智慧社区综合管理平台

——业主APP的设计和实现

摘 要

当今时代，科学技术飞速发展，应用到了生活中的方方面面，甚至包括我们所生活的城市和社区。智慧社区在我们日常生活中的应用和推广水平还停留在初期。本篇文章的主要内容介绍了一种智慧社区中社工APP的设计和实现。社工APP的开发目的是为了方便社区工作者对智慧社区内各种服务和业务的管理，此应用为社区中的各种角色提供了极大的使用的便捷。目前国内的社区管理系统大多都没有这种系统的实现，无论是管理难度还是管理方法，都难以让社区居民达到满意的标准。本课题项目后台主要使用Java语言来实现，框架使用主流的Spring+SpringMVC+Mybatis框架，前台使用JSP技术,本系统的主要功能有政务公开、查询医疗信息、查询养老信息、查询志愿活动、查询法律信息、查询社会组织、查看招工就业信息、失物招领、物业、二手交易。[1]

关键词： 智慧社区 业主APP 社区活动

Smart Community Integrated Management Platform

——Design and implementation of social worker APP

**ABSTRACT**

In today's era, science and technology are rapidly developing and applied to all aspects of life, even the cities and communities in which we live. The level of application and promotion of smart communities in our daily lives is still at an early stage. The main content of this article introduces the design and implementation of a social worker APP in a smart community. The social worker APP was developed to facilitate the management of various services and services in the smart community by community workers. This application provides great convenience for various roles in the community. At present, most of the domestic community management systems do not have such a system. Whether it is management difficulty or management methods, it is difficult for community residents to reach satisfactory standards. The background of this project is mainly implemented in Java language. The framework uses the mainstream Spring+SpringMVC+Mybatis framework, and the foreground uses JSP technology. The main functions of this system include grid information viewing, building information viewing, owner information viewing, visiting objects, community activities, photo uploading and business inquiry. [2]

**Key Words:** Smart community Owner APP Community activity

目 录

[第一章 绪论 5](#_Toc9230949)

[1.1 智慧社区平台的开发背景 5](#_Toc9230950)

[1.2 智慧社区综合管理平台的国内外发展状况 5](#_Toc9230951)

[1.3 智慧社区综合管理平台的基本特征 7](#_Toc9230952)

[1.4 智慧社区综合管理平台的开发意义 7](#_Toc9230953)

[第二章 开发环境 9](#_Toc9230954)

[2.1 开发环境 9](#_Toc9230955)

[2.2 相关技术 9](#_Toc9230956)

[2.2.1 所使用的语言 9](#_Toc9230957)

[2.2.2 所涉及的主要技术与框架 10](#_Toc9230958)

[第三章 需求分析 11](#_Toc9230959)

[3.1 可行性分析 11](#_Toc9230960)

[3.1.1 经济可行性 11](#_Toc9230961)

[3.1.2 技术可行性 11](#_Toc9230962)

[3.1.3 社会可行性 11](#_Toc9230963)

[3.1.4 可行性分析总结 11](#_Toc9230964)

[3.2 系统总体规划 12](#_Toc9230965)

[3.2.1 功能模块划分 12](#_Toc9230966)

[3.2.2 业主APP系统模块 12](#_Toc9230967)

[3.3 系统模块详解 13](#_Toc9230968)

[3.3.2 社会组织模块 15](#_Toc9230969)

[3.3.2.1 数据流图 15](#_Toc9230970)

[3.3.2.2 数据字典 15](#_Toc9230971)

[3.3.3 招工就业模块 16](#_Toc9230972)

[3.3.3.1 数据流图 16](#_Toc9230973)

[3.3.3.2 数据字典 16](#_Toc9230974)

[3.3.4失物招领模块 18](#_Toc9230975)

[3.3.4.1 数据流图 18](#_Toc9230976)

[3.3.4.2 数据字典 18](#_Toc9230977)

[3.3.5物业模块 19](#_Toc9230978)

[3.3.5.1 数据流图 19](#_Toc9230979)

[3.3.5.2 数据字典 20](#_Toc9230980)

[3.3.6二手交易模块 22](#_Toc9230981)

[3.3.6.1 数据流图 22](#_Toc9230982)

[3.3.6.2 数据字典 22](#_Toc9230983)

[第四章 概要设计 23](#_Toc9230984)

[4.1 系统功能设计 23](#_Toc9230985)

[4.1.1 查看政务公开信息功能 24](#_Toc9230986)

[4.1.2 查看医疗信息功能 24](#_Toc9230987)

[4.1.3 查看养老信息功能 24](#_Toc9230988)

[4.1.4 查看志愿活动信息功能 25](#_Toc9230989)

[4.1.5 查看法律信息功能 25](#_Toc9230990)

[4.1.6 查看社会组织信息功能 25](#_Toc9230991)

[4.1.7 查看招工就业信息功能 26](#_Toc9230992)

[4.1.8 失物招领功能 26](#_Toc9230993)

[4.1.9 物业功能 26](#_Toc9230994)

[4.1.10 二手交易功能 27](#_Toc9230995)

[4.2 系统用例分析 27](#_Toc9230996)

[4.2.1 业主APP总功能用例 27](#_Toc9230997)

[4.3 数据库设计 30](#_Toc9230998)

[4.3.1 概念结构设计 30](#_Toc9230999)

[4.3.2 逻辑结构设计 31](#_Toc9231000)

[4.3.3 物理结构设计 34](#_Toc9231001)

[第五章 详细设计 39](#_Toc9231002)

[5.1 功能类图 39](#_Toc9231003)

[5.2 功能详细设计 40](#_Toc9231004)

[第六章 项目结构与功能实现 46](#_Toc9231005)

[6.1 项目结构 46](#_Toc9231006)

[6.2 功能实现 47](#_Toc9231007)

[第七章 软件测试 57](#_Toc9231008)

[7.1 测试目标 57](#_Toc9231009)

[7.2 测试环境 57](#_Toc9231010)

[7.3 测试内容 57](#_Toc9231011)

[第八章 项目总结 63](#_Toc9231012)

[致 谢 67](#_Toc9231013)

# 第一章 绪论

* 1. 智慧社区平台的开发背景

现如今，各个城市在不断地发展科技，随着科技的进步，人们的生活住所已经从过去的平房房区变成了现代化的楼房，加上人们的生活水平也在不断的大幅度提升，每家每户都至少拥有一间房子，而建造楼房不可避免的就要建设社区，从而城市的社区辆总数在逐渐增加，社区的管理问题逐渐显露出来，社区的高效管理逐渐成为中国大城市的一个难题之一[1]。在很多城市，智慧社区并没有普遍起来，目前存在的还是过去那种老式社区。相对于新兴的智慧社区来说，这种社区的社区服务并没有智慧社区那么高效便捷。所有社区服务都需要社区工作者管理者通过人工去完成，这样消耗了大量的时间和精力，并且服务完成的效率也很低，社区居民们的生活体验也随之下降并没有达到理想的标准。[2]而智慧社区的出现，在一定程度上解决了这种低效率的工作管理方式，转而实现了一种高效、稳定、便捷的社区服务管理，这大大提高了服务效率，使社区居民对社区服务的使用有个更加方便和快捷的理解，很大程度上提高了社区居民们的生活水平[3]。

* 1. 智慧社区综合管理平台的国内外发展状况

不同地区智慧社区综合管理平台的发展情况都不是很相同。国外经历了八十年代初期的居民住宅电子化、八十年代中期的居民住宅自动化接着到九十年代美国提出的“智慧屋”（Wise House），欧洲提出的“聪明屋”（Smart Home）的住宅智能化三个阶段[4]。“智慧的社区”这一现代化系统平台的建设雏形灵感起源于美国。由于当今社会和市场飞速发展的硬性需求，同时，1980年左右“智能化住宅技术联盟”的成立，对居民住宅进行现代化设计和建造。[3]

二十世纪末，国外出现了面向中国的“维纳斯”计划“，其核心是通过计算机操控居民家中的各种家居来将家具使用更加方便。接着国内以TCL为首的一些电视厂家也发布提出了一个“女娲计划”，主要目标是以电视为核心来监控家中所有电器的使用。这两个计划的实践结果都证明了两种方式均存在着不足。其实电脑、电话、电视它们各有所长、各有利弊，相互互补、渗透共存。在某些发达国家的大城市中，应用于智慧社区综合管理的产品已逐步形成了一个系列产品。美国、意大利、日本等国的产品，相当一大部分产品系列已经完美通过了国际质量认证，在国际上此类产品的研发占有了很大优势[4]。目前国外智慧社区的发展已经拥有了很高的水平，注重生态保护、环保节能和环境保护，充分体现了以人为本的研发思想[5]。

在国内，虽然智慧社区的智能化程度和电子化程度没有国外的高，但智慧社区综合管理平台也正在逐渐普及，我国智慧社区综合管理建设水平仍然处于初级水平，存在着一系列难以解决的困难和问题。比如社区中的基础设施配置完成度参差不齐，社区中的综合服务管理平台的内容缺乏，其应用尚未形成具体的规模；社区治理能力低下且社区治理者的职能水平亟待进一步加强完善，公共的娱乐服务项目过少且不便于使用和修理；社区房屋以及物业管理服务水平层次不高，社区自治能力尚未充分得到挖掘和提高；整个社区缺乏统筹合理的社区规划，体制和机制建设不够顺畅，智慧社区相关技术人才队伍欠缺足够的生命力和活力，可持续的运营模式还没有完全形成；便民利民服务领域未能大面积推广[6]。智慧社区作为智慧城市建设的核心环节，智慧社区的建设具有见效快、可行性强、安全可靠的特征，智慧社区还可以增强社区居民对于智慧城市建设的认可度和社会接受度，为智慧城市建设的发扬和普及添砖加瓦且增光添彩。

* 1. 智慧社区综合管理平台的基本特征

智慧社区综合管理平台面向广大的社会公众，它有其很多独有的特点。手持终端用户还可以通过该平台更快更准确地了解智慧社区内的各种服务情况，更加便捷地使用智慧社区为社区居民所提供的各种服务[7]。该平台可以连接所有的社区居民和社区工作者，以社区中的所有居民为服务的核心，为居民提供便捷、高效率、安全保障的智慧化服务，从而全面满足所有居民的生活需要。通过智慧社区管理平台不仅能打开社区高效管理的缺口，有效提高社区服务的使用效率，还能为人们的日常生活提供帮助，使生活变得更加便捷、高效和轻松。

* 1. 智慧社区综合管理平台的开发意义

在当前社会上，许多的各种类型的社区管理系统纷纷建立，可以很大程度上地解决人们的社区服务管理问题，而智慧社区综合管理平台不仅能提供新型的社区服务管理模式，更好地解决服务难问题，还加入了社区问题模块，功能更加完善。智慧社区是当今时代社区管理的一种最新理念，是当今新形势下社区服务管理创新的一种最新模式。[8]智慧社区是充分利用物联网、互联网、云计算等现代化新兴信息技术的集成产品应用,为智慧社区居民提供一个更加安全、舒适、便捷的现代化、智慧化的日常生活娱乐环境，从而形成了一种基于信息化、现代化、智能化社区服务管理与配备的新的创新管理形态的智慧社区。智慧社区综合管理平台有着巨大的长远的发展空间，它为智慧社区的居民们提供了更好的更便捷的服务，为社区居民的生活带来了更多的便利。

# 第二章 开发环境

2.1 开发环境

本系统中采用以下运行平台：

采用Windows 10专业版 和移动端华为荣耀6手机。

1. 数据库平台：My SQL

MySQL是一个在日常开发中最常使用的关系型数据库，这个数据库系统具有高度的可移植性，并且由于其体积小，速度快，总拥有成本低，特别是开源，在开发中小型站点时，一般选择MySQL。[11]

1. 开发工具：IDEA

IDEA 全称 IntelliJ IDEA，一个用于开发Java编程语言的集成环境。IntelliJ在业界被公认为最好的java开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE支持、各类版本工具等方面的功能可称得上顶尖水平。

1. 开发工具：HbuilderX

HBuilderX是[DCloud](https://baike.baidu.com/item/DCloud" \t "https://baike.baidu.com/item/HBuilder/_blank)（[数字天堂](https://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E5%A4%A9%E5%A0%82/244387)）推出的一款支持[HTML5](https://baike.baidu.com/item/HTML5)的[Web](https://baike.baidu.com/item/Web/150564" \t "https://baike.baidu.com/item/HBuilder/_blank)开发[IDE](https://baike.baidu.com/item/IDE/8232086" \t "https://baike.baidu.com/item/HBuilder/_blank)。HBuilderX的编写用到了[Java](https://baike.baidu.com/item/Java/85979" \t "https://baike.baidu.com/item/HBuilder/_blank)、[C](https://baike.baidu.com/item/C/7252092" \t "https://baike.baidu.com/item/HBuilder/_blank)、Web和[Ruby](https://baike.baidu.com/item/Ruby/11419" \t "https://baike.baidu.com/item/HBuilder/_blank)。HBuilderX[本身](https://baike.baidu.com/item/%E6%9C%AC%E8%BA%AB/126627" \t "https://baike.baidu.com/item/HBuilder/_blank)主体是由Java编写。HBuilderX[本身](https://baike.baidu.com/item/%E6%9C%AC%E8%BA%AB/126627" \t "https://baike.baidu.com/item/HBuilder/_blank)主体是由Java编写。

2.2 相关技术

### 2.2.1 所使用的语言

1. Java语言

Java是一种面向对象编程语言，它具有很多其他语言所不具备的有点，而这就是我们使用它来完成项目开发的理由，它是一种简单、面向对象、分布式、解释性、健壮、可移植、高性能、多线程和动态的语言。[13]

1. Javascirpt语言

JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。最初用于HTML网页（标准通用标记语言下的应用），以向HTML页面添加动态功能[10]

### 2.2.2 所涉及的主要技术与框架

1. AJAX技术

AJAX 是一种创建动态快速网页的技术。通过在后台与系统服务器进行少量数据交换，AJAX 可以使网页实现异步更新，这可以在不重载整个网页的情况下，对网页的某个部分进行实时更新[11]。使用JavaScript将所有的东西绑定在一起，使用SOAP以XML的格式来传送方法名和方法参数[12]。

2. Spring+SpringMVC+Mybatics框架

Spring框架是由于软件开发的复杂性而创建的。Spring使用的是基本的JavaBean来完成以前只可能由EJB完成的事情[13]。

Spring MVC是Spring WEB模块的一个应用框架，实现程序MVC模式的分层定制[14]。

MyBatis 是一款优秀的持久层框架，使用Mybatis技术，只需要自己提供SQL语句，其他的工作，诸如建立连接，JDBC相关异常处理等等都交给Mybatis去做[15]。

# 需求分析

# 3.1 可行性分析

3.1.1 经济可行性

当今时代，科学技术飞速发展，应用到了生活中的方方面面，甚至包括我们所生活的城市和社区。智慧社区在我们日常生活中的应用和推广水平还停留在初期。之所以开发“智慧社区综合管理平台”，是为了方便社区工作者对智慧社区内各种服务和业务的管理，符合项目开发的经济可行性分析。

3.1.2 技术可行性

从技术上来说，通过使用主流开发语言（Java）和工具(IDEA+HBuilder)来开发此系统。项目后台主要使用Java语言来实现，框架使用主流的Spring+SpringMVC+Mybatis框架，前台使用JSP技术。本项目的开发风险处于安全范围内，适合现代大部分智慧社区的投入使用。[15]

3.1.3 社会可行性

在法律上来说，本项目的开发完全不涉及法律上的违反，智慧社区内的业主信息不会向外部人员所透露，并提供相对应的安全保障，因此具有法律可行性。在特定条件下，该系统可以适应当前社会发展，并且能为使用者提供更方便、快捷的环境来进行业务操作。

3.1.4 可行性分析总结

本项目的开发，从基本上解决了智慧社区居民以及社区工作者进行服务管理和申请的问题，解决了传统的人工服务方式的弊端，社区工作者可以更好地管理智慧社区中各个服务的查看和修改，解决智慧社区居民的生活便捷问题。通过以上分析，在所有可行性基本达标的情况下，认定该项目可以开发。

# 3.2 系统总体规划

3.2.1 功能模块划分

本系统主要分为以下功能系统：政务管理子系统、物业管理子系统、医疗管理子系统、养老管理子系统、业主APP后台管理子系统、业主APP、物业APP、社工APP。我负责的是业主APP系统，所以不进行其余模块的详细说明。本次需求分析得出的社工APP系统的子功能模块划分如图3.1所示



图3.1 业主APP系统的子模块图

3.2.2 业主APP系统模块

业主APP系统模块有政务公开、医疗信息、养老信息、志愿活动、法律信息、社会组织、招工就业、失物招领、物业、二手交易十个功能。

每名业主登录APP之后可以通过政务公开模块查看所在社区的政务公开信息。

每名业主登录APP之后可以通过医疗信息模块查看就近的医院信息，包括地址、电话等信息。

每名业主登录APP之后可以通过养老信息模块查看就近的养老院的信息。

每名业主登录APP之后可以通过志愿活动模块查看所在社区发布的志愿活动，包括活动时间、地点以及联系电话等。

每名业主登录APP之后可以通过法律信息模块查看律师事务所得相关信息，包括地址、联系电话等。

每名业主登录APP之后可以通过社会组织模块查看社区附近的社会组织信息。

每名业主登录APP之后可以通过招工就业模块查看平台上发布的招工信息。

每名业主登录APP之后可以通过失物招领模块查看所在社区发布的失物招领信息，同时能在平台上发布失物招领信息。

每名业主登录APP之后可以通过物业模块查看物业的收费标准，进行故障报修、投诉建议。

每名业主登录APP之后可以通过二手交易模块查看所在社区发布的二手交易信息，同时可以发布二手交易信息。

# 3.3 系统模块详解

3.3.1 业主APP系统模块数据流图

业主APP系统模块包括政务公开、医疗信息、养老信息、志愿活动、法律信息、社会组织、招工就业、失物招领、物业、二手交易十个模块，该模块的数据流图如图3.2所示：



图3.2 业主APP系统总模块数据流图

3.3.2 社会组织模块

### 3.3.2.1 数据流图

业主查询所在社区所有的社会组织详细信息，根据分析可以得出该模块的数据流图，如图3.7所示：



图3.3 社会组织模块数据流图

### 3.3.2.2 数据字典

1. 数据流描述

社会组织信息数据流如表3.1所示

表3.1 社会组织信息数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 社会组织模块 | 编号 | F6 |
| 名称 | 社会组织 | | |
| 来源：社会组织表 | | | |
| 去处：显示给业主查看 | | | |
| 数据流结构：社会组织信息={社会组织ID，社会组织名称，社会组织职责，社会组织详细信息，社区id } | | | |
| 简要说明：社会组织ID为主键，社区id为外键 | | | |

2. 数据存储描述

社会组织信息表数据存储如表3.2所示：

表3.2 社会组织信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 社会组织模块 | 编号 | D6 |
| 名称 | 社会组织表 | | |
| 组织形式：按社会组织ID顺序排列 | | | |
| 关键字：社会组织ID | | | |
| 记录组成：{社会组织信息} | | | |
| 简要说明：存放所有社会组织信息 | | | |

3. 数据处理描述

查看社会组织信息数据处理如表3.3所示：

表3.3 查看法律信息数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 社会组织模块 | 编号 | P6 |
| 名称 | 社会组织 | | |
| 输入：社区信息 | | | |
| 输出：此社区的所有社会组织详细信息 | | | |
| 处理过程：业主在APP首页点击社会组织，系统根据业主所在的社区，显示出社区中所有的社会组织详细信息。 | | | |

3.3.3 招工就业模块

### 3.3.3.1 数据流图

业主查询所所有的招工就业简介以及详细信息，根据分析可以得出该模块的数据流图，如图3.4所示：



图3.4 招工就业模块数据流图

### 3.3.3.2 数据字典

1. 数据流描述

招工就业信息数据流如表3.4所示

表3.4 招工就业信息数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 招工就业模块 | 编号 | F7 |
| 名称 | 招工就业 | | |
| 来源：职位表、公司表 | | | |
| 去处：显示给业主查看 | | | |
| 数据流结构：职位表={职位ID，职位名称，职位要求，职位薪资，公司ID }  公司表={公司ID，公司名称，公司简介，公司标签 } | | | |
| 简要说明：职位ID为主键，公司Id为主键 职位表中公司ID为外键 | | | |

2. 数据存储描述

招工就业信息表数据存储如表3.5、表3.6所示：

表3.5 招工就业信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 招工就业模块 | 编号 | D7 |
| 名称 | 职位表 | | |
| 组织形式：按职位ID顺序排列 | | | |
| 关键字：职位ID | | | |
| 记录组成：{职位信息} | | | |
| 简要说明：存放所有招工就业信息 | | | |

表3.6 招工就业信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 招工就业模块 | 编号 | D8 |
| 名称 | 公司表 | | |
| 组织形式：按公司ID顺序排列 | | | |
| 关键字：公司ID | | | |
| 记录组成：{公司信息} | | | |
| 简要说明：存放所有公司信息 | | | |

3. 数据处理描述

查看社会组织信息数据处理如表3.7所示：

表3.7 查看招工就业信息数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 招工就业模块 | 编号 | P7 |
| 名称 | 招工就业 | | |
| 输入：社区信息 | | | |
| 输出：所有的招工就业信息 | | | |
| 处理过程：业主在APP首页点击招工信息，系统在APP页面显示出所有的招工就业简介，业主进一步点击进入招工信息详情页，查看招工的详细信息。 | | | |

3.3.4失物招领模块

### 3.3.4.1 数据流图

业主查询社区发布的所有失物招领详细信息，并且能上传失物招领详细信息，根据分析可以得出该模块的数据流图，如图3.10所示：



图3.5 失物招领模块数据流图

### 3.3.4.2 数据字典

1. 数据流描述

失物招领信息数据流如表3.8所示

表3.8 失物招领信息数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 失物招领模块 | 编号 | F8 |
| 名称 | 失物招领 | | |
| 来源：失物招领表 | | | |
| 去处：显示给业主查看 | | | |
| 数据流结构：失物招领表={失物招领ID，失物招领详情，业主ID，社区ID } | | | |
| 简要说明：失物招领ID为主键，业主ID为外键，社区ID为外键 | | | |

2. 数据存储描述

失物招领信息表数据存储如表3.9所示：

表3.9 招工就业信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 失物招领模块 | 编号 | D9 |
| 名称 | 失物招领 | | |
| 组织形式：按失物招领ID顺序排列 | | | |
| 关键字：失物招领ID | | | |
| 记录组成：{失物招领信息} | | | |
| 简要说明：存放所有失物招领信息 | | | |

3. 数据处理描述

查看失物招领信息数据处理如表3.10所示：

表3.10 查看失物招领信息数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 失物招领模块 | 编号 | P8 |
| 名称 | 失物招领 | | |
| 输入：社区信息 | | | |
| 输出：所有的失物招领信息 | | | |
| 处理过程：业主在APP首页点击失物招领，系统在APP页面显示出业主所在社区所有的失物招领信息，在发布页面可以发布失物招领信息。 | | | |

3.3.5物业模块

### 3.3.5.1 数据流图

业主在物业模块，可以进行故障报修、投诉建议操作，根据分析可以得出该模块的数据流图，如图3.6所示：



图3.6 物业模块数据流图

### 3.3.5.2 数据字典

1. 数据流描述

故障报修信息数据流如表3.11所示

表3.11 故障报修信息数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 物业模块 | 编号 | F9 |
| 名称 | 故障报修 | | |
| 来源：业主填写 | | | |
| 去处：显示给物业管理人员查看 | | | |
| 数据流结构：故障报修表={故障报修ID，故障报修详情，业主ID，社区ID } | | | |
| 简要说明：故障报修ID为主键，业主ID为外键，社区ID为外键 | | | |

投诉建议信息数据流如表3.12所示

表3.12 投诉建议信息数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 物业模块 | 编号 | F9 |
| 名称 | 投诉建议 | | |
| 来源：业主填写 | | | |
| 去处：显示给物业管理人员查看 | | | |
| 数据流结构：投诉建议表={投诉建议ID，投诉建议详情，业主ID，社区ID } | | | |
| 简要说明：投诉建议ID为主键，业主ID为外键，社区ID为外键 | | | |

2. 数据存储描述

故障报修信息表数据存储如表3.13所示：

表3.13 故障报修信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 物业模块 | 编号 | D10 |
| 名称 | 故障报修 | | |
| 组织形式：按故障报修ID顺序排列 | | | |
| 关键字：故障报修ID | | | |
| 记录组成：{故障报修信息} | | | |
| 简要说明：存放所有故障报修信息 | | | |

投诉建议信息表数据存储如表3.14所示：

表3.14 投诉建议信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 物业模块 | 编号 | D11 |
| 名称 | 投诉建议 | | |
| 组织形式：按投诉建议ID顺序排列 | | | |
| 关键字：投诉建议ID | | | |
| 记录组成：{投诉建议信息} | | | |
| 说明：存放所有投诉建议信息 | | | |

3. 数据处理描述

故障保修信息数据处理如表3.15所示：

表3.15 故障报修信息数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 物业模块 | 编号 | P9 |
| 名称 | 故障报修 | | |
| 输入：故障报修信息 | | | |
| 输出：无 | | | |
| 处理过程：业主在故障报修页面填写报修信息。 | | | |

投诉建议信息数据处理如表3.16所示：

表3.16 投诉建议信息数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 物业模块 | 编号 | P10 |
| 名称 | 投诉建议 | | |
| 输入：投诉建议 | | | |
| 输出：无 | | | |
| 处理过程：业主在投诉建议页面填写投诉建议信息。 | | | |

3.3.6二手交易模块

### 3.3.6.1 数据流图

业主在二手交易模块，可以查看社区中发布的二手交易信息，同时可以发布二手交易信息。根据分析可以得出该模块的数据流图，如图3.7所示：



图3.7 二手交易模块数据流图

### 3.3.6.2 数据字典

1. 数据流描述

二手交易信息数据流如表3.17所示

表3.17 二手交易信息数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 二手交易模块 | 编号 | F10 |
| 名称 | 二手交易 | | |
| 来源：业主填写 | | | |
| 去处：显示给社区中的业主查看 | | | |
| 数据流结构：商品表={商品ID，商品名称，商品详情，商品价格，社区ID，业主ID } | | | |
| 说明：商品ID为主键，业主ID为外键，社区ID为外键 | | | |

2. 数据存储描述

二手交易信息表数据存储如表3.18所示：

表3.18 二手交易信息表数据存储

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 二手交易模块 | 编号 | D10 |
| 名称 | 二手交易 | | |
| 组织形式：按商品ID顺序排列 | | | |
| 关键字：商品ID | | | |
| 记录组成：{商品信息} | | | |
| 说明：存放所有商品信息 | | | |

3. 数据处理描述

二手交易信息数据处理如表3.19所示：

表3.19二手交易信息数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 系统名称 | 二手交易模块 | 编号 | P10 |
| 名称 | 二手交易 | | |
| 输入：商品信息 | | | |
| 输出：商品信息 | | | |
| 处理过程：业主在二手交易页面查看商品信息，在发布界面发布商品 | | | |

# 第四章 概要设计

在概要设计这个阶段，需要根据用户与系统的交互过程，并与用户的需求情况相结合，从而形成交互框架以及视觉框架。主要通过用例图与用例描述来实现这一设计过程。

### 4.1 系统功能设计

本次设计中我主要讲述的是业主APP模块，所以不在进行其余模块的讲述。业主模块主要有政务公开、医疗信息、养老信息、志愿活动、法律信息、社会组织、招工就业、失物招领、物业、二手交易十个功能。

### 4.1.1 查看政务公开信息功能

查看政务公开功能主要是为业主查看所在社区发布的政务公开信息所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看政务公开信息：

业主通过验证后，可以点击“政务公开”按钮来查看自己所在社区的政务公开详细信息。

### 4.1.2 查看医疗信息功能

查看医疗信息功能主要是为业主查看所在社区附近的医院信息所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看医疗信息：

业主通过验证后，可以点击“医疗信息”按钮来查看医院的详细信息。

### 4.1.3 查看养老信息功能

查看养老信息功能主要是为业主查看养老机构所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看养老信息：

业主通过验证后，可以点击“养老信息”按钮来查看养老机构的详细信息。

### 4.1.4 查看志愿活动信息功能

查看志愿活动功能主要是为业主查看所在社区发布的志愿活动信息所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看志愿活动信息：

业主通过验证后，可以点击“志愿活动”按钮来查看自己所在社区的志愿活动详细信息。

### 4.1.5 查看法律信息功能

查看法律信息功能主要是为业主查看律师事务所信息所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看法律信息：

业主通过验证后，可以点击“法律信息”按钮来查看律师事务所详细信息。

### 4.1.6 查看社会组织信息功能

查看社会组织功能主要是为业主查看所在社区的社会组织信息所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看社会组织信息：

业主通过验证后，可以点击“社会组织”按钮来查看自己所在社区的社会组织详细信息。

### 4.1.7 查看招工就业信息功能

查看招工就业功能主要是为业主查看招工就业信息所设计的功能模块，该模块的过程如下：

1. 业主登录：

业主需要输入自己的通行证（用户名密码）来验证身份。

2. 查看招工就业信息：

业主通过验证后，可以点击“招工就业按钮来查看招工就业详细信息。

### 4.1.8 失物招领功能

查看失物招领功能主要是为业主查看失物招领信息、发布招领信息所设计的功能模块，该模块的过程如下：

1. 业主查看招领信息：

业主查看自己社区发布的招领信息。

2. 发布招领信息：

业主在发布页面发布招领信息。

### 4.1.9 物业功能

物业功能主要是为业主故障保修、投诉建议所设计的功能模块，该模块的过程如下：

1. 业主报修故障：

业主在报修页面上报故障信息。

2. 业主投诉建议：

业主在投诉页面发表投诉建议。

### 4.1.10 二手交易功能

二手交易功能主要是为业主查看二手交易信息、发布二手交易信息所设计的功能模块，该模块的过程如下：

1. 业主查看二手交易信息：

业主在主页面查看所在小区发布的二手交易信息。

2. 业主发布二手交易信息：

业主发布页面发布二手交易信息。

### 4.2 系统用例分析

### 4.2.1 业主APP总功能用例

业主主要有查看政务公开、医疗信息、养老信息、志愿活动、法律信息、社会组织、招工就业、失物招领、物业、二手交易十个功能，该模块的总用例图如图4.1所示：



图4.1 业主APP功能总用例图

政务公开信息功能用例规约描述如表4.1所示：

表4.1 政务公开功能用例规约描述

|  |
| --- |
| 用例名称:政务公开  主要参与者：业主  基本流程：1. 登录  2. 查看社区政务公开详情  具体流程：1. 登录页面--进入主页面  2. 选择政务公开--进入政务公开信息详情页面 |

医疗信息功能用例描述如表4.2所示：

表4.2 医疗信息功能用例规约描述

|  |
| --- |
| 用例名称:医疗信息  主要参与者：业主  基本流程：1. 登录  2. 查看医疗信息详情  具体流程：1. 登录页面--进入主页面  2. 选择查看医疗信息--进入医疗信息详情页面 |

养老信息功能用例规约描述如表4.3所示：

表4.3 养老信息功能用例规约描述

|  |
| --- |
| 用例名称:养老信息  主要参与者：业主  基本流程：1. 登录  2. 查看养老信息详情  具体流程：1. 登录页面--进入主页面  2. 选择查看养老信息--进入养老信息详情页面 |

志愿活动活动信息功能用例规约描述如表4.4所示：

表4.4 志愿活动信息功能用例规约描述

|  |
| --- |
| 用例名称:志愿活动  主要参与者：业主  基本流程：1. 登录   1. 查看社区志愿活动详情   具体流程：1. 登录页面--进入主页面   1. 查看社区志愿活动信息--社区志愿活动信息界面 |

法律信息功能用例规约描述如表4.5所示：

表4.5 法律信息功能用例规约描述

|  |
| --- |
| 用例名称:法律信息  主要参与者：业主  基本流程：1. 登录  2. 查看法律信息详情  具体流程：1. 登录页面--进入主页面  2. 选择查看法律信息--进入法律信息详情页面 |

社会组织功能用例规约描述如表4.6所示：

表4.6 社会组织功能用例规约描述

|  |
| --- |
| 用例名称:社会组织  主要参与者：业主  基本流程：1. 登录  2. 查看社区社会组织详情  具体流程：1. 登录页面--进入主页面  2. 选择查看社区社会组织详情问题--进入社区社会组织详情页面 |

招工就业功能用例规约描述如表4.7所示：

表4.7 招工就业功能用例规约描述

|  |
| --- |
| 用例名称:招工就业  主要参与者：业主  基本流程：1. 登录  2. 查看招工就业信息详情  具体流程：1. 登录页面--进入主页面  2. 选择招工就业--进入招工就业详情页 |

失物招领功能用例规约描述如表4.8所示：

表4.8 失物招领功能用例规约描述

|  |
| --- |
| 用例名称:失物招领  主要参与者：业主  基本流程：1. 查看招领信息  2. 发布招领信息  具体流程：1. 点击失物招领，查看招领信息  2. 进入发布页面，发布招领信息 |

二手交易功能用例规约描述如表4.9所示：

表4.9 二手交易功能用例规约描述

|  |
| --- |
| 用例名称:二手交易  主要参与者：业主  基本流程：1. 发布二手商品  具体流程：1. 查看页面—查看社区中发布的二手商品  2. 管理页面—发布和管理自己的二手商品 |

### 4.3 数据库设计

### 4.3.1 概念结构设计

在需求分析阶段得到的应用需求首先应抽象为信息结构，然后才能进行更改，从而更准确地用某一数据库管理系统来实现这些需求。

E-R图提供了表示实体类型、属性和联系的方法，每个项目都需要整体E-R图来表示实体之间的不同关系如图4.2所示：



图4.2 整体E-R图

### 4.3.2 逻辑结构设计

我负责的业主APP总模块具体的表有社区表（Community）、业主信息表（user）、政务信息表（government）、医院信息表（medical）、养老机构表（old）、志愿活动表（volunteer）、法律机构表（law）、社会组织表（organization）、招工信息表（job）公司表（company）公司标签表（label）、失物招领表（lost）、投诉建议表（advice）、故障保修表（fault）、商品表（good）。

1. 社区表（社区ID，社区名称）

主键：社区ID

2. 业主信息表（业主ID，登录密码，业主电话，业主名称，社区ID）

主键：业主ID

外键：社区ID

3. 政务信息表（政务信息ID，标题，详情，社区ID）

主键：政务信息ID

外键：社区ID

4. 医院信息表（医院ID，医院名称，医院地址，路线，电话）

主键：医院ID

5. 养老机构表（机构ID，机构名称，机构服务项目，机构宣言，社区ID）

主键：机构ID

外键：社区ID

6. 志愿活动表（活动ID，活动名称，活动时间，活动地点，联系电话，社区ID）

主键：活动ID

外键：社区ID

7. 法律机构表（机构ID，机构名称，机构地址，机构详情，机构联系电话）

主键：机构ID

外键：无

8. 社会组织表（组织ID，组织名称，组织介绍，组织职责，组织联系方式，社区ID）

主键：组织ID

外键：社区ID

9. 招工信息表（招工信息ID，职位名称，薪资，职位详情，公司ID， 面试时间，面试地点，联系方式）

主键：招工信息ID

外键：公司ID

10. 公司表（公司ID，公司名称，公司详情，公司logo）

主键：公司ID

外键：无

11. 公司标签表（标签ID，标签名，公司ID）

主键：标签ID

外键：公司ID

12. 失物招领表（招领ID，招领详情，业主ID，社区ID）

主键：招领ID

外键：业主ID、社区ID

13. 投诉建议表（投诉建议ID，标题，详情，联系电话，社区ID）

主键：投诉建议ID

外键：社区ID

14. 故障报修表（报修ID，故障详情，联系地址，联系电话，社区ID）

主键：报修ID

外键：社区ID

15. 二手商品表（商品ID，商品名称，商品价格，商品详情，商品照片，商品类别，业主ID，社区ID）

主键：商品ID

外键：社区ID、业主ID

### 4.3.3 物理结构设计

1. 社区信息表（community）

社区信息表如表4.10所示：

表4.10 社区信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| communityId | int(11) | int |  | NO | 社区ID |
| communityName | varchar(255) | varchar | 255 | YES | 社区名 |

说明：社区ID为社区信息表的主键。

1. 业主信息表（user）

业主信息表如表4.11所示：

表4.11 业主信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| userId | int(11) | int |  | NO | 用户ID |
| userPassword | varchar(255) | varchar | 255 | YES | 密码 |
| userPhone | varchar(255) | varchar | 255 | YES | 电话 |
| username | varchar(255) | varchar | 255 | YES | 用户名 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：业主ID为业主信息表的主键，社区ID作为外键。

1. 政务信息表（government）

政务信息表如表4.12所示：

表4.12 政务信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| governmentId | int(11) | int |  | NO | 信息ID |
| governmentTitle | varchar(255) | varchar | 255 | YES | 标题 |
| governmentDetail | varchar(255) | varchar | 255 | YES | 详情 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：信息ID为政务信息表的主键，社区ID作为外键。

1. 医院信息表（medical）

政务信息表如表4.13所示：

表4.13 医院信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| medicalId | int(11) | int |  | NO | 医院ID |
| medicalName | varchar(255) | varchar | 255 | YES | 医院名称 |
| medicalCall | varchar(255) | varchar(255) | 255 | YES | 联系电话 |
| medicalAddressl | varchar(255) | varchar | 255 | YES | 地址 |
| medicalRoad | int(11) | int |  | YES | 路线 |

说明：医院ID为医院信息表的主键。

1. 养老机构表（old）

政务信息表如表4.14所示：

表4.14 养老机构表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| oldId | int(11) | int |  | NO | 机构ID |
| oldName | varchar(255) | varchar | 255 | YES | 机构名称 |
| oldItem | varchar(255) | varchar | 255 | YES | 服务项目 |
| oldDeclarer | varchar(255) | varchar | 255 | YES | 机构宣言 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：机构ID为养老机构表的主键，社区ID为外键。

1. 志愿活动表（volunteer）

志愿活动表如表4.15所示：

表4.15 志愿活动表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| volunteer Id | int(11) | int |  | NO | 活动ID |
| volunteerName | varchar(255) | varchar | 255 | YES | 活动名称 |
| volunteerTime | varchar(255) | varchar | 255 | YES | 活动时间 |
| volunteerCall | varchar(255) | varchar | 255 | YES | 联系电话 |
| volunteerPlace | varchar(255) | varchar | 255 | YES | 活动地点 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：活动ID为志愿活动表的主键，社区ID为外键。

1. 法律机构表（law）

法律机构表如表4.16所示：

表4.16 法律机构表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| lawId | int(11) | int |  | NO | 机构ID |
| law Name | varchar(255) | varchar | 255 | YES | 机构名称 |
| lawAddress | varchar(255) | varchar | 255 | YES | 机构地址 |
| lawDetail | varchar(255) | varchar | 255 | YES | 机构详情 |
| lawCall | varchar(255) | varchar | 255 | YES | 联系电话 |

说明：机构ID为法律机构表的主键。

1. 社会组织表（organization）

社会组织表如表4.17所示：

表4.17 社会组织表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| organizationId | int(11) | int |  | NO | 组织ID |
| organizationName | varchar(255) | varchar | 255 | YES | 组织名称 |
| organizationDetail | varchar(255) | varchar | 255 | YES | 组织详情 |
| organizationIntrodue | varchar(255) | varchar | 255 | YES | 组织简介 |
| organizationCall | varchar(255) | varchar | 255 | YES | 联系方式 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：组织ID为社会组织表的主键，社区ID为外键。

1. 招工信息表（job）

招工信息表如表4.18所示：

表4.18 招工信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| jobId | int(11) | int |  | NO | 招工ID |
| jobName | varchar(255) | varchar | 255 | YES | 职位名称 |
| jobSalary | varchar(255) | varchar | 255 | YES | 职位薪资 |
| jobDetail | varchar(255) | varchar | 255 | YES | 职位详情 |
| companyId | int(11) | int |  | YES | 公司ID |
| jobInterviewTime | varchar(255) | varchar | 255 | YES | 面试时间 |
| jobInterviewPlace | varchar(255) | varchar | 255 | YES | 面试地点 |
| jobCall | varchar(255) | varchar | 255 | YES | 联系电话 |

说明：招工ID为招工信息表的主键，公司ID为外键。

1. 公司信息表（Company）

公司信息表如表4.19所示：

表4.19 公司信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| companyId | int(11) | int |  | NO | 公司ID |
| companyDetail | varchar(255) | varchar | 255 | YES | 公司详情 |
| companyName | varchar(255) | varchar | 255 | YES | 公司名称 |
| companyLogo | varchar(255) | varchar | 255 | YES | 公司logo |

说明：公司ID为公司信息表的主键。

1. 失物招领表（lost）

失物招领表如表4.20所示：

表4.20 失物招领表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| lostId | int(11) | int |  | NO | 招领ID |
| lostDetail | varchar(255) | varchar | 255 | YES | 招领详情 |
| userId | int(11) | int |  | YES | 业主ID |
| communityId | int(11) | int |  | YES | 社区ID |

说明：招领ID为失物招领表的主键，业主ID、社区ID为外键。

1. 投诉建议表（advice）

投诉建议表如表4.21所示：

表4.21 投诉建议表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| adviceId | int(11) | int |  | NO | 投诉ID |
| adviceTitle | varchar(255) | varchar | 255 | YES | 投诉标题 |
| adviceDetail | varchar(255) | varchar | 255 | YES | 投诉详情 |
| adviceCall | varchar(255) | varchar | 255 | YES | 联系电话 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：投诉ID为投诉建议表的主键，社区ID为外键。

1. 故障报修表（fault）

故障报修表如表4.22：

表4.22 故障保修表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| faultId | int(11) | int |  | NO | 报修ID |
| faultDetail | varchar(255) | varchar | 255 | YES | 报修详情 |
| faultAddress | varchar(255) | varchar | 255 | YES | 联系地址 |
| faultCall | varchar(255) | varchar | 255 | YES | 联系电话 |
| communityId | int(11) | int |  | YES | 社区ID |

说明：报修ID为故障报修表主键，社区ID为外键。

1. 商品表（good）

商品表如表4.23：

表4.23 商品表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段类型 | 长度 | 是否为空 | 备注 |
| goodId | int(11) | int |  | NO | 商品ID |
| userId | int(11) | int |  | YES | 业主ID |
| goodName | varchar(255) | varchar | 255 | YES | 商品名称 |
| gooddPrice | varchar(255) | varchar | 255 | YES | 商品价格 |
| goodDetail | varchar(255) | varchar | 255 | YES | 商品详情 |
| goodImg | varchar(255) | varchar | 255 | YES | 商品照片 |
| communityId | int(11) | int |  | YES | 社区ID |
| goodType | varchar(255) | varchar | 255 | YES | 商品类别 |

说明：商品ID为商品表主键，业主ID、社区ID为外键。

# 第五章 详细设计

详细设计是项目对概要设计的一个具体细化的设计过程，通过详细设计来为每个功能实现具体的实现代码和实现算法，从而设计出满足用户需求的合格的软件产品。

## 5.1 功能类图

功能类图使用户可以清晰地看到和了解每个类之间的关系，使类与类之间形成必要的联系。类中的方法和属性也全部显示在类图中，方便用户的查看和使用修改。

1. 模块总体类图

通过模块总体类图，我们可以清晰的了解各个模块之间的联系和交互。本模块的总体类图如下图5.1所示：



图5.1 模块总体类图

## 5.2 功能详细设计

时序图是一种典型的UML交互图。它通过描绘对象间发送消息的时间顺序来显示对象间的动态交互。

1. 查看政务公开信息

说明：业主在APP主页面点击查看政务公开信息，跳转到政务公开信息列表页，下一步进行查看详情操作，此功能的时序图如图5.2所示：



图5.2 政务公开信息时序图

此功能中，与功能相关的类有Government、GovernmentController.java、GovernmentServiceImpl.java，涉及到的接口有GovernmentService、GovernmentMapper，涉及到的方法有getAllGovernmentCommunityId()、getGovernmentByGovernmentId()。

2. 查看医疗信息

说明：业主主页面点击医疗信息，进入医疗信息详情页，查看医疗信息的时序图如图5.3所示：



图5.3 查看医疗信息时序图

此功能中，与功能相关的类有Medical、MedicalController.java、MedicalServiceImpl.java，涉及到的接口有MedicalService、MedicalMapper，涉及到的方法有getAllMedical ()。

3. 查看养老信息

说明：业主主页面点击养老信息，进入养老机构详情页，查看养老信息的时序图如图5.4所示：



图5.4 查看养老信息时序图

其中，涉及到的类有Old、OldController.java、OldServiceImpl.java，涉及到的接口有OldService、OldMapper，涉及到的方法有getAllOldByCommunityId ()。

4. 查看志愿活动信息

说明：业主在主页面点志愿活动，进入志愿活动详情页，查看志愿活动信息的时序图如图5.5所示：



图5.5 查看志愿活动信息时序图

其中，涉及到的类有volunteer、volunteerController.java、volunteerServiceImpl.java，涉及到的接口有volunteerService、volunteerMapper，涉及到的方法有getAllVolunteerByCommunityId ()。

5. 查看法律信息信息

说明：在主页面，业主点击法律信息，系统跳转页面显示律师事务所详情，法律信息功能对应的时序图如图5.6所示：



图5.6 查看法律信息时序图

其中，涉及到的类有Law、LawController.java、LawServiceImpl.java，涉及到的接口有LawService、LawMapper，涉及到的方法有getAllLaw ()。

6. 查看社会组织

说明：业主在主页面可以查看社区中的社会组织，查看社会组织的时序图如图5.7所示：



图5.7 查看社会组织信息时序图

其中，涉及到的类有Organization、OrganizationController.java、OrganizationServiceImpl.java，涉及到的接口有OrganizationService、OrganizationMapper，涉及到的方法有getAllOrganization ByCommunityId()。

7. 查看招工就业信息

说明：在招工就业页面，业主可以查看招工信息，进一步点击进入详情页面，查看招工就业信息的时序图如图5.8所示：



图5.8 查看招工就业信息时序图

其中，涉及到的类有Job、JobController.java、JobServiceImpl.java，涉及到的接口有JobService、JobMapper，涉及到的方法有getAllJobs()、getJobByJobId（）。

8. 失物招领

说明：在失物招领页面，业主查看本小区发布的失物招领信息，在发布页面发布失物招领信息，失物招领的时序图如图5.9所示：



图5.9 失物招领时序图

其中，涉及到的类有Lost、LostController.java、LostServiceImpl.java，涉及到的接口有LostService、LostMapper，涉及到的方法有getAllLost ByCommunityId()、releaseLost（）。

9. 物业模块

说明：在物业模块，业主可以在报修界面报修故障，在投诉建议界面进行投诉，物业模块的时序图如图5.10所示：



图5.10 物业模块时序图

其中，涉及到的类有Fault、Advice、PropertyController.java、PropertyServiceImpl.java，涉及到的接口有PropertyService、PropertyMapper，涉及到的方法有addAdvice（）、addFault（）。

8. 二手交易模块

说明收交易模块，业主可以查看本社区发布的二手交易信息，发布管理自己的商品，二手交易的时序图如图5.11所示：



图5.11 二手交易时序图

其中，涉及到的类有Good、GoodController.java、GoodServiceImpl.java，涉及到的接口有GoodService、GoodMapper，涉及到的方法有getGoodByCommunityId()、getGoodByUserId()、getGoodByGoodName()、getGoodByGoodId()、updateGoodByGoodId()、deleteGoodByGoodId()、lightGood()、getGoodByGoodType()、releaseGood()。

# 项目结构与功能实现

为了使项目的整体结构更加方便用户查看，所以项目结构要简洁有序。每个功能需要没有异常报告的成功实现，完成所有详细设计中提到的功能。

## 6.1 项目结构

业主APP系统的项目结构如下图6.1所示：

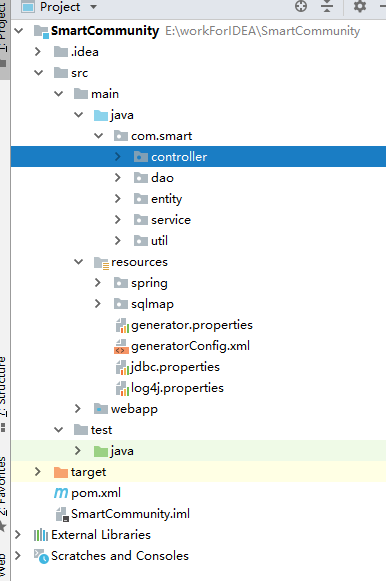


图6.1 项目目录结构图

## 6.2 功能实现

业主APP系统模块包括政务公开、医疗信息、养老信息、志愿活动、法律信息、社会组织、招工就业、失物招领、物业、二手交易十个功能，下面详细介绍业主APP系统模块每个功能的实现过程。本项目功能代码详见附录。

1. 查看政务公开信息

查看政务公开信息界面如图6.2、图6.3所示：



图6.2 查看政务公开信息界面

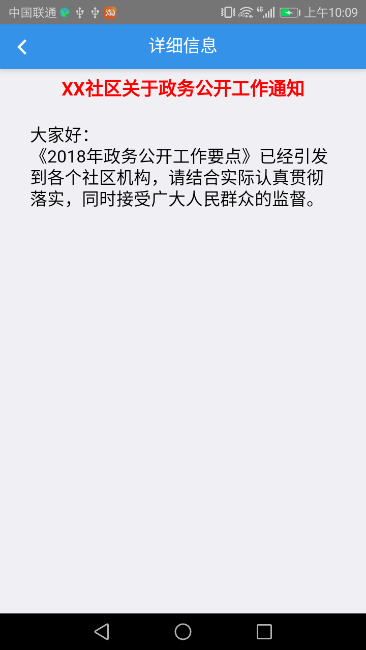


图6.3 查看政务公开信息界面

说明：在列表页查看政务公开信息标题，在详情页查看详细信息。

2. 查看医疗信息

查看医疗信息界面如图6.4所示：



图6.4 查看医疗信息界面

说明：在信息也查看医院的详细信息。

3. 查看养老信息

查看养老信息界面如图6.5所示：

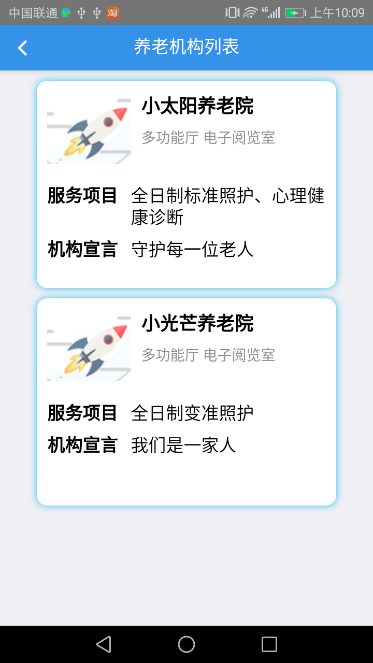


图6.5 查看养老信息界面

说明：在此界面查看养老机构的详细信息。

4. 查看志愿活动信息

查看志愿活动信息界面如图6.6所示：



图6.6 查看志愿活动界面

说明：在此页面查看志愿活动的详细。

5. 查看法律信息

查看法律信息界面如图6.7所示：



图6.7 查看法律信息界面

说明：在此页面查看法律机构的详细信息。

6. 查看社会组织

查看社会组织界面如图6.8所示：



图6.8 查看社会组织界面

说明：在此页面查看社会组织的详细信息。

7. 查看招工就业信息

查看招工就业界面如图6.9、6.10所示：



图6.8 查看招工就业界面



图6.10 查看招工就业界面

说明：在列表页查看招工简介、在详情页查看招工详细信息。

8. 失物招领

失物招领界面如图6.11、6.12所示：



图6.11 查看失物招领信息界面



图6.12 发布失物招领信息界面

说明：在列表页查看失物招领信息，在发布页面发布失物招领信息。

9. 物业

物业界面如图6.13、6.14、6.15、6.16所示：



图6.13 物业界面

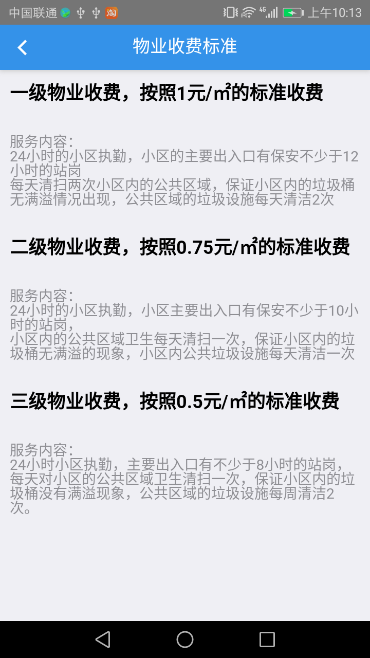


图6.13 收费标准界面

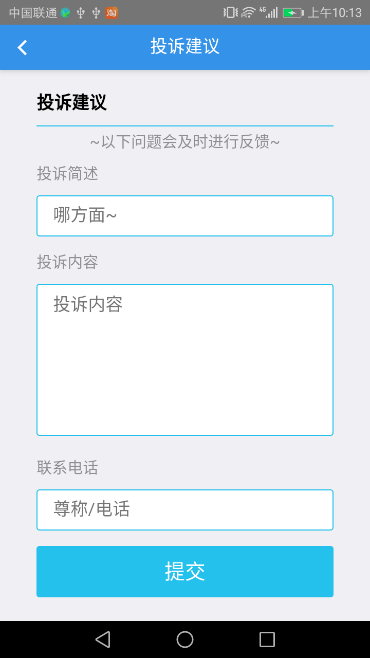


图6.14 投诉建议界面

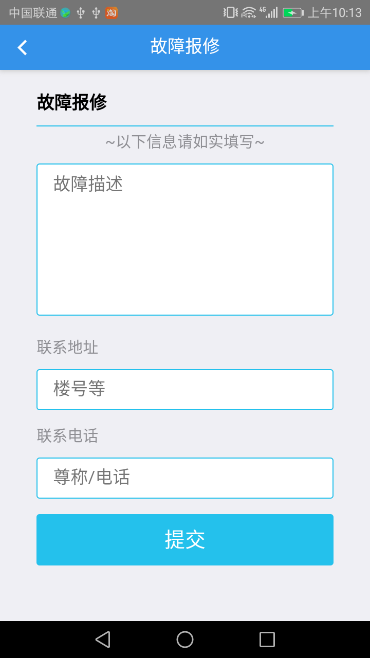


图6.15 故障报修界面

说明：在物业界面选择下一步操作，在收费标准页面查看物业收费标准，在投诉建议发布投诉建议，在故障保修界面进行故障报修。

10. 二手交易

二手交易界面如图6.17、6.18、6.19、6.20所示：



图6.17 二手交易界面



图6.18 我的二手界面



图6.19 发布商品界面



图6.20 商品详情界面

说明：二手交易页面查看社区中发布的二手交易信息，在商品详情页面查看商品详细信息，在我的二手界面管理自己发布的二手商品，在发布页面发布商品。

# 第七章 软件测试

系统的实现阶段和代码的编写必须将前一阶段的所有逻辑设计转换为易于使用的系统，这一阶段主要的任务是代码编写，将设计的功能全部实现

## 7.1 测试目标

软件测试，是一种用来促进鉴定软件的正确性、整体性和安全性的过程。在本项目中，主要应用了黑盒测试来进行测试，主要是对业主APP模块进行测试。

## 7.2 测试环境

具体测试环境如表7.1所示。

表7.1 测试环境表

|  |  |
| --- | --- |
| 硬件环境 | CPU：Intel(R) Core(TM) i5-5200U 2.20GHz  内存：8G  显卡：Intel(R) HD Graphics 5500 |
| 网络环境 | 100M |
| 软件环境 | 系统：windows 7  工具：HBuilder+IDEA+Navicat for MySQL  移动端：华为荣耀6 |

## 7.3 测试内容

本系统的测试，通过人工对系统进行随机操作，主要目的是查看是否存在处理异常或者操作结果是否是按照设计的初志来完成的，主要是对整个业主APP系统进行测试。

内容：社工进行查看网格信息、楼宇信息、社区活动等功能。

目的：通过测试来检查业主APP系统模块中各个功能是否能正常进行，是否能得出预期的结果。

表7.2详情展示测试结构表

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 测试用例 | 测试结果截图 | 测试结果 |
| 1 | 业主登录业主APP系统 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-140623.png C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-140612.png | 密码为空或用户名密码错误时无法进入 |
| 2 | 业主登录业主APP系统 |  | 用户名和密码正确时进入系统 |
| 3 | 点击医疗信息，查看医疗信息 |  | 正确显示 |
| 4 | 点击政务公开，查看政务公开信息 |  | 成功显示相关信息 |
| 5 | 点击养老信息，查看养老信息 |  | 正确显示相关信息 |
| 6 | 点击志愿活动，查看志愿活动信息 |  | 正确显示相关信息 |
| 7 | 点击法律信息，查看法律信息 |  | 正确显示相关信息 |
| 8 | 点击社会组织，查看社会组织信息 |  | 正确显示相关信息 |
| 9 | 点击招工就业，查看招工信息，同时进一步查看详情 |  | 正确显示相关信息 |
| 10 | 点击失物招领，查看失物招领信息，进入发布页面，发布信息 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-142300.png | 正确显示相关信息，发布成功之后给与争取提示 |
| 11 | 物业点击收费标准，查看收费标准信息 |  | 正确显示相关信息 |
| 12 | 进行故障报修操作 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-142615.png | 上传成功，并给与正确提示 |
| 13 | 进行投诉建议操作 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-142819.png | 上传成功 |
| 14 | 二手交易按类别查询 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-142951.png | 正确显示相关内容 |
| 15 | 我的二手，修改价格 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-143044.png C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-143048.png | 程序运行正确 |
| 16 | 发布商品 | C:\Users\刘贺\Documents\Tencent Files\1105942357\FileRecv\MobileFile\Screenshot_20190519-143318.png | 发布成功，并且能正确 |

# 第八章 项目总结

本次毕业设计从2018年11月发布任务之后，小组八位成员共同努力，在2019年五月之前最终完成项目，在这个过程中，我们遇到过无数的困难，经历过很多挫折，但是整个小组一起讨论、沟通，把所有问题一一解决，这其中当然也少不了老师的帮助，在后期优化阶段导师给出了非常好的优化方法。最终项目如期完成。

软件开发的过程是一个完整的过程，每一步都必不可少。只有我们按部就班，一步一个脚印，细心仔细的完成，才完成了智慧社区综合管理平台的开发结果。在本次系统开发中，我的数据库设计出现了一些纰漏，从而导致系统开发出现了障碍，但最终我重新将数据库设计了一遍，最后成功完成了整个系统的开发。

总而言之，在本次毕业设计的实现过程中，跟同学、老师学到了很多东西，学习了新的开发环境，学习了新的开发框架，甚至对代码优化有了一些了解与实际操作，这些经验都将有益于我日后的学习和日后的工作。

参考文献

1. 何志红.智慧城市平台中的大数据关键技术研究与应用[J].电子技术与软件工程,2019(07):184.
2. 侯文华,陈婷.智慧养老服务产业发展和实现路径研究[J].现代管理科学,2019(04):100-102.
3. 冯利,李海凌,王蕾.智慧城市发展理念导向下的智慧社区建设研究[J].四川建材,2019,45(04):37-38.
4. 钟达.构建智慧社区服务平台的思路与技术实现[J].科技经济导刊,2019,27(09):39.
5. 胡雄伟.智慧社区平台建设:用户体验是王道[J].城市开发,2019(06):18-19.
6. 戴晓虎,吴恒,巴勉.基于Web的智慧社区信息系统设计[J].计算机时代,2019(03):29-31+35.
7. 伊国强.以物业管理为核心的智慧社区解决方案[J].数字通信世界,2019(03):158.
8. 本刊综合.这些年,智慧社区走过的那些路[J].城市开发,2019(03):13-17.
9. 吴满华. 城市智慧社区发展的现状及发展对策研究——以诸暨市暨阳街道为例[A]. 中国教育发展战略学会教育教学创新专业委员会.2019全国教育教学创新与发展高端论坛论文集（卷二）[C].中国教育发展战略学会教育教学创新专业委员会:中国教育发展战略学会教育教学创新专业委员会,2019:3.
10. 柳霞,张发平.舟山建设智慧社区的对策思考[J].农村经济与科技,2018,29(23):249-251.
11. 洪艺萍.智慧城市视角下的社区电子商务[J/OL].黎明职业大学学报:1-9[2019-05-19].http://kns.cnki.net/kcms/detail/35.1212.G4.20181107.0941.002.html
12. 郭海智,冯玉松.基于Spring+SpringMVC+Mybatis的业余足球俱乐部管理系统[J].电脑知识与技术,2019,15(09):71-72+76.
13. 徐展宇.计算机软件开发的JAVA编程语言与实际应用[J].电子技术与软件工程,2019(09):44.
14. 李胜.基于JavaScript的面向对象语言特性及其程序设计[J].科技与创新,2019(06):112-113+116.
15. Traian Turc. AJAX Technology for Internet of Things[J]. Procedia Manufacturing,2019,32.