

PTcheck检测报告简明打印版

对比结果 (相似度) :

总相似度: 28% (相似字数占总字数的百分比)

红色相似度: 14% (句子相似度70%-100%的字数占总字数的百分比)

橙色相似度: 14% (句子相似度40%-70%的字数占总字数的百分比)

验真伪二维码



编号: a8eb0638-125a-4156-b263-e963e10a589a

标题: 9602180759585256

作者: 96021807595852561

长度: 20892字符 (不计空格)

段落: 357个

句子: 628句

词语: 10848个

时间: 2017-4-24 7:48:42

比对库: 学术期刊、学位论文、互联网资源

相似资源列表:

(1) 28%即5849字来源于其他网络

全文简明报告:

基于手机的小区物业服务APP的设计与开发

作者: 钱文军 指导教师: 韩冰

{62%: (海南师范大学信息科学技术学院软件工程系, 海口, xxxxx) }

{68%:摘要:随着我国房地产行业的不断发展,人们对住宅的要求日益增高。}人们已经不再满足于对于“数量型”的追求,而是更偏向于追求高质量的小区生活,更便利的功能和服务体验。物业服务作为一门学科,现在迫切的需要解决由传统的信息采集人工管理的方式,向智能化数据分析,自动化管理,人性化服务的方向发展。

{57%: 近些年来我国互联网和科学技术不断发展,互联网技术已经渗透到生活的方方面面。}尤其是移动互联网技术的蓬勃发展,使得手机已经从原来的单纯的通讯工具,发展成为搭载独立的操作系统,能够安装各种应用,并极大地改变人们生活方式的设备。现在大部分的物业管理机制仍是传统的社区服务,尤其是在一些二线三线城市,物业管理的现状更是堪忧。人们迫切需要网络化信息传输,依托移动互联网蓬勃发展的大趋势,提高物业服务质量和效率。为了更好的整合资源,提高资源利用率,解决小区业主的痛点,我们要顺应时代发展的潮流,利用我们学习到的互联网相关的知识,在小区开发建设如火如荼的同时,开发基于手机的现代化的,集信息收集和处理,及时反馈,功能丰富的物业服务系统,以提高管理质量,改善人们的生活的目标。

本篇论文将采用面向对象的软件设计思想,按照安卓手机APP的开发规范,系统分析手机物业服务系统的需求和业务逻辑,从而得出APP的详细设计。本APP系统主要面向安卓手机用户,APP的功能涵盖了物业信息采集、物业缴费、物业投诉、社区动态、交房装修等方面的服务。论文主要完成安卓端APP的框架设计,页面的分析与设计,与后台服务端的数据调取和交互,性能的优化等等。论文采用IOC的设计思想,使用JSON作为接口数据交互格式,采用各种设计模式和第三方库等等设计与开发系统。后端采用Java的

SSH框架编写，采用MYSQL数据库作为后台数据系统，实现了对小区用户的信息和小区设备信息进行及时搜集和处理，同时在手机上操作，大大方便了小区业主的生活。

本文开发的物业服务APP是将移动互联网与传统的物业缴费，管理模式结合在一起，在此基础上设计和实现的。为了使APP能够发挥出便携性、及时性、安全性的优势，我们以操作简便、界面友好为目标开发和完善。

关键词：物业服务； 安卓； IOC；

Design And Development of Residential Property Service System

Based On Mobile APP

Author: Qian Wen jun Tutor: Han Bing

(Department of Software Engineering, School of Information Science and Technology, HaiNan Normal University, HaiKou, xxxxx)

Abstract: With the incessant development of the Real Estate Industry in our nation, people have increasing desire of their house. And they turn their attention from the number to high-quality community life, more convenient functions and service experiences. As a subject, property service is going to solve the crying need that turning the development direction from the traditional information acquisition and labor management to Intelligent data analysis, automated management and people-oriented service.

Recent years, {40%:the science and technology especially the internet technology are continuously developing,} {91%: and the internet technology has penetrated every aspect of social life.} Especially, the rapid development of Mobile Internet Technology has made mobile phones develop to equipment that can install an OS into it and can also install a lot of apps and greatly change the life-style of people, not just communication tools. Nowadays, most of the property management mechanism is still the traditional community services, especially in some second-tier third-tier cities, {50%: property management is even more worrying about the status quo.} People urgently need networked information transmission, {50%: relying on the mobile Internet vigorous development trend,} {75%: improve the quality of property services and efficiency.} In order to better integrate resources, improve resource utilization, to solve the pain of community owners, {64%: we should follow the trend of the development of the times,} {82%: we use the Internet to learn the knowledge,} {54%: in the community development and construction in full swing at the same time,} the development of mobile phone based on the modernization, {50%: Set the information collection and processing, timely feedback,} {40%: feature-rich property service system to improve the quality of management and improve people's lives.}

{50%:This paper will use the object-oriented software design ideas,} according to the development of Android mobile phone APP specification, system analysis of mobile phone property service system needs and business logic, resulting in the detailed design of APP. The APP system mainly for Android mobile phone users, APP's function covers the property information collection, property payment, property complaints, community dynamics, delivery decoration and so on services. The paper mainly completes the framework design, the analysis and design of the page, the data retrieval and interaction, {75%: the optimization of the performance and so on.} The paper adopts the design idea of IOC, uses JSON as the interface data interactive format, adopts various design patterns and third party library and so on design and development system. Back-end using Java's SSH framework, the use of MYSQL database as a back-end data system to achieve the community information and community equipment information for timely collection and

processing, while operating on the phone, {43%: greatly facilitate the community owners of life.}

This paper develops the property service APP is the mobile Internet and the traditional property payment, management mode together, {57%: on the basis of design and implementation.} In order to make APP to play a portability, timeliness, security advantages, we operate with simple, user-friendly development and improvement.

Key words: the Property Service; Android;IOC

1引言

1.1课题背景

{46%:随着我国房地产行业日渐成熟,再加上近年来我国的科学技术水平和经济发展状况都有显著的提高,我国城市化进程不断加快,}物业服务的质量直接反映出我国的社会发展状况和人民的生活水平质量。目前互联网技术不断成熟,当前社会正处于信息技术的研究和应用的中间点,移动互联网技术已经发展到了一定的阶段,但是移动互联网技术的应用还处在初期,很多传统行业还没有被普及。有研究表明,目前我们国家的小区物业管理由于近十年来的不断发展,正处于蓬勃发展的态势。{53%:由于ISO9002国际质量管理体系等在物业管理与服务中的应用,我国的物业服务整体上朝着正规化,人性化的方向发展。}

从另一个角度看,我国的居住物业服务行业并没有快速地跟上互联网发展的步伐,整体上还是以传统的人工服务的形式为主。其一,社区公告还是以纸质的公告为主,住户无法及时了解最新的信息。其二,由于缴费接口没有统一,物业缴费形式仍然以传统的现金缴费方式为主,住户必须到缴费地点排队缴费,浪费了大量的时间和人力物力。其三,用户的投诉,房产的交接等等也没有办法方便地得到及时处理和反馈。{43%:为解决上述问题,基于手机的小区物业服务系统的设计与开发很有必要。}

1.2研究的意义和目的

{41%:近几年移动互联网蓬勃发展,网络技术的应用已经逐渐改变人们的生活方式。}而作为直接影响人们社区生活的小区物业服务行业,其服务质量和效率尤为重要,传统的服务形式早已无法满足现代人们的需求。基于以上出现的问题,结合当前手机APP蓬勃发展的大背景,基于手机的小区物业服务系统的设计与开发很有意义。

物业服务APP,顾名思义是一款能够给小区的住户提供物业服务的手机APP。它在物业公司和小区住户之间搭建起了一个服务提供的桥梁。开发这样一款APP,能够解决一下的问题:

第一:原有的物业服务形式多采用人工方式,效率低下,无论从及时性,正确率还是智能手机几乎人手一部,本文中开发的APP方便业主及时接收和查阅社区的各种信息,自动进行数据处理和统计,用户动动手指就能处理业务,查看和反馈各种信息,最大限度的节省了业主们的宝贵时间。

第二:方便物业服务公司的工作。物业公司可以使用APP,利用现在强大的第三方支付平台,自动收取物业费,水费,电费等等费用,节省了大量的人力物力,同时能很大程度上保障数据的正确性。物业公司也可以通过我们的APP及时收取业主的交房验房信息、投诉等等,这样就方便物业合理安排人员,及时处理问题。同时,传统的贴公告的方式,不仅浪费人力,而且用户能及时收到信息的概率很低。APP上面的小区动态能够方便物业公司快速的发送通知到各个小区的每一个用户,拓宽了信息渠道。

1.3物业服务的现状

{82%:物业管理既是房地产行业经营和管理的重要组成部分,又是现代城市管理不可缺少的一部分。}物业服务行业虽然发展势头良好,但是仍然存在以下的问题:

1. 业主知情权问题

{44%: 目前我国的物业服务前期在程序管理和质量控制方面虽然还需要进一步的完善和不断改进, 但是也已经开始了市场化的进程。} {78%: 后期物业管理却仍然大部分停留在指派或协议进驻阶段, 广大业主的选择权甚至是知情权都没有办法发挥, 相当于业主需要雇一个保姆, 工资都定好了, } {100%: 而且干好干坏都得付钱, 这样理所当然会引起业主的反感。} 究其原因, 主要是因为消息的传递通道闭塞, 业主无法及时收到各种社区服务的消息推送, 因此建设一个社区服务平台很有必要, 它能够将物业服务公司的消息直接推送到用户手机上, 使得业主能够第一时间掌握第一手资料, 充分发挥业主的知情权和选择权。

2. 收费精细化问题

{64%: 目前, 我国大部分的物业服务在收费上面没有做到详细透明, 不仅业主不清楚物业费用的收费项目, 甚至对当前正在享受哪些物业服务也知之甚少。} {97%: 模糊的物业收费直接导致的后果就是, 只要业主对物业有一点不满, 不管其他的服务做得多好, 都可以以此为理由拒绝交费, 这样也会造成物业公司的大量损失。}

3. 报修投诉反馈问题

目前大部分小区接收用户投诉主要采用人工接听电话的方式, 不仅效率低下, 而且很容易出现错误。导致住户反映的问题无法及时得到处理和解决, 这样不仅使得小区业主的问题迟迟得不到反馈, 而且也使得物业在处理业主问题时也因为无法及时准确获取到相应的信息而产生管理混乱, 人力资源浪费很大的问题。

1. 4研究内容

本篇论文主要是对基于手机的小区物业服务系统的开发与研究, 在研究现代智能手机APP开发技术的同时, 解决我国房地产行业物业服务中存在的一些问题。在研究了传统物业服务的方式和特点之后, 进行深入的需求分析和系统设计, 给出系统的各个功能模块和页面的详细设计。{51%: 在此基础上本文详细介绍了如何利用Android Studio进行开发和调试, 实现各个页面的编写和功能。} 在开发过程中采用了面向对象的设计思想, 着重介绍项目架构设计中用到的IOC思想和各种设计模式。最后进行APP的测试, 切实开发出一款基于安卓手机的小区物业服务APP。以下是本文的详细内容:

{40%: 第一章: 引言, 主要介绍了物业服务APP的研究背景, 并论述研究的目的和意义, 并分析目前我国的物业管理与服务的现状和痛点, 由此确定了本文的研究重点和方向, } 要达到的效果。

第二章: 相关技术, 介绍基于手机的物业服务APP运用的主要技术和架构。

{44%: 第三章: 系统需求分析, 包括物业服务系统APP端的需求来源, 进行可行性分析, 并对APP的各个功能模块和页面进行深入的功能性和非功能性需求分析, } 并论述APP的设计思想。

{41%: 第四章: 系统方案设计, 在需求分析的基础上, 对基于手机的物业服务APP进行了架构设计, 利用时序图, 类图, 程序流程图等等对APP的各个功能模块进行分析, } 设计出各个页面的样式和功能。

第五章: 系统的实现, 主要是在系统设计的基础上, 通过相关的安卓手机APP开发工具和技术, 实现各个页面的UI编写和功能逻辑, 数据渲染等等, 展示了系统实现的内容。

第六章: 结束语, 本文的全面总结和在总结的基础上对未来的展望。

2相关技术

2.1 Android平台介绍与系统框架

{98%:Android系统是Google开发的一款开源移动OS, Android中文名被国内用户俗称“安卓”。}{100%:Android操作系统基于Linux内核设计, 使用了Google公司自己开发的Dalvik Java虚拟机。}{43%:Android操作系统已经成为全球最大的智能手机操作系统。Android系统架构如下图: }

图2-1安卓系统架构图

在系统架构上, Android平台共分为五层, 分别是:

1. Linux内核 (Linux Kernel) : Android系统是基于Linux内核的, 这也是安卓系统具有高度可移植性和极高安全性的原因之一。

{98%:Android Runtime (ART) 依靠 Linux 内核来执行底层功能, 例如线程和低层内存管理。}{58%:在Linux内核层, Android系统提供了大量的硬件驱动, 比如USB驱动、蓝牙驱动、电源管理、相机驱动、WiFi驱动等等。}

2. {65%:Android硬件抽象层 (Hardware Abstraction Haler) : HAL是对Linux内核驱动程序的封装, 向上提供接口, }屏蔽掉底层的实现细节。{100%:也就是说, 把对硬件的支持分成了两层, 一层放在用户空间 (User Space), 一层放在内核空间 (Kernel Space), 其中, }{100%:硬件抽象层运行在用户空间, 而Linux内核驱动程序运行在内核空间。}{97%:为什么要这样安排呢? 把硬件抽象层和内核驱动整合在一起放在内核空间不可行吗? }{100%:从技术实现的角度来看, 是可以的, 然而从商业的角度来看, 把对硬件的支持逻辑都放在内核空间, 可能会损害厂家的利益。}{100%:我们知道, Linux内核源代码版权遵循GNU License, 而Android源代码版权遵循Apache License, 前者在发布产品时, }{100%:必须公布源代码, 而后者无须发布源代码。}{100%:如果把对硬件支持的所有代码都放在Linux驱动层, 那就意味着发布时要公开驱动程序的源代码, 而公开源代码就意味着把硬件的相关参数和实现都公开了, }{100%:在手机市场竞争激烈的今天, 这对厂家来说, 损害是非常大的。}

3. {65%:原生C/C++库 (Native C/C++ Libraries) 和Android Runtime: 许多核心 Android 系统组件和服务在构建自原生代码时, }需要以 C 和 C++ 编写的原生库, 例如OpenGL 库提供给应用框架层API, 能够实现2D和3D图形的绘制。{87%:而Android Runtime为安卓系统提供基于Java虚拟机的Dalvik虚拟机, 每个Dalvik虚拟机运行一个独立的应用, 并占用独立的进程。}

4. {51%:应用框架层 (Application Framework) : 这一层是通过Java语言编写的一个API集合, 包含了所有Android OS的核心功能集。}它为系统的四大组件和其他核心组件都提供了相应的管理器 (Manager) 。

5. {55%:应用层 (Application) : 系统应用和第三方开发的应用都处于这一层。}

2.2 Android系统的四大组件

一个安卓应用各种功能的实现, 通常是由安卓系统的四大组件来完成的。四大组件都有自己的生命周期, 他们都是通过Intent (我们称之为“意图”) 来传递消息, 并且在安卓应用中担任不同的职责。这四大组件分别是:

1. Activity (与用户交互的界面)

{67%:一个Activity在Android应用中就是一个用户界面, 用户可与其提供的屏幕进行交互, }{78%:以执行拨打电话、拍摄照片、发送电子邮件或查看地图等业务逻辑需要的操作。}{44%:Activity 是四大组件中唯一一个用户可见的系统组件, 也是使用得最多的一个组件。}{100%:每个 Activity 都会获得一个用于绘制其用户界面的窗口。}{100%:窗口通常会充满屏幕, 但也可小于屏幕并浮动在其他窗口之上。}

2. Service (后台服务)

{97%:Service 是一个可以在后台执行长时间运行操作而不提供用户界面的应用组件。}{100%:服务可由其他应用组件启动, 而且即使用户切换到其他应用, 服务仍将在后台继续运行。}{100%: 此外, 组件可以绑定到服务, 以与之进行交互, 甚至是执行进程间通信(IPC)。}{100%: 例如, 服务可以处理网络事务、播放音乐, 执行文件 I/O 或与内容提供程序交互, 而所有这一切均可在后台进行。}

3. Broadcast Receiver (广播接收器)

{100%:一个Android应用可以使用广播接受器对外部事件进行过滤, 只对感兴趣的外部事件(如当电话呼入时, 或者数据网络可用时)进行接收并做出响应。}广播接收器没有用户界面。{95%:然而, 它们可以启动一个activity或服务来响应它们收到的信息, 或者用Notification Manager 来通知用户。}{100%:通知可以用很多种方式来吸引用户的注意力——闪烁背灯、震动、播放声音等。}{100%:一般来说是在状态栏上放一个持久的图标, 用户可以打开它并获取消息。}{61%:在Android系统中, 广播可以分为有序广播和无序广播两种。}

4. Content Provider (内容提供者)

{88%:内容提供程序管理对结构化数据集的访问。它们封装数据, 并提供用于定义数据安全性的机制。}内容提供程序是连接一个进程中的数据与另一个进程中运行的代码的标准界面。

{100%:如果您想要访问内容提供程序中的数据, }{96%:可以将应用的Context 中的 ContentResolver 对象用作客户端来与提供程序通信。}{100%: ContentResolver 对象会与提供程序对象(即实现 Content Provider 的类实例)通信。}{100%: 提供程序对象从客户端接收数据请求, 执行请求的操作并返回结果。}

{100%:如果您不打算与其他应用共享数据, 则无需开发自己的提供程序。}{100%: 不过, 您需要通过自己的提供程序在您自己的应用中提供自定义搜索建议。}{88%: 如果您想将复杂的数据或文件从您的应用复制并粘贴到其他应用中, 也需要创建您自己的提供程序。}

{94%:Android 本身包括的内容提供程序可管理音频、视频、图像和个人联系信息等数据。}{100%: android.provider 软件包参考文档中列出了部分提供程序。}{100%: 任何 Android 应用都可以访问这些提供程序, 但会受到某些限制。}

2.3 控制反转 (IOC) 技术

{76%:控制反转 (Inversion of Control, 英文缩写为IOC), 是一种在现代面向对象编程技术中重要的技术, 它是为了减少程序中的耦合性而诞生的。}{50%:在Java后端框架中, 控制反转技术是Spring框架的核心。}{75%:IOC一般分为两种类型, 一种是依赖注入 (Dependency Injection), 另一种是依赖查找 (Dependency Lookup), 其中, }依赖注入运用的更加广泛。在面向对象的编程中, 要实现一个功能, 通常需要两个或者更多的类协同来实现, 这个时候一个类就必须包含与之合作的类(我们称之为依赖类)的实例。通常的做法是这个类自己去获取依赖对象, 这样做无疑会导致程序的高度耦合。

而控制反转(也称为“依赖倒置原理”)是通过构建一个能够调控系统对象的组件, 这个组件能够在某个系统组件或者类需要某一个依赖对象时, {64%:将一个实例化后的依赖对象的引用传递给这个组件。}{49%:通常我们认为, 这个依赖对象是被我们构建的这个“调控组件”注入到了需要的类中。}而控制反转的“反转”在于系统组件获取依赖对象的过程被反转了, 它不再需要类和组件自己创建依赖对象, 从而削减了系统的耦合度。

在本文介绍的物业服务APP的开发中, 我们通过控制反转技术, 实现了页面内容视图的自动注入和页面中各个控件对象的注入。并通过Java的动态代理机制, 实现了控件点击事件等多种事件的动态代理, 节省了很多代码, 而且提高了代码的质量和可阅读性。

2.4 C/S模式概念

本文中开发的小区物业服务APP，是一个典型的C/S架构的软件系统，其数据的交互是由一台远端Apache服务器来提供的。这里我来阐述一下什么是C/S架构。

{63%:C/S是客户端/服务器 (Client/Server) 的缩写，在客户端/服务端网络中，服务器处于核心地位，它为客户端提供所需要的网络资源，}而客户端是系统的基础，客户端的所有数据都需要通过HTTP请求从服务端获取。{59%:一个C/S架构的软件系统最核心的就是由两部分组成，即客户端应用程序和服务端应用程序。}与B/S（浏览器/服务器）架构相比，C/S架构中客户端程序通常需要安装到特定的终端，其主要的特点是能够减轻服务端的压力，而且安全性和稳定性更高。

图2-2 C/S模式二层体系结构

2. 50KHttp框架介绍

如上文所述，本文中开发的小区物业服务APP，是一个典型的C/S架构软件系统的客户端应用程序。C/S架构体系中，客户端与服务端的交互一般是通过HTTP请求来获取JSON（一种以键值对为基础的数据格式）格式数据，并进行解析来进行的。本文中开发的小区物业服务APP，在数据交互上面使用了当前最流行的OKHttp框架。

{80%:一般在Java平台上，我们会使用Apache HttpClient作为Http客户端，用于发送 HTTP 请求，并对响应进行处理。}{83%:比如可以使用HTTP客户端与第三方服务(如SSO服务)进行集成，当然还可以爬取网上的数据等。}{83%:OKHttp与HttpClient类似，也是一个HTTP客户端，提供了对 HTTP/2 和 SPDY 的支持，并提供了连接池，}{100%:GZIP 压缩和 HTTP 响应缓存功能。}

{98%:当通过OkHttp 来请求某个 URL 时，OkHttp 首先从 URL 中得到地址信息，再从连接池中根据地址来获取连接。}{100%:如果在连接池中没有找到连接，则选择一个路由来尝试连接。}{93%:尝试连接需要通过 DNS 查询来得到服务器的 IP 地址，也会用到代理服务器和 TLS 版本等信息。}{97%:当实际的连接建立之后，OkHttp 发送HTTP请求并获取响应。}{100%:当连接出现问题时，OkHttp会自动选择另外的路由进行尝试。}{75%:这使得OkHttp可以自动处理可能出现的网络问题。}{93%:当成功获取到HTTP请求的响应之后，当前的连接会被放回到连接池中，提供给后续的请求来复用。}{100%:连接池会定期把闲置的连接关闭以释放资源。}

3系统需求分析

3.1开发环境

在进行基于手机的物业服务APP的开发中，项目主要使用到的软件开发环境如下：

操作系统：windows 10

{98%:开发工具：Android Studio 2.2}

{78%:Android SDK：Android 6.0 (API Level 23.0) }

测试机：小米5 Android 7.0，模拟器 Android 6.0

{100%:建模工具：Rational Rose 2007}

作图工具：Visio 2010

{67%:服务器：阿里云 Windows Server 2008 (搭载Apache 6.0) }

3.2可行性分析

1. 经济可行性

在经济上，客户使用基于手机的物业服务APP，一方面做到了信息的及时性，完整性，大大节省了小区住户在缴费，投诉，获取社区通知，报修等等流程的时间，给人们的生活带来极大的便利；另一方面，物业公司也能因此节省大量人力，同时因为提供了更好的服务，让小区住户享受该有的权利，收到的投诉更少，消费水平也得到改善，这样能大大降低运营成本。

2. 技术可行性

在技术上，开发一款基于Android手机的物业服务APP需要的核心技术一个是对安卓架构和API的熟悉和了解，另外一个就是实现数据交互和缴费。对于Android技术已经有了一定的开发经验，数据交互上面使用OKHttp和Gson进行JSON数据解析，而缴费上我们可以使用支付宝支付SDK来实现。所以，在技术上是可行的。

3. 运行可行性

由于我们基于最新版本的Android Studio2.2来开发和调试，使用稳定的Android6.0SDK，并使用真机进行测试，保障了程序的正确性。由于Android是向上兼容的，所以APP可以在Android 2.3以上的所有安卓手机上面正常运行。

4. 法律可行性

本文开发的APP是一个为小区住户提供物业服务的APP，方便住户的社区生活，不会恶意盗取用户的隐私信息，用户的缴费，投诉，报修等等都有相应的历史记录可以查询。

3.3物业服务APP的功能性需求

3.3.1物业服务APP需求概述

基于手机的物业服务APP是物业管理部门提供给小区住户使用的，能够方便的搭建小区住户和物业管理公司之间信息的桥梁，并将传统的物业服务于移动互联网相结合，方便双方物业公司的管理，也方便住户的生活的一款APP。根据需求，这款APP要能够方便小区住户在手机上实时查阅物业公司或者其他同一个社区的用户发送的通知和动态。用户能够在APP上办理交房验房的初步手续，并保障信息的合法性和正确性。同时，物业公司将各个物业的缴费接口提供给APP，使得小区住户能够通过APP实时缴物业费，水电费等等费用，并能够有详细的历史记录可以查询。物业管理部门能够通过APP的后台管理系统了解用户的投诉，报修等等图文并茂的信息，并统一处理。

经过收集资料和分析，最终确定该物业服务APP的主要功能模块有APP首页，邻里圈和个人中心，其中APP首页的主要功能有小区动态，报修，交房验房，投诉和缴费。邻里圈主要是同一个社区的住户发表的动态，能够发表文字和图片，发表的动态只要是同一个小区的用户都能够浏览到。个人中心的主要功能有查看缴费的历史记录、意见反馈，修改密码，登录和注册等。下面是APP的功能需求图：

{76%:图3-1APP功能需求图}

3.3.2物业服务APP功能描述

(1) 小区动态：小区动态主要是由物业管理和居委会等发布的社区周边新闻和公告等信息，包含文字和图片。用户可以在小区动态

页面查看，并支持下拉刷新和上拉加载。

(2) 报修：用户发现自家或者公共的设施出现异常，可以在APP的报修界面申请报修。申请时需填入用户姓名和电话，报修设施类型（公共设施或者私人设施），以及描述等信息，并可上传图片最多3张图片方便维修人员了解情况。一旦报修信息提交成功，后台人员确认之后，便会安排维修人员上门维修。并可以查看历史报修记录。

(3) 交房验房：在交房验房模块，用户可以了解收楼手续，装修手续的流程，并了解装修管理协议等信息，并能根据需要在线提交收楼需要的家庭情况登记表，装修申请表，和业主收楼确认表等。

(4) 投诉：当业主需要投诉时，可以在APP的投诉页面发表投诉，投诉可以输入遇到的问题和上传图片证据，物业部门在后台收到业主投诉，经过查实会采取相应措施。

(5) 缴费：用户可以在APP首页缴费页面通过支付宝缴纳物业费，水电费，燃气费，宽带费等等费用，所有的缴费都会有历史清单，用户可以在缴费记录查看。

(6) 查看邻里圈动态：查看社区住户发表的图文动态。

(7) 发表邻里圈动态：编辑动态并发表。

(8) 查看缴费记录：用户可以查看自己的历史缴费清单，所有的账单一目了然。

(9) 完善资料：有一些敏感操作需要用户完善自己的信息，比如缴费和验收房。用户可以在完善资料页面填写并提交。物业公司后台收到后会更新改用户的资料。

(10) 意见反馈：用户可以向物业公司反馈意见。

(11) 修改密码：用户可以修改登录密码，修改密码时需要输入6位数的短信验证码，验证通过并且输入的密码合理才可修改。

(12) 登录注册：用户可以登录APP或者注册为新用户。

{55%:基于手机的物业服务APP的功能设计如下图所示:}

图3-2 物业服务APP功能模块结构图

3.4 物业服务APP功能用例分析

APP的功能模块主要分为缴费模块，用户模块，邻里圈和个人中心模块。

其中用户管理模块是本APP的核心模块之一，用户未注册之前视为“游客”，游客只能查看小区动态、查看邻友圈的权限，注册用户享受大部分权限，而交房验房、缴费功能只有完善资料的用户才可以使用。具体的用例如下

图3-3 物业服务APP系统用例图

针对APP的系统用例，下面的各个表中给出了详细的用例分析。

(1) 用户注册

用户注册的用例分析如下表所示：

表3-1注册用例分析

用例名称注册

用例ID001

角色用户

{71%:用例描述用户点击登录页注册按钮进入注册页，填写手机号并点击“获取验证码”，填写收到的6位数字验证码，并输入密码，完成注册。}

前置条件用户进入登录页

后置注册成功跳转到登录页

(2) 用户登录

用户登录的用例分析如下表所示：

表3-2登录用例分析

用例名称登录

用例ID002

角色用户

{65%:用例描述用户注册完成后，进入登录页，输入手机号和密码，点击登录按钮完成登录。}

前置条件用户完成注册

后置登录成功跳转到APP首页

(3) 用户报修

用户报修的用例分析如下表所示：

{55%:表3-3用户报修用例分析}

用例名称报修

用例ID003

角色用户

用例描述用户进入首页，点击报修按钮进入报修页面，选择报修类型，输入姓名和电话以及报修描述，并添加图片（最多3张），点击提交按钮完成报修。

前置条件用户已登录

后置报修成功跳转到首页

(4) 用户投诉

用户投诉的用例分析如下表所示：

表3-4用户投诉用例分析

用例名称登录

用例ID004

角色用户

用例描述用户进入首页，点击投诉按钮进入投诉页面，输入投诉的文字内容，选择要上传你的图片（最多3张，也可不选择），点击提交按钮完成投诉。

前置条件用户完成注册

后置投诉成功跳转到APP首页

(5) 用户新增缴费地址

用户新增缴费地址的用例分析如下表所示：

{40%:表3-5用户新增缴费地址用例分析}

用例名称新增缴费地址

用例ID005

角色用户

用例描述用户进入首页，点击任意一个缴费按钮，进入缴费地址页面，点击“新增地址”，进入新增缴费地址页面，填入地址信息。

前置条件用户已登录

后置缴费地址列表新增一条地址

(6) 物业缴费

用户物业缴费的用例分析如下表所示：

表3-6用户物业缴费用例分析

用例名称物业缴费

用例ID006

角色用户

用例描述用户进入首页，点击任意一个缴费按钮，进入缴费地址页面，选择一个缴费地址，并进入缴费清单界面，选择缴费的月份，并核对金额，确认后输入支付密码进行缴费。

前置条件用户已登录且完善资料

后置用户缴费成功，回到首页

(7) 用户完善个人资料

用户完善个人资料的用例分析如下表所示：

{40%:表3-7用户完善个人资料用例分析}

用例名称完善个人资料

用例ID007

角色用户

用例描述用户进入首页，点击个人中心完善资料条目，进入完善资料页面，按照要求填入各项信息，并点击提交按钮。

前置条件用户已登录

后置完善资料成功，回到首页

(8) 用户查看小区动态

用户查看小区动态的用例分析如下表所示：

{40%:表3-8用户查看小区动态用例分析}

用例名称查看小区动态

用例ID008

角色用户

用例描述用户进入首页，点击小区动态按钮，进入小区动态页面，查看小区动态。

前置条件用户已登录

后置查看完点击返回键回到APP首页

(9) 用户查看邻里圈动态

用户查看邻里圈动态的用例分析如下表所示：

表3-9用户查看邻里圈动态用例分析

用例名称用户查看邻里圈动态

用例ID009

角色用户和游客

用例描述用户进入首页，点击邻里圈，进入邻里圈页面，查看当前邻里圈动态。

前置条件用户点击邻里圈

后置无

(10) 用户发表邻里圈动态

用户发表邻里圈动态的用例分析如下表所示：

表3-10用户发表邻里圈动态用例分析

用例名称用户发表邻里圈动态

用例ID010

角色用户

用例描述用户进入首页，点击邻里圈，进入邻里圈页面，点击右上角的添加按钮，进入发表邻里圈页面，输入文字，选择图片，点击发表按钮。

前置条件用户已登录

后置发表完成进入邻里圈页面

(11) 用户意见反馈

用户意见反馈的用例分析如下表所示：

表3-11用户意见反馈用例分析

{97%:用例名称用户意见反馈}

用例ID011

角色用户

用例描述用户进入首页，点击个人中心，点击 意见反馈条目，进入意见反馈页面，输入反馈内容，点击提交按钮。

前置条件用户已登录

后置反馈成功，返回个人中心

{76%: (12) 用户修改手机号}

用户修改手机号用例分析如下表所示：

表3-12用户修改手机号用例分析

用例名称用户修改手机号

用例ID012

角色用户

用例描述用户进入首页，点击个人中心，点击修改手机号条目，进入修改手机号页面，输入新手机号和收到的短信验证码与密码，点击确认按钮进行修改。

前置条件用户已登录

后置修改成功，进入登录页重新登录

4. 物业服务APP详细设计

4.1系统总体架构

系统的总体架构图如下：

图4-1 APP的系统架构图

上图是从整个软件系统来考虑，不仅仅局限于APP的设计。从整体上来看，整个系统共分为4层，分别是数据层、服务层、控制层和访

问层。我们开发的APP属于访问层。小区用户通过APP注册成为客户端用户，并通过接口与Web服务器进行交互，实现缴费、投诉、发表邻里圈动态、报修、反馈意见，接收消息等功能。而物业公司管理人员通过浏览器注册登录系统，成为系统管理员。登录之后可以实现缴费管理、会员管理、投诉管理、报修管理、消息管理等功能。而所有的交互数据都是由WEB服务器处理后反馈给APP用户和系统管理员的。Web服务器所需要的数据存储于数据库中，web服务器通过增删改查等操作，与数据库进行交互。

4. 2APP功能模块设计

4. 2. 1用户模块设计

APP的用户模块主要有用户的登录和注册，投诉，报修，查看社区通知动态，办理交房验房手续等功能。上述功能只有登录用户才可以使用。APP用户的唯一标识是手机号，在APP中只有通过手机号注册才可登录，只有登录成功之后才可以进行投诉、报修、验收房等操作。

1. APP用户注册的时序图如下图所示：

图4-1 APP用户注册时序图

用户注册的时序图中我们可以看出，用户点击进入APP首页之后才有注册页面的入口，进入注册页面之后，首先需要填入手机号、密码、确认密码、验证码等信息，填入之后通过API接口提交给后台服务器进行验证，服务器验证通过，则会将此用户信息加密后存储到数据库表中，并返回给APP注册成功的信息。同样，用户的登录过程的时序图是相同的，这里不再重复阐述。

2. 用户报修时序图如下图所示：

图4-2 APP用户报修时序图

从时序图中我们可以知道，用户进入首页后，APP会先验证当前登录信息是否有效，如果当前登录信息无效（可能是超过15天未登录导致过期，或者未登录），则会先调取登录流程进行登录。登录成功之后，点击报修进入报修页面，并填入报修相关信息，访问接口将数据提交给服务器验证，验证通过则存储到数据库表中，并返回报修成功的信息。用户投诉的时序图和报修的时序图几乎是一样的，这里不再重复阐述。

3. 用户缴费时序图如下所示：

图4-3APP用户缴费时序图

从时序图中我们可以知道，用户在缴费之前需要先选择缴费的类型和缴费地址，这是因为考虑到用户可能有多套房子，拥有多个缴费地址的情况。如果没有缴费地址，可以在缴费地址页面添加缴费地址之后进行缴费。缴费的类型有物业费、水费、电费、燃气费、宽带费等等。缴费是通过调用支付宝支付的接口，当缴费成功之后，会提交数据到服务端，并存储到数据库中。

4. 2. 2邻里圈模块设计

邻里圈模块主要是一个社区动态功能，用户在登录之后可以发表心情动态，发表心情时可以输入文字，图片，表情等内容。用户在邻里圈可以发表动态和浏览动态，随时随地掌握周边发生的生活趣闻，也可以将自己的生活点滴与小区的其他住户分享。用户浏览邻里圈的时序图如下图所示：

图4-4用户浏览邻里圈时序图

从时序图可以了解到，邻里圈属于APP中的一个主要模块，用户在点击邻里圈之后，需要调用API获取当前的邻里圈数据，在调取的同时需要给服务端发送当前用户的验证信息，服务端获取这个信息之后经过处理，从数据库中加载到可用的邻里圈动态列表数据，并通过接口返回给APP，经过解析显示到Activity界面上。

同样，用户发表邻里圈的时序图如下图所示：

图4-5用户发表邻里圈时序图

从上图可以看出，登录用户发表邻里圈时，需要先填写要发表的内容，这些心情动态数据可以是图片，文字和表情等等。**{45%:如果有图片，需要先调用一个接口将图片上传到服务器，当上传成功时，在上传文字等信息。}**服务器处理后存储到数据库中。

4.2.3个人中心模块设计

个人中心是APP的第三个主要模块，在APP的首页底栏点击“个人中心”即可进入个人中心页面。在个人中心中，如果当前用户已登录并且登录验证没有过期，则会现实当前用户的手机号和头像等信息。如果未登录，则会显示登录注册的按钮，用户点击可以进行登录或者注册为新的用户。登录之后可以查看缴费记录，完善用户资料，反馈意见，修改手机号，修改密码，清除缓存，以及退出当前账号等操作。

1. APP用户查看缴费记录

用户缴费之后，可以在个人中心模块查看缴费记录，缴费记录会按照缴费时间降序排序，每一个缴费记录都会有缴费金额，缴费类型，缴费时间，缴费地址，缴费状态等信息，点击条目可以查看详情。用户查看缴费记录的时序图如下所示：

图4-6用户查看缴费记录时序图

从上图可以看出，查看缴费记录是比较敏感的操作，APP会先进行登录信息验证，如果验证通过，才会将用户的最近缴费情况的数据通过API调用返回给用户查看。

2. APP用户修改登录信息

{43%:用户修改登录信息分为修改手机号和修改密码，修改手机号需要验证新手机号的验证码，修改密码可以通过旧密码修改或者短信验证码修改。}时序图如下所示：

图4-7用户修改手机号时序图

从上图可以看出，用户修改手机号需要第三方平台的短信验证码，用户首先调用接口获取短信验证码，手机收到短信验证码之后，输入短信验证码和新的密码，并将数据通过接口提交到服务器，服务器验证没有问题后，更新数据库中的记录，并返回提示信息。

个人中心的其他功能的时序图均和用户查看缴费记录的时序图是相同的，这里不再重复阐述。

5. 基于手机的物业服务APP的实现

5.1开发平台介绍

{48%:本文的物业服务APP是基于安卓的一款APP，在开发的过程中主要使用了Android Studio2.2作为开发工具。}**{48%:早在2013年，以Eclipse为开发工具，集成ADT插件和Android SDK的开发形式就逐渐被淘汰，因为Google推出了一款更为强大的，}****{75%:专门针**

对Android开发的工具，这就是Android Studio。}{41%:Android Studio的代码编辑器是基于IntelliJ IDEA的，并集成了gradle工具，使得项目的集成更方便。}{49%:与Eclipse ADT相比，Android Studio无论从编译器的编译速度，对性能的损耗，}代码编辑的方便性还是项目集成的方便性上面都有了质的飞跃。

在开发过程中，主要使用了Android6.0SDK进行开发，这个版本是去年推出的，市场上新机型的自带系统版本大部分都是6.0。Android6.0可以保障一定的稳定性，同时也提供了大量的新特性供开发者调用。

项目开发中调用接口，主要使用OKHttp框架作为接口请求的框架，并使用一些设计模式进行了封装。{48%:JSON数据解析主要使用了Gson框架，Gson框架是Google推出的一款非常方便的JSON数据解析框架，}能够很方便的将JSON字符串或者JSON数组解析成为Java对象，反过来，也能很方便的将一个Java对象解析成JSON字符串。

项目开发的主要思想是面向对象的软件设计思想（OOP），在开发过程中使用了控制反转（IOC）技术，同时在设计的过程中也用到了一些基于OOP的设计模式。

5.2APP的架构设计

5.2.1基于IOC实现依赖注入

（1）为什么要使用IOC技术

IOC是控制反转的英文缩写，IOC的核心思想是能够实现代码的解耦，同时使用IOC也能提高代码的可读性和可扩展性。{40%:在Android开发中，页面的实现过程一般是通过在资源文件夹中定义一个xml格式的资源文件，}并在xml中定义这个页面的各个视图组件以及这些组件的关系还有各种属性等等。在页面生成过程中，xml中定义的每一个组件都会被解析成一个视图对象（View类的子类对象），这些对象可以绑定给Activity的布局，并实现相应的业务逻辑。

{41%:对于Activity组件来说，页面中的每个控件对象都是一个依赖对象，}{61%:传统的方式是通过Activity调用自身的findViewById方法将这些获取到。}而给这些控件绑定各种事件监听也需要Activity自身来完成。这样就造成了大量的冗余代码，并且使代码的可阅读性大大降低。

上文中提到，IOC的思想就是将一个组件获取其依赖对象的实现过程“反转”，通过一个第三方组件完成，从而大大降低了程序的耦合性。在本文中，我们就通过编写一个IOC的容器，来实现所有页面视图控件的绑定，页面内容视图的绑定以及事件监听的绑定。

（2）基于IOC的依赖注入具体实现

首先包结构和主要的类如下图所示：

图5-1 IOC系统主要类设计

为了更方便的了解这些注解和类的关系，其类图如下图所示：

图5-2 IOC依赖注入类图

从上图中我们可以知道，在设计中主要有四个包和3个核心类。这三个和核心类分别是BaseAnnoInjectActivity、AnnoInjectUtils和MyEventInvocationHandler，下面分别介绍这三个类的作用：

（1）BaseAnnoInjectActivity，所有页面的父类，在页面创建的OnCreate函数通过调用AnnoInjectUtils中的函数，实现页面布局和

事件监听以及控件对象的注入。

(2) AnnoInjectUtils, IOC的核心, 这个类首先反射获取具体页面Activity对象的各个控件属性和方法, 并获取其中注解的值, 并根据获取到的值, 通过反射机制将具体的依赖对象赋值给当前引用。

(3) MyEventInvocationHandler, 通过Java动态代理机制实现的一个类, {61%:这个类继承自Java的InvocationHandler类, 并实现其invoke方法。}其作用是能够动态代理控件的点击事件, 长按事件, 触摸事件等等常用事件的监听。

通过IOC框架实现的效果如下:

图5-3 IOC实现控件以及页面布局的绑定

{49%:从上图我们可以看出, 使用IOC架构设计, 在一个Activity类上方使用我们定义的注解ContentView, 并将资源文件Id传入, }便可实现页面布局文件的注入, 同理, 我们使用ViewId注解, 并传入控件Id, 便可给页面需要的控件绑定绑定控件对象。使用上面非常的方便, 在节省了大量代码的同时, 也让代码更加整洁美观。

图5-4 IOC架构实现事件的动态绑定

从上图我们可以看出, 我们可以在Activity中定义一个方法, 并通过OnClick注解来实现点击事件的动态绑定, 同理, 在图5-1中出现的其他事件类, {40%:如OnLongClick, onTouch等等事件均可用同样的方式实现事件的动态绑定。}

5.2. 2Http请求的架构设计

上文介绍, APP采用OkHttp框架来实现客户端与服务端的数据交互。由于所有的页面几乎都需要调取接口, 所以我们对Http请求做了架构上的设计, 使APP的网络请求更方便快捷。其类图如下图所示:

图5-5 Http请求的架构设计类图

{44%:从上图可以知道, MyOkHttpClient类使用了单例模式, 提供OkHttpClient的单例对象给HttpRequest类。}HttpRequest类封装了6个方法实现互联网请求, 并通过反射HttpParam对象来获取请求参数。{51%:而所有页面的父类BaseActivity中获取了HttpRequest类的实例, }从而保证所有的页面都可以直接调用HttpRequest的6和方法之一请求接口。请求的参数只需要新建一个类继承自HttpParam类即可。响应的数据只需要创建一个类继承自ResultCallback, 并指定接口将会返回的数据类型, 框架会自动解析为相应的相应数据。

5.3APP的页面功能实现

5.3.1首页各页面的实现

(1) APP首页

APP首页上部分为轮播图, 首页有缴费的入口和小区动态、报修、交房验房、投诉等功能的入口。APP首页下面有热门邻里圈列表, 用户可以浏览访问量最大的邻里圈动态。

图5-6 APP首页

(2) 小区动态页面

小区动态页面可以浏览物业和居委会发布的各种消息，并可以点击阅读全文查看全文信息。

图5-7 小区动态页面

(3) 用户投诉页面

用户投诉页面可以上传图片，输入投诉的内容，并提交进行投诉。

图5-8用户投诉上传图片

图5-9用户投诉

(4) 用户报修

用户报修前需要登录，报修可进入APP相册选取图片，并可输入文字说明。

图5-10用户报修选取图片

图5-11用户报修页面

(5) 交房验房的页面如下图所示：

图5-12 交房验房

(6) 用户缴费

用户在登录成功后，APP首页点击缴费的类型可以进行缴费，缴费调用支付宝支付接口，相应的界面如下：

图5-13获取缴费列表

图5-14 缴费

5.3.2 邻里圈各页面的实现

(1) 邻里圈首页

邻里圈可以浏览周边动态，点击右上角的添加按钮可以发表动态，其页面效果如图5-15所示：

图5-15 邻里圈

(2) 发表邻里圈

发表邻里圈和投诉的操作类似，相应的页面效果如图5-16所示：

图5-16发表邻里圈

5.3.3 个人中心各页面的实现

(1) 个人中心首页

当用户未登录时，个人中心有按钮提示用户登录，当前用户显示“您好，游客”，相应的效果如图5-17所示：

图5-17未登录状态下的个人中心

当用户已经登录，则提示登录的文字消失，当前登录用户的头像右边显示登录用户的手机号，如图5-18所示：

图5-18登录状态下的个人中心

当用户向上滑动页面时，个人中心头部标题栏变为白色，相应的页面如图5-19所示：

图5-19个人中心滑动显示效果

(2) 用户注册

用户可以通过手机号注册为新用户，相应的页面如图5-20所示：

图5-20APP用户注册

(3) 用户登录

个人中心所有的操作只有在用户登录的情况下才可以使用，相应的页面如下图所示：

图5-21APP用户登录

(4) 个人中心查看缴费记录

登录用户可以在个人中心查看最近几个月的缴费记录，缴费记录可以查看日期，缴费金额，缴费类型，缴费状态等等，相应的页面如图5-22所示：

图5-22缴费记录页面

(5) APP用户完善资料

APP用户只有在完善资料的前提下，才可以进行缴费，交房验房等操作。用户完善资料需要填入真实姓名、身份证号、性别、出生年月、家庭人口数、婚姻状况、工作单位、职业、联系方式、小区名称、楼栋、单元和房号等信息，点击右上角的完成按钮会提交数据到服务器。相应的页面如图5-23所示：

图5-23APP用户完善资料

(6) 用户意见反馈，相应的页面如图5-24所示：

图5-24APP用户意见反馈

6结束语

{40%:本文通过分析国内物业管理行业的现状,由此找到了目前我国大部分小区物业服务的缺陷和不足。}经过细致的研究,从多个方面探讨,最终确定了APP的需求。在需求分析的基础上,利用UML建模工具进行细致的分析,确定了整个APP用户各个功能模块的时序图和流程,并设计出APP的页面原型。在得到APP详细设计之后,本文分析并实现了APP的架构设计。APP的架构设计整体上采用了控制反转(Inversion Of Control)的思想,实现了APP页面内容视图、各个页面控件对象的依赖注入,架构中使用到了很多Java反射和注解的知识。通过Java的动态代理机制,实现了页面内各个控件的监听事件代理。架构设计的第二个重点是对OKHttp框架的封装,在设计的过程中主要使用面向对象(OOP)的思想和单例模式等设计模式,设置了一些接口和超类,并利用反射实现了万能的Http请求类,大大简化了APP请求网络的代码。最后通过对各个页面的分析和实现,快捷地完成了各个页面的功能。

整体上来看,本APP有以下一些优点:

- (1) 界面设计合理,简洁,方便用户的操作。
- (2) 本APP通过使用加密算法对数据进行加密操作,大大提高了APP用户的数据安全性。
- (3) 本APP通过支付宝缴费,缴费安全系数高,并且各项支出都有详细的清单,用户一目了然。
- (4) 本APP通过小区动态和邻里圈模块的设计,在物业公司和住户,小区住户之间搭建了信息互通的渠道,提升了APP的趣味性和交互性。

与此同时,由于开发周期不长,开发模块较多等问题,本APP还存在运行不够流畅,数据比较单一等等问题,希望在日后的开发中能够逐步完善。

参考文献

- [1] 李宁.Android开发完全讲义(第三版).北京:水利水电出版社,2015 30-325
- [2] 郭霖.第一行代码 Android 第2版.北京:人民邮电出版社,2016 50 221
- [3] 毕小鹏.精通Android Studio.北京:清华大学出版社,2016 15-125

检测报告由PTcheck文献相似度检测系统生成

CopyRight 2007-2017 PTcheck