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： 宿舍管理系统

指 导 教 师：

设 计 地 点：

起 迄 日 期：

目 录

[摘 要 1](#_Toc508224534)

[Abstract 2](#_Toc508224535)

[第一章 绪论 3](#_Toc508224536)

[1.1研究背景 3](#_Toc508224537)

[1.2研究现状 3](#_Toc508224538)

[1.3 课题目标 4](#_Toc508224539)

[第二章 系统开发环境 5](#_Toc508224540)

[2.1开发工具 5](#_Toc508224541)

[2.1.1 MyEclipse 5](#_Toc508224542)

[2.1.2 Tomcat 5](#_Toc508224543)

[2.1.3 MYSQL 数据库 5](#_Toc508224544)

[2.2 JSP技术 6](#_Toc508224545)

[2.3 B/S结构 7](#_Toc508224546)

[2.4 JavaScript 7](#_Toc508224547)

[2.5软硬件需求 8](#_Toc508224548)

[第三章 需求分析 9](#_Toc508224549)

[3.1可行性分析 9](#_Toc508224550)

[3.1.1操作的可行性 9](#_Toc508224551)

[3.1.2 发展可行性 9](#_Toc508224552)

[3.1.3 技术可行性 9](#_Toc508224553)

[3.2流程分析 10](#_Toc508224554)

[3.3 性能需求分析 11](#_Toc508224555)

[3.3.1 系统的安全性 11](#_Toc508224556)

[3.3.2 数据的完整性 12](#_Toc508224557)

[3.4功能需求分析 12](#_Toc508224558)

[3.5用例分析 13](#_Toc508224559)

[3.5.1管理员用例图 13](#_Toc508224560)

[3.5.2学生用例图 14](#_Toc508224561)

[第四章 数据库设计 15](#_Toc508224562)

[4.1数据库设计原则 15](#_Toc508224563)

[4.2数据库E-R图设计 15](#_Toc508224564)

[4.3数据库设计表 18](#_Toc508224565)

[第五章 系统的实现 20](#_Toc508224566)

[5.1系统登录界面 20](#_Toc508224567)

[5.2管理员主界面 20](#_Toc508224568)

[5.2.1 基本信息管理界面 21](#_Toc508224569)

[5.2.2公告管理界面 21](#_Toc508224570)

[5.2.3 学生管理界面 22](#_Toc508224571)

[5.2.4 班级信息管理界面 22](#_Toc508224572)

[5.2.5 宿舍楼管理界面 22](#_Toc508224573)

[5.2.6宿舍管理界面 23](#_Toc508224574)

[5.2.7 离校学生信息界面 23](#_Toc508224575)

[5.2.8报修管理界面 23](#_Toc508224576)

[5.2.9 水电费管理界面 24](#_Toc508224577)

[5.2.10来访管理界面 24](#_Toc508224578)

[5.3学生管理主界面 24](#_Toc508224579)

[5.3.1 基本信息管理界面 25](#_Toc508224580)

[5.3.2通知公告管理界面 26](#_Toc508224581)

[5.3.3保修管理界面 26](#_Toc508224582)

[5.3.4水电费管理界面 26](#_Toc508224583)

[第六章 系统测试与维护 28](#_Toc508224584)

[6.1 功能测试 28](#_Toc508224585)

[6.2可用性测试 28](#_Toc508224586)

[6.3性能测试 29](#_Toc508224587)

[6.4测试结果分析 29](#_Toc508224588)

[6.5系统维护 29](#_Toc508224589)

[结束语 30](#_Toc508224590)

[致 谢 31](#_Toc508224591)

[参考文献 32](#_Toc508224592)

# 摘 要

随着科技的发展，计算机的应用，人们的生活方方面面都和互联网密不可分。计算机的普及使得人们的生活更加方便快捷，网络也遍及到我们生活的每个角落，为我们的学习、生活和工作带来了极大的方便。例如宿舍的管理，数据信息处理工作量大，数据繁多，使得宿舍管理系统的运行和管理就显得尤为重要。因此，具体现代的互联网多媒体信息技术的宿舍管理系统的建设，对于学校宿舍的管理具有非常重要的意义。采用宿舍管理系统是一种能够有效提高管理效率的方法。

JSP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，基于B/S模式，采用JSP技术，MYSQL数据库，从而保证系统的稳定性。宿舍管理系统实现了信息的查询、添加、管理等基本的操作，运用了模块化的设计方案，依据管理员及学生的需求，系统采用软件构件化的开发方式，使系统的结构分层，业务与实现分离，逻辑与数据分离；从功能上来说，系统是比较完备的，系统以Web界面与用户交互，为用户提供了信息，并易于用户操作，同时信息数据通过数据库的管理系统存储。

**关键字**：JSP技术；宿舍管理系统；MYSQL数据库 ；B/S模式

# Abstract

With the development of science and technology, the application of computer, all aspects of people's life and the Internet are inseparable. The popularization of computer makes people's life more convenient and quick, the network spreads to every corner of our life, has brought the great convenience for our study, the life and the work. For example, dormitory management, data information processing workload, a large amount of data, making dormitory management system operation and management is particularly important. Therefore, the construction of a dormitory management system for a specific modern Internet multimedia information technology is of great significance to the management of school dormitories. Dormitory management system is a way to effectively improve management efficiency.

JSP dormitory management system is currently an important part of the management of colleges and universities, the dormitory management system to the actual use of the background for the development, based on B / S mode, using JSP technology, MYSQL database, so as to ensure system stability. The dormitory management system realizes the basic operation such as inquiry, addition and management of information, and uses the modular design scheme. Based on the needs of administrators and students, the system adopts the development mode of software component to make the system structure hierarchical and business And the realization of the separation of logic and data separation; functionally speaking, the system is relatively complete, the system interface to the user with the Web interface to provide users with information, and easy for users to operate, and information data through the database management system storage.

**Keyword**：JSP technology; dormitory management system; MYSQL database; B / S mode

# 第一章 绪论

## 1.1研究背景

随着时代的发展，信息化进程不断加快，人们的工作和生活已经逐步进入到了信息化时代。本宿舍管理系统是以校园宿舍管理为实例而设计的一种实用型管理系统。随着高校住宿学生的不断增多，宿舍管理的负担越来越重，为了提高工作效率而开发出这套JSP宿舍管理系统。

近来年，我国高等教育迅猛发展，教育投入和招生人数不断增多，高校规模不断扩大，学校可调配资源也明显增加。我们经过对校园宿舍管理信息进行调查，发现了随着高校招生人数的不断扩大，校园宿舍管理工作也越来越繁重和琐碎，如学生信息、宿舍信息等种种问题，手工管理校园宿舍已经不能满足当前需要，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，来提高校园宿舍管理的工作效率，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。

随着计算机技术的发展以及计算机网络的逐渐普及，互联网成为人们查找信息的重要场所，二十一世纪是信息的时代，所以信息的交换和信息流通显得特别重要。因此，使用计算机来管理宿舍信息成为必然。开发合适的宿舍管理系统，可以方便学生宿舍的管理，提高管理工作效率及查询效率，有利于更好的了解学生、管理学生、贴近学生。

## 1.3 课题目标

首先要对本JSP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，并记录和处理相关的数据。

再次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；

最后要研究宿舍管理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

本系统的设计主要运用了MyEclipse和Tomcat后台服务器技术，设计时中采用了JSP技术完成了页面设计过程，接下来我们主要对MyEclipse、Tomcat和SqlServer数据库进行说明。

## 2.1.1 MyEclipse

MyEclipse，通常用来设计Java， J2EE的Eclipse插件集合，利用它可以完成对数据库和J2EE的开发。通过对应用程序服务器进行整合，可以大大的增强运算速度。MyEclipse的功能十分强大，几乎涵盖了各种设计功能。

## 2.1.2 Tomcat

Tomcat是一种是一个免费的开放源代码的Web 应用服务器。在访问客户比较少的情况下被经常采纳，在设计JSP 程序时成为最好的选择。换句话说，将Tomcat服务器安装到一台电脑上时，可通过它对HTML页面的访问请求进行配合。事实上，Tomcat可以看成Apache 服务器基础上的延伸，但它工作时相对比较独立，因此当tomcat工作时，事实上它是看成一个与Apache 服务器相对独立的进程运行的。

## 2.1.3 MYSQL 数据库

在软件项目中，通过使用数据库来操作数据，可以保证数据的独立性、一致性和安全性，为系统提供有效地访问数据的方式，同时为程序员减少了应用程序的开发时间。

常见的数据库无非分为两类，一类是关系型数据库(Oracle，DB2，MySQL，SQL Server )和非关系型数据库(NoSql、MongeDB)。不同的数据库有各自的优缺点，此处不做具体介绍。

MySQL支持多线程的特点，可以方便的利用系统资源，有效提高速度，而且提供了TCP/IP、ODBC和JDBC等多种方式途径来连接数据库；虽说其功能不够强大、规模也相对较小，但是对于本系统来说足够了，同时也可以进行二次开发的数据库表结构空间的扩展。

使用MySQL建立系统数据库，不仅有利于前期对数据的整合处理操作，同时利于后期的二次开发的数据扩展操作，对于有限级的数据处理，MySQL可以很好的胜任。

## 2.2 JSP技术

JSP程序使用了Java编程语言，JSP技术可以对动态网页进行封装。通过tags和scriptlets，网页还能访问存在于服务端的资源的应用逻辑。JSP可以分离网页逻辑与网页设计和显示，对可重用的基于组件的开发进行支撑，更容易的对基于Web的应用程序进行设计。

当Web服务器接收到访问JSP网页的请求时，首先运行的程序段，接下来将JSP文件中的HTML代码和运行效果一并返还给用户。通过Java文件的插入能够对数据库、网页多重定向等运行，从而满足构建动态网页所需要的程序。JSP和Servle相同，都可以通过服务器端运行。由于能够将一个HTML文本返回给用户端，所以用户端具备浏览器就可以进行浏览。HTML程序和穿插在内部的Java程序可以构建JSP网页。在服务器被用户端访问时，能够处理相应的Java代码，然后将产生的HTML页面再返回给用户端的浏览器。JSP的设计关键是Servlet，通常大型的Web应用程序的设计成果也通过Java Servlet和JSP的协作。JSP既拥有了方便快捷的Java程序，又统统的面向用户，既实现了平台的无关性危险还比较小，可以具备互联网的全部优势。JSP技术的优点：

（1）一次编写，到处运行。除了系统之外，代码无需做任何改动。

（2）系统的多平台支持。通常情况下，能够在任何情况下的全部平台上进行设计，能够安排在任何情况中，也能够在任何情况下进行发展。相比ASP/.net来说，优点是显而易见的。

（3）强大的可塑性。通过一个不大的Jar程序能够对Servlet/JSP进行运行，也能够通过很多服务器进行集群和负载平衡，甚至能够通过多台Application解决问题。单台服务器到多台服务器，Java展现了一个强大的功能。

（4）具有强大和多样化的开发工具支持。Java已经有了很多的设计方法，何况大部分工具都是没有花费的，这中间有很多都在多种平台之下顺畅的进展。

(5)支持服务器端组件。只有很有力的服务器端组件才能支撑web运行，因此开发者可以在调用web页面时通过另外的开发工具来达成纷乱效用的组件，来加强系统的可操作性。JSP可以通过完善的JAVA BEANS 组件来达成纷乱的商务功能。

内部对象表现出：request 用户端请求，此请求会涵盖来自GET/POST请求的参数； response网页传回用户端的反应；管理pageContext 网页的属性；session 与请求相关的会话；application servlet正在进行的内容；out 用来输送响应的输出流； config　代码片段配置对象；page　JSP网页本身。

## 2.3 Spring框架

Spring 是一个开源框架，是为了解决企业应用程序开发复杂性而创建的。框架的主要优势之一就是其分层架构，分层架构允许您选择使用哪一个组件，同时为 J2EE 应用程序开发提供集成的框架。

Spring 框架是一个 Java 平台, 为开发 Java 应用程序提供全面的基础结构支持。Spring 处理基础结构, 以便可以专注于应用程序。

Spring 使您能够从 "普通旧 Java 对象" (Pojo) 构建应用程序, 并将企业服务非侵入性地应用于 Pojo。此功能适用于 Java SE 编程模型以及全部和部分 Java EE。作为应用程序开发人员, 可以从 Spring 平台中受益的示例:

* 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：**

Java 应用程序是一个松散的术语, 它运行从约束的嵌入式应用程序到 n 层服务器端的企业应用程序的范围--通常由协作形成应用程序本身的对象组成。因此, 应用程序中的对象彼此具有依赖关系。

尽管 Java 平台提供了大量的应用程序开发功能, 但它缺乏将基本构建块组织为一个连贯的整体的方法, 而将该任务留给架构师和开发人员。尽管您可以使用设计模式 (如 "工厂"、"抽象工厂"、"生成器"、"装饰器" 和 "服务定位器") 来组合组成应用程序的各种类和对象实例, 但这些模式只是: 给出名称的最佳实践, 并使用描述模式的作用、应用它的位置、它所解决的问题等等。模式是必须在应用程序中实现自己的形式化最佳实践。

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, 从而解决了这一问题。Spring 框架将形式化设计模式编纂为一流的对象, 可以将其集成到自己的应用程序中。许多组织和机构以这种方式使用 Spring 框架来设计可靠、可维护的应用程序。

**面向切面的编程：**

面向切面的编程，即 AOP，是一种编程技术，它允许程序员对横切关注点或横切典型的职责分界线的行为（例如日志和事务管理）进行模块化。AOP 的核心构造是方面，它将那些影响多个类的行为封装到可重用的模块中。

AOP 和 IOC 是补充性的技术，它们都运用模块化方式解决企业应用程序开发中的复杂问题。在典型的面向对象开发方式中，可能要将日志记录语句放在所有方法和 Java 类中才能实现日志功能。在 AOP 方式中，可以反过来将日志服务模块化，并以声明的方式将它们应用到需要日志的组件上。当然，优势就是 Java 类不需要知道日志服务的存在，也不需要考虑相关的代码。所以，用 Spring AOP 编写的应用程序代码是松散耦合的。

AOP 的功能完全集成到了 Spring 事务管理、日志和其他各种特性的上下文中。

IOC 容器

Spring 设计的核心是 org.springframework.beans 包，它的设计目标是与 JavaBean 组件一起使用。这个包通常不是由用户直接使用，而是由服务器将其用作其他多数功能的底层中介。下一个最高级抽象是 BeanFactory 接口，它是工厂设计模式的实现，允许通过名称创建和检索对象。BeanFactory 也可以管理对象之间的关系。

**设计模式：**

1. 简单工厂

又叫做静态工厂方法（StaticFactory Method）模式，但不属于23种GOF设计模式之一。

简单工厂模式的实质是由一个工厂类根据传入的参数，动态决定应该创建哪一个产品类。

Spring中的BeanFactory就是简单工厂模式的体现，根据传入一个唯一的标识来获得Bean对象，但是否是在传入参数后创建还是传入参数前创建这个要根据具体情况来定。

2. 工厂方法（Factory Method）

定义一个用于创建对象的接口，让子类决定实例化哪一个类。Factory Method使一个类的实例化延迟到其子类。通常由应用程序直接使用new创建新的对象，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。

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。

3. 单例（Singleton）

保证一个类仅有一个实例，并提供一个访问它的全局访问点。

Spring中的单例模式完成了后半句话，即提供了全局的访问点BeanFactory。但没有从构造器级别去控制单例，这是因为Spring管理的是是任意的Java对象。

4. 适配器（Adapter）

将一个类的接口转换成客户希望的另外一个接口。Adapter模式使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

在Spring的Aop中，使用的Advice（通知）来增强被代理类的功能。Spring实现这一AOP功能的原理就使用代理模式（1、JDK动态代理。2、CGLib字节码生成技术代理。）对类进行方法级别的切面增强，即，生成被代理类的代理类， 并在代理类的方法前，设置拦截器，通过执行拦截器重的内容增强了代理方法的功能，实现的面向切面编程。

5.包装器（Decorator）

动态地给一个对象添加一些额外的职责。就增加功能来说，Decorator模式相比生成子类更为灵活。

6. 代理（Proxy）

为其他对象提供一种代理以控制对这个对象的访问。Spring的Proxy模式在aop中有体现，比如JdkDynamicAopProxy和Cglib2AopProxy。

7.观察者（Observer）

定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。Spring中Observer模式常用的地方是listener的实现。如ApplicationListener。

8. 策略（Strategy）

定义一系列的算法，把它们一个个封装起来，并且使它们可相互替换。本模式使得算法可独立于使用它的客户而变化。Spring中在实例化对象的时候用到Strategy模式

9.模板方法（Template Method）

在模板模式（Template Pattern）中，一个抽象类公开定义了执行它的方法的方式/模板。它的子类可以按需要重写方法实现，但调用将以抽象类中定义的方式进行。这种类型的设计模式属于行为型模式。Template Method模式一般是需要继承的。这里想要探讨另一种对Template Method的理解。Spring中的JdbcTemplate，在用这个类时并不想去继承这个类，因为这个类的方法太多，但是我们还是想用到JdbcTemplate已有的稳定的、公用的数据库连接，那么我们怎么办呢？我们可以把变化的东西抽出来作为一个参数传入JdbcTemplate的方法中。但是变化的东西是一段代码，而且这段代码会用到JdbcTemplate中的变量

## 2.3 Mybatis框架

MyBatis SQL 映射器框架使在面向对象的应用程序中使用关系数据库变得更加容易。MyBatis 使用 XML 描述符或注释将对象与存储过程或 SQL 语句结合在一起。与对象关系映射工具相比, 简单性是 MyBatis 数据映射器的最大优势。

MyBatis 是支持自定义 SQL 存储过程的一流持久性框架和高级映射。MyBatis 几乎消除了所有的 JDBC 代码和手动设置参数和结果的检索。MyBatis 可以使用简单的 XML 或批注进行配置并将原语、映射接口和 Java Pojo (纯旧 Java 对象) 映射到数据库记录。

* 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, 您会立即看到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，相信大家在组合复杂的的SQL语句的时候，需要去拼接，稍不注意哪怕少了个空格，都会导致错误。Mybatis的动态SQL功能正是为了解决这种问题， 其通过 if, choose, when, otherwise, trim, where, set, foreach标签，可组合成非常灵活的SQL语句，从而提高开发人员的效率。

MYBATIS 对 JDBC 来说, 大大简化了您的代码, 并使其保持干净、易于理解和维护。MyBatis 3 引入了一些重大改进，以使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。

SpringMVC是一种基于Java，实现了Web MVC设计模式，请求驱动类型的轻量级Web框架，即使用了MVC架构模式的思想，将Web层进行职责解耦。基于请求驱动指的就是使用请求-响应模型，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。

1）首先用户发送请求——>DispatcherServlet，前端控制器收到请求后自己不进行处理，而是委托给其他的解析器进行处理，作为统一访问点，进行全局的流程控制；

（2）DispatcherServlet——>HandlerMapping，处理器映射器将会把请求映射为HandlerExecutionChain对象（包含一个Handler处理器对象、多个HandlerInterceptor拦截器）对象；

（3）DispatcherServlet——>HandlerAdapter，处理器适配器将会把处理器包装为适配器，从而支持多种类型的处理器，即适配器设计模式的应用，从而很容易支持很多类型的处理器；

（4）HandlerAdapter——>调用处理器相应功能处理方法，并返回一个ModelAndView对象（包含模型数据、逻辑视图名）；

（5）ModelAndView对象（Model部分是业务对象返回的模型数据，View部分为逻辑视图名）——> ViewResolver， 视图解析器将把逻辑视图名解析为具体的View；

（6）View——>渲染，View会根据传进来的Model模型数据进行渲染，此处的Model实际是一个Map数据结构；

（7）返回控制权给DispatcherServlet，由DispatcherServlet返回响应给用户，到此一个流程结束。

## 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。

日志一般分5个等级，从低到高分别是 DEBUG INFO WARN ERROR FATAL。对应中文的意思就是 调试信息 一般信息 警告信息 错误信息 严重错误信息。在记录日志的时候，就可以根据不同的日志级别来进行记录。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

在三层体系结构的B/S（Browser/Server，浏览器/服务器结构）系统中，B/S架构的全称为Browser/Server，即浏览器/服务器结构。Browser指的是Web浏览器，极少数事务逻辑在前端实现，但主要事务逻辑在服务器端实现，Browser客户端，WebApp服务器端和DB端构成所谓的三层架构。B/S架构的系统无须特别安装，只有Web浏览器即可。

## 2.4 JavaScript

JavaScript 是 Web 的编程语言。所有现代的 HTML 页面都使用 JavaScript。

JavaScript 非常容易学。JavaScript 是互联网上最流行的脚本语言，这门语言可用于 HTML 和 web，更可广泛用于服务器、PC、笔记本电脑、平板电脑和智能手机等设备。

同其他脚本语言一样，JavaScript也是一种解释性语言，其提供了一个非常方便的开发过程。JavaScript的语法基本结构形式与C、C++、Java十分类似。但在使用前，不像这些语言需要先编译，而是在程序运行过程中被逐行地解释。JavaScript与HTML标识结合在一起，从而方便用户的使用操作。

Javascript的应用能够达成网页和客户之间及时、动态、交互性的关系。这样网页可以呈现非常丰富的信息和非常好看的内容。在本系统中不少设计都采用了javascript技术，例如校验用户输入数据的有效性等

## 2.5软硬件需求

硬件需求：

CPU：Pentium以上计算机

内存: 512M以上

软件需求：

操作系统版本：Windows XP /vista/Win7

开发工具：MyEclipse 6.0.1

后台服务器：Apache Tomcat 6.0

开发语言：Java

浏览器：Google Chrome

数据库：MYSQL5.0+

# 第三章 需求分析

## 3.1可行性分析

## 3.1.1操作的可行性

本JSP宿舍管理系统的登录界面简单，采用常见的界面窗口的登录界面，而宿舍管理系统的开发则采用的是较为常用的JSP语言开发，而且学校科技十分现代化，学生及宿舍管理人员的学历都较高，而高校的电脑等设施齐全，具备着做宿舍管理的基础条件，因此校园很有必要，也很有条件做宿舍管理系统，以更方便管理学生，更贴近学生的需求。

## 3.1.2 发展可行性

随着高校招生人数的不断增加，学生越来越多，宿舍管理数据繁多，学生信息量巨大，需要有一个完善的宿舍管理系统来完整准确的记录学生、宿舍信息，以便于更好的管理学生，贴近学生。最重要的是要确保科学的管理方法，为学校获得长远发展的竞争力提高的唯一途径。因此，对于学校的发展创新来说，开发一个宿舍管理系统是可行的。

## 3.1.3 技术可行性

本JSP宿舍管理系统的开发基于B/S模式，主要包括前端应用程序的开发以及后台数据库的建立和维护两个方面。对于应用程序的开发要求具备功能要完备、使用应简单等特点，而对于数据库的建立和维护则要求建立一个数据完整性强、数据安全性好、数据稳定性高的库。

在过去很长一段时间，网络软件的开发都采用客户端（Client）/服务器（Server）模式，即所谓的C/S模式。比如在Java中，可以通过Socket实现一个客户端/服务器的架构。在这种模式下，主要业务逻辑都集中与客户端的程序，因此，必然导致以下问题：

1.系统的安装、运行、调试、维护和升级的困难；

2.在宿舍管理系统中，客户端分布在信息逻辑和用户界面之间，必然增加了系统的安全隐患。

正是由于C/S架构的这些问题，人们逐渐接受了基于浏览器(Brower)/服务器(Server)结构的系统，即B/S模式。在B/S这个模式中，用通用的浏览器来取代之前的客户端系统程序就是主要的核心。

B/S模式对C/S模式虽然只作了少许改动，但却带来了很大的优势：因为客户端统称为浏览器，所以，应用程序在下载、升级、维护时，仅需要在服务器端进行配置就可以了。所有的配置工作均集中在服务器端，大大的解决了了系统的安全隐患。

虽然B/S模式拥有很多优势，但是因为在客户端能够利用编程对象仅仅是浏览器，因此，之前在C/S模式下很简单就能实现的一些客户端功能，在B/S模式下，却变的很难实现甚至无法实现。

虽然如此，B/S模式仍然是当前许多应用系统的首要选择模式。

对于计算机专业毕业的学生，对数据库比较熟悉，该系统运行MYSQL数据库，掌握了Java编程语言，建立这样一个系统应该在能力范围之内

因此从技术方面讲开发此系统是可行的。

## 3.2流程分析

通过实际校园宿舍管理进行调查，从而对本JSP宿舍管理系统业务流程进行具体的分析。用户要想对系统进行管理操作，必须首先通过正确的账号密码进行登录系统。登录模块主要完成管理员及学生的权限登入，输入用户名密码然后验证数据的正确与否，流程图如图3-1所示



图3-1 登录流程图

本JSP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

宿舍管理系统要严格控制管理权限，具体要求如下：

1.要想对本JSP宿舍管理系统的信息进行管理，首先要依靠用户名和密码在系统中登陆，无权限的用户不可以通过任何方式登录系统和对系统的任何信息和数据进行操作，这样可以保证系统的安全可靠性和准确性。

2.在具体实现中对不同的权限进行设定，不同权限的用户在系统中登陆后，不可以越级操作，只有系统管理员能够使用全部模块。

## 3.3.2 数据的完整性

1、所有记录信息要保持全面，信息记录内容不可以是空。

2、各种数据间相互联系要保持正确。

3、相同数据在不同记录中要保持一致。

## 3.4功能需求分析

JSP宿舍管理系统主要分为两个用户角色，分别是管理员和学生，其具体功能如下：

管理员：管理员登录后主要功能有：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。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

本JSP宿舍管理系统主要满足2种类型用户的需求，这2种类型用户分别为管理员和学生，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。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：基本信息管理，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、统计和传播等操作必须在数据库管理系统的的支持下才能进行。目前，电脑系统的关键与核心部分就是数据库。数据库开发的优劣对整个系统的质量和速度有着直接影响。

## 4.1数据库设计原则

在概念设计中，通常用四种方法：

自顶向下、自底向上、逐步扩张、混合策略，以它为骨架集成由自底向上策略中设计的各局部概念结构

在物理结构设计阶段首先分两步走：

确定数据库的物理结构，在关系数据库中主要是指存取方法和存储结构。 对物理结构的评价是时间和空间效率。

选取正确的关系模式存取方法，常用的有：存取索引方法、聚簇存取、HASH存取方法等。

## 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。

在概念模式的设计中，E-R模型法是最常见的设计方法。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：通常概念设计的主要目标是设计并反映出某一个组织部门的信息需求中的数据库系统模式，数据库系统的概念模式通常是独立于：数据库系统的逻辑结构、DBMS即数据库管理系统、计算机系统等模式。

概念模式的设计方法主要是以需求分析作为基础，并且用概念数据模型即E-R模型等来表示数据之间的相互联系，设计并反映出用户所需的信息需求以及处理需求的数据库系统等概念模式。通常概念设计的主要目标是描述应用领域的信息模式，并且支持用户的相关应用，最终方便转换并为数据库系统逻辑模式并且方便用户理解。在此阶段，用户可以参与和评价数据库系统的设计，最终来保证数据库系统的设计与用户的需求相吻合。在概念模式的设计中，E-R模型法是最常见的设计方法。本系统的E-R图如下图所示：

实体间关系

属性

实体

(1)管理员信息实体E-R图如图4-1所示：



图4-1管理员信息实体图

(2)班级信息实体E-R图如图4-2所示：



图4-2班级信息实体图

(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

(4)公告信息实体E-R图如图4-4所示：



图4-4公告信息实体图

(5)来访信息实体E-R图如图4-5所示：



图4-5来访信息实体图

(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

## 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

管理员可以管理，添加管理员信息并修改或重置密码，其界面展示图如图5-35-4所示

****

图5-3添加管理员信息

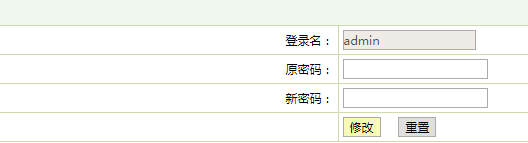


图5-4修改密码界面

## 5.2.2公告管理界面

管理员可以管理，添加公告信息，主要包括标题，内容等，其界面展示图如图5-5所示

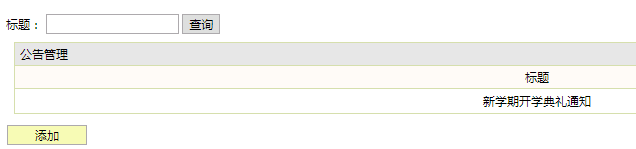
****

图5-5公告管理界面

## 5.2.3 学生管理界面

管理员可以管理，添加学生信息，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5-6所示

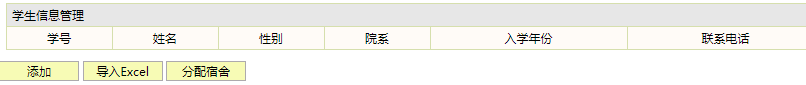
****

图5-6学生管理界面

## 5.2.4 班级信息管理界面

管理员可以管理，添加班级信息，主要包括班级编号，班级名称，其界面展示如图5-7所示

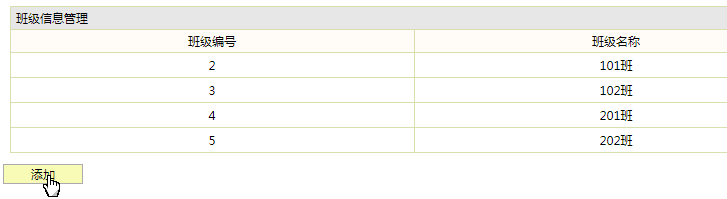


图5-7班级管理界面

## 5.2.5 宿舍楼管理界面

管理员可以管理宿舍楼信息，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



图5-8宿舍楼管理界面

## 5.2.6宿舍管理界面

管理员可以管理，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

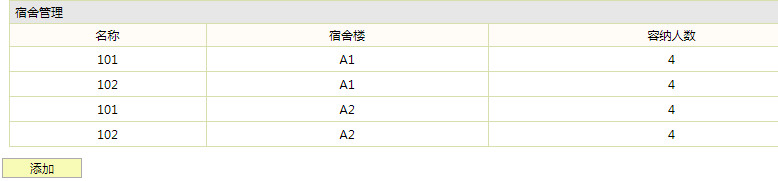


图5-9宿舍管理界面

## 5.2.7 离校学生信息界面

管理员可以管理离校学生信息，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

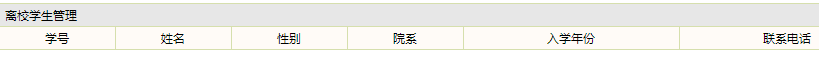


图5-10离校学生信息界面

## 5.2.8报修管理界面

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



图5-11报修管理界面

## 5.2.9 水电费管理界面

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

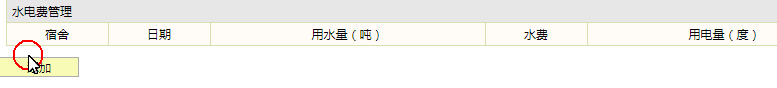


图5-12水电费管理界面

## 5.2.10来访管理界面

管理员可以管理，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

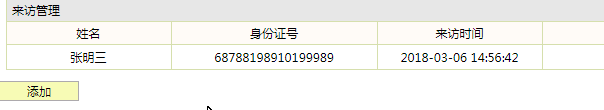


图5-13来访管理界面

## 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

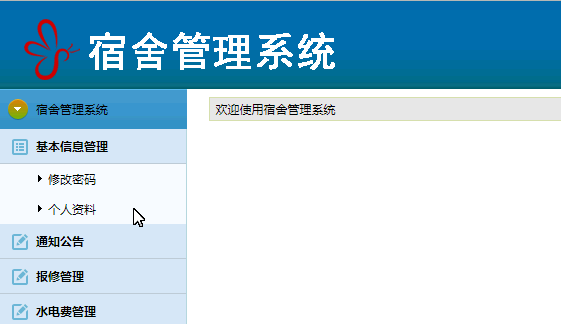


图5-14学生管理主界面

## 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



图5-15修改密码界面



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

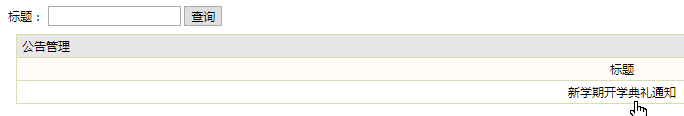
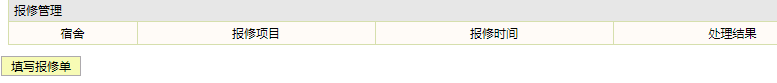


图5-17通知公告管理界面

## 5.3.3保修管理界面

学生可以管理报修管理界面，主要包括宿舍，报修项目，报修时间，处理时间等，其界面如图5-18所示

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，主要包括宿舍，日期，用水量，用电量，总金额等，其界面如图5-19所示。



图5-19水电费管理界面

# 第六章 系统测试与维护

系统测试是软件设计的最后一道工序，也是非常重要的一步。用于检查软件的质量、性能、可靠性等是否符合用户需求。一套严谨的、规范的、完善的测试过程将大大提高软件的质量、可信度、可靠性，降低软件的出错率，降低用户风险系数。通常系统测试包括：功能测试，可用性测试，性能测试、兼容性测试等方面。

## 6.1 功能测试

功能测试主要包括五项内容：适用性、准确性、可操作性、依从性、安全性。

本系统功能测试如表6-1所示：

表6-1 系统功能测试

|  |  |
| --- | --- |
| 测试内容 | 测试结果 |
| 适用性 | 好 |
| 准确性 | 好 |
| 可操作性 | 好 |
| 依从性 | 好 |
| 安全性 | 好 |

## 6.2可用性测试

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6-2所示。

表6-2 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 |

## 6.3性能测试

性能测试主要通过模拟系统运行环境，测试系统性能是否符合客户需求。性能测试的重要技术指标就是：系统运行速度、网络响应时间和支持并发节点数。

1）系统运行速度：通过在不同计算机上试运行本系统，没有发现有任何迟滞、停顿现象。

2）网络响应时间：网络响应时间主要包括网络最小响应时间、平均响应时间、最大响应时间三个参数。经过测试，在网络运营良好状态下，校园局域网内响应时间三参数为：1/2/6s，校园外网响应时间三参数为3/7/12s，符合客户需求，属于用户心理可承受范围。

3）支持并发节点数：经过模拟环境测试，本系统在并发节点达46个时，网络运营速度会发生较大波动，延迟时间10秒左右，符合客户需求。

## 6.4测试结果分析

经过对上述测试结果分析，本系统符合JSP宿舍管理系统的客户要求和用户需求。所有基本功能点实现，操作简单，操作流程符合业务需求，产品运行性能良好，是一款值得推广的大众化软件。

## 6.5系统维护

本JSP宿舍管理系统软件设计要求基本都可以达到，此系统具有完整的软件功能，良好的用户界面，能够正确的处理错误信息，而且能够准确的提出错误的种类。但是系统测试时也出现了一些系统的不足和缺陷，例如系统在用户第一次应用软件登陆时，必须使用提前设定好的管理员登陆账号，才可以对系统进行操作，需要在软件进一步修改和维护时予以纠正。总体说来，软件通过测试。

# 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，所有困难终都会被解决的。

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，以及经过自己的不懈努力，最终圆满的完成了毕业设计。

通过这套JSP宿舍管理系统的开发，我巩固了之前学过的知识，平时所学到的知识不仅融合了，还获得了许多JSP知识，我深人体会到学科之间的关联效应。在整个设计过程中明白了许多东西，也培养独立工作能力，树立信心，提高了自己的工作能力，我相信在以后的学习和工作生活中有至关重要的作用。这次设计也大大提高了我的动手的能力，让我在学习中充分体会到了探索的乐趣，享受成功的创作过程，在本次设计过程中汲取的东西，是一笔宝贵的财富，让我受益匪浅。

回顾过去做毕业设计的整个过程，充满了付出和收获，虽然经历了种种困难，不断修改，不断探索，但是当你看到成果时候的感觉，是一种难以用言语表达的喜悦之感，这些在毕业设计过程中学习到的东西将会使我终身受益！

最后，感谢指导老师的关心和指导，在我毕业设计的整个过程中，他给与了我很多的帮助和讲解，在导师的帮助下我的毕业设计才能如此顺利的完成。

# 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，并耐性地为我的论文作校正，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的知识，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](http://www.ebookee.com.cn/JDBC-API-Tutorial-and-Reference-Third-Edition_6012.html). 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