

|  |  |  |
| --- | --- | --- |
| **中工**  **信商** |  | **2018-JX16-** |
|  |

**本科毕业论文（设计）**

**基于人证核验的共享酒店住宿信息管理系统——用户端APP**

|  |  |
| --- | --- |
| **系 （部）** | 信息技术系 |
| **专 业** | 网络工程 |
| **学 号** | 201401024234 |
| **学生姓名** | 李金刚 |
| **指导教师** | 吴 颖 |
| **提交日期** | 2018年5月10日 |

摘要

酒店预订客户端是无线互联网和旅游行业快速发展、相互融合的产物，随着人民生活水平的不断提高，人们对于商务出行、旅游出行等的需求也越来越强烈，酒店预订是人们计划出行所必须经历的事情之一，具有广阔的市场前景，也可以给旅游公司与酒店带来丰厚的商业机遇，设计出一款方便好用的酒店预订客户端具有重要的意义。

本人参与设计并实现了一款国际酒店预订客户端系统，其基于Android平台，使用Java语言编写并釆用结构设计，进而提高了系统的灵活性与可扩展性，后台服务器釆用语言编写，本人主要负责客户端开发部分。经过对产品特点以及用户特征的分析，本文提炼出了系统的需求，该系统主要包括用户登登录、酒店搜索、酒店详情、订单管理、系统设置等五大功能模块。用户登录模块提供了用户登录与用户注册等功能。酒店搜索模块提供了酒店的搜索功能，用户可根据入住退房日期、目的地城市、关键字等信息对酒店信息进行检索，并可根据位置、价格等条件进行排序。酒店详情模块提供了酒店详情信息的查看，用户可对酒店进行点评、预订酒店，通过订单管理模块用户可査看已定好的酒店的订单信息。系统设置模块实现了第三方应用推荐、意见反馈、查看软件版本信息等功能。

关键词：酒店；预订；旅游；人脸识别；共享酒店

# 目录

[1 引言 1](#_Toc513652065)

[1.1 研究背景及意义 1](#_Toc513652066)

[1.2 本选题在国内外的研究状况及发展趋势 1](#_Toc513652067)

[2 相关技术介绍 3](#_Toc513652068)

[2.1 Android Studio简介 3](#_Toc513652069)

[2.2 MVPArms框架简介 4](#_Toc513652070)

[3 需求分析 5](#_Toc513652071)

[3.1 系统可行性分析 5](#_Toc513652072)

[3.2 业务需求分析 5](#_Toc513652073)

[4 总体设计 6](#_Toc513652074)

[4.1 业务设计 6](#_Toc513652075)

[4.1.1 登录注册模块 6](#_Toc513652076)

[4.1.2 酒店模块 7](#_Toc513652077)

[4.1.3 订单模块 8](#_Toc513652078)

[4.1.4 用户信息模块 8](#_Toc513652079)

[4.2 界面设计 9](#_Toc513652080)

[4.2.1 主界面 9](#_Toc513652081)

[4.2.2 酒店列表 10](#_Toc513652082)

[4.2.3 酒店详情界面 10](#_Toc513652083)

[4.2.4 登录注册界面 11](#_Toc513652084)

[4.2.5 人证核验界面 11](#_Toc513652085)

[5 系统实现 12](#_Toc513652086)

[5.1 登录注册模块 12](#_Toc513652087)

[5.1.1 注册模块 12](#_Toc513652088)

[5.1.2 登录模块 12](#_Toc513652089)

[5.1.3 人证核验 13](#_Toc513652090)

[5.2 酒店模块 14](#_Toc513652091)

[5.2.1 酒店搜素 14](#_Toc513652092)

[5.2.2 酒店列表 15](#_Toc513652093)

[5.2.3 查看酒店详细信息 15](#_Toc513652094)

[5.2.4 获取酒店房型列表 16](#_Toc513652095)

[5.3 订单模块 17](#_Toc513652096)

[5.3.1 在线预订 17](#_Toc513652097)

[5.3.2 查询订单 18](#_Toc513652098)

[5.4 用户信息模块 19](#_Toc513652099)

[6 系统测试 20](#_Toc513652100)

[6.1 测试目的 20](#_Toc513652101)

[6.2 测试方法 20](#_Toc513652102)

[6.3 功能测试 20](#_Toc513652103)

[6.3.1 用户登录测试 21](#_Toc513652104)

[6.3.2 用户注册测试 21](#_Toc513652105)

[6.3.3 人证核验测试 21](#_Toc513652106)

[6.3.4 搜索酒店 22](#_Toc513652107)

[6.3.1 预订酒店 22](#_Toc513652108)

[7 总结与展望 23](#_Toc513652109)

[7.1 总结 23](#_Toc513652110)

[7.2 展望 24](#_Toc513652111)

[参考文献 25](#_Toc513652112)

[致谢 26](#_Toc513652113)

# 引言

## 研究背景及意义

在这个移动互联的时代，人们急切需求一款可以足不出户、随时随地通过自己的手机来浏览并预订快捷酒店的住宿信息管理系统。然而在当前市场上的一些应用及酒店入住流程太复杂，用户体验和隐私效果较差，为此开发一款简单、实用的基于人证核验的共享酒店住宿信息管理系统具有实际意义。

本系统基于共享经济和人工智能，利用共享经济的优势，让用户可以快速，方便入住酒店。然后结合人工智能的人脸识别算法，验证人脸和身份证信息来保证安全，可靠。最终实现方便用户的人证核验的共享酒店住宿信息管理系统。

## 本选题在国内外的研究状况及发展趋势

共享经济，一般是指以获得一定报酬为主要目的，基于陌生人且存在物品使用权暂时转移的一种新的经济模式。其本质是整合线下的闲散物品、劳动力、教育医疗资源。有的也说共享经济是人们公平享有社会资源，各自以不同的方式付出和受益，共同获得经济红利。此种共享更多的是通过互联网作为媒介来实现的。在北京、广州、杭州等多个城市，继共享单车、共享汽车之后，共享充电宝、共享篮球、共享雨伞等共享经济新形态不断涌现，并成为新一轮资本蜂拥的“风口”。仅以共享充电宝为例，短短40天时间就获得11笔融资，近35家机构介入，融资金额约12亿元人民币。

人脸识别，是基于人的脸部特征信息进行身份识别的一种生物识别技术。用摄像机或摄像头采集含有人脸的图像或视频流，并自动在图像中检测和跟踪人脸，进而对检测到的人脸进行脸部的一系列相关技术，通常也叫做人像识别、面部识别。人脸识别系统的研究始于20世纪60年代，80年代后随着计算机技术和光学成像技术的发展得到提高，而真正进入初级的应用阶段则在90年后期，并且以美国、德国和日本的技术实现为主；人脸识别系统成功的关键在于是否拥有尖端的核心算法，并使识别结果具有实用化的识别率和识别速度；“人脸识别系统”集成了人工智能、机器识别、机器学习、模型理论、专家系统、视频图像处理等多种专业技术，同时需结合中间值处理的理论与实现，是生物特征识别的最新应用，其核心技术的实现，展现了弱人工智能向强人工智能的转化。

将人脸识别和共享酒店结合起来的状况并不多见，所以这是一个新的课题。

共享酒店是传统酒店的升级版，在保留传统酒店的同时开发新大陆，同时互联网智能平台也是是大势所趋。将共享经济和人脸识别结合起来，将会擦除新时代不一样的火花。随着酒店市场竞争推动，竞争越来越激烈，只有酒店赶上新形势，只有更新改造、升级换代，才能不落伍。而共享酒店的人脸识别没有身体接触，不像指纹机那样要接触皮肤；人脸识别让人有自尊感，按手指那是强制式的，会让人感觉到是被操作的。

# 相关技术介绍

基于人证核验的共享酒店住宿信息管理系统——酒店端APP的开发利用Android Studio为开发工具，使用Java语言开发和实现。主要运用C/S模式，具有友好的界面，便于用户的操作和使用，同时方便与后台进行数据交互。系统采用MVP模式开发，使得系统的运行代码变的非常的简洁、高效、降低耦合，方便测试等。

## Android Studio简介

Android Studio是Google于2013 I/O大会针对Android开发新推出的开发工具，目前很多开源项目都已经在采用，Google的更新速度也很快,明显能感觉到这是Android开发的未来。它是Google推出的毫无疑问，这个是它的最大优势，Android Stuido是Google推出，专门为Android“量身订做”的，是Google大力支持的一款基于IntelliJ IDEA改造的IDE，这个应该能说明为什么它是Android的未来，其中相对于其他Android开发工具它有八大优点，其中包括：

（1）速度更快

Eclipse的启动速度慢、响应速度慢、内存占用大等问题一直是其诟病，相信大家这点应该深有体会，而且经常遇到卡死状态。Android Studio不管哪一个方面都全面领先Eclipse。

（2）UI更漂亮

I/O上演示的那款黑色主题很受欢迎，被称有“极客范”，Stuido自带的Darcula主题的炫酷黑界面实在是“高大上”，相比而言Eclipse下的黑色主题太low了。

（3）更加智能

提示补全对于开发来说意义重大，Studio则更加智能,智能保存，从此再也不用每次都 Ctrl + S了。熟悉Studio以后效率会大大提升。

（4）整合了Gradle构建工具

Gradle是一个新的构建工具，自Studio亮相之处就支持Gradle，可以说Gradle集合了Ant和Maven的优点，不管是配置、编译、打包都非常棒。

（5）强大的UI编辑器

Android Studio的编辑器非常的智能，除了吸收Eclipse+ADT的优点之外，还自带了多设备的实时预览，相对Android开发者来说简直是神器啊。

（6）内置终端Studio内置终端

（7）更完善的插件系统

Studio下支持各种插件，如Git、Markdown、Gradle等等，你想要什么插件，直接搜索下载。

（8）完美整合版本控制系统

安装的时候就自带了如GitHub、Git、SVN等流行的版本控制系统，可以直接check out你的项目。

## MVPArms框架简介

MVPArms是一个集成了大量Android主流框架，并且全部使用Dagger2管理，以及提供API将所有库连接起来方便使用，还提供详细的Wiki文档。

它可以使开发后面的所有项目都不用重复的复制粘贴，一个依赖省去很多烦恼，而且对于新手来说这些框架难的不仅仅是API的使用，更难的是怎么把它们结合到一起，应对各种场景的使用。

对于一个新的Android项目，特别是熟练使用Dagger2和Rxjava的开发者，他们只需要将此项目Clone下来，Demo只实现了一个页面，将此页面删除掉，添加所需要的Retrofit API，框架就搭建好了，然后就可以直接使用Demo进行后续的开发，包结构也适合后面的扩展。

# 需求分析

## 系统可行性分析

基于人证核验的共享酒店住宿信息管理系统——用户端APP的开发利用Android Studio为工具，用Java语言设计和实现主要运用C/S模式，系统采用MVP模式，MVP是从经典的模式MVC演变而来。在Android开发应用中，MVP的结构划分：视图（View）负责绘制UI元素、与用户交互，在Android开发中对应于Activity相关的类；模型（Model）类似于数据加工处理厂，负责对数据的获取，数据的解析，数据的存储，数据的分发，数据的增删改查等操作；表示器（Presenter）作为View与Model交互的中间纽带，处于MVP的中间层，表示器会把视图递交的命令进行一定的校验等操作，然后交给模型层处理，模型层处理完数据之后，会通知表示器，表示器主动去获取数据处理的结果递交给视图层显示。因此表示器有封装业务，更新UI界面和持有线程等功能。

## 业务需求分析

用户端APP模块主要功能：

注册模块：通过用户APP与后台进行通信，包含身份证号、姓名、手机号、人脸拍照、验证码、复杂密码等信息，完成用户注册。

登录模块：通过手机号码、密码完成用户登录。

酒店搜索：可以通过关键字，目的地，搜索酒店，通过位置、价格、评分等对酒店进行排序。

酒店和房间详情：通过酒店ID查询酒店的详细信息和房间类型列表

订单模块：用户选择对应的房间号进行下单，生成订单。

人证核验：通过用户APP与后台进行通信，用手机拍照用户本人和身份证，通过图像识别接口，确定旅客本人。

我的模块：可获取个人信息、客服中心、退出账户等。

查看订单：通过用户APP与后台进行通信，可以获取全部订单、待入住订单、代付款订单、待评价订单。

# 总体设计

## 业务设计

通过对业务需求的详细分析，对系统的业务进行分析，主要分登录注册模块，这一模块主要进行用户信息的身份验证，包括实名认证和人脸识别，其次就是酒店搜索模块，这一模块用户可以通过目的地，关键字，酒店名称进行搜索，然后可以用户可以对搜索结构进行排序。接着就是订单模块，用户选择酒店房间进行下单，然后查看订单信息等。最后就是用户信息模块，在此模块用户可以查看自己的信息，获取帮助等。没有进行实名认证的用户可以在此进行实名认证。进行退出登录等操作。模块分析图如下图4-1所示

图4-1 系统业务模块图

### 登录注册模块

登录注册模块是一个系统的基本模块，在本系统中，用户只有登录注册之后才能预订酒店。没有登录的用户只能浏览酒店信息和房间类型信息。用户可以通过手机号和密码进行登录，登录之后会保存登录状态，用户可以进行预订酒店和查看个人信息。注册模块需要用户提供手机号、用户名的密码，验证手机号的正确性，并且一个手机号只能注册一次。注册成功之后直接返回的登录信息，不需要再次登录。成功注册的用户需要进行身份认证，也就是本系统的人证核验，需要用户上传身份证照片，对身份证进行识别，提取人脸信息，接着调用相机检测人脸，对比两个人脸信息，正确上传到服务器，保存信息。该模块流程如下图4-2所示

图4-2 登录注册模块流程图

### 酒店模块

酒店模块是该系统的最主要模块，该模块又分为搜索酒店、获取酒店列表、获取酒店详细信息等。主要流程如下图4-3所示

酒店搜索：

用户可以通过目的地和关键字来搜索酒店。首先进入主界面，获取用户的当前位置，显示在目的地一栏，然后入住时间默认为今天，离店时间默认为明天。酒店搜索分为两个业务。一个是搜索附近的酒店，一个是搜索其它城市的酒店。如果用户目的地使用默认位置，就是附近位置，就会查找附近的酒店。如果用户修改了目的地并且和当前城市不一样，就会搜索目的地的酒店。

酒店列表：

接着就会进入酒店列表界面，会显示对应的结果，并且可以在当前界面进行排序。可以通过价格、评分、销量、距离进行排序。每次选择排序后，都会重新发请求到服务器，然后服务器会返回对应的数据，最后显示在界面。

酒店详情：

用户点击酒店列表会进入酒店详情界面，酒店详情界面主要展示酒店的详细信息，主要包括酒店的图片、评分、详细地址和该酒店的房型信息。用户还可以在当前界面选择入住时间。点击房型列表可以查看当前的房型的详细介绍，点击预订按钮会显示可预订的房间号。



图4-3 酒店模块流程图

### 订单模块

订单模块主要分为生成订单和查询订单，生成订单是用户选择对应的房间后，服务器会生成订单信息，发送到客户端进行确认，如果用户确认订单后，会生成订单，并进入付款界面，用户选择付款后，预订成功。查询订单是用户查询自己的订单，可以查看全部订单、待入住订单、代付款订单和待退房订单。

### 用户信息模块

用户信息模块是用来查看用户的个人的信息，主要包括用户名、手机号和是否实名认证等信息。如果没有实名认证可以进行实名认证。

## 界面设计

界面作为一个系统的门户，是直接和用户息息相关的，所以一个APP的最重要的就是界面设计，也就是UI设计。本系统本着简洁大方，实用的基本原则，APP大致分为主界面，酒店列表，酒店详细，登录界面，注册界面，实名认证界面。

### 主界面

主界面是用户看到的第一个界面，也是用户使用和点击最多的界面。所以主界面采用底部导航的方式展现给用户，主要分为三个模块，分别是首页、订单和我的。考虑到本系统主要是进行预订酒店，所以查找酒店的操作放在了首页。可以方便用户查找和预订酒店。首页主要包括了五个控件，目的地和关键字可以输入作为搜索条件。点击我的位置可以获取当前位置，点击入住时间会弹出日历控件来选择入住时间。点击搜索按钮则会根据当前条件进行酒店搜索。订单界面主要用来显示用户的订单信息，方便用户查看订单信息。在此界面又采用了顶部导航，分为4个部分，分别是全部订单、待入住订单、代付款订单和待退房订单。用户可以点击或者滑动来查看对应的订单信息。我的界面主要用来显示用户的个人信息和进行其它操作。主界面如下图4-4所示

图4-4 主界面 图4-5 酒店列表界面

### 酒店列表

酒店列表界面主要有显示搜索到的酒店和对结果进行排序的功能。所以该界面主要有两部分组成，上面是搜索框和下拉选择框，下面是一个显示列表的视图RecyclerView，每当上面的条件改变的时候，就要更新下面的RecyclerView。可以综合排序、距离优先、好评优先、销量优先、价格从高到低和价格从低到高。效果如图4-5所示。

### 酒店详情界面

酒店详情界面主要展示三部分内容，酒店的详细信息、酒店的房型信息列表和可用房间列表。该界面上部分是该酒店的详细介绍，在该界面做了一个滑动特效，就是首先实现酒店图片沉浸到状态栏。然后用户向上滑动改变状态颜色和显示ActionBar和酒店名字。（具体效果参看视频该特效是自定义一个继承ScrollView的布局，监听其中的滑动事件，然后来改变状态栏和ActionBar的状态和颜色。下面的部分是显示该酒店的所有的房间类型，点击该房间类型列表会弹出一个PopWindow显示该房间类型的详细信息。如果用户点击预订按钮就会显示该房间类型可用的房间号。效果如图4-6、4-7所示。

图4-6 酒店详细信息 图4-7 房间详细信息

### 登录注册界面

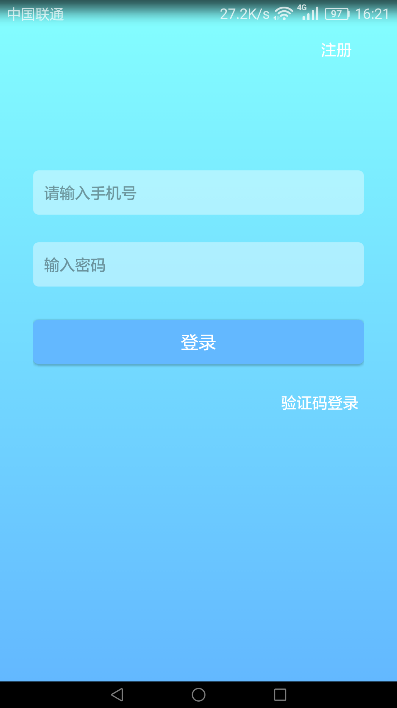
登录和注册是系统的基本功能之一，所以登录注册界面必不可少，登录和注册界面采用统一的风格，一共包括四个界面，手机号密码登录界面、验证码登录界面、验证手机号和填写用户名密码。其中是两个Activity，分别是LoginActivity和RegisterActivity，然后每个Activity包括两个Fragment，登录界面分别是手机号密码登录界面和验证码登录界面，可以在其中一个界面切换到另一种登录方式，页面布局大致一样。注册界面首先是验证手机号发送验证码，点击下一步进入到填写用户名和密码界面。用户填写完成后就可以进行注册了。效果如下图4-8所示。

图4-8 登录注册效果图

### 人证核验界面

人证核验界面也是系统重要的界面之一，该界面分为三个部分，选择身份证界面、拍照界面和人脸拍摄界面。选择身份证界面分两个部分，身份证正面和反面，用户点击可以进入到拍照界面，然后把拍照片回显到当前界面，点击下一步进入到人脸识别界面，在此界面实时显示人脸进行对比，成功的话就显示下一步按钮。

# 系统实现

## 登录注册模块

用户登录和注册是一个系统必不可少的一部分，用户只有进行登录之后才可以下单预订酒店。在本系统中分为三个部分，登录部分、注册部分和人证核验部分。

### 注册模块

用户需要登录之后才能进行酒店预订，没有账号的用户可以进行注册。进入注册界面首先要填写手机号，然后利用正则表达式来校验输入的手机号是否正确，如果不正确就提示用户输入手机号错误。只有输入正确。可以点击下一步，进入到输入用户名和密码界面，在此界面用户需要输入用户名和密码，并且密码有复杂度的要求，输入完成之后会校验两次输入密码是否一致，是否符合密码复杂度。如果都符合的话就调用P层方法把手机号、用户名、密码发送给服务器。服务器会返回一个Result对象，判断其中的success属性是否为true，如果为true就说明注册成功，跳转到下个界面。如果为false，就提示用户message属性中的错误信息。

### 登录模块

用户填写手机号和密码，首先判断手机号和密码是否符合要求，符合要求的话就调用P层的登录方法去请求服务器。如果登录成功服务器就会返回一个token，如果错误提示用户错误信息。为了方便以后登录，需要在APP端保存token信息，保存token最后的地方就是SharePreference中了。首先创建一个仅能自己APP读取的SharePreference，然后在里面保存key为token，value为具体的token字符。这样就可以在以后的请求中从SharePreference中读取token，并在请求中加入token。以后检验身份下单等操作就可以使用该token进行操作，不仅避免了频繁传输用户名密码带来的安全问题，而且减少了频繁查询数据库造成的压力。但是有一个问题就是，这样需要每次请求都要手动添加token，这样不仅麻烦，而且代码冗余比较多。为了解决这个问题，可以实现一个全局的HTTP网络请求拦截器，可以在改拦截器里面实现读取SharePreference中的token信息，然后判断SharePreference中的token是否为空，如果不为空就把token添加到request的header中。这样就可以实现每次请求都带有token。如果token信息过期或者丢失，APP就会跳转到登录界面，提示用户重新登录。

### 人证核验

人证核验是一个非常重要的阶段，在本系统中人证核验分为两步，第一步就是拍摄身份证照片信息，第二步就是对比身份证的信息和人脸信息是否一致。进入认证核验模块，首先是拍摄身份证，用户点击加号可以进入自定义相机界面，需要传入是拍摄的是正面还是反面，然后在ShootIdCardActivity界面调用相机拍摄照片，然后把该照片保存到本地，把路径传回到上个界面，然后在regIDFragment界面回显照片，如果回显的是身份证正面，就对该照片进行OCR识别，调用的是百度OCR识别技术。使用的是离线SDK，首先创建一个IDCardParams对象，把请求的图片路径和其它参数设置好，然后调用OCR.getInstance().recognizeIDCard方法进行识别，该方法有两个回调方法，一个是成功的，一个是错误的方法。在成功的方法里会有IDCardResult对象，该对象是身份证信息，包括姓名、身份证号、出生日期、家庭住址等信息。然后保存该对象，接着拍摄身份证反面，保存照片。如果信息识别成功，就可以点击下一步按钮，进入到人脸识别界面regFaceFragment。在该界面需要首先对身份证上的人脸进行识别，然后调用相机对人脸进行实时对比。该人脸识别技术主要使用虹软(ArcSoft)公司的SDK。首先就要引入该SDK对应的jar包，然后创建人脸识别引擎AFR\_FSDKEngine，接着初始化引擎，这样一个人脸识别引擎就创建成功了。下面就可以开始人脸识别了，首先要对身份证照片中的人脸进行识别，识别成功会生成一个存有人脸信息的AFR\_FSDKFace对象。接着调用系统前摄像头拍摄人脸，在第界面中使用SurfaceView控件显示相机预览画面，然后实现SurfaceHolderde接口，在该onPreviewFrame回调方法中可以拿视频流，在该方法中对人脸信息进行识别，识别成功之后也会返回一个AFR\_FSDKFace对象，最后调用对比方法进行对比，如果对比不成功就提示用户对比失败，可以重新认证。如果对比成功就立即停止相机预览，并显示下一步按钮。当用户点击下一步按钮的话，就掉用P层的方法把身份证正反面照片、姓名、身份证号和人脸信息发送到服务器，服务器保存成功之后返回客户端成功信息，人证核验完成，跳转到主界面。

## 酒店模块

酒店模块是本系统中最主要的模块，很多业务都在该模块。该模块又分为五个部分。分别是酒店搜索、酒店列表和排序、酒店详细信息、酒店房型列表和可用房间列表。

### 酒店搜素

酒店搜索部分又是酒店模块的主要部分，关系用户是否能够找到满意的酒店，是否能找到合适的酒店。所以这一部分非常复杂，当初在设计的时候也耗费了很长时间。由于传输数据多而复杂，所以我们制定了两个实体类，SearchHotel和Hotel，其中SearchHotel包含了九个属性，如下所示：

private String destination; // 目的地

private Coordinate coordinate; //当前位置信息

private String startTime; // 入住时间

private String endTime; // 截止时间

private String key; // 酒店名或地址关键字

private String sort; // 排序方式

private String sortKey; // 查询方式

private Integer currentPage; // 当前页

private T data; // 查询结果

Hotel里面是酒店的一些信息，在此就不再列举。进入到首页，会调用高德的定位服务来获取当前的位置信息，然后把市、县和街道信息显示在目的地的EditTextView中，把经纬度信息存到Coordinate对象中。然后用户可以编辑目的地信息来修改目的地。下面是入住时间，默认时间为今天入住明天离店。用户点击会弹出一个PopWindow，里面包含一个日历选择控件，该控件是在github上面网友分享的一个控件com.henry.calendarview.DayPickerView，通过修改实现了需要的功能。当用户选择好入住日期的话，就会显示确定按钮，用户点击确定按钮会给选择的日期回显到界面，并关闭PopWindow。下面就是一个关键字的EditTextView，用户可以输入需要搜索的关键字。当用户点击查找酒店按钮，就会给这些条件封装到SearchHotel中。其它没有的条件系统会给出默认值。封装以后把该SearchHotel对象传到下个界面。

### 酒店列表

用户点击搜索酒店就会跳到酒店列表界面，并传递了一个SearchHotel对象，然后通过调用P层方法去请求服务器，把SearchHotel传服务器，服务器会返回带有结果的SearchHotel对象，然后取出其中的List<Hotel>，创建一个Adapter用来解析List<Hotel>，然后把Adapter设置到RecyclerView中，就会显示酒店列表到界面。如果其中酒店列表为空，就在下面显示暂时没有符合要求的酒店。

列表页顶部搜索框使用谷歌原生的SearchView控件，设置监听器，然后当用户点击的时候，重新设置SearchHotel条件，然后重新请求服务器获取新的数据，更新Adapter和RecyclerView。

排序条件使用的是下拉菜单，一共三个下拉菜单，首先分别给这三个下拉菜单设置点击监听事件，当用户点击不同的菜单的时候，触发不同的点击事件，然后更改SearchHotel条件重新请求服务器获取新的数据，更新Adapter和RecyclerView。下面为其中一个下拉菜单监听的代码：

zhpxView.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> parent, View view, int position, long id) {

zhpxAdapter.setCheckItem(position);//获取点击的是那一项

mDropDownMenu.setTabText(position == 0 ? headers[0] : zhpx[position]);//改变下拉菜单的选中项

mDropDownMenu.closeMenu();//关闭下拉菜单

key.setSortKey(szhpx[position]);//重新设置关键字

key.setSort("DESC");

mPresenter.getHotelData(key);//调用P层方法请求服务器

}

});

### 查看酒店详细信息

用户点击酒店列表的任意一个酒店就可以进入到酒店详细界面，首先请求服务器，把该酒店的HotelId发送的服务器查询酒店的详细信息。然后进入到酒店详情界面，在酒店详情界面使用了自定义View来实现一个滑动特效，首先创建TranslucentScrollView类继承ScrollView，然后监听其中的滑动事件，当用户滑动的时候改变ActionBar的透明度，这样就实现了，进去的时候是一个全屏界面，然后向上滑动的时候就会改变ActionBar的透明度，并且显示酒店的名称。实现特效以后就是显示酒店的详细信息了，只用把服务器传过来的数据进行显示就可以。滑动前和滑动后的对比如下图5-1和5-2所示。

图5-1 滑动前效果图 图5-2 滑动后效果图

### 获取酒店房型列表

用户进入详细界面之后，就调用P层方法请求服务器。然后查询该酒店的所有的房型信息，然后服务器返回是个List<RoomType>，里面是房型列表，把该List放进TypeListAdapter里面，然后填填充到RecyclerView中，这样就可以显示房型列表了。

当用户点击房型列表的时候，出现一个弹出框PopWindow来显示房型的详细信息。首先让PopWindow显示出来，然后去服务获取数据，最后把房型列表的详细信息显示到PopWindow中。

## 订单模块

订单也是一个非常重要的模块，在该模块主要分为三部分，就是预定酒店，查看订单和修改订单状态也就取消订单和退房离店等操作。主要流程如下图5-3所示。



图5-3 订单模块流程图

### 在线预订

用户查找到指定酒店之后，选择对应的房型信息，点击预订按钮，就会弹出一个PopWindow，然后向服务器发送请求获取该房间类型的可用房间号，已经预定或者正在入住的不会返回。该界面是使用RecyclerView的网格模式，就是每行能显示多个Item，效果如下图所示。如果用户需要预定房间就可是点击对应的房间号，然后向服务器发送请求，参数包括入住时间，预订酒店ID、预订房间类型和预订的房间号。服务器会判断该用户是否登录或者是否通过人证核验，只有通过人证核验的才能进行预订。通过认证就后会跳转到确认订单界面，让用户确认订单信息是否有误，如果无误就点击确认订单，服务器会保存订单，订单状态为未付款，然后进入付款界面，用户选择支付方式进行支付。由于支付接口的限制，本系统只修改订单状态，不做支付功能。

图2-2 选择房间界面 图2-3 确认订单界面

### 查询订单

用户点击订单，就会进入到订单界面，在订单界面是使用一个顶部导航，用户可以查看全部订单、待入住、待付款和待退房订单。这四部分是使用了四个Fragment，然后放进一个ViewPage里面，然后监听ViewPage其中的滑动事件，当用户滑动的时候就会切换Fragment来显示不同的界面。每个界面的操作会不一样，待付款的界面可以取消订单，点击取消订单按钮就请求服务器，服务器就会修改订单状态为已取消。在待退房界面用户可以申请退房离店，然后服务器会给订单状态修改为退房待确认，并且提醒保洁人员进行确认房间无误并且打扫，确认无误之后，就退房成功，订单完成。

## E:\QQDownloads\1308341987\FileRecv\MobileFile\Screenshot_2018-04-25-23-39-01.pngE:\QQDownloads\1308341987\FileRecv\MobileFile\Screenshot_2018-04-25-23-38-45.png 用户信息模块

当用户点击我的界面就会进入到用户信息模块，然后请求服务器去获取用户的个人信息，如果用户没有登录就跳转到登录界面。登录之后就会返回用户的个人信息。其中包括用户名、头像、手机号和是否进行了人证核验，如果没有可以在当前界面进入到人证核验界

# 系统测试

## 测试目的

系统测试是在系统编写工作完成之后的一项不可忽视的工作。系统测试主要是利用人工和自动化软件来运行已完成的程序代码，并将运行后的结果与预期的结果进行对比，查看是否存在差异和错误。系统测试的目的简单来说就是尽可能多的发现系统的错误，然后对错误进行修正，修正完成后再次进行验证，如此反复，对系统的每一个功能模块都要涉及到。

通过对系统的测试，验证系统是否已满足所有的需求，并检查所有业务流程是否可以满足用户需要。同时，在系统测试中，还需对系统的可用性和稳定性进行测试，确保系统在生产环境中也可正常的运行。

## 测试方法

针对系统功能的测试，主要使用黑盒测试法。在进行黑盒测试时，不需要了解系统内部的具体工作流程，而是站在用户的立场上，通过编写好的测试用例，测试系统的每个功能模块是否按照用户的需求进行工作。黑盒测试也具有很多优点：首先，对于一些包含较大代码的系统来说，利用黑盒测试方法进行测试工作时的效率相对白盒测试要高，因为测试人员不需要了解很多复杂的东西，与程序员之间可以是相互独立的，不需要产生关联，任何可以操作的人员都可以进行黑盒测试；其次，站在用户的角度进行测试操作，在这个过程很容易被测试人员所理解，因为系统的功能是否正确实现，测试人员可以很简单地看出来，一旦出现错误，也很容易就被发现，所以也有助于解决问题。

## 功能测试

功能测试是对系统的功能模块进行验证，测试的重点包括两个方面：一是测试系统的功能模块是否符合用户需求，是否可以解决用户面临的问题；二是功能实现的流程和方法是否正确，是否存在潜在错误。本文重点探讨黑盒测试的运用，给出具体的测试用例。

### 用户登录测试

测试项目：用户登录

测试目的：检查用户是否成功登录，正确返回token，错误提示错误信息

测试预置条件：已经注册过的用户信息

执行步骤：

（1）打开登录界面，输入用户名和密码；

（2）点击登录按钮。

预期结果：用户名密码正确返回token，并保存到SharePreference中，然后跳转到主界面。

用户名密码错误并返回错误信息，不进行跳转。

实际结果：与预期结果一致。

### 用户注册测试

测试项目：用户注册

测试目的：查看用户是否可以成功注册，注册失败返回错误信息。

测试预置条件：注册需要填写的信息

执行步骤：

（1）打开注册界面，输入手机号；

（2）点击下一步按钮，输入用户名和密码。

（3）点击注册按钮。

预期结果：注册成功返回token，并保存到SharePreference中，然后跳转到主界面。

注册失败返回错误信息，不进行跳转。

实际结果：与预期结果一致。

### 人证核验测试

测试项目：人证核验

测试目的：查看用户是否可以进行人证核验

测试预置条件：拍摄身份证和人脸

执行步骤：

（1）打开人证核验界面，拍摄身份证正反面；

（2）点击下一步按钮，对比人脸信息；

（3）对比成功上传信息。

预期结果：上传成功，提示核验成功，然后跳转到主界面。

上传失败返回错误信息，不进行跳转。

实际结果：与预期结果一致。

### 搜索酒店

测试项目：搜索酒店

测试目的：查看用户是否可以进行酒店搜索

测试预置条件：输入搜索条件

执行步骤：

（1）进入到主界面，输入要搜索的条件；

（2）点击搜索按钮，跳转到酒店列表界面。

预期结果：搜索到结果，显示酒店列表。

未搜索到酒店，显示暂无符合要求的酒店。

出现错误提示错误信息。

实际结果：与预期结果一致。

### 预订酒店

测试项目：预订酒店

测试目的：查看用户是否可以进行酒店预订

测试预置条件：选择预订的酒店和房间号

执行步骤：

（1）进入到预订界面，选择要预订的房间号。

预期结果：预订成功，进入到确认订单界面。

用户未登录，进入到登录界面。

出现错误提示错误信息。

实际结果：与预期结果一致。

# 总结与展望

## 总结

在这个移动互联的时代，人们急切需求一款可以足不出户、随时随地通过自己的手机来浏览并预订快捷酒店的住宿信息管理系统。本系统基于共享经济和人工智能，利用共享经济的优势，让用户可以快速，方便入住酒店。然后结合人工智能的人脸识别算法，验证人脸和身份证信息来保证安全，可靠。最终实现方便用户的人证核验的共享酒店住宿信息管理系统。

在对系统进行设计前，我查阅了相关资料，阅读了大量相关文献，并对同类系统进行了学习与研究。开发利用Android Studio为工具，用Java语言设计和实现主要运用C/S模式，具有友好的界面，便于用户的操作和管理。系统采用移动端流行的MVP模式，使用MVPArms框架，快速搭建一个Dagger2+Rxjava+Retrofit的Android应用。

基于人证核验的共享酒店用户端可以满足用户预订酒店的需求，该APP通过对市场的几家APP进行研究和对用户体验进行调研，设计出了简洁实用的APP，通过对需求进行分析，把该APP一共分为四大模块：登录注册、酒店模块、订单模块和用户信息模块。首先对所有模块的业务进行总体设计，确定所有的业务需求，然后进行界面设计，紧接着对代码进行编写和业务实现，最后进行测试。测试结果完全符合预期结果和业务需求。用户可以通过该APP进行登录和注册，人证核验，通过指定条件进行搜索酒店和排序，查看酒店的详细信息和预订房间，查看订单信息，取消订单和申请离店退房等操作。用户完全可以在手机上进行预订、入住和退房的全部流程。对于用户非常方便快捷，并且才用人脸信息和身份证进行对比，保证了安全性。

通过对本系统的开发和实现，学习到了很多很多，本小组一共有三个人，从业务需求分析，到指定详细的业务逻辑，再到APP和服务器的接口和传输数据的确定。然后编写代码，最后到系统实现，通过测试。在这个过程中收获了很多，首先在软件开发上，了解了软件的开发流程，学会了分析实际业务和软件开发之间的联系，怎么实现具体的业务逻辑，学会了和小组中其它成员进行沟通和交流，共同解决遇到的问题。其次在技术上了解了MVP的设计模式，体会到了其中的精髓。还学会了Android相关的框架Dagger2+Rxjava+Retrofit的使用。除此之外，还使用了虹软的人脸识别API、百度的文字识别API和高德地位API。

总之通过对本系统的设计与实现，学习到了很多书本上面学习不到的东西，为以后找工作打下了良好的基础。通过几个月来和同学和老师的共同努力，最终完成了这次毕业设计。

## 展望

通过对系统的测试和使用，系统满足了用户的基本需求。但由于时间和技术的问题，系统还存在一些不足。首先由于支付接口的限制，改系统未实现支付功能。其次由于时间的限制，该系统未实现评价功能和在地图上显示酒店并且进行导航功能。由于条件的限制，只在华为、小米和VIVO部分机型进行了测试，并未在其它机型进行测试。最后就是某些功能在用户体验方面不是很好。

通过以上的不足，如果时间和条件允许的情况下，将会在以后的开发中解决这些问题。并且逐步优化代码，提升系统的稳定性和流畅性；优化业务逻辑，减少一些不必要的操作；优化界面，让用户不仅在业务上有很好的体验还能喜欢上APP的界面；优化细节，让用户获得更好的使用体验。最终为用户打造一个简洁、美观、实用的APP。

# 参考文献

[1]李刚.疯狂Android讲义[M].北京:电子工业出版社,2015.6.

[2]满江月.生物特征识别技术的实现原理与前景分析[J].中国安防,2014(11):72-75.

[3]刘坤.基于人脸识别的身份认证系统的设计与开发[D].河北大学,2017.

[4]刘曦芝.基于 Android平台人脸识别算法移植的研究[D]. 武汉理工大学,2015.

[5]明日科技.Java从入门到精通[M].北京:清华大学出版社.2016

[6]明日科技.Java Web从入门到精通[M].北京：清华大学出版社.2016

[7]欧阳燊. Android Studio开发实战:从零基础到App上线[M].北京：清华大学出版社.2017

[8] 孙卫琴. Tomcat与Java Web开发技术详解[M].第2版.北京:电子工业出版社,2010

[9] 温谦. HTML+CSS网页设计与布局从入门到精通[M].北京:人民邮电出版社,2011

[10] 李兴华.名师讲坛—Java Web开发实战经典基础篇[M].北京:清华大学出版社,2010

[11] 张建兵,蔡长安.基于JAVA的B2C网上购物系统的设计与实现[J].重庆工商大学学报,2009,06:563-566

[12] 赵洋,张丽,王恩东,张素宁.基于Structs+Spring+Hibernate的J2EE的架构研究[J].现代电子技术,2009,2(289):107-110

[13] 贺松平.基于MVC模式的B/S架构的研究及应用[D].武汉:华中科技大学,2009

[14] 毕建信.基于MVC设计模式的Web应用研究与实现[D].武汉:武汉理工大学,2009

[15] 孙志权.东辛农场网上办公系统的设计与实现[D].上海:华东师范大学,2009

[16] 杨振娟.高职院校物资管理系统的设计与实现[D].河北:河北科技大学,2013

[17] 凌宇翔.华荣宾馆物资管理系统的设计与实现[D].成都:电子科技大学,2014

[18] Lee H L. A multi-echelon inventory model for repairable items with emergency lateral transshipments [J]. Management Science, 2004, 20(6): 1302-1316

[19] Axsater S. Modelling emergency lateral transshipments in inventory systems[J]. Management Science, 2003, 15(2): 1369-1338

[20] Zoya Ali. Designing Object Oriented Software Applications within the Context of Software

Frameworks[D]. Ohio State University,2011

# 致谢