

实

训

报

告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **人工智能** | | | **学院** | **2023** | **级** |
| **计算机科学与技术** | | | **专业** | **1** | **班** |
| **姓名：** | **马丽茗** | | | | |
| **学号：** | **202335010132** | | | | |
| **指导教师：** | | **郝小芳** | | | |
| **日期：** | **2025.06.24** | | | | |

目 录

1 系统定义..............................................1

2 需求分析..............................................1

2.1 功能需求分析.....................................1

2.2 可行性分析.......................................1

3 概念结构设计..........................................1

3.1 主要数据库设计...................................1

3.2 管理员模块设计...................................1

4 逻辑结构设计..........................................1

5 系统实现..............................................1

5.1 业主模块实现.....................................1

5.2 管理员模块实现...................................1

6 系统测试..............................................1

**物业管理系统的设计与实现**

# 系统定义

本文设计并开发了一个物业管理系统, 参考了《物业管理系统报告：深度解读智慧物业解决方案》。在该系统中，业主可查看并确认相关信息，实现管理处与业主之间的信息透明化；管理员增加信息，丰富系统数据资源，为后续管理与决策提供全面依据；删除信息，清理无效或过时数据，确保系统信息的准确性和时效性；查询信息，快速定位所需数据，提升管理效率与响应速度；修改信息，及时更新数据内容，保证系统信息的准确性和完整性。

该系统采用python进行设计开发页面，以MySQL作为数据库。根据模拟现实需求设计和开发系统，测试结果表明，系统页面结构清晰，响应迅速，每项功能运行正常，促进小区管理员和业主之间的信息整合与交互。

关键词：物业管理系统；信息化；python；MySQL

当前，随着我国信息化时代的不断发展以及经济的持续增长，数据信息关联和整合的需求也日益增长，小区管理员和业主之间的数据管理，能够更好地打造智能便利小区。因此、对物业相关信息进行管理成为了一个维护小区服务功能的一个重要环节。

随着各个小区入住的业主不断增多，业主的房屋信息或维修信息等日趋繁杂。因此，需要有一完整、便利的信息管理系统。通过MySQL打造数据库，业主的各项信息管理系统可以和管理员的操作系统有机地关联在一起，加大对信息的更新和检索工作力度。此外，通过系统地收录数据，管理员对业主的需求能够及时收到并且提高工资效率。

# 需求分析

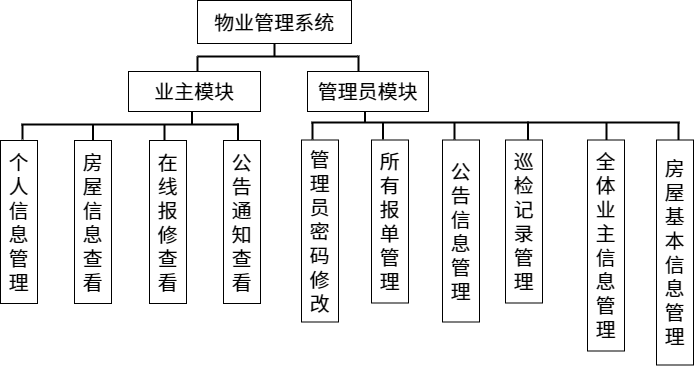
1. 功能需求分析

根据对物业信息管理的需求进行分析，系统共包含管理员和业主两个角色。其中，管理员的功能包括全体业主信息管理、房屋基本信息管理、巡检记录管理、公共信息管理、所有保单管理、管理员密码修改等。业主的功能包括个人信息管理、房屋信息查看、在线保修申请、公告通知查看等。

1. 可行性分析

本数据库项目作为学习实践内容，采用Python语言开发用户交互界面，并通过规范的数据库设计确保与MySQL的高效兼容性。在数据表结构设计方面，严格遵循关系型数据库的范式原则，各表间通过合理的主外键关联实现数据完整性，同时结合生活场景中的实际案例进行建模，使数据结构既清晰易懂又具备实用价值。技术选型上优先采用团队熟悉的Python及MySQL技术栈，显著降低了学习成本并提高了开发效率，最终在保证功能完整性的前提下，以较低的经济成本和较短的开发周期完成了这一简洁而高效的数据库系统实现。

物业信息管理系统的功能模块图如图1所示，系统由业主模块和管理员模块构成，图中展示的是系统模块包含的功能以及之间的关系。

图1 系统功能模块图

# 概念结构设计

(E-R图设计，数据字典)

本系统使用MySQL数据库，因为数据是系统的内容主要组成部分，数据能够被合理地存储，使得用户和管理员能够查看和管理。使用E-R图用来表示概念关系，可以让主题之间的关系和联系展示得更加清楚，用于描述现实世界的模型。本系统的E-R图如图2所示。

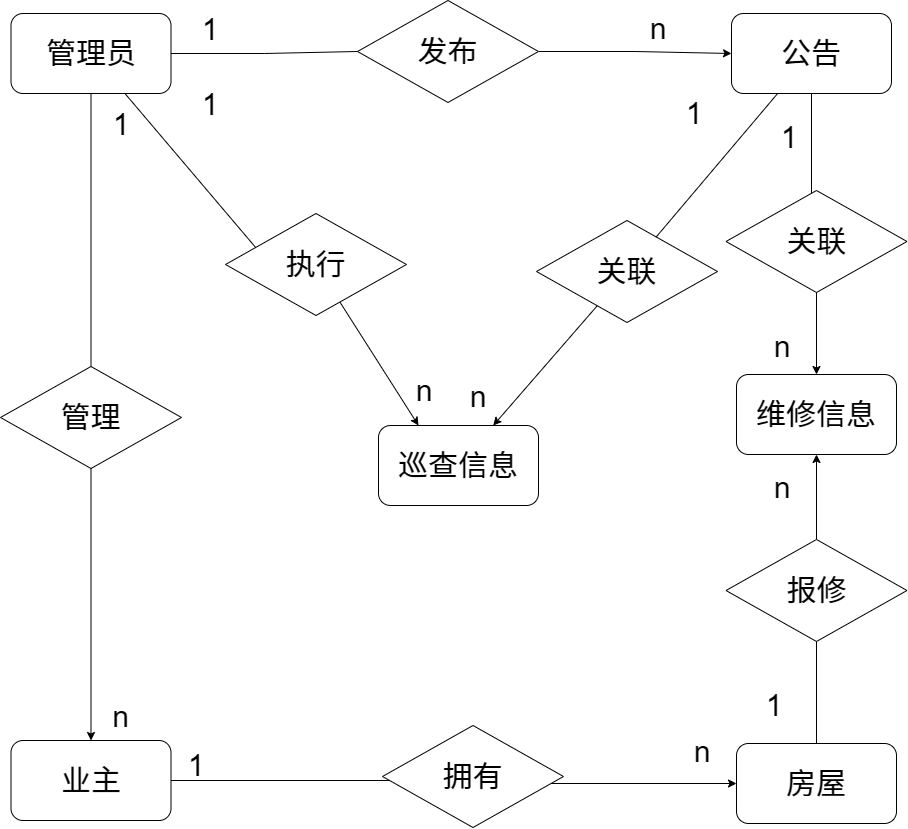


图 2 系统E-R图

1. 主要数据库

该系统采用MySQL数据库存储数据，数据库中的表以及表中的字段是对应实体及其相关性得到的。系统的主要数据表结构如下所示。

表 1房产信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **是否为主键** | **为空** |
| 房屋编号 | int(4) | 是 | 空 |
| 门牌号 | varchar(20) | 否 | 空 |
| 楼号 | varchar(20) | 否 | 空 |
| 房屋类型 | varchar(20) | 否 | 空 |
| 房屋所在地区 | varchar(20) | 否 | 空 |
| 房屋出售状态 | varchar(20) | 否 | 空 |
| 房屋单元号 | varchar(20) | 否 | 空 |
| 房屋楼层号 | varchar(20) | 否 | 空 |
| 房屋朝向 | varchar(20) | 否 | 空 |
| 信息备注 | varchar(100) | 否 | 空 |
| 业主编号 | int(4) | 否 | 空 |

表 2保安信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **是否为主键** | **为空** |
| 巡逻人编号 | int(4) | 是 | 空 |
| 巡逻人姓名 | varchar(20) | 否 | 空 |
| 类别 | varchar(20) | 否 | 空 |
| 事件时间 | date | 否 | 空 |
| 处理人姓名 | varchar(20) | 否 | 空 |
| 当事人姓名 | varchar(20) | 否 | 空 |
| 处理结果 | varchar(50) | 否 | 空 |
| 备注信息 | varchar(100) | 否 | 空 |

表 3保修管理表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **是否为主键** | **为空** |
| 事件编号 | int(4) | 是 | 空 |
| 报修物名称 | varchar(20) | 否 | 空 |
| 保修单状态 | varchar(20) | 否 | 空 |
| 房门号 | varchar(20) | 否 | 空 |
| 保修日期 | date | 否 | 空 |
| 维修日期 | date | 否 | 空 |
| 预计花费 | double(7,2) | 否 | 空 |
| 实际花费 | double(7,2) | 否 | 空 |
| 报修人姓名 | varchar(20) | 否 | 空 |
| 保修详情 | varchar(100) | 否 | 空 |

表 4公告信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **是否为主键** | **为空** |
| 公告编号 | int(4) | 是 | 空 |
| 公告内容 | varchar(100) | 否 | 空 |
| 公告日期 | date | 否 | 空 |
| 公告标题 | varchar(20) | 否 | 空 |
| 发布者 | varchar(20) | 否 | 空 |

1. 管理员模块

管理员作为本系统的最高权限管理者，负责业主账号的分配与物业管理核心事务的统筹，包括业主信息的录入与权限配置、房屋维修申请的审核与工单派发、设备巡检记录的核查与整改跟进，以及社区公告的发布与更新。在业主端，管理员通过后台维护基础数据，确保业主能够查看个人房产信息、提交图文报修申请并跟踪处理进度，同时接收分类推送的社区公告。

表5管理员信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **是否为主键** | **为空** |
| 管理员编号 | int(4) | 是 | 空 |
| 管理员名字 | varchar(20) | 否 | 空 |
| 管理员性别  管理员年龄 | varchar(20)  varchar(20) | 否  否 | 空  空 |
| 管理员性别 | Int(4) | 否 | 空 |
| 联系电话 | Varchar(11) | 否 | 空 |
| 居住地址 | Varchar(50) | 否 | 空 |
| 备注信息 | Varchar(100) | 否 | 空 |

# 逻辑结构设计

该数据表结构图展示了物业管理系统核心业务关系：管理员表作为中枢管理业主房产（通过业主编号关联房产信息表），巡查信息表记录巡查信息（巡查人关联管理员），公告表通过发布者关联管理员实现公告发布，并与维修表（通过房屋编号关联房屋）形成业务闭环，房产信息表则通过业主编号绑定业主信息确立业主房屋产权关系。外键房屋编号关联业主编号，实现删除业主信息调试情况房屋信息的效果。

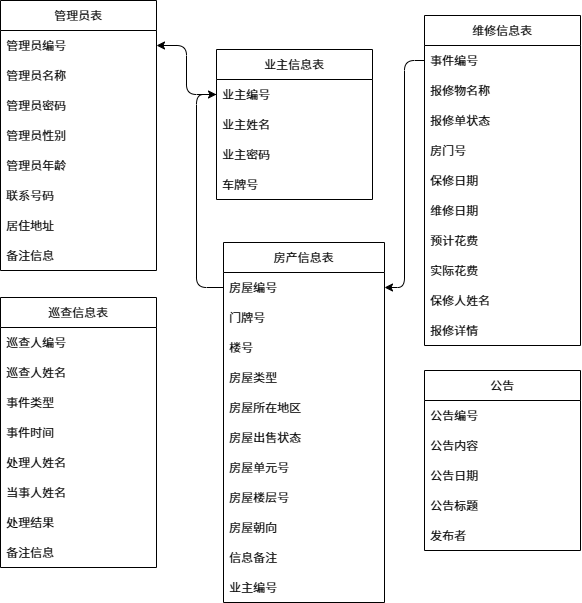


图 3数据库结构

# 系统实现

1. 业主模块

该登陆页面包括了业主和管理员登录，业主实现数据的呈现和查看，管理员实现数据的增删查改及其管理系统功能。该设计旨在实现读写分离，提高系统的整体性能和数据可扩展性，以下是系统封面。

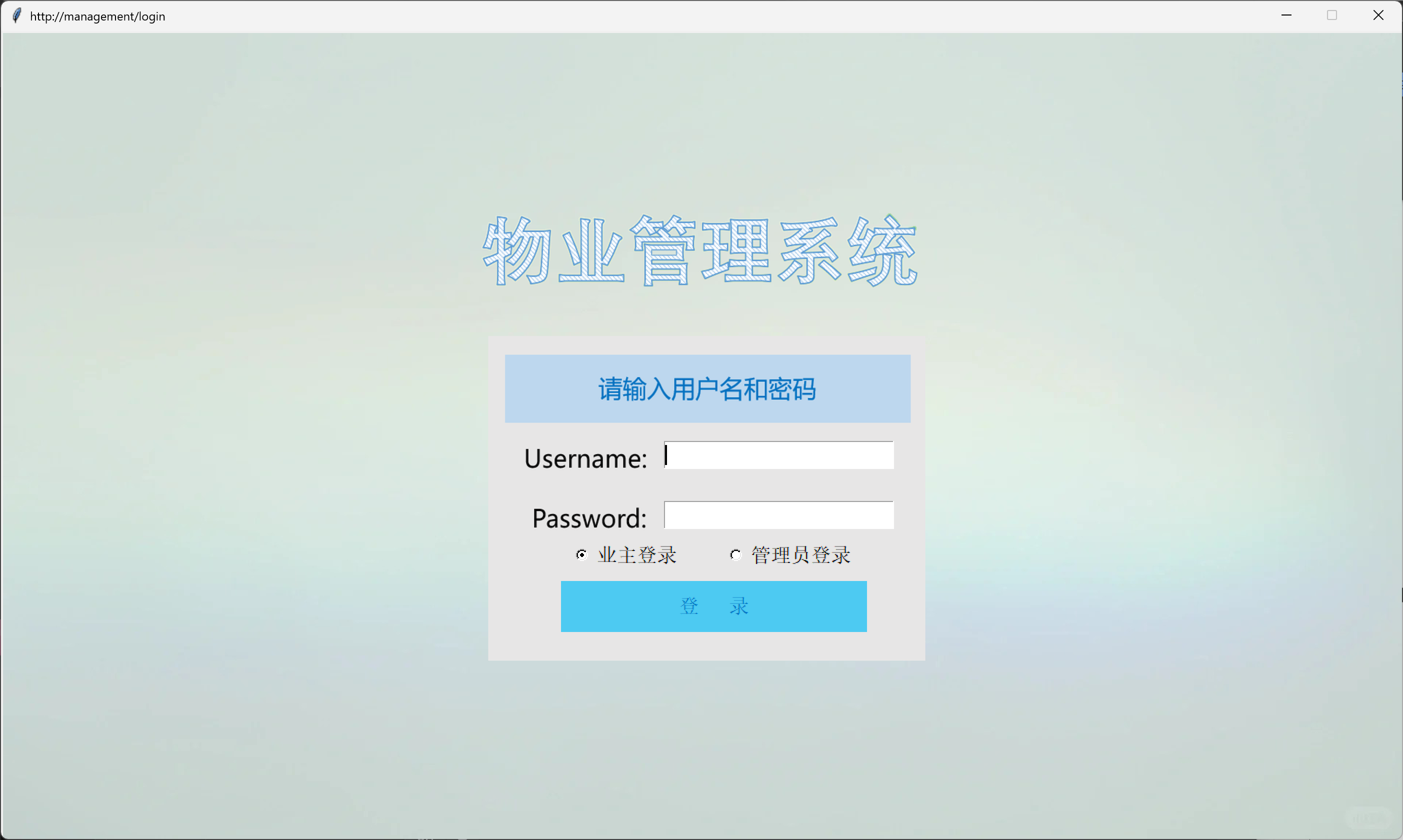


图 4系统登陆界面

业主是该系统展示平面设计和数据呈现的主要用户，选择业主登录后，可以查看各项数据，来完成信息的检查和验证。



图 5业主界面展示

1. 管理员模块

管理员作为系统的最高权限管理者，全面负责物业管理平台的运维工作。其核心职能包括：业主信息管理，可对全体业主的基本信息进行录入、修改和查询；房屋数据维护，确保房产信息的准确性和实时更新；报修工单处理，从分配到状态跟踪实现全流程管理；公告信息发布，及时推送社区重要通知；以及巡检计划制定与执行监督，持续提升服务质量。同时，管理员拥有密码修改等系统维护权限，为平台安全稳定运行提供保障。通过以上功能的协同运作，管理员能够有效统筹物业管理工作，为业主创造更优质的居住体验。以下是管理员页面截图以及删除、新增、修改运行示例。



图 6管理员界面展示



图 7删除功能示例



图 8新增功能示例



图 9修改功能示例

# 系统测试

物业信息管理系统的测试包括管理员模块的测试和业主模块的测试。通过测试用例，详细测试系统的各项功能，内容有不同弄角色登录，管理员角色的修改，各项模块信息的查询、录入、修改和删除等。结果测试表明，各项功能均可使用，却界面操作响应时间短。

表 6功能测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **用例编号** | **测试模块** | **操作步骤** | **输入数据** | **预期结果** |
| TC-ADM-001 | 管理员登录 | 1. 输入用户名和密码 2. 点击登录 | 用户名: admin 密码: admin | 登录成功，进入管理员主界面 |
| TC-ADM-002 | 添加业主 | 1. 进入业主管理 2. 填写业主信息 3. 点击保存 | 用户名: test\_user 密码: 123456 车牌: 津A88888 | 提示'添加成功'，custom\_account表新增记录 |
| TC-ADM-003 | 管理房产信息 | 1. 进入房产管理 2. 添加新房源信息 3. 点击保存 | 门牌号: A-1001 楼号: 5 类型: 高层 业主ID: 1 | house表新增记录，外键关联正确 |
| TC-ADM-004 | 处理报修单 | 1. 进入报修管理 2. 选择未处理报修 3. 更新状态和实际花费 4. 保存 | 报修ID: 1 状态: 已完成 实际花费: 150.00 | maintain表状态更新，scost字段更新 |
| TC-ADM-005 | 发布公告 | 1. 进入公告管理 2. 填写公告内容 3. 点击发布 | 标题: 停水通知 内容: 明天上午停水检修 发布者: 物业部 | notice表新增记录，业主端可见新公告 |
| TC-OWN-001 | 业主登录 | 1. 输入用户名和密码 2. 点击登录 | 用户名: ysj 密码: 1234 | 登录成功，进入业主主界面 |
| TC-OWN-002 | 查看个人信息 | 1. 进入个人中心 |  | 显示业主用户名、车牌信息 |
| TC-OWN-003 | 查看房产信息 | 1. 进入我的房产 |  | 显示关联的房产信息(门牌号、楼号等) |
| TC-OWN-004 | 查看报修记录 | 1. 进入报修记录 |  | 显示当前业主的报修历史记录 |
| TC-OWN-005 | 查看社区公告 | 1. 进入公告栏 |  | 显示所有已发布的社区公告 |

表 7数据库验证表

|  |  |  |  |
| --- | --- | --- | --- |
| **用例编号** | **测试动作** | **SQL验证语句** | **预期数据状态** |
| TC-DB-001 | 添加业主后 | SELECT \* FROM custom\_account WHERE username='test\_user' | 返回1条记录，车牌为'津A88888' |
| TC-DB-002 | 添加房产后 | SELECT \* FROM house WHERE num='A-1001' AND ownerid=1 | 返回1条记录，dep='5', type='高层' |
| TC-DB-003 | 更新报修单后 | SELECT status, scost FROM maintain WHERE id=1 | status='已完成', scost=150.00 |
| TC-DB-004 | 发布公告后 | SELECT \* FROM notice WHERE title='停水通知' | 返回1条记录，content包含'停水检修' |
| TC-DB-005 | 业主查看房产 | SELECT COUNT(\*) FROM house WHERE ownerid=(SELECT ownerid FROM custom\_account WHERE username='ysj') | 返回大于0条记录 |

表8异常流测试

|  |  |  |  |
| --- | --- | --- | --- |
| **用例编号** | **异常场景** | **非法输入** | **预期系统提示** |
| TC-EXC-001 | 使用错误密码登录(管理员) | 用户名: admin 密码: wrong | 提示'用户名或密码错误' |
| TC-EXC-002 | 添加重复门牌号 | 门牌号: A-1001 (已存在相同门牌号) | 提示'门牌号已存在' |
| TC-EXC-004 | 输入负数的维修费用 | 实际花费: -100 | 提示'费用不能为负数' |