种市场中，网上珠宝店处于上风。开发这个商场非常有必要。

### 1.1.2 国内外研究现状

近年来，电子商务作为一种新型的商业从而流通大力发展，中国的电子商务发展不再局限于中国。已开始拓展海外市场，跨境电商仍保持快速增长态势。

作为一种新型的贸易方式，跨境电子商务依托互联网和国际物流直接连接终端，满足客户需求。它具有门槛低，环节少，成本低，周期短等优点，并在全球范围内蓬勃发展。 。

近年来，中国跨境电子商务进口快速增长。 “海淘”和“海代”（海外代购）已成为流行的购物方式，将中国消费者和国外品牌紧密联系在一起。中国电子商务的发展已初具规模，促进了跨境电子商务出口的快速发展。

目前，跨境电子商务的发展也面临一些瓶颈。在新闻报道中，云南最具代表性的翡翠电子商务公司玉皇朝在美国，德国，加拿大，新西兰，新加坡，泰国和香港很受欢迎。澳门，台湾等国家和地区的市场销售也呈现良好态势。然而，缺乏安全顺畅的物流保证一直限制了公司拓展海外市场的瓶颈。

随着市场的扩大，海外订单不再是问题，跨境物流体系的不成熟是阻碍跨境电商发展的关键。物流公司网络覆盖相对分散，跨境能力不足，交货时间长，运营成本高，用户体验差，容易产生产品损坏或丢包。这一直是制约云南省企业跨境电子商务发展问题的症结所在之一。

### 1.1.3 研究的意义

珠宝商城的开发，能够很好的解决单品牌、款式少、风格单调等问题，我们将会将与各大品牌商合作，实现品牌多元化，并且推出海外市场，引进海外的商品，降低用户海购风险以及减少用户购物成本节约减少海外物流时间，让用户能够在一个平台上买到心仪的商品。加强政府，企业和行业组织的合作，督促企业进一步履行主要职责。 解决电子商务平台企业缺乏严密准入，产品信息审查不力，销售行为管理混乱，信用评估缺陷，内部人员疏漏控制等问题，以确保用户的财产安全。

## 1.2 系统目标

本项目需要为珠宝商城实现简洁干净的前台页面，做到商品能够分门别类，首页有轮播图展示商城活动或者热门商品，同时按照不同的分类在不同的区域展示。该系统提供登陆注册功能，用户可以在未登录的情况下进行浏览商品以及查看商品详细信息，如果需要查看购物车或者查看用户信息资料需要进行登录，未注册的用户可以使用快速注册进行注册，用户在登录后可以去进行加入购物车和购买商品一系列操作，并可以修改个人信息上传个人头像以及查看个人订单等，实现珠宝商城用户购买的一系列功能。

2 需求分析

## 2.1 功能需求

### 2.1.1 子模块说明

本系统是一个聚合工程，使用Maven管理工程，其基本结构如下：

common-parent：父工程，打包方式pom，管理jar包的版本号。

| 项目中所有工程都应该继承父工程。

|-- common -util：通用的工具类。打包方式jar

|-- common -manager：服务层工程。聚合工程：Pom工程

|-- common -manager-dao：与Mybatis直接交互的持久层，

包含Mapper接口和Mapper映射文件，打包方式jar

|-- common -manager-pojo：主要是对应数据库的Bean，已经一些包装类，打包方式jar

|-- common -manager-interface：主要是Service的接口，打包方式jar

|-- common -manager-service：主要是逻辑层的实现，为表现层提供接口，打包方式jar

|-- common -manager-web：表现层工程，与view进行交互。打包方式war

本系统分为四个模块：用户管理模块，商品管理模块，购物车管理模块，订单管理模块。

### 2.1.2 功能需求描述

1. 用户管理模块：用户通过注册成为商城的会员，才能购买商品。在用户登录界面，用户可以使用注册的账号进行登录，登陆后可以自行购买商品，本商城支持支付宝、微信、QQ钱包付款。用户若忘记了密码，可以在登录界面点击忘记密码，进行重置密码。用户可以对个人信息进行管理，例如上传自己的头像。
2. 商品管理模块：该模块主要针对商品的查询，以及将查询后的数据进行前台的渲染，包括首页各个分区商品的展示，以及全部商品的展示，并且可以按条件进行筛选商品。
3. 购物车管理模块：用户对心仪的商品可以直接放在购物车里，在购物车里可以对商品进行编辑，可选择性的进行支付操作。
4. 订单管理：该模块主要是生成用户提交的订单，以及支付流程的记录，并且还可以在订单页面对历史订单进行管理。

## 2.2 非功能需求

### 2.2.1 性能需求

1. 系统操作界面

① 购物界面简洁明了，在相关输入框设置相应的提示，方便用户操作，对产生的错误进行过滤，以说明性文字提醒用户，而不是弹出错误代码。

② 保密性，对用户的个人信息、购物信息、账单信息进行严格保密，并且加以一定的加密措施。

③ 安全性，保证用户在支付时的安全性，以及保证用户资金的正确流转。

1. 数据查询性能

① 查全率，保证所有商品的信息完整，通过过滤来保证商品的名称、价格、图片、详情等信息的完善。

② 查准率，用户可以通过一些过滤条件来筛选商品，例如价格区间、价格升降序等，保证在所有条件下返回合适的商品。

1. 适应性

系统设计应尽可能地满足不同操作环境的要求，确保系统在不同性能的计算机上稳定运行，并确保在不同操作系统上的稳定运行。

### 2.2. 2其它需求

（1）数据的安全性

要求输入正确的用户名称及密码后方可进入系统，不同的用户的数据查询权限也不同。

（2）系统的可靠性

由于系统设置的用户较多，为便于系统维护及管理，应设置系统管理员，对系统信息进行管理并维护，确保系统的可靠运行。

（3）系统的可扩充性

编写程序的时候严格服从一定的标准，不同性质的代码分开，获取信息都是从数据库里面获取，使得信息功能都可以通过加入相应的代码进行扩充。

（4）系统的易用性

由于本系统为网上选课系统，核心操作为选课。因此应将界面设计人性化，方便用户进行操作。

3 总体设计

## 3.1 运行环境

高校网上选课系统的运行环境主要包括软件环境和硬件环境。其中硬件环境包括计算机及与计算机相关的外围设备组成的计算机物理系统，即硬件设施。软件环境是运行安装于计算机硬件之上的用来驱动计算机及其外围设备来实现系统运行的软件系统。

1. 硬件环境

本系统应用平台主要为开设了自主选课的高校，因此系统的访问量在选课时期会较大，考虑到这个原因系统的硬件硬件环境配置较为严格。服务器要求CPU：4GHZ以上，内存容量在8G以上，更具系统数据库的存储需求及预测未来用户的增长，要求硬盘容量在1T以上。

1. 软件环境

本系统所用的后台操作系统是现在最新的Windows10操作系统，将MyEclipse6.5作为系统开发工具，并采用MySQL开源数据库作为系统的数据库对系统数据进行存储管理，采用Tomcat 服务器作为系统的Web服务器。

## 3.2 基本处理流程

高校网上选课系统的管理员权限的基本构架如图3-1所示。

系统管理员登录

教务管理员权限授予

管理教务管理员

数据库

图3-1 系统管理员权限架构图

高校网上选课系统的教务管理员权限的基本架构如图3-2所示。

教务管理员登录

学生管理

任课教师管理

课程管理

选课记录查询

学生成绩查询

数据库

图3-2教务管理员权限架构图

高校网上选课系统学生权限的基本流程如图3-3所示。

学生登录系统

选课成绩查询

个人信息管理

课程信息查询

个人选课记录查询

数据库

图3-3 学生权限功能基本架构图

高校网上选课系统的任课教师权限基本流程如图3-4所示。

任课教师登录系统

选课成绩管理

个人信息修改

数据库

课程记录查询

课程信息查询

图3-4 任课教师权限功能基本架构图

### 3.2.1总功能模块介绍

系统的功能模块如图3-5所示，主要分为八个模块。

系统

教务管理员登录

教师登录

学生登录

学生管理

任课教师管理

课程管理

选课记录查询

成绩记录查询

选课成绩管理

个人信息管理

选课记录查询

选课管理

成绩记录查询

个人信息管理

系统管理员登录

教务管理员管理

图3-5模块结构图

（1）用户登录模块，在登录界面中用户可通过此模块在界面中输入与其对应的用户名称、密码及选择相对应的身份登录系统并获得不同的用户权限，系统用户主要包括：任课教师、教务管理员、学生和系统管理员。

（2）密码修改模块，用户在成功登录系统后，可在相关界面中进入密码修改模块进行密码修改，每一位用户的初始密码均为111111。

（3）学生信息管理，该模块主要为教务管理员登录系统后通过此模块将学生信息输入系统并存入数据库，学生信息具体包括：姓名、备注、班级、身份证、联系电话、住址和学号。并且可以通过此模块对学生信息进行查询及修改。

（4）课程信息管理模块，该模块主要在教务管理员登录系统后通过此模块将课程信息进行输入系统并存入数据库，其中课程信息具体包括：授课老师、上课地点、学分、课程名称、课程简介、上课人数和上课时间，并可以对录入的课程信息进行查询、修改及删除。

（5）选课记录查询，不同的用户可以通过此模块选课记录进行查询。

（6）任课老师模块，教务管路员可通过此模块对任课老师信息进行录入，其中任课教师信息包括：工号（教师登录时的用户名）、姓名、性别身份证、地址、联系电话和备注，并且可以对录入的信息进行查询和修改。

（7）成绩查询模块，任课老师通过此系统对学生学习进行评分，学生及管理员可通过此模块对成绩进行查询。

（8）教务系统管理模块，此模块由系统管理员主要负责，其功能是对教务管理员进行管理，其中包括添加、修改、删除和查询教务管理员。

（9）选课管理模块，此模块主要为学生在登陆后通过查询课程名称来进行选课操作。

# 4 数据库设计

## 4.1系统和数据库的配置

系统按通用的B/S（浏览器、服务器）模式进行设计，如图4-1所示。

浏览器

Web服务器

数据库

图4-1 B/S结构图

数据库结构设计是高校网上选课系统设计中一个非常重要的环节，在高校选课的高峰期会对系统数据存储产生很大的压力，因此设计一个优秀合理的数据库可以提高系统的运行效率。合理的数据库结构设计不仅可以保证数据的完整性、一致性、提高数据存取的效率，还更加有利于程序的实现。

## 4.2概念结构设计

根据对高校网上选课系统的需求分析得到的用户需求转化为对高校网上选课系统数据库的概念设计，是数据库设计的关键步骤。本系统的数据库采用了自底向上的方法对数据库进行数据抽象和结构设计。E-R图是概念模型设计使用的典型方法之一，即用实体-联系模型表示，其中矩形代表实体型，并在其中输入实体名；椭圆代表实体的属性，并以没有箭头无向边将其与相应的实体型连接起来；菱形代表实体型间的联系，在菱形框内写明联系名，并与有关实体型间以没有箭头的无向边分别连接起来，同时在无向边注明联系类型（1:1，1:n或m:n）。对高校网上选课系统进行综合分析后，E-R图如图4-2所示。

系统管理员

管理

任课教师

教务管理员

管理

课程

学生

评价

授课

选课

成绩

查看

1

1

1

n

n

n

n

n

n

m

m

n

m

图4-2 高校网上选课系统数据库E-R图

（1）任课教师实体图

任课教师实含有的属性有工号、姓名、地址、联系电话、身份证、性别和备注。任课教师实体图如图4-3所示。

任课教师

图4-3任课教师实体图

（2）课程实体图

课程实体包含的属性有课程名称、上课时间、上课地址、授课老师、学分、可上课人数和课程简介。课程实体图如图4-4所示。

课程

4-4课程实体图

（3）学生实体图

学生实体含有的属性有姓名、身份证、班级名称、地址、学号、性别、联系电话和备注。学生实体图如图4-5所示。

学生

4-5学生实体图

## 4.3逻辑结构设计

### 4.3.1 设计思路

根据之前由自底向上的方法得出的数据库的概念结构设计，本系统的逻辑结构主要包含5张表：任课教师表（t\_teacher）, 课程信息表（t\_kecheng）, 选课信息表（t\_xuanke）, 学生信息表（t\_student）, 用户表（t\_user）。

### 4.3.2 逻辑模型

（1）任课教师信息表

任课教师信息表如表4-1所示，该表主要由教务管理员登录系统后在相关界面添加教师信息，主要字段包括：id、地址、备注、联系电话、性别、身份证、教师姓名、工号等。

表4-1任课教师信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 长度 | 类型 | 空 | 备注 |
| tno | 255 | varchar | yes | 工号（登录时账号） |
| address | 255 | varchar | yes | 地址 |
| id | 11 | int | no | 主键 |
| deletestatus | 11 | int | yes | 是否删除状态 0表示未删除 1表示已删除 |
| sex | 255 | varchar | yes | 性别 |
| beizhu | 255 | varchar | no | 备注信息 |
| sfz | 255 | varchar | yes | 任课教师身份证 |
| tname | 255 | varchar | yes | 任课教师姓名 |
| lianxidianhua | 255 | varchar | yes | 任课教师联系电话 |

（2）课程信息表

课程信息表如表4-2所示，该表主要由教务管理员在登录系统在相关界面添加课程信息，其中主要字段包括:id、已报名人数、可上课人数、上课地点、上课时间、课程简介、课程名称、学分、任课教师id等。

表4-2课程信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 长度 | 类型 | 空 | 备注 |
| teacherid | 11 | int | yes | 任课老师，外键 |
| id | 11 | int | no | 主键 |
| jianjie | 255 | varchar | yes | 课程简介 |
| kname | 255 | varchar | yes | 课程名字 |
| renshu | 255 | varchar | no | 可上课人数 |
| shijian | 255 | varchar | yes | 上课时间 |
| baoming | 11 | int | no | 已报名人数 |
| xuefen | 255 | varchar | yes | 学分 |
| didian | 255 | varchar | yes | 上课地点 |
| deletestatus | 11 | int | no | 是否删除标志 0未删除 1已删除 |

（3）选课信息表

选课信息表如表4-3所示，在学生登陆后，查询课程信息并进行选课操作后，相关的数据便存储在这张表中，主要包括字段有：id、评价、选课时间、课程id、用户id、成绩等。

表4-3选课信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 长度 | 类型 | 空 | 备注 |
| userid | 11 | int | yes | 选课的用户，外键 |
| shijian | 255 | varchar | yes | 选课时间 |
| id | 11 | int | no | 主键 |
| chengji | 255 | varchar | yes | 成绩 |
| deletestatus | 11 | int | no | 是否删除状态 0表示未删除 1表示已删除 |
| pingjia | 255 | varchar | yes | 评价 |
| kechengid | 11 | int | yes | 关联的课程，外键 |

1. 学生信息表

学生信息表如表4-4所示，该表主要为教务管理员登录系统后，在相关界面管理学生信息并保存在这张表中，主要字段包括：姓名、性别、班级名称、身份证、id、学号、地址、联系电话和备注。

表4-4学生信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 长度 | 类型 | 空 | 备注 |
| beizhu | 255 | varchar | yes | 备注 |
| id | 11 | int | no | 主键 |
| deletestatus | 11 | int | no | 是否删除状态 0表示未删除 1表示删除 |
| address | 255 | varchar | yes | 地址 |
| banjimingchen | 255 | varchar | yes | 班级名称 |
| lianxidianhua | 255 | varchar | yes | 联系电话 |
| sfz | 255 | varchar | yes | 身份证 |
| sname | 255 | varchar | yes | 学生姓名 |
| sno | 255 | varchar | yes | 学号 |
| sex | 255 | varchar | yes | 性别 |

（5）用户表

该表主要用来存储所有用户的信息，其中用户包括管理员、任课教师、学生及教务管理员，其中主要存储的字段包括：用户标识（1为系统管理员、2为学生用户、3 为任课老师用户）、id、添加时间、姓名、用户名（学生为学号、任课教师为工号）、学生id、任课教师id。

表4-5用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 长度 | 类型 | 空 | 备注 |
| role | 11 | int | no | 1系统管理员 2学生用户 3 任课老师用户 4教务管理员 |
| id | 11 | int | no | 主键 |
| deletestatus | 11 | int | no | 是否删除标志，0表示未删除 1表示删除 |
| truename | 255 | varchar | yes | 姓名 |
| password | 255 | varchar | yes | 密码 |
| studentid | 11 | int | yes | 关联的学生信息，外键 |
| createtime | 255 | datetime | yes | 添加时间 |
| teacherid | 11 | int | yes | 关联的教师信息，外键 |
| username | 255 | varchar | yes | 用户名 |

## 4.4物理结构设计

在对高校网上选课系统进行了数据库库的逻辑结构设计及概念结构设计之后，需要将数据库的逻辑结构设计在具体的环境中实现，因此还需要实现数据库的物理结构设计。数据库逻辑结构设计在是在具体的工作环境中实现的一个过程便可称之为数据库物理结构设计，它的设计质量对系统的影响同样直接决定了整个系统的质量。

根据所使用的计算机系统的特点，为之前设计的高校选课系统的数据库模型确定合理的存储方式和存储结构，在尽可能的减少物理数据库所占的存储空间的同时，还要对数据库的操作尽可能高的速度。因此需要充分了解数据库管理系统的特征并由此来设计高校网上选课系统的数据库物理结构设计。

### 4.4.1 存取方式

根据高校网上选课系统的设计特要求，在各个界面中有多处数据库查询的特点，因此采用索引方式作为数据库的存取方式，这样可以极大地减少在进行数据查询时所消耗的时间，可以极大地提高在数据操作时的响应时间，提高系统的运行效率。

### 4.4.2 存储结构

高校网上选课系统的数据存储结构需要确定数据的存放位置和存储结构以及确定存储参数的配置。高校网上选课系统主要应用于开设了自主的高校设计，因此在高校选课的高峰期会对系统会相对较大，对数据存储产生也会产生很大的压力，根据这个要求，需要将主要的数据表存放在不同的磁盘上面，在提高了系统库流畅度的同时也加快了数据的存取速度，提高了工作效率。

5 界面设计

## 5.1 界面关系图

高校网上选课系统首先需在系统的登录界面中根据自己的身份选择相对应的选项登录系统，其中选项分别有任课教师、学生、系统管理员及教务管理员，用户以自己相对应的身份进行选择并输入用户名及密码进行登录。当系统管理员登录后则可对系统进行管理，在其主界面可以教务管理员并对其进行相关管理操作。当教务管理员登录后便可以对课程信息、任课教师信息及学生信息进行添加、修改、查询及删除，还可以对学生选课、成绩等相关信息进行查询。若为任课教师登录则可对个人信息进行管理，查询自己任课信息、任教课程的选课信息及对学生成绩进行评价。若为学生登录则可对个人信息进行管理、查询相关课程信息并进行选课，当教师对自己选课成绩做出评价后即可查询到相关课程的成绩。每个用户在登陆后均可在相关界面修改自己的登录密码及相关个人信息，界面关系图如图5-1所示。

登录首页

用户名称

密码

用户角色

任课教师

教务管理员

系统管理员

学生

教务员管理

密码修改

选课记录查询

课程管理

任课教师管理

学生管理

密码修改

成绩记录查询

课程信息查询

课程查询

个人信息维护

密码修改

成绩管理

密码修改

个人信息维护

成绩查询

选课记录查询

选课管理

图5-1 高校网上选课系统界面关系图

## 5.2 界面设计成果

### 5.2.1 主界面

根据不同的用户登录系统，系统将对其权限判断并显示对应权限的菜单，菜单列表的显示与该用户对相应功能进行操作。由图可见，主界面的菜单列表具有简洁、人性化的特点，用户可直观的根据其需求进行添加、查询等操作。主界面图如图5-2所示。 

图5-2 高校网上选课系统主界面图

### 5.2.2 子界面

（1）用户登录界面

在用户打开系统后，首先必须在该界面选择用户角色并输入与之相对应的信息方可进入系统，在输入完成选择确认登陆后系统将对用户输入的信息判断并判别该用户的权限。用户登录界面如图5-3所示。



图5-3 用户登录界面

（2）教师信息管理界面

该界面为教务管理员登陆后对教师信息进行管理，主要可实现教师信息的添加（教师信息主要包括：工号、姓名、性别、身份证、地址、手机号码和备注）、修改、删除和查询等功能。教师信息管理界面分别如图5-4，5-5，5-6所示。



图5-4教师信息管理界面

图5-5教师信息添加界面



图5-6 教师信息修改界面

（3）教务员信息管理界面

该界面为系统管理员登录后在其主界面中点击教务员管理便可进入该界面对教务员进行管理，主要可实现的对教务员信息的添加（用户名、姓名）、查询、修改及删除。教务员信息管理界面分别如图5-7，5-8，5-9所示。



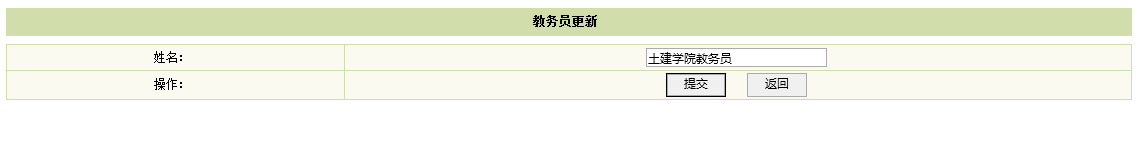
图5-7 教务员信息管理界面

图5-8 教务员信息修改界面

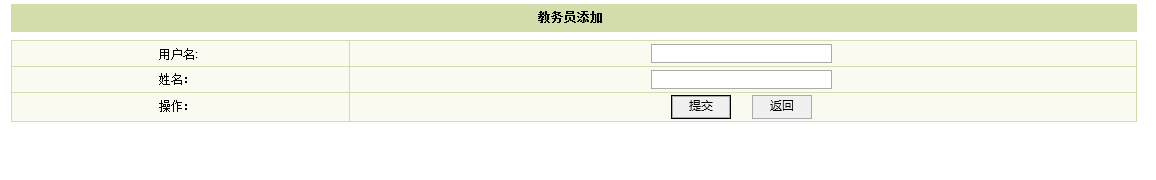


图5-9教务员信息添加界面

（4）选课功能界面

在学生用户登录系统后，可在其菜单列表中选择选课管理并进入该界面，在查询框中输入课程名称后可查询课程信息课，然后进行选课操作。选课功能界面如图5-10所示。



图5-10选课功能界面

（5）课程信息管理界面

该界面为教务管理员登陆后对课程信息进行相关管理操作，主要实现课程信息的添加（课程信息主要包括：课程名字、上课时间、上课地点、授课教师、学分、可上课人数和课程简介）、修改、删除和查询。课程信息管理界面分别如图5-11，5-12，5-13所示。



图5-11课程信息管理界面



图5-12课程信息添加界面



图5-13 课程信息修改界面

（6）学生信息管理界面

该界面该界面为教务管理员登陆后对学生信息进行管理，主要可实现学生信息的添加（学生信息主要包括：学号、姓名、性别、身份证、地址、手机号码、班级名称和备注）、修改、删除和查询。学生信息管理界面分别如图5-14，5-15，5-16所示。



图5-14学生信息添加界面



图5-15学生信息管理界面



图5-16学生信息修改界面

（7）学生已选课程信息界面

学生已选课程信息界面为学生登陆后，可以查询已选择的课程并可对已选课程进行管理。学生已选课程信息界面如图5-17所示。



图5-17学生已选课程的信息界面

6 详细设计

## 6.1 系统主要功能模块介绍

（1）密码修改模块，用户可以通过此模块对用户密码进行修改，每一位用户的初始密码均为111111。

（2）用户登录模块，用户通过此模块输入对应的用户名称及密码登录系统并获得不同的用户权限，系统用户主要包括：学生、系统管理员、任课教师和教务管理员。

（3）学生信息管理，该模块主要为教务管理员登录系统后通过此模块将学生信息输入系统并存入数据库，学生信息具体包括：姓名、备注、班级、身份证、联系电话、住址和学号。并且可以通过此模块对学生信息进行查询及修改。

（4）课程信息管理模块，该模块主要在教务管理员登录系统后通过此模块将课程信息进行输入系统并存入数据库，其中课程信息具体包括：课程名称、学分、课程简介、授课老师、上课地点、上课人数和上课时间，并可以对录入的课程信息进行查询、修改及删除。

（5）选课记录查询，不同的用户可以通过此模块选课记录进行查询。

（6）任课老师模块，教务管路员可通过此模块对任课老师信息进行录入，其中任课教师信息包括：工号（教师登录时的用户名）、姓名、性别身份证、地址、联系电话和备注，并且可以对录入的信息进行查询和修改。

（7）成绩查询模块，任课老师通过此系统对学生学习进行评分，学生及管理员可通过此模块对成绩进行查询。

（8）教务系统管理模块，此模块由系统管理员主要负责，其功能是对教务管理员进行管理，其中包括添加、修改、删除和查询教务管理员。

（9）选课管理模块，此模块主要为学生在登陆后通过查询课程名称来进行选课操作。

## 6.2 用户登录模块设计

### 6.2.1 用户登录算法描述

高校网上选课系统的登录模块是用户使用系统时接触的第一个模块，在打开系统登录界面后，首先用户需在登录界面根据自己的身份选择相对应的选项并输入用户名称及密码，在确认登陆后，系统将从页面中获取用户输入的信息并调用userDao中的selectBean方法将其与数据库中数据进行对比，如果该条数据在数据库中存在，用户便成功登录系统并进入其用户主界面中，用户可进行接下来的操作，若输入的登录信息在对比后不存在，则将跳转回登陆页面并提示用户信息错误。

### 6.2.2 登录程序流程图

登录模块的程序流程图如图6-1所示。

开始

输入用户登录信息

验证是否正确

成功登陆

各种操作

系统主界面

结束

F

T

图6-1 高校网上选课系统登录流程图

### 6.2.3 登录模块关键类说明

系统登录模块中主要包含了UserDao、UserDaoImpl、User这三个类。不同的用户根据其相对应的身份进行选择后输入正确的信息便可通过此模块登录系统进入与其相关的字界面中进行其他操作。

## 6.3 学生信息管理模块设计

### 6.3.1 学生信息管理算法描述

高校网上选课系统中学生信息管理功能是在教务管理员登录后进行操作的，在教务管理员在登录后可通过本模块添加学生信息并对其进行查询、修改及删除。首先在教务管理员登录后在其主界面中选择学生管理则可进入模块的相关界面，在界面中选择添加新学生便可对学生信息进行添加，教务管理员可在界面中输入学号或学生姓名来查询学生信息，输入查询信息后，通过相关的判别语句判断输入值是否为空，系统会自动请求数据库并调用StudentDao中selectBeanList方法查询数据库中的记录，并通过request.setAttribute方法将该记录在学生信息查询的显示界面中显示出来，这样可对学生信息进行查询，在该界面还可实现对学生信息的修改及删除。

### 6.3.2 学生信息管理程序流程图

学生信息管理模块的程序流程图如图6-2所示。

添加学生信息

删除学生信息

修改学生信息

教务管理员登录系统

教务管理员主界面

查询学生信息

开始

是否通过验证

F

T

结束

图6-2学生信息管理流程图

6.3.3 学生信息管理关键类说明

学生信息管理模块中主要包含Student、StudentDao、StudentDaoImpl这三个类，教务管理员在该模块的添加信息界面可添加身份证号、学号、地址、性别、联系方式、学生姓名等信息，同时可在可在该模块的其他功能界面对任课教师信息修改、删除及查询。

## 6.4 课程信息管理模块设计

### 6.4.1 课程信息管理算法描述

根据高校网上选课系统需求分析后可知课程信息管理模块需能实现课程信息的新增，修改，删除和查询。该模块为教务员登录后进行操作，在教务员的主界面选择课程管理即可进入该模块的界面，其中在查询框中课程名称后，系统会通过一个判别语句判别输入是否为空，若不为空则系统将请求数据库并调用KechengDao中的selectBeanLis方法查询相关记录，若查询到了则通过request.setAttribute将记录传回相关界面，这样便完成了对课程的查询，同时在该界面中还可完成对课程信息的添加、删除及修改。

### 6.4.2 课程信息管理程序流程图

课程信息管理模块的程序流程图如图6-3所示。

添加课程信息

删除课程信息

修改课程信息

教务管理员登录系统

教务管理员主界面

查询课程信息

开始

是否通过验证

F

T

结束

图6-3课程信息管理流程图

### 6.4.3 课程信息管理关键类说明

课程信息管理模块中主要包含Kecheng、KechengDao、KechengDaoImpl这三个类,可以记录课程名称、上课时间、上课地点、授课教师、学分、可上课人数等信息，同时实现信息的添加、修改、删除和查询。

## 6.5 选课管理模块设计

6.5.1 选课管理算法描述

高校网上选课系统中本模块为学生用户登录后进行操作，首先在学生登陆后在其主界面中选择选课管理后出现相关界面，界面中的查询框中输入课程名称进行查询，系统将通过request.getParameter语句从页面中获取查询项并通过条件语句判断输入的查询项是否为空，若输入不为空时则系统将数据库的查询语句通过StiingBuffer来叠加组装，并使用kechengDao中的selectBeanList方法调用数据库查询语句查找相关记录并通相关方法将查询到的记录传回显示界面，这个过程便实现了对学生用户对课程的查询，同时在该界面中还可对课程进行选择及查看课程的详细信息。

6.5.2 选课管理程序流程图

选课模块的程序流程图如图6-4所示。

查询课程信息

更改选课信息

选课记录查询

学生登录系统

学生用户主界面

选课操作

开始

是否通过验证

F

T

结束

图6-4选课功能程序流程图

### 6.5.3 选课管理关键类说明

选课管理模块中主要包含Xuanke、XuankeDao、XuankeDaoImpl这三个类，通过此模块，学生用户可对课程信息进行查询并查看相关课程的详细信息，在选课成功后可对选课记录进行查询及修改等操作。

## 6.6 任课教师信息管理模块设计

### 6.6.1 任课教师信息管理算法描述

高校网上选课系统中任课教师信息管理功能是在教务管理员登录后进行操作的，在教务管理员在登录后可通过本模块添加任课教师信息并对其进行查询、修改及删除。首先在教务管理员登录后在其主界面中选择任课教师管理则可进入此模块的相关界面，在界面中选择添加新教师便可对任课教师信息进行添加，教务管理员可在界面中输入工号或教师姓名来查询任课教师信息，在输入查询信息后，系统会自动请求数据库并调用TeacherDao中selectBeanList方法，该方法主要功能是将从数据库中查询符合条件的数据，然后通过request.setAttribute方法将查询到的数据在查询结果的界面中显示出来，这样可对任课教师信息进行查询，在该界面还可实现对任课教师信息的修改及删除。

### 6.6.2 任课教师教师信息管理程序流程图

任课教师教师信息模块的程序流程图如图6-5所示。

添加教师信息

删除教师信息

修改教师信息

教务管理员登录系统

教务管理员主界面

修改教师信息

开始

是否通过验证

F

T

结束

图6-5任课教师信息管理流程图

### 6.6.3 任课教师信息管理关键类说明

任课教师信息管理模块中主要包含Teacher, TeacherDao、TeacherDaoImpl这三个类。教务管理员在该模块的添加信息界面可添加身份证号、工号、地址、性别、联系方式、教师姓名等信息，同时可在可在该模块的其他功能界面对任课教师信息修改、删除及查询。

7 编码

## 7.1 代码实现与核心算法

1. 学生管理模块代码：

//删除学生信息操作

public void studentdelete() throws IOException {

HttpServletRequest request = ServletActionContext.getRequest();

Student bean = studentDao.selectBean(" where id= "

+ request.getParameter("id"));

bean.setDeletestatus(1);

studentDao.updateBean(bean);

User user = userDao.selectBean(" where username='"+bean.getSno()+"' ");

user.setDeletestatus(1);

userDao.updateBean(user);

HttpServletResponse response = ServletActionContext.getResponse();

response.setCharacterEncoding("gbk");response.setContentType("text/html; charset=gbk");

response

.getWriter() }

(2课程信息管理模块的代码如下：

//删除课程信息操作

public void kechengdelete() throws IOException {

HttpServletRequest request = ServletActionContext.getRequest();

Kecheng bean = kechengDao.selectBean(" where id= "

+ request.getParameter("id"));

bean.setDeletestatus(1);

kechengDao.updateBean(bean);

HttpServletResponse response = ServletActionContext.getResponse();

response.setCharacterEncoding("gbk");response.setContentType("text/html; charset=gbk");

response

.getWriter() }

## 7.2 代码优化分析

本系统中有多处需要对数据库进行访问，当进行数据库请求时，需先得到数据库连接，在进行数据库的相关操作，在操作完成后，又需关闭数据库，这样的话效率较低。可将对数据库的相关操作函数编写成类，在用方法进行调用，提高系统运行效率。

8 测试

## 8.1 测试方案设计

优秀的软件测试方案将有利于系统性能的提升及功能的优化，同时测试方案的确定也是系统测试的核心内容。它能够检测系统的设计是否符合需求分析的要求，测试过程中应详细的对系统的各个模块进行测试并记录好测试结果，以减少在测试过程中的出现的错误。

### 8.1.1 测试策略

针对高校网上选课系统的相关特征将采用静态测试与动态测试相结合的方式进行检测。在系统进行静态测试时，无需将系统打开运行，同时也不需要设计相关的测试用例等操作，可对系统功能进行分析后，将需求分析详细设计相结合，根据逻辑判断分析系统的详细流程及操作。而在进行动态测试时，则需将系统打开进行动态运行并将准备好的测试用例输入系统，将结果与预期对比并检查是否存在差异，这样便可检测系统运行是否正确可靠，同时还可以借此分析系统的运行效率等其他性能。在动态测试阶段，本系统将采用黑盒测试及白盒测试交替进行的方式进行测试，根据不同的模块功能进行测试，黑盒测试主要是测试模块已实现的功能，检测其是否达到了需求分析的要求并完整运行。白盒测试则需结合代码进行测试，需要知道系统及模块的结构。

本次系统的动态测试将首先根据系统的功能模块的设计相对应的测试用例并预测其输出结果，然后再将事先设计好的测试用例在系统中运行并记录其运行结果，最后将运行的结果与事先预测的结果进行对比后分析得出相对应的结论并记录在已经设计好的测试用例表中。

考虑到本系统只有本人一人开发，系统较为精简，界面结构等都比较简洁，所以本系统不会采用任何自动化测试工具进行测试，只使用传统的记录方式测试。

### 8.1.2 测试进度安排

根据高校网上选课系统在设计时的模块分类，主要将进行以下功能模块的测试，各个模块的测试安排如表8-1所示。

表8-1测试进度安排表

|  |  |  |
| --- | --- | --- |
| 模块名称 | 测试员 | 测试时间 |
| 系统用户登录模块 | 杨易麟 | 2016.4.20—2016.4.21 |
| 任课教师信息管理模块 | 杨易麟 | 2016.4.22—2014.4.23 |
| 学生选课模块 | 杨易麟 | 2016.4.22—2016.4.23 |
| 学生信息管理模块 | 杨易麟 | 2016.4.25—2016.4.25 |
| 课程信息管理模块 | 杨易麟 | 2016.5.2—2016.5.3 |
| 系统管理员管理模块 | 杨易麟 | 2016.5.4—2016.5.4 |

### 8.1.3 测试资源

1. 人力需求

考虑到本系统只有本人一人开发，系统较为精简，界面结构等都比较简洁，所以整个测试过程将由本人一人完成。

### 8.1.4 关键测试点

（1）用户能否根据自己对应的身份进入其主界面。

（2）教务管理员能否对任课教师信息进行管理。

（3）学生用户能否根据查询课程名称进行选课。

（4）教务管理员能否对课程信息进行管理。

（5）教务管理员能否对学生信息进行管理。

（6）系统管理员能否实现对教务管理员的管理。

（7）用户能否在相关界面完成对密码的修改。

## 8.2 测试用例构建

### 8.2.1 测试用例编写约定

在系统的开发过程中，测试用例编写约定主要是将测试用例统一，使其规范化、合理化，为测试人员编写测试用例提供依据及指导，提高测试用例的可执行性及测试效率等。系统测试测试用例主要依据系统的需求分析进行编写，集成测试测试用例主要依据概要设计及结构设计进行编写，而单元测试测试用例则依据详细设计进行编写。

### 8.2.2 测试用例设计

根据系统需求分析及前文中提出的测试关键点的要求，测试用例如表8-2所示，其中需要包括测试测试关键点、测试步骤、预期结果、测试结果。

表8-2 测试用例设计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试编号 | 测试关键点 | 测试步骤 | 预期结果 | 测试结果 |
| test01 |  |  |  |  |
| test02 |  |  |  |  |
| …… | …… | …… | …… |  |

### 8.2.3 关键测试用例

根据前文中表格设计的格式，关键测试用例如表8-3所示。

表8-3关键测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试编号 | 测试关键点 | 测试步骤 | 预期结果 | 测试结果 |
| test01 | 用户登录测试 | （1）打开系统登录界面 | 用户需输入正确的用户名称及密码并选择相对应的身份方可登录系统。 | 与预期结果一致 |
| （2）输入用户名称及密码 |
| （3）选择相对应的身份并确认登录 |
| test02 | 任课教师信息管理操作 | （1）教务管理员登录系统并进入任课教师管理界面 | 教务管理员在登录可成功实现对任课教师信息的添加及管理。 | 与预期结果一致 |
| （2）教务管理员添加新教师操作 |
| （3）教务管理员对任课教师信息进行管理操作，包括查询、修改及删除 |
| test03 | 学生选课操作 | （1）学生用户登录系统并进入选课管理相关界面 | 学生在登录系统后可进入选课管理界面查询课程信息后选课，并可实现相关管理操作。 | 与预期结果一致 |
| （2）以课程名称查询课程并查看课程详细信息 |
| （3）选课并进行相关选课操作 |
| test04 | 课程信息管理操作 | （1）教务管理员登录系统并进入课程信息管理界面 | 教务管理员在登录可成功实现对课程信息的添加及管理。 | 与预期结果一致 |
| （2）教务管理员添加新课程操作 |
| （3）教务管理员对任课教师信息进行管理操作，包括查询、修改及删除 |
| test05 | 学生信息管理操作 | （1）教务管理员登录系统并进入学生管理界面 | 教务管理员在登录可成功实现对学生信息的添加及管理。 | 与预期结果一致 |
| （2）教务管理员添加新学生操作 |
| （3）教务管理员对学生信息进行管理操作，包括查询、修改及删除 |
| test06 | 教务管理员管理 | （1）系统管理员登录系统并进入教务员管理界面 | 系统管理员在登录可成功实现对教务员信息的添加及管理。 | 与预期结果一致 |
| （2）系统管理员添加新教务员操作 |
| （3）系统管理员对教务员信息进行管理操作，包括查询、修改及删除 |
| test07 | 用户密码修改 | （1)用户登录系统进入主界面后选择修改密码 | 用户在修改密码后不能以原密码进入系统，需重新输入修改后密码方可进入。 | 与预期结果一致 |
| （2）在修改密码操作完成后确认并退出系统 |
| （3）重新使用新密码登录系统 |

### 8.2.4 测试用例维护

随着软件测试工作的进行，在软件测试中经常会遇到测试用例的相关错误，当出现错误时则需对测试用例进行维护。当需求发生改变时可能造成某些测试用例不再符合系统测试的要求，此时则需将这类测试用例删除并重新设计。随着测试的进行，系统中测试用例的增加，则会出现测试用例不受控制的情况，此时则需将该类测试用例进行改进，使其符合测试用例的要求。同时还需要定期的清理重复的测试用例及添加新的测试用例。

9 总结与展望

## 9.1 设计工作总结

高校网上选课系统是实现高校信息化教学管理的重要组成部分。而网上选课系统的开发必须从实际出发，熟悉传统选课方式的过程，从中总结经验，找出问题，并通过计算机网络来实现教学选课这项工作的计算机管理，简化管理操作，使相关师生从复杂繁琐的程序中解脱出来，进一步实现高校教学信息化、办公无纸化。

在设计之初，做了充分准备，查阅了许多相关的资料，除了与传统的选课系统进行比较之外，还进入了其他高校的选课系统进行学习，通过四个多月的努力，在老师的帮助和指导下，最终完成了对高校网上选课系统的设计，这不仅提高了本人的程序设计水平及数据库应用能力，也加深我对程序设计的相关工作的理解，使我受益匪浅。

由于时间和水平限制，本系统还有很多不完善的地方，许多细节之处的设计仍然有所缺陷，没有达到设计之初的预期效果，仍然需要在不断地学习之后才能够做的更加人性化，符合自己的目标。希望能够在老师和同学们的鼓励下，继续努力学习。

## 9.2 未来工作展望

这次软件设计虽然达到了预期设计的要求，但是也有很多未解决的难题，山高人为峰，求知的道路是不断进取的，希望自己以后能够继续努力学习，精益求精，不断充实自己的学识，开发出更加优秀的产品。

谢 辞

时光匆匆，大学最后阶段非常感谢指导老师的帮助。从开题到现在，无论是在设计过程中还是论文编写过程中，我都遇到了难题，走了许多弯路，还好有老师的帮助和指导，让我每次在遇到问题之后总是能够及时的解决，让我最终能够完成此次的毕业设计，也让我大学生涯有了一个圆满的结束。

## 在这次毕业设计后，我将不在是学生的身份，我还要感学我们的学校华东交通大学，在他的关怀下，我得以在安全，温馨的环境下学习生活。这次的毕业设计，让我受益匪浅，提高了我软件项目的开发能力，虽说遇到了许多困难，但都在老师和同学的帮助下解决了，同时在此我也感谢那些给过我帮助的同学。

参考文献

[1]刘晓华．JSP应用开发详解．北京电子工业出版社，2005．

[2]滕佳东． 管理信息系统．东北财经大学出版社，2008．2．

[3]董豪．基于JSP技术动态教学管理系统设计与实现[D]．成都：电子科技大学，2012．

[4]孙鑫．Java Web开发详解．电子工业出版社，2004．

[5]席晓峰．使用 J2EE 框架技术构件可重用的 web 应用．计算机工程与应用，2005 (29)，208-214

[6]向传杰．JSP-Servlet 基础案例教程.北京：电子工业出版社，2011．

[7]王军鹏．刘路放．基于统一建模语言的关系数据库模型设计．现代计算机，2001，1：16~18

[8]Jacob Kohler, Matthias Lange, Ralf Hofestadt, Steffen Schulze-Kremer. Logical and Semantic Database Integration. 2000.

[9]Ali Bazghandi. Web Database Connectivity Methods (using Mysql) in Windows Platform. 2006.

[10]孙卫琴．精通struts.电子工业出版社，2004．

[11]蔡剑．景楠．Java Web应用开发：J2EE和Tomcat[M]．清华大学出版社，2004．

附录A 外文翻译—原文部分

**Struts——an open-source MVC implementation**

**By**: Malcolm Davis.

**Source:** Struts--an open-source MVC implementation[J].IBM Systems Journal

This article introduces Struts, a Model-View-Controller implementation that uses servlets and JavaServer Pages (JSP) technology. Struts can help you control change in your Web project and promote specialization. Even if you never implement a system with Struts, you may get some ideas for your future servlets and JSP page implementation.

1）Introduction

Kids in grade school put HTML pages on the Internet. However, there is a monumental difference between a grade school page and a professionally developed Web site. The page designer (or HTML developer) must understand colors, the customer, product flow, page layout, browser compatibility, image creation, JavaScript, and more. Putting a great looking site together takes a lot of work, and most Java developers are more interested in creating a great looking object interface than a user interface. Java Server Pages (JSP) technology provides the glue between the page designer and the Java developer.

If you have worked on a large-scale Web application, you understand the term *change*. Model-View-Controller (MVC) is a design pattern put together to help control change. MVC decouples interface from business logic and data. Struts is an MVC implementation that uses Servlets 2.2 and JSP 1.1 tags, from the J2EE specifications, as part of the implementation. You may never implement a system with Struts, but looking at Struts may give you some ideas on your future Servlets and JSP implementations.

2）Model-View-Controller (MVC)

JSP tags solved only part of our problem. We still have issues with validation, flow control, and updating the state of the application. This is where MVC comes to the rescue. MVC helps resolve some of the issues with the single module approach by dividing the problem into three categories:

（1）Model

The model contains the core of the application's functionality. The model encapsulates the state of the application. Sometimes the only functionality it contains is state. It knows nothing about the view or controller.

（2）View

The view provides the presentation of the model. It is the *look* of the application. The view can access the model getters, but it has no knowledge of the setters. In addition, it knows nothing about the controller. The view should be notified when changes to the model occur.

(3)Controller

The controller reacts to the user input. It creates and sets the model.

3）MVC Model 2

The Web brought some unique challenges to software developers, most notably the stateless connection between the client and the server. This stateless behavior made it difficult for the model to notify the view of changes. On the Web, the browser has to re-query the server to discover modification to the state of the application.

Another noticeable change is that the view uses different technology for implementation than the model or controller. Of course, we could use Java (or PERL, C/C++ or what ever) code to generate HTML. There are several disadvantages to that approach:

1. Java programmers should develop services, not HTML.
2. Changes to layout would require changes to code.
3. Customers of the service should be able to create pages to meet their specific needs.
4. The page designer isn't able to have direct involvement in page development.
5. HTML embedded into code is ugly.

For the Web, the classical form of MVC needed to change. Figure 1 displays the Web adaptation of MVC, also commonly known as MVC Model 2 or MVC 2.

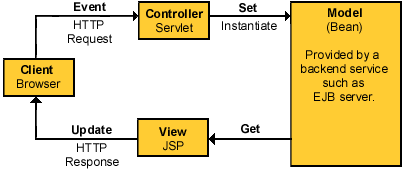


Figure 1. MVC Model 2

4)Struts, an MVC 2 implementation

Struts is a set of cooperating classes, servlets, and JSP tags that make up a reusable MVC 2 design. This definition implies that Struts is a framework, rather than a library, but Struts also contains an extensive tag library and utility classes that work independently of the framework. Figure 2 displays an overview of Struts.

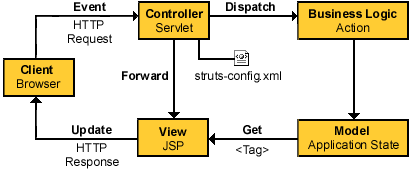


Figure 2. Struts overview

5)Struts overview

(1)Client browser

An HTTP request from the client browser creates an event. The Web container will respond with an HTTP response.

(2)Controller

The Controller receives the request from the browser, and makes the decision where to send the request. With Struts, the Controller is a command design pattern implemented as a servlet. The struts-config.xml file configures the Controller.

(3)Business logic

The business logic updates the state of the model and helps control the flow of the application. With Struts this is done with an Action class as a thin wrapper to the actual business logic.

(14)Model state

The model represents the state of the application. The business objects update the application state. ActionForm bean represents the Model state at a session or request level, and not at a persistent level. The JSP file reads information from the ActionForm bean using JSP tags.

(5)View

The view is simply a JSP file. There is no flow logic, no business logic, and no model information -- just tags. Tags are one of the things that make Struts unique compared to other frameworks like Velocity.

6)Struts details

Displayed in Figure 6 is a stripped-down UML diagram of the org.apache.struts.action package. Figure 3 shows the minimal relationships among ActionServlet (Controller), ActionForm (Form State), and Action (Model Wrapper).

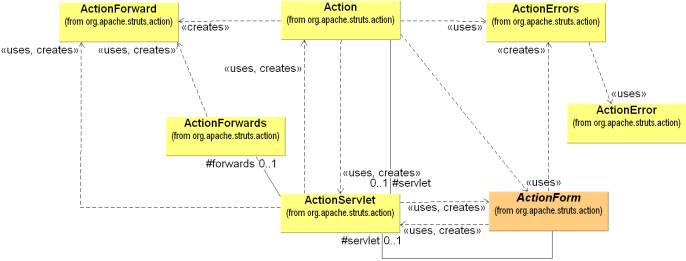


Figure 3. UML diagram of the relationship of the Command (ActionServlet) to the Model (Action & ActionForm)

(1)The ActionServlet class

Do you remember the days of function mappings? You would map some input event to a pointer to a function. If you where slick, you would place the configuration information into a file and load the file at run time. Function pointer arrays were the good old days of structured programming in C.

Life is better now that we have Java technology, XML, J2EE, and all that. The Struts Controller is a servlet that maps events (an event generally being an HTTP post) to classes. And guess what -- the Controller uses a configuration file so you don\_t have to hard-code the values. Life changes, but stays the same.

ActionServlet is the Command part of the MVC implementation and is the core of the Framework. ActionServlet (Command) creates and uses Action, an ActionForm, and ActionForward. As mentioned earlier, the struts-config.xml file configures the Command. During the creation of the Web project, Action and ActionForm are extended to solve the specific problem space. The file struts-config.xml instructs ActionServlet on how to use the extended classes. There are several advantages to this approach:

1. The entire logical flow of the application is in a hierarchical text file. This makes it easier to view and understand, especially with large applications.
2. The page designer does not have to wade through Java code to understand the flow of the application.
3. The Java developer does not need to recompile code when making flow changes.

Command functionality can be added by extending ActionServlet.

(2)The ActionForm class

ActionForm maintains the session state for the Web application. ActionForm is an abstract class that is sub-classed for each input form model. When I say *input form model*, I am saying ActionForm represents a general concept of data that is set or updated by a HTML form. For instance, you may have a UserActionForm that is set by an HTML Form. The Struts framework will:

1. Check to see if a UserActionForm exists; if not, it will create an instance of the class.
2. Struts will set the state of the UserActionForm using corresponding fields from the HttpServletRequest. No more dreadful request.getParameter() calls. For instance, the Struts framework will take fname from request stream and call UserActionForm.setFname().
3. The Struts framework updates the state of the UserActionForm before passing it to the business wrapper UserAction.
4. Before passing it to the Action class, Struts will also conduct form state validation by calling the validation() method on UserActionForm. Note: This is not always wise to do. There might be ways of using UserActionForm in other pages or business objects, where the validation might be different. Validation of the state might be better in the UserAction class.
5. The UserActionForm can be maintained at a session level.

Notes:

1. The struts-config.xml file controls which HTML form request maps to which ActionForm.
2. Multiple requests can be mapped UserActionForm.
3. UserActionForm can be mapped over multiple pages for things such as wizards.

(3)The Action class

The Action class is a wrapper around the business logic. The purpose of Action class is to translate the HttpServletRequest to the business logic. To use Action, subclass and overwrite the process() method.

The ActionServlet (Command) passes the parameterized classes to ActionForm using the perform() method. Again, no more dreadful request.getParameter() calls. By the time the event gets here, the input form data (or HTML form data) has already been translated out of the request stream and into an ActionForm class.

Note: "Think thin" when extending the Action class. The Action class should control the flow and not the logic of the application. By placing the business logic in a separate package or EJB, we allow flexibility and reuse.

Another way of thinking about Action class is as the Adapter design pattern. The purpose of the Action is to "Convert the interface of a class into another interface the clients expect. Adapter lets classes work together that couldn\_t otherwise because of incompatibility interface" (from *Design Patterns - Elements of Reusable OO Software* by Gof). The client in this instance is the ActionServlet that knows nothing about our specific business class interface. Therefore, Struts provides a business interface it does understand, Action. By extending the Action, we make our business interface compatible with Struts business interface. (An interesting observation is that Action is a class and not an interface. Action started as an interface and changed into a class over time. Nothing's perfect.)

(4)The Error classes

The UML diagram (Figure 3) also included ActionError and ActionErrors. ActionError encapsulates an individual error message. ActionErrors is a container of ActionError classes that the View can access using tags. ActionErrors is Struts way of keeping up with a list of errors.

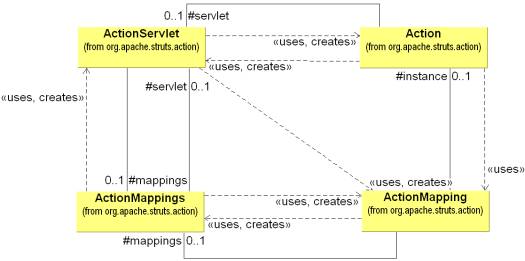


Figure 4. UML diagram of the relationship of the Command (ActionServlet) to the Model (Action)

(5)The ActionMapping class

An incoming event is normally in the form of an HTTP request, which the servlet Container turns into an HttpServletRequest. The Controller looks at the incoming event and dispatches the request to an Action class. The struts-config.xml determines what Action class the Controller calls. The struts-config.xml configuration information is translated into a set of ActionMapping, which are put into container of ActionMappings. (If you have not noticed it, classes that end with *s* are containers)

The ActionMapping contains the knowledge of how a specific event maps to specific Actions. The ActionServlet (Command) passes the ActionMapping to the Action class via the perform() method. This allows Action to access the information to control flow.

(6)ActionMappings

ActionMappings is a collection of ActionMapping objects.

7)Struts pros

Use of JSP tag mechanism

The tag feature promotes reusable code and abstracts Java code from the JSP file. This feature allows nice integration into JSP-based development tools that allow authoring with tags.

(1)Tag library

Why re-invent the wheel, or a tag library? If you cannot find something you need in the library, contribute. In addition, Struts provides a starting point if you are learning JSP tag technology.

(2)Open source

You have all the advantages of open source, such as being able to see the code and having everyone else using the library reviewing the code. Many eyes make for great code review.

(3)Sample MVC implementation

Struts offers some insight if you want to create your own MVC implementation.

(4)Manage the problem space

Divide and conquer is a nice way of solving the problem and making the problem manageable. Of course, the sword cuts both ways. The problem is more complex and needs more management.

8)Struts cons

(1)Youth

Struts development is still in preliminary form. They are working toward releasing a version 1.0, but as with any 1.0 version, it does not provide all the bells and whistles.

(2)Change

The framework is undergoing a rapid amount of change. A great deal of change has occurred between Struts 0.5 and 1.0. You may want to download the most current Struts nightly distributions, to avoid deprecated methods. In the last 6 months, I have seen the Struts library grow from 90K to over 270K. I had to modify my examples several times because of changes in Struts, and I am not going to guarantee my examples will work with the version of Struts you download.

(3)Correct level of abstraction

Does Struts provide the correct level of abstraction? What is the proper level of abstraction for the page designer? That is the $64K question. Should we allow a page designer access to Java code in page development? Some frameworks like Velocity say no, and provide yet another language to learn for Web development. There is some validity to limiting Java code access in UI development. Most importantly, give a page designer a little bit of Java, and he will use a lot of Java. I saw this happen all the time in Microsoft ASP development. In ASP development, you were supposed to create COM objects and then write a little ASP script to glue it all together. Instead, the ASP developers would go crazy with ASP script. I would hear"Why wait for a COM developer to create it when I can program it directly with VBScript?" Struts helps limit the amount of Java code required in a JSP file via tag libraries. One such library is the Logic Tag, which manages conditional generation of output, but this does not prevent the UI developer from going nuts with Java code. Whatever type of framework you decide to use, you should understand the environment in which you are deploying and maintaining the framework. Of course, this task is easier said than done.

(4)Limited scope

Struts is a Web-based MVC solution that is meant be implemented with HTML, JSP files, and servlets.

(5)J2EE application support

Struts requires a servlet container that supports JSP 1.1 and Servlet 2.2 specifications. This alone will not solve all your install issues, unless you are using Tomcat 3.2. I have had a great deal of problems installing the library with Netscape iPlanet 6.0, which is supposedly the first J2EE-compliant application server. I recommend visiting the Struts User Mailing List archive (see [Resources](http://www.ibm.com/developerworks/library/j-struts/index.html?S_TACT=105AGX52&S_CMP=cn-a-j#resources#resources)) when you run into problems.

(6)Complexity

Separating the problem into parts introduces complexity. There is no question that some education will have to go on to understand Struts. With the constant changes occurring, this can be frustrating at times. Welcome to the Web.

(7)Where is...

I could point out other issues, for instance, where are the client side validations, adaptable workflow, and dynamic strategy pattern for the controller? However, at this point, it is too easy to be a critic, and some of the issues are insignificant, or are reasonable for a 1.0 release. The way the Struts team goes at it, Struts might have these features by the time you read this article, or soon after.

9)Future of Struts

Things change rapidly in this new age of software development. In less than 5 years, I have seen things go from cgi/perl, to ISAPI/NSAPI, to ASP with VB, and now Java and J2EE. Sun is working hard to adapt changes to the JSP/servlet architecture, just as they have in the past with the Java language and API. You can obtain drafts of the new JSP 1.2 and Servlet 2.3 specifications from the Sun Web site. Additionally, a standard tag library for JSP files is appearing.

附录B 外文翻译—译文部分

**Struts——MVC 的一种开放源码实现**

**作者：**Malcolm Davis

**来源：**Struts--an open-source MVC implementation[J]. IBM Systems

本文介绍Struts，它是使用servlet和avaServer Pages技术的一Model-View-Controller实现。Struts可帮助您控制Web项目中的变化并提高专业化水平。尽管您可能永远不会用Struts实现一个系统，但您可以将其中的一些思想用于您以后的servlet和JSP网页的实现中。

1）简介

小学生也可以在因特网上发布HTML网页。但是，小学生的网页和专业开发的网站有质的区别。网页设计人员（或者 HTML 开发人员）必须理解颜色、用户、生产流程、网页布局、浏览器兼容性、图像创建和JavaScript等等。设计漂亮的网站需要做大量的工作，大多数Java开发人员更注重创建优美的对象接口，而不是用户界面。JavaServer Pages(JSP)技术为网页设计人员和Java开发人员提供了一种联系钮带。

如果您开发过大型Web应用程序，您就理解变化这个词的含义。“模型-视图-控制器”(MVC)就是用来帮助您控制变化的一种设计模式。MVC 减弱了业务逻辑接口和数据接口之间的耦合。Struts是一种MVC实现，它将 Servlet2.2和JSP1.1标记（属于 J2EE 规范）用作实现的一部分。尽管您可能永远不会用Struts实现一个系统，但了解一下Struts或许使您能将其中的一些思想用于您以后的 Servlet的JSP实现中。

2）模型-视图-控制器 (MVC)

JSP标记只解决了部分问题。我们还得处理验证、流程控制和更新应用程序的状态等问题。这正是MVC发挥作用的地方。MVC 通过将问题分为三个类别来帮助解决单一模块方法所遇到的某些问题：

（1）Model（模型）

模型包含应用程序的核心功能。模型封装了应用程序的状态。有时它包含的唯一功能就是状态。它对视图或控制器一无所知。

（2）View（视图）

视图提供模型的表示。它是应用程序的外观。视图可以访问模型的读方法，但不能访问写方法。此外，它对控制器一无所知。当更改模型时，视图应得到通知。

（3）Controller（控制器）

控制器对用户的输入作出反应。它创建并设置模型。

3）MVC Model 2

Web 向软件开发人员提出了一些特有的挑战，最明显的就是客户机和服务器的无状态连接。这种无状态行为使得模型很难将更改通知视图。在Web上，为了发现对应用程序状态的修改，浏览器必须重新查询服务器。

另一个重大变化是实现视图所用的技术与实现模型或控制器的技术不同。当然，我们可以使用Java（或者PERL、C/C++或别的语言）代码生成HTML。这种方法有几个缺点：

1. Java程序员应该开发服务，而不是HTML。
2. 更改布局时需要更改代码。
3. 服务的用户应该能够创建网页来满足它们的特定需要。
4. 网页设计人员不能直接参与网页开发。
5. 嵌在代码中的HTML很难看。

对于Web，需要修改标准的MVC形式。图1显示了MVC的Web改写版，通常也称为MVC Model2或MVC2。

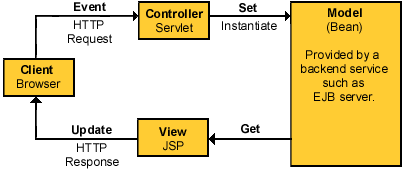


图 1. MVC Model 2

4)Struts，MVC 2 的一种实现

Struts是一组相互协作的类、servlet 和 JSP 标记，它们组成一个可重用的 MVC 2 设计。这个定义表示 Struts 是一个框架，而不是一个库，但 Struts 也包含了丰富的标记库和独立于该框架工作的实用程序类。图 2 显示了 Struts 的一个概览。

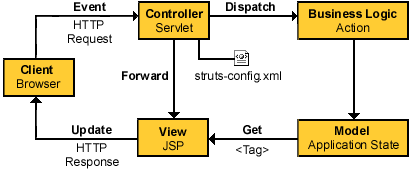


图 2. Struts 概览

5)Struts 概览

(1)Client browser（客户浏览器）

来自客户浏览器的每个 HTTP 请求创建一个事件。Web 容器将用一个 HTTP 响应作出响应。

(2)Controller（控制器）

控制器接收来自浏览器的请求，并决定将这个请求发往何处。就 Struts 而言，控制器是以 servlet 实现的一个命令设计模式。 struts-config.xml 文件配置控制器。

(3)业务逻辑

业务逻辑更新模型的状态，并帮助控制应用程序的流程。就 Struts 而言，这是通过作为实际业务逻辑“瘦”包装的 Action 类完成的。

(4)Model（模型）的状态

模型表示应用程序的状态。业务对象更新应用程序的状态。ActionForm bean 在会话级或请求级表示模型的状态，而不是在持久级。JSP 文件使用 JSP 标记读取来自 ActionForm bean 的信息。

(5)View（视图）

视图就是一个 JSP 文件。其中没有流程逻辑，没有业务逻辑，也没有模型信息 -- 只有标记。标记是使 Struts 有别于其他框架（如 Velocity）的因素之一。

6)详细分析 Struts

图 3 显示的是 org.apache.struts.action 包的一个最简 UML 图。图 3 显示了 ActionServlet (Controller)、 ActionForm (Form State) 和 Action (Model Wrapper) 之间的最简关系。

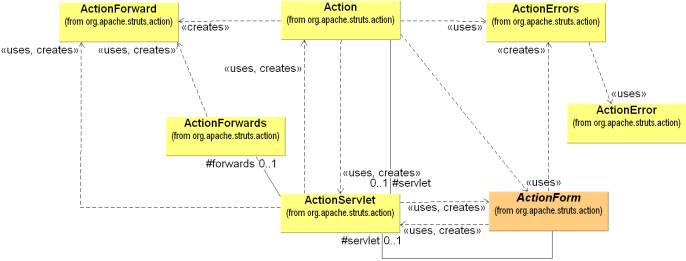


图 3. Command (ActionServlet) 与 Model (Action & ActionForm) 之间的关系的 UML 图

(1)ActionServlet 类

您还记得函数映射的日子吗？在那时，您会将某些输入事件映射到一个函数指针上。如果您对此比较熟悉，您会将配置信息放入一个文件，并在运行时加载这个文件。函数指针数组曾经是用 C 语言进行结构化编程的很好方法。

现在好多了，我们有了 Java 技术、XML、J2EE，等等。Struts 的控制器是将事件（事件通常是 HTTP post）映射到类的一个 servlet。正如您所料 -- 控制器使用配置文件以使您不必对这些值进行硬编码。时代变了，但方法依旧。

ActionServlet 是该 MVC 实现的 Command 部分，它是这一框架的核心。 ActionServlet (Command) 创建并使用 Action 、 ActionForm 和 ActionForward 。如前所述， struts-config.xml 文件配置该 Command。在创建 Web 项目时，您将扩展 Action 和 ActionForm 来解决特定的问题。文件 struts-config.xml 指示 ActionServlet 如何使用这些扩展的类。这种方法有几个优点：

1. 应用程序的整个逻辑流程都存储在一个分层的文本文件中。这使得人们更容易查看和理解它，尤其是对于大型应用程序而言。
2. 网页设计人员不必费力地阅读 Java 代码来理解应用程序的流程。
3. Java 开发人员也不必在更改流程以后重新编译代码。
4. 可以通过扩展 ActionServlet 来添加 Command 功能。

(2)ActionForm 类

ActionForm 维护 Web 应用程序的会话状态。 ActionForm 是一个抽象类，必须为每个输入表单模型创建该类的子类。当我说 输入表单模型 时,是指 ActionForm 表示的是由 HTML 表单设置或更新的一般意义上的数据。例如，您可能有一个由 HTML 表单设置的 UserActionForm 。Struts 框架将执行以下操作：

1. 检查 UserActionForm 是否存在；如果不存在，它将创建该类的一个实例。
2. Struts 将使用 HttpServletRequest 中相应的域设置 UserActionForm 的状态。没有太多讨厌的 request.getParameter() 调用。例如，Struts 框架将从请求流中提取 fname ，并调用 UserActionForm.setFname() 。
3. Struts 框架在将 UserActionForm 传递给业务包装 UserAction 之前将更新它的状态。
4. 在将它传递给 Action 类之前，Struts 还会对 UserActionForm 调用 validation() 方法进行表单状态验证。 注： 这并不总是明智之举。别的网页或业务可能使用 UserActionForm ，在这些地方，验证可能有所不同。在 UserAction 类中进行状态验证可能更好。
5. 可在会话级维护 UserActionForm 。
6. 注：
7. struts-config.xml 文件控制 HTML 表单请求与 ActionForm 之间的映射关系。
8. 可将多个请求映射到 UserActionForm 。
9. UserActionForm 可跨多页进行映射，以执行诸如向导之类的操作。

(3)Action 类

Action 类是业务逻辑的一个包装。 Action 类的用途是将 HttpServletRequest 转换为业务逻辑。要使用 Action ，请创建它的子类并覆盖 process() 方法。

ActionServlet (Command) 使用 perform() 方法将参数化的类传递给 ActionForm 。仍然没有太多讨厌的 request.getParameter() 调用。当事件进展到这一步时，输入表单数据（或 HTML 表单数据）已被从请求流中提取出来并转移到 ActionForm 类中。

注：扩展 Action 类时请注意简洁。 Action 类应该控制应用程序的流程，而不应该控制应用程序的逻辑。通过将业务逻辑放在单独的包或 EJB 中，我们就可以提供更大的灵活性和可重用性。

考虑 Action 类的另一种方式是 Adapter 设计模式。 Action 的用途是“将类的接口转换为客户机所需的另一个接口。Adapter 使类能够协同工作，如果没有 Adapter，则这些类会因为不兼容的接口而无法协同工作。”（摘自 Gof 所著的 Design Patterns - Elements of Reusable OO Software ）。本例中的客户机是 ActionServlet ，它对我们的具体业务类接口一无所知。因此，Struts 提供了它能够理解的一个业务接口，即 Action 。通过扩展 Action ，我们使得我们的业务接口与 Struts 业务接口保持兼容。（一个有趣的发现是， Action 是类而不是接口）。 Action 开始为一个接口，后来却变成了一个类。真是金无足赤。）

(4)Error 类

UML 图（图 3）还包括 ActionError 和 ActionErrors 。 ActionError 封装了单个错误消息。 ActionErrors 是 ActionError 类的容器，View 可以使用标记访问这些类。 ActionError 是 Struts 保持错误列表的方式。

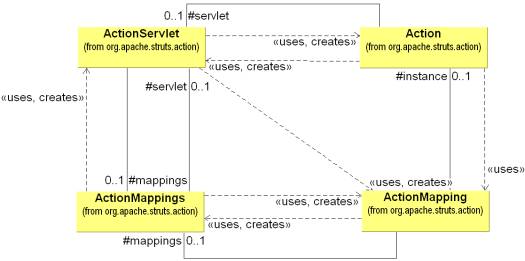


图 4. Command (ActionServlet) 与 Model (Action) 之间的关系的 UML 图

(5)ActionMapping 类

输入事件通常是在 HTTP 请求表单中发生的，servlet 容器将 HTTP 请求转换为 HttpServletRequest 。控制器查看输入事件并将请求分派给某个 Action 类。 struts-config.xml 确定 Controller 调用哪个 Action 类。 struts-config.xml 配置信息被转换为一组 ActionMapping ，而后者又被放入 ActionMappings 容器中。（您可能尚未注意到这一点，以 s结尾的类就是容器）

ActionMapping 包含有关特定事件如何映射到特定 Action 的信息。 ActionServlet (Command) 通过 perform() 方法将 ActionMapping 传递给 Action 类。这样就使 Action 可访问用于控制流程的信息。

(6)ActionMappings

ActionMappings 是 ActionMapping 对象的一个集合。

7)Struts 的优点

(1)JSP 标记机制的使用

标记特性从 JSP 文件获得可重用代码和抽象 Java 代码。这个特性能很好地集成到基于 JSP 的开发工具中，这些工具允许用标记编写代码。

(2)标记库

为什么要另发明一种轮子，或标记库呢？如果您在库中找不到您所要的标记，那就自己定义吧。此外，如果您正在学习 JSP 标记技术，则 Struts 为您提供了一个起点。

(3)开放源码

您可以获得开放源码的全部优点，比如可以查看代码并让使用库的每个人检查代码。许多人都可以进行很好的代码检查。

(4)MVC 实现样例

如果您希望创建您自己的 MVC 实现，则 Struts 可增加您的见识。

(5)管理问题空间

分治是解决问题并使问题可管理的极好方法。当然，这是一把双刃剑。问题越来越复杂，并且需要越来越多的管理。

8)Struts 的缺点

(1)仍处于发展初期

Struts 开发仍处于初级阶段。他们正在向着发行版本 1.0 而努力，但与任何 1.0 版本一样，它不可能尽善尽美。

(2)仍在变化中

这个框架仍在快速变化。Struts 1.0 与 Struts 0.5 相比变化极大。为了避免使用不赞成使用的方法，您可能隔一天就需要下载最新的 Struts。在过去的 6 个月中，我目睹 Struts 库从 90K 增大到 270K 以上。由于 Struts 中的变化，我不得不数次修改我的示例，但我不保证我的示例能与您下载的 Struts 协同工作。

(3)正确的抽象级别

Struts 是否提供了正确的抽象级别？对于网页设计人员而言，什么是正确的抽象级别呢？这是一个用 $64K 的文字才能解释清楚的问题。在开发网页的过程中，我们是否应该让网页设计人员访问 Java 代码？某些框架（如 Velocity）说不应该，但它提供了另一种 Web 开发语言让我们学习。在 UI 开发中限制访问 Java 有一定的合理性。最重要的是，如果让网页设计人员使用一点 Java，他将使用大量的 Java。在 Microsoft ASP 的开发中，我总是看到这样的情况。在 ASP 开发中，您应该创建 COM 对象，然后编写少量的 ASP 脚本将这些 COM 对象联系起来。但是，ASP 开发人员会疯狂地使用 ASP 脚本。我会听到这样的话，“既然我可以用 VBScript 直接编写 COM 对象，为什么还要等 COM 开发人员来创建它呢？”通过使用标记库，Struts 有助于限制 JSP 文件中所需的 Java 代码的数量。Logic Tag 就是这样的一种库，它对有条件地生成输出进行管理，但这并不能阻止 UI 开发人员对 Java 代码的狂热。无论您决定使用哪种类型的框架，您都应该了解您要在其中部署和维护该框架的环境。当然，这项任务真是说起来容易做起来难。

(14)有限的适用范围

Struts 是一种基于 Web 的 MVC 解决方案，所以必须用 HTML、JSP 文件和 servlet 来实现它。

(5)J2EE 应用程序支持

Struts 需要支持 JSP 1.1 和 Servlet 2.2 规范的 servlet 容器。仅凭这一点远不能解决您的全部安装问题，除非使用 Tomcat 3.2。我用 Netscape iPlanet 6.0 安装这个库时遇到一大堆问题，按理说它是第一种符合 J2EE 的应用程序服务器。我建议您在遇到问题时访问 Struts 用户邮件列表的归档资料。

(6)复杂性

在将问题分为几个部分的同时也引入了复杂性。毫无疑问，要理解 Struts 必须接受一定的培训。随着变化的不断加入，这有时会令人很沮丧。欢迎访问本网站。

9)Struts 的前景

在这个软件开发的新时代，一切都变得很快。在不到 5 年的时间内，我已经目睹了从 cgi/perl 到 ISAPI/NSAPI、再到使用 VB 的 ASP、一直到现在的 Java 和 J2EE 的变迁。Sun 正在尽力将新的变化反映到 JSP/servlet 体系结构中，正如他们对 Java 语言和 API 所作的更改一样。您可以从 Sun 的网站获得新的 JSP 1.2 和 Servlet 2.3 规范的草案。此外，一个标准 JSP 标记库即将出现。

附录C 软件使用说明书

本高校网上选课系统，根据不同用户的权限设置，对不同用户的使用方法进行说明。

1. 系统管理员操作

系统管理员的登录名为admin，初始密码为111111，可在其主界面实现各个学院教务管理员的权限授予。

1. 教务管理员操作

教务管理员的登录名为系统管理员授予的用户名，初始密码为111111，可在其主界面实现对任课教师及学生用户的管理操作，可对课程信息进行管理操作，同时还可实现对成绩及选课信息的查询。

1. 任课教师操作

任课教师的登录名为教务管理员授予的工号，初始密码为111111，可在其主界面实现个人信息维护、课程信息查询及成绩管理。

1. 学生用户操作

学生用户的登录名为教务管理员授予的学号，初始密码为111111，可在其主界面个人信息维护、选课管理、选课记录查询及成绩查询。

1. 通用操作

每一位用户均可在系统中的修改密码界面对用户密码进行修改。

附录D 主要源代码

1. 添加课程信息主要代码

public void kechengadd2() throws IOException {

HttpServletRequest request = ServletActionContext.*getRequest*();

HttpServletResponse response = ServletActionContext.*getResponse*();

response.setCharacterEncoding("gbk");response.setContentType("text/html; charset=gbk");

String didian = request.getParameter("didian");

String jianjie = request.getParameter("jianjie");

String kname = request.getParameter("kname");

String tid = request.getParameter("tid");

String week = request.getParameter("week");

String jie = request.getParameter("jie");

String xuefen = request.getParameter("xuefen");

String renshu = request.getParameter("renshu");

Kecheng bean = kechengDao.selectBean(" where week='"+week+"' and jie='"+jie+"' and didian='"+didian+"' ");

if(bean!=null){

response

.getWriter()

.print(

"<script language=javascript>alert('操作失败，该地点在该时间点有课程');window.location.href='method!kechenglist';</script>");

return ;}

bean = new Kecheng();

bean.setWeek(week);

bean.setJie(jie);

bean.setDidian(didian);

bean.setJianjie(jianjie);

bean.setKname(kname);

bean.setTeacher(teacherDao.selectBean(" where id= "+tid));

bean.setXuefen(xuefen);

bean.setRenshu(Integer.*parseInt*(renshu));

kechengDao.insertBean(bean);

response

.getWriter()

.print(

"<script language=javascript>alert('操作成功');window.location.href='method!kechenglist';</script>"); }

1. 选课功能主要代码

public void kechengupdate6() throws IOException {

HttpServletRequest request = ServletActionContext.*getRequest*();

HttpServletResponse response = ServletActionContext.*getResponse*();

response.setCharacterEncoding("gbk");response.setContentType("text/html; charset=gbk");

Kecheng kecheng = kechengDao.selectBean(" where id= "

+ request.getParameter("id"));

if(kecheng.getRenshu()-kecheng.getBaoming()<=0){

response

.getWriter()

.print(

"<script language=javascript>alert('操作失败，该课程已经选满！');window.location.href='method!kechenglist2';</script>");

return; }

HttpSession session = request.getSession();

User user = (User)session.getAttribute("user");

Xuanke bean = xuankeDao.selectBean(" where deletestatus=0 and kecheng.id= "+kecheng.getId()+" and user.id= "+user.getId());

if(bean!=null){

response

.getWriter()

.print(

"<script language=javascript>alert('操作失败，该课程已被该学生选中，选课失败');window.location.href='method!kechenglist2';</script>");

}else{

bean = new Xuanke();

bean.setKecheng(kecheng);

bean.setUser(user);

bean.setShijian(Util.*getTime*());

xuankeDao.insertBean(bean);

kecheng.setBaoming(kecheng.getBaoming()+1);

kechengDao.updateBean(kecheng);

response

.getWriter()

.print(

"<script language=javascript>alert('操作成功');window.location.href='method!xuankelist';</script>");

}

}