

宿舍管理系统

【原文对照报告-大学生版】

报告编号: 8418c8f0f7912dba

检测时间: 2019-04-22 18:17:38

检测字数: 14,768字

作者名称: 祁唱唱

所属单位: 重庆人文科技学院

检测范围:

- | | | |
|------------------|-----------------|-------------------|
| ◎ 中文科技期刊论文全文数据库 | ◎ 中文主要报纸全文数据库 | ◎ 中国专利特色数据库 |
| ◎ 博士/硕士学位论文全文数据库 | ◎ 中国主要会议论文特色数据库 | ◎ 港澳台文献资源 |
| ◎ 外文特色文献数据全库 | ◎ 维普优先出版论文全文数据库 | ◎ 互联网数据资源/互联网文档资源 |
| ◎ 高校自建资源库 | ◎ 图书资源 | ◎ 古籍文献资源 |
| ◎ 个人自建资源库 | ◎ 年鉴资源 | ◎ IPUB原创作品 |

时间范围: 1989-01-01至2019-04-22

检测结论:

全文总相似比	=	复写率	+	他引率	+	自引率	+	专业术语
24.21%		23.70%		0.00%		0.00%		0.51%

其他指标:

自写率: 75.79%

专业用语: 0.51%

高频词: 管理, 信息, 界面, 系统, 宿舍

典型相似性: 无

指标说明:

复写率: 相似或疑似重复内容占全文的比重

他引率: 引用他人的部分占全文的比重, 请正确标注引用

自引率: 引用自己已发表部分占全文的比重, 请正确标注引用

自写率: 原创内容占全文的比重

专业用语: 公式定理、法律条文、行业用语等占全文的比重

典型相似性: 相似或疑似重复内容占互联网资源库的比重, 超过60%可以访问

总相似片段: 254

期刊: 15 博硕: 145 外文: 4 综合: 6 自建库: 0 互联网: 84

颜色标注说明:

- 自写片段
- 复写片段 (相似或疑似重复)
- 引用片段
- 引用片段(自引)
- 专业用语 (公式定理、法律条文、行业用语等)

毕业设计 (论文)

系 别:

专 业:

班 号:

学 生 姓 名:

学 生 学 号:

设计 (论文) 题目: 基于B/S架构的高校宿舍信息管理系统的设计与实现

指 导 教 师:

设 计 地 点:

起 迄 日 期:

目 录

摘 要 1

Abstract 2

第一章 绪论 3

1.1 研究 背景 3

1.2 研究现状 3

1.3 课题目标 4

第二章 系统开发环境 5

2.1 开发工具 5

2.1.1 MyEclipse 5

2.1.2 Tomcat 5

2.1.3 MYSQL 数据库 5

2.2 JSP 技术 6

2.3 B/S 结构 7

2.4 JavaScript 7

2.5 软硬件需求 8

第三章 需求分析 9

3.1 可行性分析 9

3.1.1 操作的可行性 9

3.1.2 发展 可行性 9

3.1.3 技术可行性	9
3.2 流程分析	10
3.3 性能需求分析	11
3.3.1 系统的安全性	11
3.3.2 数据的完整性	12
3.4 功能需求分析	12
3.5 用例分析	13
3.5.1 管理员用例图	13
3.5.2 学生用例图	14
第四章 数据库设计	15
4.1 数据库设计原则	15
4.2 数据库 E-R 图设计	15
4.3 数据库设计表	18
第五章 系统的实现	20
5.1 系统登录界面	20
5.2 管理员主界面	20
5.2.1 基本信息管理界面	21
5.2.2 公告管理界面	21
5.2.3 学生管理界面	22
5.2.4 班级信息管理界面	22
5.2.5 宿舍楼管理界面	22
5.2.6 宿舍管理界面	23
5.2.7 离校学生信息界面	23
5.2.8 报修管理界面	23
5.2.9 水电费管理界面	24
5.2.10 来访管理界面	24
5.3 学生管理主界面	24
5.3.1 基本信息管理界面	25
5.3.2 通知公告管理界面	26
5.3.3 保修管理界面	26
5.3.4 水电费管理界面	26
第六章 系统测试与维护	28
6.1 功能测试	28
6.2 可用性测试	28
6.3 性能测试	29
6.4 测试结果分析	29
6.5 系统维护	29
结束语	30
致 谢	31

参考文献 32

摘 要

随着全国各大高校的扩招，和大学的扩建，让更多的考生能进入大学获得学习的机会。在开放的国家政策下，我国公民也获得了更多的教育机会。然而在大学扩招的同时，也带来一些校园方面的维护问题。在高校随着学生的增多，以为着数据量的增加。数据种类繁多，它使宿舍管理系统的运作和管理变得尤为重要。因此，基于现代互联网多媒体信息技术构建宿舍管理系统对学校宿舍的管理具有重要意义。

B/S宿舍管理系统是目前各大高校管理的重要部分，本宿舍管理系统以实际运用，为开发背景，使用B/S架构，采用JSP、Spring、SpringMVC、Mybatis等技术。数据库使用了MYSQL，保证安全性和稳定。该系统采用MVC的设计模式。根据管理员和学生的需求，让模型数据和控制器层和视图的表现分离。由于使用的是WEB，所以非常方便操作人员，易于维护。同时信息数据可以高效和安全的存储在数据库中。

关键字：MVC；B/S宿舍管理；MYSQL数据库；SSM框架

Abstract

With the expansion of colleges and universities across the country, and the expansion of the university, more candidates can enter the university to gain the opportunity to learn. Under the open national policy, our citizens have also gained more educational opportunities. However, while the university is expanding, it also brings some maintenance issues on campus. With the increase of students in colleges and universities, I thought that the amount of data increased. With a wide variety of data, it makes the operation and management of the dormitory management system particularly important. Therefore, building a dormitory management system based on modern Internet multimedia information technology is of great significance to the management of school dormitory. The use of the dormitory management system is an effective way to improve management efficiency. The B/S dormitory management system is an important part of the management of major universities. The dormitory management system uses the B/S architecture and JSP, Spring, SpringMVC, Mybatis and other technologies for practical development. The database uses MYSQL to ensure security and stability. The dormitory management system implements the basic operations of information query, addition and management, and adopts a modular design. According to the needs of administrators and students, the system adopts the software component development model to separate the system structure, separate the business and implementation, and separate the logic and data. Thanks to the WEB, it is very convenient for the operator and easy to maintain. At the same time, information data can be stored in the database efficiently and securely.

Keyword: JSP technology; dormitory management system; MYSQL database; B/S architecture; SSM framework

第一章 绪论

1.1 研究背景

在信息化的当代。本系统从解决校园管理人员的负担为设想出发，随着全国高校住宿学生的增多。提高工作人员的效率，开发出的这套B/S宿舍管理系统解放了管理人员，让管理人员有更多的时间去做想做的事情。

高校自扩展后，校园宿舍管理人员的工作越来越繁杂，如学生信息、宿舍信息等种种问题，手工管理校园宿舍容易出错，数据繁多，因此需要对宿舍管理的管理模式进行及时改善，来提高管理员的工作效率，从而也可避免因系统的不完善而导致管理漏洞，使得宿舍管理系统的运行和管理就显得尤为重要。

1.2 研究现状

根据最新调查得知，在此之前对校园宿舍信息的管理通常是基于文本，表格等纸介的手工处理，学校宿舍信息，学生信息等情况的统计和核实等通常采用人工检查与核实，以及对

相关宿舍查询信息等通过人工计算或手抄等方式进行，信息量大，但在数据处理方面存在工作量大且易出错，其次还存在数据繁多与数据丢失等现象出现，而且还不方便查找，在总体上存在缺乏系统与规范的信息管理系统现象，为了提高工作效率我们急需开发出这套宿舍管理系统。

当前社会信息的流通和交换得特别重要。因此，管理宿舍信息需要使用计算机已经成为必然。假设能有一套非常便捷的系统，那么就能让现有工作更加效率、解放了重复劳动，让管理员花更少的经理在重复劳动中。

1.3 课题目标

首先要对本B/S架构宿舍管理系统的相关信息有整体把握，了解校园宿舍管理的现状以及学生的基本信息。

其次要实地调查了解校园宿舍管理工作的业务流程，处理相关的数据并记录下来。

要研究开发系统所使用的数据库管理系统MYSQL，设计好所使用的数据库表；

最后要研究系统各个功能模块的详细设计以及掌握所要使用的开发工具，从而完成系统开发。

第二章 系统开发环境

2.1 开发工具

IDE:MyEclipse，服务器软件：Tomcat，数据库软件：Mysql

2.1.1 MyEclipse

MyEclipse IDE，是软件开发人员最喜爱的IDE之一，其本身的高度集成，界面简洁，功能强悍。通过自带的Tomcat可以快速实现服务器和后端的程序整合，加快开发的速度。MyEclipse的功能十分强大，被企业和个人开发者亲睐。

2.1.2 Tomcat

由开源社区不断对版本的升级和维护，几乎是所有初学者必使用的软件。在并发量不是特别高的情况下，也算是较适合该项目。

2.1.3 Mysql 数据库

没有数据库的软件大多是一般的桌面产品，不能适应大批量数据。而要解决了数据量大时的数据存储问题，则须借助数据库，Mysql完全可以应付本项目的数据库功能需求。对方便维护来说也是个不错的选择。

2.2 JSP和Servlet技术

使用Swing组件非常的棘手，而几乎没有几个GUI应用能被数百人知道，甚至数万人，使用JSP而言，完全不用对付独立应用。可以仅仅通过浏览器就能把应用交付给用户，极大限度的解决了繁琐的问题和程序。

用户通过web浏览器请求想要的资源，Web服务器在得到请求。

JSP负责显示数据，servlet作为控制器正好符合MVC设计思想。

JSP比普通静态网页好的一个地方就是，可以嵌入逻辑，在网页基础上定制自己想要的功能。因为，原先的普通的静态网页，怎么说呢，感觉就好像是上古时期的东西了，有点跟不上现在这个时代了。现代网页对实时性和交互性比以往更强烈。谁能在做出更漂亮或者交互性更友好的页面，谁就能在竞争中占有一席之地。况且，有些页面的内容就应该按照用户意愿动态生成。而JSP和servlet刚好能满足这样的需求。

2.3 Spring框架

Spring 框架能许多事情，能为WEB应用提供很多功能，同时为 J2EE 应用程序开发提供集成的环境。

Spring的理念就是让开发者专注于开发新的东西，而不是重复造轮子。我认为这是该框架非常优秀的地方之一。另外，spring框架不排斥其它框架。像有些框架如果使用了它，在后期升级维护上如果遇到困难想弃用，那么前面已经开发的那些东西就可能白白荒废了，因为这个框架可能不兼容其它的框架。但spring不会，因为它的主题就是不去让开发者做一些无聊的重复事情。

使 Java 方法在数据库事务中执行，而无需处理事务 Api。

使本地 Java 方法成为 HTTP 终结点，而无需处理 Servlet API。

使本地 Java 方法成为管理操作，而无需处理 JMX API。

依赖注入与控制反转：

普通的java应用程序中的对象彼此具有依赖关系。而这种依赖通常是很强的，专业术语是强耦合。比方说：在spring之前，人们创建一个对象通常是通过new关键字来的，这么说可能看不出什么问题来。但是程序一旦大起来，甚至不用太大，比方说一个工程有100个类（这通常来说是很正常的事情，甚至可能还不止呢），这个时候想想，在一个类里面显式的new了很多其他类的对象，到最后你可能根本不知道某些类到底是哪里来的。这真的是给二次开发和维护带了很多的噩梦。

而现在一切都可以解决了！借助于spring的控制反转，开发人员终于可以获得更多的假期时间了。

现在，我们创建对象不用在显式的去new了，我们可以把想创建对象的任务交给spring容器来实现。而容器就像是一个大的加工厂，加工出我们想要的对象。

如果这还不能说服开发人员使用spring的话。那么上传环境的实践已经告诉我们spring的成功。愈来愈多的企业正在转向轻量，灵巧的spring。

Spring 框架反转控制（IoC）组件通过提供一种形式化的方法，将不同的组件组合到一个完全工作的应用程序中，可以随时可用、可维护。

面向切面的编程：

为什么要使用aop呢？主要就是为了方便。因为他把自己做的事情都让程序去做了。用了AOP能让你少写很多代码，这个理由想必是足够充分了。

另外就是为了更清晰的逻辑，可以让你的业务逻辑去关注自己本身的业务，而不去想一些其他的事情。

在切面中有一些常见的概念。

切面是通知和切点的组合。而切点一般是需要被增强的业务。对应程序中就是业务方法（service）。通知就是我们想植入进去的额外功能。

目标：引入了文章中提到的目标类，即要通知的对象，即真实的业务逻辑，可以在不知情的情况下编织。其次，关注业务本身的逻辑。

以上是一些关于切面的概念，也是AOP可以大显神通的地方。

IOC 容器

容器创建对象，将它们连接在一起，配置它们，并管理从创建到销毁的整个生命周期。

Spring用到的一些设计模式：

1. 工厂方法模式：

为了将对象的创建和使用相分离，采用工厂模式，即应用程序将对象的创建及初始化职责交给工厂对象，可以假设汽车对象是一个复杂的对象。汽车包括轮胎，引擎，玻璃，车窗等等。那么当每次创建一个汽车对象，都要做这些重复的零件创建，最后才能生成一个完成的汽车对象。这样的工作量是非常大的，一旦程序中需要大量的使用到汽车对象，此时就显得特别麻烦。而依赖工厂设计模式，就。我们只关心结果，将过程隐藏了。事实上开发人员只在意得到结果，有些过程可以合适的省略掉，比如重复性的工作。

一般情况下，应用程序有自己的工厂对象来创建bean，如果将应用程序自己的工厂对象交给Spring管理，那么Spring管理的就不是普通的bean，而是工厂Bean。

2. 单例模式：

单例就像是给类计划生育，保证应用的整个上下文只有一个对象，这样做的好处还有减少对内存的浪费。尤其是当我们频繁的使用某个类的对象时。这个时候好处就体现出来了。

3. 适配器模式：

当客户端调用适配器的方法时，将在适配器方法内调用适配器的方法。客户不是直接访问适配器，而是通过调用适配器方法来访问适配器。因此此模式可以使互不兼容的类能够合作愉快。

5. 包装器模式：

当某个抽象类的实现类的比较多，功能可能各不相同，而开发人员有想组合不同的功能，这个时候就要用到此模式了。使用此模式相

比之前，同样的功能组合，但是类的个数却能降低一个数量级。而组合本身就是灵活的。

6. 代理模式：

主要就是在已经完成的的功能的基础上，又可以在不改变原有已经编译好的类上添加新功能，spring中许多地方都有代理模式的影子，甚至声明式事务，都要借助于代理模式。而面向切面本身也是借助该模式实现的。

7. 观察者模式：

最常见的是就是订阅发布这样的例子了。举个例子，比如在报社那里办了个会员，那么每次一有新闻更新，报社都会把报纸送给我们。Spring中的监听器就是利用该模式的。

8. 策略模式：

是利用编程语言内置的多态特性，可以让父类的对象在运行时确定所指向的引用。

有许多注解就是利用该模式。

9. 模板方法：

顾名思义就是利用给好的模板填空就行了。Spring中的base-schema通知、拦截器的底层都离不开此模式。

2.3 Mybatis框架

使用mybatis 时不需要编写实现类,只需要写需要执行的 sql 命令。

Mapper映射文件

MyBatis 的真正力量在于映射语句。这就是魔法发生的地方。对于所有的他们的力量，映射器 XML 文件是相对简单的。当然，如果你比较他们的等效的 JDBC 代码，你会立即看到95% 的代码的节省。MyBatis 被修造了专注于 SQL，并尽最大努力不让你的方式。

Mybatis提供了强大的动态sql功能：

传统的使用JDBC的方法真的很麻烦，不小心的话还会陷入到坑中。而动态的sql语句可以根据不同的条件需要执行不同的 SQL 命令。MyBatis 中动态 SQL 在 mapper.xml 中添加逻辑判断等。和 JDBC相比来说，，Mbatis sql映射更好。

2.3 SpringMVC框架

很多应用程序的问题在于处理业务数据的对象和显示业务数据的视图之间存在紧密耦合，通常，更新业务对象的命令都是从视图本身发起的，使视图对任何业务对象更改都有高度敏感性。而且，当多个视图依赖于同一个业务对象时是没有灵活性的。

框架的目的就是帮助我们简化开发，SpringMVC也是要简化我们日常Web开发。

MVC框架已经出现了很多了，早前市场上流行的是Struts，再到后来的Struts2。和这些优秀的MVC框架不同的是，springmvc框架吸取了前辈框架的缺点，由于它是基于方法的级别的MVC框架，使得它能独树一帜。

MVC设计模式的任务是将包含业务数据的模块与显示模块的视图解耦。这是怎样发生的？在模型和视图之间引入重定向层可以解决问题。此重定向层是控制器，控制器将接收请求，执行更新模型的操作，然后通知视图关于模型更改的消息。

SpringMVC 中重要组件

1) DispatcherServlet：前端控制器,接收所有请求(如果配置/不包

含 jsp)

但是要注意静态资源的放行。

```
<mvc:annotation-driven></mvc:annotation-driven>
```

```
<!放行静态资源 -->
```

```
<mvc:resources location="/js/" mapping="/js/**"></mvc:resources>
```

```
<mvc:resources location="/css/" mapping="/css/**"></mvc:resources>
```

```
<mvc:resources location="/images/"></mvc:resources>
```


2) HandlerMapping: 解析请求格式的. 判断希望要执行哪个具体的方法.

3) HandlerAdapter: 负责调调用具体的方法.

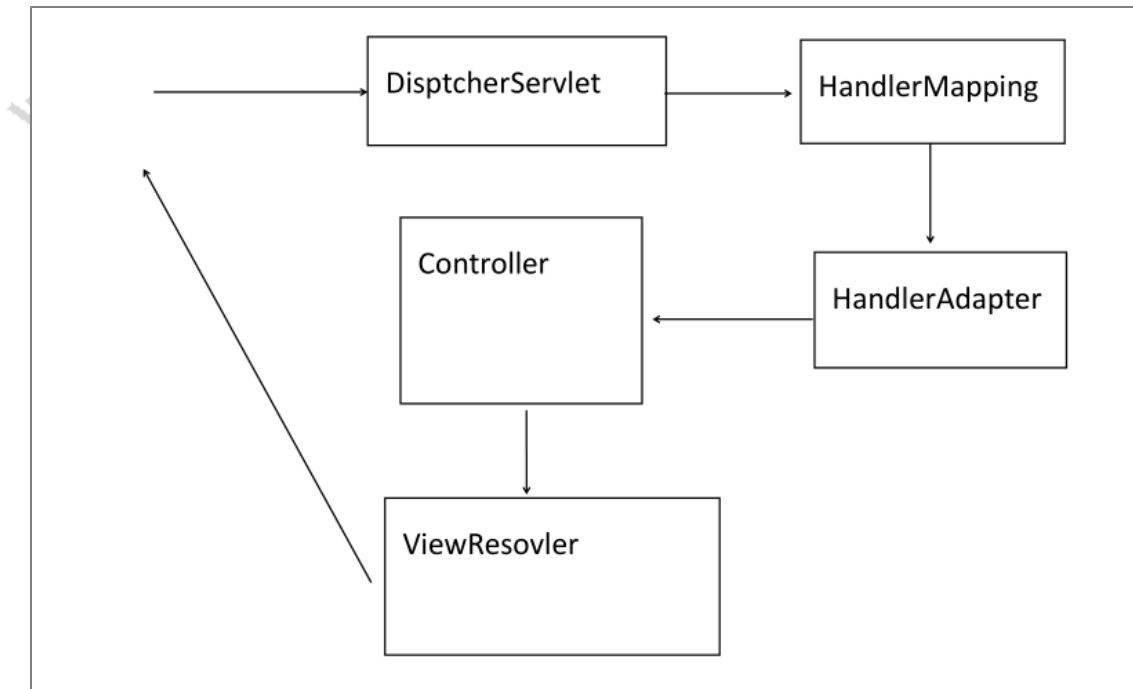
4) ViewResovler:视图解析器. 解析结果, 准备跳转到具体的物理视图

5) springmvc的拦截器是一种跟过滤器比较像的技术. 发送请求时被拦截器拦截, 在控制器的前后添加额外功能.

跟 AOP 区分开. AOP 在特定方法前后扩充(对 ServiceImpl)

拦截器, 请求的拦截. 针对点是控制器方法. (对 Controller)

SpringMVC 运行原理图



2.3 LOG4J日志支持

Log4j是Apache下的一款开源的日志框架, 能够满足我们在项目中对于日志记录的需求. 一般来讲, 在项目中, 我们会结合slf4j和log4j一起使用. Log4j提供了简单的API调用, 强大的日志格式定义以及灵活的扩展性. 我们可以自己定义Appender来满足我们对于日志输出的需求.

早就听说有日志级别这么回事, 日志级别到底是个什么? 刚才提到了两个需求, 一个是保存日志到不同的地方, 另一个是开发和生产环境打印不同的日志, 那么如何在不同的环境打印不同的日志, 就是由日志级别来实现的.

日志分了好几个等级, 是根据不同需求可以控制日志的输出. 比如, 你要监控一下用户提交上来的注册信息, 我们认为这是在开发环境才需要的, 因此可以使用DEBUG级别记录. 再比如一个支付场景, 系统抛出了一个未知的异常, 这肯定是一个非常严重的问题了, 那么可以用FATAL或者ERROR来记录.

了解清楚等级划分之后, 再来控制生成环境和开发环境输出的信息不同就很简单了, 我可以配置一个最低的日志输出级别, 在开发环境我设置为DEBUG, 也就是所有的日志信息都能输出. 在生产环境我可以设置为WARN, 也就是只有WARN及以上级别的日志才会被输出, 这样是不是就可以在不同的环境控制不同的日志输出了.

对于开发者, 学习如何记录各等级的日志非常简单, 日志框架一般都提供了非常人性化的API. 比如记录debug信息就是 logger.debug(String msg) 记录 info信息就是 logger.info(String msg) 等等, logger是实例化的日志对象.

2.3 B/S架构

B/S架构下的应用，所有的配置和维护都在服务端完成，而在前端页面只要展示功能就行了。用户无须关心应用的升级和维护。

2.4 JavaScript

现代大多数浏览器的默认的脚本语就是JavaScript。

JavaScript 非常容易学。JavaScript 是互联网上最流行的脚本语言，这门语言几乎通用于现代所有的web浏览器。

JavaScript是一种脚本型的解释语言，且是一种部分变量类型的弱类型语言。

若类型语言的好处尤为明显。开发人员不需要生命变量的类型，而只是用一个简单的var关键字就可以声明一个变量。在方法的参数中，只要写上名字就行了，也不用写类型。但弱类型若是处理不好，也有可能引发一系列安全的问题。但这些都阻止JavaScript变成流行的脚本语言。毕竟实用才是最主要的。

第三章 需求分析

3.1可行性分析

可行性一般是对现状和发展作出分析，避免类似于盲目去做或者盲目投资，最后没有获得任何回报。

3.1.1操作的可行性

计算机专业的本人，对于学校现在宿舍管理方面的缺陷更是深有体会。

每当开学的时候，新生进入校园，宿舍的分配就是一大问题。在平时生活中比如宿舍的东西需要换休，先需要在前台纸质登记，有时候过了很长时间都得不到反馈总的来说，现有操作还是有地方需要改进。

3.1.2 发展可行性

学生越来越多，管理宿舍数据信息量大，为了建立更完善的制度，需要对学生、宿舍信息进行记录准确，为了更好地管理学生，贴近学生。重要的是确保房里方法科学准确，这也是提高学校发展和竞争力的长远因素。因此，为了学校的创新和发展，新的宿舍管理系统也是必须的。

3.1.3 技术可行性

开发一套基于B/S的宿舍管理系统。前端显示数据用JSP，后台逻辑代码的实现用SSM框架。必须具有功能完整、使用便捷的特点。为了建立和维护数据库，建立一个数据完整性强、数据安全好、数据稳定性高的数据库是非常有必要的。

以前，软件的开发大多采用客户/服务器模式，即C/S 模式。例如，在 Java 中，可以通过套接字实现客户端服务器通讯的体系结构。在此模式下，业务逻辑与客户端程序混合，这不可避免地导致问题：服务端和客户端需要分别开发出一套系统，给维护和升级存在困难。

由于 C/S体系存在的问题，人们逐渐接受基于浏览器服务器体系的系统，即 B/S模式。在 B/S模式下，用大多数人都会使用的浏览器来替换以前笨重的客户端系统程序是主要核心。

B/S对C/S模式进行了一些更改，它却带来了巨大优势：因为现在客户端变成了浏览器。应用程序下载、升级和维护的工作，只用在服务端配置。服务器端集中了所有的配置工作，安全问题被极大的降低和隐藏了。

所以，B/S模式是许多开发者主要的选择模式。

计算机专业的本人来说，熟悉MYSQL 数据库并掌握 Java 编程语言。它还算是个人的能力范围内。

因此，技术上分析到开发该系统是可行的。

3.2流程分析

通过对校园宿舍管理的实际情况的调查，分析出了该 JSP 宿舍管理系统需要的业务流程。在操作系统之前，需要对欲登录用户做校验，只有输入正确帐户密码才能登录。登录模块主要验证数据合法性。流程图如图3-1所示

图3-1 用户登录流程图

基于B/S的宿舍管理系统开发流程图，如图3-2所示

图3-2开发流程图

3.3 性能需求分析

3.3.1 系统的安全性

宿舍管理为控制管理权限，具体要求如下：

1. 用户依据正确用户名和密码登录系统。未授权用户不能以任何方式登录，以保证数据安全性。
2. 对不同类别用户的权限进行设定，登陆系统后，不同类型用户不能越权限操作，只有超级管理员才有权限访问所有资源。

3.3.2 数据的完整性

- 1、对登录信息进行信息记录，依靠log4j日志框架的支持。
- 2、数据间的关系要保持准确。

3.4 功能需求分析

该宿舍管理系统主要分为两个类别的用户，如下：

管理员：管理员具有所有功能的权限，如基本信息、学生、公告，班级信息、宿舍楼、离校学生、宿舍、保修、水电费、来访管理。基础信息管理包括管理员信息管理和修改密码。水电费管理包括水电费管理和水电费统计等。

学生：学生登录后主要功能有：基本信息管理，通知公告，保修管理，水电费管理；基本信息管理包括修改密码和个人资料等。

根据上述分析，得出JSP宿舍管理系统的系统功能结构图，如图3-3所示：

图3-3系统功能结构图

3.5 用例分析

宿舍关系系统有两种类型的用户。分别是学生和超级管理员。其用例图如下。

3.5.1 管理员用例图

管理员登录后主要功能有：基本信息、学生、公告，班级信息、宿舍楼、离校学生、宿舍、保修、水电费、来访管理。其用例图如图3-4所示

图3-4管理员用例图

3.5.2 学生用例图

学生登录后主要功能有：学生登录后主要功能有：自身基本信息的维护，查看通知公告，保修管理，水电费管理；其用例图如图3-5所示。

图3-5 学生用例图

第四章 数据库设计

数据库是为了实现一定目的按某种规则和方法组织起来的数据的集合。

数据库可以直观的理解为存放数据的仓库，只不过这个仓库是在计算机的大容量存储器上，而且数据必须按照一定的格式存放，因为它不仅需要存放，而且要便于查找，数据库是计算机信息系统的基础。信息的收集、整理、存储、检索、更新、加工等都依赖于数据库才能高效率的操作。

4.1 数据库设计注意事项

任何公司或者开发者，都要用好数据库（我们这里说的数据库指的是关系型数据库），由于数据库足够复杂，在计算机领域，它已经发展成了一个独立的方向。而作为一个开发者，能不能用好数据库，也成了衡量他是否有价值的指标。

数据库设计的注意问题：

1. 数据库的容量是它的瓶颈
2. 大量的读或写操作是它的瓶颈
3. 复杂的SQL语句是它的瓶颈，例如大量没有经过优化的SQL语句
4. 索引越来越大是它瓶颈。

4.2 数据库E-R图设计

概念模式的设计方法是在需求分析的基础上，用概念数据模型（例如E-R模型）来表示数据以及数据之间的联系，开发出能够解决用户信息要求和处理要求的数据库系统概念模式。

最常见的设计是E-R模型法。本系统的E-R图表现了系统中各个实体之间的联系，本系统的各实体E-R图如下图所示：

(1)管理员的信息E-R图如图4-1所示：

图4-1管理员信息实体图

管理员实体类的代码如下：

```
package com.model;

/**
 * TAdmin generated by MyEclipse Persistence Tools
 */
public class TAdmin implements java.io.Serializable {
    // Fields
    private Integer id;
    private String uname;
    private String upwd;

    // Constructors
    /** default constructor */
    public TAdmin() {
    }

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getUname() {
        return uname;
    }

    public void setUname(String uname) {
        this.uname = uname;
    }

    public String getUpwd() {
        return upwd;
    }

    public void setUpwd(String upwd) {
        this.upwd = upwd;
    }
}
```

(2) 班级信息E-R图如图4-2所示:

图4-2班级信息实体图

班级对应的实体类如下:

```
package com.model;

public class Banji
{
    public Banji() {
    }

    private Long id;
    private String name;
    private int rs;
    public Long getId() {
        return this.id;
    }
    public void setId(Long id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public int getRs() {
        return rs;
    }
    public void setRs(int rs) {
        this.rs = rs;
    }
}
```

(3) 报修信息实体E-R图如图4-3所示:

图4-3报修信息实体图

报修信息实体对应的类是: package com.model;

```
public class Baoxiu
{
    public Baoxiu() {
    }

    private Long id;
    private Long susheid;
```

```
private String bxinfo;
private String bxsj;
private String chuli;
private String clsj;
private String state;
private Long sid;
private Sushe sushe;
public Sushe getSushe() {
return sushe;
}
public void setSushe(Sushe sushe) {
this.sushe = sushe;
}
public Long getId(){
return this.id;
}
public void setId(Long id){
this.id = id;
}
public Long getSusheid(){
return this.susheid;
}
public void setSusheid(Long susheid){
this.susheid = susheid;
}
public String getBxinfo(){
return this.bxinfo;
}
public void setBxinfo(String bxinfo){
this.bxinfo = bxinfo;
}
public String getBxsj(){
return this.bxsj;
}
public void setBxsj(String bxsj){
this.bxsj = bxsj;
}
public String getChuli(){
return this.chuli;
```

```

}

public void setChuli(String chuli) {
    this.chuli = chuli;
}

public String getClsj() {
    return this.clsj;
}

public void setClsj(String clsj) {
    this.clsj = clsj;
}

public String getState() {
    return this.state;
}

public void setState(String state) {
    this.state = state;
}

public Long getSid() {
    return this.sid;
}

public void setSid(Long sid) {
    this.sid = sid;
}
}

```

(4) 公告实体E-R图如图4-4所示:

图4-4公告信息实体图

公告信息对应的类:

```

package com.model;

public class Gonggao {
    {
        public Gonggao() {
        }

        private Long id;
        private String title;
        private String contents;
        private String fbsj;
        private int offset;
        private int pagesize;
        public Long getId() {
            return this.id;
        }
    }
}

```

```

}

public void setId(Long id) {
    this.id = id;
}

public String getTitle() {
    return this.title;
}

public void setTitle(String title) {
    this.title = title;
}

public String getContents() {
    return this.contents;
}

public void setContents(String contents) {
    this.contents = contents;
}

public String getFbsj() {
    return this.fbsj;
}

public void setFbsj(String fbsj) {
    this.fbsj = fbsj;
}

public int getOffset() {
    return offset;
}

public void setOffset(int offset) {
    this.offset = offset;
}

public int getPagesize() {
    return pagesize;
}

public void setPagesize(int pagesize) {
    this.pagesize = pagesize;
}
}

```

(5) 来访实体E-R图如图4-5所示:

图4-5来访信息实体图

来访信息的类:

```
package com.model;
```



```
public class Laifang
{
public Laifang(){

private Long id;
private String name;
private String idno;
private String lfsj;
private String remarks;
public Long getId(){
return this.id;
}
public void setId(Long id){
this.id = id;
}
public String getName(){
return this.name;
}
public void setName(String name){
this.name = name;
}
public String getIdno(){
return this.idno;
}
public void setIdno(String idno){
this.idno = idno;
}
public String getLfsj(){
return this.lfsj;
}
public void setLfsj(String lfsj){
this.lfsj = lfsj;
}
public String getRemarks(){
return this.remarks;
}
public void setRemarks(String remarks){
this.remarks = remarks;
}
```

}

(6) 学生信息实体E-R图如图4-6所示:

图4-6学生信息实体图

学生信息实体类:

```
package com.model;

public class Student
{
    public Student() {

    }

    private Long id;
    private String sno;
    private String name;
    private String sex;
    private String rxnf;
    private Long banji;
    private String yuanxi;
    private String state;
    public String getState() {
        return state;
    }
    public void setState(String state) {
        this.state = state;
    }
    public Long getBanji() {
        return banji;
    }
    public void setBanji(Long banji) {
        this.banji = banji;
    }
    private Sushe sushe;
    private String mtel;
    private String ftel;
    private Long susheid;
    private String upwd;
    private String wfp;
    public Sushe getSushe() {
        return sushe;
    }
}
```

```
public void setSushe(Sushe sushe) {  
    this.sushe = sushe;  
}  
  
public Long getId() {  
    return this.id;  
}  
  
public void setId(Long id) {  
    this.id = id;  
}  
  
public String getSno() {  
    return this.sno;  
}  
  
public void setSno(String sno) {  
    this.sno = sno;  
}  
  
public String getName() {  
    return this.name;  
}  
  
public void setName(String name) {  
    this.name = name;  
}  
  
public String getSex() {  
    return this.sex;  
}  
  
public void setSex(String sex) {  
    this.sex = sex;  
}  
  
public String getRxnf() {  
    return rxnf;  
}  
  
public void setRxnf(String rxnf) {  
    this.rxnf = rxnf;  
}  
  
public String getYuanxi() {  
    return yuanxi;  
}  
  
public void setYuanxi(String yuanxi) {  
    this.yuanxi = yuanxi;  
}
```

```
public String getMtel() {
    return mtel;
}

public void setMtel(String mtel) {
    this.mtel = mtel;
}

public String getFtel() {
    return ftel;
}

public void setFtel(String ftel) {
    this.ftel = ftel;
}

public Long getSusheid() {
    return susheid;
}

public void setSusheid(Long susheid) {
    this.susheid = susheid;
}

public String getUpwd() {
    return upwd;
}

public void setUpwd(String upwd) {
    this.upwd = upwd;
}

public String getWfp() {
    return wfp;
}

public void setWfp(String wfp) {
    this.wfp = wfp;
}
}
```

4.3数据库设计表

本宿舍管理系统需要后台数据库，采用MYSQL数据库进行数据储存，下面介绍数据库中的各个表的详细信息。

表4-1 admin管理员信息表

列名	数据类型	长度	主键	外键	说明
id	int	11	是	否	编号
uname	varchar	50	否	是	姓名
upwd	varchar	50	否	是	密码

表4-2 banji班级信息表

--	--	--	--	--	--

列名	数据类型	长度	主键	外键	说明
id	int	11	是	否	编号
name	varchar	50	否	否	班级名
rs	int	11	否	否	人数

表4-3baoxiu报修信息表

列名	数据类型	长度	主键	外键	说明
id	int	11	是	否	编号
susheid	int	11	否	否	宿舍编号
bxinfo	varchar (255	否	否	报修内容
bxsj	varchar	20	否	否	报修时间
chuli	int	255	否	否	处理
clsj	varchar	20	否	否	处理时间
state	varchar	20	否	否	状态
sid	int	11	否	否	操作

表4-4 gonggao公告信息表

列名	数据类型	长度	主键	外键	说明
id	int	11	是	否	编号
title	varchar	200	否	否	标题
contents	varchar	500	否	否	内容
fbsj	varchar	20	否	否	发布时间

表4-5 laifang来访信息表

列名	数据类型	长度	主键	外键	说明
id	int	11	是	否	编号
name	varchar	20	否	否	姓名
idno	varchar	20	否	否	身份证号
lfsj	varchar	20	否	否	来访时间
remarks	varchar	255	否	否	备注

表4-6student学生信息表

列名	数据类型	长度	主键	外键	说明
id	int	11	是	否	编号
sno	varchar	50	否	否	学号
name	varchar	50	否	否	姓名
sex	datetime	10	否	否	性别
banji	int	11	否	是	班级
rxnf	varchar	50	否	是	入学时间
yuanxi	varchar	50	否	是	院系
mtel	varchar	50	否	是	母亲电话
ftel	varchar	50	否	是	父亲电话

susheid	int	11	否	是	宿舍编号
state	varchar	20	否	是	状态
upwd	varchar	50	否	是	在校情况

第五章 系统的实现

5.1 系统登录界面

管理员和学生可以通过用户名和密码进行登录，如图5-1所示。



图5-1 登录界面

5.2 管理员主界面

管理员登录后进入管理员主界面，主要包括基本信息管理，公告管理，学生管理，班级信息管理，宿舍楼管理，宿舍管理，离校学生管理，保修管理，水电费管理，来访管理；基础信息管理包括管理员信息管理和修改密码，其界面展示如图5-2所示。



图5-2管理员主界面

5.2.1 基本信息管理界面

管理员同样也能维护自身的信息，其界面展示图如图5-35-4所示

管理员信息管理	
帐号	密码
admin	admin
<input type="button" value="添加"/>	

图5-3添加管理员信息

登录名：	<input type="text" value="admin"/>
原密码：	<input type="password"/>
新密码：	<input type="password"/>
	<input type="button" value="修改"/> <input type="button" value="重置"/>

图5-4修改密码界面

5.2.2 公告管理界面

管理员管理、添加公告信息如图5-5所示

标题：

公告管理
<div>标题</div> <div>新学期开学典礼通知</div>

图5-5公告管理界面

5.2.3 学生管理界面

管理员可以操作、添加学生信息展示如图5-6所示

学生信息管理					
学号	姓名	性别	院系	入学年份	联系电话
<input type="button" value="添加"/> <input type="button" value="导入Excel"/> <input type="button" value="分配宿舍"/>					

图5-6学生管理界面

5.2.4 班级信息管理界面

管理员管理、添加班级信息展示如图5-7所示

班级信息管理	
班级编号	班级名称
2	101班
3	102班
4	201班
5	202班
<input type="button" value="添加"/>	

图5-7班级管理界面

5.2.5 宿舍楼管理界面

管理员可以操作宿舍楼信息，主要包括名称，宿舍数，总人数，性别等，其界面展示如图5-8所示

添加

名称：	<input type="text"/>
宿舍数：	<input type="text"/>
总人数：	<input type="text"/>
性别：	男 ▼
<input type="button" value="提交"/> <input type="button" value="重置"/>	

图5-8宿舍楼管理界面

5.2.6宿舍管理界面

管理员可以添加宿舍管理，主要包括名称，宿舍楼，容纳人数等。其界面如图5-9所示

宿舍管理		
名称	宿舍楼	容纳人数
101	A1	4
102	A1	4
101	A2	4
102	A2	4
添加		

图5-9宿舍管理界面

5.2.7 离校学生信息界面

管理员可以操作离校学生信息，主要包括学号，姓名，性别，院系，入学年份，联系电话等，其界面展示如图5-10所示

离校学生管理					
学号	姓名	性别	院系	入学年份	联系电话

图5-10离校学生信息界面

5.2.8报修管理界面

管理员可以操作报修管理界面，包括宿舍名，保修项目，报修时间，处理结果等。其界面如图如图5-11所示

报修管理			
宿舍	报修项目	报修时间	处理结果

图5-11报修管理界面

5.2.9 水电费管理界面

管理员可以操作水电费管理界面，并统计水电费信息，包括宿舍，日期，用水量，水费，用电量等。其界面展示如图5-12所示

水电费管理				
宿舍	日期	用水量（吨）	水费	用电量（度）
添加				

图5-12水电费管理界面

5.2.10来访管理界面

管理员可以操作，添加来访信息，主要包括姓名，身份证号，来访时间等。其界面展示如图5-13所示

来访管理			
姓名	身份证号	来访时间	
张明三	68788198910199989	2018-03-06 14:56:42	
添加			

图5-13来访管理界面

5.3学生管理主界面

学生登录后进入学生管理主界面，主要包括基本信息管理，通知公告，保修管理，水电费管理。其界面如图如图5-14所示



图5-14学生管理主界面

5.3.1 基本信息管理界面

学生可以查看个人资料并修改密码，修改密码主要包括学号，原密码，新密码，其界面展示如图5-15所示。个人资料包括学号，姓名，性别，入学年份，院系，联系电话，家长联系电话等，其界面如图5-16所示

学号：	<input type="text" value="201801001"/>
原密码：	<input type="password"/>
新密码：	<input type="password"/>
	<input type="button" value="修改"/> <input type="button" value="重置"/>

图5-15修改密码界面

个人资料	
学号：	201801001
姓名：	张明明
性别：	男 ▼
入学年份：	18
院系：	计算机
联系电话：	15567888878
家长联系电话：	15567889999
	提交 重置

图5-16个人资料界面

5.3.2通知公告管理界面

学生可以管理通知公告信息，主要包括标题，内容等，其界面展示如图5-17所示

标题：	<input type="text"/>	查询
公告管理		
	标题	
	新学期开学典礼通知	

图5-17通知公告管理界面

5.3.3保修管理界面

学生可以操作报修管理界面，包括宿舍，报修项目，报修时间，处理时间等，其界面如图5-18所示

报修管理			
宿舍	报修项目	报修时间	处理结果
填写报修单			

图5-18保修管理界面

5.3.4水电费管理界面

学生可以管理水电费管理，包括宿舍，日期，用水量，用电量，总金额等，其界面如图5-19所示。

查看	
宿舍：	A1-101
日期：	2018-02
用水量（吨）：	23 费用：34.5
用电量（度）：	150 费用：150.0
总金额：	184.5元
	返回

图5-19水电费管理界面

第六章 系统测试与维护

测试作为软件开发上线前的最终保障，有着不可忽视的重大意义。测试的目的也是为了保证在投入使用之前，软件是可以正常使用，且尽量没有BUG的。

同时在执行测试时，还可以跟踪问题，做到软件发布前，就把问题BUG扼杀在摇篮之中。

6.1 测试原则

1. 测试组人员和开发组人员尽量不要一致。
2. 严格按测试步骤。
3. 尽可能考虑多方面的情况，如各种异常，断电、死机、断网等
4. 测试过程中发现问题及时记录。
5. 测试完后对文档总结，修复一个问题后，要考虑是否产生其他的问题。

6.2 功能测试

1. 对登录功能进行测试：



图6-1登录界面测试)

在登录界面输入用户名admin，密码admin，用户类型选择管理员

登记登录按钮尝试登录。

运行结果



图6-2登录成功结果
登录成功后跳转页面。

2. 对公告管理测试:

标题:	<input type="text" value="2019年新学期开学啦"/>
内容:	<div>2019年新学期开学啦, 请各位同同学完善宿舍信息</div>
	<input type="button" value="提交"/> <input type="button" value="重置"/>

图6-3 添加公告测试
测试结果:

标题
新学期开学典礼通知
2019年新学期开学啦

图6-4 添加公告成功

测试结果：良好。

3. 对学生管理测试：

学号：	2015113020
姓名：	牛奶
性别：	男 ▼
入学年份：	2015
院系：	计算机
联系电话：	1800000000
家长联系电话：	1100000000
密码：
	提交 重置

图6-5添加学生测试

测试结果：

201801002	刘大磊	男	计算机	2018	15567812222
201801003	李霞	女	计算机	2018	15567815678
201801004	张芳	女	计算机	2018	15567816666
201801005	刘敬	女	计算机	2018	15567817878
201802001	王芳	女	计算机	2018	15567817845
2015113020	牛奶	男	计算机	2015	1800000000

图6-6添加学生成功

4. 对班级管理信息测试

添加	
名称:	<input type="text" value="A3"/>
宿舍数:	<input type="text" value="20"/>
总人数:	<input type="text" value="70"/>
性别:	<input type="text" value="男 ▼"/>
<input type="button" value="提交"/> <input type="button" value="重置"/>	

图6-7添加或修改班级信息

测试结果良好。

班级编号	班级名称	班级人数
2	101班	40
3	102班	40
4	201班	45
5	202班	45
6	303班	40

图6-8添加或修改班级信息

5. 对宿舍楼管理测试

名称:		<input type="text" value="A3"/>
宿舍数:		<input type="text" value="10"/>
总人数:		<input type="text" value="40"/>
性别:		<input type="text" value="男 ▼"/>
		<input type="button" value="提交"/> <input type="button" value="重置"/>

图6-9添加或修改班级信息

测试结果：添加或修改成功如图

名称	宿舍数	总人数
A1	20	80
A2	20	80
A3	10	40

图6-10添加或修改班级信息成功

6. 对保修处理功能测试，点击处理按钮，处理成功则弹出提示框处理成功。

处理结果	处理时间	状态	操作
已安排工人处理	2019-04-21 00:10:13	已处理	<input type="button" value="处理"/> <input type="button" value="删除"/>
	2019-04-21 00:10:21	已处理	<input type="button" value="处理"/> <input type="button" value="删除"/>

图6-11保修处理测试

7. 对水电费管理的测试

宿舍:	A2-101 ▼
日期:	2019-04
用水量(吨):	20
用电量(度):	100
	提交 重置

图6-12对水电费管理测试

测试结果：添加成功。

日期	用水量 (吨)	水费	用电量 (度)	电费	总费用	操作
2018-02	23	34.5	150	150.0	184.5	查看 修改 删除
2019-04	20	30.0	100	100.0	130.0	查看 修改 删除

图6-13对水电费管理测试结果

7. 对来访管理进行测试：

姓名:	牛奶
身份证号:	3515456456
备注:	家长参观测试
	提交 重置

图6-14对来访管理测试

测试结果：添家来访信息成功。

姓名	身份证号	来访时间	备注
张明三	68788198910199989	2018-03-06 14:56:42	找人
牛奶	3515456456	2019-04-21 00:16:58	家长参观测试

图6-14对来访管理测试结果

小结：对系统所有功能测试，未发现重大BUG，且系统能用户需求。

6. 3测试结果汇总

可用性测试用于检测系统的可操作性、可理解性、可学习性等方面内容。具体测试方面如表6-15所示。

表6-15 系统可用性测试

测试项	测试人员的评价
窗口移动、大小改变、关闭等操作是否正常	是
操作模块是否友好	是
模块、提示内容等文字描述是否正确	是
模块布局是否协调、合理	是
模块的状态是否正确（对选中项能否发生对应切换）	是
鼠标、键盘操作是否支持	是
所需数据项是否正确显示	是
操作流程是否合理	是
是否提供帮助信息	是

结束语

在这次毕业设计中，刚开始进行毕业设计的时候感觉十分困难，根本不知道该从何处下手，在经过一定的调研和资料累积后，并参考了相关的网站、查阅了大量的资料以后，对系统进行深入的需求调查和分析，结合辅导老师的指导以及学过的相关知识，经过不断的坚持、努力，设计最终被完成。我相信无论多么的困难，只要能够坚持下来，努力查阅资料，善于去发现研究，在研究中充分利用资源，再大的困难也能被解决。

在毕业设计的开始，心理上还是有一些忐忑。其中有一点便是对自己所掌握的知识量少的担心。毕竟首先得有相关的开发知识才能开始。其次就是对以往的知识遗忘成了开发过程中比较大的阻碍。以前老师在课堂上讲的一些重要知识点，有一些因为长时间没有去使用，记忆的东西便有些模糊了。最后，我逐渐认识到自信比专业技能可能更重要。知识点不会可以去学习，东西忘记了可以才去看看以前学过的，查缺补漏；但是一开始我就有点气馁了，是因为面对它，我有点不知所措了。

毕业设计是最能体现出我们所学知识的应用情况，是对我们大学期间所学知识的应用巩固和提高的时刻。设计本宿舍管理系统的过程不仅仅是对我学计算机知识的一次考验，更是对我运用所学知识解决处理现实问题能力的培养和锻炼。通过这次的毕业设计让我对软件的开发有了很深的认识了解，我的编程能力也得到了很大的提升。虽然在系统开发过程中也遇到了很多的困难，但是在老师和同学们的帮助下，虽然经理了一些小坎坷，但最终还是圆满的完成了毕业设计。

通过这套SSM宿舍管理系统的开发，我复习了之前所学的知识，也对以前学过的知识也做出了思考，将一些知识融合贯穿，并能在遇到某些问题的时候，尝试去用这些学过的东西来解读它们。这个过长不尽让我感到开心，也让我变得比以前自信了一点。

最后，我想感谢一些我的论文指导教师孙老师，如果在这过程中没有他的指导和督促，没有他给我的帮助和鼓励，我的毕业设计和论文也不可能完成的那么顺利。

致 谢

历时几个多月的紧张学习和努力，我的毕业设计终于如期、顺利的完成了。此次毕业设计是对我们日常所学理论的一次综合性评测，也是将理论应用到实践的一项考察。

首先我要感谢我的指导老师，在教学任务繁忙的情况下，抽出时间帮助我纠正我在设计当中出现的问题，并耐心地对我的论文作校正，是他的定期检查和指导使得我们的毕业设计高质量完成。他在我整个课题开发和设计的过程中，为程序的设计、框架的设计、代码的撰写方面以及论文框架的设计提供了很多宝贵的意见，并且为我推荐了许多有用的资料和文献，他的指导和建议使我受益匪浅，有了指导老师的辅导和指点，我论文才能够顺利完成。老师的认真负责的工作态度和治学严谨之道使我们这些即将踏入社会的毕业生受益匪浅。

同时，我也要感谢在设计过程中帮助过我的同学，是他们不厌其烦的解答才使得我的设计能够顺利完成。

答辩组的导师对我论文进行了严格的考核，并且在我进行答辩的过程中提出了非常有用的意见，我都悉心的接受，并且重新发现论文的不足，尽快的修改完成，感谢答辩组的各位导师能够对我论文做出客观的评价。

我还要感谢在大学的这几年学校对我们投入的精力，学校的老师不辞辛苦为我们上课和辅导，让我们更好的掌握的专业知识，提高我们的实际问题的处理能力，这将成为我人生中难忘的经历。

在最后要祝愿学校的领导教师以及和我一起奋斗的同学们工作顺利，事业有成，也要祝愿学校的前景更加辉煌。

参考文献

- [1] 王玉英. 基于JSP的MySQL数据库访问技术[J]. 现代计算机: 专业版, 2010, 19(14):63-66
- [2] 陈Maydene Fisher, Jon Ellis, Jonathan Bruce著. JDBC API Tutorial and Reference, Third Edition . 11 June, 2003
- [3] Jason Brittain Ian F.Darwin[美]. Tomcat权威指南. 北京: 中国电力出版社, 2008.
- [4] 石正喜. MySQL数据库实用教程. 北京: 北京师范大学出版社 2014
- [5] 张海潘. 软件工程理论[M]. 北京: 清华大学出版社, 1998.
- [6] Patrick O'Neil Elizabeth O'Neil[英]等. 数据库-原理、编程与性能(第二版 影印版)[M]. 北京: 高教育出版社, 2002.
- [7] 王雨竹. MySQL入门经典. 北京: 机械工业出版社 2013
- [8] 邓子云. JSP网络编程从基础到实践[M]. 北京: 电子工业出版社, 2009.
- [9] 陈圣国. Java信息系统设计与开发实例[M]. 北京: 机械工业出版社, 2004.
- [10] Todd Cook. JSP从入门到精通[W]. 北京: 电子工业出版社, 2003.
- [11] Bruce Eckel. Thinking in java(English Edition). 北京: 机械工业出版社, 2006
- [12] Hsiao I H, Sosnovsky S, Brusilovsky P. Guiding students to the right questions: adaptive navigation support in an e-learning system for Java programming[J]. Journal of Computer Assisted Learning, 2010, 26(4):270-283.
- [13] Vegh A. MySQL Database Server[M]. Web Development with the Mac®. Wiley Publishing, Inc., 2011, 179-194
- [14] Rasoolzadegan A, Barforoush A A. Reliable yet flexible software through formal model transformation (rule definition)[J]. Knowledge & Information Systems, 2014, 40(1):79-126

• 说明:

相似片段中“综合”包括:

《中文主要报纸全文数据库》 《中国专利特色数据库》 《中国主要会议论文特色数据库》 《港澳台文献资源》
《图书资源》 《维普优先出版论文全文数据库》 《年鉴资源》 《古籍文献资源》 《IPUB原创作品》

• 声明:

报告编号系送检论文检测报告在本系统中的唯一编号。

本报告为维普论文检测系统算法自动生成，仅对您所选择比对资源范围内检验结果负责，仅供参考。

客服热线: 400-607-5550 | 客服QQ: 4006075550 | 客服邮箱: vpcs@cqvip.com

唯一官方网站: <http://vpcs.cqvip.com>



