

学号：17104020058



信阳学院

XINYANG UNIVERSITY

《Android 应用系统开发》 课程设计报告

班 级:_____17 计科本 02_____

姓 名:_____李贤奇_____

设计题目:_____搜索集合_____

成 绩:_____

2020 年 6 月 17 日

目 录

1.引言.....	1
1.1 选题背景.....	1
1.2 选题意义.....	1
2.系统分析.....	1
2.1 系统可行性分析.....	1
2.2 系统需求分析.....	2
3.系统功能设计.....	2
3.1 系统总体设计.....	2
3.2 系统模块详细设计.....	3
3.2.1 菜单功能模块.....	3
3.2.2 首页功能模块.....	3
3.2.3 增加功能模块.....	4
3.3 数据库设计.....	4
3.4 其他分析.....	5
4.功能模块展示与实现.....	5
4.1 主页.....	5
4.2 header.....	5
4.3 map.....	7
4.4 time.....	7
4.5 increase.....	8
4.6 about.....	9
5.收获.....	10
6.附录.....	11

1.引言

1.1 选题背景

随着互联网的迅速发展，搜索引擎已成为人们必不可少的一部分。搜索引擎以其强大的抓取能力，抓取范围现阶段已经覆盖了全网百分之二十左右的网页。Google（谷歌）的网页覆盖量有 60 亿左右，百度也达到 10 亿，而且它们形成了它们自己的生存方式(出售技术、竞价排名、植入广告)。Google 被誉为互联网最伟大的公司，百度成为中国第二大互联网公司，这都宣告着搜索时代的大发展，搜索引擎给在互联网遨游的人们带来了巨大的方便。

另一方面，站点门户网站的搜索也越来越普及，当一个网站做大做强后，其内容必然非常丰富，用户想要查找到需要的内容（例如昨天刚看过的视频，但是今天已经不在首页展示了），或者想要看到这方面的相关内容，都需要搜索引擎的支持

1.2 选题意义

当用户可能只对一个方面的知识非常感兴趣，对其他方面的知识不感兴趣时候，就需要专题搜索，例如用户只对新闻专题类内容感兴趣，新闻类专题搜索显然能更好满足用户。专题搜索引擎的效率由于针对专题类内容，比网页搜索的准确度更高，受到很大关注，例如百度新闻，涵盖互联网、国内、国际、军事、财经等各个方面的新闻，用户只需输入自己感兴趣的新闻题目或者内容，各大新闻网站关于这则消息的新闻立马罗列出来。软件的意义在于让用户添加自己喜欢的专题，集合了用户真正喜欢的内容。

2.系统分析

2.1 系统可行性分析

本系统在开发过程中，Android 客户端使用 Android Studio 开发工具进行开发，模拟器选用了高性能的夜神模拟器，可调试不同设备，不同分辨率，适配市面上大部分手机。Web 服务器端用 tomact 作为图片发送端，系统中加入第三方 UI 框架，在 github 上开源，可以轻松获取。

2.2 系统需求分析

用户可自定义添加自己喜欢的内容，增加删除简单，交互性良好，并集成日历，地图功能。程序运行流程如图 2-2 所示。

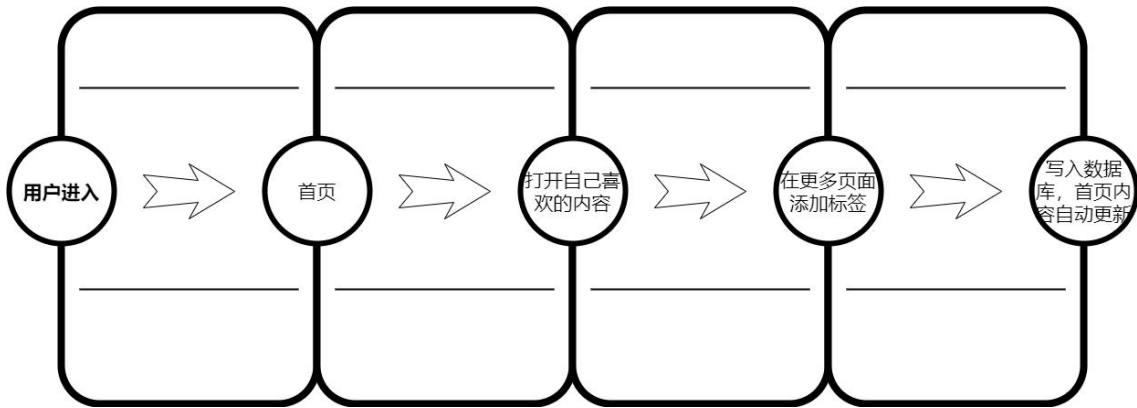


图 2-2 系统运行逻辑图

3.系统功能设计

3.1 系统总体设计

本系统主要划分为六个部分，一个欢迎页，五个具体功能页面，整个程序框架功能如图 3-1 所示。

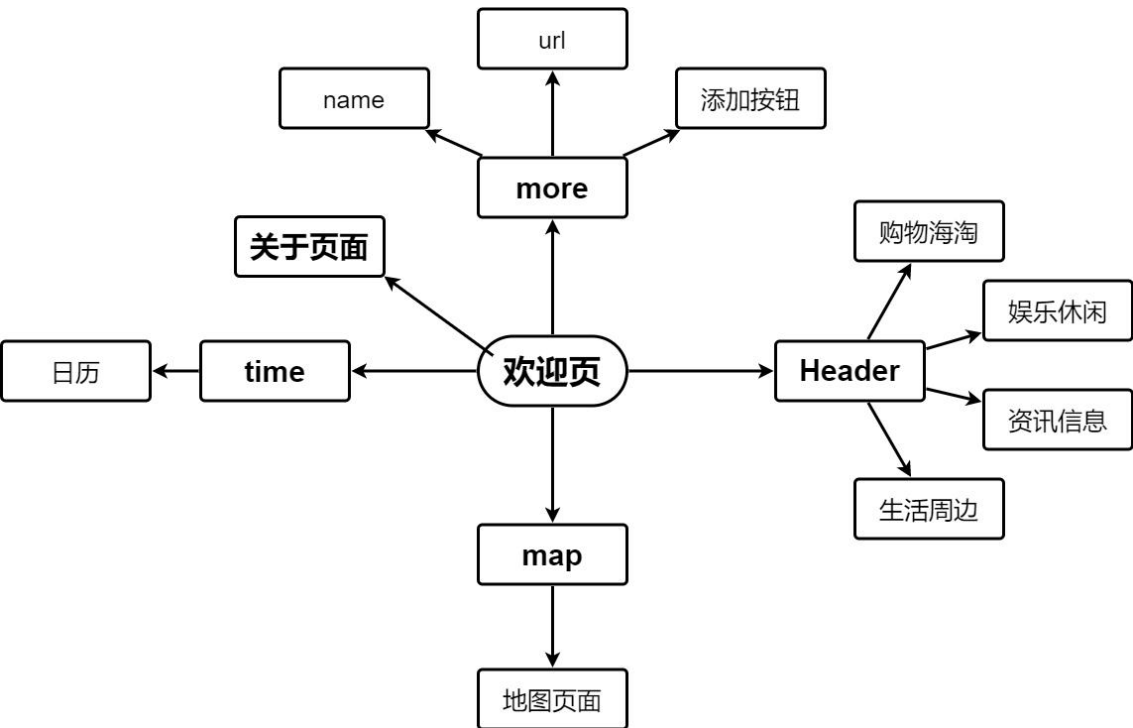


图 3-1 系统整体框架图

3.2 系统模块详细设计

3.2.1 菜单功能模块

主页可以跳转到五个不同的模块，通过一个弹出弹入的动画，完成不同页面的跳转，如图 3-2-1 所示。

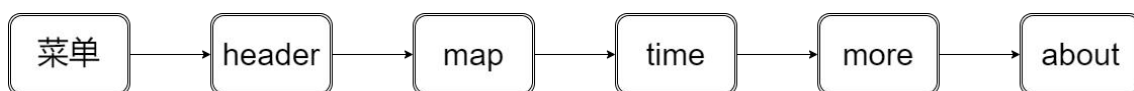


图 3-2-1 菜单跳转流程图

3.2.2 首页功能模块

首页由四个 fragment 组成，分为四个不同的模块，点击上方选项卡，可以滑动跳转，每一个模块由多个 item 组成，item 有相应具体的功能，程序运行流程如图 3-2-2 所示。

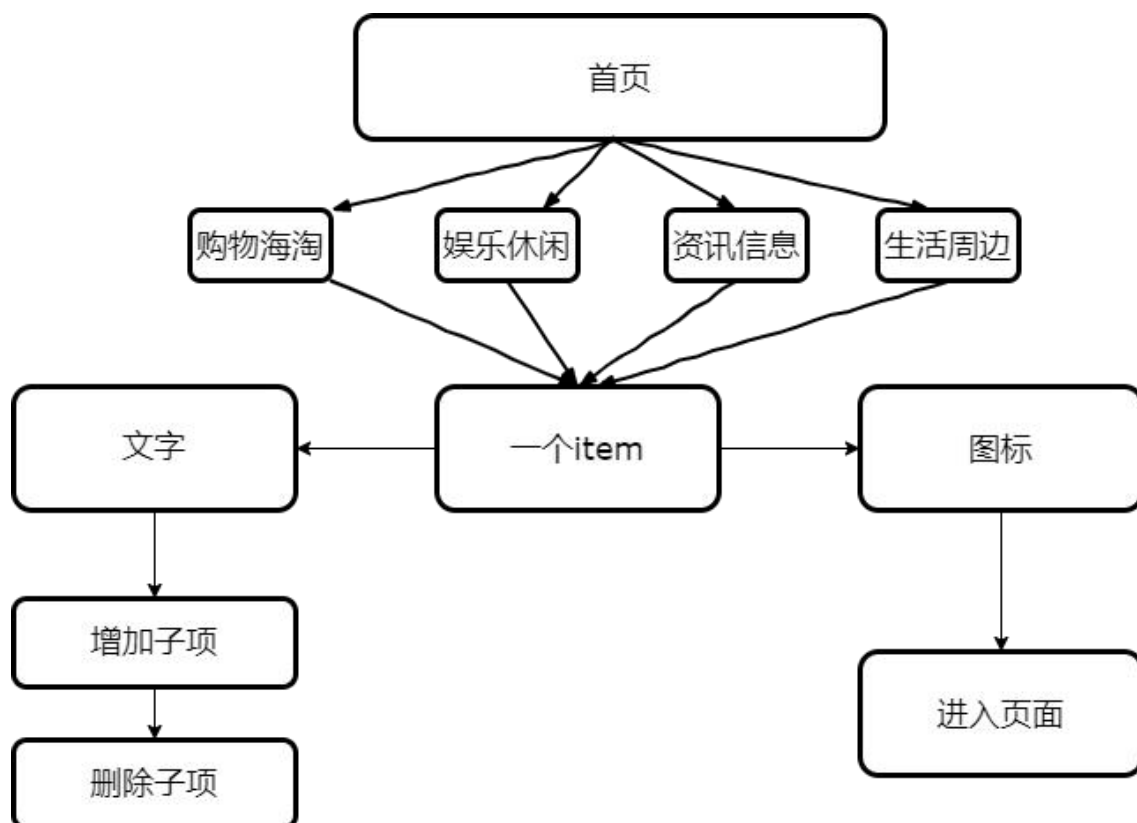


图 3-2-2 首页运行流程图

3.2.3 增加功能模块

增加功能可以为后端数据库添加相应的数据，将添加的数据在首页显示出来。流程图如图 3-2-3 所示。

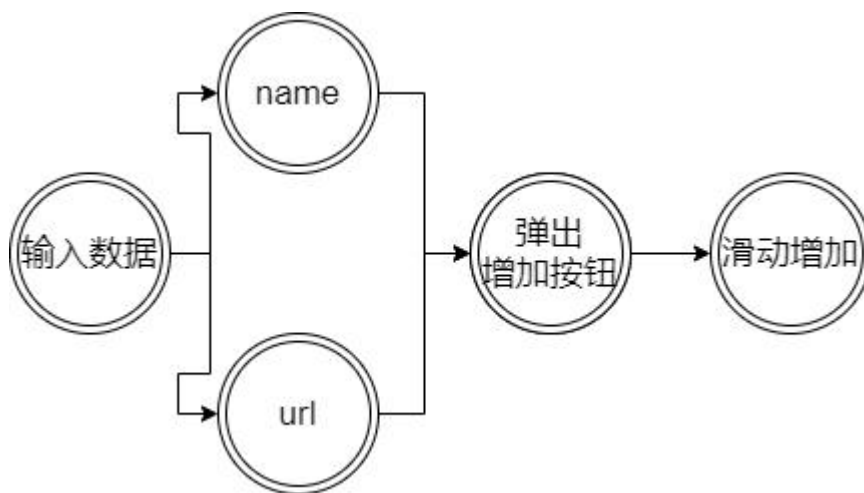


图 3-2-3 增加功能流程图

3.3 数据库设计

数据库的设计主要为了首页内容的动态生成，以及保存用户操作信息，主要有 shopping,entertainment,information,life 四张表，表的内容相同，包含字段 name,url,icon,itme_content,item_url。数据库 E-R 图如 3-3 所示。

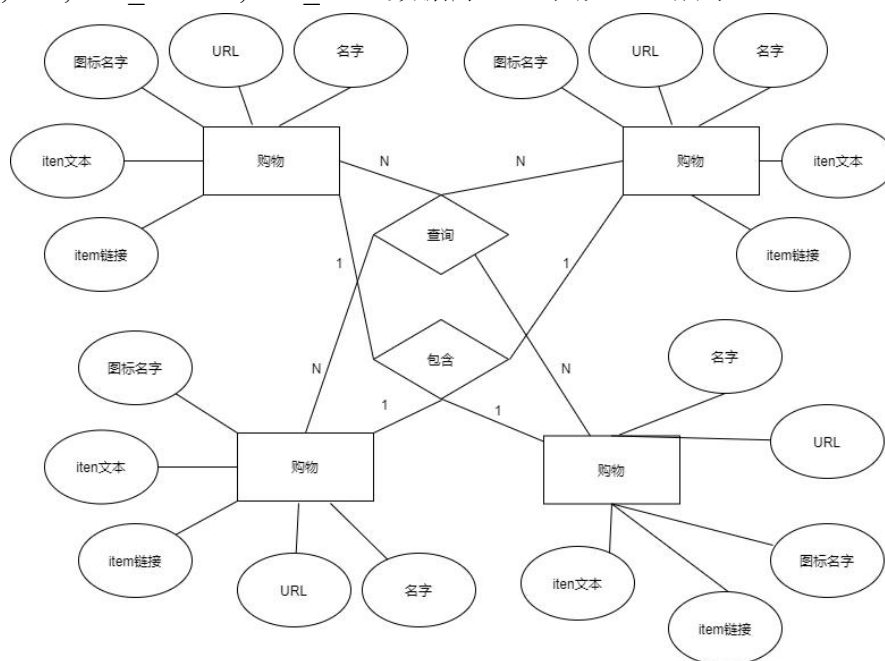


图 3-3 数据库 E-R 图

3.4 其他分析

剩下的功能主要为了填充页面，内容简化，提升用户体验。

4.功能模块展示与实现

4.1 主页

主页如图 4-1-1 所示，功能分为三个部分，1.顶部下拉可以刷新图片的内容 2.图片通过 Glide 网络加载 3.菜单动态弹出实现跳转，如图 4-1-2 所示。



图 4-1-1 主页图

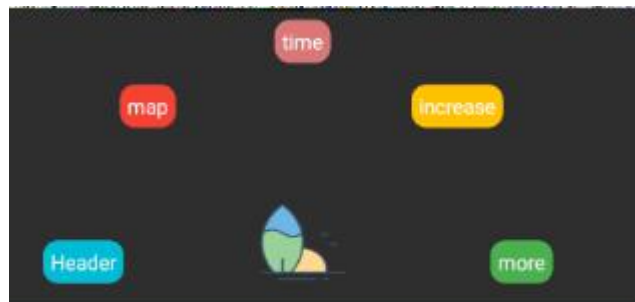


图 4-1-2 菜单跳转图

4.2 header

header 的内容是由 4 个 fragment 组成，可以左右滑动切换选项卡，如图 4-2-1 所示。



图 4-2-1 首页图

单一的 item 如图 4-2-2 所示，功能主要实现有，下拉可以展开 item，展示当前 item 的子项，点 x 可以删除子项。其中，开头颜色图标皆为随机生成，每次进入颜色图标都不相同，点击文字例如京东，可以显示两个隐藏的控件，一个删除当前 item，一个增加一个 item 的子项。增加子项会弹出两个对话框，如图 4-2-3 所示。

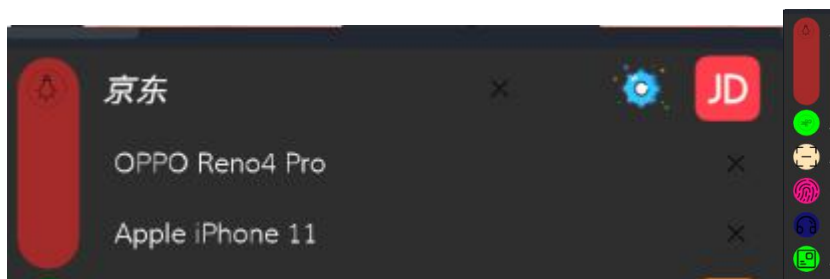


图 4-2-2item 功能图

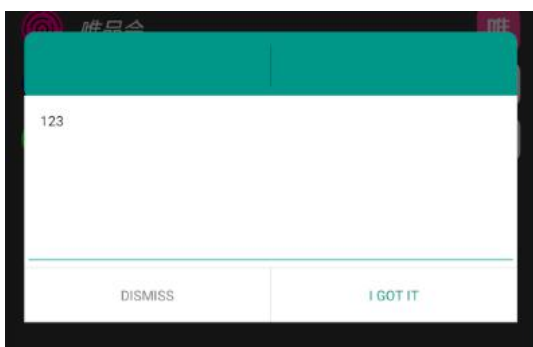


图 4-2-3 输入框图

填写内容后，做相应处理，提交内容给数据库，item 变化，京东的图标点击可以进入一个内置的浏览器。运行如 4-2-4 所示。



图 4-2-4 浏览器图

4.3 map

此页直接集成了百度地图的 API，显示出一个地图，如图 4-3 所示。

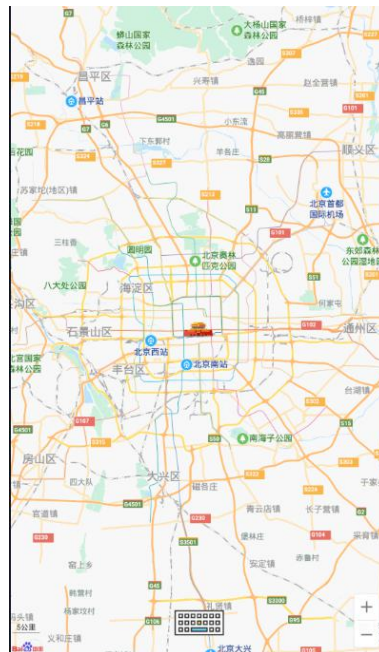


图 4-3 地图

4.4 time

此页用了一个最简单的日历。实现了点击菜单可以隐藏，显示，如图 4-4 所示。

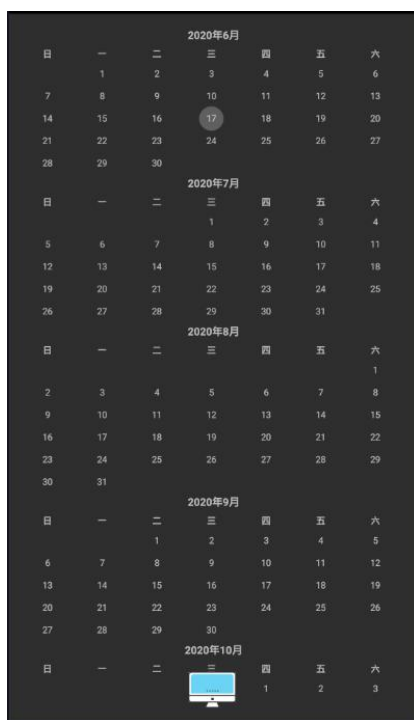


图 4-4 日历图

4.5 increase

为首页添加用户喜欢的 item 项。Name 和 Url 点击后展开，并在 Url 填写后显示出一个提交按钮，如图 4-5-1 所示。

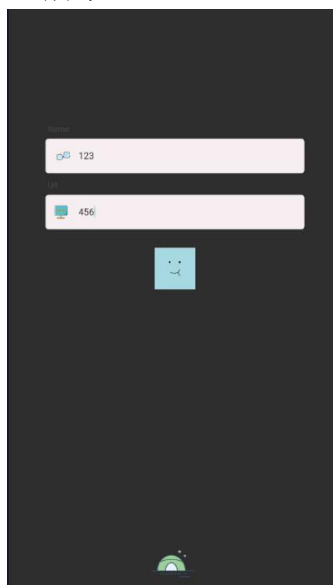


图 4-5 增加 item 图

点击提交，如果程序第一次点击，底部会出现一个提示框，并出现四个滑动按钮，供用户选择具体的分类，点击成功后，出现提示框提示成功。并隐藏其他控件，如图 4-5-2 所示



图 4-5-2 更多页面图

4.6 about

直接套用 UI 框架生成，如图 4-6 所示。



图 4-6 关于页面图

5.收获

通过一系列功能的开发，并解决开发中各种 bug，对个人编程能力有了很好的提升，对于 android 整个架构有了系统化的进一步认识，对各个主流框架，主流功能有了详细的了解，对于 UI 设计，页面布局也显著提升。积累下 app 实际项目经验，加强了本学期 android 知识的基础，对老师的讲解有了更深一步的认识。

6.附录

Maintivity.java

```
package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
import android.animation.AnimatorSet;
import android.animation.ObjectAnimator;
import android.content.Intent;
import android.content.SharedPreferences;
import android.database.sqlite.SQLiteDatabase;
import android.graphics.drawable.Drawable;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.Gravity;
import android.view.View;
import android.view.animation.BounceInterpolator;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import com.bumptech.glide.Glide;
import com.diegodobelo.expandingview.ExpandingItem;
import com.diegodobelo.expandingview.ExpandingList;
import android.widget.Toast;
import com.eminayar.panter.DialogType;
import com.eminayar.panter.PanterDialog;
import com.eminayar.panter.interfaces.OnTextInputConfirmListener;
import com.thefinestartist.finestwebview.FinestWebView;
import com.yalantis.phoenix.PullToRefreshView;
import androidx.core.content.ContextCompat;
import java.lang.reflect.Field;
import java.util.Random;
import blcs.lwb.utils.greendao.DaoMaster;
import blcs.lwb.utils.greendao.DaoSession;
import blcs.lwb.utils.greendao.testDao;
```

```
//模拟器运行用本地图片加载会卡顿，所以采用网络加载图片方法
// 真机流畅，所以打包出来的 apk 用的本地加载图片方法
```

//真机网络加载图片会加载不出来

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {
    Button button = null;
    TextView t1, t2, t3, t4, t5 = null;
    TextView ti;
    boolean LockMenu = false;
    private ExpandingList mExpandingList;
    boolean tag = true;
    private static DaoSession daoSession;
    private static testDao sdao;
    private String name;
    private String url;
    private String icon;
    private String item_content;
    private String item_url;
    private PullToRefreshView mPullToRefreshView;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        button = (Button) findViewById(R.id.button1);
        t1 = (TextView) findViewById(R.id.tV1);
        t2 = (TextView) findViewById(R.id.tV2);
        t3 = (TextView) findViewById(R.id.tV3);
        t4 = (TextView) findViewById(R.id.tV4);
        t5 = (TextView) findViewById(R.id.tV5);
        Random random = new Random();
        ImageView myim=(ImageView)findViewById(R.id.myimage);
        // 网络加载图片方法
        Glide.with(this).load("http://47.101.144.65/image/0.jpg").into(myim);
        // 本地加载图片方法
        // myim.setImageResource(R.drawable.p0);

        // 下拉刷新
        mPullToRefreshView = (PullToRefreshView) findViewById(R.id.pull_to_refresh);
        mPullToRefreshView.setOnRefreshListener(new PullToRefreshView.OnRefreshListener()
        {
            @Override
            public void onRefresh() {
                mPullToRefreshView.postDelayed(new Runnable() {
                    @Override
                    public void run() {
                        myim.setVisibility(View.GONE);
                        mPullToRefreshView.setRefreshing(false);
                    }
                }, 1000);
            }
        });
    }
}
```

```

        int i=random.nextInt(20);
//        网络加载速度较慢，故直接将文件放在了本地，影响不大
//        网络获取图片方法
        String b="" + i;
        String a="http://47.101.144.65/image/" + b + ".jpg";
        Glide.with(MainActivity.this).load(a).into(myim);
//        本地获取的方式
//        myim.setImageResource(pic[i]);
        myim.setVisibility(View.VISIBLE);

    }
    },1500);
    }
});

button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        it();
    }
});
mExpandingList = findViewById(R.id.expanding_list_main);
initGreenDao();
//获取 GreenDao 数据库操作对象
DaoSession daoSession = MainActivity.getDaoSession();
//获取指定的操作对象
sdao = daoSession.getTestDao();
try {
    database();
} catch (NoSuchFieldException e) {
    e.printStackTrace();
} catch (IllegalAccessException e) {
    e.printStackTrace();
}
}
//    createItems();

}
//1.菜单相关
//    封装具体方法
    public void it(){
        if (!LockMenu) {
            LockMenu = true;

```

```

        openAnim(t1, 0, 5, 300);
        openAnim(t2, 1, 5, 300);
        openAnim(t3, 2, 5, 300);
        openAnim(t4, 3, 5, 300);
        openAnim(t5, 4, 5, 300);
    } else {
        LockMenu = false;
        closeAnim(t1, 0, 5, 300);
        closeAnim(t2, 1, 5, 300);
        closeAnim(t3, 2, 5, 300);
        closeAnim(t4, 3, 5, 300);
        closeAnim(t5, 4, 5, 300);
    }
}

```

//打开菜单

```

public void openAnim(View view, int index, int num, int radius) {
    if (view.getVisibility() != View.VISIBLE) {
        view.setVisibility(View.VISIBLE);
    }
    double angle = Math.toRadians(180) / (num - 1) * index;
    int translationX = -(int) (radius * Math.cos(angle));
    int translationY = -(int) (radius * Math.sin(angle));

    ObjectAnimator one = ObjectAnimator.ofFloat(view, "translationX", 0, translationX);
    ObjectAnimator two = ObjectAnimator.ofFloat(view, "translationY", 0, translationY);
    ObjectAnimator three = ObjectAnimator.ofFloat(view, "rotation", 0, 360);
    ObjectAnimator four = ObjectAnimator.ofFloat(view, "scaleX", 0f, 1f);
    ObjectAnimator five = ObjectAnimator.ofFloat(view, "scaleY", 0f, 1f);
    ObjectAnimator six = ObjectAnimator.ofFloat(view, "alpha", 0f, 1);
    AnimatorSet set = new AnimatorSet();
    set.playTogether(one, two, three, four, five, six);
    set.setDuration(2000);
    //回弹效果
    set.setInterpolator(new BounceInterpolator());
    set.start();
}

```

//关闭菜单

```

public void closeAnim(View view, int index, int num, int radius) {
    if (view.getVisibility() != View.VISIBLE) {
        view.setVisibility(View.VISIBLE);
    }
    double angle = Math.toRadians(180) / (num - 1) * index;

```



```

        Log.e("angle", angle + "");
        int translationX = -(int) (radius * Math.cos(angle));
        int translationY = -(int) (radius * Math.sin(angle));

        ObjectAnimator one = ObjectAnimator.ofFloat(view, "translationX", translationX, 0);
        ObjectAnimator two = ObjectAnimator.ofFloat(view, "translationY", translationY, 0);
        ObjectAnimator three = ObjectAnimator.ofFloat(view, "rotation", 0, 360);
        ObjectAnimator four = ObjectAnimator.ofFloat(view, "scaleX", 1f, 0f);
        ObjectAnimator five = ObjectAnimator.ofFloat(view, "scaleY", 1f, 0f);
        ObjectAnimator six = ObjectAnimator.ofFloat(view, "alpha", 1f, 0f);

        AnimatorSet set = new AnimatorSet();
        set.playTogether(one, two, three, four, five, six);
        set.setDuration(2000);
        //回弹效果
        set.setInterpolator(new BounceInterpolator());
        set.start();
    }
}
//2.数据库相关
public void database() throws NoSuchFieldException, IllegalAccessException {
    //    判断是否第一次执行
    SharedPreferences
sharedPreferences=this.getSharedPreferences("share",MODE_PRIVATE);
    boolean isFirstRun=sharedPreferences.getBoolean("isFirstRun", true);
    SharedPreferences.Editor editor=sharedPreferences.edit();
    if(isFirstRun){
        //        Toast.makeText(MainActivity.this, "第一次运行",
        Toast.LENGTH_SHORT).show();
        editor.putBoolean("isFirstRun", false);
        editor.commit();
        data_init ds = new data_init();
        ds.init();
        init();
    }else{
        //        Toast.makeText(MainActivity.this, "不是第一次运行",
        Toast.LENGTH_SHORT).show();
        init();
    }
    //    首次插入数据
    //    insert("京东","https://www.jd.com/","jingdong","OPPO Reno4 Pro|Apple iPhone
11","https://item.jd.com/100007357987.html|https://item.jd.com/100008348542.html");
    //    insert("淘宝
","https://www.taobao.com/","taobao","Switch","https://detail.tmall.com/item.htm?spm=a230r.1.14.

```

```
1.39ff7830q4UNqd&id=615922280920&cm_id=140105335569ed55e27b&abbucket=15&sku_properties=5919063:57445706;122216431:27772");  
//      初始化
```

```
}
```

```
//      public void insert(String name,String Url,String Icon,String Item_content,String Item_url){  
//          test sm=new test();  
//          sm.setName(name);  
//          sm.setUrl(Url);  
//          sm.setIcon(Icon);  
//          sm.setItem_content(Item_content);  
//          sm.setItem_url(Item_url);  
//          sdao.insert(sm);  
//      }
```

```
public void delete(String content){  
    test sm=sdao.queryBuilder().where(testDao.Properties.Name.eq(content)).unique();  
    daoSession.delete(sm);  
}
```

```
public void update(String name,String item_content,String item_url){  
    test sm=sdao.queryBuilder().where(testDao.Properties.Name.eq(name)).unique();  
    sm.setItem_content(item_content);  
    sm.setItem_url(item_url);  
    daoSession.update(sm);  
}
```

```
public void item_add(String name, String item,String url){  
    test sm=sdao.queryBuilder().where(testDao.Properties.Name.eq(name)).unique();  
//    item.content  
    item_content=sm.getItem_content();  
    String[] a = item_content.split("\\|");  
    List<String> list1= Arrays.asList(a);  
    List<String> arrList = new ArrayList<String>(list1);  
    arrList.add(item);  
    String str= TextUtils.join("|", arrList);  
//    item.url  
    item_url=sm.getItem_url();  
    String[] b = item_url.split("\\|");  
    List<String> list2= Arrays.asList(b);
```

```

        List<String> arrList2 = new ArrayList<String>(list2);
        arrList2.add(url);
        String str2=TextUtils.join("|", arrList2);
        update(name,str,str2);

    }

    public void item_remove(String name, String item,String url){
        test sm=sdao.queryBuilder().where(testDao.Properties.Name.eq(name)).unique();
        // item.content
        item_content=sm.getItem_content();
        String[] a = item_content.split("\\|");
        List<String> list1= Arrays.asList(a);
        List<String> arrList = new ArrayList<String>(list1);
        arrList.remove(item);
        String str=TextUtils.join("|", arrList);
        // Toast.makeText(MainActivity.this, str, Toast.LENGTH_SHORT).show();
        // item.url
        item_url=sm.getItem_url();
        String[] b = item_url.split("\\|");
        List<String> list2= Arrays.asList(b);
        List<String> arrList2 = new ArrayList<String>(list2);
        arrList2.remove(url);
        String str2=TextUtils.join("|", arrList2);
        // Toast.makeText(MainActivity.this, str2, Toast.LENGTH_SHORT).show();
        update(name,str,str2);

    }

    public void init() throws NoSuchFieldException, IllegalAccessException {
        List<test> list=sdao.queryBuilder().list();
        for(int i=0;i<list.size();i++){
            name=list.get(i).getName();
            url=list.get(i).getUrl();
            icon=list.get(i).getIcon();
            item_content=list.get(i).getItem_content();
            String[] a = item_content.split("\\|");
            item_url=list.get(i).getItem_url();
            String[] b = item_url.split("\\|");
            // 通过字符串的方式获取 R.mipmap.的内容
            Field field=R.mipmap.class.getField(icon);
            int Icon= field.getInt(new R.mipmap());
            addItem(name,a,b,colors[(int) (Math.random()*(colors.length-1))],icons[(int)
(Math.random()*(icons.length-1))],Icon,url);

        }
    }

```

```

    }
    private void initGreenDao() {
        DaoMaster.DevOpenHelper helper = new DaoMaster.DevOpenHelper(this, "Smile.db");
        SQLiteDatabase db = helper.getWritableDatabase();
        DaoMaster daoMaster = new DaoMaster(db);
        daoSession = daoMaster.newSession();
    }
    public static DaoSession getDaoSession() {
        return daoSession;
    }
}
// 3.具体 item 相关
private void createItems() {
    addItem("京东", new String[]{"OPPO Reno4 Pro", "Apple iPhone 11", },new
String[]{"https://item.jd.com/100007357987.html","https://item.jd.com/100008348542.html"},R.col
or.pink, R.mipmap.ic_ghost,R.mipmap.jingdong, "https://www.jd.com/");
    addItem("淘宝", new String[]{"Switch"},new
String[]{"https://detail.tmall.com/item.htm?spm=a230r.1.14.1.39ff7830q4UNqd&id=615922280920
&cm_id=140105335569ed55e27b&abucket=15&sku_properties=5919063:57445706;122216431:2
7772"}, R.color.blue, R.mipmap.ic_ghost,R.mipmap.taobao,"https://www.taobao.com/");
}
private void addItem(String title, String[] subItems, String[] url_list,int colorRes, int iconRes,
int iconRes2, final String url) {
    //获取到整个 item 的布局
    final ExpandingItem item = mExpandingList.createNewItem(R.layout.expanding_layout);

    //如果获取成功，设置 item 的颜色和 icon
    if (item != null) {
        item.setIndicatorColorRes(colorRes);
        item.setIndicatorIconRes(iconRes);
        //设置分割线
        View fg=(View) item.findViewById(R.id.fengexian);
        Drawable back = ContextCompat.getDrawable(this,colorRes);
        fg.setBackground(back);
        //这里设置了 item 的标题,绑定 title 的点击
        TextView ti=((TextView) item.findViewById(R.id.title));
        ti.setText(title);
        ti.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View view) {
                ImageView am=(ImageView)
item.findViewById(R.id.add_more_sub_items);
                ImageView rm=(ImageView) item.findViewById(R.id.remove_it);
                if(tag){
                    am.setVisibility(View.VISIBLE);

```

```

        rm.setVisibility(View.VISIBLE);
        tag=false;
    }else{
        am.setVisibility(View.GONE);
        rm.setVisibility(View.GONE);
        tag=true;
    }
    }
});
//设置图标，自定义方法
    ImageView ib=(ImageView)item.findViewById(R.id.item_icon);
    ib.setImageResource(iconRes2);

    //根据传入的 subItems 的长度创建子 item
    item.createSubItems(subItems.length);
    for (int i = 0; i < item.getSubItemsCount(); i++) {
        //从 0 到最后一个，逐个获取
        final View view = item.getSubItemView(i);
        //设置相关标题，封装具体方法，传入 url 并绑定点击
        configureSubItem(item, view, subItems[i],url_list[i]);
    }
    //为每一个 item 的添加绑定事件
    item.findViewById(R.id.item_icon).setOnClickListener(new View.OnClickListener()
    {
        @Override
        public void onClick(View v) {
            Web(url);
            Uri uri = Uri.parse("https://www.baidu.com");
            Intent intent = new Intent(Intent.ACTION_VIEW, uri);
            startActivity(intent);
            WebView webView = (WebView) findViewById(R.id.web_view);
            webView.setVisibility(View.VISIBLE);
            //WebView 是否可以后退
            boolean canGoBack = webView.canGoBack();
            //WebView 后退
            webView.goBack();
            //WebView 是否可以前进
            boolean canGoForward = webView.canGoForward();
            //WebView 前进
            webView.goForward();
            //以当前的 index 为起始点前进或者后退到历史记录中指定的 steps
            //如果 steps 为负数则为后退，正数则为前进
            boolean canGoBackOrForward = webView.canGoBackOrForward(step);

```

```

//
//
//          webView.getSettings().setJavaScriptEnabled(true);
//          webView.setWebViewClient(new WebViewClient());
//          webView.loadUrl("http://www.baidu.com");
//          new
FinestWebView.Builder(MainActivity.this).show("http://www.baidu.com");
//          Toast.makeText(MainActivity.this,"你点击了确定按钮
",Toast.LENGTH_LONG ).show();
        }
    });
    ImageView am=(ImageView) item.findViewById(R.id.add_more_sub_items);
    am.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            showInsertDialog(new OnItemCreated() {
                @Override
                public void itemCreated(String title,String url) {
                    //创建一个新的最小子项
                    View newSubItem = item.createSubItem();
                    configureSubItem(item, newSubItem, title,url);
//                    TextView name=((TextView) findViewById(R.id.title));
                    String Name=ti.getText().toString();
                    item_add(Name,title,url);
                }
            });
        }
    });
    //删除
    item.findViewById(R.id.remove_it).setOnClickListener(new View.OnClickListener()
    {
        @Override
        public void onClick(View v) {
            mExpandingList.removeItem(item);
            TextView name=((TextView) findViewById(R.id.title));
            delete(name.getText().toString());
        }
    });
}

private void configureSubItem(final ExpandingItem item, final View view, String
subTitle,String url) {
    //为最小子项设置标题
    TextView st=((TextView) view.findViewById(R.id.sub_title));
    st.setText(subTitle);

```

```

//设置绑定浏览器跳转
st.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Web(url);
    }
});
//设置绑定删除当前最小子项
view.findViewById(R.id.remove_sub_item).setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        item.removeSubItem(view);
        String subT=st.getText().toString();
        TextView name=((TextView) item.findViewById(R.id.title));
        String Name=name.getText().toString();

        Toast.makeText(MainActivity.this, Name+subT+url,
Toast.LENGTH_SHORT).show();
        item_remove(Name,subT,url);

    }
});
//    rb.setChecked(true);
}
private void showInsertDialog(final OnItemCreated positive) {
//    设置传回对象，方便回传
final String[] Text = new String[1];
final String[] Url = new String[1];

//两个输入框，但不知什么原因，第二个输入框反而先执行，所以将输入框位置对换
new PanterDialog(this)
    .setHeaderBackground(R.drawable.duihuakuang)
    .setHeaderLogo(R.drawable.duihuakuang_zhong)
    .setDialogType(DialogType.INPUT)
    .isCancelable(true)
    .input("URL:\n"+"http://\n"+"https://",
        null, new
            OnTextInputCompleteListener() {
                @Override
                public void onTextInputConfirmed(String text) {
                    Url[0]=text;
                    //        将输入框内容传回
                    positive.itemCreated(Text[0],Url[0]);
                }
            }
    );
}

```

```

//                                Toast.makeText(MainActivity.this, text,
Toast.LENGTH_LONG).show();
                                }
                                })
                                .setPositive("I GOT IT")
                                .setNegative("DISMISS")
                                .show();

    new PanterDialog(this)
        .setHeaderBackground(R.drawable.content)
        .setHeaderLogo(R.drawable.duihuakuang_zhong)
        .setDialogType(DialogType.INPUT)
        .isCancelable(true)
        .input("NAME:",
            null, new
                OnTextInputCompleteListener() {
                    @Override
                    public void onTextInputConfirmed(String text) {
                        Text[0] =text;
                        //                                Toast.makeText(MainActivity.this,Text[0],
Toast.LENGTH_LONG).show();
                    }
                })
        .setPositive("I GOT IT")
        .setNegative("DISMISS")
        .show();

}
//接口，传回第一个输入框中的 name，第二个输入框中的 url
interface OnItemCreated {
    void itemCreated(String title, String url);
}
//    自定义浏览器相关方法
private void Web(String url){
    new FinestWebView.Builder(MainActivity.this).theme(R.style.FinestWebViewTheme)
        .titleDefault("Dribbble")
        .toolbarScrollFlags(0)
        .statusBarColorRes(R.color.blackPrimaryDark)
        .toolbarColorRes(R.color.blackPrimary)
        .titleColorRes(R.color.finestWhite)
        .urlColorRes(R.color.blackPrimaryLight)
        .iconDefaultColorRes(R.color.finestWhite)
        .progressBarColorRes(R.color.finestWhite)
        .swipeRefreshColorRes(R.color.blackPrimaryDark)

```



```

        .menuSelector(R.drawable.selector_light_theme)
        .menuTextGravity(Gravity.CENTER_VERTICAL | Gravity.RIGHT)
        .menuTextPaddingRightRes(R.dimen.defaultMenuTextPaddingLeft)
        .dividerHeight(0)
        .gradientDivider(false)
        //
        .setCustomAnimations(R.anim.slide_up, R.anim.hold,
R.anim.hold, R.anim.slide_down)
        .setCustomAnimations(R.anim.slide_left_in, R.anim.hold, R.anim.hold,
            R.anim.slide_right_out)
        //
        .setCustomAnimations(R.anim.fade_in_fast,
R.anim.fade_out_medium, R.anim.fade_in_medium, R.anim.fade_out_fast)
        .disableIconBack(false)
        .disableIconClose(false)
        .disableIconForward(false)
        .disableIconMenu(false)
        .show(url);
    }
    @Override
    public void onClick(View v) {
        switch (v.getId()) {
            case R.id.tV1:
                Intent intent1=new Intent(MainActivity.this,One.class);
                startActivity(intent1);
                it();
                this.finish();

                break;
            case R.id.tV2:
                Intent intent2=new Intent(MainActivity.this,Two.class);
                startActivity(intent2);
                it();
                this.finish();
                break;
            case R.id.tV3:
                Intent intent3=new Intent(MainActivity.this,Three.class);
                startActivity(intent3);
                it();
                this.finish();
                break;
            case R.id.tV4:
                Intent intent4=new Intent(MainActivity.this,Four.class);
                startActivity(intent4);
                it();
                this.finish();

```

```

        break;
    case R.id.tv5:
        Intent intent5=new Intent(MainActivity.this,Five.class);
        startActivity(intent5);
        it();
        break;
    }
}
@Override
public void startActivity(Intent intent) {
    super.startActivity(intent);
    // activity 切换动画
    // 参数一： 新 Activity 进入的动画(最右侧进来)
    // 参数二： 旧 Activity 退出的动画(最左侧出去)
    overridePendingTransition(R.anim.fade_in, R.anim.fade_out);
}
}

```