# 项目功能简述

功能一、登录功能

功能二、通讯录管理的功能（增删查改）

功能三、附加功能：实现事务提醒功能

# 应用功能详解和分析

\*\*\*本项目要将用户交互与业务处理分开，所有业务处理以Service实现

首先需要掌握并运用service相关知识

过程中需要解决的问题和解决方法步骤：

**Activity如何与Service之间进行通信？**

通过Binder对象（以login为例子）

Service:

1. 新建一个Activity，再新建一个相应的Service
2. 在Service的onbund里面返回Binder对象→自定义一个方法loginServiceBinder
3. loginServiceBinder继承自Binder，再在该方法创建getService方法来获取实例

Activity:

1. 在oncreate里发起对Service的绑定
2. 建立一个ServiceConnection，绑定成功后新建一个对象从Binder对象中得到Service
3. 在监听事件里调用Service里的方法
4. 如果要调用service的方法直接向activity的界面初始化值，需要new一个新的线程来判断服务连接是否成功，然后在子线程中做耗时操作

\*\*\*实现对数据库的增删查改需要掌握和运用：SQLite和SQL语句的相关运用

首先新建一个类OderDBHelper用于建数据库和建表

1. **如何打开或者创建数据库？**
   * 1. 在Android 中使用SQLiteDatabase的静态方法openOrCreateDatabase(String  path,SQLiteDatabae.CursorFactory  factory)打开或者创建一个数据库
     2. 参数1 数据库创建的路径
     3. 参数2 一般设置为null
2. **如何创建一张表?** 
   * 1. 编写创建表的SQL语句
     2. 调用SQLiteDatabase的execSQL()方法来执行SQL语句

**再分别在相应的Service中书写相关的数据操作方法（详细步骤见相应的功能↓）**

##### 登陆功能

\*\*\* 布局的实现需要掌握和运用：运用**Layout布局**相关基础知识

功能详述：

一次登录后保存用户名和密码以便下次自动登录，自动登录功能只能维持十天，十天后应重新要求用户输入用户名和密码登录

过程中需要解决的问题和解决方法步骤：

1. **如何保存用户名和密码？（并且只保存十天）**

具体参考SharedPreferences的使用：

* 1. 得到SharedPreferences对象
  2. 调用SharedPreferences对象的edit()方法来获取一个SharedPreferences.Editor对象
  3. 用putString()方法向SharedPreferences.Editor对象中添加数据
  4. 调用commit方法将添加的数据提交
  5. 写两个获取时间的方法，获取上次登陆和这次登陆的时间，并通过判断时间差是否大于10天来选择是否显示用户名和密码。

1. **如果使登陆界面全屏达到更加美观的效果？**

一开始参考：<http://bbs.csdn.net/topics/391963786?page=1>

发现因为继承了AppCompatActivity状态栏消失了，但一直无法去除标题栏

又参考了：<http://blog.csdn.net/s297165331/article/details/52138317>

才知道AppCompatActivity根本没标题栏，ActionBar才感觉是个标题栏

所以书写直接隐藏ActionBar的代码，得以实现想要的全屏效果

1. **如何写几秒跳转的欢迎界面？**

设置一个定时器TimeTask，在定时器里写一个线程run方法里面实现页面的跳转

1. **如何实现注册功能？**
   1. 在数据库中建立一张user表
   2. 调用service里的add（）方法，将屏幕上获取的用户数据插入数据表user中

##### 通讯录管理

**如何使用ListView？**

* 1. Layout引入ListView
  2. 初始化数据（利用service从数据库中得到）
  3. 初始化界面（获得控件，创建自定义适配器，使用setAdapter()方法）
  4. 编辑自定义适配器MyAdapter中的getCount和getView方法
  5. 编辑MyCallback()方法里的onActionItemClicked()方法【当点击Action菜单时调用】

2.1 添加联系人

功能详述：

可以添加联系人的基本信息，如：姓名、电话、单位、住址等信息；

过程中需要解决的问题和解决方法步骤：

**如何添加屏幕上输入的信息到数据表之中？**

* 1. 获取屏幕上的信息，返回ContactInfo的列表
  2. 调用service里写的add( )方法，将列表中的值插入数据表中（运用了put和insert方法）

2.2 查看信息

功能详述：

查看某个联系人的具体信息；

过程中需要解决的问题和解决方法步骤：

**如何使用ActionMode达到长按列表menu菜单弹出修改和删除选项？**

\*\*在 res/menu 下面新建activity\_main 的布局文件作为初始菜单

\*\*同时在 res/menu 下面 新建一个xml 文件action\_mode\_main，作为 ActionMode 菜单的布局文件

\*\*设置lisrview长按的点击事件，用setOnLongClickListner()方法，在里面new一个MyCallback()方法接口于ActionMode

2.3 删除联系人

功能详述：

根据输入的条件查询联系人；

过程中需要解决的问题和解决方法步骤：

\*\* 通过点击传入的name值，调用delete方法，删除数据库中该联系人的信息

\*\* 刷新屏幕上显示的数据

2.4 修改联系人的信息

功能详述：

修改某个联系人的具体信息；

过程中需要解决的问题和解决方法步骤：

\*\* 通过点击传入的name值，调用查询方法（返回数据列表），来初始化页面的数据

\*\*获取屏幕上更新的数据，返回一个数据列表

\*\*调用updata方法，传入返回的数据列表

\*\*\*

2.5 数据备份与还原（导入导出功能？）

功能详述：

可以从Android系统通信录或特定文件中导入联系人，也可以将应用中的联系人导出到对应文件（导入导出的文件最好是.vcf文件）

过程中需要解决的问题和解决方法步骤：

**如何从系统通讯录中导入联系人？**

\*\* 获取权限【详细如下】

\*\* 通过ContentResolver获取安卓系统联系人，返回联系人列表

\*\* 调用add方法，将返回的列表增加到数据表中，再刷新界面显示的数据

**如何从导出vcf文件到SD卡中？**

\*\* 获取权限

\*\* 导入cardme jar包

\*\* 创建一个vcf文件，利用循环将联系人列表的信息以此转换为Vcard格式的字符串存入vcard

\*\* 添加成功，关闭文件流

**如何导入SD中的vcf文件到通讯录？**

\*\* 获取权限

\*\* 用VcardEngine的parseMultiple( )方法解析vcf文件

\*\* 将解析出的数据存入联系人列表并返回

\*\* 调用add方法，将数据添加到数据表中

\*\* 刷新界面显示的数据

2.6 实现拨打电话和发送短信的功能

过程中需要解决的问题和解决方法步骤：

**Android6.0运行时权限解决方案**

\*\* 新建一个封装类BaseActivity（Android 6.0 权限控制代码封装）

\*\* 在需要获取权限的activity中继承自封装类

\*\* 调用permissionCodeWithPermissoin方法，将需要获取权限才能实行的操作写入hasPerssoin( )内

\*\* 在AndroidMainifest.xml里书写相关权限

**拨打电话**

\*\* 获取权限

\*\* 创建intent，setAction

\*\* 获取点击的号码，setData

**发送短信**

\*\* 获取权限

\*\* 获取点击的号码和输入的文本信息【创建一个可输入的对话框】

\*\* 调用sendTextMessage方法发送短信

##### 附加功能（事务提醒）

功能详述：

提醒的内容至少要包含：

联系人，提醒标题，提醒详情，联系人电话和提醒动作（动作类型至少应有打电话、发短信、无）

在用户确认提醒后，执行相关动作。

过程中需要解决的问题和解决方法步骤：

1. 设置时间，将设置的时间数据传入下一个内容页面
2. 获取页面上输入的数据（其中判断一下联系人名字是否正确）
3. 调用事务提醒方法，传入获取的数据（发送广播）
4. 写一个继承自BroadcastRececiver的接收广播的方法，接收传来的数据，再将数据发送给下一个打开的页面
5. 将传入的数据展示在打开的页面中，再设置打电话和发短信相关事件【详细参考上面】

**如何进行事务提醒？需要用到哪些方法？**

1. 调用AlarmManager方法，获得提醒管理的对象
2. setAction( )方法设置提醒的动作，在里面发送需要传达的数据
3. pendingIntent.getBroadcast( )将设置提醒的动作发送广播，利用Calendear的set方法，设置提醒时间
4. 调用onReceive方法接收发送的广播，再在里面接收接收传来的数据，并创建状态通知栏
5. 设置状态通知栏的意图，并将数据发送到下一个点开的页面
6. 在点开的页面里根据接受的数据，设置需要显示的信息