**黑龙江科技大学**

**工程实训报告**

**基于Android移动办公系统**

姓 名: 吕振斌

专业班级: 软件12-2班

**计算机与信息工程学院（软件学院）**

**2015.10.23**

# 摘 要

现阶段，房地产行业仍主导着中国的经济，各地房地产中介公司如雨后春笋般的出现在我们的视野中。在移动互联网快速发展阶段，传统的PC办公已经无法适应快节奏的生活，移动办公应运而生，它的理念是：办公人员可以任何时间在任何地点处理与业务相关的任何事情。在移动终端操作系统行业中，Android和IOS独领风骚，占据了整个市场的近89%的比重。所以打造一款面向房地产中介公司销售人员的基于Android移动办公系统势在必行，同时有一个非常好的发展前景。

**关键词**：房地产中介，PC办公，移动办公，Android，IOS

# Abstract

At present, the real estate industry is still dominated China economy, local real estate conduit company such as bamboo shoots after a spring rain as to appear in our field of vision. The stage of rapid development in the mobile Internet, the traditional PC office has been unable to adapt to the fast pace of life, mobile office emerge as the times require, the idea is: the office staff can at any time in any place and any thing related business. In the mobile terminal operating system in the industry, Android and IOS dominate, occupy the market share of nearly 89% . So to build a real estate sales staff of the conduit company for the mobile office system based on Android be imperative, also has a very good development prospects.

**Keywords**:real estate conduit,PC office, mobile office, Android, IOS

# 目 录

[第一章 绪论 1](#_Toc27233)

[1.1 研究的背景、主要内容及意义 1](#_Toc17989)

[1.1.1 研究背景 1](#_Toc4762)

[1.1.2 研究的主要内容 1](#_Toc7499)

[1.1.3 研究的意义 2](#_Toc16153)

[1.2 国内外相关研究情况 2](#_Toc8506)

[1.2.1 国外相关研究情况 2](#_Toc15527)

[1.2.2 国内相关研究情况 3](#_Toc7079)

[第二章 系统概述 4](#_Toc18116)

[2.1实现目标 4](#_Toc6468)

[2.2条件与限制 4](#_Toc25005)

[2.3 运行环境 5](#_Toc22123)

[2.3.1服务器端 5](#_Toc9525)

[2.3.2客户端 5](#_Toc11513)

[2.3.3开发环境 5](#_Toc1977)

[第三章 系统总体设计 7](#_Toc30208)

[3.1系统功能模块图 7](#_Toc29059)

[3.1总体功能设计 7](#_Toc9760)

[3.1.1系统设计目标 7](#_Toc25673)

[3.1.2总体设计 8](#_Toc24961)

[3.2数据库设计 9](#_Toc8114)

[3.2.1系统数据分析 9](#_Toc27844)

[3.2.2系统E-R图 9](#_Toc13068)

[3.3数据字典 14](#_Toc19052)

[第四章 系统详细设计与实现 19](#_Toc11664)

[4.1 客户端 19](#_Toc4231)

[4.1.1 系统登陆、注册、找回密码模块 19](#_Toc27114)

[4.1.2 任务管理模块 21](#_Toc8350)

4.1.3 客户管理模块 23

[4.1.4 公司管理模块 25](#_Toc5854)

[4.1.5 系统设置模块 26](#_Toc31801)

[4.2 服务端 28](#_Toc28228)

[第五章 测试结果和性能分析 33](#_Toc638)

[5.1 系统运行环境 33](#_Toc21680)

[5.2 系统测试 33](#_Toc1961)

[5.2.1 测试目的 33](#_Toc12157)

[5.2.2 测试方法 34](#_Toc32120)

[第六章 结论 37](#_Toc27671)

[6.1结论 37](#_Toc23483)

[6.1.1 本系统的贡献 37](#_Toc19129)

[6.1.2 本系统存在的不足 37](#_Toc12973)

[6.1.3 经验体会 38](#_Toc20476)

[致谢 38](#_Toc23739)

[参考文献 39](#_Toc7140)

# Contents

**[Chapter I Introduction](#_Toc22674)** [1](#_Toc22674)

[1.1 Background, main content and significance of the study 1](#_Toc12282)

[1.1.1 Research background 1](#_Toc1916)

[1.1.2 The main contents of research 1](#_Toc31799)

[1.1.3 Significance of research 2](#_Toc12020)

[1.2 Domestic and foreign related research 2](#_Toc7139)

[1.2.1 Foreign related research 2](#_Toc24013)

[1.2.2 Domestic research situation 3](#_Toc30043)

**[Chapter II System overview](#_Toc28801)** [3](#_Toc28801)

[2.1 Achieve the goal 3](#_Toc27790)

[2.2 Conditions and limitations 4](#_Toc22295)

[2.3 Operating environment 4](#_Toc17884)

[2.3.1 Server 4](#_Toc20254)

[2.3.2 Client 5](#_Toc20635)

[2.3.3 Development environment 5](#_Toc23175)

**[Chapter III System design](#_Toc7943)** [6](#_Toc7943)

[3.1 System function module diagram 6](#_Toc17009)

[3.1 Overall function design 7](#_Toc32358)

[3.1.1 System design goal 7](#_Toc11547)

[3.1.2 Overall design 7](#_Toc14765)

[3.2 Database design 8](#_Toc18584)

[3.2.1 System data analysis 8](#_Toc16871)

[3.2.2 System E-R Figure 9](#_Toc22524)

[3.3 Data dictionary 13](#_Toc29478)

**[Chapter IV System design and implementation](#_Toc7962)** [18](#_Toc7962)

[4.1 Client 18](#_Toc8662)

[4.1.1 System login, register, retrieve password module 18](#_Toc18701)

[4.1.2 Task management module 21](#_Toc15254)

4.1.3 Customer management module 23

[4.1.4 Management module 25](#_Toc2077)

[4.1.5 System setting module 26](#_Toc24594)

[4.2 Server 29](#_Toc25028)

**[Chapter V Test results and performance analysis](#_Toc14703)**  [33](#_Toc14703)

[5.1 System operating environment 33](#_Toc28634)

[5.2 System testing 33](#_Toc22851)

[5.2.1 Test purposes 33](#_Toc26681)

[5.2.2 Test method 34](#_Toc17975)

**[Chapter VI Conclusions](#_Toc1615)** [37](#_Toc1615)

[6.1 Conclusions 37](#_Toc1982)

[6.1.1 Contribution to the system 37](#_Toc3708)

[6.1.2 The shortcomings of this system 37](#_Toc10380)

[6.1.3 Experience 38](#_Toc8499)

**[Thanks](#_Toc11690)** [38](#_Toc11690)

**[Reference](#_Toc1068)** [39](#_Toc1068)

# 第一章 绪论

近几年，移动互联网得到了快速的发展，而移动终端也逐渐成为人们生活中不可或缺的一部分。据天猫官方网站统计，2014年双十一活动期间，总成交额为571亿，其中243亿来自于无线设备，占总成交额的42.6%，这一现象直接导致了移动应用的需求量迅猛增加。在移动终端的操作系统行业，Android和IOS独领风骚，占据了整个市场近89%的比重。

## 1.1 研究的背景、主要内容及意义

### 1.1.1 研究背景

现阶段，房地产行业发展的如火如荼，各大房地产中介公司纷纷进军市场；在移动终端设备操作系统市场上，Android和IOS牢牢掌控全局；随着4G互联网时代的到来，开发一款面向房地产中介公司的基于Android移动办公系统有很大的发展前景，不仅可以极大的方便销售人员及时了解公司、客户的情况，便于公司管理；而且客户的需求能及时得到反馈，符合人性化需求。

### 1.1.2 研究的主要内容

本系统将主要的功能分为4个大的模块，即任务管理模块，客户管理模块，公司管理模块，系统设置模块：

1. 任务管理：销售人员可以查看未被领取的任务信息，并且可以领取任务，任务领取后将不会被其他销售任何看到。
2. 客户管理：显示所有跟进过或者正在跟进的客户信息以及需求等。
3. 公司管理：会议信息（查看已发布的会议信息）；企业通讯录（以部门为单位显示公司所有员工的联系方式，并能通过用户点击条目快速进入拨号界面）；企业公告（查看已发布的公告信息）。
4. 系统设置：个人信息（查看个人信息并修改）；已领取任务（查看已经领取过的任务，可以取消任务）；关于（系统的相关信息）；修改密码（修改用户登录系统的密码）；退出登录（清除保存在本地的用户名和密码信息，返回到登录界面）

### 1.1.3 研究的意义

开发这款基于Android的移动办公系统是为了适应现代化快节奏的生活。有它作为媒介，销售人员可以在任何时间、任何地点进行办公，简化了复杂的办公流程，节约了时间成本。同时它也符合互联网的发展潮流。

## 1.2 国内外相关研究情况

### 1.2.1 国外相关研究情况

据相关统计，在美国，有70%以上的企业员工使用个人的智能终端设备接入企业数据实现移动办公，日常管理如批示工单、看邮件、查看工资等，都可以在移动终端实现；一些有外勤岗位的企业，对移动办公的需求更强。据广州APP开发公司别样网络统计数据显示，截至2015年，全球可见范围内的企业移动APP市场将达到125亿，移动办公系统的开发大有可为。

### 1.2.2 国内相关研究情况

在国内，华为紧紧跟着互联网发展的脚步，已推出了移动办公的解决方案，可以让我们更安全、更有效地进行移动办公。像国内大型的房地产中介公司——德佑地产就自己研发了一款面向企业内部员工的移动办公系统，目前已经投入使用。而针对中小型房地产中介公司却迟迟没有动静，但恰恰相反的是国内中小型房地产中介公司占有市场比重的非常重要的一部分，所以在这方面的移动办公的解决方案亟待提出并实施。

# 

# 第二章 系统概述

## 2.1实现目标

1. 公司员工能实时获得任务跟新内容，并能在新任务发布时在客户端获得推送消息。
2. 员工可以管理查看客户信息。
3. 员工可以查看公司的会议信息，公司所有员工的联系方式以及相关的公告信息。
4. 员工可以查看个人详细信息并能修改信息，修改登陆密码和退出登陆等功能。

## 2.2条件与限制

① 人力、资源方面的限制

② 进行系统方案选择比较的时间

③ 硬件、软件、运行环境方面的条件和限制

④ 可利用的信息和资源

⑤ 系统完成的截止时间

## 2.3 运行环境

### 2.3.1服务器端

服务器运行环境请见表2-1。

表2-1 服务器运行环境

|  |  |
| --- | --- |
| **软件产品** | **软件版本** |
| 操作系统 | Windows 7 64 bit |
| 应用服务器 | Tomcat 7.0 |
| 数据库 | MySQL Community Server 5.6.17 |
| 编程语言 | Java |

### 2.3.2客户端

客户端运行环境请见表2-2。

表2-2 客户端运行环境

|  |  |
| --- | --- |
| **软件产品** | **软件版本** |
| 操作系统 | Android 4.2.2 |
| 软件支持 | 需安装有邮件客户端（例如：QQ邮箱） |
| 网络支持 | 需要和服务器端（即PC端）连接统一WIFi |
| 软件安装需求 | 需设置允许安装未知来源的软件 |

### 2.3.3开发环境

系统开发环境请见表2-3。

表2-3 系统开发环境

|  |  |
| --- | --- |
| **软件产品** | **软件版本** |
| 操作系统 | Windows 7 64 bit |
| 应用服务器 | Tomcat 7.0 |
| 数据库 | MySQL Community Server 5.6.17 |
| 服务端编程语言 | Java |
| 客户端编程语言 | Java |
| Java虚拟机 | JDK 1.7.0\_70 |
| 开发工具 | Android Studio 1.3.2 |
| Android SDK | Android 4.2.2 |
| 版本控制工具 | GitHub |

# 

# 第三章 系统总体设计

## 3.1系统功能模块图

系统功能模块图如图3-1所示。



图3-1 系统功能模块图

## 3.1总体功能设计

### 3.1.1系统设计目标

该系统主要实现查看任务信息、会议信息、公告信息的服务功能，并把服务端提供的数据按需求友好的展现出来，提供客户端与服务端交互的空间，从而达到所需完成的要求，同时要保证客户端与服务端间的通信相对稳定，并且确保各个模块之间关联的合理性和应用系统有合理的框架结构。

### 3.1.2总体设计

通过服务端提供的相关信息，移动客户端通过用户发出请求想服务端请求服务而获取相关的数据信息，服务端与客户端的通信方式使用的是HTTP协议，封装在Volley框架中，交互使用的数据格式是JSON数据报，客户端接收到服务端传过来的JSON数据后，经过一系列的解析、转换成需要的数据后，友好的展示给用户。客户端与服务器交互简略流程图如图3-2所示。



图3-2 客户端与服务器交互简略流程图

## 3.2数据库设计

### 3.2.1系统数据分析

通过对移动办公系统的需求、总体架构方面的分析，可得出本系统中涉及到一下几个类实体：

1. 员工信息类实体（包括员工基本信息、个人基本信息等实体子项）；
2. 房源信息类实体（包括房子相关信息、房东相关信息等实体子项）；
3. 客户信息类实体（包括客户基本信息、客户需求等实体子项）；
4. 会议信息类实体（包括会议基本内容、开会地址等实体子项）；
5. 公告信息类实体（包括公告基本内容、发布日期等实体子项）。

### 3.2.2系统E-R图

1. 员工实体主要包括个人的基本信息、员工的基本信息以及个人的其他信息，实体属性如图3-3所示。

图3-3 员工实体属性



1. 房子实体包括房子的基本信息、小区部分信息和房东信息，实体属性请见图3-4所示。

图3-4 房子实体属性



1. 客户实体主要包括客户基本信息、看房时间等相关信息，实体属性请见图3-5所示。

图3-5 客户实体属性



1. 会议信息实体主要包括ID、会议主题、会议内容、开会地址以及会议开始和结束时间6个属性，实体属性如图3-6所示。

图3-6 会议信息实体属性



1. 公告信息实体主要包括ID、公告主题、公告内容、发布日期以及发布员工5个属性，实体属性请见图3-7。

图3-7 公告信息实体属性



1. 上述5个实体之间的关系：客户和房子之间是多对多的关系，一个客户可以选择多套房子，一套房子也可以被多个客户选中；员工和客户之间是一对多的关系，一个员工可以跟进多个客户，但一个客户只能有一个跟进员工；员工和房子之间是多对多的关系，一个员工可以负责多套房子，一套房子也可以被多个员工负责；员工和会议信息之间是多对多的关系，一个员工可以查看多条会议信息，一条会议信息可以被多个员工查看；员工和公告信息是多对多的关系，一个员工可以查看多条公告信息，一条公告信息可以被多个员工查看。具体的E-R图请见图3-8。

图3-8 系统E-R图



## 3.3数据字典

描述：员工基本信息表（emp\_info），记录所有在职员工的相关信息，具体如表3-1所示：

表3-1 员工基本信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名** | **字段中文名** | **字段类型** | **字段长度** | **属性** |
| emp\_nickname | 用户昵称 | VARCHAR | 50 | Unique |
| emp\_name | 员工姓名 | VARCHAR | 50 |  |
| emp\_sex | 员工性别 | VARCHAR | 50 |  |
| emp\_age | 员工年龄 | VARCHAR | 50 |  |
| emp\_phone\_no | 联系方式 | VARCHAR | 50 | Unique |
| emp\_email | 电子邮件 | VARCHAR | 50 | Unique |
| emp\_no | 员工工号 | VARCHAR | 50 | Primary Key |
| emp\_department | 所属部门 | VARCHAR | 50 |  |
| emp\_position | 员工职位 | VARCHAR | 50 |  |
| emp\_entry\_date | 入职日期 | DATE | 10 |  |
| emp\_borthday | 出生日期 | DATE | 10 |  |
| emp\_nation | 民族 | VARCHAR | 50 |  |
| emp\_identify | 身份证号 | VARCHAR | 50 | Unique |
| emp\_city | 所在城市 | VARCHAR | 50 |  |
| emp\_address | 家庭住址 | VARCHAR | 50 |  |
| emp\_password | 登陆密码 | VARCHAR | 50 |  |

描述：房源基本信息表（house\_info），记录了房源的相关信息，具体如表3-2所示：

表3-2 房源基本信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名** | **字段中文名** | **字段类型** | **字段长度** | **属性** |
| house\_no | 房源编号 | VARCHAR | 50 | Primary Key |
| house\_type | 房子类型（几房几厅） | VARCHAR | 50 |  |
| house\_area | 房子面积 | VARCHAR | 50 |  |
| house\_price | 租/售价格 | VARCHAR | 50 |  |
| house\_flag | 租/售标识（0：租/1：售） | VARCHAR | 50 |  |
| house\_city | 房子所在城市 | VARCHAR | 50 |  |
| house\_address | 房子详细地址 | VARCHAR | 200 |  |
| house\_green\_rating | 小区绿化面积 | VARCHAR | 50 |  |
| house\_property | 小区物业 | VARCHAR | 50 |  |
| house\_owner\_name | 房东姓名 | VARCHAR | 50 |  |
| house\_owner\_phone\_no | 房东联系方式 | VARCHAR | 50 |  |
| house\_emp\_no | 跟进员工号 | VARCHAR | 50 |  |
| house\_out\_flag | 交易是否完成（0：完成/1：未完成） | VARCHAR | 50 | Default 0 |

描述：客户基本信息表（roomer\_info），登记房客的基本信息，具体如表3-3所示：

表3-3 客户基本信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名** | **字段中文名** | **字段类型** | **字段长度** | **属性** |
| roomer\_no | 房客编号 | VARCHAR | 50 | Primary Key |
| roomer\_name | 房客姓名 | VARCHAR | 50 |  |
| roomer\_sex | 房客性别 | VARCHAR | 50 |  |
| roomer\_phone\_no | 房客联系方式 | VARCHAR | 50 |  |
| roomer\_email | 房客电子邮件 | VARCHAR | 50 |  |
| roomer\_house\_no | 所选房源编号 | VARCHAR | 50 |  |
| roomer\_date | 看房日期 | DATE | 10 |  |
| roomer\_period | 看房时间段 | VARCHAR | 50 |  |
| roomer\_rent | 房客需求 | VARCHAR | 50 | 0：租/1：买 |
| roomer\_complete | 交易是否完成 | VARCHAR | 50 | Default：0 |
| roomer\_emp\_no | 跟进员工工号 | VARCHAR | 50 |  |

描述：会议信息表（meeting），记录与会议相关的信息，具体如表3-4所示：

表3-4 会议信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名** | **字段中文名** | **字段类型** | **字段长度** | **属性** |
| meeting\_id | 会议编号 | INT | 4 | Primary Key |
| meeting\_theme | 会议主题 | VARCHAR | 200 |  |
| meeting\_content | 会议内容 | TEXT | 65535 |  |
| meeting\_address | 开会地址 | VARCHAR | 200 |  |
| meeting\_start | 会议开始时间 | DATETIME | 19 |  |
| meeting\_end | 会议结束时间 | DATETIME | 19 |  |

描述：公告信息表（notice），记录企业公告相关信息，具体如表3-5所示：

表3-5 公告信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名** | **字段中文名** | **字段类型** | **字段长度** | **属性** |
| notice\_id | 公告编号 | INT | 4 | Primary Key |
| notice\_theme | 公告主题 | VARCHAR | 50 |  |
| notice\_content | 公告内容 | TEXT | 65535 |  |
| notice\_emp\_no | 发布公告员工号 | VARCHAR | 50 |  |
| notice\_date | 公告发布日期 | DATE | 10 |  |

# 第四章 系统详细设计与实现

## 4.1 客户端

### 4.1.1 系统登陆、注册、找回密码模块

(1) 登陆（LoginActivity）

登陆界面是整个应用程序的主入口，用户输入已注册成功的用户名和密码来登陆系统，第一次登陆时会将用户名和密码存到本地，再次登陆时，只需点击登陆按钮。登陆界面如图4-1和4-2所示。



图4-1 登陆界面 图4-2登陆中的界面

(2) 注册（SMSRegisterActivity & RegisterActivity）

注册界面是通过在登陆界面点击“新用户”标签而跳转的页面，注册分为两步：

第一步是获取验证码，这一部分是用第三方的免费短信验证码SDK（具体请见：<http://mob.com/sms>），使用其开发的接口来随机获取4位数字验证码，点击下一步会对输入的验证码进行校验，校验成功则进入下一页面，失败则提示失败原因。短信验证界面如图4-3所示；

第二步是真正的注册，按照要求填写好相关信息后，客户端向服务端发送请求，服务端会先检查所输入的员工号在系统数据库中是否存在，如果存在，则允许该用户注册，否则，用户将注册失败；注册成功后会跳转到登陆界面，并且自动填写好用户名和密码信息。图4-3是用户注册界面。

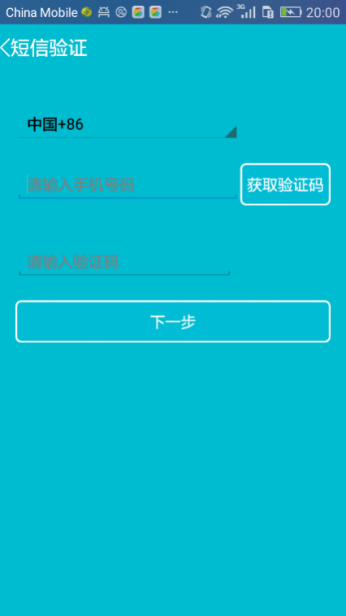


图4-3 用户注册界面 图4-4 找回密码成功界面

(3) 找回密码（BackPassword）

找回密码界面是通过点击登陆界面的“忘记密码？”标签而跳转的页面，帮助一些忘记了登陆密码的用户找回密码。按照提示输入用户的相关信息，点击“找回密码”按钮，客户端向服务端发送请求，服务端会先对输入的用户相关信息进行验证，如果有一项输入出错则提示用户输入的信息有误，保证用户密码的安全性；若输入的信息都正确，服务端会向客户端返回用户的登录密码，用Dialog显示，点击确定后，会进入登录界面。效果如图4-4所示。



图4-4 找回密码界面

### 4.1.2 任务管理模块

(1) 任务基本信息

任务基本信息包括房客需求、看房日期、时间段以及看房地址等基本信息，用户可以根据自身的情况选择不同的任务进行查看；这里显示出来的任务都是没有被领取过的任务，即没有跟进人的客户的需求。图4-5所示的即为任务基本信息界面。



图4-5 任务基本信息界面 图4-6 任务详细信息界面

(2) 任务详细信息

通过点击任务基本信息界面中的某一条目跳转到相应的详细信息界面，这里显示了客户的详细信息，用户可以通过点击领取任务按钮作为该任务的跟进人，领取任务后，其他人将无法看到这条任务，除非用户取消任务。图4-6为任务详细信息界面，图4-7为点击领取任务成功后的界面。



图4-7 任务领取成功界面

## 4.1.3 客户管理模块

(1) 客户基本信息界面

客户基本信息界面显示了客户的基本信息，包括客户名称、客户的联系方式、以及日期信息，排列方式是按照日期递减排列，可以通过点击某一客户的条目来查看相应客户的详细信息。客户的基本信息界面请见图4-8。

(2) 客户详细信息界面

客户详细信息界面是根据用户的需求来显示的，用户在客户基本信息界面选择查看某一客户的详细信息，就会跳转到相应客户的详细信息界面，包括客户的相关信息（客户个人信息、客户需求等）、房源相关信息（房子类型、地址等）、房东基本信息（姓名、联系方式等）；具体内容请见图4-9。用户可以通过点击“联系方式”条目来跳转到拨号界面，可以快速联系客户，简化了记电话号码这一复杂的步骤，图4-10所示为跳转到拨号界面的状态； 还可以通过点击“电子邮件”这一条目快速进入发邮件的界面，简化了发邮件的步骤，图4-11为跳转到发邮件界面的状态；



图4-8 客户基本信息界面 图4-9 客户详细信息界面

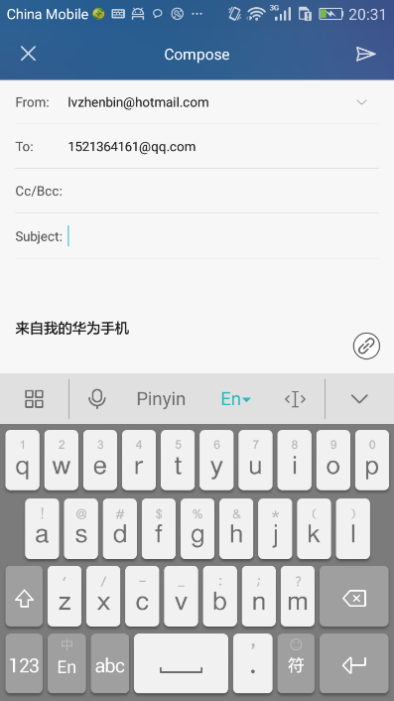
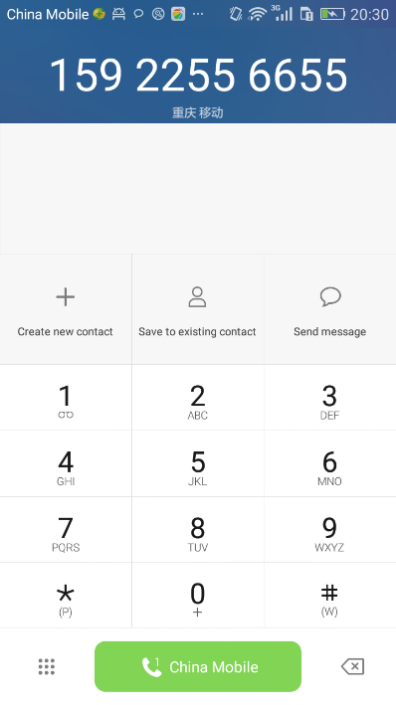


图4-10 跳转到拨号界面 图4-11 跳转到发送邮件界面

### 4.1.4 公司管理模块

公司管理模块分为三大部分，包括会议信息，企业通讯录以及企业公告这三个模块

1. 会议信息模块。目前未实现。
2. 企业通讯录模块。目前未实现。
3. 企业公告模块包含两部分，一部分是显示公告的基本信息，另一部分是公告的详细信息。目前未实现

### 4.1.5 系统设置模块

系统设置模块主要分为六个分模块，分别为个人信息模块、已领取任务模块、地图功能模块、关于、修改密码和退出登陆，具体请见图4-18。

1. 个人信息

个人信息界面主要显示当前用户的相关信息，用户可以查看和修改自己的信息，但是输入与公司相关的一些信息（例如员工号、所属部门等信息是不能被修改的）。图4-12所示为查看用户信息界面，图4-13为编辑/修改用户信息界面。



图4-12 个人信息界面 图4-13 编辑/修改个人信息界面

1. 已领取任务

已领取任务界面显示了用户在任务管理界面已经领取过的任务信息，进入详细信息界面后，可以查看任务的状态和取消任务。图4-15为已领取任务基本信息界面，图4-16为已领取任务的详细信息界面。

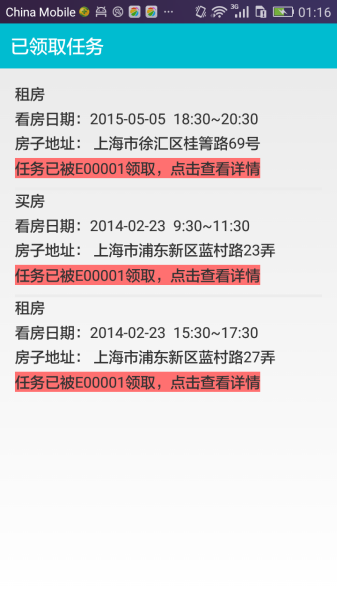


图4-15 已领取任务基本信息界面 图4-16 已领取任务详细信息界面

1. 关于

关于界面是显示与本系统相关的信息，包括系统的版本信息，常见问题和服务协议。详情请见图4-15。



图4-15 关于界面 图4-16 修改密码界面

1. 修改密码

在修改密码界面上，用户根据提示输入相关信息，服务端对输入用户名和旧密码与数据库中的信息对照，如果输入的信息正确则将密码改为新密码，否则密码修改不成功，修改成功后会跳转到登陆界面，提示用户重新登陆。图4-16为修改密码界面。

1. 退出登陆

当用户点击退出登陆时，系统跳转到用户登陆界面，并且清除用户已经保存在本地的用户名和密码等信息，当再次登陆时，用户名和密码框中的内容会置空，需要用户重新填写。图4-17所示为用户退出登陆后显示的内容。



图4-17 退出之后的界面

## 4.2 服务端

本次实训课题的重点在客户端而不是服务端，所以对服务端的投入不是很大，主要是处理客服的请求，服务端的界面未实现。JavaEE作为Sun公司主推的平台，有大量官方和非官方的资料可供参考，而且有大量成熟的开源框架可供选择，所以而本课题相的服务端，使用JavaEE平台来操作数据库。

本次服务端分为两大部分，一部分是客户端对服务端发出请求、服务端对客户端的请求作出相应的响应，另一部分是作为测试使用，即向数据库中添加测试数据。

4.2.1 服务端作为客户端响应部分

这一部分的主要功能就是通过用户在客户端的相关操作，向服务端发出指令，服务端对这些指令作出用户所需要的响应（这里响应主要是指对数据库进行增删改查的操作）。经过相关研究，有两套方案可供选择：

1. 自己编写一个用户连接服务端的工具类，这需要借助于HttpClient/HttpURLConnection类来实现，具体实现代码片段如下：

HttpParams params = new BasicHttpParams();

//设置连接超时时间

HttpConnectionParams.setConnectionTimeout(params, 50000);

HttpClient client = new DefaultHttpClient(params);

HttpPost request = new HttpPost(url);

StringEntity se = new StringEntity(json.toString(), "UTF-8");

se.setContentEncoding("UTF-8");

se.setContentType("application/json");

request.setEntity(se);

request.setHeader("json", json.toString());

HttpResponse response = client.execute(request);

//获取服务器返回的数据

if(response.getStatusLine().getStatusCode()== HttpStatus.SC\_OK){

String res = EntityUtils.toString(response.getEntity(), "utf-8");

Log.d("httpResponse", res);

Message msg = new Message();

msg.what = 0;

Bundle bundle = new Bundle();

bundle.putString("res", res);

msg.setData(bundle);

handler.sendMessage(msg);

}

这样的确可以实现与服务端的交互问题，但对于数据安全、网络安全等的问题并没有纳入考虑范围之内，同时，HttpCient和HttpURLConnection的用法相对来说比较复杂，如果没有做适当的封装，很容易出现代码冗余的问题，所以此举不是最优选择。经过多方资料的查找，发现了一款和适合的框架——Volley框架。

1. Volley框架实现客户端与服务器的交互

考虑到上述的问题，一些Android网络通信框架也就应运而生，例如AsyncHttpClient，他把所有网络通信的细节全部封装到框架内部，不需要使用者了解它的原理以及一些复杂的优化操作，只需要简单的调用几行代码就可以轻松完成网络通信操作；在比如Universal-Image-Loader框架，它的主要功能是极大的简化了在界面上显示网络图片的操作，开发者不用关心如何才能从网络上获取图片，也不用关心开启线程、回收图片资源等的细节问题，因为这些操作已经被封装在了Universal-Image-Loader框架中。

Volley是在2013年的Google I/O开发者大会上推出的一款新的网络通信框架，它集合了AsyncHttpClient和Universal-Image-Loader两者的优点，即可以像AsyncHttpClient非常简单的就能进行HTTP通信，又可以和Universal-Image-Loader一样轻松的加载网络上的图片。（注：Volley在性能方面也进行了大幅度的调整，它的设计目标就是非常适合的去进行数据量不大但网络通信频繁的操作，但对于大数据量的网络操作，比如说下载文件等，Volley的表现就会非常糟糕）。

使用Volley非常方便，只要简单的三步即可完成复杂的网络通信操作：

1. 创建一个RequestQueue对象，具体代码如下：

RequestQueue mQueue = Volley.newRequestQueue(context);

1. 创建一个JsonRequest对象，具体代码如下：
2. JsonObjectRequest jsonObjectRequest = new JsonObjectRequest(LOGIN\_URL, json,

new Response.Listener<JSONObject>() {

@Override

public void onResponse(JSONObject response) {

Log.d("TAG", response.toString());

}

}, new Response.ErrorListener() {

@Override

public void onErrorResponse(VolleyError error) {

Log.e("TAG", error.getMessage(), error);

} });

创建JsonRequest（JsonObjectRequest/JsonArrayResquest）对象时，有四个参数：第一个参数是String类型的url，即需要访问网络的IP地址或者域名（本课题使用的是IP地址）；第二个参数是JSONObject类型的JSON数据，它的作用是向服务端传递一些数据作为操作数据库时的条件；第三个参数是Listener<JSONObject>类型的监听器，监听从服务端返回的信息，同时内部从写了onResponse()方法，该方法中接收一个JSONObject类型参数response，reponse中的信息即为服务端向客户端返回的信息，在该方法中可以作更新UI的操作；最后一个参数是ErrorListener类型的监听器，主要作用是用于监听当出现无法访问服务端时的问题，同样重写了内部的方法onErrorResponse()，该方法接收VolleyError类型参数error，用于显示无法访问服务端的具体原因，如：IP地址找不到，网络未连接……

1. 将JsonRequest对象添加到RequestQueue中，具体代码如下：

mQueue.add(jsonObjectRequest);

当完成这三步之后，最难解决的网络通信问题也就引刃而解了。

# 第四章 测试结果和性能分析

## 5.1 系统运行环境

经过对本系统的详细分析和测试，建议系统的运行环境应具有以下配置，以确保系统能够稳定运行：

1. Intel(R) Core(TM) i3以上的CPU，2G以上的内存，100G以上的硬盘。
2. Android手机或模拟器，搭载Android4.0以上的系统（最优建议是Android 4.2.2、5.0版本）。

## 5.2 系统测试

### 5.2.1 测试目的

一个系统或者应用开发完成后，对系统或应用做相关测试是必不可少的环节，这一环节目的有以下几点：

1. 从用户的角度出发，希望在对系统或应用的测试环节中找出存在的错误和还需完善的地方，例如：用户界面的体验是否合理，控件是否分布合理、界面展示是否友好美观，页面操作是否符合大众用户的使用习惯，页面在加载数据时是否会出现页面闪烁的情况，点击页面是是否会有明显的反应迟钝等一些不利于用户友好体验的问题，最最重要的是产品能否稳定、流程的运行。
2. 从软件开发者出发，校验该软件是否能很好的达到用户的需求，应用系统对网络通信封装的是否合理，网络发生异常时的处理是否合情合理，控件加载数据时是否存在明显的问题，应用系统是否会因为一些数据的不恰当处理致使软件闪退，测试软件运行时内存是否会不断地增加，导致内存溢出（OutOfMemory）的问题，测试是会有页面上显示数据相互叠加的情况出现或者页面数据加载出错，测试系统中是否会有一些在前期需求分析或者开发中没有考虑到的问题，确保软件质量过关。

### 5.2.2 测试方法

对该应用系统使用的测试方法有功能测试、系统稳定性测试和压力测试，最后对代码进行整改，确保代码质量，减少代码冗余。

1. 对整个应用程序的使用流程做一个总体的测试，发现应用中存在的不足或者还有待优化的地方，例如：用户界面体验不友好，页面跳转不流畅有明显的卡顿现象，用户操作不合理、不够人性化。
2. 对每个功能是否能够完整、合理的实现，例如：企业通讯录中分类显示是否正确。测试各个功能是否能够达到需求，测试数据是否对应用系统的稳定性有影响。
3. 测试应用界面上数据加载时界面友好性以及数据的完整性。例如：加载页面时是否会出现闪屏、加载时等待页面是否友好。对页面数据从开始加载到加载完成所用时间是否过长，考虑产生该问题的原因，如何优化这一问题，是否可以通过减少传输的数据量来提高页面加载速度减少加载数据用时。
4. 测试时，不间断的去加载页面上的数据，观察应用程序的内存使用情况，是否会出现异常（通常是OutOfMemory异常）或占用系统内存过大的情况出现。
5. 压力测试，使用的是SDK中附带的一个测试工具——Monkey，该工具用于压力测试，然后开发人员结合monkey打印的日志和系统打印的日志信息，解决测试中出现的问题，它的特点是所有事件都是随机产生的，不带有任何个人的主观性。测试结果如图5-1所示：

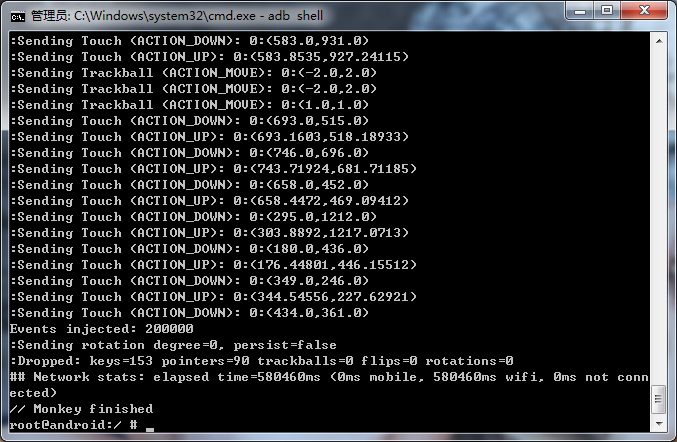


图5-1 monkey测试结果示意图

5.2.3 测试结论

对移动办公系统做了功能、兼容性、安全性、可使用性、可靠性等方面的测试，情况如下：

1. 整个系统的操作流程与需求一致，应用布局合理、基本上符合普通大众的操作习惯，操作时不会出现闪屏、卡顿等影响用户体验等问题。
2. 完整的实现了需求上的功能模块，各模块能够友好的交互。
3. 在网络通畅的情况下，应用系统加载数据所用时长控制在合理、用户可接受的范围之内。
4. 在不停的向加载页面上的数据时，没有出现其所占系统内存不断增加，而出现内存溢出等问题导致应用闪退情况出现。
5. 压力测试时，让应用系统产生20万次随机事件，系统没有出现卡顿或者闪退的情况，系统占用内存也在合理的范围之内，没有超出预算。

测试结果表明，基于Android移动办公系统达到了需求中的要求，并具有以下特点：

1. 系统结构清晰明确，采用B/S架构，可满足基本使用要求，使用了一些稳定的框架；
2. 系统可靠性、稳定性较好，对网络环境弱的情况有相关的异常捕获；
3. 系统兼容性好，可以在不同版本的系统上顺畅运行；
4. 测试结论：通过。

# 第五章 结论

## 6.1结论

### 6.1.1 本系统的贡献

本系统简化了在外奔走的房地产中介公司的销售人员的办公流程，让他们可以节省时间去更好的为客户服务，同时可以实时得到任务信息和客户相关信息，便于及时与客户取得联系，不会因为不能及时联系客户而导致客户的流失。

### 6.1.2 本系统存在的不足

1. 在需求方面，由于前期的需求分析由于资源和时间的限制问题，无法大规模的做市场调研，导致系统的功能性问题与实际中有偏差，且系统中功能有待优化的地方还有很多，这与个人的见识以及所属行业等有关，导致功能的不完善；
2. 在数据安全方面，限于现阶段的知识面以及时间成本，并没有对其做一个系统地、完整性地规范；
3. 在用户体验上，由于对于美工这一方面没有研究，审美观只是停留在大致的观感上，并没有做出太细致的优化，同时也没有太多的时间对界面进行审美优化；
4. 在系统测试方面，目前掌握的测试技术并没有涉及到很深的层面上，所有只是粗浅的做了压力测试；
5. 在屏幕适配上，由于本系统为了简便性，没有对平板做适配，导致在平板上运行在用户体验上会出现问题。

### 6.1.3 经验体会

在设计与实现系统的过程中，遇到了许多的问题，经过多方求助、网上查找相关资料最终攻克了这些难题；该系统是参考相关资料自己独立开发出来的，并没有假借他人之手，这培养了自己的动手能力，并对Android应用开发有了一个更深层次的理解，熟悉了以前没有接触过的相关知识。

# 致谢

首先感谢我的毕业导师兰翠老师，为我的工程实训设计上耐心的指导。在该设计过程中，从功能设计、实现，数据库设计，论文格式等方面提出具有建设性的意见，对设计工作的顺利进行有很大的帮助。

在论文设计过程中，得到了同学们、同事们和从事房地产行业朋友们的帮助，他们积极提出一些建设性的意见，在此致上我最诚挚的谢意。在此，感谢黑龙江科技大学能给我这样一个平台去完成学业，提供了优质的教学资源和师资力量。

# 参考文献

1 任玉刚. Android开发艺术探索. 北京：电子工信出版社，2015-09

2 林学森. Android深入理解Android内核设计思想. 北京：人民邮电出版设，2014-05

3 孟远. Android 网络通信框架 Volley 的解析和比较[J]. 软件， 2014-12

4 高静, 段会川. JSON 数据传输效率研究[J]. 计算机工程与设计，2011-07

5 尤卫军. 移动办公平台的实现方式★[J]. 计算机时代, 2012, 2: 35-36

6 罗军舟, 吴文甲, 杨明. 移动互联网: 终端, 网络与服务[J]. 计算机学报, 2011, 34(11): 2029-2051

7 Murphy M. Communicating via the Internet[M] Beginning Android 3. Apress, 2011: 377-392

8 董川. 基于 Android 系统的移动终端稳定性测试方法的研究与实践[D]. 北京邮电大学, 2012

9 Nurseitov N, Paulson M, Reynolds R, et al. Comparison of JSON and XML Data Interchange Formats: A Case Study[J]. Caine, 2009, 2009: 157-162

10 郭霖. 第一行代码Android[M]. 北京: 人民邮电出版社,2014-08

11 Murphy M. Communicating via the Internet[M] Beginning Android 3. Apress, 2011: 377-392

12 董川. 基于 Android 系统的移动终端稳定性测试方法的研究与实践[D]. 北京邮电大学, 2012

13 李刚. 疯狂Android讲义（第二版）[M]. 北京: 电子工业出版社,2013-03

14 孙鑫. Java Web开发详解. 北京: 电子工业出版社,2012-05

15 邓子云. JSP网络编程从基础到实践(第二版). 北京: 电子工业出版社,2007-05