第 1 章 宿舍管理系统简介  
  
 该宿舍管理管理系统采用Java，vue和MySql数据库等工具，在工作平台Itellij IDEA，Visual Studio Code上开发而成，使用企业里最新用的Spring+ Spring MVC+Mybatis三大框架整合开发,更符合企业级的开发流程和代码规范。采用vue前端开发框架,使用户有更好的交互体验，使程序使用更加方便快捷。  
  
 本系统能提高了学生宿舍的管理效率，大大地减轻了宿管员的负担，使宿管员能随时随地查询楼层信息和宿舍住宿信息。宿舍管理系统使宿管员可以更加直观的了解到校园的宿舍楼的栋数和宿舍楼每层的各个宿舍的成员信息，可以对宿舍成员的信息进行修改，删除，对校园住宿学生信息的查询等相关操作，关于宿管员登录采用手机号，登录密码，验证码进行登录，具有记住密码，忘记密码等功能  
  
关键字: JavaEE，宿舍管理系统， spring, Spring MVC,Mybatis

第2章系统开发技术及 主要架构的简介  
  
2.1开发技术的选择  
  
 由于JAVA面向对象，安全性高，跨平台，易移植，易掌握等特点，尤其是在安全性和跨平台性两个方面非常显著。因此，最后我选择采用Java技术开发这个系统。  
  
2.2项目开发的工具和技术  
  
 这个web系统的服务器采用开源的Tomcat6.0,数据库采用开源的MySQL。在MyEcipse工作平台，上使用Java、JavaScript 与JSP技术，采用MVC模式开发，使用Spring+ Spring MVC+Mybatis三大框架整合开发。  
  
2.3 MyEclipe 简介  
  
 MyEclipse企业级工作平台(MyEclipse Enterprise Workbench ),简称MyEclipse是对EclipseIDE的扩展，利用它我们可以很方便地使用数据库和JavaEE开发和发布系统，提高了工作效率。  
  
2.4 MySql简介  
  
 MySQL名字的来历是一个小型关系型数据库管理系统，开发者为瑞典MySQLAB公司，在2008年1月16 号被Sun公司收购。MySQL被广泛地应用在Internet\_上的中小型网站中。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一-特点， 许多中小型网站为了降低网站总体拥有成本而选择了MySQL作为网站数据库。  
  
2.5 Tomcat 简介

Tomcat是Apache软件基金会( Apache Software Foundation)的Jakarta项目中的一个核心项目，由Apache、Sun和其他一些公司及个人共同开发而成。因为Tomcat技术先进、性能稳定，而且免费，因而深受Java爱好者的喜爱并得到了部分软件开发商的认可，成为目前比较流行的Web应用服务器。  
  
2.6 JDBC简介  
  
 JDBC (Java Data Base Connectivity,java数据库连接)是一种用于执行SQL语句的JavaAPI，可以为多种关系数据库提供一个标准的API。据此可以构建更高级的工具和接口，使数据库开发人员能够用纯Java API编写数据库应用程序,本系统采用JDBC连接数据库。  
  
2.7 spring 简介  
  
 Spring是一个开源框架，Spring 是于2003年兴起的一个轻量级的Java开发框架，由Rod Johnson在其著作Expert One-On-One J2EE Development and Design中阐述的部分理念和原型衍生而来。它是为了解决企业应用开发的复杂性而创建的。Spring使用基本的JavaBean来完成以前只可能由EJB完成的事情。然而,Spring的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何Java应用都可以从Spring中受益。简单来说， Spring 是一个轻量级的控制反转(IOC)和面向切面( AOP)的容器框架。  
  
2.8 springmvc 简介  
  
 Spring MVC属于SpringFrameWork的后续产品，已经融合在Spring Web Flow里面。SpringMVC分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制。  
  
架构流程如下:  
  
1、用户发送请求至前端控制器DispatcherServlet  
  
2、DispatcherServlet 收到请求调用HandlerMapping处理器映射器。  
  
3、处理器映射器根据请求url找到具体的处理器，生成处理器对象及处理器拦截器(如果有则生成)一并 返回给DispatcherServlet。  
  
4、DispatcherServlet 通过Handler Adapter处理器适配器调用处理器  
5、执行处理器(Controller，也叫后端控制器)。

6、Controller 执行完成返回ModelAndView  
  
7、HandlerAdapter 将controller 执行结果ModelAndView 返回给DispatcherServlet  
  
8、DispatcherServlet 将ModelAndView传给ViewReslover视图解析器9、ViewReslover 解析后返回具体View  
  
10、DispatcherServlet对View进行渲染视图(即将模型数据填充至视图中)。11、DispatcherServlet 响应用户  
  
2.9 mybatis简介  
  
MyBatis本是apache的一个开源项目iBatis,2010年这个项目由apache software foundation迁移到了google code，并且改名为MyBatis。MyBatis是一个基于Java的持久层框架。iBATIS提供的持久层框架包括SQLMaps和Data Access Objects (DAO) MyBatis 消除了几乎所有的JDBC代码和参数的手工设置以及结果集的检索。MyBatis使用简单的XML或注解用于配置和原始映射，将接口和Java的POJOs (Plain Old Java Objects，普通的Java对象)映射成数据库中的记录。  
  
架构流程如下:  
  
1、mybatis 配置  
  
SqlMapConfig .xml，此文件作为mybatis的全局配置文件，配置了mybatis的运行环境等信息。  
  
mapper.xml文件即sql映射文件，文件中配置了操作数据库的sql语句。此文件需要在SqlMapConfig .xml中加载。  
  
2、通过mybatis环境等配置信息构造SqlSessionFactory即会话工厂  
  
3、由会话工厂创建sqlSession即会话,操作数据库需要通过sqlSession进行。4、mybatis 底层自定义了Executor 执行器接口操作数据库，Executor 接口有两个实现，-一个是基本执行器、一个是缓存执行器。  
  
5、Mapped Statement也是mybatis - -个底层封装对象，它包装了mybatis 配置信息及sql映射信息等。mapper.xml 文件中一个 sql对应一个 Mapped Statement对象，sql 的id即是Mapped statement的id。  
  
6、Mapped Statement对sql执行输入参数进行定义，包括HashMap、基本类，型、pojo, Executor通过Mapped Statement 在执行sql前将输入的java对象映射至sql中，输入参数映射就是jdbc编程中对preparedStatement设置参数。  
  
7、Mapped Statement对sql执行输出结果进行定义，包括HashMap、基本类型、pojo，Executor通过MappedStatement在执行sql后将输出结果映射至java对象中，输出结果映射过程相当于jdbc编程中对结果的解析处理过程。

2.10 vue简介

**1. 什么是 Vue**

Vue (读音 /vjuː/，类似于 view) 是一套用于构建用户界面的渐进式框架。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，Vue 也完全能够为复杂的单页应用提供驱动。

**2. 为什么要使用 Vue**

**2.1 Vue.js 让基于网页的前端应用程序开发起来更加方便**

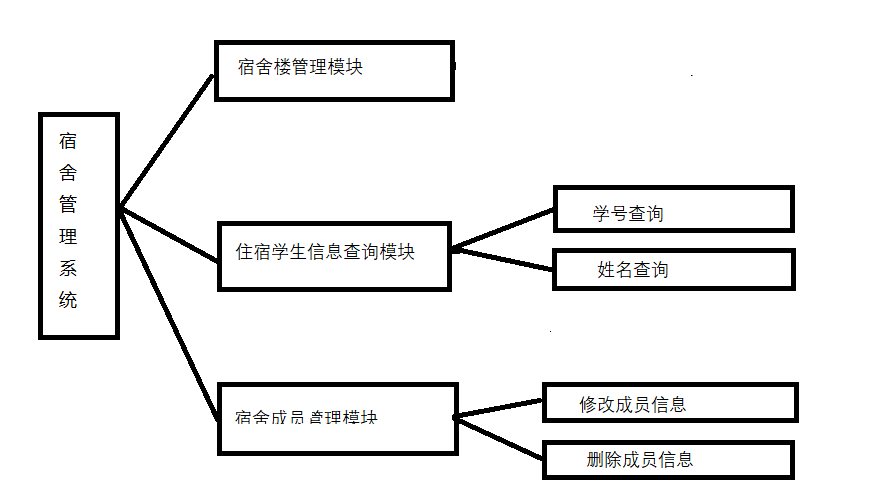
相信同学们一定或多或少做过网页开发，如果你使用原生 JavaScript 或者 Jquery 来开发，那么不可避免的要大量操作 DOM，而 Vue.js 不同，因为 Vue.js 有声明式，响应式的数据绑定，与组件化的开发，并且还使用了 Virtual DOM 这个看名字就觉得高大上的技术，Vue.js 让我们尽量避免了繁琐的 DOM 操作，它可以根据数据的改变来驱动视图的更新，这极大的提高了我们的开发效率。  
第3章需求分析  
  
3.1功能分析  
  
 系统结构功能图能够更加明确的、直观的体现内部组织关系，更加清晰的理清内部逻辑关系，使系统各个功能模块一目了然。宿舍管理系统的用户有宿管员。该系统包括宿舍楼管理模块，住宿学生信息查询模块，宿舍成员管理模块。

宿舍楼管理模块：随着学校的不断发展，学校的宿舍楼也会随着学校规模的扩大而增加，该模块可以实现对宿舍楼及宿舍楼层的添加。

住宿学生信息查询模块：为了方便管理员对住宿人员的住宿信息的查询，该模块可以根据学生学号或姓名进行查询进而得到学生的具体住宿情况

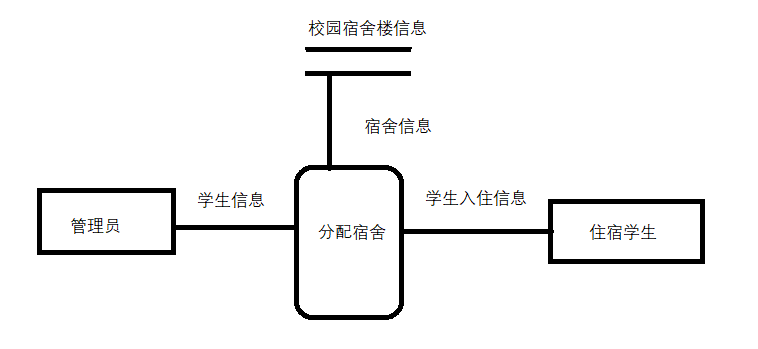
宿舍成员管理模块：该模块可以看到宿舍的住宿情况，管理员可以对宿舍成员的信息进行修改与删除操

由功能分析可以得到该系统的系统功能结构，如下图所示：

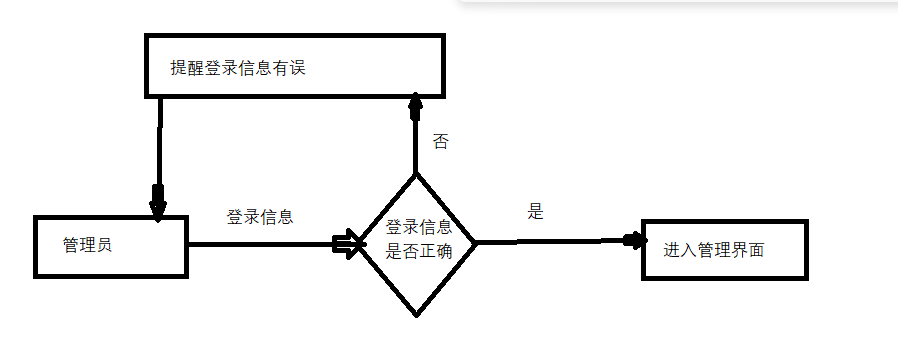


3.2系统数据流程图

由以上的功能分析，得出下图的系统数据流程图，该数据流程图直观的反映了该系统的整体框架和设计要求

3.3登录流程

管理员登录时，系统先判断管理员信息是否存在，手机号，密码，验证码信息是否正确，如果正确则进入宿舍管理界面，进行相关的管理操作，如果手机号或密码错误，则提示管理员手机号或密码输入错误请重新输入，验证码错误则提示验证码错误请重新输入，登录流程图入下图所示



3.4概念数据模型设计

本系统的实体有学生基本信息实体，宿舍楼信息实体，宿舍信息实体，管理员信息实体，入住学生信息实体。

用实体—联系的方法来描述概念模型，是通过E-R图来实现的，E-R图有三种基本成分，实体，联系，属性。

在此宿舍管理系统中，各个实体包含的属性分别如下：

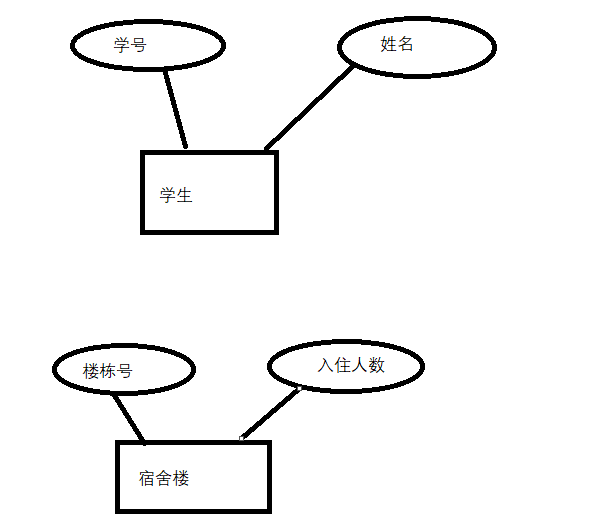
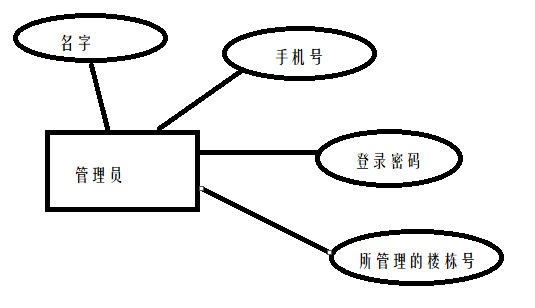
学生基本信息实体：学号，姓名

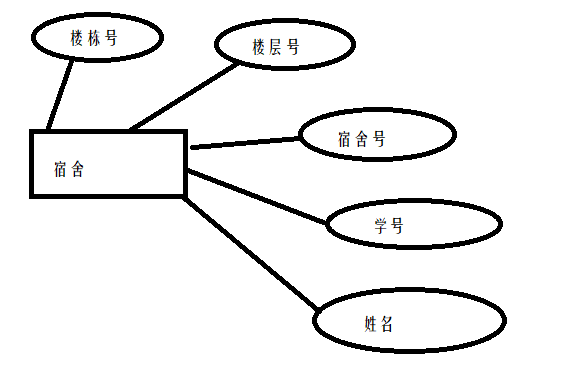
宿舍楼信息实体：楼栋号，入住人数

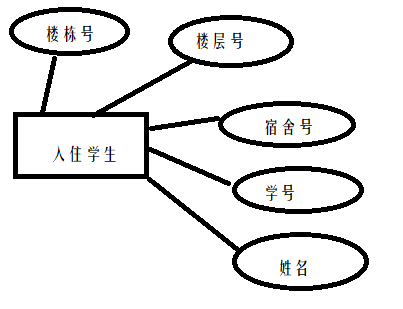
宿舍信息实体：楼栋号，楼层号，宿舍号，学号，姓名

管理员信息实体：名字，电话，密码，管理的楼栋号

学生入住信息实体：学号，姓名，楼栋号，楼层号，宿舍号

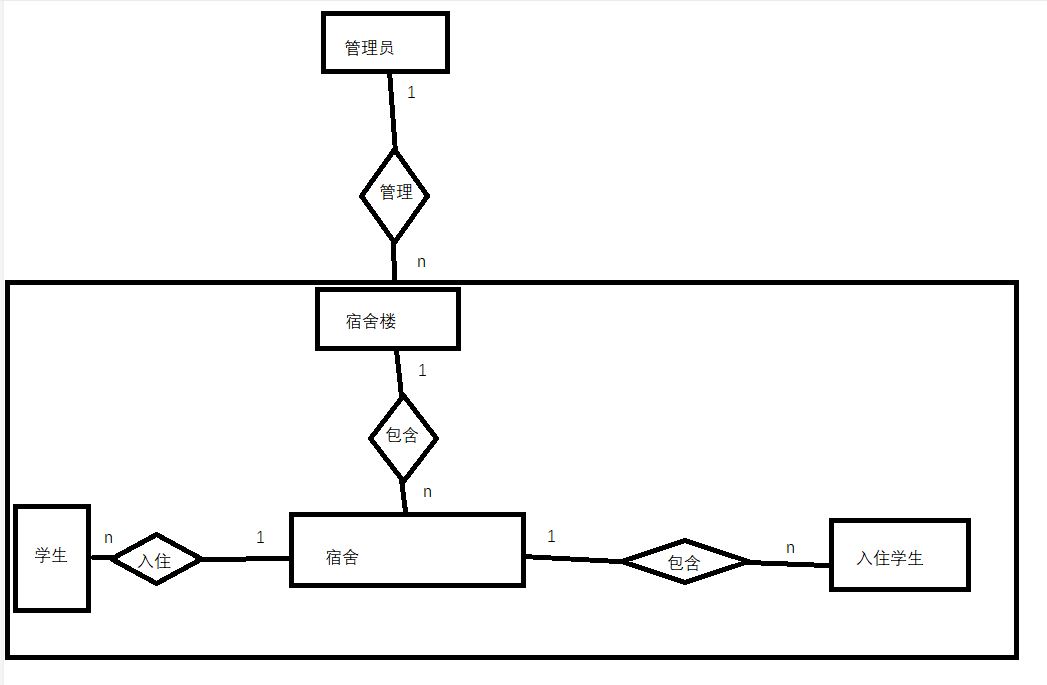
 





3.5实体联系图

从上面的需求分析和总体设计得出学生基本信息，宿舍楼信息，宿舍信息，管理员信息，学生入住信息实体之间得联系图如下



3.6数据库表结构设计

根据系统的功能描述，设计要用的数据表，因为这些表之间相互关联，存储着系统的相关数据，所以在设计数据表的过程中，要求设计一个表只存储一个实体或对象的相关信息，不同实体存储在不同的表中，本系统需要设计的数据表图如下：

学生基本信息表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 可否为空 | 主键 |
| studentId | 学号 | Char(10) | 否 | 是 |
| studentName | 姓名 | Varchar(50) | 否 |  |

宿舍楼信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 可否为空 | 主键 |
| Build | 楼栋号 | Char(10) | 否 | 是 |
| Count | 入住人数 | Varchar(50) | 否 |  |

宿舍信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 可否为空 | 主键 | 外键 |
| build | 楼栋号 | Varchar(2) | 否 |  |  |
| floor | 楼层号 | Char(1) | 否 |  |  |
| room | 宿舍号 | Char(3) | 否 | 是 |  |
| studentId | 学号 | Char(10) | 否 |  | 是 |
| studentName | 姓名 | Varchar(50) | 否 |  |  |

管理员信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 可否为空 | 主键 | 外键 |
| Name | 姓名 | Varchar(50) | 否 |  |  |
| phone | 电话 | Char(11) | 否 | 是 |  |
| pwd | 登录密码 | Varchar(16) | 否 |  |  |
| build | 所管理的楼栋号 | Varchar(2) | 否 |  | 是 |

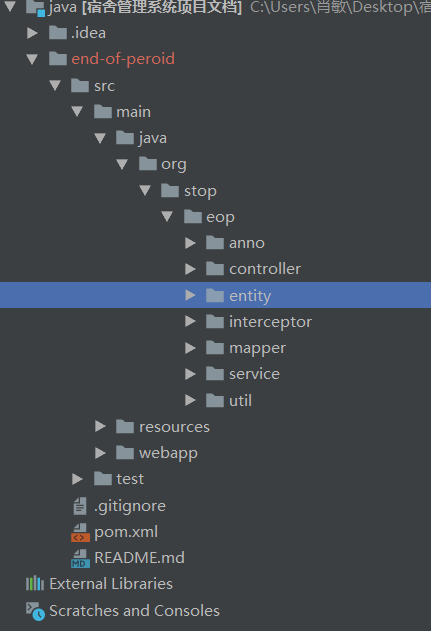
入住学生信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 可否为空 | 主键 | 外键 |
| build | 楼栋号 | Varchar(2) | 否 |  |  |
| floor | 楼层号 | Char(1) | 否 |  |  |
| room | 宿舍号 | Char(3) | 否 |  | 是 |
| studentId | 学号 | Char(10) | 否 | 是 |  |
| studentName | 姓名 | Varchar(50) | 否 |  |  |

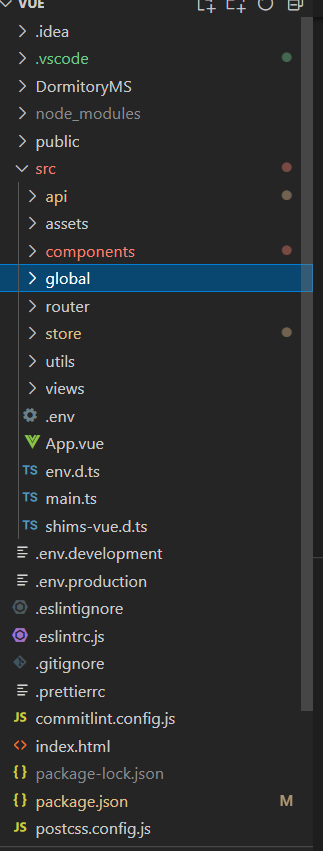
第4章 系统主要功能模块详细设计

4.1整个项目文件结构

Java文件结构如下



Vue文件结构如下



第5章 系统测试

5.1测试原则  
 软件测试的基本原则是站在用户的角度，对产品进行全面测试，尽早、尽可能多地发现问题，并负责跟踪和分析产品中的问题，具体测试原则有如下几点:

第一，测试应该尽早进行，最好在需求阶段就开始介入，因为最严重的错误不外乎是系统不能满足用户的需求。  
 第二，设计测试用例时应考虑到合法的输入和不合法的输入以及各种边界条件，特殊情况下不要制造极端状态和意外状态。  
 第三，制定严格的测试计划。- -定要制定测试计划，并且要有指导性。测试时间安排

尽量宽松，不要希望在极短的时间内完成--个高水平的测试。  
 第四，妥善保存测试计划、测试用例、出错统计和最终分析报告，为维护提供方便。  
5.2测试用例  
 根据系统的设计流程，一个一个模块地测试，要测试的模块如下:登录模块,忘记密码模块，住宿学生信息查询模块，宿舍成员管理模块。具体的测试用例，预期结果如下详细介绍:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例 | 场景 | 测试用例 | 预期结果 | 实际结果 |
| TC1 | 用户登录 | 打开系统，输入手机号，登录密码，验证码，点击登录按钮 | 判断管理员的手机号是否存在，登录密码，验证码是否正确，正确则登录成功，错误则提示错误信息 | 同预测 |
| TC2 | 退出登录 | 点击“退出登录”后在弹出的提示框内选择“确定”或“取消” | 点击“确定”则跳转至登录界面，点击“取消”则返回原界面 | 同预测 |
| TC3 | 修改入住学生信息 | 选定楼栋，楼层，宿舍，查看宿舍成员信息，点击操作栏中的“修改“按钮后跳出入住学生信息弹窗，进行修改后点击“取消“或“确认“按钮 | 点击“确认”后弹出修改成功信息，再次查看该宿舍成员信息已发生变化，点击“取消”按钮返回原界面 | 同预测 |
| TC4 | 删除入住学生信息 | 选定楼栋，楼层，宿舍，查看宿舍成员信息，点击操作栏中的“删除“按钮后跳出警告弹窗，点击“取消“或“确认“按钮 |  |  |

6.3测试小结  
 本章主要测试了宿舍管理系统的可行性，并且介绍了专门针对本系统的测试方式，描述了一些简单的测试用例。此外，还介绍了对系统进行测试的预期结果和最后的实际结果。从总体上来说，整个系统的设计都比较符合要求，圆满地完成了开题报告中所提出的设计要求，基本.上达到了这次系统开发的预期效果。

第6章.结论  
 课程设计是大学本科教育最后一个重要的实践环节，重培养学生综合运用本学科的基本理论、专业知识和基本技能，提高分析与解决实际问题的能力。在使用java开发这个系统的过程中，的问题，这时就需要多在网上或书上找资料，我深刻地认识到需求分析在整个软件开发流程中的重要性，没有做好需求分析就开发系统，问题发现地越晚，所付出的的代价就越大。同时，我也感受到了在整个软件开发的流程中，可能会碰到--些难以预料多向老师和同学寻求帮助，不断思考，寻找各种解决方案。

通过这个项目的开发，对企业级的项目开发流程有了很好的理解,对Spring。Spring MVC, Mybatis 三大框架的整合开发有了很好的开发经验，对以后工作打下了坚实的基础。