

Movitech Framework

Overview

Movitech Framework, 这个项目是用来展示Android组对于一些技术的研究成果的载体。

Parts

1) Package and Class Introduction

2) Need to do

Packages and Classes Introduction

1. **com.movitech.framework.Activity**

BaseActivity

BaseActivity是一切Activity的父类，它包括一个用来关闭所有Activity的本地广播。除此之外，如果我们希望让所有Activity都做某件事情的话，我们需要将其放在BaseActivity中去做。

MainActivity

MainActivity是项目的真正主界面当所有数据加载完成后，用户看到的就是这个页面。他包括了一个ViewPager和一个PageSelectBar。

TestActivity

TestActivity用来测试某些功能，目前用来测试DataHandler的网络与数据库访问整合。

TradingResultActivity

TradingResultActivity是微信支付的回调类在回调之后所前往的页面。

2. **com.movitech.framework.adapter**

MainActivityPageAdapter

MainAcitivityPagerAdapter是ViewPagerAdapter的子类，用来管理MainActivity中的几个Fragments。

ViewPagerAdapter

ViewPagerAdapter是PagerAdapter的子类，它是View的PagerAdapter。

3. com.movitech.framework.constant

BroadcastActions

BroadcastActions包含了所有本地广播的广播标识，所有的本地广播标识都需要在这里注册。

Constant

Constant包含了应用程序中的一些重要的Flags。

ImageLoaderHelper

ImageLoaderHelper是ImageLoader的帮助类，它通过拼接Schema来帮助ImageLoader来定位图片资源。

PayMode

PayMode是用来标记使用的是微信支付还是支付宝支付等其他方式。

ReqCode

ReqCode包含了应用中所有的RequestCode和ResultCode，这些用在Activity间通信的Code需要在此定义。

Timeout

Timeout包含了网络超时常用时间。

WeatherCityCode

WeatherCityCode包含了城市的天气代码。

4. com.movitech.framework.db

DatabaseHelper

DatabaseHelper是使用OrmLiteSqliteOpenHelper的DatabaseHelper。是整个db操作的核心类。包括了建表操作和表结构升级操作。

IDbHandler

IDbHandler是继承于IDataHandler，由于IDataHandler是为了在View的组件层提供数据库和网络访问的统一接口，所以凡是IDataHandler要处理的方法（通常是获取数据的get/post方法），IDbHandler都需要有，同时，IDbHandler还需要具有一些插入表数据的方法（通常是set，或者说update方法）。

OrmliteDbHandler

OrmliteDbHandler是IDbHandler的实现类，使用了Ormlite开源库。通过OO的方式，向View层提供数据对象。需要注意的是，在编写set类型的方法时，需要作一些必要的逻辑判断，例如，set方法的参数为null的时候，是需要清空这个表，还是不做任何操作等。

5. com.movitech.framework.fragment

BaseFragment

BaseFragment是所有Fragment的根类。它使用getActivity方法初始化了mContext作为上下文对象。

6. com.movitech.framework.generic

Utils

Utils是静态工具类，包括Debug调试函数，标准Toast调用以及诸如防止多次点击的工具。

interfaces.ISDCard

SDCard相关操作接口。

SDCard

SDCard相关操作实现类。

7. com.movitech.framework.handler

IDataHandler

IDataHandler是为了对View层提供统一数据库与网络访问的接口，View层发出访问请求后，具体是通过数据库取数据还是通过网络来取数据由DataHandler得实现类来决定。同时，它还包含了一个DataCallback，用来给其实现类执行View层的对数据的获取成功或获取失败的回调。

DataHandler

DataHandler是IDataHandler的实现类，由DataHandler来决定通过数据库取数据还是通过网络来取数据。一般情况下，数据库用来进行界面数据的临时存储，当没有网络的时候，DataHandler可以决定从数据库中取出数据，返回给View层。

IFileHandler

IFileHandler是文件处理者的接口类。

FileHandler

FileHandler用来处理文件操作。

IImageHandler

IImageHandler是图片处理者的接口类。

UIImageHandler

UIImageHandler是使用了Universal Image Loader来控制图片的单例类，它可以加载普通图片，加载圆角图片，加载圆形图片以及选择图片来源，对图片进行切割处理等功能。

IWechatHandler

IWechatHandler是微信操作的接口类。

WechatHandler

WechatHandler是包含了对微信操作的一系列工具，目前只包含微信支付部分。

8. com.movitech.framework.listener

BannerViewOnItemClickListener

BannerViewOnItemClickListener是显示各种滚动广告的控件的监听器，可以对不同类型的广告进行不同的处理。

9. com.wudayu.model

DafUser

DafUser是UserModel的Object Json对象，目前是用来做本地数据库建表的例子。

DafWeather

DafWeather是WeatherModel的Object Json对象。

DatabaseConfigUtil

DatabaseConfigUtil是用来生成Ormlite建表文件的主程序，一般在新建了某些与数据库关联的model之后，需要用其来生成db_config文件，此文件在DatabaseHelper类中加载使用，为的是快速建立表，使用annotation会比较慢，且容易出现一系列问题。需要注意的是，在单独运行这个Util的时候，需要从运行的BuildPath中去除Android的jar引用。

TypedImage

TypedImage是对图片进行对象化的类，若需要上传图片，则需要将图片变成次对象类型。

10. com.wudayu.net

converter.JacksonConverter

JacksonConverter是为Retrofit编写的使用Jackson来解析对象的Json Converter，默认使用的mime_type是 application/json，可以在构造函数中选择text/html。

protocol.BaseResult

BaseResult是由服务器返回的通用Json所对应的对象，其所包含的字段在每个服务器返回的字段中都包含。

protocol.DafStringResult

DafStringResult是继承于BaseResult的数据对象，其承载了一个String对象。

protocol.DafUserResult

DafUserResult是继承于BaseResult的数据对象，其承载了一个DafUser对象。

protocol.WeatherResult

WeatherResult是根据气象台接口所单独设定的数据对象，接收天气Object Json。

service.ImageService

ImageService是用来向上传图片的Retrofit的Service。

service.UserService

UserService是用来完成用户相关的工作的Retrofit的Service。

service.WeatherService

WeatherService是用来向气象台请求天气信息的Retrofit的Service。

INetHandler

INetHandler是网络访问接口，其中还包括了网络地址等信息。

RetrofitNetHandler

RetrofitNetHandler是INetHandler的Retrofit实现，目前本项目仅打算使用Retrofit作为其网络访问框架。

11. com.movitech.framework.receiver

PushReceiver

PushReceiver是用来唤醒PushService的广播。它们是一对，通过向服务器拉取推送来进行推送显示的，PushService的开关是Constant.needPush。

12. com.movitech.framework.service

PushService

PushService是通过向服务器拉取推送来进行推送显示的，PushService的开关是Constant.needPush。

13. com.movitech.framework.views

imagezoom.GestureImageView

imagezoom整个包都是用来支持可控制放大缩小的GestureImageView的控件，可以直接将ImageView用GestureImageView替换。

BannerView

BannerView是用来展示广告的Banner控件，通过setRolling函数能够设置它的自动滚动机制。

BaseViewPager

BaseViewPager继承于support.v4.view.ViewPager。可以修改ViewPager切换时滚动的速度，但这部分代码暂时被注释了。

CountDownView

CountDownView是一个显示倒计时的控件。

DotPageIndicator

DotPageIndicator是给ViewPager显示个数以及当前所在界面的圆点标记控件。

NoSlideViewPager

NoSlideViewPager是一个不能用手指滑动的ViewPager控件。

PageIndicator

PageIndicator是DotPageIndicator的接口。

PageSelectBar

PageSelectBar是主页面下方控制不同标签页的按钮，它可以用来帮助ViewPager来选择相应的页面。

ProcessingDialog

ProcessingDialog是一个AlertDialog，当应用后台有任务的时候用来显示加载信息。

SelectPicPopupWindow

SelectPicPopupWindow是一个提示板，用来给用户选择以一种获取图片的方式。

SlideBarBaseView

SlideBarBaseView是用来在数据中有一系列字母开头的情况下通过滑动选择响应字母数据的控件，一般位于屏幕右侧。

SwitchViewPager

SwitchViewPager是可以自动滚动的ViewPager。

14. com.movitech.wxapi

WXPayEntryActivity

WXPayEntryActivity是微信的回调类，在微信执行完一定任务后，就会调用此类相关方法。

15. com.movitech.framework

MainApp

MainApp是整个项目使用的Application，其中初始化了一些操作，如果有需要也可以添加一些全局变量。

Need to do

1. Wechat Pay Support --> **David Wu** has finished this task.
2. Push Service Support --> **David Wu** has finished this task.
3. Local DataBase Support --> **David Wu** has finished this task.