数据库原理与应用

课程设计报告

**学生宿舍数据库应用系统**

**设计与实现**

**班级：信息17-2班**

**姓名：杨楚峤**

**学号：171001220**

**设计日期：2019年12月05日**

目录

[1 绪论 1](#_Toc26524316)

[2 数据库应用系统需求分析 1](#_Toc26524317)

[2.1 系统的目标 1](#_Toc26524318)

[2.2 数据库应用系统的需求分析 2](#_Toc26524319)

[2.2.1 系统需求分析 2](#_Toc26524320)

[2.2.2 数据流程图 5](#_Toc26524321)

[3 数据库概念模型设计 7](#_Toc26524322)

[3.1 概念模型设计 7](#_Toc26524323)

[3.1.1 全局ER模型图 7](#_Toc26524324)

[3.1.2 宿舍数据库中的实体集 7](#_Toc26524325)

[3.1.2 宿舍数据库中的联系集 8](#_Toc26524326)

[3.2 物理逻辑模型设计 8](#_Toc26524327)

[4 关系结构设计 9](#_Toc26524328)

[4.1 关系模式设计和表结构 9](#_Toc26524329)

[4.1.1 关系模式设计 9](#_Toc26524330)

[4.1.2 表结构设计 10](#_Toc26524331)

[4.2 数据库和表创建 16](#_Toc26524332)

[4.2.1 数据库创建的SQL语句 16](#_Toc26524333)

[4.2.2 数据库中每个表创建的SQL语句 16](#_Toc26524334)

[4.2.3 视图创建的SQL语句 18](#_Toc26524335)

[4.3 查询与统计 19](#_Toc26524336)

[4.4 触发器设计与实现 20](#_Toc26524337)

[4.5 存储过程设计与实现 20](#_Toc26524338)

[5 数据库安全性策略设计 27](#_Toc26524339)

[5.1 数据库角色及权限设计 27](#_Toc26524340)

[5.2 数据库备份 28](#_Toc26524341)

[6 系统设计与实现 28](#_Toc26524342)

[6.1 系统的架构与功能结构 29](#_Toc26524343)

[6.1.1 系统体系架构 29](#_Toc26524344)

[6.1.2 系统功能结构 29](#_Toc26524345)

[6.2 系统运行环境、开发技术和语言 30](#_Toc26524346)

[6.2.1 系统运行环境 30](#_Toc26524347)

[6.2.2 系统开发技术 30](#_Toc26524348)

[6.3 功能模块设计 30](#_Toc26524349)

[6.3.1 用户登录功能 30](#_Toc26524350)

[6.3.2 学生信息管理功能 32](#_Toc26524351)

[6.3.3 宿舍信息管理功能 39](#_Toc26524352)

[6.3.4 职工信息管理功能 41](#_Toc26524353)

[6.3.5 宿舍报修服务功能 44](#_Toc26524354)

[6.3.6 宿舍费用服务功能 47](#_Toc26524355)

[6.3.7 公告通知服务功能 49](#_Toc26524356)

[6.3.8 来访登记管理功能 55](#_Toc26524357)

[6.3.9 晚归登记管理功能 59](#_Toc26524358)

[6.3.10 综合查询功能 62](#_Toc26524359)

[6.3.11 综合统计功能 66](#_Toc26524360)

[6.3.12 数据备份功能 68](#_Toc26524361)

[6.3.13 数据还原功能 70](#_Toc26524362)

[6.3.14 用户管理功能 71](#_Toc26524363)

[6.3.15 修改密码功能 73](#_Toc26524364)

[7 系统运行实例 75](#_Toc26524365)

[8 结束语 83](#_Toc26524366)

[8.1 遇到的问题以及解决途径 84](#_Toc26524367)

[8.2 待改进的方面 84](#_Toc26524368)

[8.3 课程设计心得与收获 84](#_Toc26524369)

[附件：数据字典 84](#_Toc26524370)

图目录

[图1 组织机构图 2](#_Toc26524452)

[图2 宿舍管理业务流程图 3](#_Toc26524453)

[图3 系统顶层图 5](#_Toc26524454)

[图4 系统第一层数据流图 5](#_Toc26524455)

[图5 入住/离宿数据流图 6](#_Toc26524456)

[图6 报修服务数据流图 6](#_Toc26524457)

[图7 晚归登记数据流图 6](#_Toc26524458)

[图8 来访登记数据流图 6](#_Toc26524459)

[图9 宿舍费用数据流图 7](#_Toc26524460)

[图10 全局ER模型图 7](#_Toc26524461)

[图11 系统物理数据模型图 8](#_Toc26524462)

[图12 存储过程pro\_laifang 21](#_Toc26524463)

[图13 存储过程pro\_student 22](#_Toc26524464)

[图14 存储过程pro\_zhigong 23](#_Toc26524465)

[图15 存储过程pro\_pwd 24](#_Toc26524466)

[图16 存储过程pro\_wanguichaxun 25](#_Toc26524467)

[图17 存储过程pro\_feiyong 26](#_Toc26524468)

[图18 存储过程pro\_wangui 27](#_Toc26524469)

[图19 系统功能结构图 29](#_Toc26524470)

[图20 用户登录功能流程图 31](#_Toc26524471)

[图21 登录功能界面 31](#_Toc26524472)

[图22 学生信息管理功能流程图 33](#_Toc26524473)

[图23 学生信息管理功能界面 33](#_Toc26524474)

[图24 学生信息管理功能界面 34](#_Toc26524475)

[图25 学生信息管理功能界面 34](#_Toc26524476)

[图26 学生信息管理功能界面 35](#_Toc26524477)

[图27 宿舍信息管理功能 39](#_Toc26524478)

[图28 学生宿舍管理功能界面 39](#_Toc26524479)

[图29 学生宿舍管理功能界面 40](#_Toc26524480)

[图30 学生宿舍管理功能界面 40](#_Toc26524481)

[图31 学生宿舍管理功能界面 41](#_Toc26524482)

[图32 职工信息管理功能流程图 41](#_Toc26524483)

[图33 职工信息管理功能界面 42](#_Toc26524484)

[图表 34 职工信息管理功能界面 42](#_Toc26524485)

[图35 职工信息管理功能界面 43](#_Toc26524486)

[图36 职工信息管理功能界面 43](#_Toc26524487)

[图37 宿舍报修服务功能流程图 44](#_Toc26524488)

[图38 宿舍报修服务功能界面 44](#_Toc26524489)

[图39 宿舍报修服务功能界面 45](#_Toc26524490)

[图40 宿舍报修服务功能界面 45](#_Toc26524491)

[图41 宿舍费用服务功能流程图 48](#_Toc26524492)

[图42 宿舍费用服务功能界面 48](#_Toc26524493)

[图43 公告通知服务功能流程图 49](#_Toc26524494)

[图44 公告通知服务功能界面 50](#_Toc26524495)

[图45 公告通知服务功能界面 50](#_Toc26524496)

[图46 公告通知服务功能界面 51](#_Toc26524497)

[图47 公告通知服务功能界面 51](#_Toc26524498)

[图48 来访登记管理功能流程图 55](#_Toc26524499)

[图49 来访登记管理功能界面 55](#_Toc26524500)

[图50 来访登记管理功能界面 56](#_Toc26524501)

[图51 来访登记管理功能界面 56](#_Toc26524502)

[图52 晚归登记管理功能流程图 59](#_Toc26524503)

[图53 晚归登记管理功能界面 60](#_Toc26524504)

[图54 晚归登记管理功能界面 60](#_Toc26524505)

[图55 综合查询功能流程图 62](#_Toc26524506)

[图56 综合查询功能界面 63](#_Toc26524507)

[图57 综合查询功能界面 63](#_Toc26524508)

[图58 综合查询功能界面 64](#_Toc26524509)

[图59 综合查询功能界面 64](#_Toc26524510)

[图60 综合统计功能界面 67](#_Toc26524511)

[图61 综合统计功能界面 67](#_Toc26524512)

[图62 数据备份功能界面 68](#_Toc26524513)

[图63 数据备份功能界面 69](#_Toc26524514)

[图64 数据还原功能界面 70](#_Toc26524515)

[图65 用户管理功能界面 71](#_Toc26524516)

[图66 用户管理功能界面 71](#_Toc26524517)

[图67 用户管理功能界面 72](#_Toc26524518)

[图68 修改密码功能界面 74](#_Toc26524519)

[图69 登录界面 75](#_Toc26524520)

[图70 登陆成功 75](#_Toc26524521)

[图71 主界面 76](#_Toc26524522)

[图72 学生信息管理界面 76](#_Toc26524523)

[图73 宿舍信息管理界面 77](#_Toc26524524)

[图74 职工信息管理界面 77](#_Toc26524525)

[图75 宿舍报修服务界面 78](#_Toc26524526)

[图76 宿舍费用查询界面 78](#_Toc26524527)

[图77 公告信息处理界面 79](#_Toc26524528)

[图78 外来人员登记界面 79](#_Toc26524529)

[图79 学生信息查询界面 80](#_Toc26524530)

[图表 80 职工信息查询界面 80](#_Toc26524531)

[图表 81 晚归信息统计界面 81](#_Toc26524532)

[图82 宿舍电量统计界面 81](#_Toc26524533)

[图83 数据备份 82](#_Toc26524534)

[图84 数据恢复 82](#_Toc26524535)

[图85 用户管理界面 82](#_Toc26524536)

[图86 增加用户界面 83](#_Toc26524537)

[图87 修改权限界面 83](#_Toc26524538)

[图88 修改密码界面 83](#_Toc26524539)

1 绪论

随着时代的发展，各行业越来越依赖于计算机科技为人们提供越来越快捷便利的服务。管理信息系统开始被广泛应用到人们生活、学习、工作、娱乐的各个领域。现阶段，高校对于学生宿舍的管理工作依然采取宿舍管理人员手工记录数据的传统模式。信息化时代里，数据就意味着价值。面对如此庞大的数据量，人工记录的方式是相当麻烦的，既不利于数据的长期保存，也不利于数据的增删改查，更无法直接利用信息技术挖掘数据本质与内涵。基于此类现象，任务运用“互联网+”思维模式，致力于改变传统的宿舍管理手段和方式，面向高校学生宿舍，打造一个“学生宿舍数据库应用系统”来改善高校宿舍管理效率，降低宿舍管理支出，对提高高校学生宿舍管理的信息化水平具有重要的现实意义。

2 数据库应用系统需求分析

**2.1 系统的目标**

此学生宿舍数据库应用系统以北京林业大学学生宿舍管理的实际业务为例，使用C#编程语言和SQL数据存储技术进行研发，致力于实现学生宿舍管理信息化，适应快速增长学生数的要求，立足于现有的人力资源，充分利用计算机的强大功能，加强宿舍工作各阶段的科学管理，并对管理决策提供良好支持。系统具有基本信息管理，宿舍服务管理，出入登记管理，公告栏管理，数据维护管理，用户中心管理六大核心业务模块，主要实现的功能如下：

（1）基本信息管理

宿舍信息管理：记录宿舍的基本状况，可住人数，已住人数，宿舍长学号、姓名等信息，便于工作人员与宿舍取得联系。

学生信息管理：学生信息主要是由学生处提供的，宿舍管理人员从学生处提取相关信息，进行统一管理，也可对现有的学生信息表进行增删改查。

职工信息管理：记录宿舍管理工作人员的相关信息，及时的增删改查有助于宿舍的管理。

（2）宿舍服务管理

宿舍费用服务：宿舍管理人员会从后勤管理部门获取各宿舍使用费用，并根据现有的价格规定，结算出每个宿舍应该支付的费用。

宿舍报修服务：学生需要把需要维修的设备向宿舍管理员提出申请，填写相应的报修单信息，再由宿舍管理员联系相应部门进行检修。

公告栏服务：宿舍管理人员可以将信息发布在公告栏，通知学生电梯维修，停电通知等日常事宜，方便同学们接收最新消息。

（3）出入登记管理

外来人员登记：宿舍管理人员记录外来人员所登记的信息，系统自动与学生信息表进行息匹配，经过核实，确定可以访问。学生晚归登记：宿舍管理人员登记晚归学生的个人信息，以及相应缘由，方便宿舍管理。可定期导出晚归学生信息表，由教务处对晚归次数过多的学生进行相应处理

（4）查询统计管理

综合查询：可以根据用户所选择的筛选条件进行筛选查询。

综合统计：可以根据相应的要求进行数据统计。

（5）数据维护管理

学生晚归登记：宿舍管理人员登记晚归学生的个人信息，以及相应缘由，方便宿舍管理。可定期导出晚归学生信息表，由教务处对晚归次数过多的学生进行相应处理。

（4）查询统计管理

综合查询：可以根据用户所选择的筛选条件进行筛选查询。

综合统计：可以根据相应的要求进行数据统计。

（5）数据维护管理

数据备份：备份所选数据表以免数据发生错误而导致数据丢失。

数据还原：如因管理人员操作不当或系统故障等原因导致数据报错，丢失，混乱，可以选择还原之前备份好的数据，充分保护数据的安全可靠。

（6）用户中心管理

用户管理：管理员可以更改用户权限以及增加新用户。

修改密码：用户可以更改自身密码。

重新登录：用户可以重新登陆。

退出系统：用户可以退出系统。

**2.2 数据库应用系统的需求分析**

2.2.1 系统需求分析

2.2.1.1 组织机构

宿舍属于特殊行业，根据调查，北京林业大学每个宿舍均有楼长全权负责本宿舍楼一切事务，全权管理学生宿舍各种事宜，下面的员工主要有日常管理员、维修人员、保洁人员、保安人员等，各员工业务相对独立，各尽其责，共同维护宿舍正常工作。

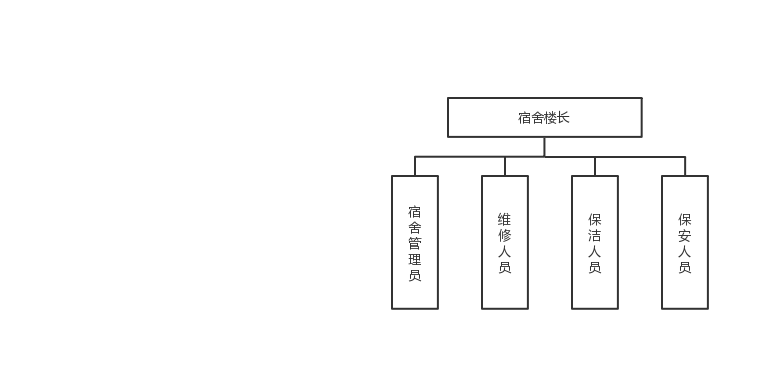


图1 组织机构图

其中各工作人员的职责为：

1）日常管理员：负责管理学生的日常起居，处理学生报修单等事务

2）维修人员：负责整个学生宿舍内部的物品维修

3）保洁人员：负责清洁学生宿舍，保持清洁卫生

4）保安人员：负责学生宿舍的安全保卫工作

2.2.1.2 宿舍管理业务内容

系统的业务是指系统要达到的业务目标，业务流程分析也是系统分析中的基础环节。现阶段，北京林业大学学生宿舍仍未完全实现管理系统信息化，在一些核心任务环节上仍然采取手工记录的形式，降低办公效率。经过对其工作流程的长期摸索考察，我对其业务流程做出如下几点具体分析：

1）入住办理：新同学入住宿舍，需要进行详细的核对审查，按照院系专业班级分配宿舍，然后把确定入住的同学存进档案，并修改其入住宿舍的相关信息。当宿舍容量不足时会向学生处反映情况，由学生处统筹安排。

2）离宿办理：同学因休学、毕业等原因办理离宿手续，需携带学院审批意见到宿舍管理员处办理离宿手续，删除学生信息，并修改宿舍相关信息。

3）公告通知：公告通知单由宿舍管理员在每一楼层以及电梯内部张贴，每一个通知单保留平均8天左右，同学可利用进出宿舍时间查看相关信息。

4）宿舍报修：宿舍报修是由宿舍同学到宿舍管理员处反映报修物件以及具体情况，并进行相应信息登记，由宿舍管理员联系维修人员进行维修。

5）晚归登记：宿舍会有值班管理员，学生需要填写晚归信息，并写出晚归缘由。

6）外来人员登记：因为宿舍管理员不能快速调出学生个人信息，无法根据外来人员所描述信息进行比对，因此探访必须由同学亲自接送才可进出宿舍，来访时必须登记相关信息。

7）宿舍费用：宿舍费用主要在于用水，用电，学校的后勤部门会按月份结算水费，电费，对欠费的宿舍进行停电处理。学生得知欠费后，自行前往后勤部门缴费。

根据对现行学生宿舍业务流程的分析，绘制宿舍管理业务流程图如下所示：

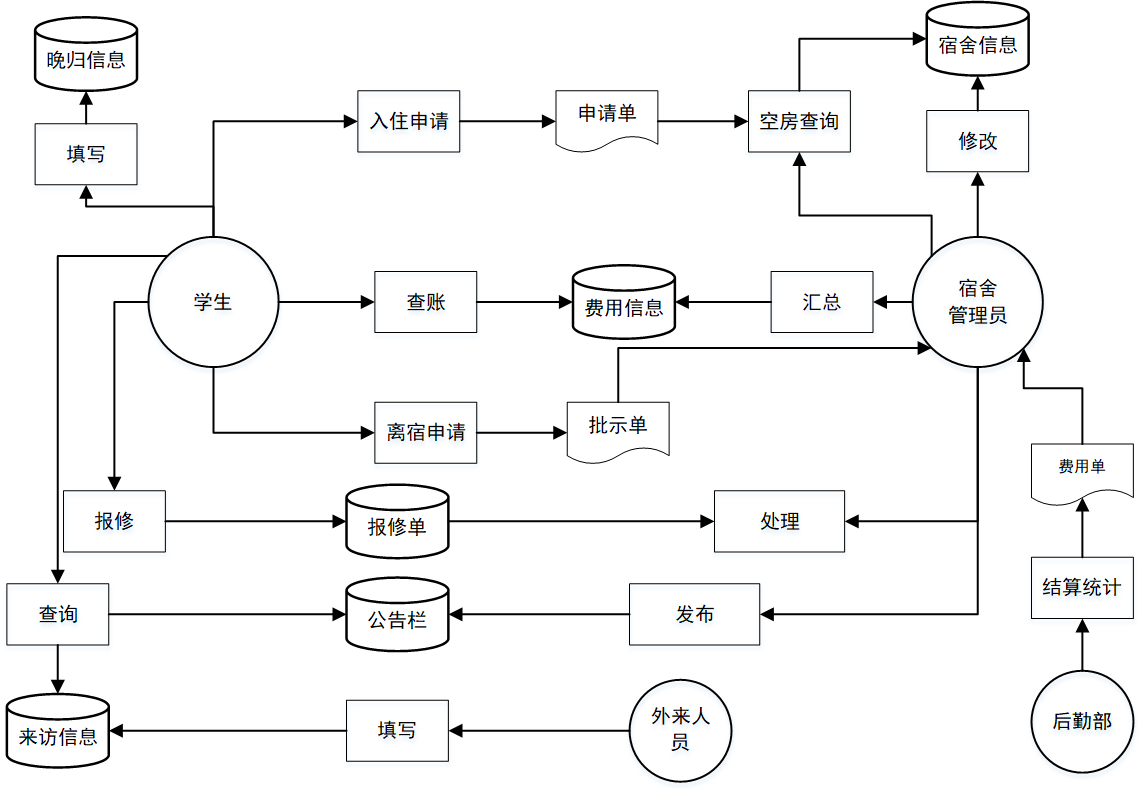


图2 宿舍管理业务流程图

2.2.1.3 系统功能需求

学生宿舍数据库应用系统需要实现日常宿舍管理业务的基本功能，主要包括基本信息管理、宿舍服务管理、出入登记管理、查询统计管理、数据维护管理、用户中心管理。

（1）基本信息管理：

①实现学生信息的查询、增加、删除、修改；

②实现宿舍信息查询、增加、删除、修改；

③实现职工信息查询、增加、删除、修改；

（2）宿舍服务管理：

①实现宿舍报修信息的查询、增加、删除、修改；

②实现宿舍费用信息查询、增加、删除、修改；

③实现公告栏信息查询、发布、删除、修改；

（3）出入登记管理：

①实现晚归学生信息的查询、登记、删除、修改；

②实现来访登记信息查询、增加、删除、修改；

（4）查询统计管理：

①按照查询要求，实现学生情况，宿舍情况的信息查询；

②按照统计条件，实现学生情况，宿舍情况的信息统计；

（5）数据维护管理：

①实现数据备份的功能；

②实现数据还原的功能；

（6）用户中心管理：

①实现用户信息的查询、增加、删除、修改；

②实现用户权限的设置；

③实现用户密码的更改；

④实现本系统的退出、重登录功能。

2.2.1.4 系统要求

（1）数据保密

系统为不同用户设置相应权限，为保证数据的安全可靠，用户只能进行自身权限范围内的操作，具体权限要求如下：

①各部门经理以及宿舍管理员具有处理数据库中所有表格的权限;

②维修人员可以处理报修业务数据；

③安保人员可以查询宿舍以及学生基本信息；

④学生只可以处理与自己以及自己所在宿舍有关的业务数据。

（2）数据完整

系统需要实现对数据完整性的判断功能，保证基本信息务必填写并准确无误。系统除可以自动判断是否存在非法字符外，还可以实现对输入的用户信息自动进行数据库数据比对。

（3）查询响应时间的要求

系统全天运行在宿舍管理员主机上，因为学生人数较多，为节约办理业务时间，对数据的查询速度要求较高，系统会通过在数据库中建立视图、存储过程等来尽可能提高查询速度，满足用户需求。

（4）输入方式与输出报表

系统需要为用户建立良好的图形化界面，引导用户进行相应信息的输入。在某些重要的查询模块，需要增加导出Excel表格的功能。

2.2.2 数据流程图

为建立一个满足用户需求的管理信息系统，系统开发人员需要对调查材料进行加工提炼，抽取出其中反映信息运动规律的部分。根据上述对业务流程和业务处理功能详细分析，可以将整个系统各种数据的产生源头、流向和终点，各种数据的产生、修改、查询及更新过程，表示为以下数据流程图：

2.2.2.1 系统顶层图

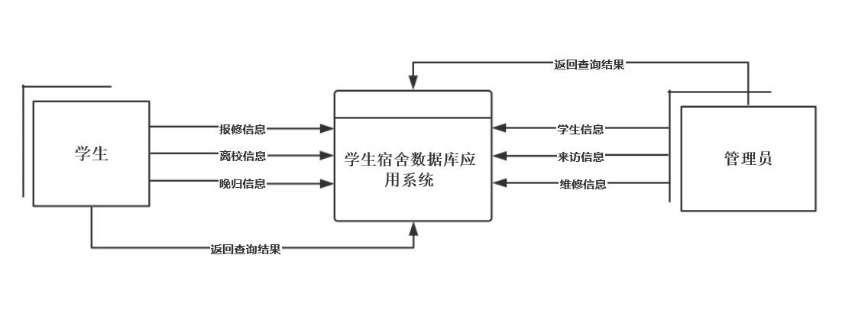


图3 系统顶层图

2.2.2.2 系统第一层数据流图

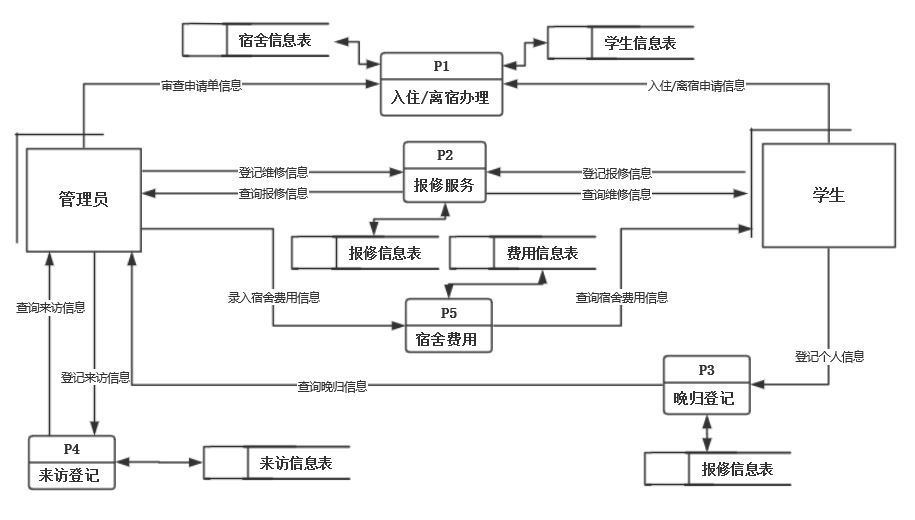


图4 系统第一层数据流图

2.2.2.3 系统第二层数据流图

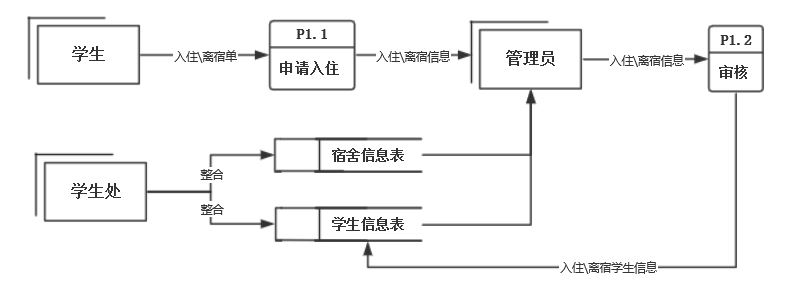


图5 入住/离宿数据流图

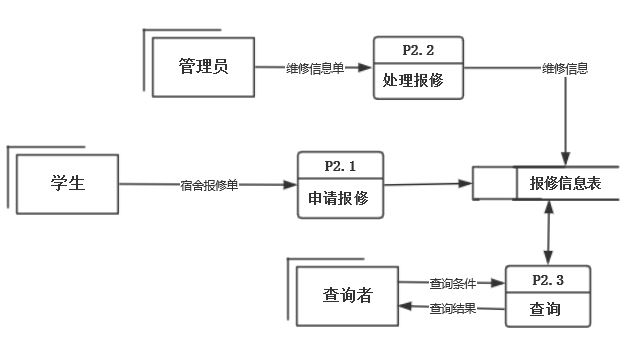


图6 报修服务数据流图

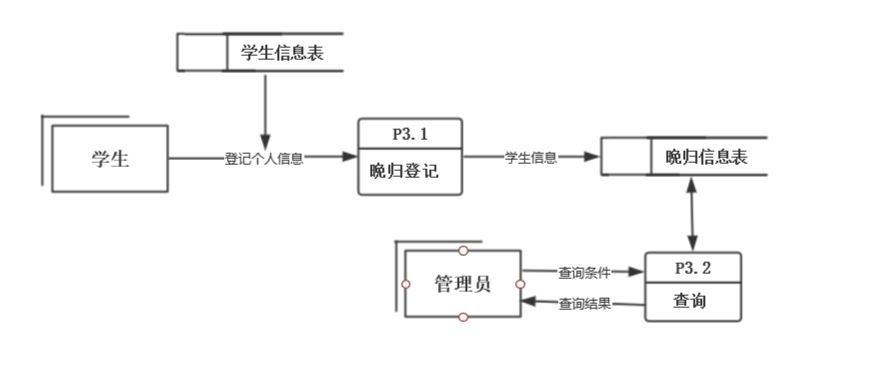


图7 晚归登记数据流图

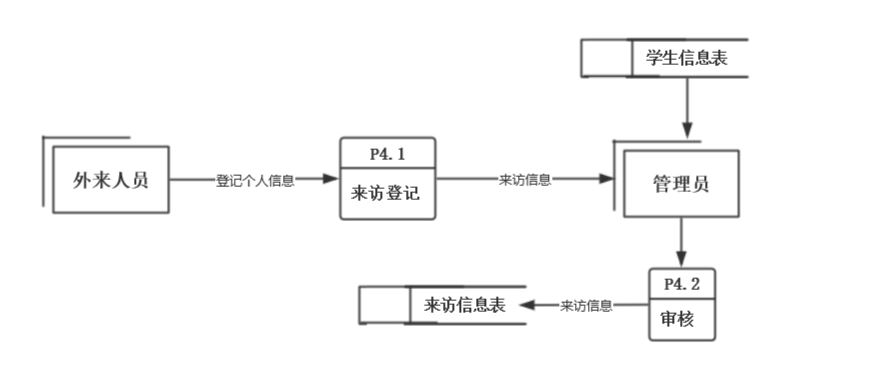


图8 来访登记数据流图

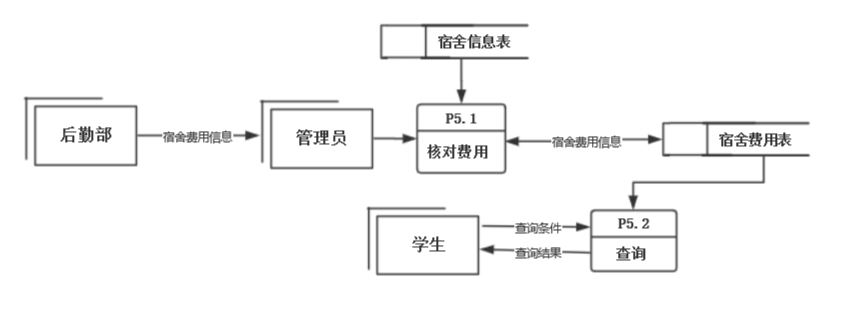


图9 宿舍费用数据流图

3 数据库概念模型设计

**3.1 概念模型设计**

3.1.1 全局ER模型图

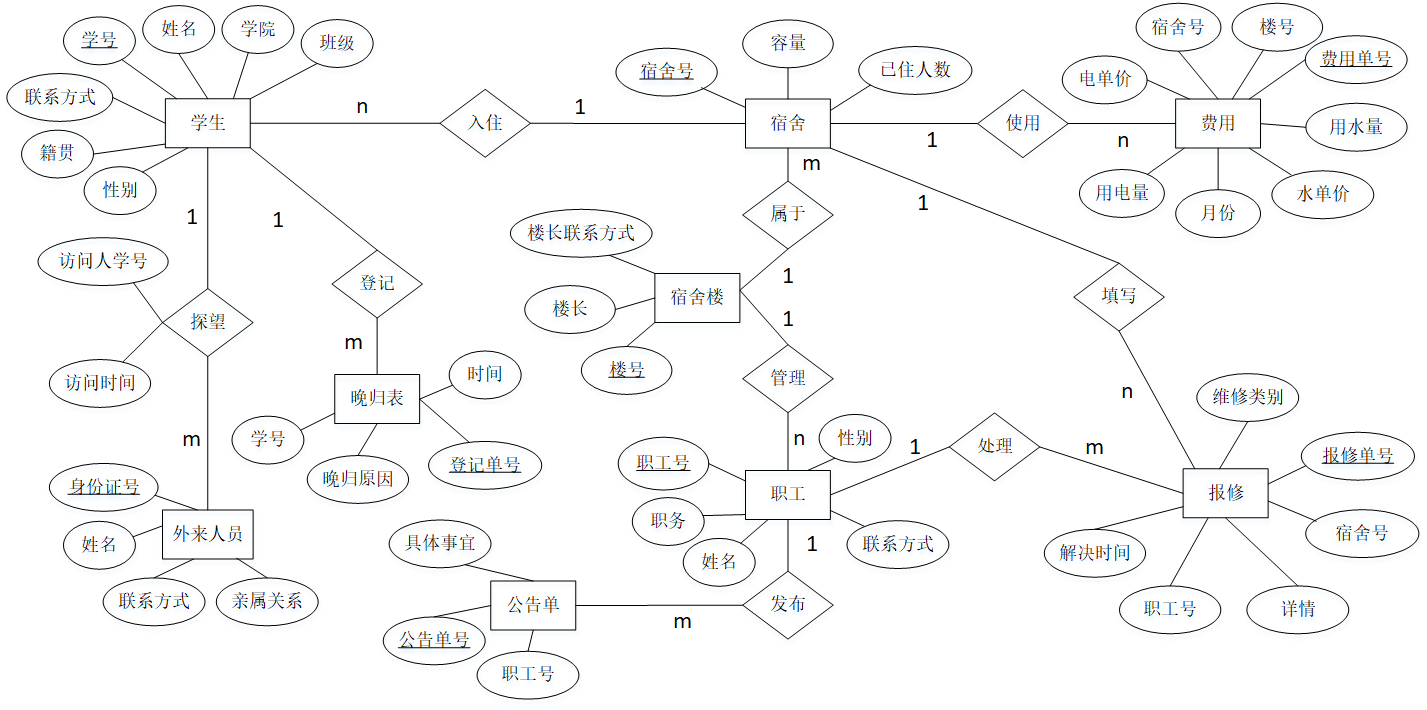


图10 全局ER模型图

3.1.2 宿舍数据库中的实体集

* 实体集学生信息，具有属性学号，姓名，学院，班级，籍贯，性别，联系方式
* 实体集宿舍信息，具有属性宿舍号，容量，已住人数
* 实体集宿舍楼信息，具有属性楼号，楼长，楼长联系方式
* 实体集职工信息，具有属性职工号，姓名，性别，联系方式，职务
* 实体集报修信息，具有属性报修单号，宿舍号，维修类别，详情，解决日期
* 实体集费用信息，具有属性费用单号，宿舍号，月份，水单价，用水量，电单价，用电量
* 实体集公告单信息，具有属性公告单号，职工号，具体事宜
* 实体集外来人员信息，具有属性身份证号，姓名，联系方式，亲属关系
* 实体集学生晚归信息，具有属性登记单号，学号，晚归原因，晚归时间

3.1.2 宿舍数据库中的联系集

* 入住，宿舍与学生之间具有一对多的关系；一个学生只能入住一个宿舍，一个宿舍可以有多名学生入住；
* 属于，宿舍楼与宿舍之间具有一对多的关系；一个宿舍只能属于一个宿舍楼，一个宿舍楼拥有多间宿舍；
* 管理，宿舍与职工之间具有一对多的关系；一个宿舍由多个职工管理，一个职工负责一个宿舍楼；
* 探望，学生与外来人员具有一对多的关系，具有属性访问时间，访问人学号；外来人员探望一位学生，一位学生可以多维外来人员探望；
* 维修，宿舍与报修表之间具有一对多的关系；一个宿舍对应多张报修表，一个报修表隶属于一个宿舍；
* 费用，宿舍与费用表之间具有一对多的关系；一个宿舍对应多张费用表，一个费用表隶属于一个宿舍；
* 晚归，学生与晚归表之间具有一对多的关系；一个学生对应多张晚归表，一个晚归表隶属于一个学生；
* 发布，职工与公告单之间具有一对多的关系；一个职工对应多张公告单，一个公告单隶属于一个职工；
* 处理，职工与报修单之间具有一对多的关系；一个职工对应多张报修单，一个报修单隶属于一个职工；

**3.2 物理逻辑模型设计**

根据上述分析设计学生宿舍数据库应用系统物理数据模型如下所示：

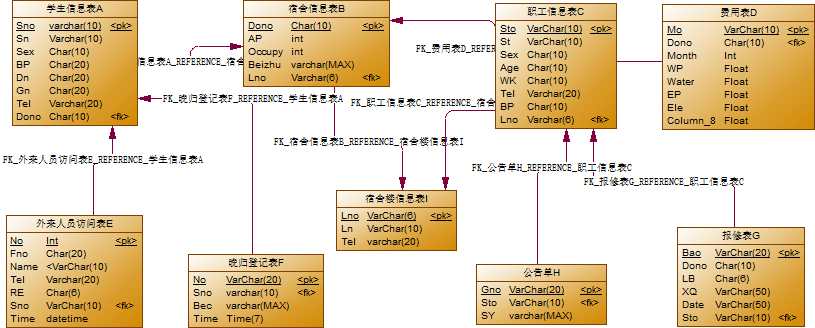


图11 系统物理数据模型图

（1）学生信息表A：将ER图中实体集学生相关属性学号，姓名，学院，班级，籍贯，性别，联系方式，宿舍号，分别抽象为Sno，Sn，Sex，BP，Dn，Gn，Tel，Dono字段。

（2）宿舍信息表B：将ER图中实体集宿舍相关属性宿舍号，容量，已住人数，楼号，分别抽象为Dono，AP，Occupy，Lno字段。

（3）职工信息表C：将ER图中实体集职工相关属性职工号，姓名，性别，年龄，职务，联系方式，职务，楼号，分别抽象为Sto,St,Sex,Age,WK,Tel, BP ,Lno字段。

（4）费用表D：将ER图中实体集费用相关属性费用单号，宿舍号，月份，水单价，用水量，电单价，用电量，分别抽象为Mo,Dono,Month,WP,Water,EP,Ele字段。

（5）外来人员访问表E：将ER图中实体集外来人员相关属性访问登记号，身份证号，姓名，联系方式，亲属关系，访问时间，访问人学号，分别抽象为No,Fno,Name,Tel, RE,Time, Sno字段。

（6）晚归登记表F：将ER图中实体集晚归相关属性登记单号，学号，晚归原因，晚归时间，分别抽象为No,Sno,Bec,Time字段。

（7）报修表G：将ER图中实体集报修相关属性报修单号，宿舍号，维修类别，详情，解决日期，职工号，分别抽象为Bao,Dono,LB,XQ,Date,Sto字段。

（8）公告单H：将ER图中实体集公告相关属性公告单号，职工号，具体事宜，分别抽象为Gno,Sto,SY字段。

4 关系结构设计

**4.1 关系模式设计和表结构**

4.1.1 关系模式设计

* **学生信息表A(Sno，Sn，Sex，BP，Dn，Gn，Tel，Dono)**

主键：Sno

中文含义：学生（学号，姓名，学院，班级，籍贯，性别，联系方式，宿舍号）

表中字段类型与长度选择的依据：Sno，Sn，Sex，BP，Dn，Gn，Tel，Dono的类型皆定义为字符串类型。其中，Sno、Sn定义为varchar类型，Sex，BP，Dn，Gn，Dono定义为char类型。其中，Dono参照宿舍信息表B（Dono），为本关系外键。

* **宿舍信息表B(Dono，AP，Occupy，Lno)**

主键：Dono

中文含义：宿舍（宿舍号，容量，已住人数，楼号）

表中字段类型与长度选择的依据：Dono，Lno，Beizhu定义为字符串类型。AP，Occupy定义为整数型。其中，Lno参照宿舍楼信息表I（Lno），为本关系外键。

* **职工信息表C(Sto,St,Sex,Age,WK,Tel, BP ,Lno)**

主键：Sto

中文含义：职工（职工号，姓名，性别，年龄，职务，联系方式，职务，楼号）

表中字段类型与长度选择的依据：Sto，St，Sex，Age，WK，Tel，Lno，BP定义为字符串类型。其中，Lno参照宿舍楼信息表I（Lno），为本关系外键。

* **宿舍楼信息表I(Lno,Ln,Tel)**

主键：Lno

中文含义：宿舍楼（楼号，楼长，楼长联系方式）

表中字段类型与长度选择的依据：Sto，St，Sex，WK，Tel，Lno，BP定义为字符串类型。Age定义为整数型。

* **费用表D(Mo,Dono,Month,WP,Water,EP,Ele)**

主键：Mo

中文含义：费用表（费用单号，宿舍号，月份，水单价，用水量，电单价，用电量）

表中字段类型与长度选择的依据：Mo，Dono定义为字符串类型。Month定义为整数型。WP，Water，EP，Ele定义为浮点数类型。其中，Dono参照宿舍信息表B（Dono），为本关系外键。

* **外来人员访问表E(No,Fno,Name,Tel,RE,Time,Sno)**

主键：No

中文含义：外来人员访问表（访问登记号，身份证号，姓名，联系方式，亲属关系，访问时间，访问人学号）

表中字段类型与长度选择的依据：Fno，Name，Tel，RE定义为字符串类型。No定义为整数型。Time定义为datatime类型。其中，Sno参照宿舍信息表A（Sno），为本关系外键。

* **晚归登记表F(No,Sno,Bec,Time)**

主键：No

中文含义：晚归登记表F（登记单号，学号，晚归原因，晚归时间）

表中字段类型与长度选择的依据：No，Sno，Bec定义为字符串类型。Time定义为time(7)类型。其中，Sno参照宿舍信息表A（Sno），为本关系外键。

* **报修表G(Bao,Dono,LB,XQ,Date,Sto)**

主键：Bao

中文含义：报修表（报修单号，宿舍号，维修类别，详情，解决日期，职工号）

表中字段类型与长度选择的依据：Bao，Dono，LB，XQ，Date，Sto定义为字符串类型。其中，Sto参照职工表C（Sto），为本关系外键。

• **公告单H(Gno,Sto,SY)**

主键：Gno

中文含义：公告单（公告单号，职工号，具体事宜）

表中字段类型与长度选择的依据：Gno，Sto，SY定义为字符串类型。其中，Sto参照职工表C（Sto），为本关系外键。

4.1.2 表结构设计

表格1 学生信息表A的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Sno | 学号 | varchar(10) | 主键：PRIMARY KEY |
| Sn | 姓名 | Varchar(10) | NOT NULL |
| Sex | 性别 | Char(10) | ‘男‘ or ‘女‘ |
| BP | 籍贯 | Char(20) |  |
| Dn | 学院 | Char(20) | NOT NULL |
| Gn | 班级 | Char(20) | NOT NULL |
| Tel | 联系方式 | Varchar(20) |  |
| Dono | 宿舍号 | Char(10) | 外建：参照B表的Dono字段 |

表格2 学生信息表A的的实例

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Sno | Sn | Sex | BP | Dn | Gn | Dono | Tel |
| 171001201 | 韩宝轩 | 男 北京 | 北京 | 信息学院 | 信息17-2 | 12 #805 | NULL |
| 171001202方雅浩 | 方雅浩 | 男 | 广东 | 信息学院 | 信息17-2 | 12#805 | NULL |
| 171001203 | 宋守诚 | 男 | 河南 | 信息学院 | 信息17-2 | 12#805 | 13811316398 |
| 171001204 | 武江浩 | 男 | 河北 | 信息学院 | 信息17-2 | 12#805 | 18501926388 |
| 171001206 | 王胤钦 | 男 | 江西 | 信息学院 | 信息17-2 | 12#806 | NULL |
| 171001208 | 王泽昌 | 男 | 山东 | 信息学院 | 信息17-2 | 12#806 | NULL |
| 171001210 | 祝志炳 | 男 | 陕西 | 信息学院 | 信息17-2 | 12#806 | NULL |
| 171001212 | 毛江啸 | 男 | 浙江 | 信息学院 | 信息17-2 | 12#806 | 13730584597 |
| 171001214 | 季晓阳 | 女 | 北京 | 信息学院 | 信息17-2 | 13#20810 | NULL |
| 171001215 | 张珮苇 | 女 | 甘肃 | 信息学院 | 信息17-2 | 13#20809 | NULL |
| 171001216 | 陆潇悦 | 女 | 广西 | 信息学院 | 信息17-2 | 13#20809 | NULL |
| 171001217 | 曹煜珺 | 女 | 贵州 | 信息学院 | 信息17-2 | 13#20810 | 13718694182 |
| 171001218 | 赛竞艳 | 女 | 河南 | 信息学院 | 信息17-2 | 13#20809 | NULL |
| 171001219 | 顾涵 | 女 | 河南 | 信息学院 | 信息17-2 | 13#20810 | 18811719073 |
| 171001220 | 杨楚峤 | 女 | 黑龙江 | 信息学院 | 信息17-2 | 13#20809 | 15611266155 |
| 171001221 | 王诗缘 | 女 | 湖北 | 信息学院 | 信息17-2 | 13#20809 | 17710629021 |
| 171001222 | 姜洪微 | 女 | 吉林 | 信息学院 | 信息17-2 | 13#20810 | NULL |
| 171001223 | 殷悦 | 女 | 江苏 | 信息学院 | 信息17-2 | 13#20810 | 15501090218 |
| 171001224 | 刘雅洁 | 女 | 青海 | 信息学院 | 信息17-2 | 13#20809 | 18810095528 |
| 176003101 | 李欣然 | 男 | 海南 | 林学院 | 林学17-1 | 12#805 | NULL |
| 176003102 | 赵力康 | 男 | 西藏 | 林学院 | 林学17-1 | 12#805 | NULL |
| 186003201 | 程大壮 | 男 | 湖南 | 林学院 | 林学17-2 | 12#805 | 18801276068 |
| 186003202 | 王国庆 | 男 | 辽宁 | 林学院 | 林学17-2 | 12#805 | NULL |
| 171001213 | 金欣然 | 女 | 北京 | 材料学院 | 林工17-1 | 13#20814 | NULL |

表格3 宿舍信息表B的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Dono | 宿舍号 | Char(10) | 主键：PRIMARY KEY |
| AP | 容量 | int | NOT NULL |
| Occupy | 已住人数 | int |  |
| Beizhu | 宿舍备注 | varchar(MAX) |  |
| Lno | 楼号 | Varchar(6) | 外建：参照I表的Lno字段 |

表格4 宿舍信息表B的的实例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dono  Sn  Sex | AP  Dn  Cn | Occupy  Dono | Lno | Beizhu |
| 12#805 | 8 | 8 | 12 | 2019年5月检查宿舍卫生不合格； |
| 12#806 | 6 | 4 | 12 |  |
| 13#20809 | 8 | 5 | 13 | 2019年6月曾发生过宿舍矛盾； |
| 13#20810 | 8 | 6 | 13 |  |
| 13#20814 | 8 | 1 | 13 |  |

表格5 职工信息表C的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Sto | 职工号 | VarChar(10) | 主建：PRIMARY KEY |
| St | 姓名 | VarChar(10) | NOT NULL |
| Sex | 性别 | Char(10) | ‘男‘ or ‘女‘ |
| Age | 年龄 | Char(10) |  |
| WK | 职务 | Char(10) |  |
| Tel | 联系方式 | Varchar(20) |  |
| BP | 籍贯 | Char(10) |  |
| Lno | 楼号 | Varchar(6) | 外建：参照I表的Lno字段 |

表格6 职工信息表C的的实例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sto | St | Sex | Age | WK  Cn | Tel  Dono | BP |
| 0001 | 李艳梅 | 女 | 35 | 保洁部经理 | 650618 | 黑龙江 |
| 0002 | 王小丽 | 女 | 45 | 维修部经理 | 650620 | 辽宁 |
| 0003 | 彭燕燕 | 女 | 38 | 后勤部经理 | 620618 | 吉林 |
| 0004 | 杨立国 | 男 | 41 | 宿舍管理员 | 630618 | 浙江 |
| 0005 | 赵大强 | 男 | 44 | 宿舍管理员 | 650718 | 江苏 |
| 0006 | 陆爱花 | 女 | 55 | 宿舍管理员 | 650667 | 河南 |

表格7 费用表D的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Mo | 费用单号 | VarChar(10) | 主建：PRIMARY KEY |
| Dono | 宿舍号 | Char(10) | 外建：参照B表的Dono字段 |
| Month | 月份 | Int | 取值：1-12 |
| WP | 水单价 | Float |  |
| Water | 用水量 | Float |  |
| EP | 电单价 | Float |  |
| Ele | 用电量 | Float |  |

表格8 费用表D的的实例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mo | Dono | Month | WP | Water  Cn | EP | Ele  Dono |
| 1 | 13#20809 | 9 | 2.8 | 1 | 0.5 | 240 |
| 10 | 13#20814 | 10 | 2.7 | 0.98 | 0.6 | 220 |
| 2 | 13#20810 | 9 | 2.8 | 0.8 | 0.5 | 280 |
| 3 | 12#805 | 9 | 2.8 | 1.2 | 0.5 | 320 |
| 4 | 12#806 | 9 | 2.8 | 1 | 0.5 | 300 |
| 5 | 13#20814 | 9 | 2.8 | 0.6 | 0.5 | 250 |
| 6 | 13#20809 | 10 | 2.7 | 1 | 0.6 | 260 |
| 7 | 13#20810 | 10 | 2.7 | 1 | 0.6 | 250 |
| 8 | 12#805 | 10 | 2.7 | 1.1 | 0.6 | 210 |
| 9 | 12#806 | 10 | 2.7 | 0.7 | 0.6 | 190 |

表格9 外来人员访问表E的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| No | 表单号 | Int | 主建：PRIMARY KEY |
| Fno | 身份证号 | Char(20) |  |
| Name | 姓名 | VarChar(10) | NOT NULL |
| Tel | 联系方式 | Varchar(20) | NOT NULL |
| RE | 亲属关系 | Char(6) | NOT NULL |
| Sno | 访问人学号 | VarChar(10) | 外建：参照A表的Sno字段 |
| Time | 访问时间 | datetime | NOT NULL |

表格10 外来人员访问表E的的实例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Fno | Name | Tel | RE  Cn | Sno | Time  Dono |
| 1 | 11011196802023444 | 李桂秀 | 15612124567 | 母女 | 171001204 | 2019-09-14 12:59:59.000 |
| 2 | 212020197401010111 | 杨国民 | 13945547710 | 父女 | 171001220 | 2019-10-01 21:23:59.000 |
| 3 | 345678199901030897 | 谢贺 | 15611266155 | 朋友 | 171001220 | 2019-10-03 08:00:23.000 |
| 4 | 346211200007051221 | 王俊凯 | 13767850987 | 朋友 | 171001220 | 2019-10-21 18:12:58.977 |
| 5 | 521521199911211314 | 易烊千玺 | 17818191314 | 恋人 | 171001216 | 2019-10-18 13:14:52.000 |
| 6 | 562162198609270101 | 华晨宇 | 17713140686 | 朋友 | 171001220 | 2019-10-28 10:32:34.457 |
| 7 | 676768200010140811 | 赛争奇 | 18816719090 | 姐妹 | 171001218 | 2019-10-14 07:56:14.000 |

表格11 晚归登记表F的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| No | 登记单号 | VarChar(20) | 主建：PRIMARY KEY |
| Sno | 学号 | varchar(10) | 外建：参照A表的Sno字段 |
| Bec | 晚归原因 | varchar(MAX) | NOT NULL |
| Time | 晚归时间 | Time(7) | NOT NULL |

表格12 晚归登记表F的的实例

|  |  |  |  |
| --- | --- | --- | --- |
| No  Sn  Sex | Sno  Dn  Cn | Bec  Dono | Time |
| 2019091201 | 171001220 | 外出学习 | 23:45:23 |
| 2019091202 | 171001218 | 外出学习 | 23:46:46 |
| 2019092401 | 171001215 | 学生工作 | 23:58:30 |
| 2019092501 | 171001216 | 休闲娱乐 | 00:45:56 |
| 2019092502 | 171001220 | 外出学习 | 23:58:18 |
| 2019092503 | 186003202 | 学生工作 | 23:59:00 |
| 2019100101 | 171001220 | 休闲娱乐 | 23:57:32 |
| 2019102601 | 171001220 | 外出学习 | 00:12:34 |
| 2019102701 | 171001216 | 休闲娱乐 | 00:56:54 |

表格13 报修表G的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Bao | 报修单号 | VarChar(20) | 主建：PRIMARY KEY |
| Dono | 宿舍号 | Char(10) | 外建：参照B表的Dono字段 |
| LB | 维修类别 | Char(6) | NOT NULL |
| XQ | 详情 | VarChar(50) |  |
| Date | 解决日期 | varchar(50) |  |
| Sto | 职工号 | VarChar(10) | 外建：参照C表的Sto字段 |

表格14 报修表G的的实例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bao | Dono | LB | XQ | Date | Sto |
| 2019101101 | 13#20809 | 水 | NULL | 2019-10-12 | 0001 |
| 2019101102 | 12#805 | 电 | NULL | 2019-10-12 | 0002 |
| 2019101401 | 12#806 | 门窗 | NULL | 2019-10-15 | 0005 |
| 2019101402 | 13#20809 | 电 | 突然无电 | 2019-10-15 | 0003 |
| 2019101801 | 13#20814 | 门窗 | 窗户玻璃破裂 | 2019-10-18 | 0001 |

表格15 公告单H的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Gno | 公告单号 | VarChar(20) | 主建：PRIMARY KEY |
| Sto | 职工号 | VarChar(10) | 外建：参照C表的Sto字段 |
| SY | 具体事宜 | varchar(MAX) | NOT NULL |

表格16 公告单H的的实例

|  |  |  |
| --- | --- | --- |
| Gno  Sn  Sex | Sto  Dn  Cn | SY  Dono |
| 2019101101 | 0001 | 【停电通知】为确保广大同学们宿舍用电安全，10月11日晚23：30-10月12日早6：00，12号、13号宿舍楼将进行高压电检修工作，检修期间全楼停水停电，请同学们提前做好准备！ |
| 2019103001 | 0001 | 【电梯检修】10月31号，11号宿舍楼东侧2号、3号电梯全天维修，请同学们乘坐其他电梯通行！ |
| 2019110101 | 0002 | 【供暖通知】学校将于11月1日-11月5日陆续对各个宿舍进行供暖试水工作，特此通知！ |
| 2019110102 | 0001 | 【违纪通报】朱小云同学，经管学院本科生，2019年10月18日被发现在学13宿舍楼11516宿舍违规使用电夹板，被记以警告处分，特此通报！请同学们严格遵守宿舍规章纪律！ |

表格17 宿舍楼信息表I的关系结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Lno | 楼号 | VarChar(6) | 主建：PRIMARY KEY |
| Ln | 楼长 | VarChar(10) |  |
| Tel | 联系方式 | varchar(20) |  |

表格18 宿舍楼信息表I的的实例

|  |  |  |
| --- | --- | --- |
| Lno  Sn  Sex | Ln  Cn | Tel |
| 6 | 安黎哲 | 15611266154 |
| 12 | 谢学文 | 18711561667 |
| 13 | 王洪元 | 15266789192 |

为了提高在表中搜索元组的速度，在实际实现的时候应该基于某些属性建立索引，下面是基于数据库实际需求所建立的索引：

（1）对A表建立索引

CREATE INDEX Dono\_ind

ON A (Dono)

GO

（2）对D表建立索引

CREATE INDEX Dono\_ind

ON D (Dono)

GO

**4.2 数据库和表创建**

4.2.1 数据库创建的SQL语句

GO

CREATE DATABASE YCQ

on primary

(

name= 'YCQ',

filename= 'C:\数据库\YCQ.mdf',

size= 5120KB,

maxsize= UNLIMITED,

filegrowth= 1024KB

)

log on

(

name=' YCQ\_log',

filename=' C:\数据库\YCQ\_log.ldf',

size= 1024KB,

maxsize= 2048GB,

filegrowth= 10%

)

GO

4.2.2 数据库中每个表创建的SQL语句

(1) 学生信息表A

CREATE TABLE A

( Sno varchar(10) PRIMARY KEY,

Sn varchar(10) NOT NULL,

Sex char(10) CHECK(Sex= '男' OR Sex= '女') DEFAULT ('男'),

BP char(20),

Dn char(20) NOT NULL,

Gn char(20) NOT NULL,

Dono Char(10) References B(Dono),

Tel varchar(20),

)

(2) 宿舍信息表B

CREATE TABLE B

( Dono char(10) PRIMARY KEY,

AP int NOT NULL,

Occupy int,

Beizhu varchar(MAX) ,

Lno varchar(6) References I(Lno)

)

(3) 职工信息表C

CREATE TABLE C

( Sto varchar(10) PRIMARY KEY,

St varchar(10) NOT NULL,

Sex char(10) CHECK(Sex= '男' OR Sex= '女') DEFAULT ('男'),

Age Char(10) ,

WK char(10) ,

Tel varchar (20),

BP Char(10) ,

Lno varchar(6) References I(Lno)

)

(4) 费用表D

CREATE TABLE D

( Mo varchar(10) PRIMARY KEY,

Dono char(10) References B(Dono),

Month int CHECK(Month >= 1 OR Month <= 12) DEFAULT (1),

WP float,

Water Float,

EP float,

Ele Float

)

(5) 外来人员访问表E

CREATE TABLE E

( No Int PRIMARY KEY

Fno char(20),

Name varchar (10) NOT NULL,

Tel varchar (20) NOT NULL,

RE char (6) NOT NULL,

Sno varchar(10) References A(Sno),

Time datetime NOT NULL

)

(6) 晚归登记表F

CREATE TABLE F

( No varchar(20) PRIMARY KEY,

Sno varchar (10) References A(Sno),

Bec varchar (max) ,

Time Time(7),

)

(7) 报修表G

CREATE TABLE G

( Bao varchar (20) PRIMARY KEY,

Dono char (10) References B(Dono),

LB char (6) NOT NULL,

XQ varchar(50),

Date varchar(50),

Sto varchar(10) References C(Sto)

)

(8) 公告单H

CREATE TABLE H

( Gno varchar (20) PRIMARY KEY,

Sto varchar (10) References C(Sto),

SY varchar(max) NOT NULL

)

(9) 宿舍楼信息表I

CREATE TABLE I

( Lno varchar (6) PRIMARY KEY,

Ln varchar (10),

Tel varchar(20)

)

4.2.3 视图创建的SQL语句

（1）报修信息视图

可见用户：宿舍管理员

功能：查询全部报修单，报修宿舍以及相关职工详细信息

SQL语句：

create view v\_baoxiu(报修申请日期,报修类别,报修详情,宿舍号,楼号,解决日期,解决人姓名,解决人联系方式)

as

select left(G.Bao,8),G.lb,G.XQ,G.Dono,B.Lno,G.Date,C.St,C.Tel

from G join B on G.Dono=B.Dono join C on G.sto=C.Sto

（2）晚归信息视图

可见用户：宿舍管理员

功能：查询晚归学生名单以及个人详细信息

SQL语句：

create view v\_wangui(学生学号,学生姓名,晚归日期,晚归时间,晚归原因,学院,班级,宿舍号,楼号)

as

select A.Sno,A.Sn,left(F.No,8),F.Time,F.Bec,A.Dn,A.Gn,A.Dono,B.Lno

from A join F on A.Sno=F.Sno join B on A.Dono=B.Dono

（3）来访信息视图

可见用户：宿舍管理员

功能：查询外来人员来访名单以及相关学生个人详细信息

SQL语句：

create view v\_laifang(表单号,姓名,身份证号,联系方式,访问人学号,访问学生姓名,学生宿舍号,学生联系方式,关系,访问时间)

as

select E.no,E.name,E.Fno,E.tel, A.Sno ,A.Sn,A.Dono,A.tel,E.RE,E.Time

from E join A on E.Sno=A.Sno

（4）公告信息视图

可见用户：系统全部用户

功能：查询公告发布记录以及发布人相关信息

SQL语句：

create view v\_gonggao(公告时间,公告内容,发布人姓名,发布人职务,发布人联系方式)

as

select left(H.Gno,8),H.SY,C.ST,C.WK,C.Tel

from H join C on H.STo=C.Sto

（5）费用信息视图

可见用户：系统全部用户

功能：查询宿舍费用相关信息

SQL语句：

CREATE VIEW [dbo].[v\_feiyong] (宿舍号,已住人数,月份,总电费, 人均费用)

AS

SELECT B.Dono, B.AP, D.Month,SUM(D.EP\*D.Ele), SUM(D.EP\*D.Ele)/B.AP

FROM B JOIN D ON B.Dono=D.Dono

GROUP BY B.Dono, B.AP, D.Month;

**4.3 查询与统计**

在系统中调用了一些select查询语句，主要用于从数据库中查询所需要的数据。

（1）查询宿舍费用信息

select B.Dono,B.Occupy,SUM(D.EP\*D.Ele), MAX(D.Ele),MIN(D.Ele)

from B join D on B.Dono=D.Dono

group by B.Dono,B.Occupy

order by SUM(D.EP\*D.Ele)

（2）查询晚归学生信息

select A.sno,A.sn,A.dn,A.gn,count(F.sno)

from F join A on F.sno=A.sno

group by A.sno,A.sn,A.dn,A.gn

order by count(F.sno)

**4.4 触发器设计与实现**

（1）创建触发器trigger\_addstudent

**文字描述：**每当宿舍有一名新同学办理入住时，宿舍管理员就会向学生信息表A中插入一条记录，触发器自动将宿舍信息表中对应的宿舍号记录的已住人数加1。

**触发器实现：**

create trigger trigger\_addstudent on A

for insert

as

declare @m\_dono char(10),@num int

select @m\_dono=dono from inserted

update B set Occupy=Occupy+@num

where dono=@m\_dono

（2）创建触发器trigger\_ deletestudent

**文字描述：**每当宿舍有一名同学办理离宿时，宿舍管理员就会在学生信息表A中删除该学生的个人信息，触发器自动将宿舍信息表中对应的宿舍号记录的已住人数减1。

**触发器实现：**

create trigger trigger\_deletestudent on A

for delete

as

declare @m\_dono char(10),@num int

select @m\_dono=dono from deleted

update B set Occupy=Occupy-@num

where dono=@m\_dono

**4.5 存储过程设计与实现**

（1）创建一个存储过程pro\_laifang

**文字描述：**实现外来人员来访业务。该事务实现外来人员申请看望宿舍学生a的功能。要求程序能够自动审核外来人员填写的a同学的信息与学生信息表中信息是否匹配。

**存储过程实现：**

create procedure pro\_laifang

@m\_name varchar(10),@m\_tel varchar(20),

@m\_Fno char(20),@m\_sno varchar(10),@m\_sn varchar(10),

@m\_dono char(10),@m\_re char(6)

as

if exists(select \* from A where sno=@m\_sno and sn=@m\_sn and dono=@m\_dono)

begin

declare @m int

select @m=no from E

insert into E(no,fno,name,tel,re,sno,time)

values(@m+1,@m\_Fno,@m\_name,@m\_tel,@m\_re,@m\_sno,getdate())

end

else

return 404

GO

**代码段：**

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_laifang '" + name + "','" +tel+ "','" + fno+ "','" + sno+ "','" + sn+ "','" + dono+ "','" + re + "'";

**实例：**

****

图12 存储过程pro\_laifang

（2）创建一个存储过程pro\_student。

**文字描述**：根据所给出的姓名、年龄等字段分别进行精确查询，实现字段作为动态参数，按照输入要求，可自动筛选。

**存储过程实现：**

create procedure pro\_student

@m1 varchar(100),@m2 varchar(100)

as

begin

declare @sql varchar(100)

set @sql='select \* from A where '+@m1+'='+@m2

exec(@sql)

end

**代码段：**

DataTable dt = new DataTable();//存储查询出来的记录

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_student '" + t1 + "','" + t2 + "'";

dt = db1.runSql(sql1);

**实例：**



图13 存储过程pro\_student

（3）创建一个存储过程pro\_zhigong。

**文字描述**：根据所给出的职工号号等字段分别进行精确查询，实现字段作为动态参数，按照输入要求，可自动筛选。

**存储过程实现：**

create procedure pro\_zhigong

@m1 varchar(100),@m2 varchar(100)

as

begin

declare @sql varchar(100)

set @sql='select \* from C where '+@m1+'='+@m2

exec(@sql)

end

**代码段：**

DataTable dt1 = new DataTable();//存储查询出来的记录

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_zhigong '" + t4 + "','" + t3 + "'";

dt1 = db1.runSql(sql1);

dataGridView2.DataSource = dt1;

**实例：**



图14 存储过程pro\_zhigong

（4）更新用户密码pro\_pwd。

**文字描述**：创建一个存储过程pro\_pwd，使得修改用户密码时可以自行判断是否拥有修改权限。

**存储过程实现：**

Create procedure pro\_pwd

@m1 varchar(10),@m2 varchar(30),@m3 varchar(30),@m4 varchar(30)

as

if exists(select \* from U where name=@m1 and Pwd=@m2 and @m3=@m4)

begin

update U set pwd=@m3 where name=@m1

end

**代码段：**

private void btnDRevert\_Click(object sender, EventArgs e)

{

string m1 = this.textBox1.Text.Trim();

string m2 = this.txtUOPwd.Text.Trim();

string m3 = this.txtUNPwd.Text.Trim();

string m4 = this.txtFUNPwd.Text.Trim();

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_pwd '" + m1 + "','" + m2 + "','" + m3+ "','" + m4 + "'";

int j = 0;

j = db1.executeSqlResult(sql1);

if (j > 0)

{

MessageBox.Show("修改成功！！");}

Else{

MessageBox.Show("修改失败！！");

return;}}

**实例：**

****

图15 存储过程pro\_pwd

（5）创建存储过程pro\_wanguichaxun

**文字描述：**查询所有学生的学号，姓名，学院，班级，以及晚归总次数，并按照晚归次数降序排序。

**存储过程实现：**

create procedure pro\_wanguichaxun

as

begin

select A.sno,A.sn,A.dn,A.gn,count(F.sno)

from F join A on F.sno=A.sno

group by A.sno,A.sn,A.dn,A.gn

order by count(F.sno)

end

**代码段：**

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_wanguichaxun ";

dt = db1.runSql(sql1);

**实例：**



图16 存储过程pro\_wanguichaxun

（6）创建存储过程pro\_feiyong

**文字描述：**查询所有宿舍的宿舍号，已住人数，和总电费，以及单月最高用电量和单月最低用电量，按照总电费金额降序排列。

**存储过程实现：**

create procedure pro\_feiyong

as

begin

select B.Dono,B.Occupy,SUM(D.EP\*D.Ele), MAX(D.Ele),MIN(D.Ele)

from B join D on B.Dono=D.Dono

group by B.Dono,B.Occupy

order by SUM(D.EP\*D.Ele)

end

**代码段：**

string sql2 = "execute pro\_feiyong ";

dt = db1.runSql(sql2);

**实例：**



图17 存储过程pro\_feiyong

（8）创建一个存储过程pro\_wangui

**文字描述：**实现晚归学生登记。该事务实现晚归学生登记个人信息功能。要求程序能够自动审核外来学生填写个人信息与注册信息是否匹配，可防止同学填写错误信息。

**存储过程实现**：

create procedure pro\_wangui

@m\_sno varchar(10), @m\_sn varchar(10),

@m\_lno varchar(6),@m\_dono char(10),@m\_bec varchar(MAX),

@m varchar(20)

as

if exists(select \* from A where A.sno=@m\_sno and A.sn=@m\_sn and A.dono=@m\_dono )

begin

insert into F(no,sno,bec,time)

values(CONVERT(varchar(100), GETDATE(), 112)+@m,

@m\_sno,@m\_bec,CONVERT(varchar(100), GETDATE(), 24))

end

else

return 404

go

**代码段：**

string sql12 = "execute pro\_wangui '" + sno + "','" + sn + "','" + lno + "','" + dono + "','" + bec + "','" + no + "'";

**实例：**

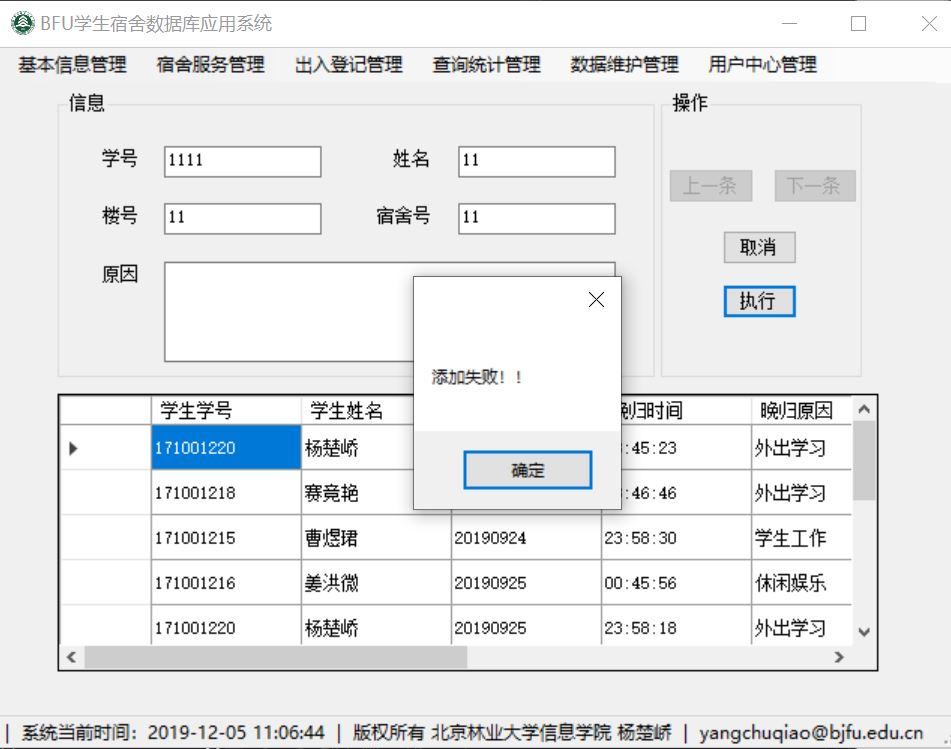


图18 存储过程pro\_wangui

5 数据库安全性策略设计

**5.1 数据库角色及权限设计**

一个好的数据库管理系统不仅要具备良好的操作界面、全面系统的功能，更要有良好的安全性措施来保障系统的顺利运行。而对于学生宿舍数据库应用系统，考虑到不同角色的用户拥有不同的权限，故设置了实际中存在的四类用户。

表格 19 用户权限总览表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表或视图 | 宿舍管理员 | 维修人员 | 保安人员 | 宿舍学生 |
| 学生信息表（A） | 全部 | \* | 只读 | \* |
| 宿舍信息表（B） | 全部 | 只读 | 只读 | \* |
| 宿舍楼信息表（I） | 全部 | 只读 | 只读 | 只读 |
| 职工信息表（C） | 全部 | 全部 | 全部 | \* |
| 宿舍费用表（D） | 全部 | \* | \* | 只读 |
| 来访登记表（E） | 全部 | \* | \* | 只写 |
| 晚归登记表（F） | 全部 | \* | \* | 只写 |
| 宿舍报修表（G） | 全部 | \* | \* | 全部 |
| 公告信息表（H） | 全部 | 全部 | 全部 | 只读 |
| 报修信息视图（v\_baoxiu） | 全部 | 全部 | 全部 | 全部 |
| 费用信息视图（v\_feiyong） | 全部 | \* | \* | 只读 |
| 公告信息视图（v\_gonggao） | 全部 | 全部 | 全部 | 只读 |
| 来访信息视图（v\_laifang） | 全部 | \* | 只读 | 只读 |
| 晚归信息视图（v\_wangui） | 全部 | \* | 只读 | 只读 |

**注**：表中\*代表没有操作权限，全部代表有全部操作权限，只读代表有只读权限，只写代表有只写权限。

**5.2 数据库备份**

为防止数据库遭到破坏影响正常的工作与业务，要时常对数据库进行备份，以保障出现问题时可以进行数据库恢复。由于本系统的进销存功能需要对数据库实时进行操作，增删改查操作十分频繁，因此需要保证数据库实时记录改变，并可以将数据库还原到时间点。这就需要使用SQL Server的完整恢复模式。

备份数据库时选择完整备份，因为日志备份必须有完整备份才可以进行。然后选择需要备份的数据库,为每个数据库创建备份文件，并且为每个数据库创建子目录，最后将存储完整备份文件路径设为“C:\\数据库”(注意：此处路径仅作为示例，因为本人电脑只有C盘，实际上存储于D盘更常见)。

6 系统设计与实现

**6.1 系统的架构与功能结构**

6.1.1 系统体系架构

该系统采用客户机/服务器(C/S)二层结构，以ADO.NET为中间件来连接访问数据库。ADO .NET会提供最直接的方法，让开发人员在 .NET Framework 中进行资料存取，ado.net是一组用于和数据源进行交互的面向对象类库。所写应用软件为客户端，可实现数据的前端处理和用户界面。服务器端由数据库管理系统完成完成存取、查询、并发控制、数据恢复等操作。

6.1.2 系统功能结构

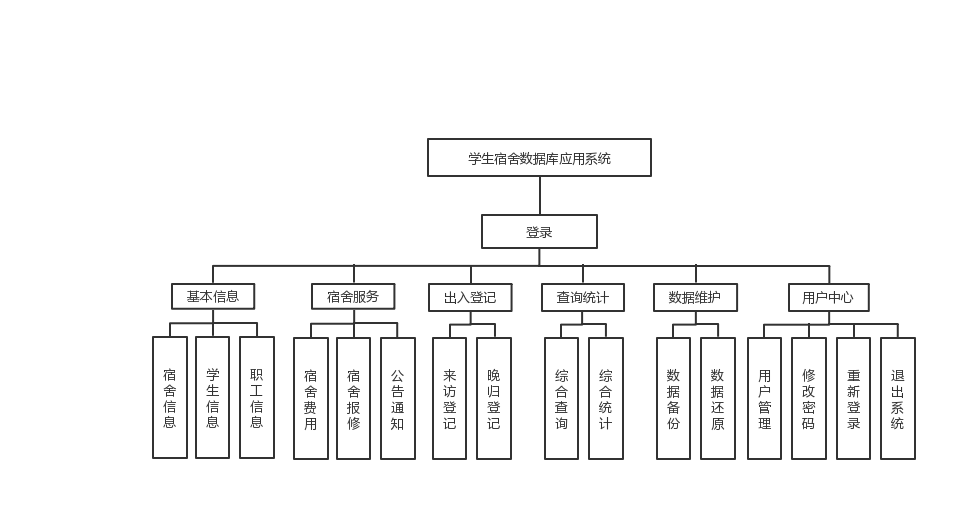


图19 系统功能结构图

（1）基本信息管理

宿舍信息管理：记录宿舍的基本状况，可住人数，已住人数，宿舍长学号、姓名等信息，便于工作人员与宿舍取得联系。

学生信息管理：学生信息主要是由学生处提供的，宿舍管理人员从学生处提取相关信息，进行统一管理，也可对现有的学生信息表进行增删改查。

职工信息管理：记录宿舍管理工作人员的相关信息，及时的增删改查有助于宿舍的管理。

（2）宿舍服务管理

宿舍费用服务：宿舍管理人员会从后勤管理部门获取各宿舍使用费用，并根据现有的价格规定，结算出每个宿舍应该支付的费用。

宿舍报修服务：学生需要把需要维修的设备向宿舍管理员提出申请，填写相应的报修单信息，再由宿舍管理员联系相应部门进行检修。

公告栏服务：宿舍管理人员可以将信息发布在公告栏，通知学生电梯维修，停电通知等日常事宜，方便同学们接收最新消息。

（3）出入登记管理

外来人员登记：宿舍管理人员记录外来人员所登记的信息，系统自动与学生信息表进行息匹配，经过核实，确定可以访问。

学生晚归登记：宿舍管理人员登记晚归学生的个人信息，以及相应缘由，方便宿舍管理。可定期导出晚归学生信息表，由教务处对晚归次数过多的学生进行相应处理。

（4）查询统计管理

综合查询：可以根据用户所选择的筛选条件进行筛选查询。

综合统计：可以根据相应的要求进行数据统计。

（5）数据维护管理

数据备份：备份所选数据表以免数据发生错误而导致数据丢失。

数据还原：如因管理人员操作不当或系统故障等原因导致数据报错，丢失，混乱，可以选择还原之前备份好的数据，充分保护数据的安全可靠。

（6）用户中心管理

用户管理：管理员可以更改用户权限以及增加新用户。

修改密码：用户可以更改自身密码。

重新登录：用户可以重新登陆。

退出系统：用户可以退出系统。

**6.2 系统运行环境、开发技术和语言**

6.2.1 系统运行环境

（1）系统硬件

处理器：Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.60GHz

内存：4.00GB

系统类型：64位操作系统

（2）系统软件

操作系统：Windows 2008 Server

数据库管理系统：SQL Server 2008 R2

6.2.2 系统开发技术

（1）系统开发技术：ADO.NET

（2）系统开发语言：C#

（3）选择C#作为开发语言的原因：C#是基于.NET平台的一种编程开发语言，是一种现代的、面向对象的语言，相较于C++而言更为简便，因而选择C#语言开发。开发时以ADO.NET为中间件来访问SQL数据库。

**6.3 功能模块设计**

6.3.1 用户登录功能

（1）功能：判断系统用户输入的登录账号是否存在，是什么权限，密码输入是否正确，如果都正确，在登录时就可以显示主菜单。注意可以在此设置用户权限，根据不同的用户权限可以对系统以及数据库进行不同的操作。

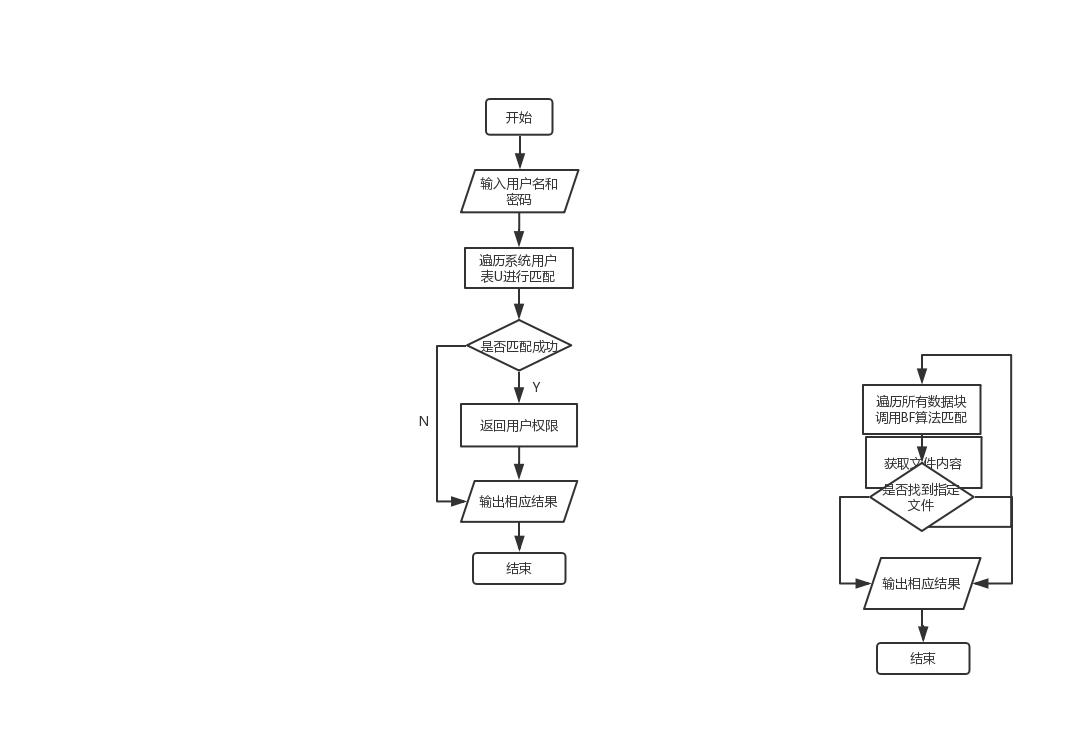


图20 用户登录功能流程图

（2）功能界面：



图21 登录功能界面

（3）功能界面简单文字描述：

① 权限可以实现自动选择，是在click事件中执行了SQL语句查询已输入的登录账户是否存在于U表中，若存在返回其权限。

② 点击“确定”按钮执行SQL语句查询已输入的登录账户是否存在于U表中，且其对应得password是否与用户输入一致。

（4）SQL语句与宿主语言嵌套使用代码：

private void button2\_Click(object sender, EventArgs e){

string m1 = this.txtLKWord.Text.Trim();

string m2 = this.txtPwd.Text.Trim();

DBConnection db1 = new DBConnection();

string sql1 = "select \* from U where name='" + m1 + "'and pwd='" + m2 + "'and power='宿舍管理员'";

int i = 0;

int flag = 1;

i = db1.runSqlInt(sql1);

if (i > 0){

n1 = m1;

p1 = m2;

p2 = "宿舍管理员";

db1.UserLogin("ycq","yangchuqiao");

MessageBox.Show("用户登陆权限为宿舍管理员","提示");

flag = 0;

this.Hide();

Form1 form1 = new Form1();

form1.Show();}

string sql2 = "select \* from U where name='" + m1 + "'and pwd='" + m2 + "'and power='学生'";

int j = 0;

j = db1.runSqlInt(sql2);

if (j > 0) {

n1 = m1;

p1 = m2;

p2 = "学生";

db1.UserLogin("ycq2", "yangchuqiao");

MessageBox.Show("用户登陆权限为学生", "提示");

flag = 0;

this.Hide();

Form1 form1 = new Form1();

form1.Show();}

if(flag==1) {

MessageBox.Show("用户名或者密码输入错误", "提示"); } }

6.3.2 学生信息管理功能

（1）功能：增加、删除、修改、显示学生信息。

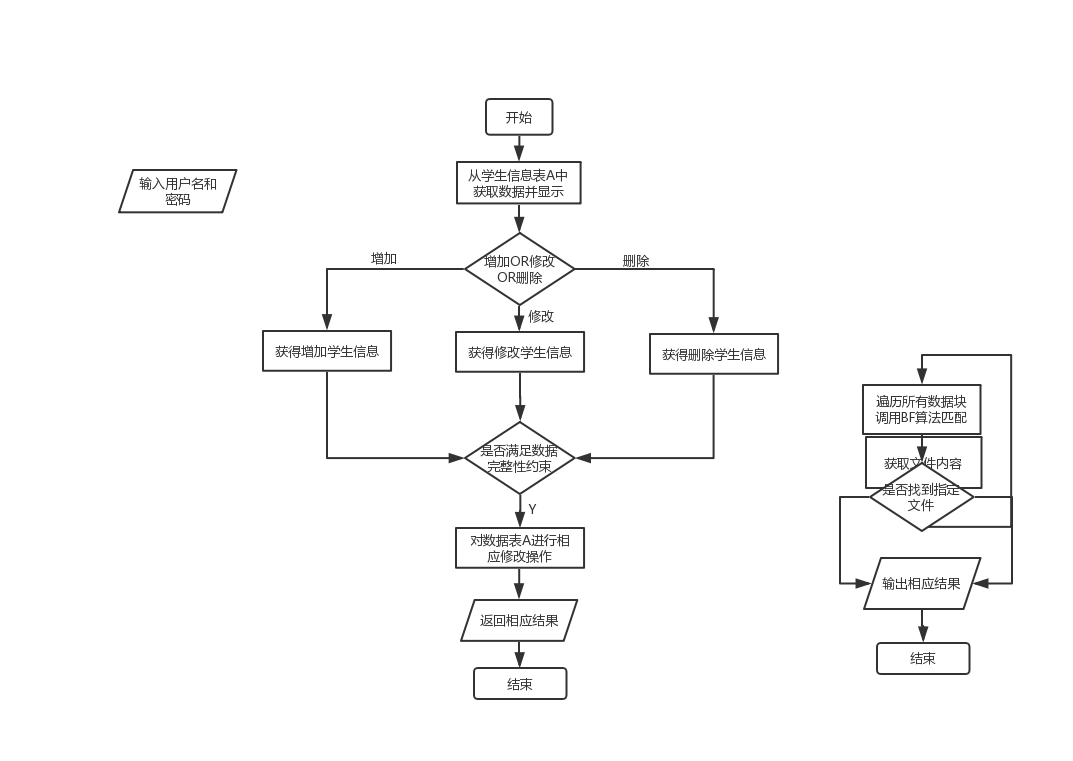


图22 学生信息管理功能流程图

（2）功能界面：



图23 学生信息管理功能界面

（3）功能界面简单文字描述：

① 点击“增加+执行”时会先判断数据库中是否有新添加的学生信息，如果有就无法添加，如果没有就执行添加学生信息。



图24 学生信息管理功能界面

② 点击“修改+执行”按钮可以更改学生信息，但是修改后的信息要符合格式要求，否则无法修改。



图25 学生信息管理功能界面

③点击“删除+执行”按钮删除当前选择的学生信息后，执行delete语句删除该学生信息，如果该学生已经在晚归等表中中有了信息，就无法删除（参照完整性）。



图26 学生信息管理功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button4\_Click(object sender, EventArgs e) {

//增加记录

if (this.button1.Enabled) {

string sn = this.textBox1.Text.Trim();

string sno = this.textBox2.Text.Trim();

string dn = this.textBox3.Text.Trim();

string gn = this.textBox4.Text.Trim();

string sex = this.textBox5.Text.Trim();

string bp = this.textBox6.Text.Trim();

string dono = this.textBox8.Text.Trim();

string tel = this.textBox7.Text.Trim();

if (textBox2.Equals("")){

MessageBox.Show("学号不为空！");

return; }

if (textBox1.Equals("")){

MessageBox.Show("姓名不为空！");

return; }

if (!sex.Equals("男") && !sex.Equals("女")){

MessageBox.Show("性别只能为男或女！");

return; }

if (dn.Length == 0) {

MessageBox.Show("学院不为空！");

return; }

if (gn.Length == 0) {

MessageBox.Show("班级不为空！");

return; }

DBConnection db1 = new DBConnection();

string sql1 = "select \* from A where sno='" + sno + "'"; ;

int i = 0;

i = db1.runSqlInt(sql1);

if (i > 0) {

MessageBox.Show("学号已有！");

return; }

//进行插入记录操作

string sql2 = "insert into A(sn,sno,dn,gn,sex,bp,dono,tel) values('" + sn + "','" + sno + "','" + dn + "','" + gn + "','" + sex + "','" + bp + "','" + dono + "','" + tel + "')";

int j = db1.executeSqlResult(sql2);

if (j > 0) {

MessageBox.Show("添加成功！！");

//刷新datatable

string sql21 = "select \* from A";

dt = db1.runSql(sql21);

if (dt.Rows.Count > 0) {

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt); }}

else{

MessageBox.Show("添加失败！！");

return; } }

else if (this.button2.Enabled) {

//修改

string sn = this.textBox1.Text.Trim();

string sno = this.textBox2.Text.Trim();

string dn = this.textBox3.Text.Trim();

string gn = this.textBox4.Text.Trim();

string sex = this.textBox5.Text.Trim();

string bp = this.textBox6.Text.Trim();

string dono = this.textBox8.Text.Trim();

string tel = this.textBox7.Text.Trim();

if (textBox2.Equals("")){

MessageBox.Show("学号不为空！");

return; }

if (textBox1.Equals("")){

MessageBox.Show("姓名不为空！");

return; }

if (!sex.Equals("男") && !sex.Equals("女"))

{

MessageBox.Show("性别只能为男或女！");

return;

}

if (dn.Length == 0)

{

MessageBox.Show("学院不为空！");

return;

}

if (gn.Length == 0)

{

MessageBox.Show("班级不为空！");

return;

}

if (MessageBox.Show("您真的要更新吗？", "请确认", MessageBoxButtons.YesNo) == DialogResult.Yes) {

string sql3 = "update A set sn='" + sn + "',sex='" + sex + "',bp='" + bp + "',tel='" + tel + "', dono='" + dono + "', gn='" + gn + "' ,dn='" + dn + "'where sno='" + sno + "'";

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult(sql3);

if (m > 0)

{

MessageBox.Show("更新成功！");

//重新绑定新的datatable

string sql31 = "select \* from A";

dt = db1.runSql(sql31);

if (dt.Rows.Count > 0){

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);}}

else{

MessageBox.Show("更新失败！");

return;}}}

else if (this.button3.Enabled)

{

if (MessageBox.Show("您真的要删除吗？", "此删除不可恢复", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

string sno = this.textBox2.Text.Trim();

string sql4 = "delete from A where sno='" + sno + "'";

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult(sql4);

if (m > 0) {

MessageBox.Show("删除成功！");

string sql41 = "select \* from A";

dt = db1.runSql(sql41);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);}}

else {

MessageBox.Show("删除失败！");

return;}}}}

6.3.3 宿舍信息管理功能

（1）功能：增加、删除、修改、显示宿舍信息。

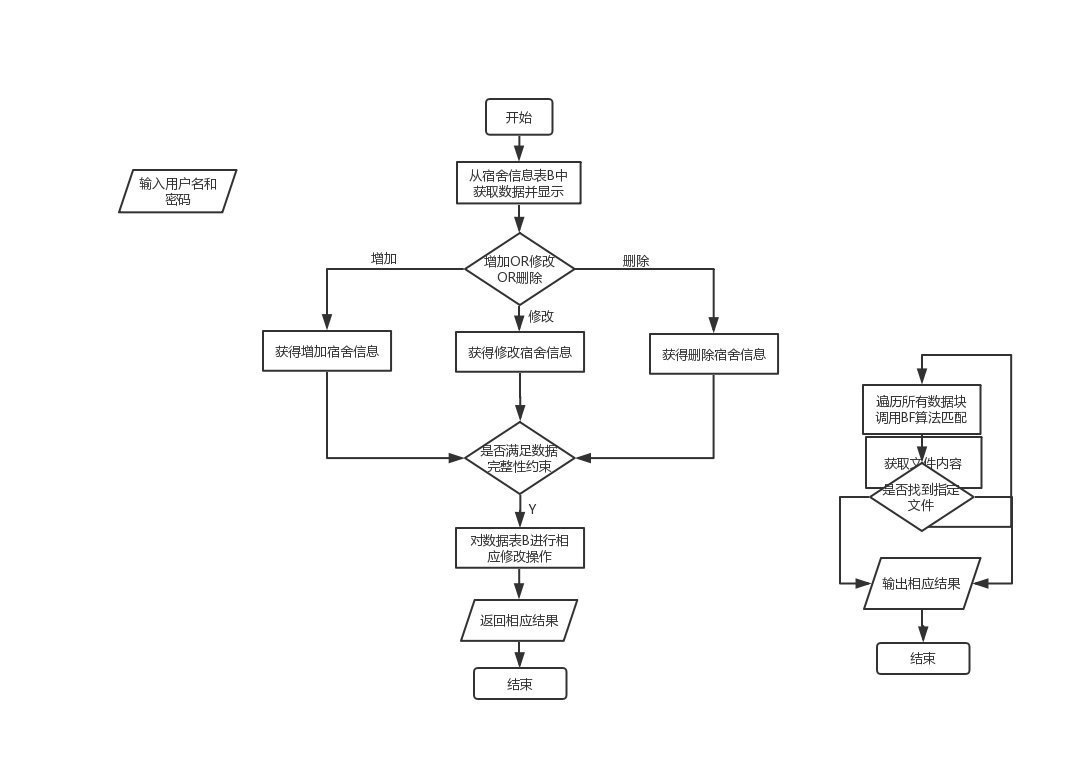


图27 宿舍信息管理功能

（2）功能界面：



图28 学生宿舍管理功能界面

（3）功能界面简单文字描述：

① 点击“增加+执行”时会先判断数据库中是否有新添加的宿舍信息，如果有就无法添加，如果没有就执行添加宿舍信息。

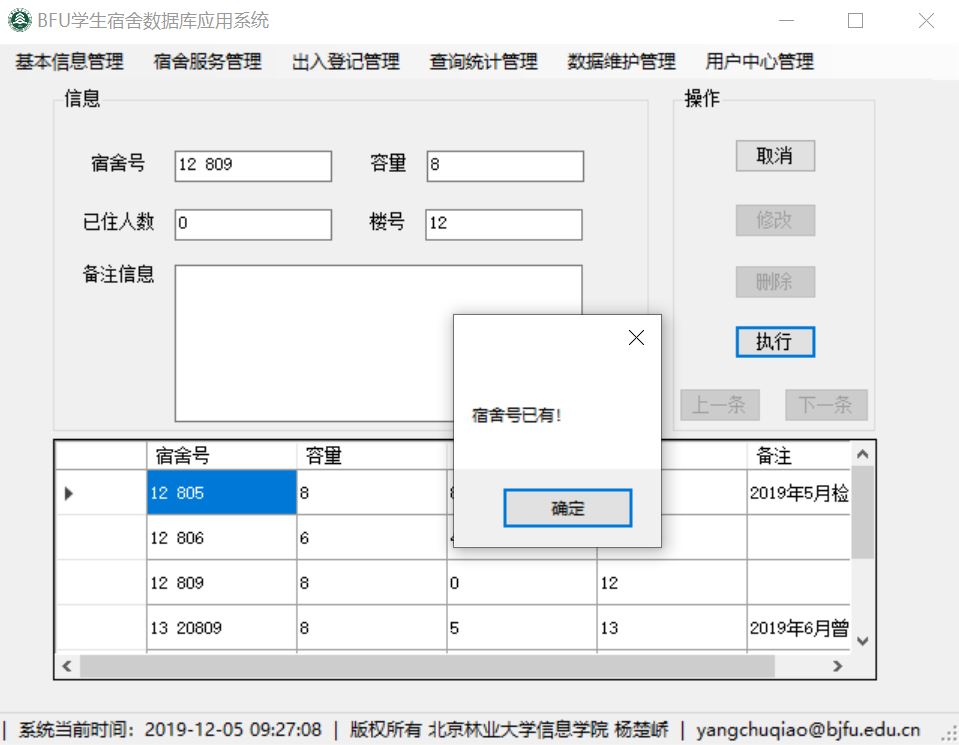


图29 学生宿舍管理功能界面

② 点击“修改+执行”按钮可以更改宿舍信息，但是修改后的信息要符合格式要求，否则无法修改。



图30 学生宿舍管理功能界面

③ 点击“删除+执行”按钮删除当前选择的宿舍信息后，执行delete语句删除该宿舍信息，如果该学生已经在费用等表中中有了信息，就无法删除（参照完整性）。



图31 学生宿舍管理功能界面

（4）SQL语句与宿主语言嵌套使用代码：

与6.3.2（4）大致相同，此处省略

6.3.4 职工信息管理功能

（1）功能：增加、删除、修改、显示职工信息。

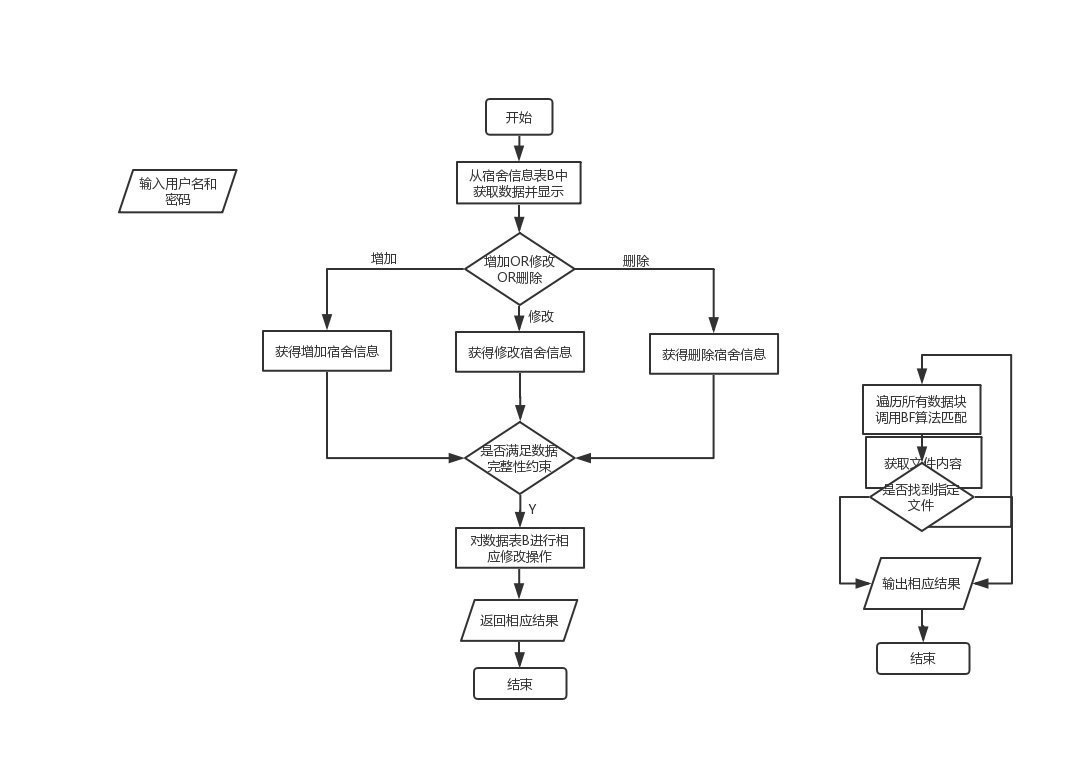


图32 职工信息管理功能流程图

（2）功能界面：



图33 职工信息管理功能界面

（3）功能界面简单文字描述：

① 点击“增加+执行”时会先判断数据库中是否有新添加的职工信息，如果有就无法添加，如果没有就执行添加职工信息。



图表 34 职工信息管理功能界面

② 点击“修改+执行”按钮可以更改职工信息，但是修改后的信息要符合格式要求，否则无法修改。



图35 职工信息管理功能界面

③ 点击“删除+执行”按钮删除当前选择的宿舍信息后，执行delete语句删除该职工信息，如果该学生已经在报修等表中中有了信息，就无法删除（参照完整性）。



图36 职工信息管理功能界面

（4）SQL语句与宿主语言嵌套使用代码：

与6.3.2（4）大致相同，此处省略

6.3.5 宿舍报修服务功能

（1）功能：实现宿舍报修功能。

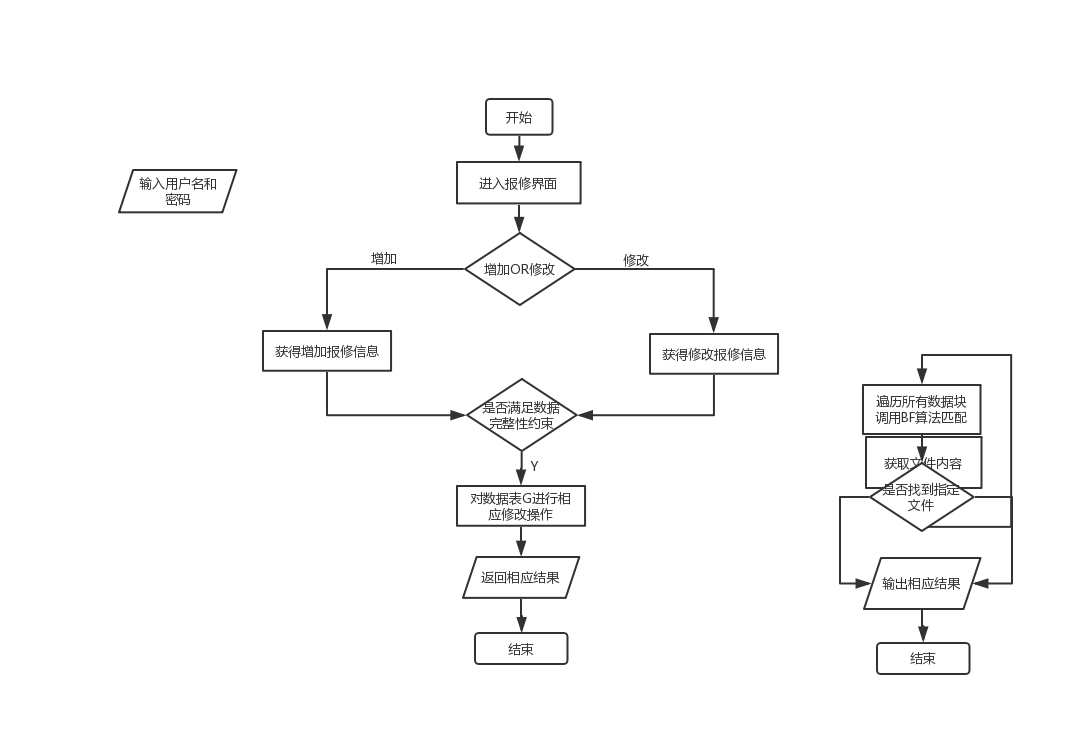


图37 宿舍报修服务功能流程图

（2）功能界面：



图38 宿舍报修服务功能界面

（3）功能界面简单文字描述：

① 点击“增加”按钮，可自动生成报修单号，学生或者管理员按照要求输入报修信息，如果未被解决，解决时间填写空。



图39 宿舍报修服务功能界面

② 点击“修改+执行”按钮可以更改报修信息，但是修改后的信息要符合格式要求，否则无法修改。报修解决后可以由学生或者管理员添加解决日期来保证报修的顺利解决。

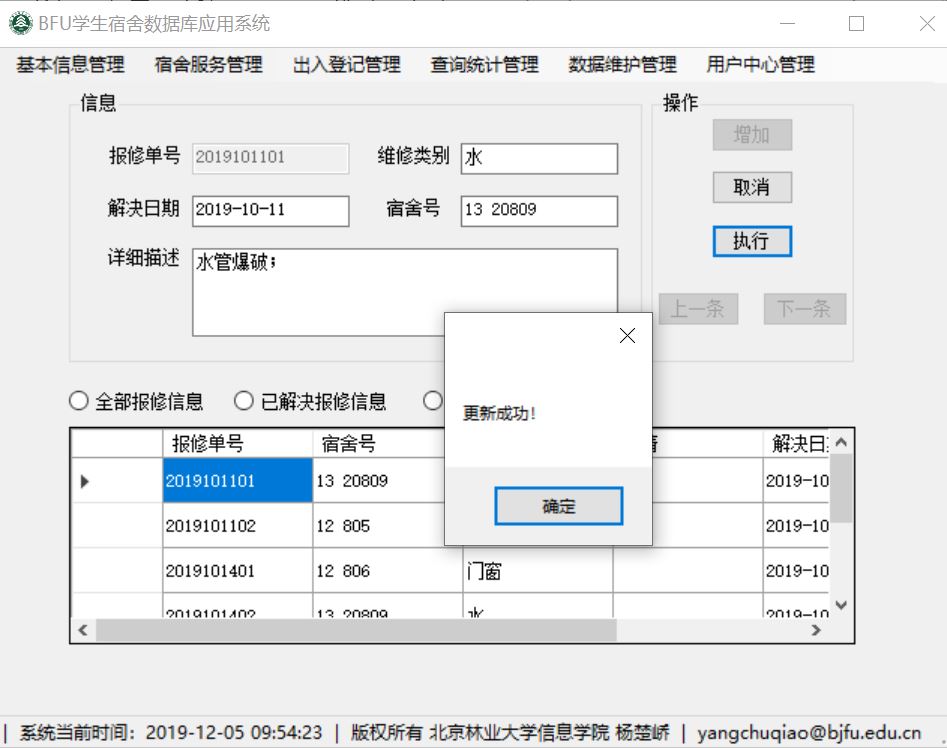


图40 宿舍报修服务功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button4\_Click(object sender, EventArgs e) {

//增加记录

if (this.button1.Enabled) {

string bao=this.textBox1.Text.Trim();

string dono = this.textBox3.Text.Trim();

string lb = this.textBox2.Text.Trim();

string sto ;

string date = this.textBox4.Text.Trim();

string xq = this.txtPRemark.Text.Trim();

if (textBox1.Equals("")){

MessageBox.Show("宿舍号不为空！");

return; }

if (textBox2.Equals("")){

MessageBox.Show("维修类别不为空！");

return; }

//进行插入记录操作

string sql2 = "insert into G(Bao,dono,lb,xq) values('" + bao + "','" + dono + "','" + lb + "','" + xq + "')";

DBConnection db1 = new DBConnection();

int j = db1.executeSqlResult(sql2);

if (j > 0) {

MessageBox.Show("添加成功！！");

//刷新datatable

string sql21 = "select \* from G";

dt = db1.runSql(sql21);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);}}

else

{

MessageBox.Show("添加失败！！");

return;

}

}

else if (this.button2.Enabled)

{

//修改

string bao = this.textBox1.Text.Trim();

string dono = this.textBox3.Text.Trim();

string lb = this.textBox2.Text.Trim();

string sto;

string date = this.textBox4.Text.Trim();

string xq = this.txtPRemark.Text.Trim();

if (MessageBox.Show("您真的要更新吗？", "请确认", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

string sql3 = "update G set lb='" + lb + "',date='" + date + "', xq='" + xq + "'where dono='" + dono + "'";

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult(sql3);

if (m > 0)

{

MessageBox.Show("更新成功！");

//重新绑定新的datatable

string sql31 = "select \* from G";

dt = db1.runSql(sql31);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else{

MessageBox.Show("更新失败！");

return;}}}

6.3.6 宿舍费用服务功能

（1）功能：实现宿舍费用信息的查询。

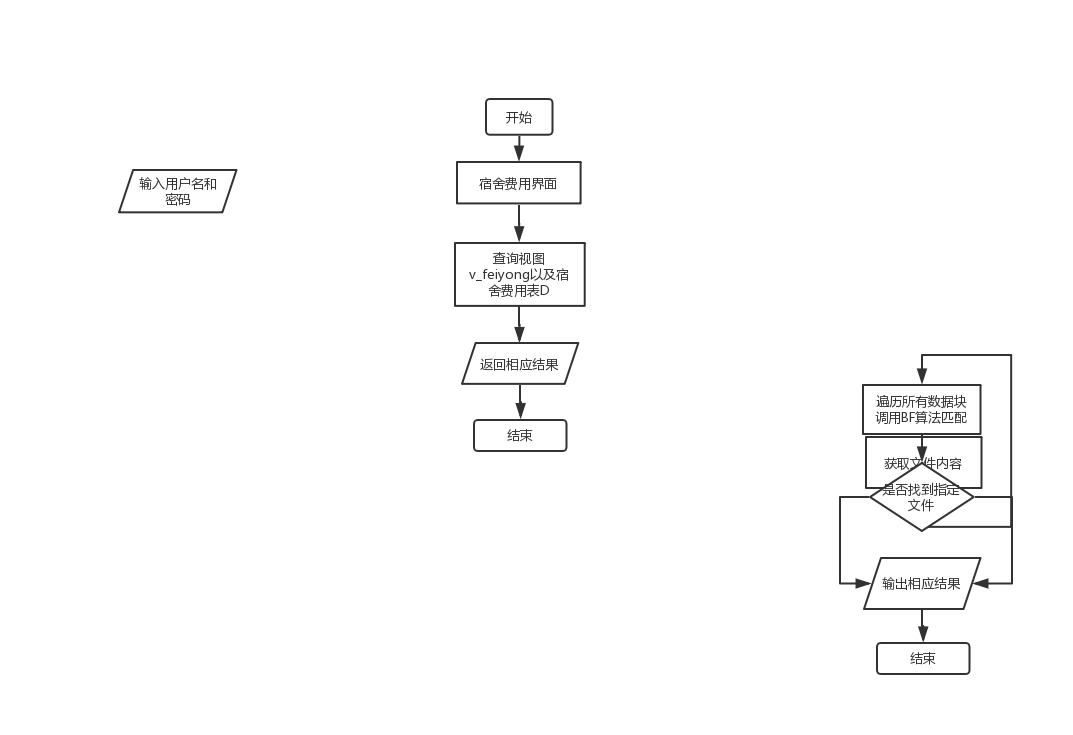


图41 宿舍费用服务功能流程图

（2）功能界面：



图42 宿舍费用服务功能界面

（3）功能界面简单文字描述：点击“上一条”、“下一条”可以实现数据的显示。

（4）SQL语句与宿主语言嵌套使用代码：

private void member\_show(DataTable dt) {

this.textBox1.Text = dt.Rows[NoPosition]["Dono"].ToString();

this.textBox2.Text = dt.Rows[NoPosition]["Month"].ToString();

this.textBox3.Text = dt.Rows[NoPosition]["Mo"].ToString();

this.textBox4.Text = dt.Rows[NoPosition]["Ele"].ToString();

this.textBox5.Text = dt.Rows[NoPosition]["WP"].ToString();

this.textBox6.Text = dt.Rows[NoPosition]["Water"].ToString();

this.textBox8.Text = dt.Rows[NoPosition]["EP"].ToString();}

private void button2\_Click(object sender, EventArgs e) {

if (NoPosition > 0)

{

NoPosition--;

member\_show(dt);

}

else

{

MessageBox.Show("已经是首条！");}

private void button3\_Click(object sender, EventArgs e)

{

if (NoPosition != datanum - 1)

{

NoPosition++;

member\_show(dt);)}

Else{

MessageBox.Show("已经是末条！");}}

6.3.7 公告通知服务功能

（1）功能：增加、删除、修改、显示公告信息。

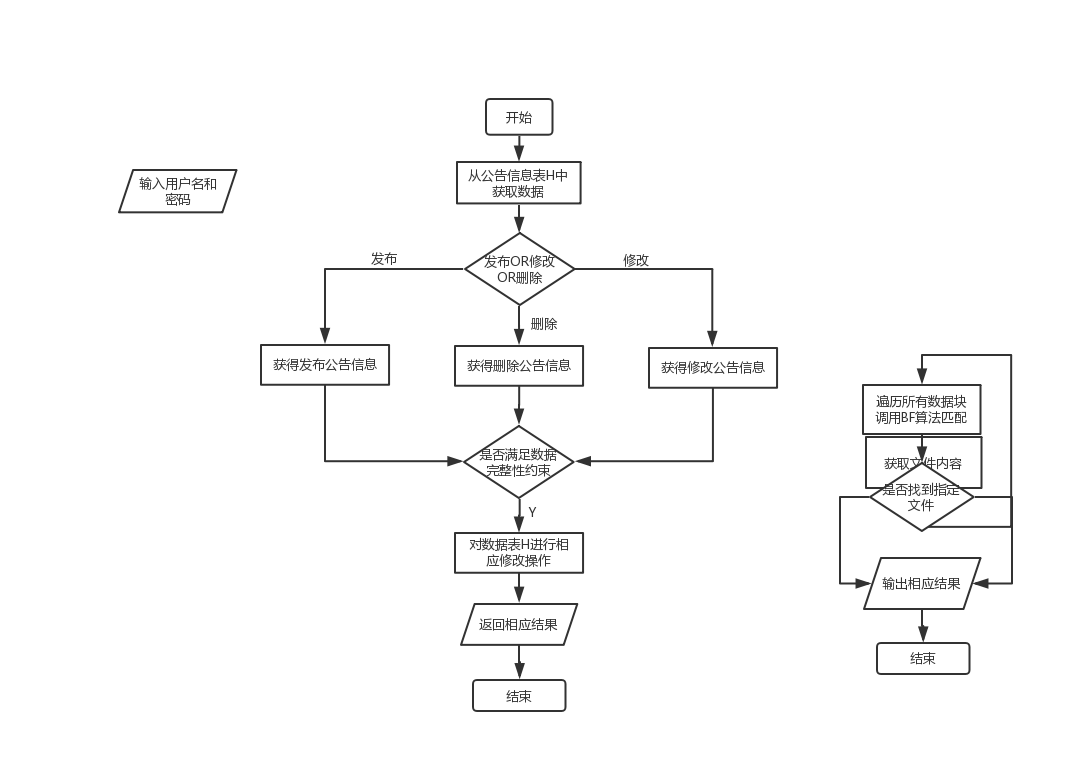


图43 公告通知服务功能流程图

（2）功能界面：



图44 公告通知服务功能界面

（3）功能界面简单文字描述：

① 点击“增加+执行”时会自动生成新的公告单号，宿舍管理员可以发布最新消息。

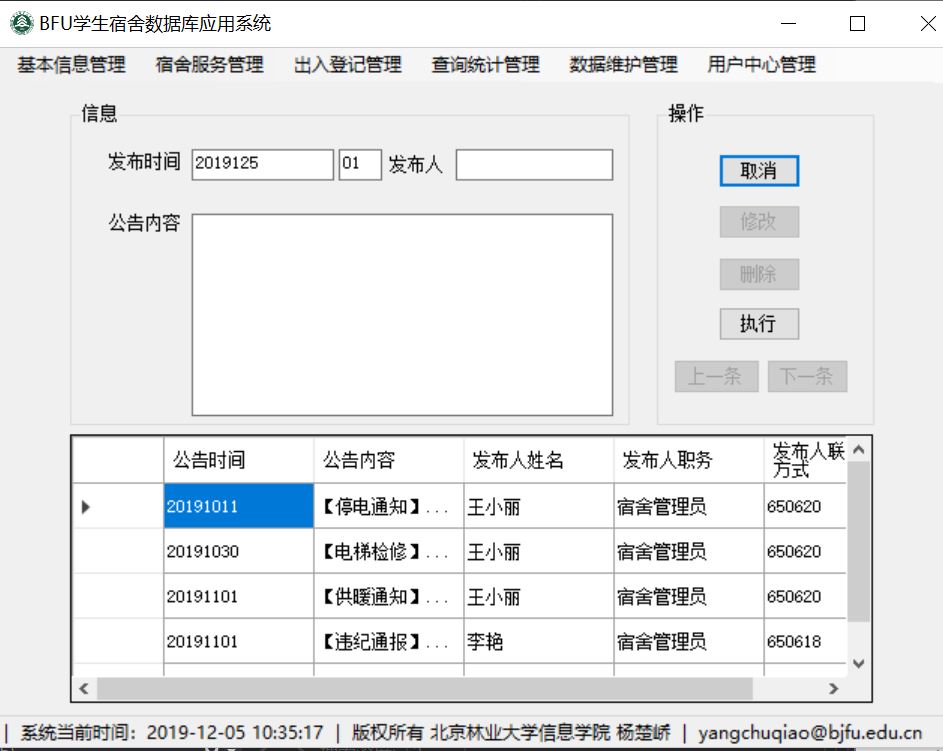


图45 公告通知服务功能界面

② 点击“修改+执行”按钮可以更改公告信息，但是修改后的信息要符合格式要求，否则无法修改。



图46 公告通知服务功能界面

③ 点击“删除+执行”按钮删除当前选择的公告信息。



图47 公告通知服务功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button4\_Click(object sender, EventArgs e)

{

//增加记录

if (this.button1.Enabled)

{

String gno = this.textBox1.Text.Trim()+ this.textBox2.Text.Trim();

string sto = this.textBox3.Text.Trim();

string sy = this.txtPRemark.Text.Trim();

if (textBox3.Equals(""))

{

MessageBox.Show("公告人职工号不为空！");

return;

}

//进行插入记录操作

string sql2 = "insert into H(gno,sto,sy) values('" + gno + "','" + sto + "','" + sy + "')";

DBConnection db1 = new DBConnection();

int j = db1.executeSqlResult(sql2);

if (j > 0)

{

MessageBox.Show("添加成功！！");

//刷新datatable

string sql21 = "select \* from H";

dt = db1.runSql(sql21);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else

{

MessageBox.Show("添加失败！！");

return;

}

}

else if (this.button2.Enabled)

{

//修改

string gno = this.textBox1.Text.Trim()+this.textBox2.Text.Trim();

string sto = this.textBox3.Text.Trim();

string sy = this.txtPRemark.Text.Trim();

if (textBox3.Equals(""))

{

MessageBox.Show("职工号不为空！");

return;

}

if (MessageBox.Show("您真的要更新吗？", "请确认", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

string sql3 = "update H set sto='" + sto + "',sy='" + sy + "' where gno= '" + gno + "'";

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult(sql3);

if (m > 0)

{

MessageBox.Show("更新成功！");

//重新绑定新的datatable

string sql31 = "select \* from H";

dt = db1.runSql(sql31);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else

{

MessageBox.Show("更新失败！");

return;

}

}

}

else if (this.button3.Enabled)

{

//删除

if (MessageBox.Show("您真的要删除吗？", "此删除不可恢复", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

string gno = this.textBox1.Text.Trim()+this.textBox2.Text.Trim();

string sql4 = "delete from H where gno='" + gno + "'";

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult(sql4);

if (m > 0)

{

MessageBox.Show("删除成功！");

string sql41 = "select \* from H";

dt = db1.runSql(sql41);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else

{

MessageBox.Show("删除失败！");

return;

}}}}

6.3.8 来访登记管理功能

（1）功能：增加、删除、显示外来人员来访信息。

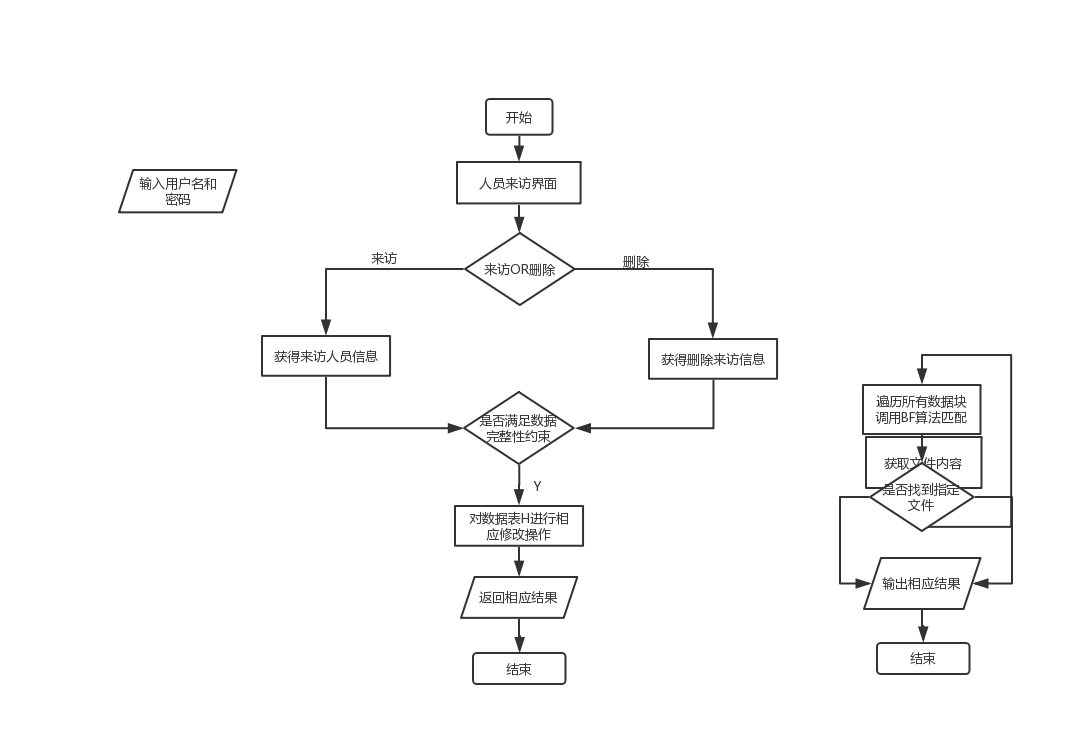


图48 来访登记管理功能流程图

（2）功能界面：



图49 来访登记管理功能界面

（3）功能界面简单文字描述：

① 点击“增加+执行”时会自动生成表单号，系统会利用存储过程pro\_laifang自动判断外来人员填写的信息是否符合要求。



图50 来访登记管理功能界面

② 点击“删除+执行”按钮删除当前选择的宿舍信息。

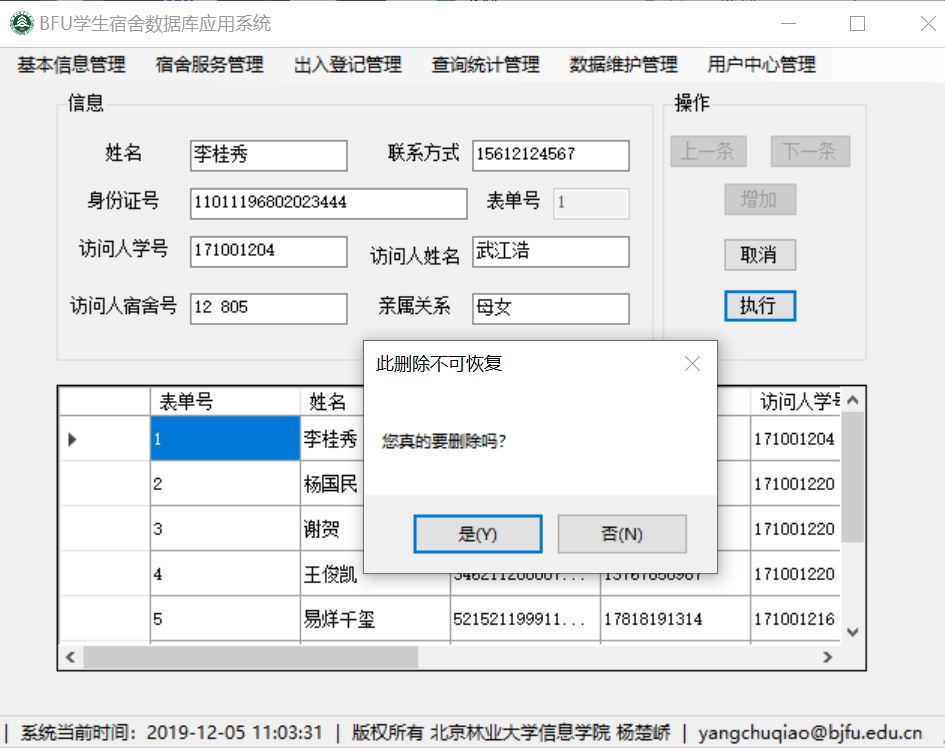


图51 来访登记管理功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button6\_Click(object sender, EventArgs e)

{

//增加记录

if (this.button1.Enabled)

{

string name = this.textBox1.Text.Trim();

string tel = this.textBox2.Text.Trim();

string sno = this.textBox3.Text.Trim();

string fno = this.textBox4.Text.Trim();

string sn = this.textBox5.Text.Trim();

string dono = this.textBox6.Text.Trim();

string re = this.textBox7.Text.Trim();

string no = this.textBox8.Text.Trim();

if (textBox2.Equals(""))

{

MessageBox.Show("联系方式不为空！");

return;

}

if (textBox1.Equals(""))

{

MessageBox.Show("姓名不为空！");

return;

}

if (textBox4.Equals(""))

{

MessageBox.Show("身份证号不为空！");

return;

}

//int age = 0;

//age = int.Parse(agestring);//age转为整型

//首先根据会员号查询是否有这样的记录

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_laifang '" + name + "','" +tel+ "','" + fno+ "','" + sno+ "','" + sn+ "','" + dono+ "','" + re + "'";

int i = 0;

i = db1.executeSqlResult(sql1);

if (i > 0)

{

MessageBox.Show("添加成功！！");

//刷新datatable

string sql21 = "select \* from v\_laifang";

dt = db1.runSql(sql21);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else

{

MessageBox.Show("添加失败！！");

return;

}

}

else if (this.button5.Enabled)

{

//删除

if (MessageBox.Show("您真的要删除吗？", "此删除不可恢复", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

string no = this.textBox8.Text.Trim();

string sql4 = "delete from E where sno='" + no + "'";

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult(sql4);

if (m > 0)

{

MessageBox.Show("删除成功！");

string sql41 = "select \* from v\_laifang";

dt = db1.runSql(sql41);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else

{

MessageBox.Show("删除失败！");

return;

}

}

}

}

6.3.9 晚归登记管理功能

（1）功能：增加、显示学生晚归信息。

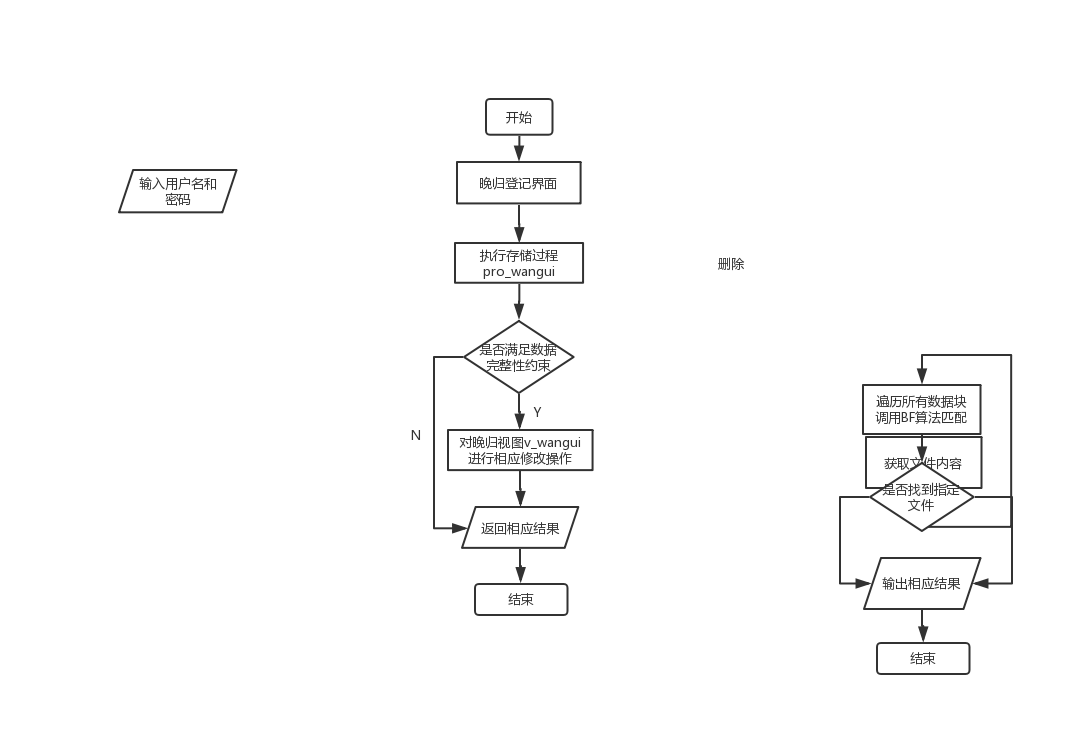


图52 晚归登记管理功能流程图

（2）功能界面：



图53 晚归登记管理功能界面

（3）功能界面简单文字描述：

点击“增加+执行”时会执行存储过程pro\_wangui自动判断学生填写信息与数据库中学生信息是否匹配，如果不匹配添加失败，可防止学生填写错误信息。

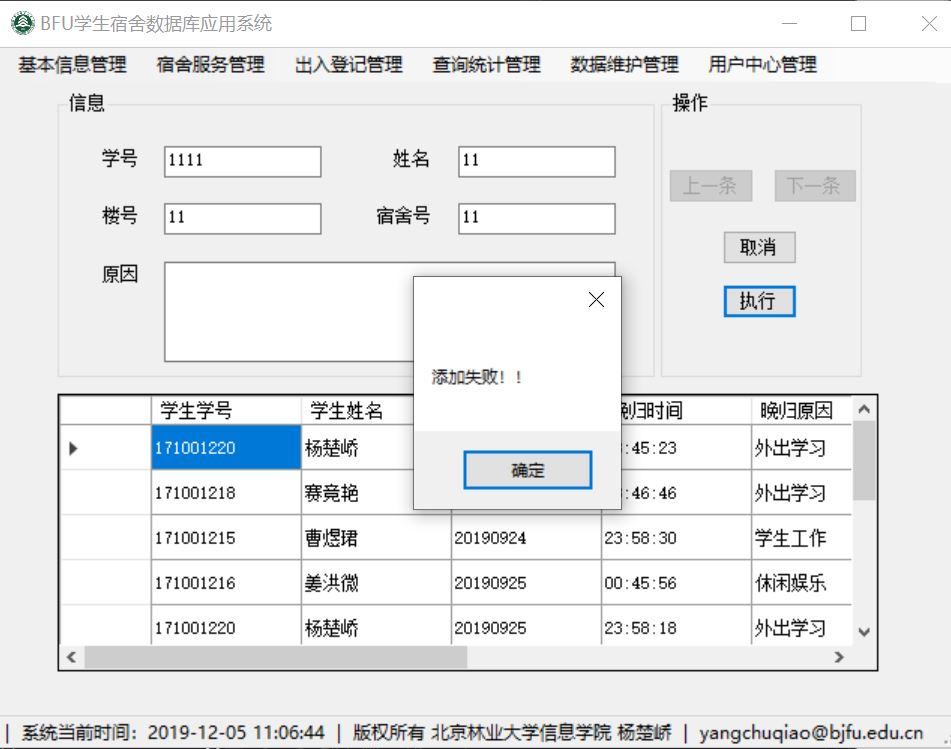


图54 晚归登记管理功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button6\_Click(object sender, EventArgs e)

{

//增加记录

if (this.button1.Enabled)

{

string sno = this.textBox1.Text.Trim();

string sn = this.textBox2.Text.Trim();

string lno = this.textBox3.Text.Trim();

string dono = this.textBox5.Text.Trim();

string bec = this.txtPRemark.Text.Trim();

string gno = "01";

DBConnection db1 = new DBConnection();

string sql1 = "select \* from F where left(no,8)= Convert(varchar,GETDATE(),112)";

int i = 0;

i = db1.runSqlInt(sql1);

if (i >=10)

{

// MessageBox.Show("宿舍号已有！");

//return;

gno = i.ToString();

}

else if (i > 0 && i<10)

{

gno = "0" + i.ToString();

}

string no = gno;

string sql12 = "execute pro\_wangui '" + sno + "','" + sn + "','" + lno + "','" + dono + "','" + bec + "','" + no + "'";

//string sql12 = "execute pro\_wangui '171001220','杨楚峤','13','13#20809','外出学习','01'";

int j= 0;

j = db1.executeSqlResult(sql12);

if (j > 0)

{

MessageBox.Show("添加成功！！");

//刷新datatable

string sql21 = "select \* from v\_wangui";

dt = db1.runSql(sql21);

if (dt.Rows.Count > 0)

{

NoPosition = 0;

datanum = dt.Rows.Count;

member\_show(dt);

}

}

else

{

MessageBox.Show("添加失败！！");

return;

}

}

}

6.3.10 综合查询功能

（1）功能：根据用户输入的查询要求查询学生信息和职工信息。

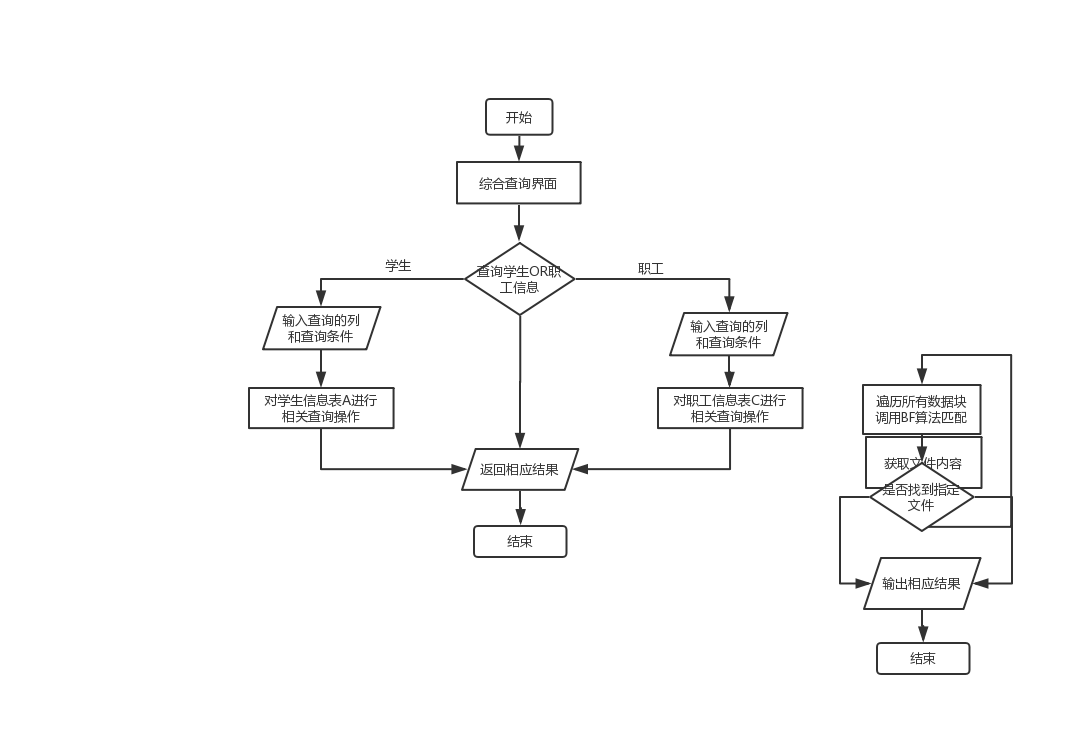


图55 综合查询功能流程图

（2）功能界面：



图56 综合查询功能界面



图57 综合查询功能界面

（3）功能界面简单文字描述：

用户可以在“查询的列”输入想要检索的列名，并给出相应的“查询条件”，系统同可自动检索相应表里面的数据（支持中英文）。



图58 综合查询功能界面



图59 综合查询功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button2\_Click(object sender, EventArgs e)

{

string t1 = this.textBox1.Text.Trim();

string t2 = this.textBox2.Text.Trim();

if (t1 == "学号")

t1 = "sno";

if (t1 == "姓名")

t1 = "sn";

if (t1 == "性别")

t1 = "sex";

if (t1 == "籍贯")

t1 = "bp";

if (t1 == "学院")

t1 = "dn";

if (t1 == "宿舍号")

t1 = "dono";

if (t1 == "班级")

t1 = "gn";

if (t1 == "联系方式")

t1 = "tel";

DataTable dt = new DataTable();//存储查询出来的记录

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_student '" + t1 + "','" + t2 + "'";

dt = db1.runSql(sql1);

//显示

dataGridView1.DataSource = dt;

}

private void button1\_Click\_1(object sender, EventArgs e)

{

string t3 = this.textBox3.Text.Trim();

string t4 = this.textBox4.Text.Trim();

if (t4 == "职工号")

t4 = "sto";

if (t4 == "姓名")

t4 = "st";

if (t4 == "性别")

t4 = "sex";

if (t4 == "籍贯")

t4 = "bp";

if (t4 == "职务")

t4 = "wk";

if (t4 == "年龄")

t4 = "age";

if (t4 == "楼号")

t4 = "lno";

if (t4 == "联系方式")

t4 = "tel";

DataTable dt1 = new DataTable();//存储查询出来的记录

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_zhigong '" + t4 + "','" + t3 + "'";

dt1 = db1.runSql(sql1);

//显示

dataGridView2.DataSource = dt1;

}

6.3.11 综合统计功能

（1）功能：显示统计信息。

（2）功能界面：



图60 综合统计功能界面



图61 综合统计功能界面

（3）功能界面简单文字描述：系统调用视图v\_wangui以及v\_feiyong展示统计信息。

（4）SQL语句与宿主语言嵌套使用代码：

private void Form11\_Load(object sender, EventArgs e)

{

DataTable dt = new DataTable();//存储查询出来的记录

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_wanguichaxun ";

dt = db1.runSql(sql1);

//显示

dataGridView1.DataSource = dt;

string sql2 = "execute pro\_feiyong ";

dt = db1.runSql(sql2);

//显示

dataGridView2.DataSource = dt;

}

6.3.12 数据备份功能

（1）功能：实现数据库备份功能。

（2）功能界面：

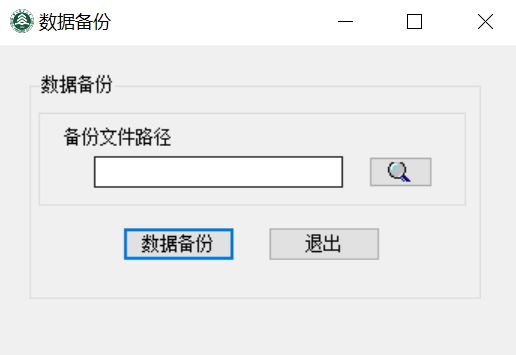


图62 数据备份功能界面

（3）功能界面简单文字描述： 点击“数据备份”按钮，选择好文件夹，即可生成备份文件 ，防止数据丢失。

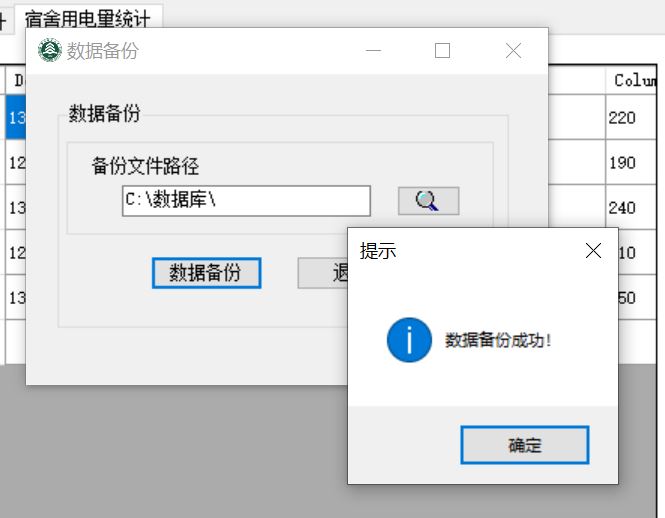


图63 数据备份功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void btnDStore\_Click(object sender, EventArgs e)

{

try

{

if (File.Exists(txtDSPath.Text.Trim() + ".bak"))

{

MessageBox.Show("该文件已经存在！", "提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

txtDSPath.Text = "";

txtDSPath.Focus();

}

else

{

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult("backup database YCQ to disk='" + txtDSPath.Text.Trim() + "back.bak'");

// int m = db1.executeSqlResult("backup database YCQ to disk=' C://数据库'");

MessageBox.Show("数据备份成功！", "提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

6.3.13 数据还原功能

（1）功能：实现数据还原功能。

（2）功能界面：

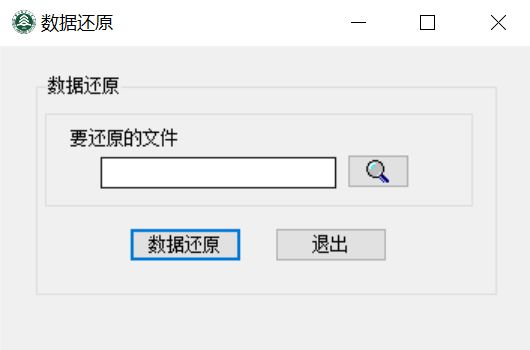


图64 数据还原功能界面

（3）功能界面简单文字描述：点击“数据还原”按钮，选择好文件夹，即可将备份文件还原，防止数据丢失。

（4）SQL语句与宿主语言嵌套使用代码：

private void btnDRevert\_Click(object sender, EventArgs e)

{

try

{

DBConnection db1 = new DBConnection();

int m = db1.executeSqlResult("use master restore database YCQ from disk='" + txtDRPath.Text.Trim() + "'");

MessageBox.Show("数据还原成功！", "提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

6.3.14 用户管理功能

（1）功能：实现系统的用户管理功能。

（2）功能界面：



图65 用户管理功能界面

（3）功能界面简单文字描述：

① 点击“增加用户”按钮即可弹出增加用户窗口，增加用户信息。



图66 用户管理功能界面

② 点击“修改权限”按钮即可弹出修改权限窗口，修改用户权限。



图67 用户管理功能界面

（4）SQL语句与宿主语言嵌套使用代码：

private void button2\_Click(object sender, EventArgs e)

{

string m1 = this.textBox1.Text.Trim();

string m2 = this.textBox2.Text.Trim();

string m3 = this.txtUPwd.Text.Trim();

string m4 = this.cboxURight.Text.Trim();

DBConnection db1 = new DBConnection();

string sql1 = "select \* from U where name='" + m1 + "'";

int i = 0;

i = db1.runSqlInt(sql1);

if (i > 0)

{

MessageBox.Show("用户名已有！");

return;

}

if (m2!=m3)

{

MessageBox.Show("两次密码输入不一致！");

return;

}

string sql2 = "insert into U(name,pwd,power) values('" + m1 + "','" + m2 + "','" + m4 + "')";

int j = db1.executeSqlResult(sql2);

if (j > 0)

{

MessageBox.Show("添加成功！！");

}

else

{

MessageBox.Show("添加失败！！");

return;

}

}

private void btnDRevert\_Click(object sender, EventArgs e)

{

string m1 = this.textBox1.Text;

string m2 = this.cboxURight.Text;

DBConnection db1 = new DBConnection();

string sql1 = "update U set power='" + m2 + "' where name='" + m1 + "'";

int j = 0;

j = db1.executeSqlResult(sql1);

if (j > 0)

{

MessageBox.Show("修改成功！！");

//刷新datatable

}

else

{

MessageBox.Show("修改失败！！");

return;

}

}

6.3.15 修改密码功能

（1）功能：修改用户密码。

（2）功能界面：



图68 修改密码功能界面

（3）功能界面简单文字描述：调用存储过程pro\_pwd，修改用户密码。

（4）SQL语句与宿主语言嵌套使用代码：

private void btnDRevert\_Click(object sender, EventArgs e)

{

string m1 = this.textBox1.Text.Trim();

string m2 = this.txtUOPwd.Text.Trim();

string m3 = this.txtUNPwd.Text.Trim();

string m4 = this.txtFUNPwd.Text.Trim();

DBConnection db1 = new DBConnection();

string sql1 = "execute pro\_pwd '" + m1 + "','" + m2 + "','" + m3+ "','" + m4 + "'";

int j = 0;

j = db1.executeSqlResult(sql1);

if (j > 0)

{

MessageBox.Show("修改成功！！");

//刷新datatable

}

else

{

MessageBox.Show("修改失败！！");

return;

}

}

7 系统运行实例

系统与运行实例请见6.3 功能模块设计，本部分将给出系统功能的全部主界面：



图69 登录界面



图70 登陆成功



图71 主界面



图72 学生信息管理界面



图73 宿舍信息管理界面



图74 职工信息管理界面



图75 宿舍报修服务界面



图76 宿舍费用查询界面

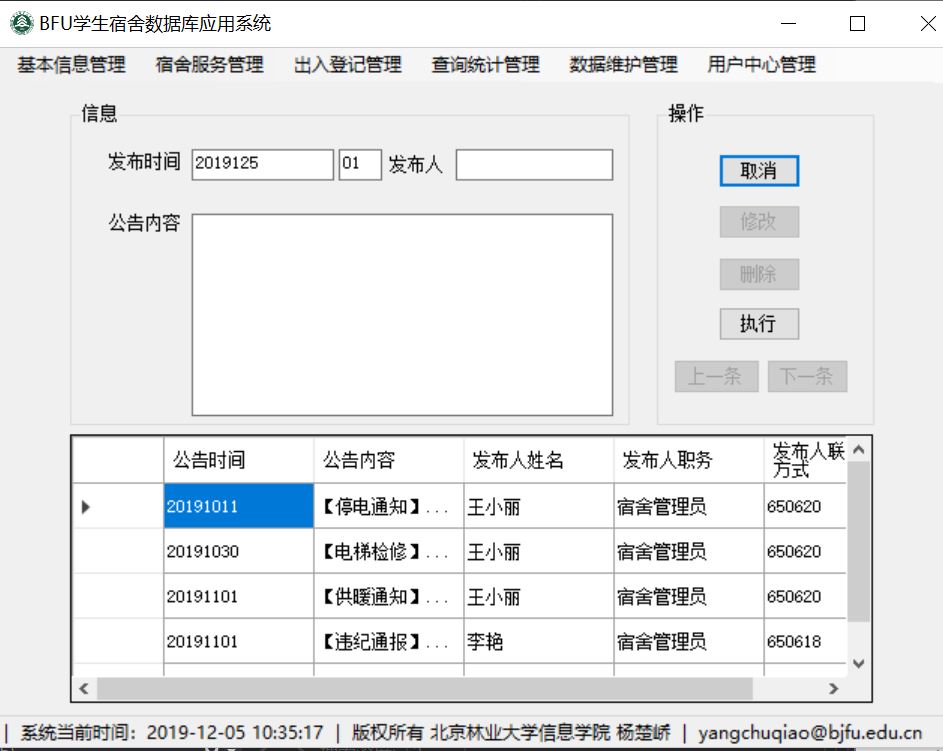


图77 公告信息处理界面



图78 外来人员登记界面



图79 学生信息查询界面



图表 80 职工信息查询界面



图表 81 晚归信息统计界面



图82 宿舍电量统计界面

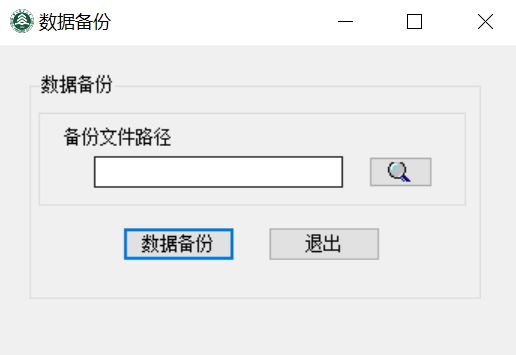


图83 数据备份

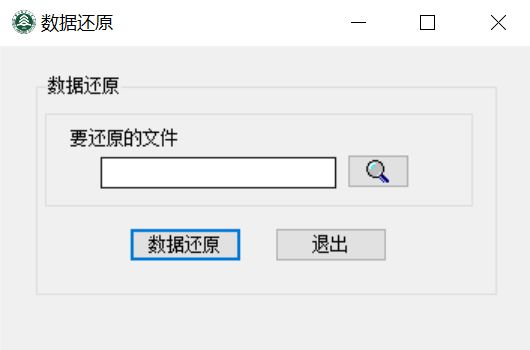


图84 数据恢复



图85 用户管理界面



图86 增加用户界面



图87 修改权限界面



图88 修改密码界面

8 结束语

**8.1 遇到的问题以及解决途径**

（1）最初未能找准ER图与关系的联系。在最初设计数据库的时候有一些迷茫，设计ER图有些逻辑不清晰，某些属性字段定不准，后来经过与老师积极的探讨以及与师兄师姐的沟通，顺利的完成了ER图的设计以及关系的转换。

（2）数据库设计与程序的间的连接。最初，因为对C#与数据库的链接没有掌握透彻，在连接以及用户名设置上走了很多弯路。最后经过在网上搜集资料学习，逐渐理解了C#与数据库连接的原理，之后的课程设计也就顺利了很多。

**8.2 待改进的方面**

（1）业务流程有待优化。经过与宿舍公寓管理员的沟通，现实生活中学生宿舍业务是更加复杂的，本次课程设计只选取了较常见的部分业务流程，并且简化了系统使用用户的设置。接下来，仍然需要优化已经实现的业务流程，并在此基础上增加其他业务，使得数据流向更加清晰。

（2）存储过程的调用有待优化。现阶段，存储过程的编写进满足了系统需要，仍需要在考虑学生宿舍管理系统用户群体庞大，对并发处理能力要求较高的基础上，优化程序与数据库的调用语句，使得执行速度能够进一步提高。

**8.3 课程设计心得与收获**

顺利完成本次数据库课程设计后，找到了大学前两年很多困惑问题的答案，也是全方位的提升了个人能力。我将心得与收获总结为以下几点：

（1）数据逻辑能力增强

因为大创项目的原因，在大二就已经接触了数据库。当时觉得数据库就是一些存储数据的文件，不理解范式，不理解表与表之间的依赖关系，仅仅简单以为把所有数据存储到其中即可。在本学期学习结束后，才算是真正意识到了数据库逻辑的紧密性，也从心里感受到这是一门很有深度的学科，作为信息学子，自己也会继续学习数据知识，提高数据敏感性。

（2）数据库深入理解

本次课设是从全方面进行数据库设计，概念模型、逻辑模型、物理模型的紧密结合保证了数据库设计的正确无误。课设结束后，我更加清楚了数据库的原理，这也为在今后的学习生活打下了坚实的基础。

（3）编程能力提升

之前比较熟悉python，最初也是想用python编程。但是后来想到正好可以趁此机会好好学习一下C#，于是最后还是选定了C#作为程序开发语言。经过这次课设，也算是深入学习了C#，独立开发出功能相对完整的C/S系统，设计完成后也是有一定小小的成就感。

附件：数据字典

**表1 学生信息表A**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Sno | 学号 | varchar(10) | 主键：PRIMARY KEY |
| Sn | 姓名 | Varchar(10) | NOT NULL |
| Sex | 性别 | Char(10) | ‘男‘ or ‘女‘ |
| BP | 籍贯 | Char(20) |  |
| Dn | 学院 | Char(20) | NOT NULL |
| Gn | 班级 | Char(20) | NOT NULL |
| Tel | 联系方式 | Varchar(20) |  |
| Dono | 宿舍号 | Char(10) | 外建：参照B表的Dono字段 |

**表2 宿舍信息表B**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Dono | 宿舍号 | Char(10) | 主键：PRIMARY KEY |
| AP | 容量 | int | NOT NULL |
| Occupy | 已住人数 | int |  |
| Beizhu | 宿舍备注 | varchar(MAX) |  |
| Lno | 楼号 | Varchar(6) | 外建：参照I表的Lno字段 |

**表3 职工信息表C**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Sto | 职工号 | VarChar(10) | 主建：PRIMARY KEY |
| St | 姓名 | VarChar(10) | NOT NULL |
| Sex | 性别 | Char(10) | ‘男‘ or ‘女‘ |
| Age | 年龄 | Char(10) |  |
| WK | 职务 | Char(10) |  |
| Tel | 联系方式 | Varchar(20) |  |
| BP | 籍贯 | Char(10) |  |
| Lno | 楼号 | Varchar(6) | 外建：参照I表的Lno字段 |

**表4 费用表D**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Mo | 费用单号 | VarChar(10) | 主建：PRIMARY KEY |
| Dono | 宿舍号 | Char(10) | 外建：参照B表的Dono字段 |
| Month | 月份 | Int | 取值：1-12 |
| WP | 水单价 | Float |  |
| Water | 用水量 | Float |  |
| EP | 电单价 | Float |  |
| Ele | 用电量 | Float |  |

**表5 外来人员访问表E**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| No | 表单号 | Int | 主建：PRIMARY KEY |
| Fno | 身份证号 | Char(20) |  |
| Name | 姓名 | VarChar(10) | NOT NULL |
| Tel | 联系方式 | Varchar(20) | NOT NULL |
| RE | 亲属关系 | Char(6) | NOT NULL |
| Sno | 访问人学号 | VarChar(10) | 外建：参照A表的Sno字段 |
| Time | 访问时间 | datetime | NOT NULL |

**表6 晚归登记表F**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| No | 登记单号 | VarChar(20) | 主建：PRIMARY KEY |
| Sno | 学号 | varchar(10) | 外建：参照A表的Sno字段 |
| Bec | 晚归原因 | varchar(MAX) | NOT NULL |
| Time | 晚归时间 | Time(7) | NOT NULL |

**表7 报修表G**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Bao | 报修单号 | VarChar(20) | 主建：PRIMARY KEY |
| Dono | 宿舍号 | Char(10) | 外建：参照B表的Dono字段 |
| LB | 维修类别 | Char(6) | NOT NULL |
| XQ | 详情 | VarChar(50) |  |
| Date | 解决日期 | varchar(50) |  |
| Sto | 职工号 | VarChar(10) | 外建：参照C表的Sto字段 |

**表8 公告单H**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Gno | 公告单号 | VarChar(20) | 主建：PRIMARY KEY |
| Sto | 职工号 | VarChar(10) | 外建：参照C表的Sto字段 |
| SY | 具体事宜 | varchar(MAX) | NOT NULL |

**表9 宿舍楼信息表I**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 约束 |
| Lno | 楼号 | VarChar(6) | 主建：PRIMARY KEY |
| Ln | 楼长 | VarChar(10) |  |
| Tel | 联系方式 | varchar(20) |  |

**表10 关系表**

|  |  |  |
| --- | --- | --- |
| 关系名 | 字段名 | 说明 |
| A | Sno，Sn，Sex，BP，Dn，Gn，Tel，Dono | 主键：sno;  外键：dono参照关系B(dono) |
| B | Dono，AP，Occupy，Lno | 主键：Dono;  外键：Lno参照关系I(Lno) |
| C | Sto,St,Sex,Age,WK,Tel, BP ,Lno | 主键：Sto，  外键：Lno参照关系I(Lno) |
| D | Mo,Dono,Month,WP,Water,EP,Ele | 主键：Mo，  外键：Dono参照关系B(Dono) |
| E | No,Fno,Name,Tel,RE,Time,Sno | 主键：No，  外键：Sno参照关系A(Sno) |
| F | No,Sno,Bec,Time | 主键：No，  外键：Sno参照关系A(Sno) |
| G | Bao,Dono,LB,XQ,Date,Sto | 主键：Bao，  外键：Sto参照关系C(Sto) |
| H | Gno,Sto,SY | 主键：Gno，  外键：Sto参照关系C(Sto) |
| I | Lno,Ln,Tel | 主键：Lno， |