**用户APP**

# 总体设计

## 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。本系统本着简介大方，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，订单界面。其中主界面采用底部导航，分为三个模块，分别是首页订单和我的。对应界面如下图1-1所示：然后就是酒店列表界面，酒店列表界面，主要给用户展示搜索到的酒店列表，然后用户可以在此界面进行搜索和排序，界面如图1-2所示

图1-1 主界面 图1-2 酒店列表界面

登录注册界面

实名认证界面

## 业务设计

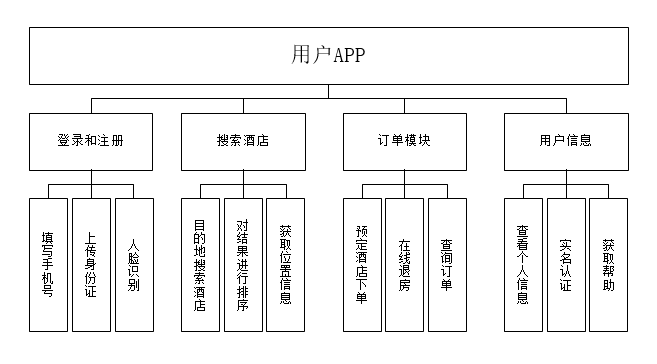
通过对业务需求的详细分析，对系统的业务进行分析，主要分登录注册模块，这一模块主要进行用户信息的身份验证，包括实名认证和人脸识别，其次就是酒店搜索模块，这一模块用户可以通过目的地，关键字，酒店名称进行搜索，然后可以用户可以对搜索结构进行排序。接着就是订单模块，用户选择酒店房间进行下单，然后查看订单信息等。最后就是用户信息模块，在此模块用户可以查看自己的信息，获取帮助等。没有进行实名认证的用户可以在此进行实名认证。进行退出登录等操作。模块分析图如下图1-3所示

图1-3 系统业务模块图

# 系统实现

## 登录注册

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。用户填写用户名和密码。如果登录成功服务器就会返回一个token，APP对此token保存到本地，以后检验身份下单等操作就可以使用该token进行操作，避免了频繁传输用户名密码带来的安全问题，减少了频繁查询数据库造成的压力。对于token的操作，为了减少每次请求都要都要手动带上token，所以APP实现了一个全局的网络请求拦截器，如果返回的信息带有token，就把该token保存到SharePreference。然后每次请求的时候，都判断SharePreference中的token是否为空，如果不为空就在该请求加上token。

clientBuilder.addInterceptor(new Interceptor() {

@Override

public Response intercept(Chain chain) throws IOException {

Request request = chain.request();

SharedPreferences sharedPreferences = context.getSharedPreferences("config",Context.MODE\_PRIVATE);

Response response = chain.proceed(request);

if (response.header("Authorization")!=null){

SharedPreferences.Editor editor = context.getSharedPreferences("config",Context.MODE\_PRIVATE).edit();

editor.putString("token",response.header("Authorization"));

editor.commit();

}

return response;

}

});

clientBuilder.addInterceptor(new Interceptor() {

@Override

public Response intercept(Chain chain) throws IOException {

Request request = chain.request();

SharedPreferences sharedPreferences = context.getSharedPreferences("config",Context.MODE\_PRIVATE);

String token = sharedPreferences.getString("token", "");

if (!token.isEmpty()){

Request authorization = request.newBuilder().addHeader("Authorization", token).build();

return chain.proceed(authorization);

}

return chain.proceed(request);

}

});

## 用户注册

用户首先要填写手机号，判断手机号是否正确和是否注册过，然后给改手机号发送验证码（该功能暂未实现），如果用户输入验证码正确就可以进行下一步，填写用户名和密码。在密码界面，要验证密码的复杂性和两次输入密码一直，如果都通过就提交用户信息到服务器。注册成功就返回token，相当于用户直接登录，注册失败提示对应的信息。注册成功之后就可以进行实名认证，首先用户拍摄身份证正面照片和反面照片，拍摄之后，对身份证正面照片进行ORC识别，返回用户身份证信息，验证身份证信息的完整性，成功的话进行下一步，人脸识别，打开用户手机前摄像头，获取上一步的身份证的人脸信息，然后获取摄像头中的人脸信息，对比两个信息的完整性，如果正确，就给人脸信息身份证信息上传到服务器，完成实名认证。

## 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，会弹出该房型对应的可用房间号，用户选择房间号如下图2-2所示，就进入订单界面如下图2-4所示（如果没有登录，是进不去的会被拦截到登录界面）让用户确认订单信息，如果信息无误的话，就可以点击确认订单，后台就会生成对应的订单信息。

图2-2 选择房间界面 图2-3 确认订单界面

## 查询订单

用户点击订单会向服务器发送消息，查询改用户的信息，最终显示到界面。如下图2-4所示：

