**小组编号：** **B08**

**项目名称： 校园宿舍管理系统**

**小组负责人： 罗正新**

**小组成员： 罗正新**

**黄智豪**

**李足均**

**刘礼嘉**

《JavaWeb编程》

期末小组项目文档

2021-2022学年第一学期

目录

[1 需求分析 1](#_Toc90913374)

[1.1 需求定义 1](#_Toc90913375)

[1.1.1 需求概述 1](#_Toc90913376)

[1.1.2 功能需求 2](#_Toc90913377)

[1.1.3 非功能需求 7](#_Toc90913378)

[1.2 用例图文档 8](#_Toc90913379)

[1.2.1 图形文档 8](#_Toc90913380)

[1.2.2 文字说明 10](#_Toc90913381)

[2 系统分析 15](#_Toc90913382)

[2.1 类图文档 15](#_Toc90913383)

[2.1.1 图形文档 15](#_Toc90913384)

[2.1.2 文字说明 17](#_Toc90913385)

[2.2 顺序图文档 22](#_Toc90913386)

[2.2.1 图形文档 22](#_Toc90913387)

[2.2.2 文字说明 23](#_Toc90913388)

[2.3 包图文档 25](#_Toc90913389)

[2.3.1 包图 25](#_Toc90913390)

[2.3.2 包图中的包描述 25](#_Toc90913391)

[3 系统设计 27](#_Toc90913392)

[3.1 系统体系结构设计 27](#_Toc90913393)

[3.1.1 程序（模块）划分 27](#_Toc90913394)

[3.1.2 程序（模块）层次结构关系 28](#_Toc90913395)

[3.2 数据库设计 30](#_Toc90913396)

[3.3 用户界面设计 33](#_Toc90913397)

[3.3.1 登录模块 33](#_Toc90913398)

[3.3.2 系统管理员模块 33](#_Toc90913399)

[3.3.3 楼层管理员模块 37](#_Toc90913400)

[3.3.4 学生管理模块 40](#_Toc90913401)

[4 测试分析报告 41](#_Toc90913402)

[4.1 引言 41](#_Toc90913403)

[4.1.1 编写目的 41](#_Toc90913404)

[4.1.2 项目背景 41](#_Toc90913405)

[4.1.3 定义 41](#_Toc90913406)

[4.1.1 参考资料 41](#_Toc90913407)

[4.2 测试计划执行情况 42](#_Toc90913408)

[4.2.1 测试项目 42](#_Toc90913409)

[4.2.2 测试机构与人员 43](#_Toc90913410)

[4.2.3 测试结果 44](#_Toc90913411)

[4.3 软件需求测试结论 55](#_Toc90913412)

[4.4 评价 55](#_Toc90913413)

[4.4.1 软件能力 55](#_Toc90913414)

[4.4.2 缺陷与限制 55](#_Toc90913415)

[4.4.3 建议 56](#_Toc90913416)

[4.4.4 测试结论 56](#_Toc90913417)

# 需求分析

本章主要包含需求定义、用例图文档。

## 需求定义

需求定义包括学生宿舍信息管理系统功能需求和非功能需求两个方面，前者从用户角度明确软件系统的业务流程及系统清单进行概述；后者对系统的总体特性相关的需求进行概述。

### 需求概述

传统的学生公寓管理倾向于人工现场登记，资料多为纸质形式，不便于管理和查看。随着信息科技的快速发展，信息技术广泛应用对于各行业，在这样的情形之下，相应的学生公寓管理系统应运而生。宿舍管理员管理的对象多且复杂，学生公寓管理系统的开发，能使其管理的各种数据资源得到良好的继承和共享，减轻宿舍管理员的工作量，进一步推动宿舍管理便捷式的发展。在与传统的管理模式对比，不仅节约人力物力，还节省纸张保护环境，同时也促进了宿舍管理员的事务管理信息化发展。

#### 目标

学生公寓管理系统是为了快速、高效地对学生公寓进行管理,为处理学生宿舍相关事务而出现的系统。

主要达到以下目标：

1. 人机交互，界面友好，查询方便，快速，数据存储安全。
2. 学生宿舍分配有效，准确，流程清晰。
3. 查询学生宿舍信息方便，可以多条件查询。
4. 较高的运行效率，较强的可维护性。

#### 运行环境

1. 硬件环境

|  |  |  |
| --- | --- | --- |
| 服务器 | 最低配置 | 推荐配置 |
| 应用和数据库服务器 | 1CPU：P4 2.0G | 1CPU：P4 2.8G |
| Mem：512M | Mem：2G |
| HD：40G | HD：120G |

**表1—1硬件环境**

1. 软件环境

|  |  |  |  |
| --- | --- | --- | --- |
|  | 名称 | 版本 | 语种 |
| 操作系统 | Windows 10 | 2019 | 简体中文 |
| 数据库平台 | MySQL、Navicat Premium | 8.0 | 简体中文 |
| 应用平台 | Intellij IDEA | 2019.6 | 简体中文 |
| 部署平台 | Tomcat | 9.0 | 简体中文 |
| 客户端软件 | 网页浏览器 | 2019.3 | 简体中文 |

#### 表1—2软件环境

#### 3 用户特点

用户主要分为三类：系统管理员、楼层管理员以及学生。学校学生数量较多，导致系统使用人数较多。由于本系统所拥有权限最多的用户是系统管理员，考虑到使用人员对电子设备的熟悉程度以及办事效率，本系统设计的较为简洁。

#### 4 关键点

系统要为不同身份的用户分配不同的操作权限。

#### 5 约束条件

开发软件必须使用[MySQL数据库](https://baike.baidu.com/item/Oracle%E6%95%B0%E6%8D%AE%E5%BA%93)系统完成数据管理功能，运行时必须基于Windows环境。

### 功能需求

学生公寓管理系统采用B/S结构，使用JSP技术、Servlet技术、Java语言和My SQL数据库开发。涵盖了前后端开发和数据库的创建三大内容。该系统主要有两大用户：系统管理员、楼层管理员以及学生。

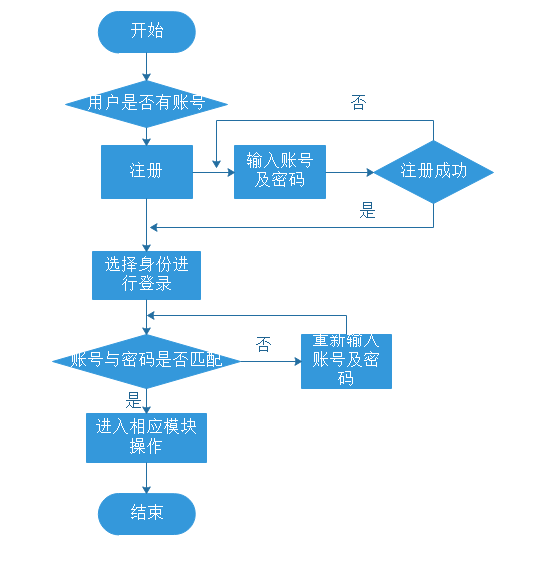
系统管理员：管理员侧重对信息的管理，包括对楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录、迁出记录。同时系统管理员也可修改自身密码。并在所有操作结束后，退出系统。

楼层管理员：楼层管理员主要是为了减轻系统管理员的压力以及及时对学生的某些合理请求做出回应。楼层管理员可以进行学生管理以及学生缺寝记录的操作。同时楼层管理员也可修改自身密码。并在所有操作结束后，退出系统。

学生：学生仅能查看自身有无缺寝记录。同时学生也可修改自身密码。并在操作结束后，退出系统。

#### 业务流程

（1）系统总体流程图：



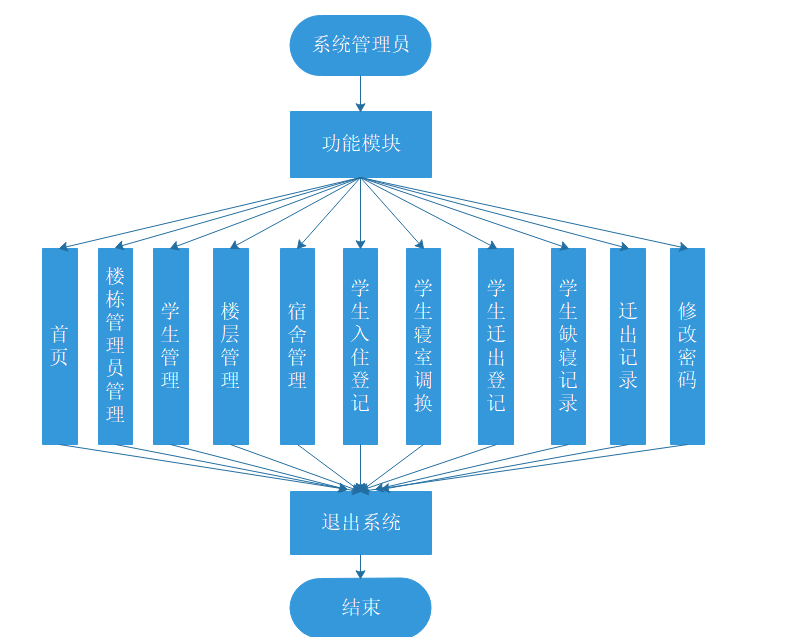
**图1.1系统总体流程图**

相关说明：当用户使用本系统时，先选择身份并输入账号与密码，系统验证账号与密码是否匹配，如果匹配，则进入相应模块操作，如果不匹配，系统提醒用户重新输入账号及密码，直到匹配成功。

注册部分仅对系统管理员开放，其余身份的用户，由系统管理员将相关信息导入进本系统即可。也就说其余身份的用户不需要自主进行注册这一操作。

下列对不同身份的用户进行相应操作的模块做进一步说明。

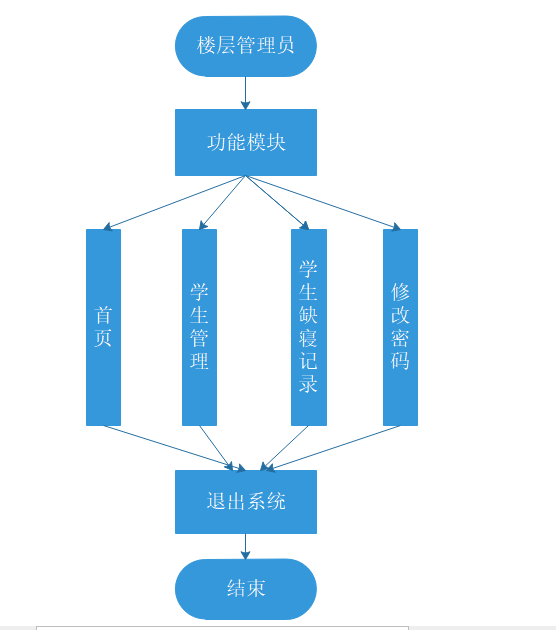
（2）系统管理员：



**图1.2系统管理员流程图**

相关说明：系统管理员进入系统后，对功能模块内的模块进行操作：楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录、迁出记录、修改密码以及在所有操作结束后退出系统。

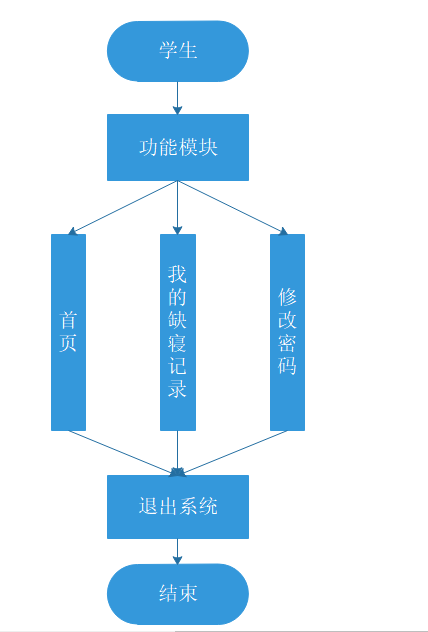
（3）楼层管理员：



**图1.3楼层管理员流程图**

相关说明：楼层管理员进入系统后，对功能模块内的模块进行操作：首页、学生管理、学生缺寝记录、修改密码以及在所有操作结束后退出系统。

（4）学生：

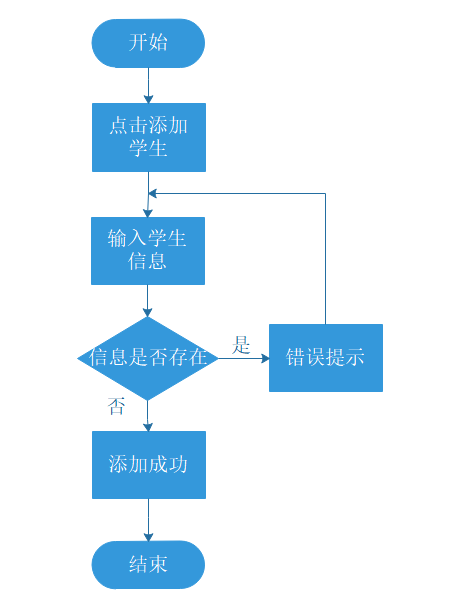


**图1.4学生流程图**

相关说明：学生进入系统后，对功能模块内的模块进行操作：首页、我的缺寝记录、修改密码以及在所有操作结束后退出系统。

学生子模块，目前涉及到的功能较少，后续开发中会逐步完善。

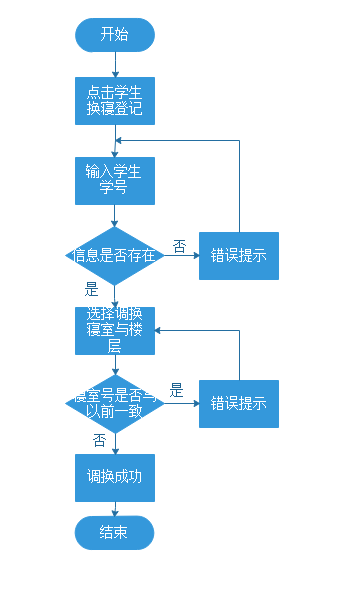
1. 学生管理流程（仅涉及添加学生）：



**图1.5学生管理流程图**

相关说明：学生管理是系统管理员与楼层管理员的一个子模块。管理员首先点击添加学生然后输入该名学生的信息，如果该名学生的信息在数据库已经存在，则系统发出错误提示，提醒管理员重新输入用户信息，如果学生信息不存在那么系统成功添加该名学生信息。

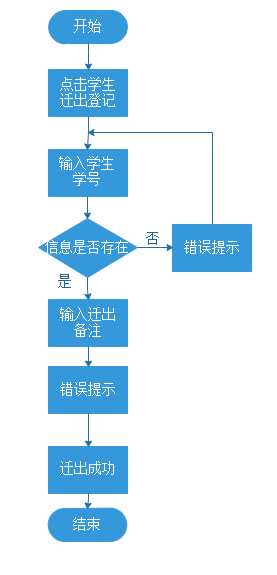
1. 学生寝室调换流程



**图1.6学生寝室调换流程**

相关说明：学生寝室调换是系统管理员的一个子模块。系统管理员首先选择学生寝室调换功能，输入要调换寝室的学生学号，系统检测该名学生信息是否存在，如果不存在，则发出提示，系统管理员重新输入，如果存在，则选择该名学生要调换的寝室楼层与寝室号，如果寝室号与以前一致，那么调换失败，提示管理员重新操作，若寝室号不一致，则该名学生寝室调换成功。

1. 学生迁出登记



**图1.7学生迁出登记流程图**

相关说明：学生迁出登记是系统管理员的一个子模块。系统管理员首先选择学生迁出登记功能，输入要迁出的学生学号，系统检测该名学生信息是否存在，如果不存在，则发出提示，系统管理员重新输入，如果存在，则输入学生迁出备注，点击确认迁出。该名学生成功迁出。

#### 事件清单

|  |  |  |
| --- | --- | --- |
| 软件名称 | 系统模块名称 | 描述 |
| 学生宿舍管理系统 | 注册登录子系统 | 1. 注册功能仅对系统管理员开放； 2. 其余用户的信息由系统管理员导入； 3. 用户在登陆时要选择自身相应的身份，输入正确的账号及密码。 |
| 系统管理员子模块 | 1. 系统管理员拥有本系统的最高权限，可以对本系统设计到的所有功能操作进行操作； 2. 楼栋管理员管理，系统管理员可以增加、删除楼栋管理员的信息； 3. 学生管理，系统管理员可以在此模块查看已入住和未入住学生的基本信息并且修改他们的信息； 4. 楼层管理，系统管理员可以设置各个寝室楼层的管理员，并修改他们的信息； 5. 宿舍管理，可以查看所有寝室的信息，如寝室能容纳几人等，也可对此部分内容进行修改； 6. 学生入住登记，记录学生的学号以及对应的楼层，寝室号等； 7. 学生寝室调换，对学生调换寝室进行操作； 8. 学生迁出登记，登记迁出学生的学号； 9. 学生缺寝记录，登记缺寝学生的楼层房间号以及学号； 10. 迁出记录，可以查看所有迁出学生的记录； 11. 修改密码，可以修改自身密码； 12. 完成所有操作后，退出系统。 |
| 楼层管理员模块 | 1. 楼层管理员拥有本系统的较多权限，可以协助系统管理员更好地完成工作，提高工作效率； 2. 学生管理，系统管理员可以在此模块查看已入住和未入住学生的基本信息并且修改他们的信息； 3. 学生缺寝记录，登记缺寝学生的楼层房间号以及学号 4. 修改密码，可以修改自身密码 5. 完成所有操作后，退出系统。 |
| 学生模块 | 1. 学生功能目前开发较少； 2. 目前仅能查看自身有无缺寝记录； 3. 完成操作后，退出系统。 |

### 非功能需求

要求系统界面简洁、友好，操作简单，系统运行稳定，有一定的容错能力，不同角色有不同的权限保证系统数据安全。

具体为： ①保密性，该系统录入的学生宿舍信息、宿管的信息会进行严格的保密。

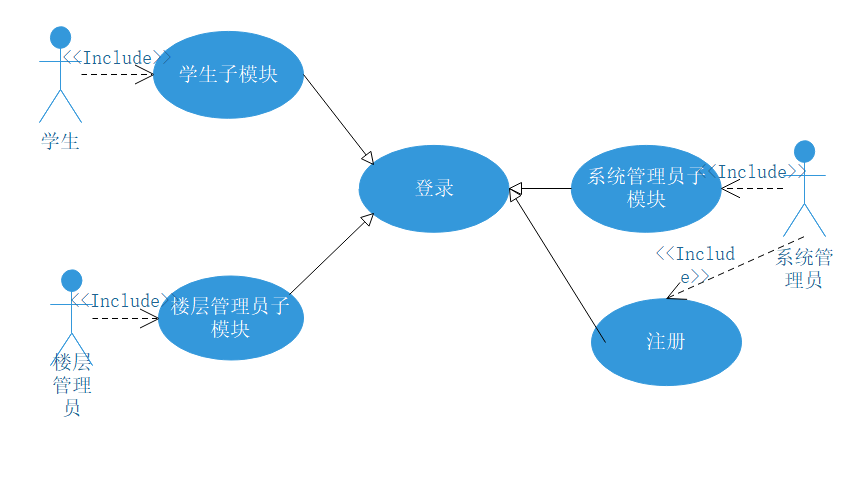
②操作简捷，学生宿舍管理系统可以方便用户的使用，容易上手操作，这样可以提高使用效率。

③可扩展性与易维护性 在开发过程中，应设计好接口，以备后续开发人员对本系统的功能进行维护和完善。

## 用例图文档

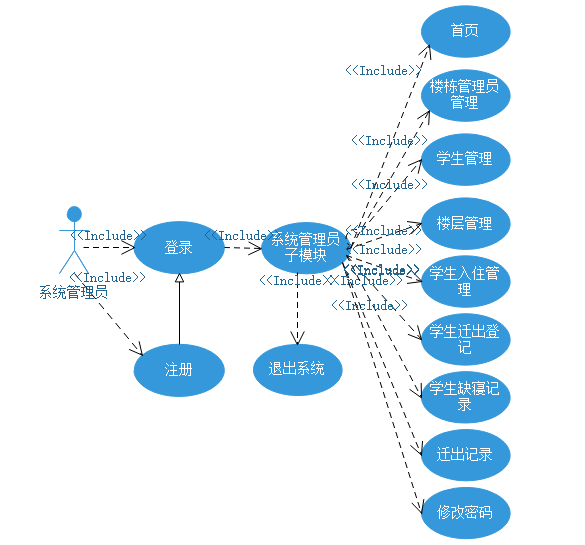
### 图形文档

系统总体用例图：

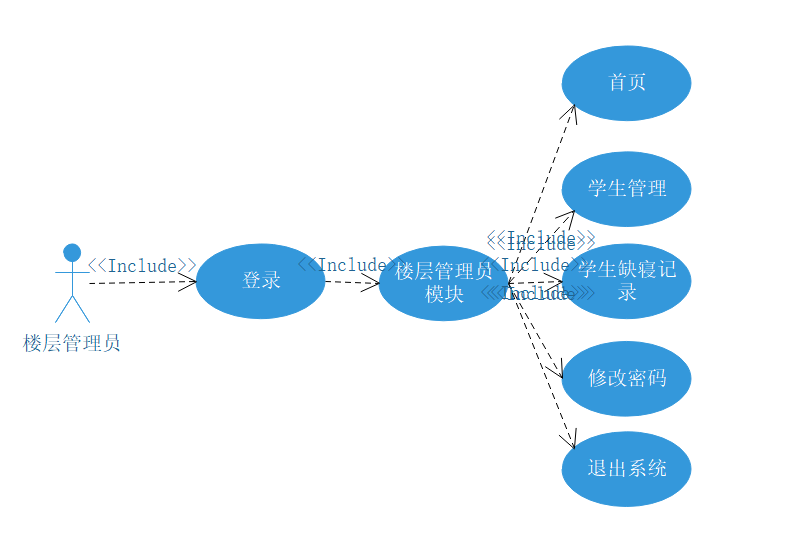


下面给出不同用户的用例图，该部分的用例图，并未对每个用例进行详细刻画。详细的用例请参照用例描述。

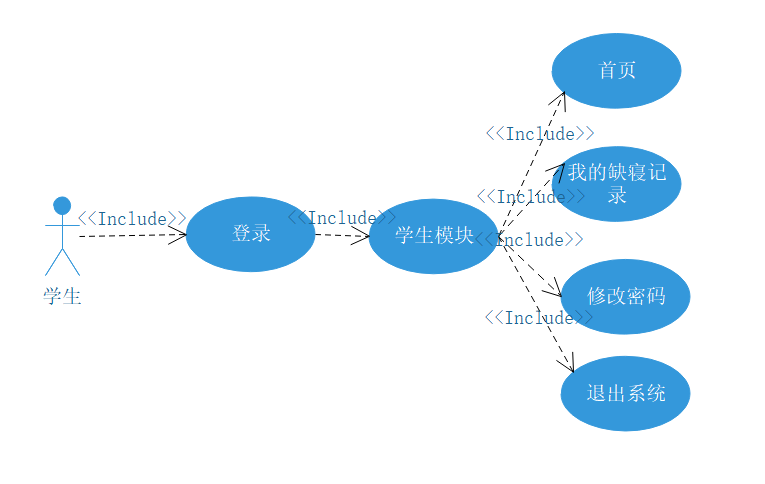
系统管理员用例图：



楼层管理员用例图：



学生用例图：



### 文字说明

#### 用例图综述

系统总体用例图描述：

系统管理员子模块、学生子模块以及楼层管理员子模块与登录是泛化关系，及系统管理员、楼层管理员、学生都需要登录这一操作进入到本系统的相应子模块中。用户进入到相应模块后，可对其模块存在的功能进行操作，它们是包含关系。

系统管理员用例图描述：

系统管理员在成功登录后，即可以进入到系统管理员子模块中，系统管理员可以对其存在的模块功能进行操作，即系统管理员有权限进行以下操作：楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录、迁出记录、修改密码以及退出系统。系统管理员子模块与模块中存在的行为是包含关系。

#### 参与者描述

对系统总体用例图进行参与者描述（排序顺序按照用户权限高低排序）：

系统管理员：1.拥有本系统的最高权限，可以对本系统所涉及到的所有功能进行操作；

1. 可以对如下模块进行管理：楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录、迁出记录、修改密码以及退出系统；
2. 系统管理员能够进行注册这一操作；
3. 其余用户账号及密码由系统管理员统一导入。

楼层管理员：1.拥有本系统的较高权限；

1. 可以对以下模块进行管理：学生管理、学生缺寝记录、修改密码以及退出系统；
2. 其余功能待完善。

学生：1.仅能查看自身缺寝记录，修改密码以及退出系统

2.其余功能待完善。

#### 用例描述

这里对系统管理员模块用例图进行用例描述（仅对核心功能进行描述）

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【001】登录 | |
| **概述** | 系统管理员进行登录操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户选择身份  3.用户点击系统管理员  4.用户输入账号及密码 | 2.系统显示：系统管理员、楼层管理员、学生  5.系统验证用户账号及密码是否正确，若正确则成功登录，不正确，弹出提示框，重新输入 |
| **分支流程** | **no** | |

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【002】楼栋管理员管理 | |
| **概述** | 系统管理员对楼栋管理人员模块进行操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户点击楼栋管理员管理  3.系统管理员点击添加楼栋管理员  5.系统管理员在该界面输入要添加为管理员的人的对应信息  6.系统管理员点击添加楼栋管理员确认按钮  8.用户还可以选择查询条件，输入内容 | 2.系统跳转到楼栋管理员管理界面  4.系统跳转到相应界面  7.系统提示添加成功，并跳转回楼栋管理员管理界面  9.系统显示符合查询条件的人员 |
| **分支流程** | **no** | |

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【003】学生管理 | |
| **概述** | 系统管理员对学生管理模块进行操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户点击学生管理  3.系统管理员点击添加学生  5.系统管理员在该界面输入要添加的学生的对应信息  6.系统管理员点击添加学生确认按钮  8.用户还可以选择查询条件，输入内容 | 2.系统跳转到学生管理界面  4.系统跳转到相应界面  7.系统提示添加成功，并跳转回学生管理界面  9.系统显示符合查询条件的人员 |
| **分支流程** | **no** | |

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【004】学生入住登记 | |
| **概述** | 系统管理员对学生入住模块进行操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户点击学生入住登记  3.系统管理员选择楼层  5.点击该名学生对应楼层数  6.系统管理员选择楼层  8.点击该名学生对应寝室号  9.输入该名学生学号  10.点击确定入住 | 2.系统跳转到学生入住登记界面  4.系统弹出已存在的楼层  7.系统弹出已存在的楼层  11.系统弹出提示，入住成功。若学生信息错误，系统也会弹出提示框提示 |
| **分支流程** | **no** | |

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【005】学生寝室调换 | |
| **概述** | 系统管理员对学生寝室调换模块进行操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户点击学生寝室调换  3.系统管理员在“请输入要调换寝室学生的学号”输入框输入请求调换寝室的学生的学号  4.点击确定按钮  6.系统管理员选择调换到楼层  8.点击该名学生申请调换的寝室楼层  9.系统管理员选择调换到寝室  11.点击该名学生申请调换的寝室号  12.点击确认调换按钮 | 2.系统跳转到学生寝室调换界面  5.系统跳转到学生寝室调换的具体界面  7.系统弹出已存在的楼层  10.系统弹出对应楼层的寝室  13.系统弹出提示，调换成功。 |
| **分支流程** | **no** | |

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【006】学生缺寝记录 | |
| **概述** | 系统管理员对学生缺寝记录模块进行操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户点击学生缺寝记录  3.系统管理员选择楼层  5.点击该名学生的寝室楼层  6.系统管理员选择寝室  8.点击该名学生的寝室号  9.在“学生学号”的输入框内输入学生学号  10.点击开始查询 | 2.系统跳转到学生缺寝记录界面  4.系统弹出所有楼层  7.系统弹出对应楼层的寝室  11.系统跳转到相关界面 |
| **分支流程** | **no** | |

|  |  |  |
| --- | --- | --- |
| **用例名称** | 【007】学生迁出登记 | |
| **概述** | 系统管理员对学生迁出登记模块进行操作 | |
| **参与者** | 系统管理员 | |
| **基本流程** | **Actor** | **System** |
| 1.用户点击学生迁出登记  3.在“学生学号”的输入框内输入学生学号  4.点击确定按钮  6.在“迁出备注”的输入框内输入备注  7.点击确定迁出按钮 | 2.系统跳转到学生迁出登记界面  5.系统弹出含有该名学生信息的迁出登记界面  8.系统弹出成功迁出窗口 |
| **分支流程** | **no** | |

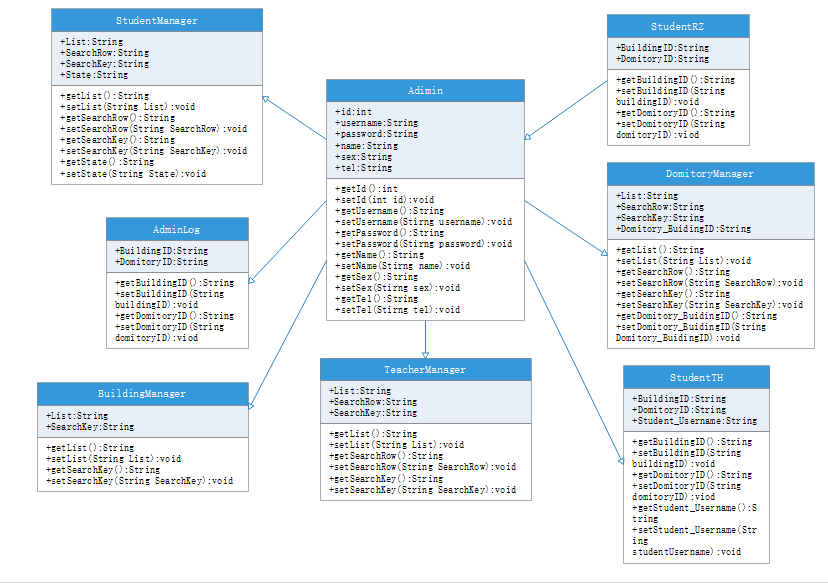
# 系统分析

本章主要包含类图文档、顺序图文档、包图文档。

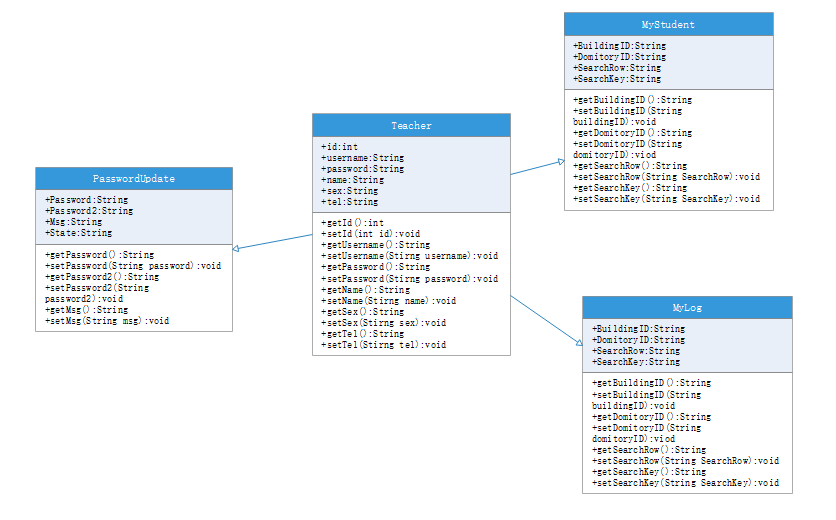
## 类图文档

### 图形文档

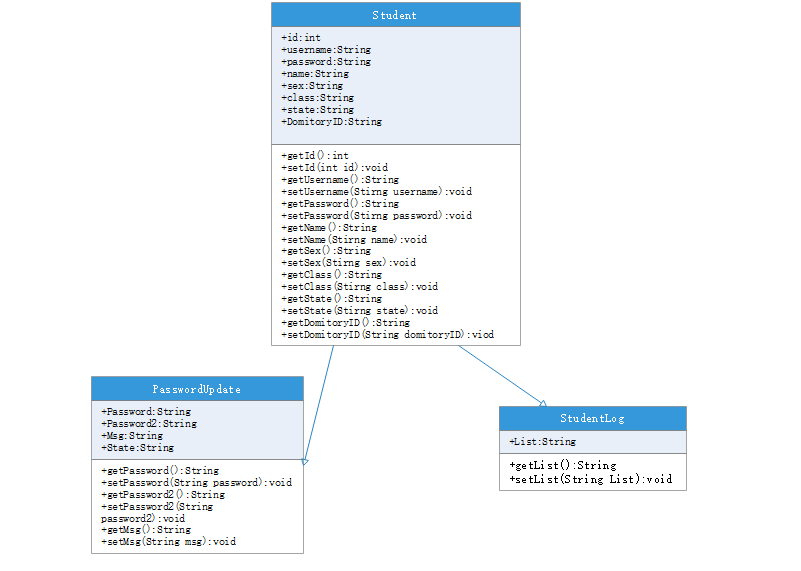
系统管理员类图：



楼层管理员类图：



学生类图：



### 文字说明

#### 类图综述

系统管理员类图：

系统管理员类图主要绘制了系统管理员对整个系统的各个功能之间的关系与联系。系统管理员拥有最大权限，对整个系统各个部分都能进行管理；主要通过后台的数据对整个系统进行维护和管理。其主要管理的功能有：楼栋管理员的管理、入住学生的管理、各楼层的管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记以及对学生的缺勤进行记录；其余功能有待完善。

楼层管理员类图：

楼层管理员类图主要提供了各个楼层相对应的管理员在整个系统中能够处理和管理的相关功能。楼层管理员主要管理各个楼层的学生信息、学生安全以及缺寝信息登记等等，把所有信息汇总最后交给系统管理员进行信息整改。

学生类图：

学生类图的功能相对比较少，学生只能够登录系统查看自己的相关信息，当天的缺寝记录。如果发现问题，可以通过向楼层管理员发出申请让楼层管理员告知系统管理员进行核对或修改；学生也可以直接向系统管理员发出申请进行数据的核对或修改。

#### 类描述

用户类：

用户类是一个抽象类，是系统管理员、楼层管理员以及学生类的父类。用于描述可登录用户的个人信息以及登陆信息。具有“用户名”、“账号”、“密码”属性。在该类中，设有三个操作：“登录系统”、“修改密码”和“退出系统”。

Admin(系统管理员)类：

Admin类继承自用户类，在此基础上设立了额外的属性：“管理员姓名”、“性别”和“电话号码”。系统管理员进行整个系统数据的管理和维护，与之关联的功能有：“楼层管理员的管理”、“入住学生的管理”、“各楼层的管理”、“宿舍管理”、“学生入住登记”、“学生寝室调换”、“学生迁出登记”以及“对学生的缺寝进行记录”。

Teacher(楼层管理员)类：

Teacher类继承自用户类，增添了楼层管理员“姓名”、“性别”和“电话号码”三个属性。具有“学生管理”、“学生缺勤记录”和“修改密码”的功能。

Student(学生）类：

Student类继承自用户类，除了含有用户类的属性之外，还包含“学生姓名”、“性别”、“班级”、“登记日期”以及“所在宿舍号”的属性。该类具有“查看个人缺寝记录”和“修改密码”的功能。

StudentManager类：

该类继承自Student类，增添了属性“入住状态”，主要用于系统管理员对入住学生进行管理。具有“查询”、“修改”和“删除”学生信息的操作。

TeacherManager类：

该类继承自Teacher类，主要用于系统管理员对各楼层管理员进行统一管理。具有“查询”、“修改”和“删除”楼层管理员信息的操作。

BuildingManager类：

该类具有属性“楼层号”和“楼层简介”，主要用于系统管理员管理各个楼层。具有“添加楼层”、“修改”和“删除”信息的功能。

DomitoryManager类：

该类具有属性“寝室类型”、“寝室人数”、“宿舍楼层号”和“宿舍电话”，主要用于系统管理员进行宿舍的分配和添加。具有“查询”，“删除”和“修改”宿舍信息的功能。

StudentRZ类：

该类具有属性“楼层号”和“宿舍号”，主要用于系统管理员管理学生入住情况。具有“添加学生入住”的功能。该类与“StudentTH类”和“StudentQC类”具有聚合关系。

StudentTH类：

该类继承自Student类，增添了属性“楼层号”和“宿舍号”，主要用于系统管理员对学生进行寝室调换。具有“调换寝室”的功能。

StudentQC类：

该类继承自Student类，主要用于系统管理员对即将毕业的学生进行迁出寝室的信息管理。具有“学生迁出”的功能，并能添加“迁出备注”。

AdminLog类：

该类继承自Student类，增添了属性“日期”和“备注”，主要用于系统管理员查看当天的学生的缺寝记录。具有“查询”的功能。

OutList类：

该类继承自Student类，主要用于系统管理员查看学生的迁出记录。具有“查询”的功能。

PasswordUpdate类：

该类具有属性“新密码”和“修改日期”，主要用于用户对自己的账号进行密码的修改。具有“修改”的功能。

MyStudent类：

该类继承自Student类，增添了属性“寝室类型”、“寝室人数”和“宿舍电话”，主要用户楼层管理员对各个楼层的学生进行缺寝登记。具有“缺寝登记”的功能并能添加“缺寝备注”。

#### 关联描述

管理楼层管理员

a) 关联名称：管理楼层管理员

b) 关联的类型：多元关联

c) 关联所连接的类:Admin, TeacherManager

d) 多重性:一个系统管理员可以对多个楼层管理员的信息进行管理；多个系统管理员也能对一个楼层管理员的信息进行管理

管理学生

a) 关联名称：管理学生

b) 关联的类型：多元关联

c) 关联所连接的类:Admin, StudentManager

d) 多重性:一个系统管理员可以对多个学生的信息进行管理；多个系统管理员也能对一个学生的信息进行管理

管理楼层

a) 关联名称：管理楼层

b) 关联的类型：多元关联

c) 关联所连接的类:Admin, BuildingManager

d) 多重性:一个系统管理员可以对多个楼层进行管理；多个系统管理员也能对一个楼层进行管理

管理宿舍

a) 关联名称：管理宿舍

b) 关联的类型：多元关联

c) 关联所连接的类:Admin, DomitoryManager

d) 多重性:一个系统管理员可以对多个宿舍进行管理；多个系统管理员也能对一个宿舍进行管理

登记学生入住信息

a) 关联名称：登记学生入住信息

b) 关联的类型：一元关联

c) 关联所连接的类:Admin, StudentRZ

d) 多重性:一个系统管理员可以对多个学生的入住信息进行登记

调换学生寝室

a) 关联名称：调换学生寝室

b) 关联的类型：一元关联

c) 关联所连接的类:Admin, StudentTH

d) 多重性:一个系统管理员可以进行多个学生的寝室调换

登记学生迁出信息

a) 关联名称：登记学生迁出信息

b) 关联的类型：一元关联

c) 关联所连接的类:Admin, StudentQC

d) 多重性:一个系统管理员可以进行多个学生的迁出信息登记

管理员查看学生缺寝信息

a) 关联名称：管理员查看学生缺寝信息

b) 关联的类型：多元关联

c) 关联所连接的类:Admin,Teacher,AdminLog,MyStudent

d) 多重性:所有管理员都可以查看所有的学生缺寝信息

学生查看个人缺寝信息

a) 关联名称：学生查看个人缺寝信息

b) 关联的类型：组合

c) 关联所连接的类:Student,StudentLog

d) 多重性:每个学生只能查看自己的缺寝信息

查看学生迁出信息

a) 关联名称：查看学生迁出信息

b) 关联的类型：多元关联

c) 关联所连接的类:Admin,OutList

d) 多重性：所有系统管理员都能查看所有学生的迁出信息

修改密码

a) 关联名称：修改密码

b) 关联的类型：一元关联

c) 关联所连接的类:Admin,Teacher,Student,PasswordUpdate

d) 多重性：每个用户只能修改自己账号的密码；每个账号的密码也只能被对应的一个用户进行修改

#### 泛化描述

类图中的每一个泛化都有如下的描述：

学生信息管理

a) 泛化关系中的父类：Student；

b) 泛化关系中的子类：StudentManager；

楼层管理员信息管理

a) 泛化关系中的父类：Teacher；

b) 泛化关系中的子类：TeacherManager；

学生寝室调换

a) 泛化关系中的父类：Student；

b) 泛化关系中的子类：StudentTH；

学生迁出

a) 泛化关系中的父类：Student；

b) 泛化关系中的子类：StudentQC；

查看迁出记录

a) 泛化关系中的父类：Student；

b) 泛化关系中的子类：OutList；

查看缺寝记录

a) 泛化关系中的父类：Student；

b) 泛化关系中的子类：AdminLog；

缺寝登记

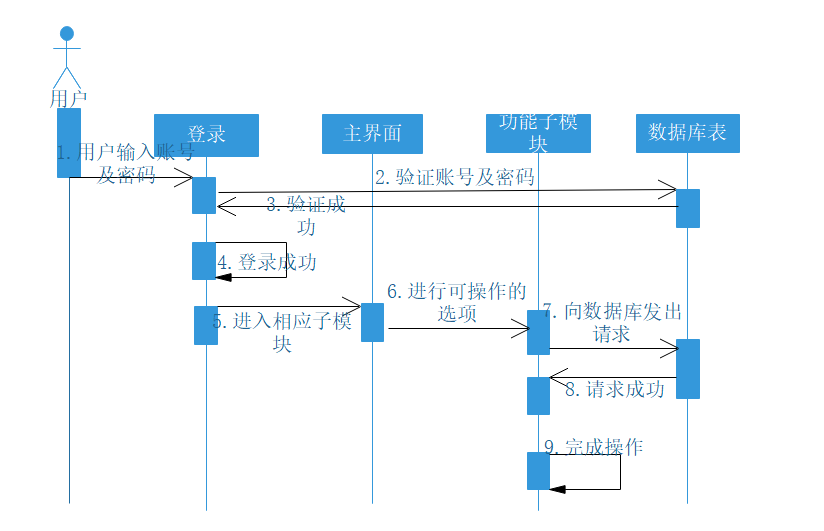
a) 泛化关系中的父类：Student；

b) 泛化关系中的子类：MyStudent；

## 顺序图文档

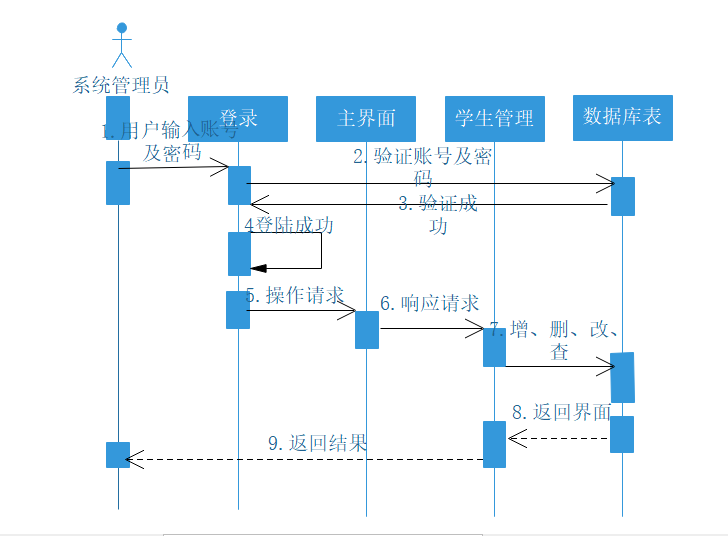
### 图形文档

（1）系统总体顺序图



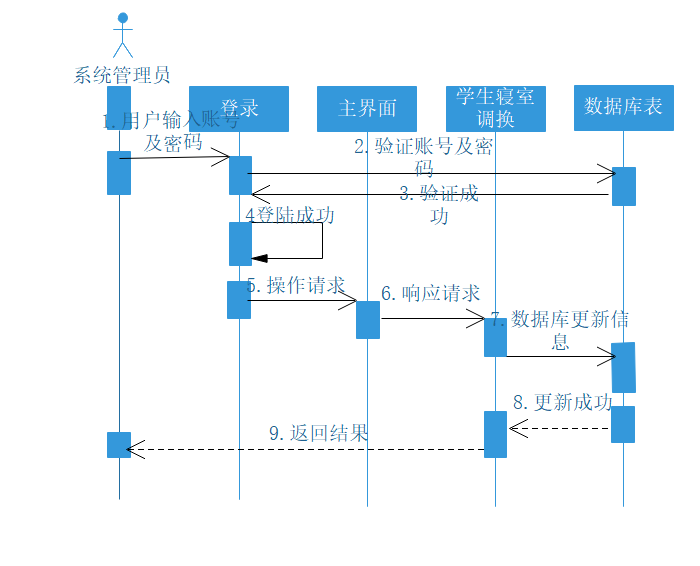
**图2.2.1系统总体顺序图**

（2）系统管理员学生管理顺序图：



**图2.2.2学生管理顺序图**

1. 学生寝室调换顺序图：



**图2.2.3学生寝室调换顺序图**

### 文字说明

#### 顺序图综述

（1）系统总体顺序图是对整个系统的顺序进行了概述。

1. 用户在输入账户和密码之后，点击登录
2. 系统验证账号及密码是否与数据库内的数据匹配
3. 如果匹配则验证成功
4. 用户成功登录
5. 不同身份的用户进入到相应的功能界面中
6. 在功能模块中进行操作
7. 用户进行的操作如果设计到数据的修改，则向数据库发出请求
8. 数据库响应用户请求
9. 用户完成操作
10. 系统管理员学生管理顺序图描述：

1.用户在输入账户和密码之后，点击登录

2.系统验证账号及密码是否与数据库内的数据匹配

3.如果匹配则验证成功

4.用户成功登录

5.管理员请求操作

6.系统响应用户请求

7.数据库更改信息

8.返回学生管理界面

9.向用户返回结果

（3）系统管理员学生寝室调换顺序图描述

1.用户在输入账户和密码之后，点击登录

2.系统验证账号及密码是否与数据库内的数据匹配

3.如果匹配则验证成功

4.用户成功登录

5.管理员请求操作

6.系统响应用户请求

7.数据库更改信息

8.返回学生寝室调换界面

9.向用户返回结果

#### 顺序图中的对象与参与者描述

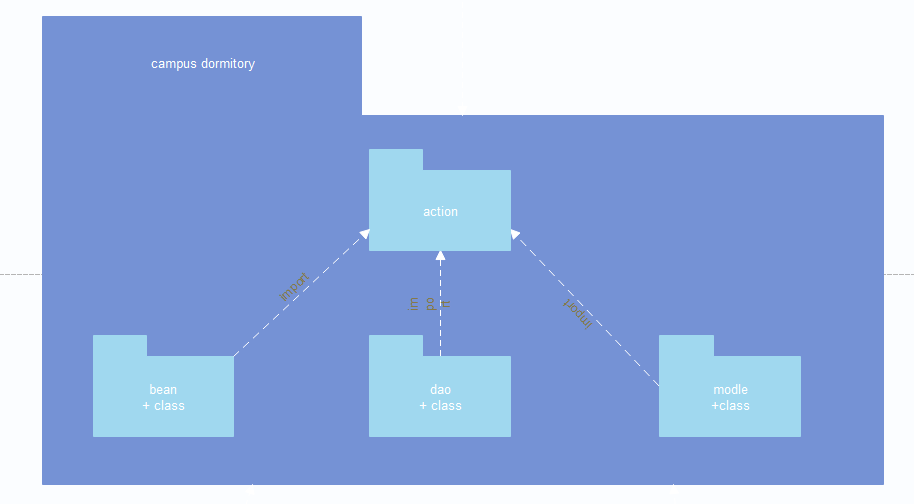
对顺序图中的所有对象和参与者，依次进行如下描述：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象类型 | 对象名称 | 是否为主动对象  （参与者不描述此内容） |
| 1 | 参与者 | 系统管理员 |  |
| 2 | 参与者 | 楼层管理员 |  |
| 3 | 参与者 | 学生用户 |  |
| 4 | 字符型 | 管理员信息 | 否 |
| 5 | 字符型 | 学生信息 | 否 |
| 6 | 字符型 | 楼层信息 | 否 |
| 7 | 字符型 | 登录身份验证 | 是 |
| 8 | 字符型 | 缺勤记录信息 | 否 |
| 9 | 字符型 | 进行迁入迁出 | 否 |

## 包图文档

### 包图

图2.3.1总体包图

****

校园宿舍管理系统的整体是通过包的嵌套进行描述的，action包通过bean、dao、modle三个公共包进行封装、依赖、嵌套，如上图2.3.1。

### 包图中的包描述

对包图中的每个包进行如下描述：

1.bean包

1. 包的名称：bean（组件）
2. 包的种类：其他
3. 用户可以使用Bean将功能、处理、值、数据库访问和其他任何可以用java代码创造的对象进行打包，并且可以通过内部的JSP页面、Servlet、其他JavaBean、applet程序或者应用来使用这些对象。
4. 与该包有关系的其他包：action里面的包，具有依赖关系。

2.dao包

a) 包的名称：dao（模型）

b) 包的种类：类包

c) DAO 数据访问对象是一个面向对象的数据库接口，它显露了 Microsoft Jet 数据库引擎，并允许通过 ODBC 像直接连接到其他数据库一样，直接连接到 Access 表。从而达到传递数据的功能。通过接口，数据库与操作代码进行连接，连接后进行操作、使用。主要是对导入数据库信息进行模型设计。

d） 与该包有关系的其他包：action里面的包，具有依赖、封装关系。

3.model包

a) 包的名称：model（模型）

b) 包的种类：其他

c) model是一种封装模型,而不是一种具体的参数或是其他的具体的体现。一般关系为model（模型层）-view（视图层）-controller（控制层），模型包括了数据模型(pojo或bean之类的东西)和业务模型（比如登陆，注册操作等），而controller层就是将你的model层能在view中表示出来。

d） 与该包有关系的其他包：action里面的包，具有封装关系，比如说对类“adminlog”来说，就是对用于封装用户请求参数，并且无具体参数等。

4.action包

a) 包的名称：action（控制台）

b) 包的种类：类包、其他

c) Action 是用于处理请求操作的，一个 Action 类代表一次请求或调用，每个请求的动作都对应一个相应的 Action 类。也就是说，用户的每次请求，都会转到一个相应的 Action 类中，由这个 Action 类进行处理。这个action被应用于接口。

d） 与该包有关系的其他包：泛化bean、dao、model包等。

# 系统设计

本章主要包含系统体系结构设计、数据库设计、用户界面设计。

## 系统体系结构设计

总体设计的基本目的是:用比较抽象概括的方式确定系统如何完成预定义的任务。即:确定系统的物理配置方案，并进而确定组成系统的每个程序的结构。

总体设计的必要性:可在全局的角度，用低成本，从较抽象的层次上分析对比多种可能的系统实现方案和软件结构，从中选出最佳方案和最合理的软件结构,从而用较低成本开发出较高质量的软件系统。

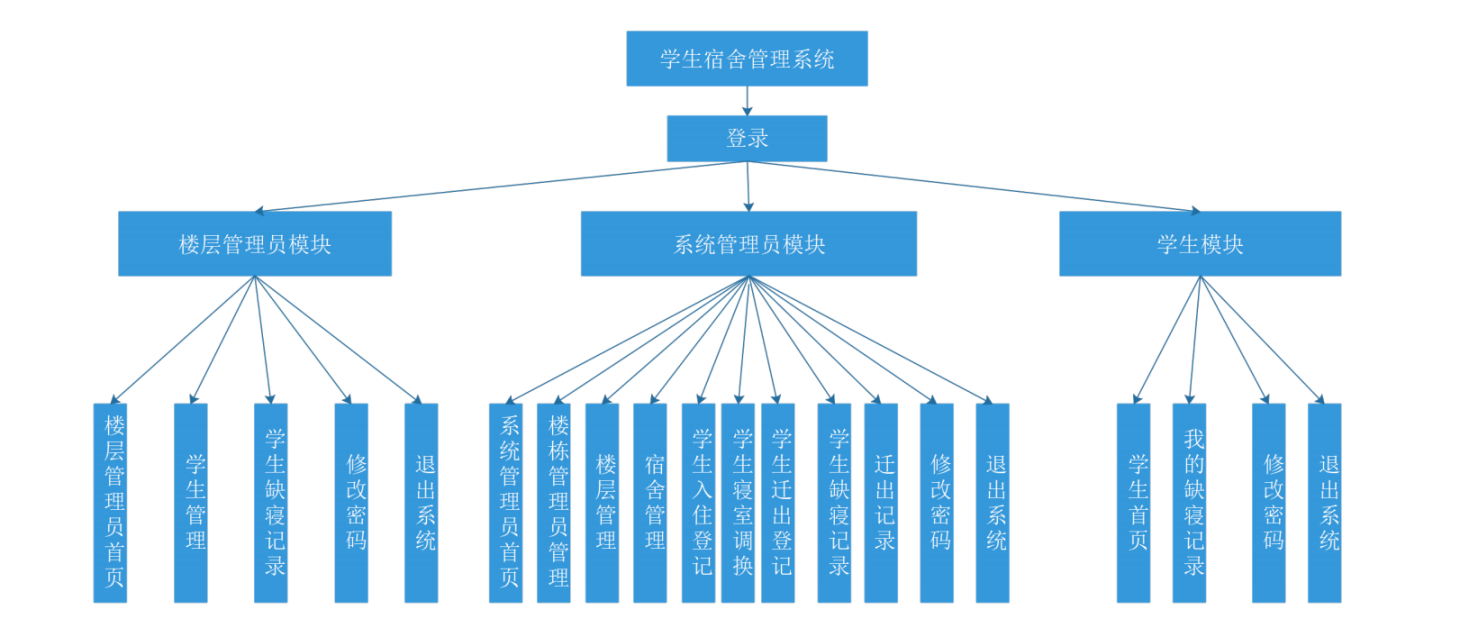
本系统主要分为5个相关的功能模块，分别为系统管理员子模块、楼层管理员子模块以及学生子模块。

系统管理员：管理员侧重对信息的管理，包括对楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录、迁出记录。同时系统管理员也可修改自身密码。并在所有操作结束后，退出系统。

楼层管理员：楼层管理员主要是为了减轻系统管理员的压力以及及时对学生的某些合理请求做出回应。楼层管理员可以进行学生管理以及学生缺寝记录的操作。同时楼层管理员也可修改自身密码。并在所有操作结束后，退出系统。

学生：学生仅能查看自身有无缺寝记录。同时学生也可修改自身密码。并在操作结束后，退出系统。

### 程序（模块）划分



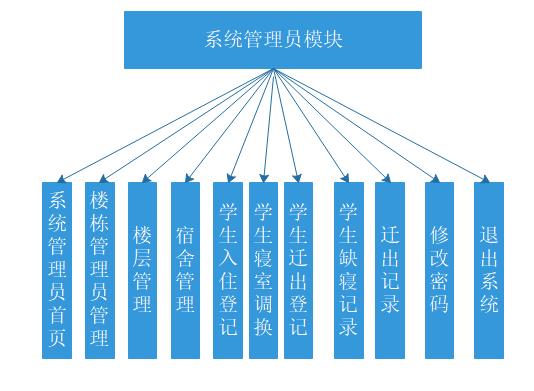
**图3.1.1系统功能模块图**

这里是对系统的总体功能进行概述，到二级功能。

### 程序（模块）层次结构关系

这里分别给出管理员子模块、楼层管理员子模块以及学生子模块的功能模块图，并对其模块图进行相应说明。

系统管理员模块功能图：



**图3.1.2系统管理员模块**

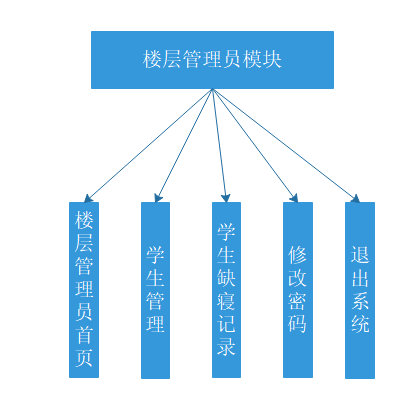
说明：系统管理员在成功登录后就可以成功进入到系统管理员模块。

系统管理员拥有以下功能：

1. 系统管理员拥有本系统的最高权限，可以对本系统设计到的所有功能操作进行操作；
2. 楼栋管理员管理，系统管理员可以增加、删除楼栋管理员的信息；
3. 学生管理，系统管理员可以在此模块查看已入住和未入住学生的基本信息并且修改他们的信息；
4. 楼层管理，系统管理员可以设置各个寝室楼层的管理员，并修改他们的信息；
5. 宿舍管理，可以查看所有寝室的信息，如寝室能容纳几人等，也可对此部分内容进行修改；
6. 学生入住登记，记录学生的学号以及对应的楼层，寝室号等；
7. 学生寝室调换，对学生调换寝室进行操作；
8. 学生迁出登记，登记迁出学生的学号；
9. 学生缺寝记录，登记缺寝学生的楼层房间号以及学号；
10. 迁出记录，可以查看所有迁出学生的记录；
11. 修改密码，可以修改自身密码；

完成所有操作后，退出系统。系统管理员可以对上述每个模块的数据进行修改，即增加、删除、修改、查询信息。

楼层管理员模块功能图：



**图3.1.3楼层管理员模块图**

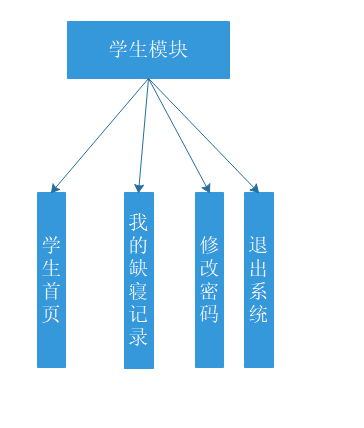
说明：楼层管理员在成功登录后就可以成功进入到楼层管理员模块。

楼层管理员拥有以下功能：

1. 楼层管理员拥有本系统的较多权限，可以协助系统管理员更好地完成工作，提高工作效率；
2. 学生管理，系统管理员可以在此模块查看已入住和未入住学生的基本信息并且修改他们的信息；
3. 学生缺寝记录，登记缺寝学生的楼层房间号以及学号
4. 修改密码，可以修改自身密码
5. 完成所有操作后，退出系统。

楼层管理员对学生管理子模块与学生缺寝记录子模块有增加、删除、修改、查询信息的功能。

学生模块功能图：



**图3.1.4学生模块图**

说明：学生在成功登录后就可以成功进入到学生模块。

学生子模块拥有以下功能：

1. 我的缺寝记录子模块，学生对该模块仅有查询功能。不能对其中的数据进行增加、删除以及修改等操作。
2. 学生仅可修改自身密码
3. 以上并不是说学生不能拥有修改其他功能模块的权限，只是目前尚未开发出来

## 数据库设计

根据逻辑结构设计，各个数据库表的设计结果如下:

1、系统管理员信息表存的是管理员的信息，包括管理员的姓名、登录密码、登录名、性别、电话号码等。详细信息如表1系统管理员表admin所示:

表1系统管理员表Admin

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | 用户id | 主键 |
| username | varchar | 20 | 是 | 用户名 |  |
| password | varchar | 20 | 是 | 用户密码 |  |
| name | varchar | 20 | 是 | 姓名 |  |
| sex | varchar | 10 | 是 | 性别 |  |
| tel | varchar | 20 | 是 | 电话号码 |  |

2、楼层信息表主要存储的是楼层的id、楼层的名字以及楼层的简介等信息。如图表2楼层信息表building所示:

表2楼层信息表building

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | 楼层d | 主键 |
| name | varchar | 50 | 是 | 楼层名称. |  |
| introduction | varchar | 1000 | 是 | 楼层介绍 |  |

3、楼层宿舍表里主要存储的是宿舍的一些状态、属性，主要包括id、所属楼层号、宿舍名称、宿舍类型、宿舍所住人数以及宿舍电话等。详细信息如表3楼层宿舍表domitory所示:

表3楼层宿舍表domitory

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | 宿舍id | 主键 |
| buildingid | int |  | 是 | 楼层号 |  |
| name | varchar | 20 | 是 | 宿舍名称 |  |
| type | varchar | 20 | 是 | 宿舍类型 |  |
| number | varchar | 20 | 是 | 宿舍人员 |  |
| tel | varchar | 20 | 是 | 宿舍电话 |  |

1. 缺勤记录表主要存储的是学生缺勤记录的信息，主要内容包括缺勤id、学生缺勤id、登记缺勤人员id、缺勤时间、缺勤备注等等，其中是缺勤id主键，详细信息如表4缺勤记录表log所示:

表4缺勤记录表log

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int | 20 | 否 | 缺勤id | 主键 |
| studentid | int | 20 | 是 | 学生缺勤id |  |
| teacherid | int | 20 | 是 | 登记缺勤人员id |  |
| date | varchar | 50 | 是 | 缺勤时间 |  |
| remark | varchar | 1000 | 是 | 缺勤备注 |  |

5、寝室迁出表主要存储的是迁出学生以及迁出时间信息，主要内容包括迁出人id、迁出学生id、迁出时间、迁出备注，详细信息如表5寝室迁出表out1 所示:

表5寝室迁出表out1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | 迁出人id | 主键 |
| studentid | varchar | 11 | 是 | 迁出学生id |  |
| date | timestamp | 6 | 否 | 迁出时间 |  |
| remark | varchar | 1000 | 是 | 迁出备注 |  |

6、学生信息表主要存储学生的在校信息，主要内容包括学生id、学生住楼层id、学号、登录密码、学生姓名、性别、班级、是否入住等信息，详细信息如表6学生信息表student所示:

表6学生信息表student

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | 学生id | 主键 |
| domitoryid | int |  | 是 | 学生住楼层id |  |
| username | varchar | 20 | 是 | 学号 |  |
| password | varchar | 20 | 是 | 登录密码 |  |
| name | varchar | 20 | 是 | 学生姓名 |  |
| sex | varchar | 20 | 是 | 性别 |  |
| class | varchar | 20 | 是 | 班级 |  |
| state | varchar | 20 | 是 | 是否入住 |  |

7、楼层管理员登记表主要存储每层楼层管理员的信息，包括id、楼层管理员id、楼层id等属性。详细信息如表7楼层管理员登记表tb所示:

表7楼层管理员登记表tb

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | id | 主键 |
| teacherid | int |  | 是 | 楼层管理员id |  |
| buildingid | int |  | 是 | 楼层id |  |

8、楼层管理员信息表主要是存储楼层管理员的相关信息，主要内容包括楼层管理员id、用户名、登录密码、姓名、性别、电话号码等信息，详细信息如表8楼层管理员信息表teacher所示:

表8楼层管理员信息表teacher

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 数据长度 | 是否为空 | 说明 | 备注 |
| id | int |  | 否 | id | 主键 |
| username : | varchar | 20 | 是 | 用户名 |  |
| password | varchar | 20 | 是 | 登录密码 |  |
| name | varchar | 20 | 是 | 姓名 |  |
| sex | varchar | 10 | 是 | 性别 |  |
| tel | varchar | 20 | 是 | 电话号码 |  |

## 用户界面设计

### 登录模块

登录用户分为三种:系统管理员、楼层管理员、学生，根据角色对应的权限不同会显示不同的菜单，用户输入用户名和密码，后台进行校验，用户名和密码确认后会根据用户的角色获取相应的权限菜单，跳转到主页面，界面图如图3.3.1所示：



图3.3.1 登录界面图

### 系统管理员模块

系统管理员模块包括楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录以及迁出记录等几个功能。主要是对楼栋管理员管理、学生管理、楼层管理、宿舍管理四个功能进行操作。学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录这几个功能是对楼层管理员的操作上进行再次操作，可以在楼层管理员未登录情况下，进行学生的操作。界面图如图3.3.2



图3.3.2 系统管理员menu界面

楼栋管理员管理是对楼层管理员进行操作的按钮。在这个页面里可以添加、修改、删除楼栋管理员以及查询楼栋管理员信息。查询可以通过姓名或者用户名进行查询。页面还展示了楼栋管理员的性别、电话、用户名的信息。如图3.3.2.1所示：



图3.3.2.1系统管理员操作楼栋管理员管理界面

学生管理是对学生的信息进行操作的按钮。在这个页面里可以添加、修改、删除学生信息以及查询学生信息。查询可以通过姓名、学号、班级或者入住、未入住、迁出进行查询。页面还展示了学生的学号、性别、班级、状态的信息。如图3.3.2.2所示：



图3.3.2.2系统管理员操作学生管理界面

楼层管理是对楼栋的楼层信息进行操作的按钮。在这个页面里可以查看楼层管理员、修改、删除楼层信息以及查询、添加楼层信息。查询只能通过名称进行查询。页面还展示了楼层的名称以及简介的信息。如图3.3.2.3所示：



图3.3.2.3系统管理员操作学生管理

宿舍管理是对楼层的宿舍信息进行操作的按钮。在这个页面里可以查看楼层管理员、修改、删除楼层信息以及查询、添加楼层信息。查询只能通过名称进行查询。页面还展示了楼层的名称以及简介的信息。如图3.3.2.4所示：



图3.3.2.4系统管理员操作宿舍管理

### 楼层管理员模块

楼层管理员模块包括学生管理以及学生缺寝记录两个功能。主要是对学生管理以及学生缺勤记录进行操作。学生缺寝记录这几个功能是对楼层管理员的操作上进行再次操作，可以在楼层管理员未登录情况下，进行学生的操作。界面图如图3.3.2：



图3.3.3.1楼层管理员menu界面

学生管理是在选择楼层的操作上进行再次操作，进而看到管理页面。界面图如图3.3.3.2





图3.3.3.2楼层管理员操作学生管理

学生缺寝记录是在选择楼层的操作上进行再次操作，进而看到管理页面。界面如图3.3.3.3：





图3.3.3.3楼层管理员操作学生缺勤记录

### 学生管理模块

学生管理模块主要是学生对自己的缺勤记录进行查看。界面展示了寝室号、姓名、性别、日期、备注以及登记人。如图3.3.4所示：



图3.3.4学生管理模块

# 测试分析报告

## 引言

由于本系统的使用范围为各大高校，用户基数庞大，学生宿舍信息管理系统所涉及到的事务较为繁杂，为了确保本系统的质量，需要对本系统进行各方面的测试。

软件测试规则如下:

1. 在做软件测试的时候必须做到全心全意地投入到软件测试中，秉承着认真、负责、规范的态度来进行测试。在进行具体测试要注意以下几点:
2. “坚持，集中注意力”应该是开发者的原则。
3. 测试用例的组成部分应包括测试输入数据和与之对应的预期输出结果。(3）程序员应该避免检查自己的程序，测试工作应该用独立的专业软件测试机构来完成。
4. 在设计测试用例时,既要有合理的输入条件又要有不合理的输入条件。(5)测试计划一定要认真、严格地执行,不能有随意性。、
5. 测试计划一定要认真、严格地执行,不能有随意性。
6. 妥善保存测试过的文档，避免测试的重现性。

### 编写目的

软件测试是软件质量保证的主要手段之一,也是将软件交付用户之前所必须完成的步骤。软件测试的目的是在软件投入生产性使用运行之前，尽可能多地发现软件产品中的错误和缺陷。

此文档根据项目需求文档，制定测试策略、评估测试风险，确定所需的资源，并对测试的工作量进行估计，进行人员和进度安排，并且列出测试项目的可交付元素。

本文档预期读者对象主要为项目经理、产品、开发、测试等。

### 项目背景

传统的学生公寓管理倾向于人工现场登记，资料多为纸质形式，不便于管理和查看。随着信息科技的快速发展，信息技术广泛应用对于各行业，在这样的情形之下，相应的学生公寓管理系统应运而生。

本项目由学校相关部门委托。

### 定义

无。

### 参考资料

《实用软件工程》，郑人杰等著，清华大学出版社；

《软件工程》第二版，李代平等著，清华大学出版社；

《软件工程第六版》，Roger S．Pressman著，机械工业出版社；

《软件工程课程实验指导书》，山东交通学院计算机科学与工程系；

《数据库系统概论》（第四版）,萨师煊 著,教育出版社；

《软件工程导论》（第五版）,张海藩 著，清华大学出版社

## 测试计划执行情况

### 测试项目

（列出每一测试项目的名称、内容和目的。）

|  |  |  |
| --- | --- | --- |
| 测试项目 | 测试内容 | 测试目的 |
| 用户登陆 | 查看用户账号及密码不符合格式时，系统作出的反应 | 验证系统对于不符合要求的登陆的反应是否正确。 |
| 楼层管理员添加 | 系统管理员对楼层管理员信息添加 | 验证系统管理员对楼层管理人员模块能否进行正确操作 |
| 楼层管理员信息修改 | 系统管理员对楼层管理员信息修改 | 验证系统管理员对楼层管理人员模块能否进行正确操作 |
| 楼层管理员查询 | 系统管理员对楼层管理员信息查询 | 验证系统管理员对楼层管理人员模块能否进行正确操作 |
| 学生添加 | 系统管理员对学生信息添加 | 验证系统管理员对学生信息模块能否进行正确操作 |
| 学生信息修改 | 系统管理员对学生信息修改 | 验证系统管理员对学生信息模块能否进行正确操作 |
| 学生查询 | 系统管理员对楼学生信息查询 | 验证系统管理员对学生信息模块能否进行正确操作 |
| 学生入住登记 | 系统管理员对学生入住进行登记 | 验证系统管理员对学生入住登记模块能否正常操作 |
| 学生寝室调换 | 系统管理员对学生寝室进行调换 | 验证系统管理员对学生寝室调换模块能否正常操作 |
| 学生缺寝登记 | 系统管理员对学生缺寝进行登记 | 验证系统管理员对学生缺寝登记模块能否正常操作 |
| 学生迁出登记 | 系统管理员对学生迁出进行登记 | 验证系统管理员对学生缺寝登记模块能否正常操作 |

### 测试机构与人员

（给出测试机构名称、负责人和参与测试人员名单。）

测试负责人：李足均

测试人员：李足均

|  |  |
| --- | --- |
| **测试编号：1-1** | **所属模块：用户登陆** |
| **测试项目：用户登陆** | |
| **测试内容：**  1. 登陆时，分别在账号处输入用例:  1.1) 18481886566  1.2) 184818865666  1.3) 1848188656  1.4)  1.5) 输入未注册的账号  1.6) \*&%\*aetbago  2. 登陆时，分别在密码处输入用例:  2.1) qq123456  2.2) 123456  2.3) 123456789112345678  2.4) ^\*%%^#%#^%  2.5) 输入不匹配账号的密码 | |
| **测试结果：**  1.1) 可以正常登陆  1.2) 提示“账号超过11位，请重新输入”  1.3) 提示“账号小于11位，请重新输入”  1.4) 提示未输入账号  1.5) 提示“该账号不存在！”  1.6) 提示“账号只能为数字！”  2.1) 可以正常登陆  2.2) 提示“密码不能小于8位”  2.3) 提示“密码不能超过16位”  2.4) 提示“密码只能为字母或数字！”  2.5) 提示“密码错误!” | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  登陆模块功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

### 测试结果

|  |  |
| --- | --- |
| **测试编号：1-2** | **所属模块：楼层管理员信息管理** |
| **测试项目：楼层管理员添加** | |
| **测试内容：**  1. 系统管理员登陆系统  2. 系统跳转到楼层管理员管理界面  3. 系统管理员点击添加楼层管理员  4. 系统跳转到楼层管理员信息界面  5. 系统管理员在该界面输入要添加为管理员的人的对应信息  7. 返回，查看楼层管理员是否添加成功  8. 退出登陆，登陆楼层管理员账号，查看楼层管理员自己是否被添加成功。 | |
| **测试结果：**  1. 在系统管理员添加楼层管理员后，返回楼层管理员界面，成功显示新添加的楼层管理员信息。  2. 在系统管理员添加楼层管理员后，使用楼层管理员登陆，成功显示自己已被添加。 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  楼层管理员添加功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-3** | **所属模块：楼层管理员信息管理** |
| **测试项目：楼层管理员信息修改** | |
| **测试内容：**  1. 系统管理员登陆系统  2. 系统跳转到楼层管理员管理界面  3. 系统管理员点击某个楼层管理员的信息修改按钮  4. 系统跳转到楼层管理员信息界面  5. 系统管理员在该界面输入要修改的管理员人的对应信息  6. 返回，查看楼层管理员信息是否修改成功  7. 退出登陆，登陆楼层管理员账号，查看楼层管理员自己的信息是否修改成功。 | |
| **测试结果：**  1. 在系统管理员修改楼层管理员信息后，返回楼层管理员界面，成功显示修改过的楼层管理员信息。  2. 在系统管理员信息修改后，使用楼层管理员登陆，成功显示自己已被添加。 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  楼层管理员信息修改功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-4** | **所属模块：楼层管理员信息管理** |
| **测试项目：楼层管理员查询** | |
| **测试内容：**  1. 系统管理员登陆系统  2. 系统跳转到楼层管理员管理界面  3.添加楼层管理员信息如下表：   |  |  |  |  | | --- | --- | --- | --- | | Id | 姓名 | 管理楼层 | 联系方式 | | 1001 | 张三 | 5 | 18481824364 | | 2008 | 李四 | 2 | 13219165588 | | 1364 | 王老五 | 7 | 19982541687 |   4. 系统管理员点击查询按钮  5. 系统管理员输入查询用例：  1).1001  2).李四  3).管理楼层 7  4).张五  5).2009  6. 查看能否查找到目标楼层管理员。 | |
| **测试结果：**  查找结果如下：1). 查找到Id为1001的楼层管理员  2). 查找到Id为2008的楼层管理员  3). 查找到Id为1364的楼层管理员  4). 提示未找到目标楼层管理员  5). 提示未找到目标楼层管理员 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  楼层管理员查询功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-5** | **所属模块：学生信息管理** |
| **测试项目：学生添加** | |
| **测试内容：**  1. 系统管理员登陆系统  2. 系统跳转到学生管理界面  3. 系统管理员点击添加学生  4. 系统跳转到学生信息界面  5. 系统管理员在该界面输入要添加的学生的对应信息  6. 返回，查看学生是否添加成功  7. 退出登陆，登陆学生账号，查看学生自己是否被添加成功。 | |
| **测试结果：**  1. 在系统管理员添加学生后，返回学生管理界面，成功显示新添加的学生信息。  2. 在系统管理员添加学生后，使用学生账号登陆，成功显示自己已被添加。 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生添加功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-6** | **所属模块：学生信息管理** |
| **测试项目：学生信息修改** | |
| **测试内容：**  1. 系统管理员登陆系统  2. 系统跳转到学生管理界面  3. 系统管理员点击某个学生的信息修改按钮  4. 系统跳转到学生信息界面  5. 系统管理员在该界面输入要修改的学生的对应信息  7. 返回，查看学生信息是否修改成功  8. 退出登陆，登陆学生账号，查看学生自己的信息是否修改成功。  9. 退出登陆，登陆学生所属楼层的楼层管理员账号，查看学生的信息是否修改成功。 | |
| **测试结果：**  1. 在系统管理员修改学生信息后，返回学生管理界面，成功显示修改过的学生信息。  2. 在系统管理员信息修改后，使用学生账号登陆，成功显示自己已被添加。  3. 在系统管理员信息修改后，使用学生所属楼层的楼层管理员账号登陆，成功显示该学生已被添加。 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生信息修改功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-7** | **所属模块：学生信息管理** |
| **测试项目：学生查询** | |
| **测试内容：**  1. 系统管理员登陆系统  2. 系统跳转到楼层管理员管理界面  3.添加楼层管理员信息如下表：   |  |  |  |  | | --- | --- | --- | --- | | Id | 姓名 | 所属楼层 | 联系方式 | | 1001 | 张三 | 5 | 18481824364 | | 2008 | 李四 | 2 | 13219165588 | | 1364 | 王老五 | 7 | 19982541687 |   4. 系统管理员点击查询按钮  5. 系统管理员输入查询用例：  1).1001  2).李四  3).所属楼层 7  4).张五  5).2009  6. 查看能否查找到目标学生。 | |
| **测试结果：**  查找结果如下：1). 查找到Id为1001的学生  2). 查找到Id为2008的学生  3). 查找到Id为1364的学生  4). 提示未找到目标学生  5). 提示未找到目标学生 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生查询功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-8** | **所属模块：学生入住** |
| **测试项目：学生入住登记** | |
| **测试内容：**  1. 系统管理员登陆系统，点击学生入住登记按钮  2. 系统跳转到学生入住登记界面  3. 系统管理员选择楼层  4. 系统弹出已存在的楼层  5. 点击该名学生对应楼层数，选择该楼层  6. 点击该名学生对应寝室号，输入该名学生学号  7. 点击确定入住  8. 返回至学生管理界面，查看该学生是否成功入住  9. 查看日志，是否有该学生登记记录 | |
| **测试结果：**  1. 该学生成功入住  2. 存在该学生入住登记记录 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生入住功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-9** | **所属模块：学生寝室调换** |
| **测试项目：学生寝室调换** | |
| **测试内容：**  1. 系统管理员登陆系统，点击学生寝室调换按钮  2. 系统管理员在“请输入要调换寝室学生的学号”输入框输入请求调换寝室的学生的学号，点击确认按钮  3. 系统跳转到学生寝室调换的具体界面  4. 系统管理员选择要调换到的寝室，确认调换  5. 返回至学生管理界面，查看该学生是否成功寝室调换 | |
| **测试结果：**  该学生成功寝室调换 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生寝室调换功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-10** | **所属模块：学生缺寝登记** |
| **测试项目：学生入住登记** | |
| **测试内容：**  1. 系统管理员登陆系统，点击学生缺寝登记按钮  2. 系统跳转到学生缺寝登记界面  3. 系统管理员输入缺寝学生学号进行查询  4. 系统弹出查询到的学生信息  5. 点击该名学生对应的缺寝登记按钮  6. 返回至学生管理界面，查看该学生是否成功显示缺寝  7. 查看日志，是否有该学生缺寝记录 | |
| **测试结果：**  1. 成功显示该学生缺寝  2. 存在该学生缺寝登记记录 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生缺寝登记功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

|  |  |
| --- | --- |
| **测试编号：1-11** | **所属模块：学生迁出登记** |
| **测试项目：学生迁出登记** | |
| **测试内容：**  1. 系统管理员登陆系统，点击学生迁出登记按钮  2. 系统跳转到学生迁出登记界面  3. 系统管理员输入迁出学生学号进行查询  4. 系统弹出查询到的学生信息  5. 点击该名学生对应的迁出登记按钮  6. 返回查看寝室信息管理界面，查看该学生原寝室是否存在该学生  7. 查看日志，是否有该学生迁出登记记录 | |
| **测试结果：**  1. 原寝室不再存在该学生，该学生成功迁出  2. 存在该学生迁出登记记录 | |
| **预期结果：**  与测试结果相同。 | |
| **结果偏差：**  无偏差。 | |
| **结果分析：**  学生迁出登记功能正常。 | |
| **发现的问题：**  未发现任何问题。 | |

## 软件需求测试结论

|  |  |  |
| --- | --- | --- |
| 测试项目 | 证实的软件能力 | 局限性 |
| 用户登陆 | 用户登陆模块功能能正常进行 | 无 |
| 楼层管理员添加 | 楼层管理员添加模块功能能正常进行 | 无 |
| 楼层管理员信息修改 | 楼层管理员信息修改模块功能能正常进行 | 无 |
| 楼层管理员查询 | 楼层管理员查询模块功能能正常进行 | 无 |
| 学生添加 | 学生添加模块功能能正常进行 | 无 |
| 学生信息修改 | 学生信息修改模块功能能正常进行 | 无 |
| 学生查询 | 学生查询模块功能能正常进行 | 无 |
| 学生入住登记 | 学生入住登记模块功能能正常进行 | 无 |
| 学生寝室调换 | 学生寝室调换模块功能能正常进行 | 无 |
| 学生缺寝登记 | 学生缺寝登记模块功能能正常进行 | 无 |
| 学生迁出登记 | 学生迁出登记模块功能能正常进行 | 无 |

## 评价

### 软件能力

学生公寓管理系统采用B/S结构，使用JSP技术、Servlet技术、Java语言和My SQL数据库开发。涵盖了前后端开发和数据库的创建三大内容。该系统主要有两大用户：系统管理员、楼层管理员以及学生。

系统管理员：管理员侧重对信息的管理，包括对楼栋管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、学生寝室调换、学生迁出登记、学生缺寝记录、迁出记录。同时系统管理员也可修改自身密码。并在所有操作结束后，退出系统。

楼层管理员：楼层管理员主要是为了减轻系统管理员的压力以及及时对学生的某些合理请求做出回应。楼层管理员可以进行学生管理以及学生缺寝记录的操作。同时楼层管理员也可修改自身密码。并在所有操作结束后，退出系统。

学生：学生仅能查看自身有无缺寝记录。同时学生也可修改自身密码。并在操作结束后，退出系统。

### 缺陷与限制

测试后发现缺陷及限制。

1.注册学生功能过于简陋，没有体现注册功能。

2.登录注册时，特殊字符被限制。

3.当用户数过多时，会影响登录。

### 建议

该软件的可读性还需要继续更新，继续保持。

### 测试结论

根据之前的概要设计和详细设计进行了编码与测试。测试主要采用了黑盒测试，通过等价划分技术，分为有效等价类和无效等价类，对系统进行了测试。

经过测试，本软件的楼层管理员管理、学生管理、楼层管理、宿舍管理、学生入住登记、

学生寝室调换登记、迁出登记、修改密码等等各模块功能皆能正常进行，符合预期结果。

附件：