

《移动终端应用开发技术》课程作业

项目名称：Android个性化定制软件My Time Diary

项 目 负 责 张美楠（B16070504）

项 目 组 成 员 张美楠（B16070504）

指 导 教 师 王亚石

指 导 单 位 物联网学院

日 期 2018-11-30

|  |  |  |  |
| --- | --- | --- | --- |
| **评**  **分**  **细**  **则** | **评 分 项 目** | **项目分值** | **项目得分** |
| 软件分析能力 | 10 |  |
| 软件功能 | 20 |  |
| 软件界面 | 20 |  |
| 技术应用能力 | 30 |  |
| 文档规范程度 | 20 |  |
|  | **总评分** |  |
| **评**  **语** |  | | |

**目录**

[《移动终端应用开发技术》课程作业 1](#_Toc499561548)

[1 软件开发可行性报告 5](#_Toc499561549)

[1.1 项目概述 5](#_Toc499561550)

[1.2 技术可行性 5](#_Toc499561551)

[1.3 经济可行性 5](#_Toc499561552)

[1.4 操作可行性 5](#_Toc499561553)

[1.5 法律可行性 5](#_Toc499561554)

[2 软件需求规格说明 6](#_Toc499561555)

[2.1 需求概述 6](#_Toc499561556)

[2.2 系统功能需求 6](#_Toc499561557)

[2.3 非功能性需求 6](#_Toc499561558)

[2.3.1 性能需求 6](#_Toc499561559)

[2.3.2 接口需求 6](#_Toc499561560)

[2.3.3 界面需求 6](#_Toc499561561)

[2.3.4 软硬件环境需求 6](#_Toc499561562)

[3 系统概要设计 7](#_Toc499561563)

[3.1 系统模块设计 7](#_Toc499561564)

[3.2 数据结构设计 7](#_Toc499561565)

[3.3 接口设计 8](#_Toc499561566)

[4 详细设计 1](#_Toc499561567)0

[4.1 模块一 1](#_Toc499561568)0

[4.1.1 局部数据结构 1](#_Toc499561569)0

[4.1.2 处理逻辑 1](#_Toc499561570)0

[4.2 模块二 1](#_Toc499561571)1

[4.3 模块三 1](#_Toc499561572)3

[5 交互界面设计 1](#_Toc499561573)5

[5.1 界面流概述 1](#_Toc499561574)5

[5.2 界面一 1](#_Toc499561575)6

[5.3 界面二 1](#_Toc499561576)6

[5.4 界面三 1](#_Toc499561577)7

[6 系统测试 1](#_Toc499561578)9

[6.1 测试计划 1](#_Toc499561579)9

[6.2 测试用例描述 1](#_Toc499561580)9

[6.2.1 单元测试用例 1](#_Toc499561581)9

[6.2.2 集成测试用例 1](#_Toc499561582)9

[6.3 测试报告 1](#_Toc499561583)9

[7 用户手册 2](#_Toc499561584)0

[7.1 安装配置手册 2](#_Toc499561585)0

[7.2 使用手册 2](#_Toc499561586)2

[7.3 维护手册 2](#_Toc499561587)2

[8 项目总结 2](#_Toc499561588)2

# 软件开发可行性报告

## 项目概述

一款基于匹配开发者自身生活习惯的Android个性化定制软件，将其一天当中可能会经常用到的多个软件需要实现的功能汇总到一个软件中，避免其频繁的需要打开各个软件，主要特点在于适应使用者自身的应用习惯。

## 技术可行性

此软件采用Android studio集成开发工具开发，主要应用Java编程语言编写开发。目前Android studio开发工具已经发展的相对完善，是一个十分适合新手进行app开发的平台，环境提供了：

* 基于Gradle的构建支持；
* Android 专属的重构和快速修复；
* 提示工具以捕获性能、可用性、版本兼容性等问题；
* 支持ProGuard 和应用签名；
* 基于模板的向导来生成常用的 Android 应用设计和组件；
* 功能强大的布局编辑器，可以让你拖拉 UI 控件并进行效果预览。

因此在技术水平上，由于强大的开发环境的支持，以及日益成熟的软件开发过程，使得此项软件开发可以顺利进行。

## 经济可行性

此软件的实现具备经济开发可行性。结合目前市场需求来看，私人功能定制化软件可以满足当前不同种类的人群对于软件产品功能的多样化需求。此项软件开发亦不会花费开发者高额的开发成本，所以其具备开发的可行性。

## 操作可行性

此款Android私人定制化软件，具有操作可行性，是一款可以在Android手机上运行的软件，对于可以操作安卓手机的人群来说是十分容易的。

## 法律可行性

此款软件无任何逾越法律法规的内容，具备法律可行性。

# 软件需求规格说明

## 需求概述

此款软件实现的用户需求为：为了使用者的方便，将其一天中经常需要浏览的软件功能结合到同一软件中，避免用户对各种软件之间频繁的切换操作。

## 系统功能需求

系统需要实现的功能需求包括，首先进入软件的欢迎界面，即展示给用户良好的视觉体验，其次实现用户的用户名注册以及登录功能，进入软件的主页界面，其中展示微信精选以及常用快递查询功能，两个功能均需要请求聚合官网的API接口，其中需要说明的一点是，微信精选的API对于个人申请者可以免费使用，但是快递查询功能，本人目前申请的账号目前只能免费查询100次，当超过100次之后需要付费。

## 非功能性需求

体现出良好的软件展示界面，以及对于用户来说便利的使用感。

### 性能需求

此款软件的体积较小，性能一般的智能手机即可安装使用。

### 接口需求

此款软件中主要的内容包括第三方网站的数据的接入，所以需要到相应的网站上申请API KEY。

### 界面需求

界面主要包括用户注册以及登录的界面，以及展示给用户的主页，其中设计成选项卡的界面，便于用户对各种功能进行选择。

### 软硬件环境需求

此款软件的开发需要配备一台可以正常运行Android studio的计算机。

# 系统概要设计

## 系统模块设计

3.1.1 引导模块

实现app的引导功能，主要用于实现界面的美观性，以及用户对于此款app的功能的大致了解，此处主要采用了view pager技术实现轮播图的效果，具体的实现方法是，采用了3张本人喜欢的图片来实现引导功能；首页跳转逻辑：前两张图片默认在屏幕的右上角显示“跳过”功能，点击后可以直接跳转到主页；最后一张引导图片需要将“跳过”隐藏，显示“进入主页”的按钮，点击后可以跳转到主页。

3.1.2 注册以及登录模块

此模块主要实现了用户本地注册以及登录的功能，对于后续开发者的一些需要用到数据库的，涉及到个人用户信息的内容进行适配。

3.1.3 主页模块

主页主要展示各个开发者结合自身实际，添加进去的一些功能集合，展现出由多个功能集合汇总在一起的界面。

3.1.4 文章详情模块

此处主要展示通过第三方API接入的微信文章数据，使得用户通过此界面获得每日最新给新闻资讯。

3.1.5 拓展模块

根据使用者的使用习惯，此app的优点之一在于，各个模块之间的独立性较强，便于用户随时添加或删除修改功能，此软件中除了文章详情界面，还引入了常用快递查询功能。

## 数据结构设计

项目架构部署如下图3.2所示：

整个项目包含的各类操作都放进相应的包下：界面activity都放进ui包内，所有工具类都放进utils包中，所有的fragment都放进fragment包下，有关数据库的内容分别放进sqlite以及bean包下，有关适配器的内容都放进adapter包下，以及一些实体entity等。

1）各个封装起来的包package用于存放项目中需要用到的各种包，例如：定义application实现初始统一的初始化，ui中存放各个需要真实展示出来的界面，utils中存放各个同一的方法，实现功能的封装，避免代码的重复，便于实际开发。

2）工具类的封装

3）Log的封装：（实现工具类）主要用于在控制台中打印一些我们运行中产

生一些信息，实际中封为5个等级，DEBUG,INFO,WARN,ERROR,FATAL,实际中这里主要使用i和e的信息；包utils中定义Java类，设置一个true/false的打印开关，再将繁琐的打印日志的语句Log.i（“TAG”,”text”）;简化为L.i(“test”);从而实现简单的工具打印功能的封装。

4）Shared Preferences的封装：Shared Preferences是一种轻型的数据存储方式，他的本质是基于XML文件存储key-value键值对数据，通常用来存储一些简单的配置信息，此处开发中主要用来实现本地用户的登录和注册功能。

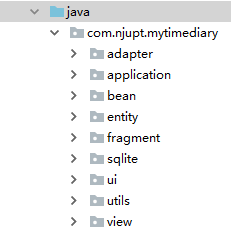
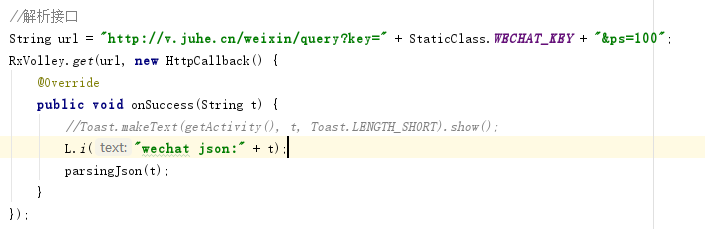


图3.2 My Time Diary架构

## 接口设计

首先进入想要接入网址数据的网站上申请开发者的API KEY，然后根据网址上的开发实例，解析json实例，拼接接口的调用方法，加入到自己的开发程序中；



# 详细设计

## 模块一---闪屏页跳转逻辑开发

### 局部数据结构

struct Splash Activity {

Text View tv\_splash;

Handler handler; }

Operation：

void on Create（）；

void init View（）；

boolean isFirst（）；

void on Back Pressed（）；

### 处理逻辑

定义一个Boolean值 is First，首先默认其为true，代表其是第一次运行，当加载软件之后，首先默认其是第一次运行，则展现引导页，同时返回一个false给变量 is First ，代表其已经经历过了第一次运行，则直接进入登陆界面；拓展功能：重写 on Back Pressed() 方法，其中不执行super语句，即可实现，禁止引导页返回的功能。

如下图4.1所示，为软件My Time Diary的初始界面的闪屏页的跳转逻辑：

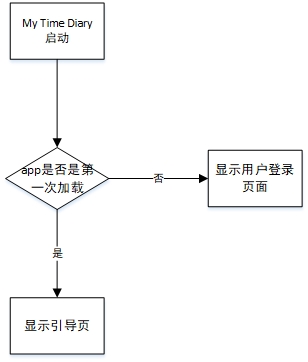


图4.1 My Time Diary-首页闪屏页的跳转逻辑

## 模块二---用户注册/登录界面逻辑

### 局部数据结构

1. 登录界面

struct Logining Activity {

private TextView tv\_main\_title;

private TextView tv\_back,tv\_register,tv\_find\_psw;

private Button btn\_login;

private String userName,psw,spPsw;

private EditText et\_user\_name,et\_psw; }

Operation:

void onCreate(Bundle savedInstanceState);

void init();

String readPsw(String userName);

void saveLoginStatus(boolean status,String userName);

void onActivityResult(int requestCode,int resultCode,Intent data);

1. 注册界面

struct Register Activity {

private TextView tv\_main\_title; //标题

private TextView tv\_back; //返回按钮

private Button btn\_register; //注册按钮

//用户名，密码，再次输入的控件的获取值

private EditText et\_user\_name,et\_psw,et\_psw\_again;

private String userName,psw,pswAgain;

private RelativeLayout rl\_title\_bar; }

Operation:

void onCreate(Bundle savedInstanceState);

void init();

void getEditString();

boolean isExistUserName(String userName);

void saveRegisterInfo(String userName,String psw);

### 处理逻辑

处理逻辑如下图4.2所示：

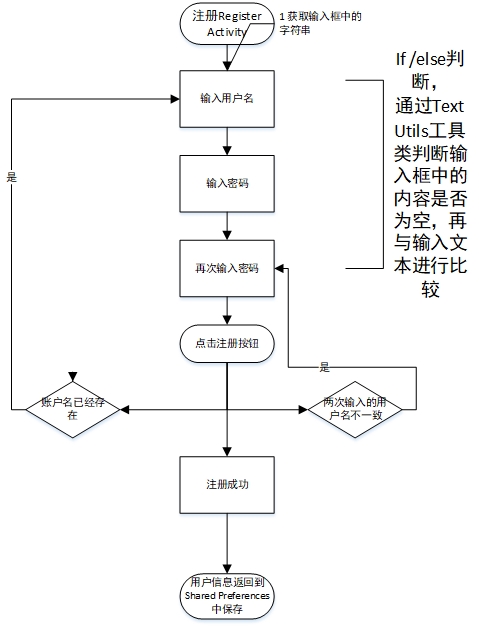


图4.2 My Time Diary注册/登录逻辑

## 模块三---主页逻辑实现

### 局部数据结构

struct MainActivity {

private TabLayout mTabLayout;

private ViewPager mViewPager;

private List<String> mTitle;

private List<Fragment> mFragment;

private long exitTime = 0; //按2次按钮退出程序 }

Operation:

void onCreate(Bundle savedInstanceState);

void initData();

void initView() ;

boolean onKeyDown(int keyCode, KeyEvent event);

### 处理逻辑

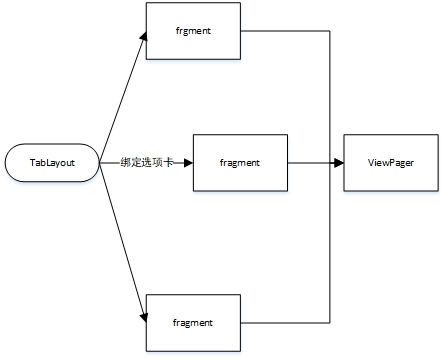


图4.3 My Time Diary主页逻辑

## 模块---加载第三方网络文章数据

### 局部数据结构以及处理逻辑

如下图4.4类图所示：

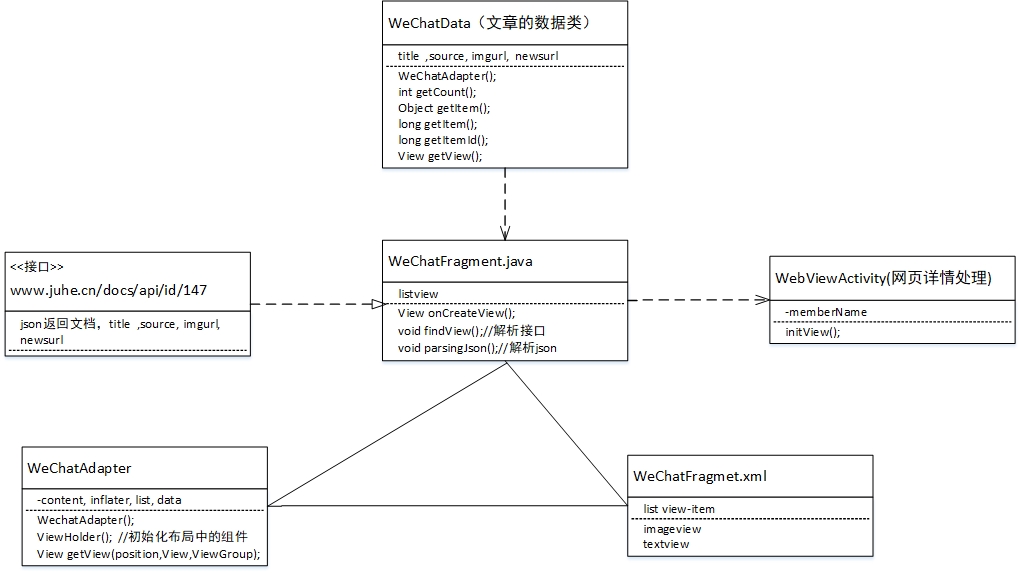


图4.4 第三方数据加载逻辑实现

# 交互界面设计

## 界面流概述

本实验项目展示的界面内容如下图5.1所示：

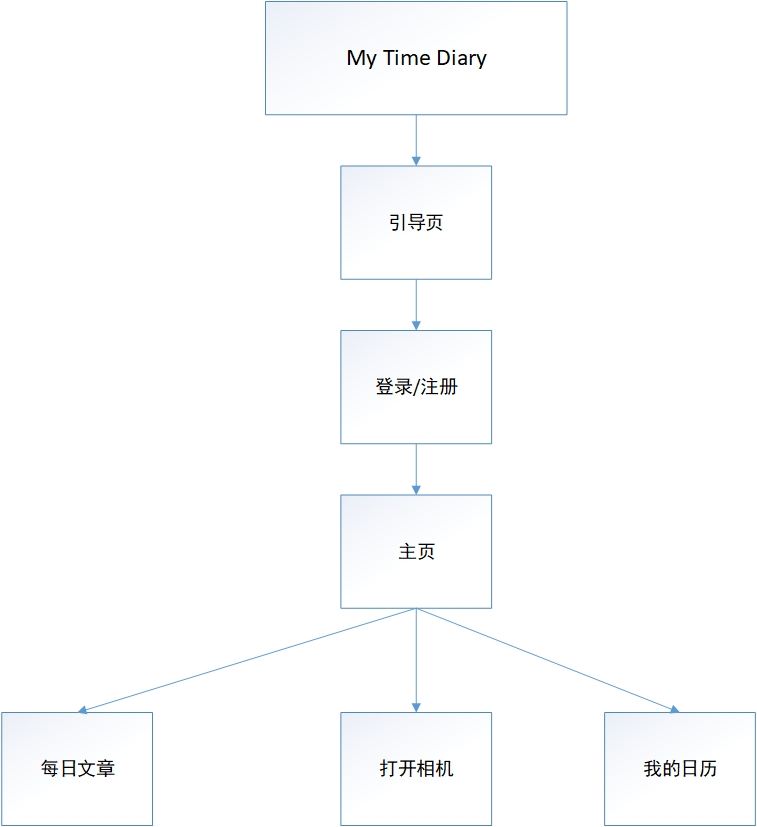


图5.1 软件界面流程图

## 闪屏页界面

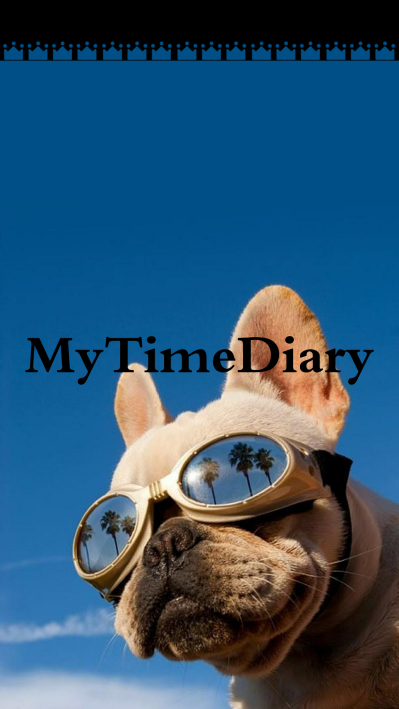


图5.2 闪屏页

## 引导页界面

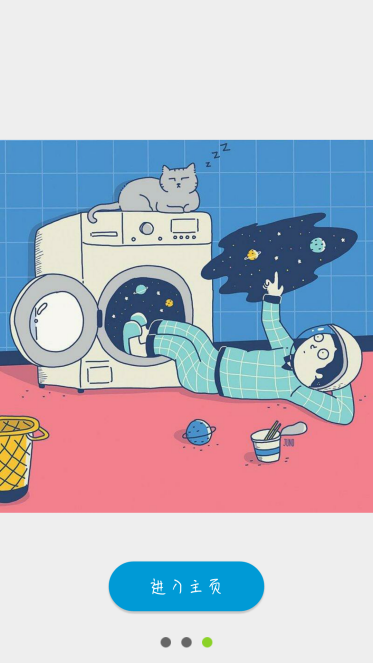
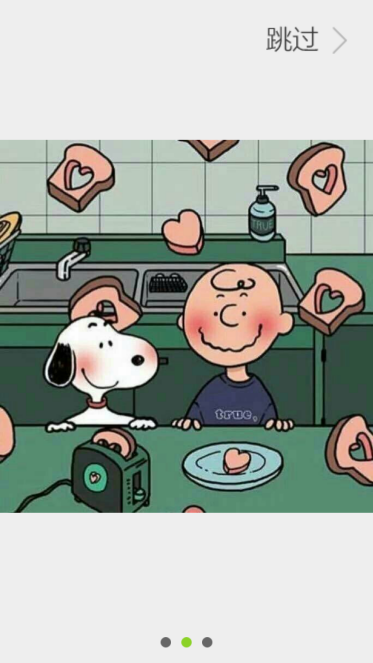


图5.3 My Time Diary-三张图片引导页

## 登录/注册界面



图5.4 My Time Diary-登录/注册界面展示

## 主页界面



图5.5 My Time Diary主页界面

## 功能详情界面



图5.6 各功能详情界面

# 系统测试

## 测试计划

首先对多个用户的注册以及登录信息进行测试；

其次接入第三方网络文章数据接口后，点击多篇文章详情，观察是否能正常显示功能；

## 测试用例描述

### 单元测试用例

用户登录以及注册测试：

1）测试方法：为每一个无效等价类设计至少一个测试用例；

2）测试用例如下表6.2.1所示：

快递查询测试用例：

搜集真实的快递单号进行测试，观察其与实际的物流信息是否相符，测试用例如下表6.2.2所示：

### 集成测试用例

|  |  |  |
| --- | --- | --- |
| 测试用例 | 期望结果 | 覆盖范围 |
| zhang | 请输入密码 | 输入框内容为空 |
| zhang,a,b | 2次密码不一致 | 2次密码不一致 |
| zhang,a,a | 注册成功 |  |
| zhang,a,a | 用户名已存在 | 用户名已经存在SP中 |

表6.2.1 用户登录以及注册测试用例

|  |  |  |
| --- | --- | --- |
| 测试用例 | 期望结果 | 覆盖范围 |
| sto, 3389986648409 | 正常显示物流信息 |  |
| zto,3389986648409 | 无显示 | 查无此快递 |

## 测试报告

1.用户登录以及注册模块，测试结果如下图6.3.1所示，从中可见，数据库已经成功保存用户的更改功能的信息，证明程序保存用户身份信息成功：

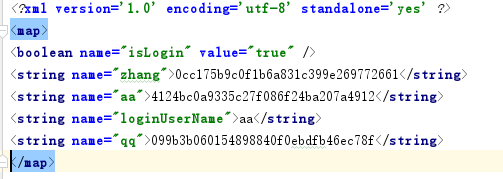


图6.3.1 Shared\_prefers\_info. xml中保存的用户信息

1. 快递查询模块测试结果如下图6.3.2所示，由实际测试用例可知，快递查询模块可以正常使用：

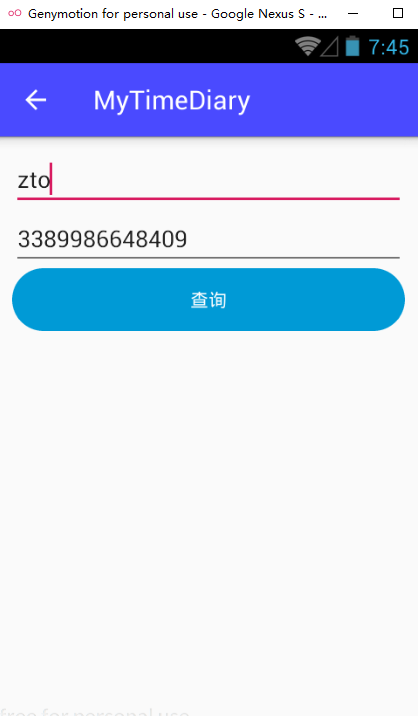


图6.3.2 快递查询模块测试用例

# 用户手册

## 安装配置手册

计算机Android Studio 环境中打开此项工程，需要的app build . gradle的文件内容为：

apply plugin: 'com.android.application'

android {

compileSdkVersion 28

defaultConfig {

applicationId "com.njupt.mytimediary"

minSdkVersion 15

targetSdkVersion 28

versionCode 1

versionName "1.0"

testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"

}

buildTypes {

release {

minifyEnabled false

proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'

}

}

}

dependencies {

implementation fileTree(dir: 'libs', include: ['\*.jar'])

implementation 'com.android.support:appcompat-v7:28.0.0'

implementation 'com.android.support.constraint:constraint-layout:1.1.3'

implementation 'com.android.support:design:28.0.0'

implementation 'com.android.support:support-v4:28.0.0'

testImplementation 'junit:junit:4.12'

androidTestImplementation 'com.android.support.test:runner:1.0.2'

androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'

/\*//Bmob

implementation('cn.bmob.android:bmob-sdk:3.5.0') {

exclude group: 'com.squareup.okhttp3', module: 'okhttp'

exclude group: 'io.reactivex:rxandroid'

exclude group: 'io.reactivex:rxjava'

exclude group: 'com.squareup.okio'

}\*/

implementation 'cn.bmob.android:bmob-sdk:3.6.8-rc3'

implementation "io.reactivex.rxjava2:rxjava:2.2.2"

implementation 'io.reactivex.rxjava2:rxandroid:2.1.0'

implementation 'com.squareup.okio:okio:2.1.0'

implementation 'com.google.code.gson:gson:2.8.5'

implementation 'com.kymjs.rxvolley:rxvolley:1.1.4'

implementation 'com.squareup.picasso:picasso:2.5.2'

implementation 'com.google.android.gms:play-services-maps:16.0.0'

/\* implementation'com.squareup.okhttp:3.9.0'\*/

}

在安卓手机上可以直接打开文件夹中的安装包.apk文件进行安装使用；

## 使用手册

用户打开软件，首先进行用户注册然后用户进行登录，软件主页展示3个内容，其中第一个选项用户可以浏览每日精选的微信文章，第二个选项卡实现打开相机的内容，第三个选项展示简单的日历展示功能，用户可以根据自己的功能需要使用软件。

## 维护手册

开发者需定期检查软件运行情况，统计使用者数量以及使用者反馈的问题，采取相应的措施解决程序运行中出现的问题。

# 项目总结

8.1 调试过程中遇到的问题及其解决方法

|  |  |
| --- | --- |
| 遇到的问题 | 解决方法 |
| 首页跳转时程序意外终止 | 新建工程，重新依据资料仔细编写代码，降低图片的分辨率，避免因内存过大，导致程序的内存泄漏 |
| 主页的滑动选项卡的功能不适配导致程序意外终止 | 主页采用的是Tab Layout+Fragment实现选项卡的功能，适配器需要仔细认真的写，任何一个在布局文件中的view,如果不在文件中进行适配以及声明都会导致程序的意外终止 |
| Fragment不同于activity的页面启动方法导致页面展示终止 | Fragment虽然也是activity的子类，但是在点击事件的页面的启动方法中稍有不同，需要先获取当前fragment所在的页面getactivity()再传值 |
| 调用第三方文章数据接口实现文章内容的加载时，只要加载外部图片等功能后，就会导致程序的意外终止 | 通过打印程序运行时的输出日志发现，程序可以正常的请求网络，能实现数据的架子啊，但是再实现具体的webview的展示页面时，会导致程序的加载问题，和程序的语句执行的逻辑顺序有关 |
| 查看数据库的数据表时，表中内容经常显示全部为null | 仔细查看代码，发现程序的设定逻辑为，先初始化表中所有的数据，默认设定为程序返回主页后重新加载此数据，所以先将模拟机停留在此界面，当模拟机执行到更改功能选择数据的时候，此时打开最新的数据表，即为最新需要展示的数据内容 |

8.2 总结和讨论

|  |  |
| --- | --- |
| 设计的优点 | 各个功能模块之间的耦合性低，独立性高，便于软件开发功能的扩展 |
| 设计的缺点 | 有些网络请求的内容尚未实现，尚未请求第三方网络的数据库，本地存储功能的导致多种功能的使用受限 |
| 时间性能 | 尚未请求网络加载的软件的时间性能较高，请求网络加载后的时间性能会降低 |
| 空间性能 | 主要占据内存空间的是一些布局文件中用到的图片的内存比较大 |
| 改进方法 | 还可以实现提升功能，将功能选择的逻辑实现 |

**参考文献**

[1]慕课网课程：**<https://coding.imooc.com/learn/list/81.html>**

[2]于连林编著，《爱上Android》，人民邮电出版社

[3]CSDN博客

[4]Git Hub开源网站上的代码的引用