# 设计说明书

|  |  |
| --- | --- |
| **题目**： | 基于python的物业管理系统的设计与实现 |
| **学院：** | 电子信息学院 |
| **专业：** | 人工智能 |
| **班级：** | 人工智能2301班 |
| **组员姓名：** | 魏锦翔、徐子尧、王伊蕾 |
| **指导老师：** | 谢乐、张瑜 |

目 录

[1 设计任务和要求](#_Toc13038)

1.1 设计任务

1.2 设计要求

[2 设计依据和设计原则](#_Toc3188)

2.1 设计依据

2.2 设计原则

[3 系统](#_Toc19151)设计

[4 系统](#_Toc25582)测试

[5 总结](#_Toc17318)

[参考文献](#_Toc30889)

# [1 设计任务和要求](#_Toc13038)

## 1.1设计任务

本文研究的目标内容主要任务由以下几部分组成:

第一，分析国内外物业管理系统的发展现状，调研多个小区物业管理体系运

营模式，具体操作了已使用的物业管理系统，从而发现并记录各系统存在的问题。

第二，与物业管理人员和小区业主进行沟通，了解他们的实际需求;调研小

区物业管理提供的服务。

第三，实现物业管理系统设计部分，根据需求分析构建系统功能设计、详细

设计(系统功能模块、数据库表等的设计)、构建数据库E-R图等。

第四，实现物业管理系统相关代码编程和测试，完成基于Python的物业

管理系统的设计与实现。

## 1.2 设计要求

目前系统设计主要采用的是Python语言来进行开发，采用Django框架技术，框架分为三层，分别是控制层Controller，业务处理层Service，持久层dao，能够采用多层次管理开发，对于各个模块设计制作有一定的安全性；数据库方面主要采用的是MySQL来进行开发，其特点是稳定性好，数据库存储容量大，处理能力快等优势；服务器采用的是Django服务，能够提供稳固的运行平台，确保系统稳定运行。通过物业管理系统来提升本课题的各项功能的工作效率，提供了一个多样功能，具有良好实用性的物业管理系统。本设计对项目物业管理系统进行了相关技术的分析与介绍，在结合技术的优点和适应性来选择数据库、服务器和开发环境，基于现有的情况，针对系统的各个模块确定系统的最优设计方案。

（1）鉴于该系统是一款面向全体使用者的系统，管理者需要录入使用者的个人信息、给予使用者应有的权限，让使用者可以更加自由的使用系统。

（2）要给予业主更高的查询权限，让业主可以全方位的了解本课题，让业主可以查询到车位信息、缴费信息、报修信息等。

（3）要提升业主的使用效率，让业主可以更加快捷的查询自己想要查询的东西。同时，还要做到同步性，在业主查询信息或者修改信息的同时，系统可以及时反馈到数据库上，进行及时更新和修复。

## 主要的技术：

## Django框架

Django是Python的一个 Web 框架，可以帮助使用快速搭建一个web项目。Django 是一个开放源代码的 Web 应用框架，由Python写成。采用了MTV的框架模式。Django的 MTV模式本质上和MVC是一样的，也是为了各组件间保持松耦合关系，只是定义上有些许不同，Django的MT分别是指：M表示模型（Model）：编写程序应有的功能，负责业务对象与数据库的映射 (ORM)。T表示模板(Template)：负责如何把页面(html)展示给用户[11]。V表示视图（View）：负责业务逻辑，并在适当时候调用 Model 和 Template。除了以上三层之外，还需要一个 URL分发器，它的作用是将一个个URL的页面请求分发给不同的 View处理，View再调用相应的Model和Template，MTV的响应模式[9]。

## B/S架构

B/S结构就是指系统客户端与服务器分离，客户端通过浏览器访问服务端进行操作。B/S结构目前广泛应用于绝大部分系统搭建中，这种结构摒弃C/S结构客户端服务端不分离的缺点，具有更多的优势：

（1）跨平台性：B/S的标准由标准化组织确立，适用于绝大多数的系统搭建，通用于应用之间。

（2）低维护成本：客户端和服务器端分离，减轻了两端的压力，尤其是客户端，对客户端设备，硬件、软件要求都比较低，并且系统需要升级或维护时，只需要在服务器端升级或维护就可以，使相应的费用减少。

## MySQL介绍

MySQL 是一种关联数据库管理系统，该系统速度快而且灵活性强，主要原因是它将数据保存在不同的表中，MySQL软件的优点是成本低、体积小、速度快，开放源码，所以一般中小型网站的开发都选择 MySQL 作为网站数据库。

## Python介绍

Python是解释型的脚本语言，在运行过程中，把程序转换为字节码和机器语言，说明性语言的程序在运行之前不必进行编译，而是一个专用的解释器，当被执行时，它都会被翻译，与之对应的还有编译性语言[10]。

同时，这也是一种用于电脑编程的跨平台语言，这是一门将编译、交互和面向对象相结合的脚本语言（script language）。

可以说其优点也是非常的突出，下面就其优点进行具体的说明：

（1）首先它是一种比较简单易学的，而且对初学者来说也是非常容易上手的一种语言，与其他的语言相比较在进行编写和阅读的时候，会更加接近人的思维模式，非常的适合用户的阅读性和易于理解的特点。

（2）由于其底层的逻辑是用c语言完成的，所以在运行的速度来说也是非常快的，就如许多的开发者所了解的，它的很多的标准库以及第三方的依赖都是由c语言所开发的，所以执行起来比较高效率。

（3）并且，它始终是一个面向对象的程序设计，它支持面向过程和面向对象的编写，在以流程为导向的语言里，一个程序是建立在一个过程或者一个简单的可重复使用的代码的功能上，在面向对象中，一个程序是以数据和函数为基础的对象构造的。

（4）如果需要代码的扩充或者是快速查看，在使用它的时候格式也是非常的清晰，还有便于维护的特点，在不同的开发人员共同开发的过程中，都会比较方便的理解所编写的内容。

（5）同时也是因为它开源的特点，能够被移植到很多的平台进行使用，并且它也是完全支持重载、派生、继承等有助于增强代码的复用的特点，同时有两个标准库（functools, itertools）提供了开发的支持。

# [2 设计依据和设计原则](#_Toc3188)

## 2.1 设计依据

本系统设计主要采用的是Python语言来进行开发，采用Django框架技术，框架分为三层，分别是控制层Controller，业务处理层Service，持久层dao，能够采用多层次管理开发，对于各个模块设计制作有一定的安全性；数据库方面主要采用的是MySQL来进行开发，其特点是稳定性好，数据库存储容量大，处理能力快等优势；服务器采用的是Django服务，能够提供稳固的运行平台，确保系统稳定运行。通过物业管理系统来提升本课题的各项功能的工作效率，提供了一个多样功能，具有良好实用性的物业管理系统。

### 2.1.1 Python框架的定义

### Django是Python的一个 Web 框架，可以帮助使用快速搭建一个web项目。Django 是一个开放源代码的 Web 应用框架，由Python写成。采用了MTV的框架模式。Django的 MTV模式本质上和MVC是一样的，也是为了各组件间保持松耦合关系，只是定义上有些许不同，Django的MT分别是指：M表示模型（Model）：编写程序应有的功能，负责业务对象与数据库的映射 (ORM)。T表示模板(Template)：负责如何把页面(html)展示给用户[11]。V表示视图（View）：负责业务逻辑，并在适当时候调用 Model 和 Template。除了以上三层之外，还需要一个 URL分发器，它的作用是将一个个URL的页面请求分发给不同的 View处理，View再调用相应的Model和Template，MTV的响应模式。

### 2.1.2 Python的基本描述

* Python的设计目标之一是让代码具备高度的可阅读性。它设计时尽量使用其它语言经常使用的标 …  
  Python开发者有意让违反了缩进规则的程序不能通过编译，以此来强制程序员养成良好的编程习惯。并且Python语言利用缩进表示语句块的开始和退出（Off-side规则），而非使用花括号或者某种关键字。增加缩进表示语句块的开始，而减少缩进则表示语句块的退出。缩进成为了语法的一 …
* age = int(input("请输入你的年龄: ")) if age < 21: print("你不能买酒。") print("不过你能买口香糖。…  
  根据PEP的规定，必须使用4个空格来表示每级缩进。使用Tab字符和其它数目的空格虽然都可以编译通过，但不符合编码规范。支持Tab字符和其它数目的空格仅仅是为兼容很旧的的Python程序和某些有问题的编辑程序。

## 2.2 设计原则

该算法的开发只需在Windows操作系统下进行，利用python进行开发，在计算机上只需安装一般的字处理软件和一些支持软件，不需要大型的系统软件和应用软件来支持。

### 2.2.1.经济可行性原则

任何系统在开发之前，经济方面的问题都是最应该被考虑的，经济问题主要体现在两个方面。一方面是在开发时，现有可使用的设计资源能否满足系统的预计开发消耗；另一方面是在开发以后，系统的盈利能否保证系统的正常运行和日复一日的更新消耗。

开发方面，因为在开发系统时使用的开发工具基本都是免费的，只有一些少量的人工费。所以，开发工具上的经济消耗都是很少的，都是在可控范围内的。另外，可以在确保质量的前提下，尽可能的减少设计成本，可以为开发之后的工作留下更多的经济资源。

在最大限度降低开发成本的同时，也可以让使用者最大限度的从系统中获利，对双方来说都是一个可行的物业管理系统。总结下来就是，该系统的各项开发费用都处于可控范围内。经过了种种分析和考虑后证明，该系统软件可以行得通的，经济方面的压力是在预料之内的。

### 2.2.2.技术可行性原则

在研究技术可行性的时候，系统的开发环境是可以确定的，所以技术可行性最好与系统功能和性能以及一些实际的情况同时考虑。在可行性研究阶段，结合实际调整开发的内容和选择能够完成的技术体系是一个可用的手段，如果系统进入开发阶段，任何的调整都意味着会耗费更多的经历。需要再次明确的问题是，技术可行性不只是考虑在技术上是否可实现，实际上还包含了在当前的实际的情况下的技术可行性。有很多的因素例如时间不足、预期的开发目标技术难度比较难、不能有充足的技术积累、而且对于技术的掌握不够等这些实际存在的情况，都是要提前部署和认真规划的。

开发物业管理系统的技术内容分析，目前的状况是从网络硬件和相关技术上看是可行的。该系统采用了Python和 MySQL技术。到现在可以知道，很多的基础知识都是经过学习使用的，在后续的设计中还需要不断的练习，学习更多相关的技术经验来充实自己对于系统的开发。

### 2.2.3.社会可行性原则

在社会可行性分析中，首先能结合国家经济和社会发展的实际情况，还需要能够让用户更加的便利，给使用者带来多方面的高效益，使得可以比较快速的发展以及应用，随着很多领域的发展，尤其是智能化的社会下，据统计许多的人都拥有属于自己的系统，这种情况下就带来很多的问题，比较突出的就是管理上的不足，从而出现低效率的原因[7]。

在可行性分析中的收益分析，是比较清晰地发掘开发的系统带来的各种直接的有效的利益以及一些隐藏起来的好处，以便在后续的设计中更加的坚定和确信系统开发的意义。解决一些社会方面存在的问题，设计并开发系统肯定是要发现某类社会存在的问题，并且能够带来一定的社会价值。还有一些社会的影响力，能够在一定的情况下提高效率，肯定会增加一些社会的进步和发展。社会上的可行性，包括法律上的可行性，也包括法律上的可能性，法律上的可行性，需要社会上的许多因素，可以来实现系统建设的现实性。如果所开发的系统与国家法律或政策等相关的因素不相符合，在某些信息化的邻域中使用的是一些加密的信息或者技术问题，还有不经过正常的操作使用其他的一些公开的信息等这些情况，这样的系统的开发在法律的范围中就是不能被允许的。

其中还有一个比较重要的用户使用可行性也是通过执行系统时的可行性，是从所有情况下用户的角度来考虑系统的可行性，由于设计开发相关的系统就是站在社会中广大的用户体验考虑的，所以在这个层面考虑也是符合的。

# [3 系统设计](#_Toc19151)

系统设计要提出具体的软件功能及要求，是系统分析后进行的流程。它的主要目的是提供软件开发的思路和具体流程，是系统运行的根本保障。它要确定软件各个模块的功能，并形成相应的流程图，方便设计者思考。同时要将每个模块需要完成的任务详细的表达出来，并对每个模块进行具体化的设计分析，提供相当数量的设计方案，再通过设计者的详细分析和比对，选择出最佳的方案进行设计开发。并对相应的方案进行验证和拓展。

3.1系统功能结构

3.1.1 系统结构图

系统结构图可以把杂乱无章的模块按照设计者的思维方式进行调整排序，可以让设计者在之后的添加，修改程序内容的过程中有一个很明显的思维方向[16]。同时结构图还可以让设计者以外的其他人更直观的看出整个系统的设计理论和设计过程。让系统的设计变的更直观。该系统的功能架构图如图3-1所示：



图3-1 物业管理系统功能结构图

3.2 数据库设计

3.2.1 数据库概念设计

从系统运行完整的角度出发，设置相应的实体，明确各实体之间的联系。下面列举整个系统中涉及核心功能的实体：

1. 业主信息实体



图3-2 业主信息实体图

1. 车位信息实体



图3-3 车位信息实体图

1. 缴费信息实体



图3-4 缴费信息实体图

1. 报修信息实体



图3-5 报修信息实体图

# [4 系统测试](#_Toc25582)

系统测试是指在设计基本完成后对软件的各项功能进行测试，检测其能否达到在系统设计之初对其的预期情况[18]。预估系统上线后的情况，对出现问题的地方进行及时修改。系统测试作为系统设计的最终阶段，它的准确性往往影响了系统的最终使用感。因为软件是由数据、文档以及程序代码组成的，所以系统测试的对象就不仅仅是程序的本身，还有包括软件设计过程中形成的数据、文档等。但系统测试也存在着问题，系统测试注重的是软件设计的最终结果，其并没有关心整个系统在开发过程中发生的问题。系统测试必须假设和考虑在设计过程中发生的问题。并对系统测试中发现的问题进行回归测试。要更好的通过系统测试检测系统以更好的找出问题来提高系统的质量。系统测试要以最少的人力和经济消耗来完成系统的设计结尾工作。而且所有的测试都要以人为本，任何的测试都要考虑到业主使用的实际体验感，要确保业主有一个好的使用体验。要确保整个系统测试过程都按照原定的规则进行，以确保最终设计的系统的完整性和实用性。

4.1 系统测试原则

编筐编篓，重在收口[19]。系统测试的完成效果直接影响到整个系统的使用效果。对于系统测试来说，如果能早点发现系统的漏洞，就能及时修改，可以让设计的成本减少。进行系统测试的时候，测试员们需要制定自己的测试脚本，让这个脚本尽可能多的覆盖系统需要测试的各个路径，避免系统疏漏。同时，在系统测试的过程中，要检测程序有没有进行与系统预期功能要求无关的代码运行。避免多余的程序运行对系统产生破坏，降低效率，增加成本。同时在测试后，还要尽量保留所有测试用例，这样可以帮助日后想要修改时可以及时找到原先的用例，可以再直接进行测试。以下是总结下来比较重要的一些系统测试原则：

（1）要尽可能早的进行系统测试。

随着系统的设计进行，设计过程中每一个阶段发生的错误都有可能导致在后续的开发设计的时候连续出错，如果等到软件设计工作基本全都完成再进行测试，可能会导致大量的错误同时出现，而且很难找到它们出错的源头[20]。因为开发设计的每一步都与后续程序紧密相连，如果不及时找出错误，可能会因一处错误导致后续整个程序瘫痪。所以要坚持尽早进行系统测试的原则，一来可以减少牵一发而动全身的错误，二来可以降低成本。

（2）系统测试证明软件存在缺陷。

通过系统测试可以检测系统设计时是否存在错误和缺陷，但并不能说明系统是不存在任何缺陷的。系统测试的目的是辅助软件开发，让程序员可以尽可能早的发现错误，提高质量。软件开发的过程或多或少都会产生一些错误，系统测试就是尽可能地发现错误和缺陷，尽可能优化程序。

（3）缺陷集群效应原则。

一个软件核心的地方可能就只有那么小的一部分，但就是那一小部分，却能引发软件大规模的损坏。当我们进行系统测试的时候要特别注意那一小部分，也就是核心的部分，因为它们几乎控制了整个软件的稳定性，它们是最容易出现错误的，控制好它们几乎就控制了软件的八成稳定性。但我们也不能只仅限于这小部分模块的检查，也要对剩余的地方进行适当的测试。

（4）以人为本原则。

物业管理系统设计的目的就是为人服务，任何的缺陷和错误的修改方向都要在使用者的角度去进行。要做到尽可能贴近使用者的日常模式，从使用者的角度去出发，去修改。

（5）经常更新测试方式。

测试人员应该尽量避免对系统所有的部分都使用同一种测试方式。一些测试方式只能检测特定的模块，并不能对整个系统都进行有效的测试，甚至无法发现一处错误。所以，为了能更好的发挥系统测试的作用，系统测试的时候应该经常更新测试方法，让测试的质量最大化。

4.2 测试用例

部分系统测试用例如表4-1所示。

表4-1 测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 测试功能、流程 | 预期结果 | 实际结果 | 结论 |
| 进行登录实验。需要管理员输入相应的信息验证，看能否进入 | 录入信息验证通过，  转入管理员界面 | 登陆成功，  成功跳转页面 | 测试成功 |
| 对业主的登录进行测试。管理员就录入特定的信息，看能否成功登录 | 录入特定信息，  可以登录，可以在数据后台找到信息 | 可以登录，在后台找到了对应信息 | 测试成功 |
| 测试登录功能是否符合规定。测试员要输入错误的登录信息，检查能否登录 | 在输入错误的信息后，系统会发现错误，禁止测试人员登录 | 输入错误信息后，系统禁止登录 | 测试成功 |

续表4-1 测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 测试功能、流程 | 预期结果 | 实际结果 | 结论 |
| 对兼容性进行测试，让软件在不同的电子类产品上进行登录测试 | 都可以成功打开软件，正常运行 | 可以在不同的设备上和其他软件共存 | 测试成功 |
| 安装测试，在试验机上登录软件，检测能否按预期正常运行 | 登录后可以进行正常操作，可以发行 | 在试验机上可以正常运行，没有卡顿 | 测试成功 |

4.3功能测试

下面对登录模块进行测试，表格里分别是输入用户名和密码后的三种不同的结果，当用户名错误时会提示用户名错误，当密码错误时会提示密码错误，当账户与密码都正确时会提示登录成功，如果以用户的账户密码登录管理员模块会提示登录失败。登录模块测试如表4-2所示。

表4-2 登录测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块名称 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 登录模块 | 用户名: admin  密码: 123 | 弹出错误提示，提示密码错误 | 弹出错误提示，提示密码错误 | 通过 |
| 登录模块 | 用户名: 123  密码: 123 | 弹出错误提示，提示用户名错误 | 弹出错误提示，提示用户名错误 | 通过 |
| 登录模块 | 用户名: admin  密码: admin | 管理员登录成功 | 管理员登录成功 | 通过 |

通过删除一则最新通知，来判断此功能是否通过测试，当选择删除这个模块时，会提示是否确认删除，确认后会提示删除成功，页面自动跳转。删除模块测试如表4-3所示。

表4-3 删除模块测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块名称 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 删除分类模块 | 分类名：最新通知 | 删除成功、页面自动跳转 | 删除成功、页面自动跳转 | 通过 |

通过输入错误的原密码、不一致的新密码和正确的密码来测试修改密码的功能是否通过测试。修改密码模块测试如表4-4所示。

表4-4 修改密码测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块名称 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 修改密码模块 | 原密码：666  新密码: 123  确认密码：123 | 弹出错误提示，提示原密码错误 | 弹出错误提示，提示原密码错误 | 通过 |
| 修改密码模块 | 原密码：admin  新密码: 123  确认密码：333 | 弹出错误提示，提示用确认密码不一致 | 弹出错误提示，提示用确认密码不一致 | 通过 |
| 修改密码模块 | 原密码：admin  新密码: 123  确认密码：123 | 密码修改成功 | 密码修改成功 | 通过 |

通过上述功能测试，结合测试结果可以看出，该项目的基本功能是可以实现的，系统中的功能很全面并且界面设计简洁清晰，最终通过了系统测试。

# [5 总结](#_Toc17318)

物业管理系统的目的是让使用者可以更方便的将人、设备和场景更立体的连接在一起。能让业主以更科幻的方式使用产品，体验高科技时代带给人们的方便，同时也能让业主体会到与以往常规产品不同的体验风格。

物业管理系统在流畅性，续航能力，等方方面面都有着很大的优势。这就意味着物业管理系统的设计可以比其他系统更为出色的能力，可以更高效的完成最新的业主信息、服务类型、便民服务、车位信息、缴费信息、报修类型、报修信息等功能。

物业管理系统通过 Django框架、Python语言等技术完成的。系统可以完成业主的基本需求，并且在测试中系统运行情况良好，没有出现异常。但是，还是有许多我需要学习和完善的地方。

物业管理系统实现了模块化连接，对应不同的平台可以进行针对性部署，可以进行全方位的覆盖。它实现了跨终端无缝协同体验。但碍于物业管理系统刚进入市场发行不久，它的各项数据，功能还没有达到完美的地步。需要进一步的实际运行来完善有缺陷的地方。在这个信息化的时代大家都希望手里的任何一个电子设备都可以完成平时的常见功能，通过物业管理系统的开发可以实现这一目标。物业管理系统作为国产系统，它的发展前途十分明亮，未来也会拥有很广泛的市场，通过未来的长时间发展，物业管理系统的软件开发一定会成为一种趋势，让中国的操作系统发展得到进一步的提升。

参考文献

1. SilberschatzKorth Sudarshan 著，北京大学 杨冬青 李红燕 唐世渭 等翻译 数据库系统概念（原书第6版）机械工业出版社，2019.
2. 于小鹏 MySQL 2019 清华大学出版社, 2021 .
3. 陈淑红. 智慧物业综合运维管理系统设计与应用[J]. 绿色建造与智能建筑, 2022(11):4.
4. 唐清华. 小区智慧物业管理系统的设计及应用[J]. 中阿科技论坛(中英文), 2022(10):134-139.
5. 胡风娟. 基于SSH架构的物业管理系统的设计与实现[J]. 电脑开发与应用, 2014, 27(11):4.
6. 刘廷龙. 基于Android平台智能门禁管理系统设计与实现[J]. 信息通信, 2016(1):3.
7. 段巍. 望月湖小区物业管理系统设计与实现[D]. 大连理工大学, 2014.
8. M Fathy, M Barshan, S Yousefi. Improving the availability of P2P-based network management systems by provisioning fault tolerance property.
9. 代瑶. 智能物业管理系统开发与设计[J]. 电子制作, 2015(2Z):1.
10. 唐开年. 基于B/S架构的物业管理系统的设计与实现[D]. 电子科技大学, 2015.
11. 肖宏启、苏畅著, .Python网站开发项目化教程(第2版) 清华大学出版社 2021.
12. 徐洁罄. 数据库技术实用教程中国铁道出版社 2020.
13. 赵良辉.MySQL数据库轻松实务-基于MySQL 2015.MySQL 2016 华南理工大学出版社 2019.
14. [英]马克·J.普赖斯（Mark J. Price）著王莉莉译清华大学出版社 C# 8.0和.Python Core 3.0高级编程 2020.
15. [英]约翰·夏普（John Sharp）著周靖译 Visual C#从入门到精通（第9版）清华大学出版社 2019.
16. 黄梯云李一军.管理信息系统（第7版） 北京: 高等教育出版社,2019年.
17. 胡艳丽. MySQL 2019数据库原理及应用北京：清华大学出版社2021.
18. Alex Petrov，Database Internals (A deep-dive into how distributed data systems work).US: O'Reilly Media. Inc, 2020.
19. Abraham Silberschatz Professor / Henry F Korth / S Sudarshan. Database System Concepts (7th Edition), US: McGraw-Hill Education, Inc, 2020.
20. James Singleton. ASP.Python Core 2 High Performance. British:Packt Publishing, Inc,2019.