

Android 移动应用开发期末项目报告

备忘

组号

小组成员：李启明、敖津聪、陈桂燕、陈紫荆

组长 李启明 14331133

成员 敖津聪 15331003

成员 陈桂燕 15331016

成员 陈紫荆 15331048

一、概述

这是一个为生活繁忙都市人而设的 APP，用户可以使用我们的“备忘”去记录生活中的各种重要事情，包括你的未来计划、你的每日纪录、家人朋友的生日等。我们为这个 APP 加入了更为人性化的操作，增加与用户的互动，希望增加用户使用友好度。的这是一个可以让你跟忘记说“不”的一个 APP；这是一个可以让你轻松管理事情的 APP，快来开始使用“备忘”吧！

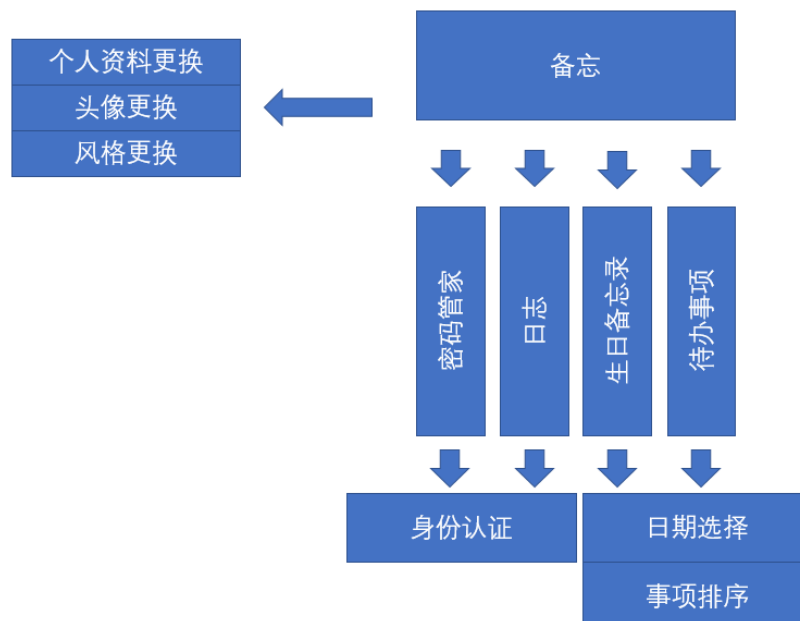
二、背景

生活在忙碌大都市，人们的生活都非常急促，每天都有数不尽的事情要处理。这时候就需要【备忘】App，将这些琐碎信息统统记下，让生活更高效。

三、项目说明

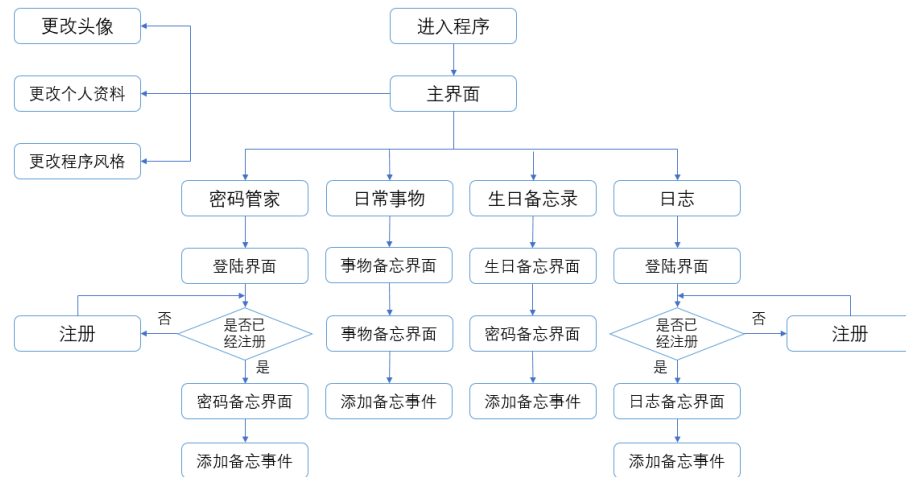
1. 功能模块

- ✓ 个人头像更改功能
- ✓ 密码储存功能
- ✓ 个人日志功能
- ✓ 朋友生日备忘功能
- ✓ 日常事物备忘功能
- ✓ 身份认证功能
- ✓ 事物排序功能（按日期）
- ✓ 日期选择功能



2. 逻辑流程

- ✚ 提供导航栏去更改个人资料及程序风格。
- ✚ 程序主要分成了四个的功能块，每个的功能块均提供不同的事件备忘功能。
- ✚ 提供资料更改功能，包括个人的资料及风格。
- ✚ 程序的其流程图如下：



3. 技术说明

- 数据库：对四个备忘模块进行数据库绑定
 - 由于 APP 定位为随时可用的个人单机应用，所以我们的数据都保存在本地，以使用户随时能查看编辑。
 - 在 SQLite 数据库里创建了四张表分别用来保存四个功能板块里的数据：

```
public class birthDataBase extends SQLiteOpenHelper {
    private static final String DB_NAME = "MyDB.db";
    private static final String TABLE_NAME = "Info"; //生日列表
    private static final String TABLE_NAME_2 = "Info_2"; //密码列表
    private static final String TABLE_NAME_3 = "Info_3"; //待办事项列表
    private static final String TABLE_NAME_4 = "Info_4"; //待办日志列表
}
```

- RecyclerView：用作各备忘事件的显示
- IntentFromGallery：进行本机图片的读取
 - 使用了安卓自带的 intent 来读取本地相册：

```
private void choseHeadImageFromGallery() {
    Intent intentFromGallery = new Intent();
    // 设置文件类型
    intentFromGallery.setType("image/*");
    intentFromGallery.setAction(Intent.ACTION_GET_CONTENT);
    startActivityForResult(intentFromGallery, CODE_GALLERY_REQUEST);
}
```

- 上图函数执行后，会跳到相册中，可选择图像如下图：

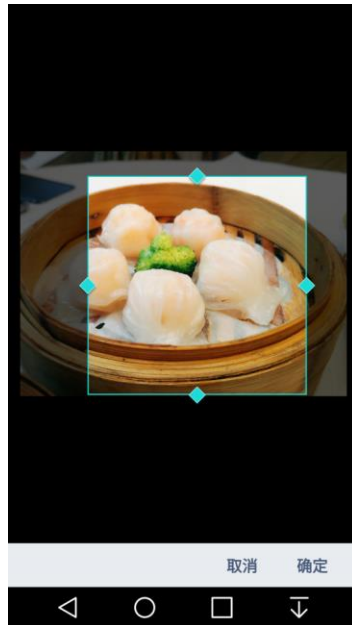


- 回调函数中显示，提取 intent 携带的 data：

```
protected void onActivityResult(int requestCode, int resultCode, Intent intent) {  
    // 用户没有进行有效的设置操作，返回  
    if (resultCode == RESULT_CANCELED) {  
        Toast.makeText(getApplication(), "取消", Toast.LENGTH_LONG).show();  
        return;  
    }  
    switch (requestCode) {  
        case CODE_GALLERY_REQUEST:  
            cropRawPhoto(intent.getData());  
            break;  
    }  
}
```

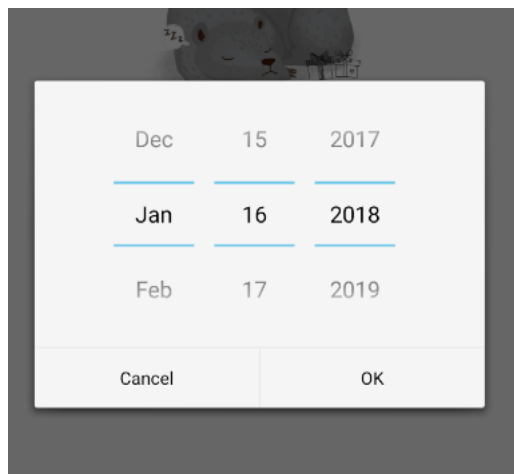
- 裁剪好按确定后，该 intent 裁剪头像、并携带裁剪好的临时图像的 uri 返回到回调函数中：

```
/**  
 * 裁剪原始的图片  
 */  
public void cropRawPhoto(Uri uri) {  
    Intent intent = new Intent("com.android.camera.action.CROP");  
    intent.setDataAndType(uri, "image/*");  
    // 设置裁剪  
    intent.putExtra("crop", "true");  
    // aspectX, aspectY : 宽高比例  
    intent.putExtra("aspectX", 1);  
    intent.putExtra("aspectY", 1);  
    // outputX, outputY : 裁剪图片宽高  
    intent.putExtra("outputX", output_X);  
    intent.putExtra("outputY", output_Y);  
    intent.putExtra("return-data", true);  
    startActivityForResult(intent, CODE_RESULT_REQUEST);  
}  
  
case CODE_RESULT_REQUEST:  
    if (intent != null) {  
        setImageToHeaderView(intent);  
    }  
    break;  
default: break;
```



iv. DatePickerDialog：进行日期的选择

➤ 点击日期，弹出日期选择框



➤ 透过 DatePickerDialog 及 OnDateSetListener 实现：

```
@Override
public void onClick(View v) {
    DatePickerDialog datePickerDialog = new DatePickerDialog(mContext, DatePickerDialog.THEME_HOLO_LIGHT, (view, year, month, dayOfMonth) {
        Year = year;
        Month = month;
        Day = dayOfMonth;

        String y = String.valueOf(year);
        String m = String.valueOf(month);
        String d = String.valueOf(dayOfMonth);
        if (month + 1 < 10) {m = "0" + String.valueOf(month + 1);}
        if (dayOfMonth < 10) {d = "0" + String.valueOf(dayOfMonth);}

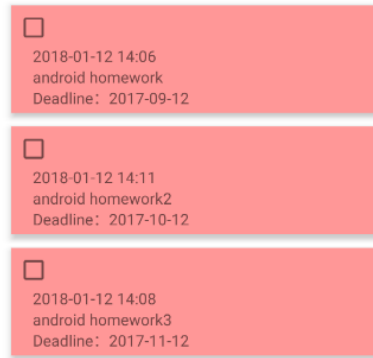
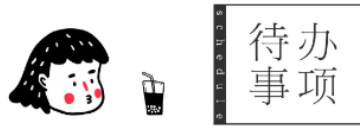
        deadline_Text.setText(y + "-" + m + "-" + d);
    }, Year, Month, Day);
    datePickerDialog.show();
}
```

v. 高级界面：利用了 Android 各种的页面布局技术

vi. Meau：实现导航栏的操作

vii. Item 的排序功能：利用 Collections 对 List 中的项进行排序

➤ 点击图片，对 Item 进行按日期排序：



➤ 利用 Collection 对 List 进行排序：

```
private List<affair_user> sortDeadlineUp(List<affair_user> users) {  
  
    Collections.sort(users, new Comparator<affair_user>() {  
        @Override  
        public int compare(affair_user o1, affair_user o2) {  
            if (o1.getFinish().equals("1") && o2.getFinish().equals("0")) {  
                if (o1.getYear() == o2.getYear() && o1.getMon() == o2.getMon()) {  
                    return o1.getDay() - o2.getDay()+100;  
                }  
                if (o1.getYear() == o2.getYear() && o1.getMon() != o2.getMon()) {  
                    return o1.getMon() - o2.getMon()+100;  
                }  
                if (o1.getYear() != o2.getYear()) {  
                    return o1.getYear() - o2.getYear()+100;  
                }  
            }  
        }  
    })  
}
```

4. 项目说明

✚ 程序主要由两部分组成：①导航栏、②功能模块。导航栏的作用是增加与用户的互动，模块功能主要实现程序的主体功能。

利用 meau 对导航栏进行实现

```
<include
    layout="@layout/app_bar_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />

<android.support.design.widget.NavigationView
    android:id="@+id/nav_view"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_gravity="start"
    android:fitsSystemWindows="true"
    app:menu="@menu/activity_main_drawer" />

<?xml version="1.0" encoding="utf-8" ?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <group android:checkableBehavior="single">
        <item
            android:id="@+id/change_image"
            android:icon="@drawable/ic_account_box_black_24dp"
            android:title="更换头像" />
        <item
            android:id="@+id/change_name"
            android:icon="@drawable/ic_chat_black_24dp"
            android:title="昵称&签名" />
        <item
            android:id="@+id/function"
            android:icon="@drawable/ic_menu_send"
            android:title="用户指南" />
        <item
            android:id="@+id/developer"
            android:icon="@drawable/ic_menu_manage"
            android:title="开发者" />
    </group>
</menu>
```

(然后对于导航栏里不同的点击，进行相应的操作)

主题功能分成四部分进行实现，对应不同的功能模块

- ▼ com.example.yc.finalproject
 - ▶ adapter
 - ▼ affairMemo
 - affair_Additem
 - affair_MainActivity
 - ▼ birthMemo
 - birth_Additem
 - birth_MainActivity
 - ▼ dairyMemo
 - dairy_Additem
 - dairy_content
 - dairy_login_activity
 - dairy_MainActivity
 - ▶ dataBase
 - ▶ main
 - ▶ model
 - ▼ pawMeno
 - psw_Additem
 - psw_login_activity
 - psw_MainActivity
 - ▶ toolbar

(各功能模块根据其使用需要进行不同的操作开发)

四、应用效果



点击程序进入主界面：



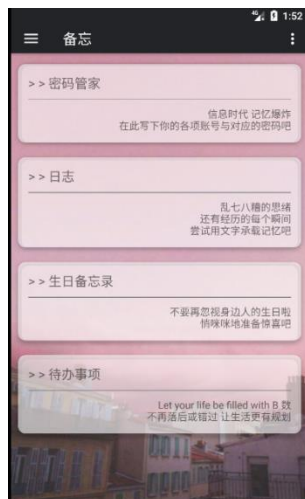
点击左上角的导航栏，弹出导航信息



点击头像，进行头像的更改：



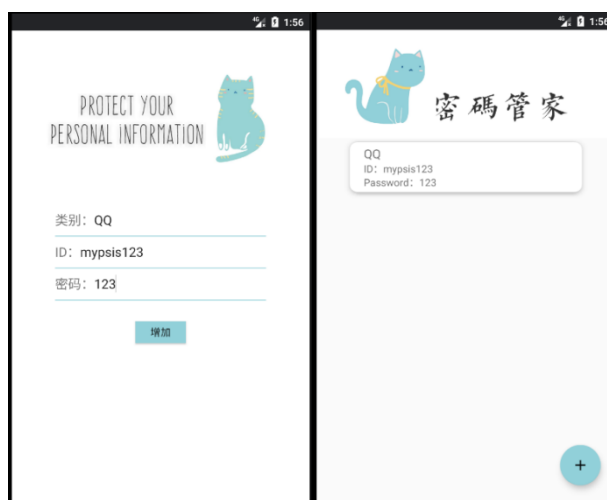
点击初始风格，更换程序的整体风格：



点击密码管家，进入密码备忘模块：



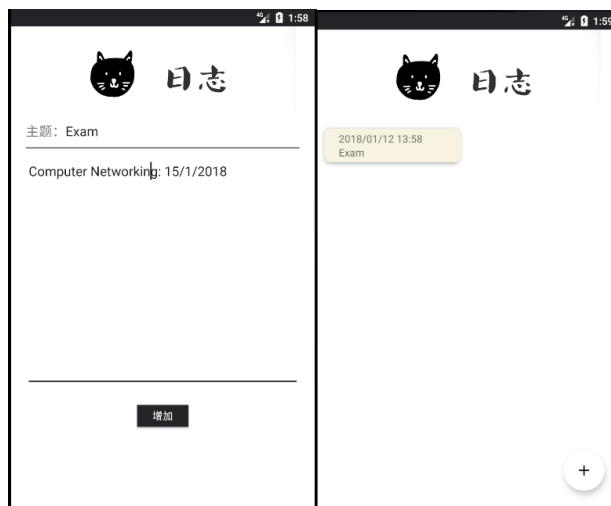
添加密码备忘：



点击日志，进入日志备忘模块：



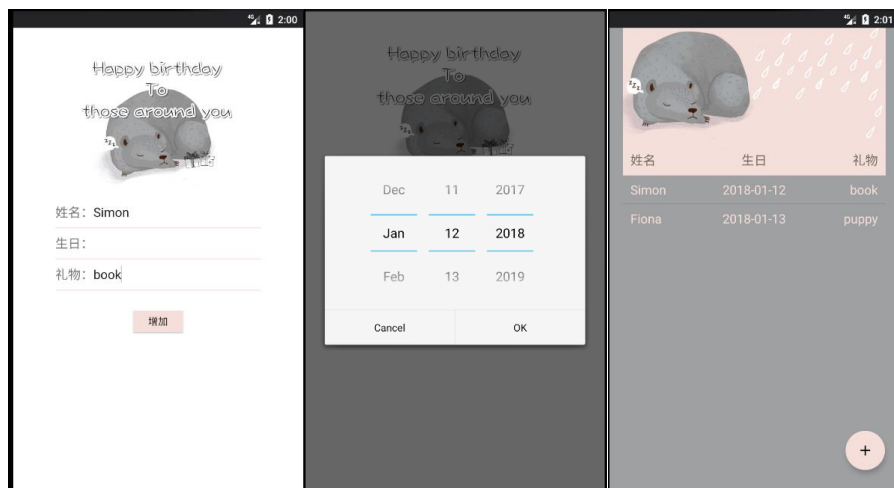
添加日志备忘：



点击生日备忘录，进入生日备忘模块：



添加朋友的生日备忘：



点击图片对生日备忘的选项进行按日期的排序：



点击待办事项，进入事务备忘模块：



添加事务备忘：



点击图片对事务进行按时间排序：



五、开发过程遇到的问题

问题一：程序的选题问题

解决方法：我们发现“备忘”比较有现实的需要，所以我们改变了我们的选题

问题二：实现功能模块比较多，需要编写大量的代码并进行开发

解决方法：透过协调，对程序的各个模块进行分工，平均组员的开发量。

问题三：技术问题

解决方法：透过网上的资料搜索，学习相关的技术并应用在程序里

六、思考及感想

首先我们小组进行了讨论，初步设定好该 app 应该展现什么 UI 界面以及如何实现各个功能，之后进行分工\接合\修改完善。本次实验最初的大部分功能实现都是基于已学的知识，遇到不懂的知识（例如日期的选择及从手机读取本地的图片）便通过互联网进行查询及学习。由于程序实现了四个功能模块，所以代码会比较复杂。而且我们为每一个模块也绑定数据库，方便对每种备忘事件进行独立的处理。从一开始组员分开开发程序，到慢慢的把各部分整合再一起，过程遇到了比较多的问题，不过经过不断的尝试，最后成功把程序开发出来。

七、小组分工

组员	学号	负责内容	贡献百分比（%）
李启明	14331133	各模块日期的选择、Item 的样式优化及排序	25%
敖津聪	15331003	生日备忘录的实现，部分测试	25%
陈桂燕	15331016	UI 界面、数据实现、完成基本的界面跳转逻辑	25%
陈紫荆	15331048	为所有内容增添“编辑/修改”功能；增加“读取本地图片并截取成头像”功能；修复各版本的 bug；	25%