|  |  |
| --- | --- |
|  |  |

**本科生毕业论文（设计）**

基于Python的租赁房屋管理

系统与实现

专 业 电气工程及其自动化

姓 名 王举

学 号 16031107

指 导 教 师 陈垚

完 成 时 间

**教务处 印**

**摘要**

随着我国经济的快速发展的日益加速，城市流动人口逐年增多，社会对房屋租赁的需求也因此快速增加，房屋的出租和租赁已经成为生活的重要组成元素。房屋租赁的管理本身就是一项琐碎、复杂而且又十分细致的工作，传统的管理方式通常会耗费工作人员的大量时间和精力，而且相关信息也容易出错。伴随着这项业务的开展，高效、合理的管理房屋租赁的重要性也日益凸现。另一方面，由于互联网技术在近几十年的飞快发展，传统的商业活动逐渐引入了互联网技术，企业利用互联网技术极大提高了商务运作效率，房屋租赁业务也从传统的手工登记模式走入了网络空间。“基于Python的租赁房屋管理系统”正是为了实现高效、准确进行房屋管理而形成的一种新型信息化房屋管理系统。

本文首先对该系统的开发环境（PyCharm、HBuilder X和微信小程序开发者工具）进行了概述，在仔细考量了各种技术的优点和缺点之后，结合本系统的需求和特点提出本系统采用的前台和后台实现方式:该系统分前台和后台两大模块，考虑到Python的Django框架强大的后台管理能力，因此后台管理功能的实现主要采用Django框架；此外， Python的Flask框架又特别的小巧、灵活，因此实现前台API接口的实现采用的是Flask框架，值得一提的是由于DCloud公司推出的uni-app前端框架具有跨平台、可转换、小巧等优点，故前台的前端代码完全采用uni-app实现。需要指出，在该系统中数据的缓存采用的是Redis缓存技术，数据的持久性存储采用的是MySQL技术。同时也引入阿里云的短信验证、身份认证以及OSS存储等第三方技术。

该系统的后台模块针对的用户群体主要有超级管理员、赞助管理员和审核管理员。其中超级管理员的权限有角色管理、用户管理、公告发布和删除、评论管理、投诉管理；审核管理员则可以进行房屋信息的审核和公告审核；赞助管理员就只能进行广告的管理和代金券的管理。

该系统的前台针对的群体则主要是普通租户或房东。其中普通租户便只能进行房屋信息查看以及评论、投诉等基本操作，如果希望发布房源，便可以在升级为房东之后发生该操作。

**关键字:**房屋租赁，Python，Django，Flask，uni-app，MySQL，后台，前台

**Abstract**

With the rapid acceleration of China's rapid economic elopment， the city's floating population is increasing year by year， and the society's demand for housing leasing has therefore rapidly increased. Housing leasing and leasing have become an important element of life. The management of house leasing itself is a trivial， complicated and very meticulous job. The traditional management method usually consumes a lot of time and energy of the staff， and the related information is also prone to errors. With the development of this business， the importance of efficient and reasonable management of house leasing has become increasingly prominent. On the other hand， due to the rapid development of Internet technology in recent decades， traditional business activities have gradually introduced Internet technology. Enterprises have used Internet technology to greatly improve the efficiency of business operations. Housing leasing business has also entered from the traditional manual registration model. Cyberspace. "Python-based rental housing management system" is a new type of information-based housing management system formed for efficient and accurate housing management.

This article first outlines the development environment of the system (PyCharm， HBuilder X， and WeChat applet developer tools). After carefully considering the advantages and disadvantages of various technologies， this paper combines the needs and characteristics of the system and proposes the use of the system. Front-end and back-end implementation methods: The system is divided into front-end and back-end modules. Considering the powerful background management capabilities of Python's Django framework， the background management function is mainly implemented by Django framework; in addition， Python's Flask framework is particularly small， Flexible， so the front-end API interface is implemented using the Flask framework. It is worth mentioning that the uni-app front-end framework introduced by DCloud has the advantages of cross-platform， convertible， and compact， so the front-end code of the front desk is completely uni- app implementation. It should be pointed out that the data cache in this system uses Redis cache technology， and the persistent storage of data uses MySQL technology. At the same time， third-party technologies such as Alibaba Cloud SMS verification， identity authentication and OSS storage are also introduced.

The user groups targeted by the background module of the system mainly include super administrators， sponsor administrators and audit administrators. Among them， the super administrator's permissions include role management， user management， announcement release and deletion， comment management， and complaint management; the audit administrator can conduct house information audit and announcement audit; Management of coupons.

The group targeted by the front desk of the system is mainly ordinary tenants or landlords. Among them， ordinary tenants can only perform basic operations such as viewing house information， comments， and complaints. If you want to release a house， you can do this after upgrading to a landlord.

**Keywords:** housing lease， Python， Django， Flask， uni-app， MySQL， background， front desk

**目录**

**第一章 绪论1**

1.1研究的目的和意义1

1.1.1 研究目的1

1.1.2 研究意义1

1.2课题的国内外研究现状1

1.2.1 国内现状1

1.2.2 国外现状3

1.3论文的主要功能与结构3

1.3.1 论文的主要作用3

1.3.2 论文的主要结构4

**第二章 开发环境及相关技术介绍**4

2.1开发环境介绍4

2.1.1 PyCharm4

2.1.2 HBuilder X5

2.2相关技术介绍5

2.2.1 Python5

2.2.2 Web框架5

2.2.3数据库技术6

2.2.4 uni-app6

**第三章 可行性分析和业务分析**7

3.1可行性分析7

3.1.1 技术可行性7

3.1.2 经济可行性7

3.1.3 操作可行性7

3.2 业务分析7

3.2.1 用户群体7

3.2.2 业务模型图7

3.2.3 模型表建立8

**第四章 系统设计及测试**

4.1数据库和数据表设计

4.1.1 数据库和数据表的概念和关系

4.1.2 数据表设计

4.1.3 数据表的逻辑关系

4.2系统后台设计

4.3系统前台设计

4.4系统测试

**第五章 总结**

**附件**

**参考文献**

**致谢**

**第一章 绪论**

**1.1 课题研究的目的和意义**

**1.1.1 研究目的**

近年来，我国经济发展迅速，城市化建设步伐日益加速。作为世界第一人口大国，第二大经济体，我国各大城市流动人口逐步增加，导致对房屋租赁的需求也日益剧增。房屋中介也大量地涌现了出来，出租、租赁的大量业务量产生了对高效合理的租赁业务信息管理系统的迫切需求。另一方面，因为计算机的崛起，设计一款信息管理系统，可以极大的提高人们工作效率。

**1.1.2 研究意义**

由于国外信息技术起源比较早，在各个行业的应用也较为普及，国外信息化在房屋租赁方面的管理应用也比较成熟。在欧美等发达国家约有一半的中低收入者都是依靠房屋租赁来免租住宿需求，这一数字远远多于发展中国家的房屋租赁人群。在国内，经过多年的经济建设和城镇化建设，国内农民所占全国总人口的比例逐渐减少，越来越多的农民主键有乡村转移到城市，还包括大批从乡村流动到城市的大学生，这些大量流动的人口组成了租赁市场的主力军。庞大的流动人口数目产生了蓬勃的房屋租赁市场，然而，国内的房屋租赁市场还不发达。进入信息化时代，房产行业的发展和居民的租房需求的多元化带动了房产中介的发展，虽然存在这比较多的中介网站，单中介的信息化管理还不够完善，不能够对提高晓丽和降低成本起到至关重­要的作用。在我国房产中介仍然限于传统的方式，信息都是孤立的。未来房产租赁信息市场面临着网络化发展的趋势，从而实现房产交易中共享信息的功能，增加交易的机会，缩短交易的时间。

**1.2 课题研究的国内外现状**

**1.2.1 国内现状**

随着我国城镇化的快速发展，越来越多的人民进入城市工作和学习。同时，我国高校的大面积扩展，将更多的毕业高校学生留在了城市工作。这为房屋租赁带来了大量的客源。这也促进了我国房屋租赁市场的开发以及各种系统网站的发布。而我国的房屋租赁市场并不是一撮而就地发展，而是经过了几个过程。

中国加入 WTO 后，房屋租赁行业受到国外竞争者的挑战。国外企业凭借先进的信息分享平台和互联网技术，对我国房屋租赁市场造成很多冲击。面对国外的竞争，原先以手工方式进行管理房屋信息的我国企业开始自己摸索和探究房屋租赁系统的开发。

进入新世纪，我国的互联网企业得到快速发展，一大部分互联网企业掌握了核心技术并结合我国市场的特色都已发展长较大的企业。例如赶集网、58 同城等。房屋租赁市场也都称为各大第三方平台业务的一部分，得到了快速发展。而我国的中小企业的房屋租赁系统发展较缓慢，各自之间没有数据共享，各自为阵。系统设计开发也不规范，没有留下相关的数据接口供第三方使用，以利用“互联网+”的思维进行数据挖掘。现如今我国房屋租赁系统还存在如下一些问题：

（1）房屋信息不正确。很多企业发布的信息与实际房屋信息不符合，并且太多的房屋信息给用户甄别带来了很大的难度。

（2）信息化程度低、专业化程度不高。租赁管理系统还无法适应现代信息管理系统的要求。

（3）信息管理系统专业人才短缺。

基于上述问题，我国很多学者对我国的房屋租赁管理系统做了一些研究。还有部分学者根据存在问题设计了一些简单的系统供借鉴。主要的研究学者有，张哲使使用 C#，ASP.NET 开发环境，采用 MVC 框架设计实现了房屋销售与租赁信息网站。唐娜从社区暂住人口为切入点使用.net 实现了房屋出租管理系统。施磊设计了基于 B/S 结构，使用.net 技术设计了房屋租赁管理系统；方义从房屋租赁管理系统的设计与开发中的问题出发，进行了相关研究，供其参考。任小艳使用 UML 建模语言分析房屋租赁管理系统。李宁使用 PHP 开发设计了一个房屋租赁管理系统。吴琼和张旭超使用 B/S 模式对房屋租赁管理系统进行了设计。高海静.采用 MVC 架构设计了一个新型房屋租售管理系统。另一些学者从别的学科借鉴技术设计了房屋租赁管理系统。例如李国华使用 GIS 技术进行了设计。

**1.2.2 国外现状**

由于信息技术产生于欧美，房屋租赁系统的开发也较早在欧美国家进行，房屋租赁市场的信息化早已完成。但国外技术房屋租赁系统的完善，使得很多学者关心使用管理思想解决在房屋租赁过程中产生的问题以及结合功能模块自己开发系统的框架。主要的研究学者包括：Forgionne G A 使用一种决策系统方法分配军队租房距离的算法，尽可能减少距离。Park M L 等提出了一种新的技术管理住房问题。Darinka Czischke 认为租房部门应保持高度警惕，应采用新的战略和技术提高房屋租赁管理的效率，这样有利于管理租赁房屋信息，并提出设计了一个可以提高经理管理效率的出租屋管理系统。对技术和方法改进的还有 Wei P X和 Roberts A等学者的研究。

国外成熟的房屋租赁公司利用Internet 实现对房源和客源双重信息的管理，通过 MLS(多重上市服务)对房源和客源信息对接，实现了两个数据库的联合管理。此外还有一些学者强调了房屋租赁管理系统的重要性。

**1.3 论文的主要功能与结构**

**1.3.1 论文的主要作用**

本文首先认真研究了当下各优秀的Web框架及数据库相关技术的理论知识以及各自的优缺点，然后从该项目的业务流程、功能性需求以及使用场景等多个方面从何考虑，基于Python开发出了一套适用于国内现状的房屋租赁系统。本论文所做的主要研究工作如下：

(1)对国内房屋租赁市场进行了一定的需求分析和研究，同时结合市面上已经成熟的租赁系统，搭建出了一套适用于国内的反诬租赁系统的业务流程图。

(2)分析管理系统的结构的特点以及各Web框架的优缺点，选择了合适的Web进行来针对性实现功能。

(3)针对数据的存储的目的以及使用场景进行分析，采用MySQL数据库进行数据的持久性存储，同时选择Redis进行了数据的缓存。

(4)采用相关技术，完成基于Python的房屋租赁系统的整体结构设计和功能模块实现。

**1.3.2 论文的主要结构**

本文共分五章。

第一章，介绍该系统的研究意义和背景、国内国外的研究现状以及本论文的主要功能与结构；

第二章，介绍本系统采用的开发环境以及主要技术；

第三章，对本系统进行可行性分析和业务分析；

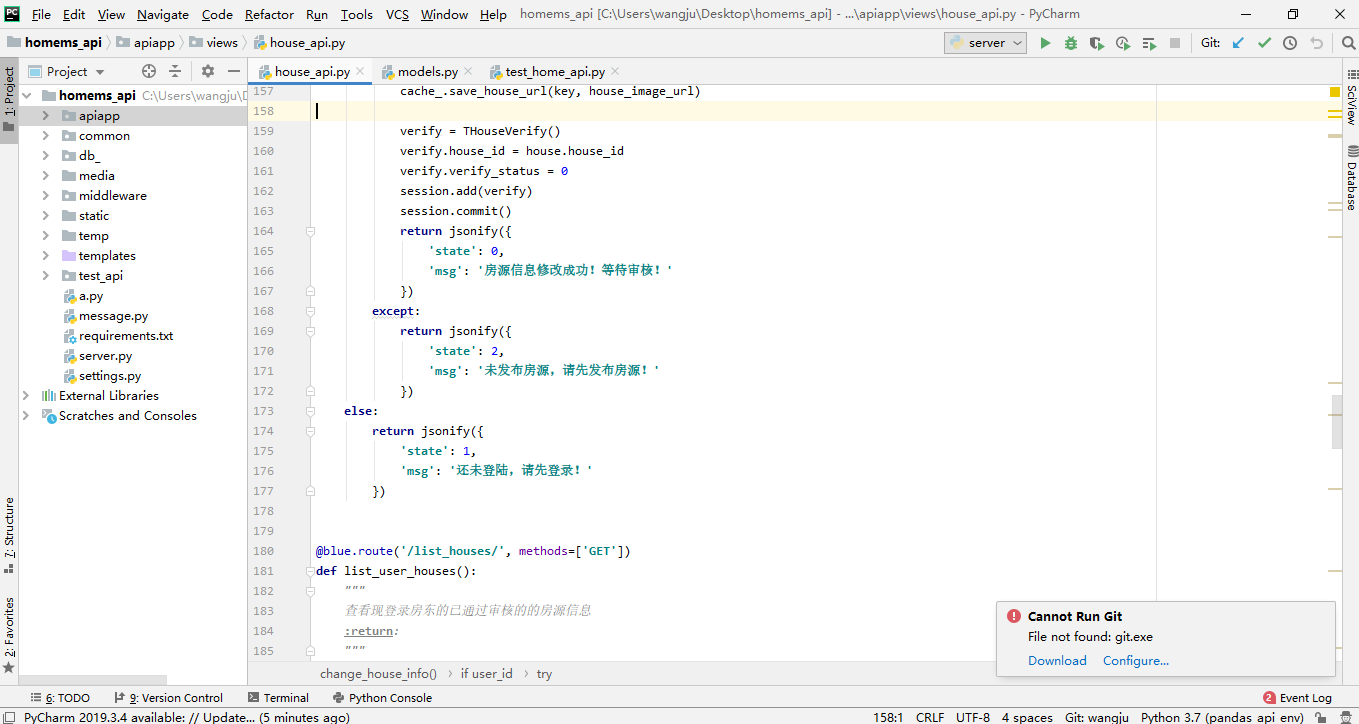
第四章，完成本系统的及数据库设计、前后台的功能实现以及测试；

第五章，总结。

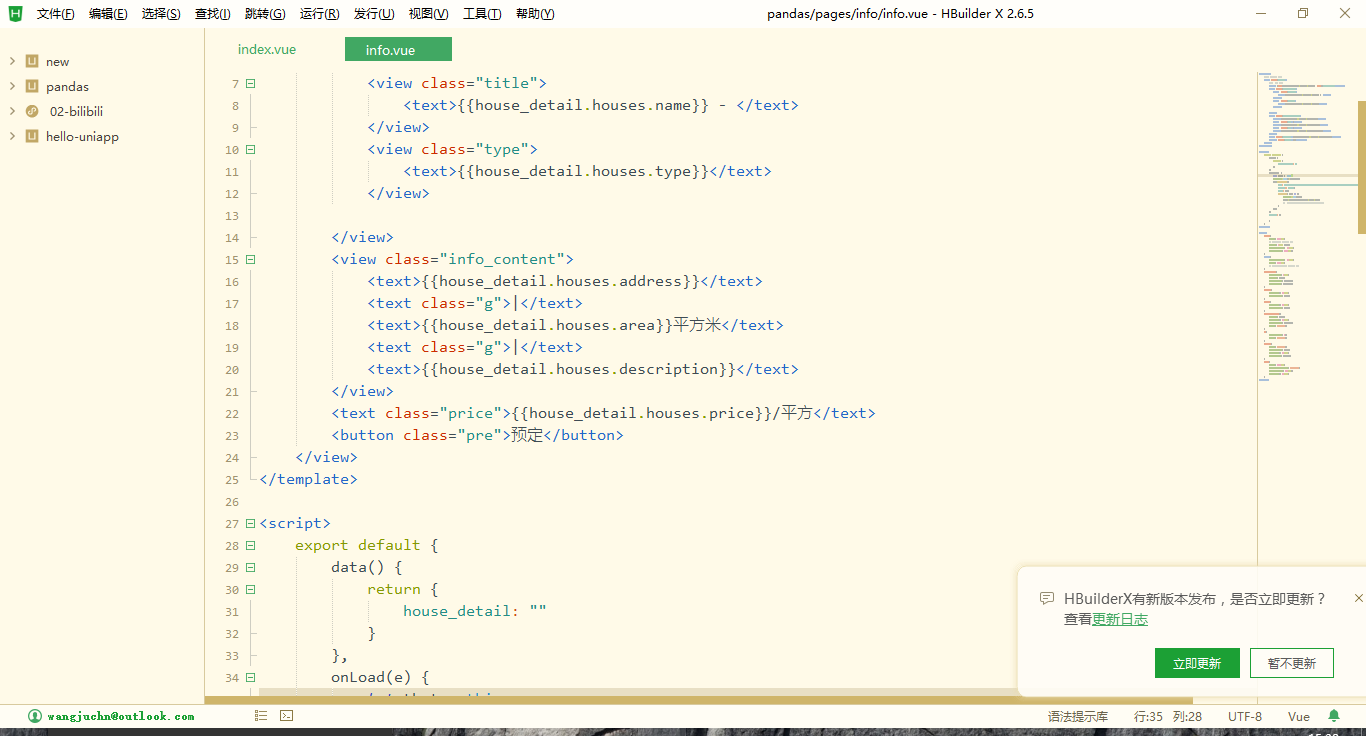
**第二章 开发环境及相关技术介绍**

**2.1 开发环境介绍**

**2.1.1 PyCharm**

PyCharm是由JetBrains打造的一款Python IDE，带有一整套可以帮助用户在使用Python语言开发时提高其效率的工具，比如调试、语法高亮、Project管理、代码跳转、智能提示、自动完成、单元测试、版本控制。此外，该IDE提供了一些高级功能，以用于支持Django框架下的专业Web开发。截图如下：

**2.1.2 HBuilder X**

HBuilder X是DCloud（数字天堂）专为vue推出的一款Web开发IDE。HBuilder X同时具有着不但轻巧、急速，而且有着相当强大的语法提示功能。HBuilder X可视化的方式比较简单，HBuilder X内置相关环境，开箱即用，无需配置nodejs。HBuilder X是通用的前端开发工具，但为uni-app做了特别强化。截图如下：

**2.2 相关技术介绍**

**2.2.1 Python**

Python是一种跨平台的[计算机程序设计语言](https://baike.baidu.com/item/%E8%AE%A1%E7%AE%97%E6%9C%BA%E7%A8%8B%E5%BA%8F%E8%AE%BE%E8%AE%A1%E8%AF%AD%E8%A8%80/7073760" \t "_blank)，是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。最初被设计用于编写自动化脚本(shell)，随着版本的不断更新和语言新功能的添加，越多被用于独立的、大型项目的开发，同时随着人工智能时代的到来，Python语言的使用范畴也正在迅速扩大，并且，Python也拥有者相当多的第三方库，也可以直接和各种数据库进行交互，极大地简化了程序员的开发成本。

**2.2.2 Web框架**

Python下有许多款不同的 Web 框架，在该系统中主要用到了Django和Flask框架。可以这样理解，如果把Flask比喻成一个毛坯房，内部家具基本没有，然而Django就相当于是一栋别墅，内部家具齐全。因此Django是重量级选手中最有代表性的一位，许多成功的网站和APP都基于Django，Django是一个开放源代码的Web应用框架，是基于Flask用Python写成的，特点则是大而全；相比于Django，Flask框架就显得比较轻量级。

**2.2.3 数据库技术**

数据库简单来说，其实就是数据的仓库，被称为DataBase，简称为DB，指的是长期保存在计算机的存储设备上，按照一定的规则将指定的数据组织起来，可以被各种应用使用的数据集合，其本质就是用于存储数据的容器，与普通的数据容器不同的地方在于：数据库依据数据结构来存储数据，正因为有数据结构的存在，导致数据库中的数据的存储比较条理化或者格式化。市面上的常见的数据库有：Oracle、SQL Server、MySQL、Redis、MongoDB等，

在本系统中主要用到了MySQL和Redis数据库。值得一提的是，这两种数据的差别决定了其不同的用途，首先在类型上MySQL数据库属于关系型数据库，然而Redis属于NoSQL，其次在数据存储位置上有非常大的差别，MySQL数据库将数据存储在硬盘上，然而Redis一般将数据存储于内存中，因此，MySQL一般用于持久化数据，偏向于存数据，然而Redis一般用于对数据进行缓存，偏向于取数据。

**2.2.4 uni-app**

uni-app是由DCloud（数字天堂）开发的可以使用[Vue.js](https://vuejs.org/)开发所有前端应用的框架，开发者编写一套代码，可运行和发布到iOS、Android、H5、以及各种小程序（微信/支付宝/百度/头条/QQ/钉钉）等多个平台，因此，uni-app的产生，极大地降低了开发者的学习成本，一套代码可以多平台转换运行。同时uni-app也引入了众多的第三方接口，如：地图、支付、导航等，极大地增强了开发效率。

**第三章 可行性分析和业务分析**

**3.1 可行性分析**

该系统的可行性分析(Feasibility Analysis)是在对该业务进行了初步分析之后，同时主要结合采用技术的优缺点、经济成本等因素，最终确定该系统的开发方案的方法。它是更深层次的思考该系统采用技术的适配性和经济成本等，从而确定其技术、经济最优的方案。可行性分析是项目开发过程中必须进行的步骤，决定了整个系统的大体方向和产品质量。本节从技术、经济、操作是哪个方面对房屋租赁系统进行了全面分析。

**3.1.1 技术可行性**

在对该系统进行了全面分析之后，用Django做后台管理，Flask前台的想法是完全合适和可行的。Django强大后台管理模块可以非常完美的解决后台管理设计繁琐的问题，且其自带的ORM完美的实现了关系型数据库和Django服务器之间的交互；其次，Flask的灵活多变性也相当方便接口的实现，而且，Flask中的sqlAlchemy插件也完美的解决了Flask服务器和数据库之间的交互问题。

**3.1.2 经济可行性**

经济上考虑的是系统相应的开发成本、维护成本与系统使用中租户和房东之间的关系。该系统开发成本较低，只需要一台云服务器以及短信服务便可以完成项目的开发和部署。另外，该系统的前台前端采用uni-app开发，可以适应更多的平台，极大地降低了开发成本。此外，系统的使用能大大增加房屋成交的机会，极大地提高房东和租客的劳动成本。

**3.1.3 操作可行性**

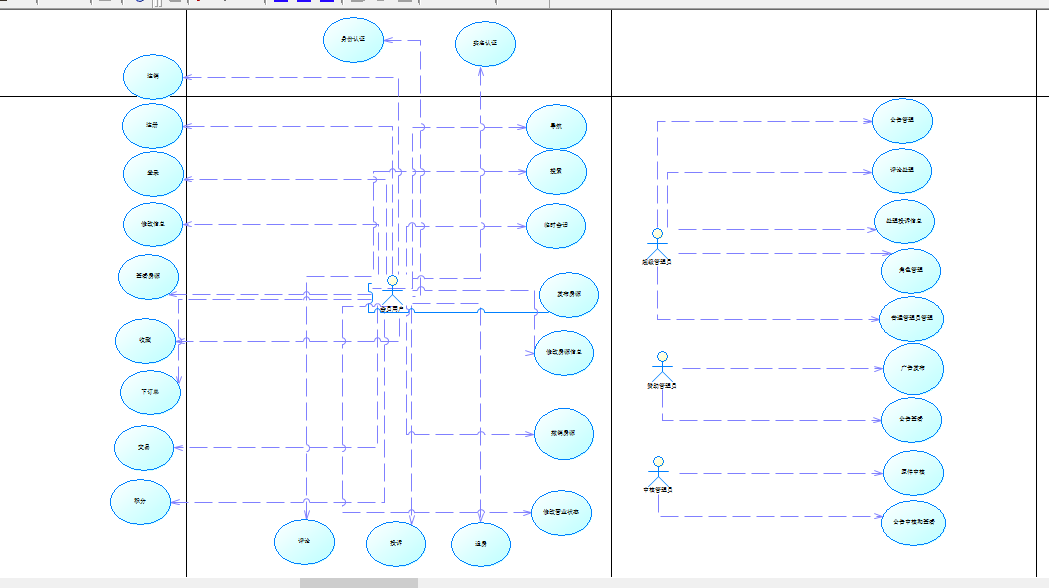
由于该系统无论前台还是后台服务，都是具有前端界面和后端服务器的，因此，本系统只需要将该后端Django和Flask服务器成功部署到云服务器中，系统管理员便可以直接通过浏览器访问指定URL就可以访问该后台管理服务，而用户便可以便可以直接安装有uni-app转换来的Apk文件便可以，从而非常方便进行各种前台操作。

**3.2 业务分析**

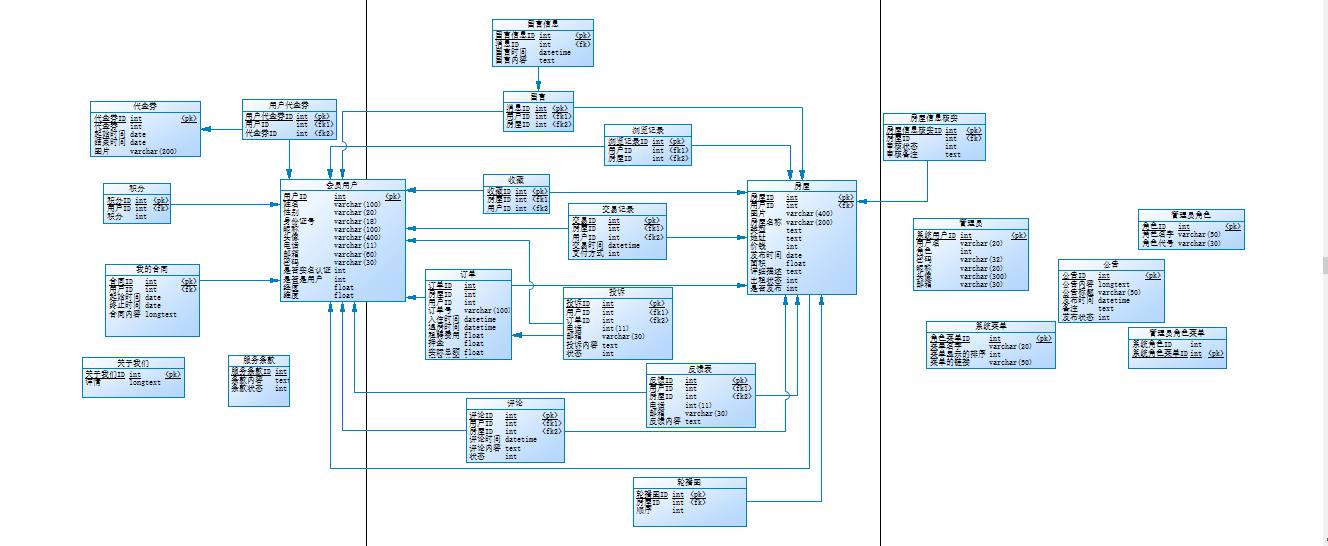
**3.2.1 用户群体**

该系统的用户群体主要分前台和后台，前台针对的主要是普通租客和房东，而后台则主要针对的是超级管理员、审核管理员和赞助管理员。

**3.2.2 业务模型图**

对该系统进行了全面的业务分析，并对该系统真的每个群体都进行了分析，并通过PowerDesigner集合分析的结果，画出了该系统的业务用例图，截图如下：

**3.2.3 模型表建立**

根据该系统的业务用例图进行分析，同样通过PowerDesier建出了物理模型图，截图如下：

**第四章 系统设计及测试**

**4.1数据库和数据表设计**

**4.1.1 数据库和数据表设计**

**4.1.2 数据表设计**

**4.1.3 数据表的逻辑关系**

**4.2系统后台设计**

**4.3系统前台设计**

**4.4系统测试**