

# 毕业设计（论文）

## 黑马智慧物业管理

姓 名：\_\_\_\_\_  
学 号：\_\_\_\_\_  
班 级：\_\_\_\_\_  
专 业：\_\_\_\_\_  
所在系：\_\_\_\_\_  
指导老师：\_\_\_\_\_  
完成日期：\_\_\_\_\_

## 摘 要

随着我国发展和城市开发，物业管理已形成规模，其效益也越来越明显。在经济效益对地方政府而言，主要体现为：减少了大量的财政补贴，对住宅区开发企业而言，能提高物业市场竞争力，使开发企业的房产畅销，加速资金周转。同时，完善的物业管理能为开发商树立良好的企业形象，吸引更多的房地产交易商和消费者。在环境效益上，住宅区内的环境和布局、治安等与整个建设风貌融为一体，提高了房地产业综合效益。但是，目前我国有相当一部分物业管理公司还停留在人工管理的基础上，这样的管理机制已经不能适应时代的发展，其管理方法存在许多缺点，如：效率低且较为繁琐，另外随着物业管理项目的增加，工作量也将大大增加，这必然增加物业管理者工作量和劳动强度，这将给物业管理信息的查找、更新和文护都带来了许多困难。随着科学技术的不断提高，这种传统的手工管理方法必然被以记算计为基础的信息管理方法所取代，智慧物业是未来物业行业发展的必然趋势。

本系统采用了 B/S 架构，Java、Html、Css、Js 等技术，使用了主流的后台开发框架 SpringBoot (SpringMVC+Spring+Mybatis)，前端开发框架使用了 LayUI、Vue、JQuery 以及 Vue 的前端组件库 Element-UI，采用了开源的轻量级数据库 Mysql 进行开发。

实现了小区管理、房产管理、设备管理、业主管理、服务管理、车位管理等主要功能。

提升物业管理水平，提高工作效率，降低成本，便于物业快速回收物业费；扩大服务范围和能力，方便住户，提升用户满意度；为物业公司扩展了新的收入渠道：租赁中介、社区团购、物业服务；为住户缴纳物业费、水电、停车费等生活费用，提供在线支付，方便快捷。

关键词：智慧城市，智慧物业，物业管理，SpringBoot，Vue

## Abstract

With China's development and urban development, property management has formed a scale, and its benefits are more and more obvious. For the local government, the economic benefits are mainly reflected in the reduction of a large number of financial subsidies. For the residential development enterprises, it can improve the competitiveness of the property market, make the real estate of the development enterprises sell well, and accelerate the capital turnover. At the same time, perfect property management can establish a good corporate image for developers and attract more real estate dealers and consumers. In terms of environmental benefits, the environment, layout and public security in the residential area are integrated with the whole construction style, which improves the comprehensive benefits of the real estate industry. However, at present, a considerable number of property management companies in our country still stay on the basis of manual management, such a management mechanism can not adapt to the development of the times, its management methods have many shortcomings, such as: low efficiency and more cumbersome, in addition, with the increase of property management projects, the workload will also increase greatly, which will inevitably increase the workload and labor intensity of property managers. This will bring a lot of difficulties to the search, update and protection of property management information. With the continuous improvement of science and technology, this traditional manual management method is bound to be replaced by the information management method based on calculation. Intelligent property is the inevitable trend of the future development of the property industry.

The system adopts B / S architecture, Java, HTML, CSS, JS and other technologies, uses the mainstream background development framework springboot (springmvc + Spring + mybatis), the front-end development framework uses layui, Vue, jQuery and element UI, the front-end component library of Vue, and uses the open source lightweight database mysql for development.

It realizes the main functions of community management, real estate management, equipment management, owner management, service management, parking management and so on.

To improve the level of property management, improve work efficiency, reduce costs, facilitate the rapid recovery of property fees; expand the scope and ability of services, facilitate residents, improve user satisfaction; expand new income channels for property companies: rental agencies, community group buying, property services; pay property fees, water and electricity, parking fees and other living expenses for residents, and provide online payment for convenience quick.

Key words: Smart City, Smart Property, Management, SpringBoot, Vue

# 目 录

黑马智慧物业管理 .....	1
ABSTRACT.....	3
目 录 .....	4
第一章 绪论 .....	6
1.1 课题背景、目的及意义.....	6
1.1.1 课题背景.....	6
1.1.2 课题目的和意义.....	6
1.2.1 国外的研究现状.....	6
1.2.2 国内的研究现状.....	6
第二章 可行性分析 .....	8
2.1 社会可行性.....	8
2.2 技术可行性.....	8
2.3 操作可行性.....	8
2.4 系统的技术介绍.....	8
2.4.1 Html.....	8
2.4.2 Vue.....	9
2.4.3 LayUI.....	9
2.4.4 MySQL.....	9
2.4.5 SpringBoot.....	9
2.4.6 MVC.....	9
2.4.7 Mybatis .....	10
2.4.8 TkMapper .....	10
2.5 系统开发平台及运行环境.....	11
2.5.1 系统开发平台.....	11
2.5.2 运行环境.....	11
第三章 需求分析 .....	12
3.1 系统功能模块概述和分析.....	12
3.2 系统功能模块设计.....	12
3.3 数据库分析.....	13
第四章 黑马智慧物业后台设计与实现 .....	17
4.1 管理员登录 .....	17
4.2 后台主页 .....	17
4.3 小区管理 .....	18
4.3.1 小区列表.....	18
5.3.2 新增小区.....	18

---

5.3.3 修改小区.....	19
5.3.4 删除小区.....	20
<b>第五章 系统测试 .....</b>	<b>21</b>
5.1 测试的目的与目标.....	21
5.2 测试方法.....	21
5.3 测试用例.....	22
5.4 测试结论.....	22
<b>第六章 结论 .....</b>	<b>24</b>
<b>第七章 致谢 .....</b>	<b>25</b>
<b>第八章 参考文献 .....</b>	<b>26</b>



## 第一章 绪论

### 1.1 课题背景、目的及意义

#### 1.1.1 课题背景

互联网+改变着我们的生活，在传统的社区物业服务领域，在服务业主、提升社区服务质量方面，如何与互联网融合？顶级互联网技术团队携手小帮住，走进社区，与物业公司共同开启一场社区服务的创新性变革。小帮住依托互联网搭建的平台为物业以及小区业主提供多元化的增值服务和全新的置业体验。

智慧物业是物业管理的一种新理念，是新形势下社会管理创新的一种新模式。智慧物业充分利用物联网、云计算、移动互联网等新一代信息技术的集成应用，为住宅者、商圈消费者和服务者提供一个安全、舒适、便利的现代化、智慧化物业环境，从而形成基于信息化、智能化社会管理与服务的一种新的管理形态的物业模式。

#### 1.1.2 课题目的和意义

随着人工智能、大数据、互联网等高新技术的发展，物业管理企业先后试水互联网，以社区生活场景为中心，搭建用户与商家、服务提供者之间的 O2O 社区服务平台。

一方面可以整合“线上”和“线下”两端的资源，并通过云端进行管理，为居民创造更加便捷的社区生活环境的同时进一步拓展增值服务发展空间，创造新的盈利点；

另一方面可提升物业管理效率，降低能耗成本与人员成本。

#### 1.2.1 国外的研究现状

物业管理在 20 世纪 60 年代在 19 世纪的英国兴起。当时正值英国工业化发展的时期,许多大城市里,农民进入出现房屋出租的现象。为文护业主的权利,需要一套有效的管理方法,从而出现了专业的物业管理机构。从那时起,物业管理遍布世界各地,受到各国的普遍重视。目前,美国、英国、中国、香港等国家或地区物业管理是非常发达是因为政府注重与业主欢迎。目前,在西方发达国家,管理信息(MIS)技术相对先进,种类齐全,管理已被广泛应用于各大企业中,通过管理信息系统实现具有检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等优点的管理模式。

#### 1.2.2 国内的研究现状

随着我国发展和城市开发,物业管理已形成规模,其效益也越来越明显。在经济效益对地方政府而言,主要体现为:减少了大量的财政补贴,对住宅区开发企业而言,能提高物业

市场竞争力，使开发企业的房产畅销，加速资金周转。同时，完善的物业管理能为开发商树立良好的企业形象，吸引更多的房地产交易商和消费者。在环境效益上，住宅区内的环境和布局、治安等与整个建设风貌融为一体，提高了房地产业综合效益。但是，目前我国有相当一部分物业管理公司还停留在人工管理的基础上，这样的管理机制已经不能适应时代的发展，其管理方法存在这许多缺点，如：效率低且较为繁琐，另外随着物业管理项目的增加，工作量也将大大增加，这必然增加物业管理者工作量和劳动强度，这将给物业管理信息的查找、更新和文护都带来了很多困难。随着科学技术的不断提高，这种传统的手工管理方法必然被以为基础的信息管理方法所取代。由于管理信息系统的发展受到以上各种变革因素的制约，传统的管理信息系统必然向现代管理信息系统发展。



## 第二章 可行性分析

### 2.1 社会可行性

随着人工智能、大数据、互联网等高新技术的发展，物业管理企业先后试水互联网，以社区生活场景为中心，搭建用户与商家、服务提供者之间的 O2O 社区服务平台。

一方面可以整合“线上”和“线下”两端的资源，并通过云端进行管理，为居民创造更加便捷的社区生活环境的同时进一步拓展增值服务发展空间，创造新的盈利点；另一方面可提升物业管理效率，降低能耗成本与人员成本。

### 2.2 技术可行性

黑马智慧物业管理系统采用当下比较流行的前端框架 Vue 和 LayUI，使用 Vue 和 LayUI 提供的现成的组件库，可以快速高效的搭建页面优美，效果炫酷的后端页面和特效。后端采用 SpringBoot 框架，也是当前 IT 软件开发行业比较前沿的技术，其自动装配和起步依赖的特点，可以帮助我们高效快速的构建 JavaEE 项目，并且项目启动异常简单。数据库采用 MySQL，可以存储海量数据，并且支持分片、分库、分表，有利于以后的程序的扩展。采用 Maven 管理项目，简化了项目依赖，并且便于项目的可持续集成。使用内置 Tomcat 部署项目，使项目的发布运行更加简单方便。

### 2.3 操作可行性

目前，大多数计算机都能运行该系统，该系统的安装、调试、运行不会改变原计算机系统的设置和布局，并且系统界面简单，提示信息完整，由相关人员指导便能够方便的操作此软件。

### 2.4 系统的技术介绍

#### 2.4.1 Html

HTML 称为超文本标记语言，是一种标记语言。它包括一系列标签。通过这些标签可以将网络上的文档格式统一，使分散的 Internet 资源连接为一个逻辑整体。HTML 文本是由 HTML 命令组成的描述性文本，HTML 命令可以说明文字，图形、动画、声音、表格、链接等。

超文本是一种组织信息的方式，它通过超级链接方法将文本中的文字、图表与其他信息媒体相关联。这些相互关联的信息媒体可能在同一文本中，也可能是其他文件，或是地理位置相距遥远的某台计算机上的文件。这种组织信息方式将分布在不同位置的信息资源用随机



方式进行连接，为人们查找，检索信息提供方便。

## 2.4.2 Vue

Vue.js 是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，Vue 采用自底向上增量开发的设计。Vue 的核心库只关注视图层，并且非常容易学习，非常容易与其它库或已有项目整合。另一方面，Vue 完全有能力驱动采用单文件组件和 Vue 生态系统支持的库开发的复杂单页应用。

Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。

Vue.js 自身不是一个全能框架——它只聚焦于视图层。因此它非常容易学习，非常容易与其它库或已有项目整合。另一方面，在与相关工具和支持库一起使用时，Vue.js 也能完美地驱动复杂的单页应用。

## 2.4.3 LayUI

由职业前端倾情打造，面向全层次的前后端开发者，低门槛开箱即用的前端 UI 解决方案

layui 更多是面向于后端开发者，所以在组织形式上毅然采用了几年前的以浏览器为宿主的类 AMD 模块管理方式，却又并非受限于 CommonJS 的那些条条框框，它拥有自己的模式，更加轻量 and 简单。layui 定义为“经典模块化”，并非是刻意强调“模块”理念本身，而是有意避开当下 JS 社区的主流方案，试图以尽可能简单的方式去诠释高效。它的所谓经典，是在于对返璞归真的执念，它以当前浏览器普通认可的方式去组织模块。layui 认为这种轻量的组织方式，仍然可以填补 WebPack 以外的许多场景。所以它坚持采用经典模块化，也正是能让人避开工具的复杂配置，重新回归到原生态的 HTML/CSS/JavaScript 本身。

## 2.4.4 MySQL

MySQL 是一个开放源码的小型关联式数据库管理系统，开发者为瑞典 MySQL AB 公司。MySQL 被广泛地应用在 Internet 上的中小型网站中。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了 MySQL 作为网站数据库。

自从 Oracle 公司收购了 MySQL 后不久，就发行了 MySQL 的企业版（不再免费）！

## 2.4.5 SpringBoot

SpringBoot 是由 Pivotal 团队在 2013 年开始研发、2014 年 4 月发布第一个版本的全新开源的轻量级框架。它基于 Spring4.0 设计，不仅继承了 Spring 框架原有的优秀特性，而且还通过简化配置来进一步简化了 Spring 应用的整个搭建和开发过程。另外 SpringBoot 通过集成大量的框架使得依赖包的版本冲突，以及引用的不稳定性等问题得到了很好的解决。

## 2.4.6 MVC

MVC 即模型—视图—控制器，是 Xerox PARC 在八十年代为编程语言 Smalltalk—80 发明的一种软件设计模式，至今已被广泛使用。最近几年被推荐为 Sun 公司 J2EE 平台的设计模式，并且受到越来越多的使用 ColdFusion 和 PHP 的开发者的欢迎。

MVC 是一种设计模式，它强制性的使应用程序的输入、处理和输出分开。使用 MVC 应用程序被分成三个核心部件：模型、视图、控制器。它们各自处理自己的任务。

### (1) 模型

模型表示企业数据和业务规则。在 MVC(model,view ,control 模型层,视图层,控制层)的三个部件中，模型拥有最多的处理任务。例如它可能用象 EJBs 和 ColdFusion Components 这样的构件对象来处理数据库。被模型返回的数据是中立的，就是说模型与数据格式无关，这样一个模型能为多个视图提供数据。由于应用于模型的代码只需写一次就可以被多个视图重用，所以减少了代码的重复性。

### (2) 视图

视图是用户看到并与之交互的界面。对老式的 Web 应用程序来说，视图就是由 HTML 元素组成的界面，在新式的 Web 应用程序中，HTML 依旧在视图中扮演着重要的角色，但一些新的技术已层出不穷，它们包括 Macromedia Flash 和象 XHTML，XML/XSL，WML 等一些标识语言和 Web services. 如何处理应用程序的界面变得越来越有挑战性。MVC 一个大的好处是它能为你的应用程序处理很多不同的视图。在视图中其实没有真正的处理发生，不管这些数据是联机存储的还是一个雇员列表，作为视图来讲，它只是作为一种输出数据并允许用户操纵的方式。

### (3) 控制器

控制器接受用户的输入并调用模型和视图去完成用户的需求。所以当单击 Web 页面中的超链接和发送 HTML 表单时，控制器本身不输出任何东西和做任何的处理。它只是接收请求并决定调用哪个模型构件去处理请求，然后确定用哪个视图来显示模型处理返回的数据。

综上所述，MVC 的处理过程是首先控制器接收用户的请求，并决定应该调用哪个模型来进行处理，然后模型用业务逻辑来处理用户的请求并返回数据，最后控制器用相应的视图格式化模型返回的数据，并通过表示层呈现给用户。

## 2.4.7 Mybatis

MyBatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，将接口和 Java 的 POJOs(Plain Ordinary Java Object, 普通的 Java 对象)映射成数据库中的记录。

## 2.4.8 TkMapper

TkMapper 又名通用 Mapper，封装了对单表的 CRUD 操作。

通用 Mapper 都可以极大的方便开发人员。可以随意的按照自己的需要选择通用方法，还可以很方便的开发自己的通用方法。

## 2.5 系统开发平台及运行环境

### 2.5.1 系统开发平台

系统的开发是在 win10 系统 JDK8 环境下进行的。使用 SpringBoot 内置的 Tomcat 作为 web 服务器，Tomcat 是一个免费的开源的 Servlet 容器，它是 Apache 基金会的 Jakarta 项目中的一个核心项目，由 Apache，Sun 和其它一些公司及个人共同开发而成。Tomcat 被 Java World 杂志的编辑选为 2001 年度最具创新的 Java 产品，可见其在业界的地位。

Tomcat 的环境主要有以下几方面技术优势：

1. Tomcat 完全开源免费。
2. 在 SpringBoot 支持使用嵌入式 Tomcat 服务器，并且默认就是使用 Tomcat 服务器，运行项目只需要执行引导类的 main 方法即可，运行项目非常简单方便。
3. Tomcat 不仅仅是一个 Servlet 容器，它也具有传统的 Web 服务器的功能：处理 html 页面。
4. Tomcat 也可以与其它一些软件集成起来实现更多的功能。

### 2.5.2 运行环境

操作系统：Windows 10 专业版。

服务器软件：Tomcat9.0.34。

浏览器：Google Chrome。

数据库：MySQL5.5.27。

开发工具：IntelliJIDEA\_2019.3。

项目管理工具：Maven3.3.9。

## 第三章 需求分析

### 3.1 系统功能模块概述和分析

黑马智慧物业管理后台，是一款高效、快捷、高质量的物业管理系统，是智慧城市的一个重要组成部分。主要包含的功能模块：小区管理、房产管理、业主信息管理、停车位管理、服务管理、资产管理、收费管理、物业管理员管理、系统设置等功能模块。本系统完全基于H5、CSS3 技术，在系统的设计与开发过程中严格遵守软件工程的规范，运用软件设计模式，从而减少系统模块间的耦合，力求保障系统的稳定性、可重用性和可扩充性。

黑马智慧物业主要功能如下：

(1) 微信小程序（业主端）：

- ① 用户管理：注册会员、登录、(邮箱)激活、退出、修改密码；
- ② 报事报修：报事报修工单列表管理及进度展示，对工单进行评价；
- ③ 物业费管理：物业费缴费明细列表及物业费用缴纳；
- ④ 水电燃气管理：水电燃气费用明细列表及费用缴纳；
- ⑤ 访客录入：录入来访亲友信息，可以开启临时停车位和门禁；
- ⑥ 车位管理：自己车位信息，车位类型（购买、年租、月租），车位费查询。

(2) 微信小程序（物业维修人员端）：

- ⑦ 用户管理：注册会员、登录、(邮箱)激活、退出、修改密码；
- ⑧ 报事报修：报事报修工单列表，工单承接及工单进度管理，工单评价查看；

(3) 黑马智慧物业后台管理系统（物业管理人员）：

- ① 小区管理：新增小区，小区列表展示，小区信息修改，小区移除；
- ② 房产管理：新增房产，房产列表展示，房产信息修改，房产移除；新增楼栋，楼栋列表展示，楼栋信息修改，楼栋移除；
- ③ 业主信息管理：新增业主，业主列表展示，业主信息修改，业主移除；
- ④ 停车位管理：车位管理，车位使用管理；
- ⑤ 服务管理：活动管理，报修管理，投诉管理，信箱管理；

...

### 3.2 系统功能模块设计

根据系统功能分析，可以画出系统的功能模块图。

黑马智慧物业后台管理系统

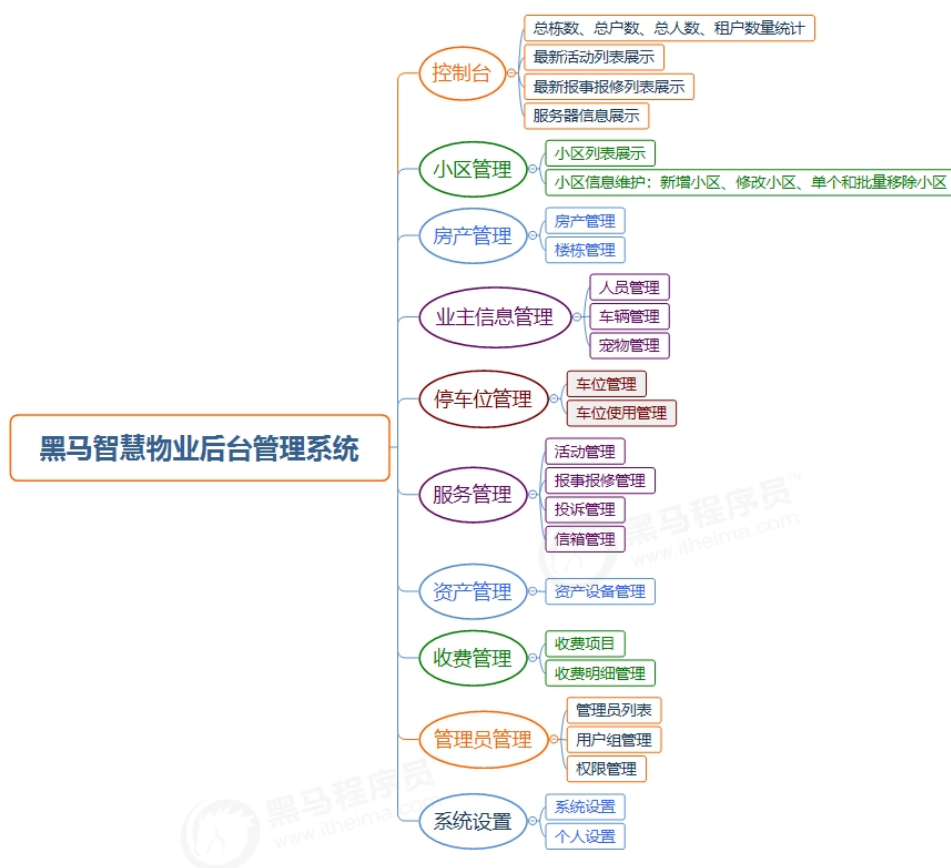


图 3.1 黑马智慧物业后台管理系统

Fig. 3.1 Heima property background management system

### 3.3 数据库分析

信息系统的主要任务是通过大量数据获得管理所需要的信息，这就要求系统本身能够存储和管理大量的数据，而这一功能的实现必须借助大型数据库系统。本系统的开发选择 MySQL 作为后台数据库开发工具。

#### 1. 概念模型设计

概念模型用于信息世界的建模，与具体的 DBMS 无关。为了把现实世界中的具体事物抽象、组织为某一 DBMS 支持的数据模型。人们常常首先将现实世界抽象为信息世界，然后再将信息世界转换为机器世界。也就是说，首先把现实世界中的客观对象抽象为某一种信息结构，这种信息结构并不依赖于具体的计算机系统和具体的 DBMS，而是概念级的模型，然后再把模型转换为计算机上某一个 DBMS 支持的数据模型。实际上，概念模型是现实世界到机器世界的一个中间层次。

信息世界中包含的基本概念有实体和联系。

##### (1) 实体 (entity)

客观存在并可相互区别的事物称为实体。实体可以是具体的人、事、物，也可以是抽象的概念或联系。例如，一个小区、一栋楼、一个房屋、一个业主、一辆车、一位车位等都是实体。

##### (2) 联系 (relationship)

在现实世界中，事物内部以及事物之间是有联系的，这些联系在信息世界中反映为实体

内部的联系和实体之间的联系。实体内部的联系通常是组成实体的各属性之间的联系。两个实体型之间的联系可以分为 3 类，一对一联系，(1:1)；一对多联系(1:n)；多对多联系(m:n)。

概念模型是对信息世界建模，所以概念模型应该能够方便、准确地表示信息世界中的常用概念。概念模型的表示方法很多，其中最为常用的是 P.P.S.Chen 于 1976 年提出的实体，联系方法(Entity-Relationship Approach)简记为 E-R 表示法)。该方法用 E-R 图来描述现实世界的概念模型，称为实体-联系模型，简称 E-R 模型。

根据设计规划的主要实体有：小区、楼栋、房屋、业主、车辆、物业人员，活动，设备，维修等。

## 2. 数据库表设计

数据库表设计主要是把概念结构设计时设计好的基本 E-R 图转换为与选用 DBMS 产品所支持的数据模型相符合的逻辑结构。它包括数据项、记录及记录间的联系、安全性和一致性约束等等。导出的逻辑结构是否与概念模式一致，从功能和性能上是否满足用户的要求，要进行模式评价。

本系统数据库表如下：

### (1) tb\_community


Field	Type	Comment
 id	int(11) NOT NULL	小区id
code	varchar(20) NOT NULL	小区编号
name	varchar(50) NOT NULL	小区名称
address	varchar(200) NULL	坐落地址
area	double NULL	占地面积 (m2)
total_buildings	int(11) NULL	总栋数
total_households	int(11) NULL	总户数
greening_rate	int(11) NULL	绿化率 (%)
thumbnail	varchar(200) NULL	缩略图
developer	varchar(100) NULL	开发商名称
estate_company	varchar(100) NULL	物业公司名称
create_time	timestamp NULL	创建时间
update_time	timestamp NULL	更新时间
status	char(1) NULL	状态:0-启用 (默认), 1-不启用

图 3.3.1 小区表

### (2) tb\_building


Field	Type	Comment
 id	int(11) NOT NULL	楼栋ID
community_name	varchar(50) NULL	所属小区名称
community_id	int(11) NULL	所属小区ID
name	varchar(50) NULL	栋数名称
total_households	int(11) NULL	总户数
description	varchar(500) NULL	描述
create_time	timestamp NOT NULL	创建时间
update_time	timestamp NOT NULL	更新时间

图 3.3.2 楼栋表

### (3) tb\_house


Field	Type	Comment
 id	int(11) NOT NULL	房屋ID
community_name	varchar(50) NULL	所属小区名称
community_id	int(11) NULL	所属小区ID
building_name	varchar(50) NULL	所属栋数名称
building_id	int(11) NULL	所属栋数ID
code	varchar(50) NULL	房产编码
name	varchar(100) NULL	房产名称
owner_id	int(11) NULL	户主（业主）ID
owner_name	varchar(50) NULL	户主（业主）名称
telephone	int(20) NULL	联系方式
room_num	int(11) NULL	房间数
unit	int(2) NULL	单元
floor	int(3) NULL	楼层
description	varchar(500) NULL	描述
live_time	timestamp NOT NULL	入住时间

图 3.3.3 房屋表

(4) tb\_owner


Field	Type	Comment
 id	int(11) NOT NULL	业主ID
community_id	int(11) NULL	所属小区
house_id	int(11) NULL	所属房产
name	varchar(50) NULL	成员名称
picture	varchar(100) NULL	成员照片
idcard	varchar(18) NULL	身份证号
telephone	int(20) NULL	联系方式
profession	varchar(20) NULL	职业
sex	char(1) NULL	性别:0-男（默认），1-女
type	char(1) NULL	类型:0-房主（默认），1-租客
remark	varchar(200) NULL	备注（默认无）
create_time	timestamp NOT NULL	创建时间
update_time	timestamp NOT NULL	更新时间
birthday	timestamp NOT NULL	出生日期

图 3.3.4 业主表

(5) tb\_car


Field	Type	Comment
 id	int(11) NOT NULL	车辆ID
picture	varchar(100) NULL	车辆照片
owner_id	int(11) NULL	所属成员（业主）
color	varchar(10) NULL	车辆颜色
car_number	varchar(20) NULL	车牌号
remark	varchar(500) NULL	备注
create_time	timestamp NOT NULL	创建时间
update_time	timestamp NOT NULL	更新时间

图 3.3.5 车辆表

(6) tb\_pet


	Field	Type	Comment
	id	int(11) NOT NULL	车位ID
	community_name	varchar(50) NULL	所属小区名称
	community_id	int(11) NULL	所属小区ID
	picture	varchar(100) NULL	车位图片
	code	varchar(20) NULL	车位编号
	name	varchar(50) NULL	车位名称
	create_time	timestamp NOT NULL	创建时间
	update_time	timestamp NOT NULL	更新时间

图 3.3.5 宠物表

- (7) tb\_parking
- (8) tb\_parking\_use
- (9) tb\_device
- (10)...



## 第四章 黑马智慧物业后台设计与实现

黑马智慧物业管理系统是为了方便物业管理工作人员，给业主提供高效、便捷、高质量的物业服务 and 舒适、安全的生活环境而设计的。

### 4.1 管理员登录

后台管理员登录页面，登录成功后到达后台主页。

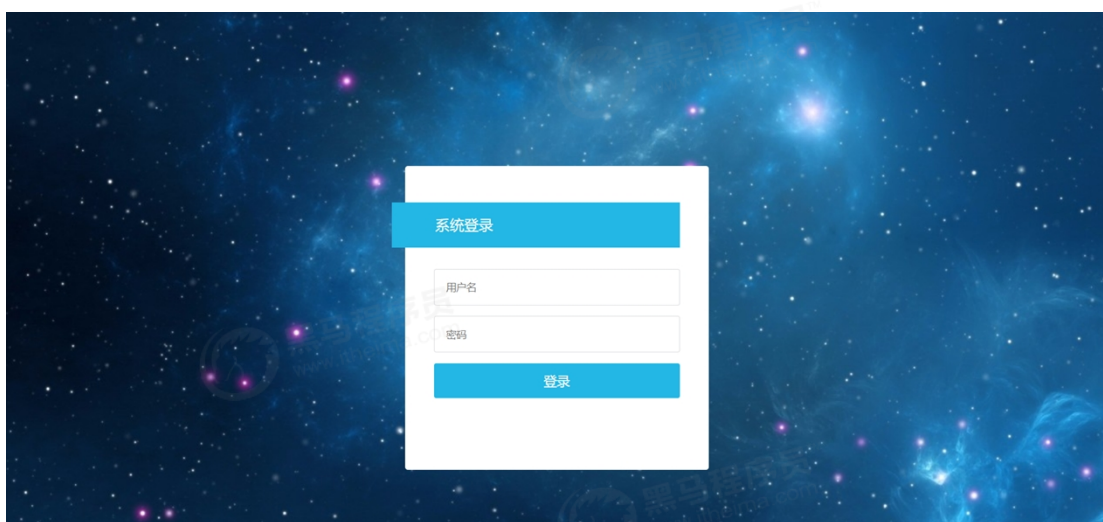


图 4.1 登录界面

### 4.2 后台主页

管理员登录成功后，到达主页。主页是框架页，由左、右两部分构成。

- 左侧显示菜单，分别为：“控制台”、“小区管理”、“房产管理”、“业主信息管理”、“停车位管理”、“服务管理”等菜单；
- 右侧默认显示“控制台”页面内容，当点击左侧菜单后，对应的页面会替换默认的“控制台”页面显示在右侧内容显示框。



## 4.3 小区管理

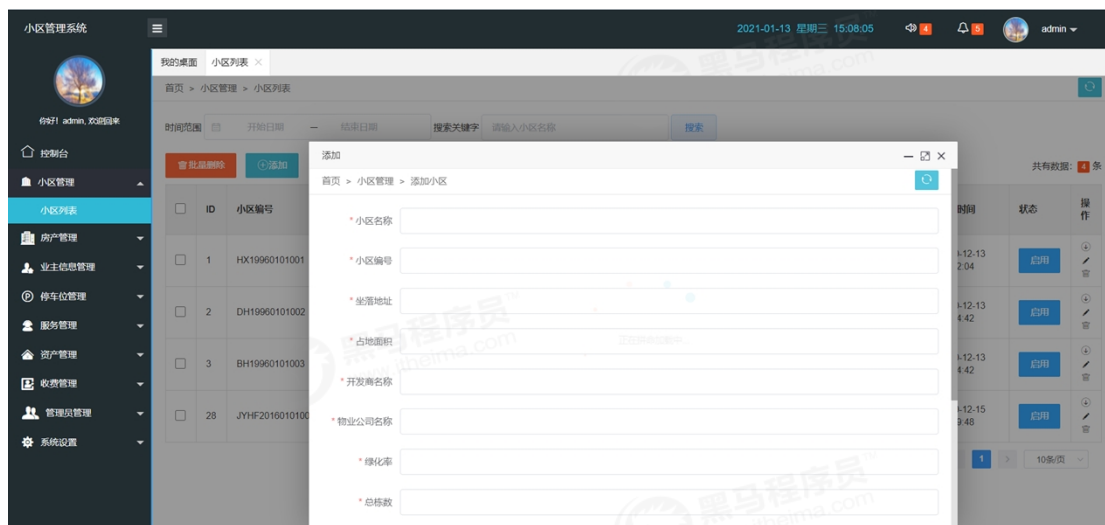
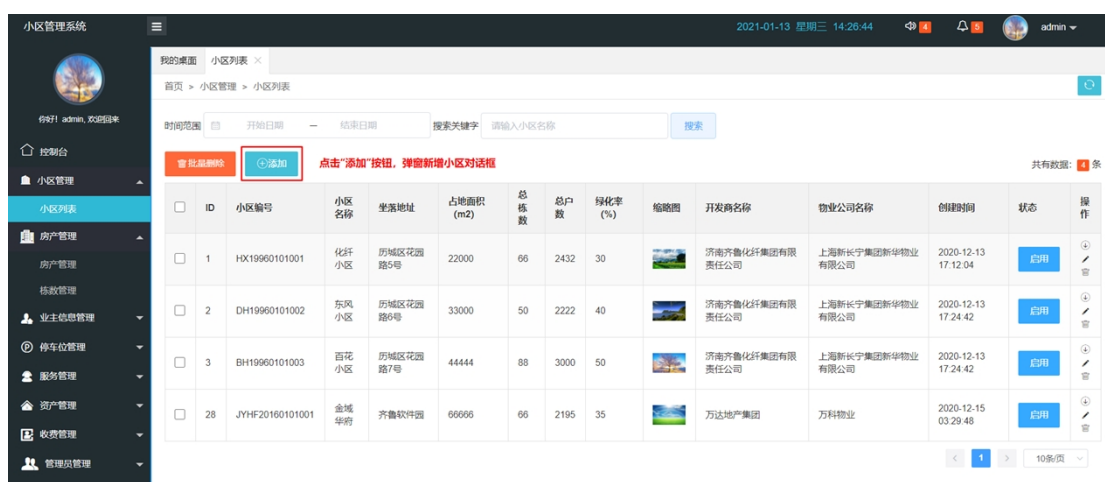
### 4.3.1 小区列表

当点击首页左侧菜单栏的“小区管理”目录下的“小区列表”链接时，会加载“小区列表页面”到右侧内容显示位置进行小区列表数据展示。也可以选择时间范围和填写小区名称，搜索自己想查看的小区的信息，并且可以自己设置每页显示的数据的条数，默认显示 2 条。

ID	小区编号	小区名称	坐落地址	占地面积 (m2)	总栋数	总户数	绿化率 (%)	地图图	开发商名称	物业公司名称	创建时间	状态	操作
1	HD19960101001	化纤小区	历城区花园路5号	22000	66	2432	30		济南齐鲁化纤集团有限责任公司	上海新长宁集团新华物业有限公司	2020-12-13 17:12:04	启用	
2	DH19960101002	东风小区	历城区花园路6号	33000	50	2222	40		济南齐鲁化纤集团有限责任公司	上海新长宁集团新华物业有限公司	2020-12-13 17:24:42	启用	
3	BH19960101003	百花小区	历城区花园路7号	44444	88	3000	50		济南齐鲁化纤集团有限责任公司	上海新长宁集团新华物业有限公司	2020-12-13 17:24:42	启用	
28	JYHF20160101001	金城华府	齐鲁软件园	66666	66	2195	35		万达地产集团	万科物业	2020-12-15 03:29:48	启用	

### 5.3.2 新增小区

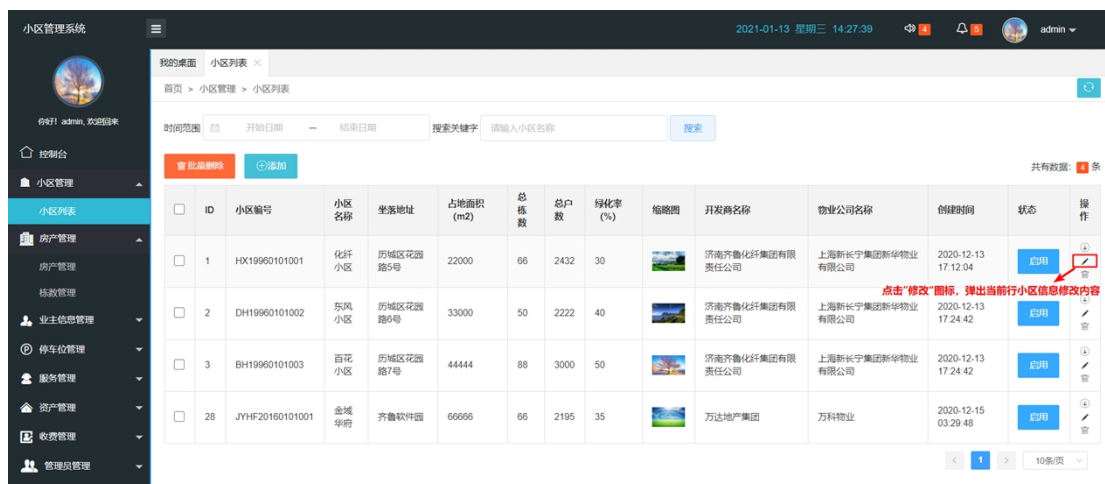
在小区列表页面中点击“添加”按钮，会弹出新增小区的弹出框。



填写完小区信息，新增成功查看小区列表

### 5.3.3 修改小区

在小区列表页面，在右侧内容展示列表，在每行数据最末端，在操作列中，点击“修改”按钮，弹窗修改对话框，对该行小区信息进行修改，修改完成后手动关闭对话框，可以在列表查看到修改后的最新小区信息。



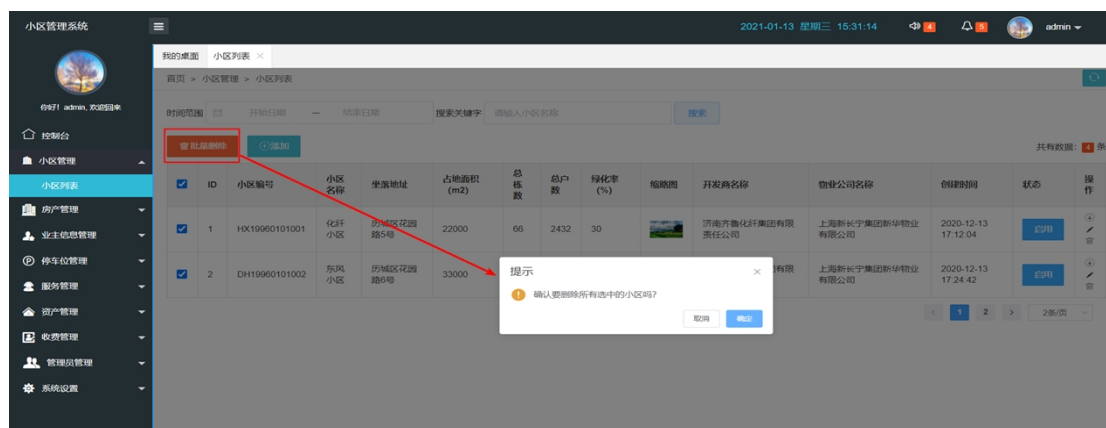
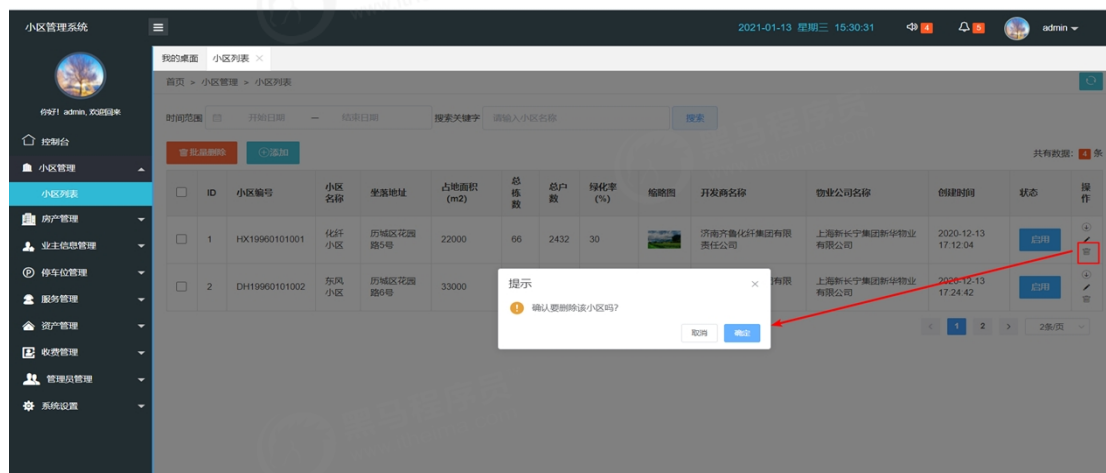


### 5.3.4 删除小区

小区信息的删除分两类，可以单行删除，也可以多行删除。

单行删除，在每行操作那一栏有一个删除图标，点击就会弹窗对话框提示是否删除该小区信息，点击“确认”按钮就可以真正删除该小区信息。

多行删除，需要先勾选每行数据前复选框，然后点击“批量删除按钮”，也会弹窗对话框提示是否删除所有小区信息，点击“确认”按钮就可以真正删除所有勾选的小区信息。



## 第五章 系统测试

### 5.1 测试的目的与目标

在此系统进行初步实现之后，开始进行对系统进行测试，找出系统中存在的 Bug，通过测试，用提交的 Bug 报告来为以后软件的改进提供标准和参考，能够在以后的系统改进中找到依据。

测试后的软件各模块基本功能可以顺利进行，尽可能的提高软件的健壮性。

### 5.2 测试方法

- 从是否关心软件内部结构和具体实现的角度划分：黑盒测试和白盒测试；
- 从是否执行程序的角度：静态测试和动态测试；
- 从软件开发的过程按阶段划分有：单元测试、集成测试、确认测试、系统测试、验收测试、回归测试、Alpha 测试、Beta 测试；

单元测试又称模块测试，是针对软件设计的最小单位——程序模块（这里所说的程序模块在 Java 中一个模块就是一个方法），进行正确性检验的测试工作。其目的在于发现各模块内部可能存在的各种差错。单元测试需要从程序的内部结构出发设计测试用例。多个模块可以平行地独立进行单元测试。

集成测试（组装测试、联合测试），通常在单元测试的基础上，需要将所有模块按照设计要求组装成为系统。这时需要考虑的问题是：

- 在把各个模块连接起来的时候，穿越模块接口的数据是否会丢失；
- 一个模块的功能是否会对另一个模块的功能产生不利的影响；
- 各个子功能组合起来，能否达到预期要求的父功能；
- 全局数据结构是否有问题；
- 单个模块的误差累积起来，是否会放大，从而达到不能接受的程度。

确认测试（Validation Testing），确认测试又称有效性测试。任务是验证软件的功能和性能及其它特性是否与用户的要求一致。对软件的功能和性能要求在软件需求规格说明书中已经明确规定。它包含的信息就是软件确认测试的基础。




系统测试（System Testing），是将通过确认测试的软件，作为整个基于计算机系统的一个元素，与计算机硬件、外设、某些支持软件、数据和人员等其它系统元素结合在一起，在实际运行环境下，对计算机系统进行一系列的组装测试和确认测试。系统测试的目的在于通过与系统的需求定义作比较，发现软件与系统的定义不符合或与之矛盾的地方。

验收测试（Acceptance Testing），在通过了系统的有效性测试及软件配置审查之后，就应开始系统的验收测试。验收测试是以用户为主的测试。软件开发人员和 QA（质量保证）人员也应参加。由用户参加设计测试用例，使用生产中的实际数据进行测试。在测试过程中，除了考虑软件的功能和性能外，还应对软件的可移植性、兼容性、可维护性、错误的恢复功能等进行确认。

## 5.3 测试用例

由于功能模块较多，测试内容篇幅较长，所以在本文中只介绍小区管理的测试用例，表 5.3 是本系统小区管理模块功能测试测试表，从测试的结果来看与期望结果完全相同。

5.3 小区管理模块功能测试

功能特性	小区管理模块功能代码实现				
测试目的	小区模块功能是否完成				
测试数据	{id=1, code='HX19960101001', name='化纤小区', address='历城区花园路5号', area=22000.0, totalBuildings=66, totalHousehold=2432, greeningRate=30, thumbnail='/fileupload/d2cea885-a986-463a-97da-7073c9e6fbf6.jpg', developer='济南齐鲁化纤集团有限责任公司', estateCompany='上海新长宁集团新华物业有限公司', createTime=Sun Dec 13 17:12:04 CST 2020, updateTime=Sun Dec 13 17:26:12 CST 2020, status='0'}				
测试内容	操作描述	数据	期望结果	实际结果	测试状态
1	小区列表展示	数据库已有测试数据	分页展示小区列表，并且可以自己设置每页展示条数		与期望结果相同
2	新增小区	测试数据如上	表中新增一条小区数据，并且小区列表可以立即展示新增的小区信息		与期望结果相同
3	修改小区	把占地面积修改为22222	显示警告信息“用户名或密码误!”		与期望结果相同
4	小区删除	删除该小区信息，列表上没有该小区信息	正确登入到会员操作界面	该小区数据从列表移除，页面不显示小区信息，并且数据库中该小区信息也不存在	与期望结果相同

## 5.4 测试结论

把开始的代码写得越好，它出现的错误也就越少，你也就越能相信所做过的测试是彻底的。系统化测试以一种有序方式设法探测潜在的麻烦位置。同样，毛病最可能出现在边界，这可以通过手工的或者程序的方式检查。自动进行测试是最理想的，用得越多越好，因为机器不会犯错误、不会疲劳、不会用臆想某此实际无法工作的东西能行来欺骗自己。回归测试检查一个程序是否能产生与它们过去相同的输出。在做了小改变之后就测试是一种好技术，能帮助我们出现问题的范围局部化，因为新问题一般就出现在新代码里面。

测试和排错常常被说成是一个阶段，实际上它们根本不是同一件事。简单地说，排错是

在你已经知道程序有问题时要做的事情。而测试则是在你在认为程序能工作的情况下，排错是在你已经知道程序有问题时要做的事情。而测试则是在你在认为程序能工作的情况下，为设法打败它而进行的一整套确定的系统化的试验。

Edsger Dijkstra 有一个非常有名的说法：测试能够说明程序中有错误，但却不能说明其中没有错误。他的希望是，程序可以通过某种构造过程正确地做出来，这样就不再会有错误了，因此测试也就不必要了。这确实是个美好生活的目标，但是，对今天的实际程序而言，这仍然还只是一个理想。所以应该集中精力讨论如何测试，如何才能更快地发现程序错误，如何才能使得工作更有成效、效率更高。



## 第六章 结论

通过开发本系统，我较全面的掌握了 SpringBoot 的基本知识和编程技巧，并在开发过程中我的 JAVA 开发能力得到了进一步的提高。如：更进一步的深入了解 MVC 设计模式，软件三层框架；SQL 语言的使用。

在开发过程中我学到了一些经验，系统分析的好坏将决定着系统开发成功与否，一份好分析设计将是成功开发主要因素。我们在着手开发之前不要急于编程，先应有较长的时间去把分析做好，做好数据库设计工作，写出相关的开发文档等。然后再开始编写程序代码，这样做到每写一步代码心底有数，有条不紊。当然也有些还需待继续深入地方如：插件的使用，设计工具的使用，代码的熟练程度等。

此外，我还觉得，我个人在这次设计中走了很多弯路。主要是因为平时很少接触软件开发工作，在应用方面缺乏经验，以后还需要更多的努力。

对我来说，这次设计的本身所产生的影响，还远远没有结束，我从本次毕业设计中学到了许多课本上没有的知识。通过自己的学习和努力，通过老师的指导和教育，使我不仅仅在知识水平和解决实际问题的能力上有了很大的提高。还从思想的深处体会到，要把自己的所学变成现实时所面对的种种难题。

系统不免有错误和待改进之处，真诚欢迎各位师长、同行提出宝贵意见。



## 第七章 致谢

在论文完成之际，我首先要向尊敬的 XXX 老师表示最真挚的谢意。

在论文写作期间，为了保证我们毕业设计的正常进行，学院抽调了优秀的老师指导我们进行毕业设计，并且不时地询问我们毕业设计的进展情况。没有老师们的细心指导我的论文与系统就不可能顺利的完成，再次对你们表示衷心地感谢。老师认真负责的工作态度、严谨的治学风格，使我深受启发；同时也很感谢帮助过我和我一同探讨问题的同学们。为我们这次设计的正常开展提供了必要的基础。本次毕业设计，就要画上一个句号了。

感谢信息工程学院给我提供学习的平台，让我在这里接受了四年正规系统的本科学习。感谢所有关心、支持、帮助过我的良师益友。

最后，向在百忙中抽出时间对本文进行评审并提出宝贵意见的各位老师表示衷心地感谢！

## 第八章 参考文献

- [1]岳欣.推进我国智慧城市发展的思考[J].宏观经济管理,2013,(11).
- [2]邓贤峰."智慧城市"评价指标体系研究[J].发展研究,2010,(12).
- [3]贺小花.合肥:努力打造特色智慧城市样板[J].中国公共安全(综合版),2014,(7).
- [4]杨再高.智慧城市发展策略研究[J].科技管理研究,2012,(7).
- [5]史璐.智慧城市的原理及其在我国城市发展中的功能和意义[J].中国科技论坛,2011,(5).doi:10.3969/j.issn.1002-6711.2011.05.017.
- [6] 张孝祥, Java 就业培训教程(附盘), 清华大学, (2007-07)
- [7] 孙素华编著. Dreamweaver CS5 Flash CS5 Photoshop CS5 网页设计从入门到精通: 中国青年出版社, 2014.10
- [8] CSDN 首页- 程序员杂志. 2015-08-11
- [9] 邓笑.基于 Spring Boot 的校园轻博客系统的设计与实现[D].湖北:华中科技大学,2018
- [10] 卢潇.软件工程.北京:清华大学出版社;北京交通大学出版社, 2005
- [11] 萨师煊, 王珊.数据库系统概论.北京:高等教育出版社, 2000
- [12] PaulJPerrone etal, 张志伟, 谭郁松, 张明杰.J2EE 构建企业系统[M].北京:清华大学出版社, 2001
- [13] Bruce Shive.Research Direction in Object-Oriented Programming,2002
- [14] Apache Software.The SpringBoot User's Guide.http://spring.io