玉林师范学院本科生毕业论文（设计）

基于B/S房屋出租管理系统的设计与实现

**Design and implementation of the house rental management system based on B/S**

|  |  |
| --- | --- |
| 学生所在学院 | 计算机科学与工程学院 |
| 专 业 | 软件工程 |
| 班 级 | 2015级152班 |
| 姓 名 | 袁迪 |
| 学 号 | 201506402243 |
|  | |
| 指导教师单位 | 计算机科学与工程学院 |
| 指导教师姓名 | 陆钊 |
| 指导教师职称 | 副教授 |

基于B/S的房屋出租管理系统的设计与实现

软件工程 2015级2班 袁迪

指导教师 陆钊

**摘要**

在中国，人口数量数以亿计，城市化建设脚步不断加快，城市流动人口也不断增加，房房屋租赁和相关管理业务也不断加大，如何高效率的提高房屋租赁业务的管理是这个行业面临的重要问题之一。本课题针对当前的这一问题，设计并实现一套基于B/S的房屋租赁管理信息系统[2]。本系统选用JavaScript开发语言，基于Visual Studio Code开发平台；后台数据库采用MongoDB数据库；前端采用Vue框架编写，后端采用Node.js框架编写;利用Ajax技术实现前后端数据交互。本设计构造了一个多功能的应用管理系统。系统主要分为前台模块和后台模块。其中前台主要实现信息资源的展示、包括房屋资源、新闻信息和用户评论等信息。后台分管理者和租赁者两个模块。它主要实现注册登录、信息发布、评论发布、和信息管理等功能。整个系统采用三层架构模式，对系统的兼容性起到了很大的作用，有利于实现系统的维护和再次升级。在管理方面区分用户类型，不同类型用户赋予不同权限。提高了系统的安全性和有效性。该系统的使用，能够实现房屋租赁业务的高效管理。

**关键词：**房屋出租管理系；Vue.js；Visual Studio Code；MVC

**Design and implementation of the house rental management system based on B/S**

Software Engineering 2015-152, Yuan Di

Supercisor Lu Zhao

**Abstracts**

There are hundreds of millions of people in China. And the pace of urbanization is accelerating. The floating population in cities is also increasing. So housing rental and related management business is also increasing. How to effectively improve the management of housing rental business is one of the important issues facing this industry. This design aims at this problem.Design and implement a web-based lease management information system. This system USES JQuery development language. Based on the Visual Studio Code development platform. The background database adopts MongoDB database. The front end is written with Vue framework. The back end is written in the node.js framework.Ajax technology is used to realize front-end and back-end data interaction. This design constructs a multifunctional application management system. It is mainly divided into the foreground module and background module. The foreground mainly realizes the display of information resources, including housing resources, news information and user comments and other information. The background is divided into two modules: administrator and renter. It mainly realizes registration and login, information publishing, comment publishing, information management and other functions. The whole system adopts the three-layer architecture pattern, which plays a great role in the compatibility of the system. And it is conducive to the maintenance and re-upgrade of the system. In terms of management, users are distinguished and different types of users are given different permissions. It really improves the security and effectiveness of the system. The use of this system can realize the efficient management of housing rental business.

**Keywords:** house rental management system,Vue.js,The Visual Studio Code, MVC

**目录**

**[1前言](#_Toc533222921)** [1](#_Toc533222921)

[1.1系统研发的背景和意义 1](#_Toc533222922)

[1.2国内外研究现状和发展前景 2](#_Toc533222923)

[1.3选题研究目标及内容 2](#_Toc533222924)

[1.3.1选题研究目标 2](#_Toc533222925)

[1.3.2选题研究内容 3](#_Toc533222926)

[1.4 小结 3](#_Toc533222927)

**[2选题所用技术](#_Toc533222928)** [3](#_Toc533222928)

[2.1 JQuery简介 3](#_Toc533222929)

[2.2 MVC设计模式 4](#_Toc533222930)

[2.3 node.js框架介绍 5](#_Toc533222931)

[2.4 Ajax技术 6](#_Toc533222932)

[2.5 Vue框架 6](#_Toc533222933)

[2.6 MongoDB数据库 7](#_Toc533222934)

**[3房屋出租管理系统的需求分析](#_Toc533222935)** [8](#_Toc533222935)

[3.1 系统总体需求 8](#_Toc533222936)

[3.3 系统租赁者需求 10](#_Toc533222937)

[3.4 系统功能需求分析 10](#_Toc533222938)

[3.5 系统数据库需求分析 11](#_Toc533222939)

**[4系统设计与实现](#_Toc533222940)** [12](#_Toc533222940)

[4.1检索功能 12](#_Toc533222941)

[4.2注册与登录功能 12](#_Toc533222942)

**[5系统实现](#_Toc533222943)** [13](#_Toc533222943)

[5.1系统功能代码 13](#_Toc533222944)

[5.1.1用户登录模块代码 13](#_Toc533222945)

[5.1.2用户注册 14](#_Toc533222946)

[5.1.3房屋管理 15](#_Toc533222947)

**[5.2系统界面实现](#_Toc533222948)** [16](#_Toc533222948)

[5.2.1系统前台界面 16](#_Toc533222949)

[5.2.2系统后台界面 17](#_Toc533222950)

[5.2.3用户登录注册界面 18](#_Toc533222951)

[5.2.4后台管理登录注册 19](#_Toc533222952)

**[5.3 系统测试](#_Toc533222953)** [20](#_Toc533222953)

[5.3.1系统测试过程 20](#_Toc533222954)

[5.3.2系统测试结果 21](#_Toc533222955)

**[致谢](#_Toc533222956)** [21](#_Toc533222956)

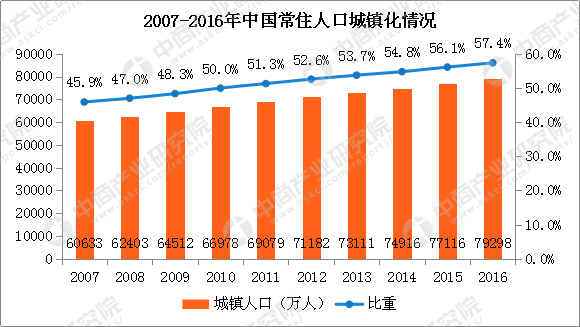
**[参考文献](#_Toc533222957)** [22](#_Toc533222957)

# 1前言

本章主要介绍课题的研究背景和研究意义，国内的相关研究现状和发展前景，以及课题的研究目标和研究内容。

## 1.1系统研发的背景和意义

改革开放40多年来，我国不断推动中国城镇化的改革，以及放松对流动人口的管理政策，城市的流动人口急剧增加。国务院第六次全国人口普查全国总人口为13.7054亿人，其中城镇常驻的人口就有66557万人，占总人口的49.68%，同2000年人口普查相比，中国城镇人口比重上升13.46个百分点[3]。据最新中国人口2018总人数口统计， 2018中国人口数量13.9008亿，城镇人口占总人口比重为58.52%，同2000年人口普查相比增长了22.48个百分点，流动人口高达2.44亿人。



**图1.1 2007-2016年中国常住人口城镇化情况**

**Fig1.1 Population urbanization in China in 2007-2016**

流动人口的大规模地在城乡与区域间流动，一方面可以有效促进我国劳动力资源的配置，为经济发展增添新的活力，一方面又随之而来地增添了一些新的问题。流动人口地性质决定了他们的居住方式都是临时性的，他们的居住资源主要包括租住、寄住、单位提供或者自己拥有。然而在高昂的房地产价格面前，绝大多数人对买房常常望而却步，因此，求租房屋和房屋出租都面临着比较大的需求[4]。但是传统的房屋管理系统存在效率低、管理不全面、缺乏宏观数据的分析与掌握等诸多问题，管理体系的改革成为房屋租赁业务的势在必行的一项任务。如今，飞速发展的互联网技术给房屋租赁行业一个全新的管理方式，房屋租赁行业与互联网结合的发展形态已经势不可挡。利用网络平台对房产公司的管理业务以进行信息化、流程化的系统管理，可以提供优质的房产中介服务，最大化提高管理效率。

## 1.2国内外研究现状和发展前景

在欧美等发达国家，信息化技术相对成熟，房地产行业也是发展久远，以计算机网络技术进行管理的已经成为主要管理手段。信息化的管理方式可以提高管理效率，大大降低管理成本。因此，欧美等发达国家较早就拥有了比较完善的房产中介管理系统。如美国的基于计算机网络的MLS（Multiple Listing System）系统就赢得了美国民众的信任[5]。Hartz、Raveis、Fraser及Doyle等对房产中介系统的设计实现和信息化管理都做过较为深入的研究[6]。

反观国内现状，受我国历史国情影响，我国的房产中介业务起步落后于发达国家很多，管理方式和相关制度发展较晚。房产中介业务发展速度前期都是缓缓而行。直至21世纪，我国计算机技术和信息化技术的不断普及，房产中介管理系统的研发得到了关注[7]。也有相关人员，通过与国际知名的技术企业合作，为我国房产中介管理的信息化做了较为深入的研究。

不可否认，进来国内互联网行业的可谓发展迅速，如今拥有并发布房屋租赁信息的网站已经是日新月异层出不穷，发展至今已经有不少较为成熟的管理系统，如房介宝、易房大师和房管家等，发展也确实不慢。但由于我国地广物博，人口众多的国情，以及不断深入的城市化建设，我国未来对于房地产信息化建设的需求将会更高。

## 1.3选题研究目标及内容

### 1.3.1选题研究目标

本课题的选题的目标是以提供更好、更快、的更优质的管理服务为宗旨，设计并实现一个基于web技术的，能够在移动端服务的，更科学化、系统化、规范化的房屋租赁管理系统。

研究内容方面，主要是对国内外房产中介管理系统的发展背景和现状进行分析，对课题的研究意义进行了分析，对系统的功能和结构的做了以下分析。

1. 分析了解现有房产中介管理系统的发展历史和现状、寻找这些系统更新或者淘汰的原因，了解房产中介管理系统的市场等。
2. 研究房屋租赁系统的信息化的意义，研究基于web的房屋租赁管理系统相对于其他的管理系统的优劣。
3. 对本设计的结构和功能进行分析，对房源采集，租房管理，办公类型，管理类型等业务进行分析。对技术架构，功能架构进行分析。

### 1.3.2选题研究内容

本课题的选题的目标是以提供更好、更快、的更优质的管理服务为宗旨，设计并实现一个基于web技术的，能够在移动端服务的，更科学化、系统化、规范化的房屋租赁管理系统。

研究内容方面，主要是对国内外房产中介管理系统的发展背景和现状进行分析，对课题的研究意义进行了分析，对系统的功能和结构的分析。

1. 分析了解现有房产中介管理系统的发展历史和现状、寻找这些系统更新或者淘汰的原因，了解房产中介管理系统的市场等。
2. 研究房屋租赁系统的信息化的意义，研究基于web的房屋租赁管理系统相对于其他的管理系统的优劣。
3. 对本设计的结构和功能进行分析，对房源采集，租房管理，办公类型，管理类型等业务进行分析。对技术架构，功能架构进行分析。

## 1.4 小结

本章主要讲述了房产中介管理系统的发展背景和研究意义，就国内外的房产中介管理系统的发展现状和前景进行了分析，同时也总结了本课题的主要研究内容和研究目标。

# 2系统所用技术

## 2.1 JQuery简介

JQuery是美国工程师John Resig2006年研究发布的一个优秀的JavaScript框架。它封装JavaScript常用的功能代码，提供简便、快速的JavaScript设计模式，能够支持优化HTML文档操作、事件处理、动画设计和Ajax数据交互[8]。John Resig在2005年开始提出改进优化当时的Prototype的"Behaviour"库等，他提倡写更少的代码，做更多的事情。于是他在blog上发表了自己的想法，逐渐地越来越多的IT爱好者得知，随后也有越来越多开发者的加入，加入越来越多的技术，功能也随之强大。而今JQuery拥有了JavaScript、CSS、DOM操作和Ajax技术于一体的强大功能的技术，可以用最少的代码， 用来完成更多复杂而困难的功能，从而得到了许多开发者的青睐[9]。JQuery可以大致分为三个模块：入口模块、底层支持模块和功能模块。分模块模式方便测试模块提供针对不同浏览器功能和bug的测试，其他模块则基于这些测试结果来解决浏览器之间的兼容性问题与功能模块之间的嵌套。入口模块在进入时开放一个接口，用于下载引入所需的模块，进入JQuery并可以执行jq相应的代码模块。底层支持模块用于增强对回调函数的处理，支持添加，删除，修改，触发，锁定，禁用等回调函数功能。功能模块中，进行事件邦定、事件处理、事件响应、DOM操作等，是JQuery执行代码的核心模块。JQuery在语法上兼容IE 6.0+，FF 2+，Safari 3.0+，Opera 9.0+，Chrome等浏览器，拥有比原生js更多的优点，拥有大量有趣的扩展和出色的社区[10]。几千种丰富多彩的插件，能够简化操作，节点操作简单，链式调用，运用灵活。

## 2.2 MVC设计模式

MVC设计模式是[Java EE](https://baike.so.com/doc/998397-1055441.html" \t "_blank)平台的下一种典范软件设计模式，全称Model View Controller，即模型-视图-控制的英文缩写。Model(模型)是指用于处理应用系统中数据逻辑的部分，主要对负责数据库中的数据存取操作。系统中引用任何程序对数据库进行的操作都属于Model部分，也称为业务逻辑层。View(视图)是指将业务逻辑层处理的数据进行显示的部分，是用户与应用系统的交互界面，也称为表示层。通常视图是依据模型的结构功能来创建。Controller (控制器)是系统中连接用户交互与业务逻辑层的部分。Controller主要负责获取用户输入数据，读取视图数据，和向模型发送数据。MVC设计模式适用于大型可拓展的Web系统的开发,就是将应用程序的业务逻辑、交互界面进行分离，使得各层次之间按规则交换数据，调用对应层次的方法，防止数据跨层调用，导致数据交换杂乱。开发人员通过软件的详细设计将软件的所有功能罗列定义分析梳理前段，后台管理需要使用的功能模块将交互界面涉及的功能调用整理到控制器，将数据传输到Controller，控制器作为中间交换器将数据传输到Model，业务逻辑层调用相应功能对数据库进行操作，获得的数据结果集回传到Model再回传到控制器，控制器将数据发送到View显示数据。通过MVC设计模式很好地诠释了系统软件的整个运作流程，清晰地阐述数据在整个系统中的流动情况。常见的MVC设计模式中运用的框架有Struct、Spring、Zend FrameWord等，MVC具有开发生命周期短、低耦合、复用性高、可维护性高等优秀之处[11]。这对于系统开发来说十分有利，因此本文的校园招聘系统采用MVC设计模式为开发带来更多的便捷之处，在学习和开发过程中能够深入掌握MVC精髓并能灵活运用。

## 2.3 node.js框架介绍

本设计使用node.js技术编写后台管理数据获取及模块。Node.js是一个基Chrome V8引擎的 JavaScript运行环境。Node.js使用了一个事件驱动、非阻塞式I/O的模型，使其轻量又高效，Node.js的包管理器npm，是全球最大的开源库生态系统[12]。Nodejs是一个框架，也是一个库，更是一门语言，node.js是基于node平台运行的一门技术语言。node.js使用Module根据功能划分出许多不同功能的模块，每一个Node.js的每一个类库都包含了非常丰富的函数，减少了许多繁琐的js代码，不用进行单个模块封装，比如http模块就包含了和http功能相关的很多函数，可以帮助开发者很容易地对比如http,tcp/ip等进行操作，还可以很容易的创建http和tcp/ip的服务器。Node平台需要开启一个服务器，本设计使用小型静态服务器，node.js平台存放前端html页面需要开启静态文件夹存放，css、less文件也需要存放在静态文件夹内。引入的类库相当于存放变量，node.js会在node\_modules文件夹下查改该查找是否存在该模块，如果查找不到时，再到全局模块缓存中查找，所谓全局缓存模块，就是自定义模块，程序编写自己需要但是node中不带有该模块时，便可自己构建一个含有所需功能模块。在模块引入时，需要将该模块暴露抛出，exports default 模块名称，暴露的目的是让主模块连接到该模块的内容能够找到并且引用该模块的功能。

## 2.4 Ajax技术

Ajax技术用于处理前台与后台的数据交互，是一种交互式的网页开发技术，是由Outlook Web Access小组编写允许客户端脚本发送http请求，建立动态网站的一大重要技术[13]。使用Ajax发送请求有许多限制，必须包含请求顺序，请求的优先级，响应时间，响应到一定时间时返回响应超时，请求错误时进行错误处理，请求成功回调进行DOM操作。使用Ajax建立与后台的动态交互，需要用到参数控制传输数据与交互地址，Ajax的参数共有21个，主要参数与作用如下：

* + 1. URL：表示交互接口，数据请求的后台接口；
    2. data：代表从前台传输到后台的限制参数，也为修改数据的参数，要求参数类型必须是Object类型或者String类型，数据传输到后台需要处理为接口渲染；
    3. Type:数据请求的方式，只有GET、POST、PUT与DELETE方式，默认为GET方式，常用的为GET与POST两种，POST请求是比较保险的请求方式，不会暴露出接口的参数，GET方式会将参数拼成地址拼接到URL地址后面直接请求数据，这种请求方式相对比较快捷；
    4. Async：意味是否异步，此参数用于控制参数使用异步请求或者同步请求，默认为异步请求，参数值为Boolean类型，true为异步，false为同步，如果设置为同步请求时就会锁定浏览器，用户的操作必须等到请求结束之后才能执行；
    5. Success：意为成功的回调，必须是function函数类型，回调函数里面有两个参数，第一个是有服务器返回的根据dataType处理后的数据，第二个为描述返回数据的状态的字符串；
    6. Error：意为错误的回调，必须为function函数类型，该回调有三个参数，即XMLHttpRequest对象、错误信息、捕获的错误对象，捕获的错误对象是一个可选参数，三个参数共同返回错误的信息。

## 2.5 Vue框架

设计主要用Vue框架编写前台页面。Vue是一个轻巧、高性能、可组件化的MVVM库，同时拥有非常容易上手的API的Vue.js是一个轻巧、高性能、可组件化的MVVM库，同时拥有非常容易上手的API，易于用户构建界面的循序渐进的框架。Vue框架的使用主要是参照文档api构建，npm文档编写功能模块。npm模块的使用分为在终端下载插件，引入使用插件，按文档使用功能模块。Vue使用的最大重难点与亮点为需要使用生命周期函数（钩子函数）控制数据流，每个 Vue 实例在被创建时都要经过一系列的初始化过程——例如，需要设置数据监听、编译模板、将实例挂载到 DOM 并在数据变化时更新 DOM 等。同时在这个过程中也会运行一些叫做生命周期钩子的函数，这给了用户在不同阶段添加自己的代码的机会。生命周期与用法具体分为以下几个过程：

1. 初始化init；
2. 实例创建之前，调用beforeCreate方法；
3. 创建实例created；
4. 判断是否有el（如果没有调用vm.$mount函数，实例中的el）；
5. 编译一个模板到rander函数里面 ,产生一个虚拟dom beforemount；
6. 编译完成，创建一个vm.$el替换el；
7. mouted挂载完成，加入dom丛；
8. 更改数据update，updaed， mounted；
9. 销毁、销毁完成 deforeDistroy、destroyed。

## 2.6 MongoDB数据库

MongoDB数据库既不是关系型数据库，也不是非关系型数据库，是介于关系型数据库与非关系型数据库之间的特殊型数据库，它是非数据型关系数据库功能最多，应用层次最为广泛的数据库，又像关系型数据库，支持json、csv、SQLDatabase等多种类型数据，因此也能存储为多种类型的数据，它的语法类似于面向对象查询语句，几乎包含关系数据库表单查询的所有功能，而且还支持建立索引值。

MongoDB数据库是面向集合数据库，所谓的集合，就是数据库里面的数据表，每个集合在数据库中都有唯一的一个标识名称，就是唯一的主键，集合可以包含无数文档，集合与关系型数据库的表也有所不同，mongoDB中不需要定义任何的schema对象，数据库本身能提供高速缓存算法，能够快速识别数据库内的大量数据，加大运行性能,数据获取时比普通关系数据库要快很多，mongoDB数据库成为了越来越多的程序员的最优选择。mongoDB数据库有自由模式（schema-free），在数据库中字段定义时可以省去字段类型的定义，使字段不受类型限制，增加了数据库的使用范围。我们可以不用拘束数据库的任何结构定义，能将不同的数据库类型，多种数据结构存储在数据库中。数据库文档存储以键值对的形式存储，被存储为键-值对的形式，键可以唯一标识文档，类型为字符串类型，值可以是任何一种类型数据。

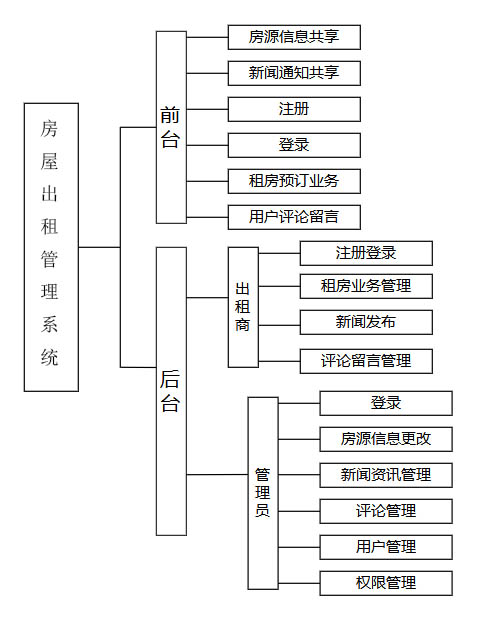
# 3统的需求分析

房屋出租管理系统分析的任务就是怎么更好更方便全面地管理楼房出租的数据与用户租房的需求，主要解决的问题是“做什么”的问题

## 3.1 系统总体需求

对于现代化管理信息化平台，几乎所有的行业都进入了信息化管理时代，使用的人越多，造成的弊端也越多。借鉴对比网络平台上已发布的平台来说，最重要的是要确保用户的真实信息，这样才能有效地保护用户的合法权益。在当今时代下，许多网站缺少监管力度，在发布要出租的楼房时缺少相关的权益合同，也有未能保证发布人的信息是否为真实，当前的形式下也会滋生越来越多坏因素，譬如利用网站的弊端骗取用户的财产，或者盗取用户个人信息做不法行为。就以上种种问题，进行了本设计，主要解决用户的信息安全问题与租房路径的可靠性，完成以下功能：

1. 使用便捷：在任何有网络的条件下都能够快速访问，管理员，租赁者，用户都能访问前端网站，租赁者能够登录后台查看并且修改只属于自己的数据，管理员能够登录后台查看所有租赁信息并且能够修改部分信息，用户不能登录后台管理系统。
2. 设计简洁：前台页面设计有地区搜索，热门楼盘列表，后台直接显示相应用户的数据，可视化表格，查看方便，操作简单，后台有多种颜色设置，可以根据喜好搭配皮肤颜色。
3. 数据真实：房屋租赁者在展示自己的楼房前要签订相关协议，保证租赁者的信息真实性，并且需管理员监督，由管理员审核通过后才能将信息发布，用户租赁时需要租赁者审核之后并且支付相应租金才能与租赁者签订租房协议
4. 安全可靠：房屋出租的交互方面有双层保护，首先由租赁者监管，再由管理员审核通过后方能完成所有信息的交互。



**图3.1 功能框架分析图**

**Fig3.1 Diagram of functional framework analysis**

**3.2 系统用户需求**

系统用户有管理员，租赁者，用户，游客四个身份登录，游客能进入前端浏览网站信息，楼盘信息，要租赁楼房时需要注册成为用户方能。用户需求都集中在前端，用户在前端注册，登录，查看已发布的楼盘。用户注册需要使用邮箱注册，注册信息必须包含姓名，性别，邮箱，身份证号，电话号码字段。查看已租过该房的用户的评论与租赁者的日记，楼盘查看可以根据楼盘出租热度、地区等方式查看。预定楼房需要在有效时间内支付租金并且租赁者返回租房信息后签订相关协议完成租赁程序。

## 3.3 系统租赁者需求

租赁者注册时也要填入姓名，性别，邮箱，身份证号，电话号码字段，在注册之前要先签订协议，如有违反网站的相关规定，该租赁者需要承担相应权重的惩罚或者承担相应的法律责任。租赁者在出租房屋时需要经过管理员的审核，并返回信息后判断是否能够发布该楼房。

## 3.4 系统功能需求分析

对系统功能需求进行细化分析。本设计名为基于web的房屋出租管理系统，顾名思义，就是利用web技术来提供方便快捷的房屋租赁服务及其相关管理服务功能的系统，因此系统上的功能应有如下几点：

1. 房屋信息发布需求分析

注册认证的租赁者能够在平台上发布楼房或者相关的新闻资讯信息，该系统支持租赁者通过平台发布楼房租赁的相关活动信息，信息发布需要插入标题与内容信息两个字段，在发布之前网站先判断登录用户的权重是否能够发布资讯新闻，权重判断出登录用户为管理员与租赁者之前客户端返回发布信息的按钮界面。

1. 意见反馈需求分析

游客注册后使用登录名注册，可对楼房信息进行查看，如果某个用户发现该网站有不符合规定或者不适合用户体验，又或者租赁者发布楼盘信息，资讯信息有误可以将错误信息反馈给网站的管理员，管理员能够对其进行相应的修改，管理员也能够对该用户进行相应的奖励。

1. 房屋发布需求分析

该系统已便利为主要目的进行设计，该系统方便于所有有闲置空房屋的人注册该系统，阅读该系统的相关条例规定，都可以发布房屋，但是在发布房屋之后需要通过管理员的审阅，管理员判断是否允许该房屋在网站上发布，发布之后租赁者与管理员均能查看该房屋的租赁状态。

1. 后台管理需求分析

本系统只允许拥有管理员权限的工作者进入后台管理系统，包括网站的管理员与发布房屋的租赁者，不一样的是根据两者的权限不一样，登录进去后拥有的功能也不一样，在数据库设置权限，管理员的权限约定为0，租赁者的权限约定为1，根据权限判定登录着是管理员还是租赁者，再对其拥有的功能进行相应的制约，严格的权限管理能够很好地拦截非管理员用户通过非正常途径进入后台。管理员的后台不编写注册功能，很好地控制网站的管理。租赁者注册后台时需要使用邮箱进行验证信息，姓名、邮箱、电话号码、身份证号等使用正则验证，保证填入的格式是正确的,提高信息的真实率。租赁者和管理者的具体管理权限如下：

管理员权限：可查看管理该网站的新闻资讯，评论信息，房屋信息以及用户信息管理，管理员对房屋信息的管理只能是管理其是否允许发布，不能修改房屋的具体信息与评论信息；能修改用户信息与新闻资讯信息。

租赁者权限：租赁者注册后台管理系统拥有查看与修改自己发布的房屋信息，自己发布的新闻资讯与用户对自己发布房屋的评论，不能查看与管理后台管理系统的用户管理与权限管理等功能。

## 3.5 系统数据库需求分析

数据库是用来存储数据信息的一大重要技术，用户在注册网站的时候都会存储身份证号码、电话号码等重要信息，因此数据加密是一项重要的技术，因此数据库的加密技术要安全可靠，保证数据的安全性和完整性，还要确保数据存储速度以及读取速度，为使系统功能能够准确无误、迅速地对数据进行操作，设计一个合理的数据结构是必不可少的，根据系统不同用户、信息模块的需求，将数据库结构设计如下：

1. 用户表：给用户与租赁者提供账号，属性包括用户ID、姓名、性别，邮箱、密码、电话号码、身份证号码、权限、注册时间。
2. 新闻资讯表：存储网站的新闻资讯，属性包括新闻ID标题、小标题、新闻类型、新闻内容、发布时间、发布人。
3. 房屋表：存储发布的房屋，属性包括房屋ID、房屋名称、房屋地址、房屋图片、占地面积、房屋类型、价格、物业公司、租赁者、租赁状态、发布时间。
4. 评论表：存储房屋的评论信息，属性包括评论ID、用户名、内容、评论屋主姓名、楼房名称、评论时间。
5. 意见反馈表：用户向管理员反馈信息的表，属性包括意见ID、用户名、反馈内容，发表时间。
6. 删除会员表：存储已经删除的会员，属性包括用户ID、姓名、性别，邮箱、密码、电话号码、身份证号码、权限、注册时间、删除时间。
7. 房屋订购表：存储用户订购房东没有处理的数据，属性包括房屋id，房屋名称，地址，面积，房屋类型，价格，物业公司，租赁者，订购日期。

# 4系统设计

## 4.1检索功能

前台：前台首页中的系统中所有的用户都能对房屋进行分类检索，该模块功能中提供关键字搜索、全称搜索、分地区搜索等方法对信息进行全方位查询，通过Ajax技术传递参数限制后台返回数据，实现无刷新界面的数据显示，用户可以选择自己想要的内容。

后台：后台管理所有页面都有检索功能，该包含关键字搜索、全称搜索、分地区搜索等方法对信息进行全方位查询，通过Ajax技术传递参数限制后台返回数据，实现无刷新界面的数据显示，用户可以选择自己想要的内容。检索功能的主要代码如下：

router.get('/getUserData2',(req,res)=>{

yu=req.query.yu \* 1;

target=req.query.targetPage

content=req.query.content

style=req.query.style

let se={}

let adname=req.query.name

console.log(adname,79);

houseModel.find({

$or:[

{"housename":{$regex:content}},

{"addr":{$regex:content}},

{"area":{$regex:content}},

{"property":{$regex:content}},

{"lander":{$regex:content}},

{"price":{$regex:content}},

{"status":{$regex:content}},

],

"lander":adname

}).sort({'\_id':style})

.then((data)=>{

console.log(data,95);

sum=data.length

console.log(sum,49);

return houseModel.find({$or:[

{"housename":{$regex:content}},

{"addr":{$regex:content}},

{"area":{$regex:content}},

{"property":{$regex:content}},

{"lander":{$regex:content}},

{"price":{$regex:content}},

{$and:[{"status":land}]}

],

"lander":adname

}).sort({'\_id':style}).limit(yu).skip((target-1)\*yu)

})

.then((data2)=>{

// console.log(sum,40)

se={sum,data2}

res.send(se)

})

.catch((err)=>{

res.send(err)

})

})

## 4.2注册与登录功能

前台注册，游客对网站进行浏览，如果想要评论或者租赁房屋必须注册系统账号，注册账号需要使用邮箱进行验证信息，必须包含用户的姓名、邮箱、电话号码、身份证号等信息，保证信息的准确性。后台注册，如果想发布自己的闲置房屋，就必须注册后台系统发布才能进行房屋的发布工作，也必须包含租赁者的姓名、邮箱、电话号码、身份证号等信息。

注册成功后进入系统登陆界面，用户输入账号密码提交，向服务器请求账户数据，匹配成功允许登录，如果登录账户不正确，会显示“该账号不存在”，不予以登录，如果密码匹配错误，会显示“密码错误”，不予以登录。

## 4.3数据库结构设计

在实现房屋出租管理系统之前，首先要做的工作就是对整个系统，进行数据库设计及创建；数据库设计的主要过程贯穿整个系统开发过程中，包括需求分析、概念结构设计、逻辑结构设计、物理设计、数据库实施和维护等几个主要阶段[13]。以下是几个业务核心数据表结构的设计及实现。用户数据表如表4.1所示：

**表4.1用户数据表（users）**

**Table 4.1 user data table (users)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 用户ID |
| name | string | 否 | 否 | 用户名 |
| sex | string | 否 | 否 | 性别 |
| uname | string | 否 | 否 | 邮箱 |
| pass | string | 否 | 否 | 密码 |
| phone | NumberInt | 否 | 否 | 电话 |
| Cart | NumberLong | 否 | 否 | 身份证号码 |
| power | NumberInt | 否 | 否 | 权限 |
| date | string | 否 | 否 | 注册日期 |

删除用户表如下：

**表4.2删除用户数据表（delusers）**

**Table 4.2 deleted user data table (delusers)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 删除的用户ID |
| name | string | 否 | 否 | 删除的用户名 |
| sex | string | 否 | 否 | 性别 |
| uname | string | 否 | 否 | 邮箱 |
| pass | string | 否 | 否 | 密码 |
| phone | NumberInt | 否 | 否 | 电话 |
| Cart | NumberLong | 否 | 否 | 身份证号码 |
| power | NumberInt | 否 | 否 | 权限 |
| date | string | 否 | 否 | 删除日期 |

房屋信息表如下：

**表4.3房屋信息表（houses）**

**Table 4.3 housing information table (houses)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 房屋ID |
| housename | string | 否 | 否 | 房屋名称 |
| addr | string | 否 | 否 | 房屋地址 |
| photo | string | 否 | 否 | 封面图片 |
| area | string | 否 | 否 | 占地面积 |
| housestyle | string | 否 | 否 | 房屋类型 |
| property | string | 否 | 否 | 所属业务公司 |
| lander | string | 否 | 否 | 租赁者 |
| status | string | 否 | 否 | 租赁状态 |
| puttime | string | 否 | 否 | 发布时间 |

预定房屋信息表如下：

**表4.4房屋信息表（orderhouses）**

**Table 4.4 reservation housing information table (orderhouses)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 预定房屋ID |
| housename | string | 否 | 否 | 预定房屋名称 |
| addr | string | 否 | 否 | 预定房屋地址 |
| photo | string | 否 | 否 | 封面图片 |
| area | string | 否 | 否 | 占地面积 |
| housestyle | string | 否 | 否 | 房屋类型 |
| property | string | 否 | 否 | 所属业务公司 |
| lander | string | 否 | 否 | 租赁者 |
| date | string | 否 | 否 | 预定时间 |

新闻资讯信息表如下：

**表4.5新闻资讯信息表（goods）**

**Table 4.5 news information table (goods)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 新闻ID |
| title | string | 否 | 否 | 新闻标题 |
| Ltitle | string | 否 | 否 | 新闻小标题 |
| style | string | 否 | 是 | 房屋类型 |
| main | string | 否 | 否 | 资讯内容 |
| img | string | 否 | 是 | 图片 |
| data | string | 否 | 否 | 发布日期 |
| adname | string | 否 | 否 | 发布者 |

用户评论表如下：

**表4.6用户评论表（discusses）**

**Table 4.6 user comment table (discusses)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 评论用户ID |
| username | string | 否 | 否 | 评论用户名 |
| content | string | 否 | 否 | 评论内容 |
| name | string | 否 | 是 | 所属租赁者名 |
| housename | string | 否 | 是 | 所属房屋名 |
| distime | string | 否 | 否 | 评论时间 |

用户意见反馈表如下：

**表4.7意见反馈表（opinions）**

**Table 4.7 feedback table (opinions)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为主键 | 是否为null | 属性描述 |
| \_id | string | 是 | 否 | 评论用户ID |
| username | string | 否 | 否 | 反馈用户名 |
| opincontent | string | 否 | 否 | 反馈内容 |
| pushtime | string | 否 | 否 | 反馈时间 |

# 5系统实现

## 5.1系统功能代码

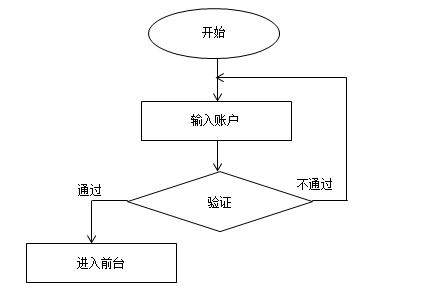
系统开发前按照系统需求分析、总体设计和详细设计认真制定开发计划，根据模块功能的流程设计图对系统方法合理定义，用简单易懂的代码设计功能，系统中很多功能模块代码可以复用，减小了本次系统开发的复杂度，通过网络、资料、数据的学习，为每个功能编写规范的文字编码，从中学习优秀的编码风格习惯。

### 5.1.1用户登录模块代码

用户登录模块功能，首先判断输入的用户名与密码是否为空，若为空，则弹出错误提示框并清空输入框，若不为空，则通过axios将输入的数据传输到后台与数据库数据进行匹配，匹配成功则登录成功，通过Vuex进行状态管理，并用localStorage在页面存储用户名并显示出已登录的用户名。



**图5.1用户登录代码图**

**Fig5.1 diagram of user login code**

### **图5.2用户登录流程图**

### **Fig5.2 flow diagram of user login**

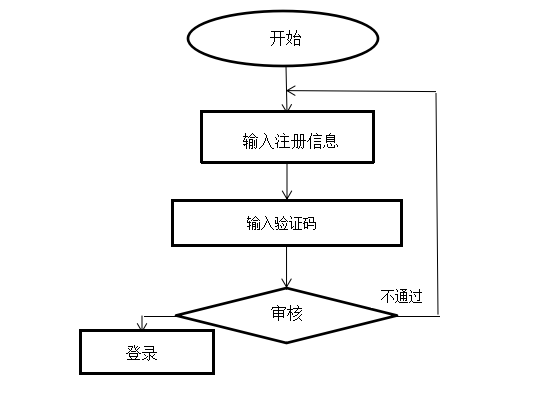
### 5.1.2用户注册

图5.3为用户注册功能代码，注册通过Ajax的post方法传输数据，post方法能将数据进行编码，加密传输，加强了用户信息的安全性。post传输方式会通过路由中的req接收，再通过模型匹配数据库，模型查询数据库返回的数据类型为promise类型，可以通过.then与.catch捕获信息，判断匹配后通过res将数据返回到前台。图5.4为用户注册流程图，描述用户注册的过程。



**图5.3 用户注册代码图**

**Fig5.3 diagram of user reg code**



### **图5.4用户登录流程图**

**Fig5.4 flow diagram of user registration**

### 5.1.3房屋管理

图5.3和图5.4为房屋信息编辑代码，主要为对房屋的修改与删除，修改信息的通过sessionStorage在客户端存储需要修改的一项数据的ID，通过该ID值使用Ajax技术结合post方法，连接相应的后台接口，将在后台查找到该数据并渲染到用来修改数据页面，修改后再次通过Ajax数据交互更新该数据信息。删除数据也是使用sessionStorage保存的ID通过后台路由删除数据库内相应的数据信息。



**图5.5房屋信息编辑代码图**

**Fig5.5 diagram of Housing information editing code**



**图5.6房屋删除代码图**

**Fig5.6 diagram of housing deleting code**

## 5.2系统界面实现

本设计参考了许多同类型的网站，研究其界面设计与色彩的适配，本设计前台已蓝色为主颜色色调，缓和的色调让用户更加愿意花时间去浏览网站，页面简洁大方，进入便能看到推荐的房屋信息，操作简单。

后台的设计主要为灰色的色调，灰色与黑色结合，点明后台管理的庄严，体现真实，品质保证。

### 5.2.1系统前台界面

系统的前台主要分为：头部、录播图、租房列表、新闻资讯、楼盘介绍与出租房屋列表。头部有细分为标题、选择框与跳转登录按钮。标题处为无论到哪个页面下，点击标题都能回到首页；选择框中可以选择北京、深圳等九个地区，可以再这九个地区内选择你要租房的地区。跳转登录按钮使用Vue下的router-link路由跳转技术实现页面的跳转。轮播图主要是运用Vue下的Mint UI框架实现轮播效果。安心合租模块为热门楼盘列表，为用户提供热门的房源信息。

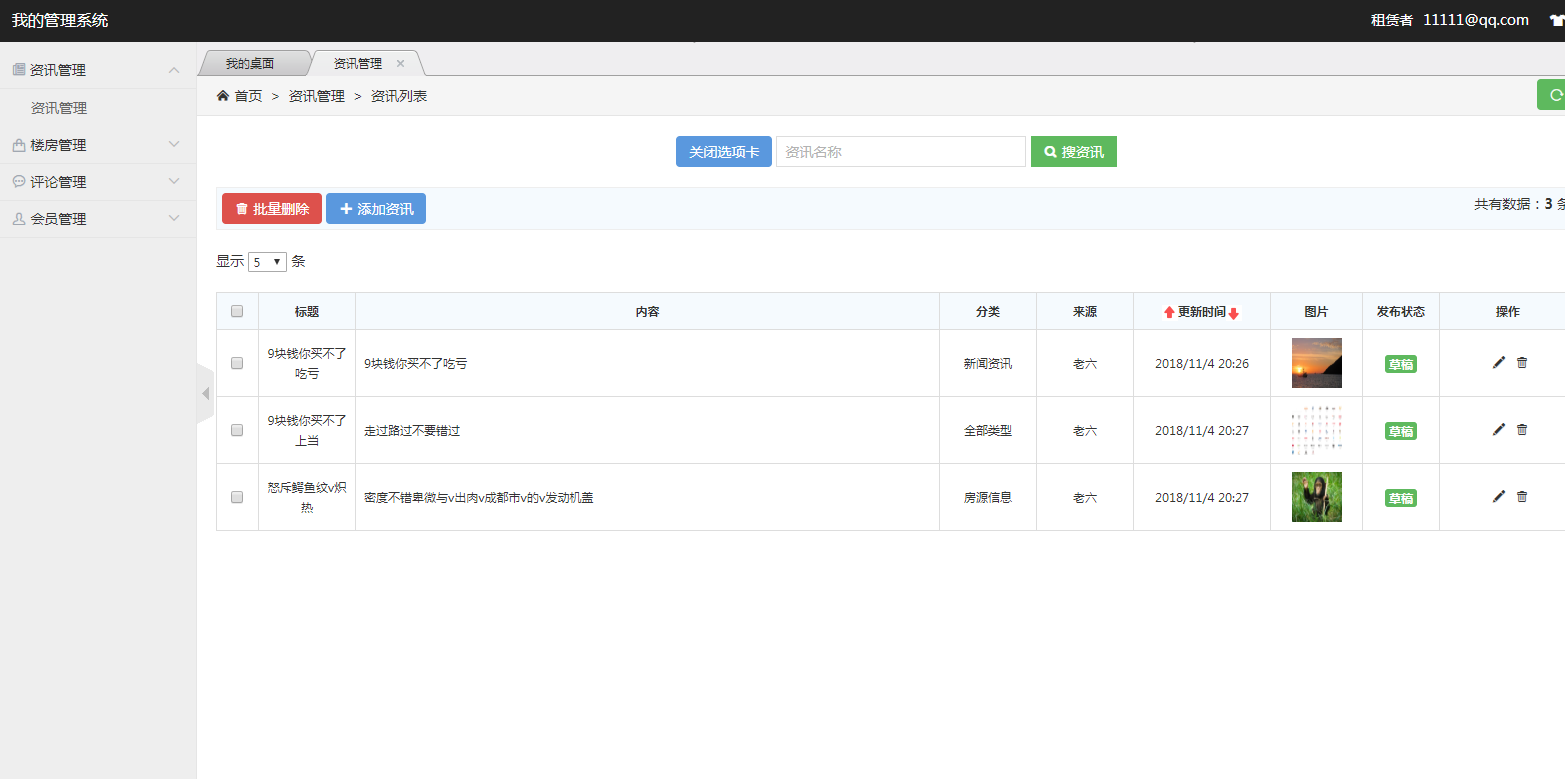


**图5.7系统前台界面图**

**Fig5.7 diagram of system foreground interface**

### 5.2.2系统后台界面

系统的后台设计中包含资讯管理、楼房管理、评论管理、会员管理四大模块。突出鲜明的模块化管理，便于管理。资讯管理包含管理对资讯的增加、删除、修改等操作，使用JQuery编写DOM操作，代码简单易懂。动态生成分页器，当咨询条数小于当前显示页面的条数时，分页器就会隐藏，反之则会显示。通过Ajax技术传递数据到指定接口，后台接口返回数据给页面进行数据渲染。后台接口使用路由与路由模型配合，路由模型通过require连接引入数据库，获取数据库数据，路由实现后台的数据操作，实现前台所需要的查询、添加、删除、修改等操作。



**图5.8 系统后台界面图**

**Fig5.8 diagram of System background boundary interface**

### 5.2.3用户登录注册界面

用户的登录页面设计较为简洁，大方，页面的头部为图片与回到主页的按钮，回到主页按钮目的是为了给用户放弃登录回到主页面去，按钮使用Font Awesome字体图标库中的字体图标。紧接着的是登录的登录名与密码的输入框，输入的信息使用正则发判断输入的数据格式是否符合网站的要求，当点击登录时，数据便传输到后台与数据库数据比对用户名与密码是否匹配，匹配成功则予以登录，反之则不予以登录。其中还有注册的文字型按钮，为没有账号的用户提供注册的页面，注册需要通过邮箱获得验证码，匹配验证码注册，防止电脑的恶意入侵造成的随意注册账号。

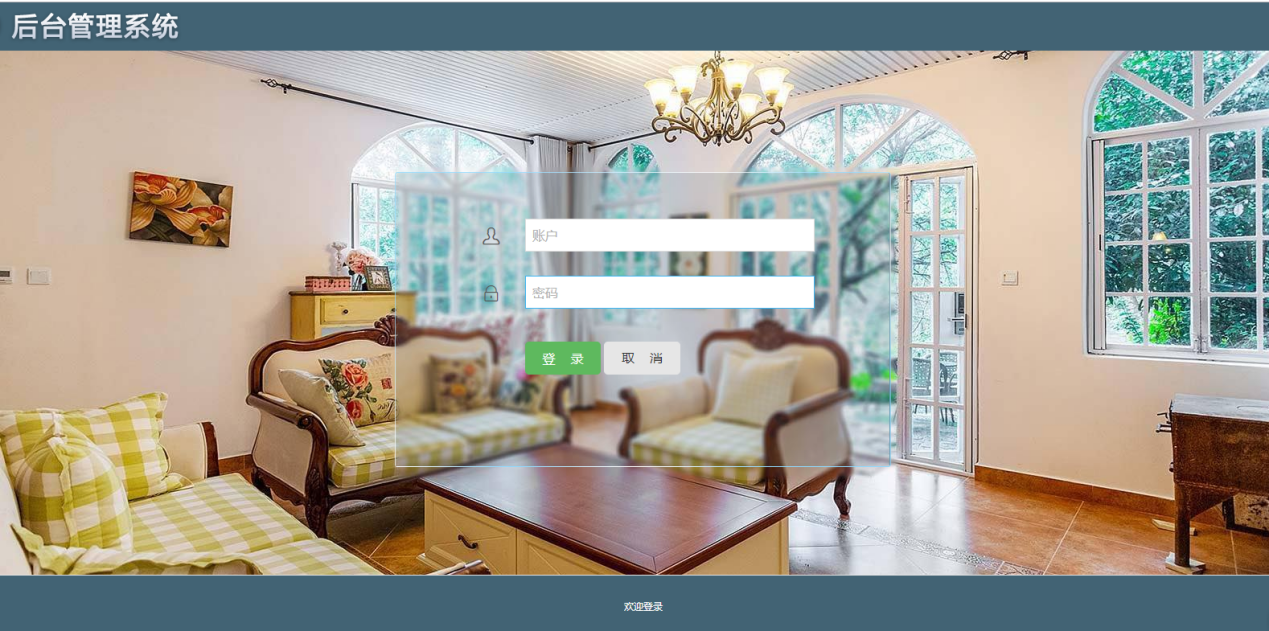


**图5.9用户登录界面**

**Fig5.9 diagram of user login interface**

### 5.2.4后台管理登录注册

后台的界面设计为华丽的房屋背景，重正面相应系统的作用。后台的登录注册于前台的登录注册相似后台的登录注册相比较于前台的登录注册区别于后台的部分分为管理员与租赁者两个身份的。通过给数据库的User表中添加权限字段，管理员的权限规定为0，租赁者的权限规定为1，用户的权限规定为2，判断权限为2的时候不予以登录，管理员与租赁者匹配相应的权限或的相应的功能。



**图5.10用户登录界面**

**Fig5.10 diagram of back-stage management login interface**

## 5.3 系统测试

系统测试是在真实系统工作环境下或系统仿真环境下检验完整的软件配置项能否和系统正确连接，并满足系统设计文档的要求。

### 5.3.1系统测试过程

本次测试是根据前后端功能不同及涉及用户的差异进行分类测试，分成用户功能，租赁者功能，管理者功能。

用户功能测试：对于测试用户功能来说，要检测用户的账号注册是否正常运行、登录、界面浏览的信息与功能是否符合该角色功能，测试用户评论与租赁房屋的过程是否完整。

租赁者功能测试：租赁者测试主要在发布房屋、在后台管理房屋信息、管理自己房屋的评论信息、自己发布的新闻信息等功能，另外租赁者注册也是一个重要部分，在租赁者注册时必须包含自己的准确信息与协议信息，保证所有信息都真是你有效。

管理员功能测试：系统管理员功能测试主要检测管理员登录、注册账号、进入后台、审核用户、审核租赁房屋，修改新闻资讯等工作。

### 5.3.2系统测试结果

经测试系统用户与租赁者在注册填写信息时，在所有验证信息都完整且都满足的条件下允许注册，登录时只要保证登录名与密码与数据库信息匹配成功后都能登录。前台登录成功后能进行对该房屋的评论与租赁等功能，后台租赁者或者管理员登录时也要用户名与密码与数据库信息匹配正确后方等登录。管理员登录后，能查看所有用户的信息、房屋发布的信息、新闻资讯、评论信息、反馈信息与权限管理，但是不能修改房屋的信息，只能删除，对新闻信息实时更新，可删除租赁者的租赁权限。租赁者登录后能查阅自己发布的房屋信息，用户对自己房屋的评论信息，自己发布的信息咨询等信息，可以修改自己的房屋信息，但是没有权限查看其它租赁者的信息与权限管理信息。

# 致谢

本系统的设计与实现从确定需求，搜集资料，实现系统功能到论文撰写的顺利完成都离不开陆老师的用心指导，我在他严格而又悉心且不厌其烦的指导下，我解决了我的设计中的许多我查阅很对文档也解决不了的难题，通过老师的不懈指导，我学会了如何自我查找问题，用什么思路去解决问题，这将帮助我今后更加便捷灵活地去解决更多的技术难题。老师的细心指导让我在规定时间内顺利完成了系统设计实现和论文撰写工作。陆老师老师利用自己的课余时间指导我的设计。陆老师对许多专业知识的有独到见解，因此我在做毕业设计的这段时间收获了许多的经验，这对我今后出去工作有莫大的帮助，在此我很感激陆导师并致以我最崇高的敬意。

通过这次毕业设计的工作，我仍要感谢身边的同学与老师的帮助，在我遇到问题的时候不单单有指导老师一直在帮助我，还有我身边的同学们也帮我解决了许多的问题，给了我许多的建议，使我在做毕业设计的时候少走了很多的弯路，从而能够按时完成我的毕业设计。

同时感谢各位评委老师抽出宝贵的时间评审本文，对于您们不辞辛苦地评审无得论文答辩和指导，学生表示诚挚的谢意。

# 参考文献

[1]. 林晓仪, 基于B/S的房产中介管理系统. 电脑知识与技术, 2016(09): 第105-106页.

[2].李志云与国伟, 基于.NET技术的房屋中介管理系统设计与实现. 电脑编程技巧与维护, 2016(2): 第5-8页.

[3].徐勤贤与石雪花, 城镇化进程中人口流动的变化趋势. 中国经贸导刊, 2013(17): 第25-27页.

[4].董莹, 基于用户意向分析的房屋租赁系统的设计与实现, 2017, 大连海事大学.

[5].张金龙, 基于Web的房产中介管理系统, 2014, 电子科技大学. 第 80页.

[6].Doyle, G.S., A methodology for making early comparative architecture performance evaluations. 2010.

[7].陈丽华, 基于房产档案查询利用的问题及应对方法分析. 东方企业文化, 2014(6).

[8].尹婷与赵思佳, 基于jQuery框架的AJAX网站设计模式的研究. 湖南生态科学学报, 2010. 16(3): 第1-4页.

[9].曾顺, 精通JavaScript+jQuery. 2008: 人民邮电出版社.

[10].王晶与温向彬, 利用jQuery操作HTML元素. 农业网络信息, 2008(4): 第98-99页.

[11].寇毅与吴力文, 基于MVC设计模式的Struts框架的应用方法. 计算机应用, 2003. 23(11): 第93-95页.

[12].王金龙, 宋斌与丁锐, Node.js:一种新的Web应用构建技术. 现代电子技术, 2015. v.38;No.437(6): 第70-73页.

[13].王星与潘郁, 基于AJAX技术的Web模型在网站开发中的应用研究. 微计算机信息, 2006. 22(27): 第206-207页.

[14].邱晹.校园招聘系统设计及实现[D].天津:天津大学,2016.