**一：页面登录模块**

1. StartActivity为启动页面,用来显示App启动动画5秒。

①、在initData（）方法中，通过UserUtils工具类中的getIsFirstLoad()方法来从sharepreference中获取ISFIRST值，若是第一次安装登陆，则跳转至SplashActivity界面，该活动包含一个ViewPaper控件，用来展示介绍App的三张本地图片。若不是第一次登陆，通过StartRequest的start方法来请求广告页图片。

②StartRequest继承自ProjectRequest，ProjectRequest中包含RequestBuilder对象，通过startRequest的start方法对RequestBuilder对象的公共请求参数，请求地址，监听器参数赋值，ProjectRequest的构造方法来实现对RequestBuilder对象的请求方法、请求地址各个参数赋值，start方法中通过create()方法创建BaseRequest对象，通过其execute()方法来执行请求图片的操作。

③在onsuccess的回调方法中，通过Gson解析类型为StartBean返回的response数据，startbean对象中的checkversionbean对象如果不为空，则给该页面中的ImageView控件加载图片，若返回的错误码为401，则tokenLost为true，用户未登录。

④在广告5秒后或者点击跳过按钮，则会调用gotoLogin()方法，在该方法中，通过判断tokenLost值，若为true则用户未登录，跳转至LoginActivity，否则用户已登录，通过UserUtils工具类的getUser()方法来从sharepreference中获取字段为USER的用户信息，并通过Gson解析为UserBean对象，若该对象为空或者isTourist()，则userbean对象赋值为null,否则如果userbean对象的getClub()不为空则直接跳转至HomeUI主界面，否则跳转至LoginActivity界面。

2、LoginActivity为登录页面（监听器及内容如何传入的）

①对电话框et\_tel和验证码框et\_code进行监听，addTextChangedListener(new TextWatcher)，TextWatcher的afterTextChanged()中，将电话框中的×号按钮显示，其点击事件将编辑框清空；当电话框 长度为11并且验证码长度大于4时，并将登录框setEnable（true），并改变背景资源为黄色，否则为false，将其背景改为灰色，且不可按动。

②点击获取验证码后，当点击获取验证码时间和当前时间相差60秒时，

通过GetCodeRequest对象的getCode(tel)方法来获取验证码，否则弹出在1分钟后操作的弹框。

③GetCodeRequest继承自ProjectRequest类，其构造方法对ProjectRequest的url、监听器参数进行赋值，getCode方法传入params参数，GET请求，通过create方法创建RequestBuilder对象，通过该对象的execute方法来执行网络请求。在LoginAct0ivity的success回调方法中，当requestCode为CODE\_GETCODE时，Gson解析格式为BaseBean的response返回值，toast该basebean对象的getMessage()信息。

④点击登录按钮tv\_login，当输入的手机号长度为11且验证码不为空长度不超过6时，调用LoginRequest对象的login方法，传入手机号、验证码、设备ID来进行验证。该PushID通过UserUtils类的getPushToken方法从sharepreference中读取PUSHTOKEN。

LoginActivity通过LoginRequest接口来获取用户名、token、验证码，在LoginAcitiviy中进行验证登录，验证成功后跳转HomeUI

1. 获取验证码：

LoginActivity中创建GetCodeRequest对象，通过其getCode(tel)方法来获取验证码，GetCodeRequest对象继承自ProjectRequest,通过requestbuilder对象的setParams()setType()setUrl()等方法来设置信息，获取验证码，这些都是写好的。具体的实现在BaseRequest()的execute()方法中。

4、

**二：主页面模块**

1. HomeUI为主页

布局内含有FragmentLayout容器和自定义布局BottomBar。

HomeUI extends BaseUI,

①initView()

在initView()中，通过框架Fragmentation的loadMultipleRootFragment（）来加载主页，宗亲汇、我的三个fragment，通过BottomBar控件的addItem(new BottomBarTab())方法来添加底部导航栏控件，并通过setOnTabSelectedListener()注册监听器，通过onTableSelected()来监听按钮的切换，在onTabSelected()回调方法中，通过showHideFragment()方法来进行三个碎片界面的切换。当切换第二个消息界面时，通过UserUtils工具类的getUser方法从sharepreference中获取UserBean对象，通过isTourst()方法来判断是否为游客，当是游客时，通过MsgDialog的showLoginHint()方法弹出还未登陆的dialog对话框，并setOnCancelClickListener()建立监听，在onCancleClick()回调方法中，隐藏该dialog框，并将当前界面转至firstfragment。在onTabReselected()方法中：？？？

②initData()

首先通过getIntent()方法获取到从LoginActivity中传来的msg积分值，然后通过showIntegralHint(msg)方法来显示获取积分的dialog框。

通过IMUserP对象的login()方法来实现腾讯云的登陆，在imLoginSuccess()回调方法中，通过PushUtil的getInstance()方法实现初始化后台消息推送，在imLoginFail()的回调方法中会直接弹出MsgDialog框显示版本更新需要重新登录，点击后通过调用logout()方法跳转至LoginActivity界面。

通过StartRequest对象的start()方法来获取启动页信息，接口：URL\_START 在success()回调方法中，解析数据类型为StartBean的response字符串，通过startbean的getData()方法来获取CheckVersionBean对象，通过UserUtils工具类从sharepreference中获取本地的CheckVersionBean对象，通过StringUtils工具的compareVersion()方法来进行比较，若返回值为1，？？通过showUpdateDialog()方法来显示软件更新对话框，在该方法中调动downLoadApk()方法来下载，downloadApk()方法中，通过DownloadManagerUtils工具类的downloadApk()方法来下载，并注册广播监听器，当下载完成后，在DownloadReceiver中，同DownloadManagerUtils的installApk()方法来进行安装。

以下三个Fragment均是采用MVP模式，Model层完成具体的操作，Presenter调用model层，View层即Activity调用P层来获取数据后展现

1. FirstFragment为首页界面

页面布局包含一个RecycleView控件，其外层包裹一个SmartRefreshLayout布局，用来实现下来刷新和上拉加载的功能。

①initData()

调用initRecyclerView()方法，来初始化头部布局，包含轮播图、同宗活动、寄思先祖、企业秀、TextView同宗动态。

通过FirstP对象调用requestToBanner()方法请求轮播图，具体实现是FirstModel中的getBanner()方法来请求，接口：URL\_GETBANNER，请求在FirstP中回调方法bannerSuccess()内给bannerList赋值，同时调用IFirstFM接口中的showbanner()方法来展图片。showbanner()方法的具体实现是在FirstFragment中。

调用其requestToListRefresh()方法请求同宗动态数据。具体实现是在FirstModel中的getListRefresh()方法，接口：URL\_GETHOMEDYNAMIC，请求回调结果在FirstP中的回调方法listSuccess()中，通过loadMore()方法给dynamics同宗动态集合赋值，在FirstFragment里为适配器初始化时通过firstP的getListData()方法来得到这一集合。

②initRecycleView()

创建DynamicAdapter()同宗动态信息填充的适配器，item子布局为item\_dynamic，数据通过FirstP对象的 getListData()方法获取。

通过适配器的.addHeaderView()方法来加载头部视图，头部视图通过getHeadView()方法来加载，其调用initHeadView()方法，在该方法中并为其内的控件注册监听器。并设立setOnPicItemClickListener()点击事件，当点击图片的时候，调用openImag()方法来查看大图，该方法是通过跳转至ImageActivity活动来展示的。

firstFragment的页面布局为一个RecyclerView，通过addHeadView方法来加载头布局，头布局包含轮播图和同宗活动、寄思先祖、企业秀三个按钮，通过viewpaper来实现轮播图，当点击图片时，通过WebLoadActivity的webview控件来加载html文件

同宗活动点击事件也是通过WebLoadActivity来加载同宗活动html文件

同宗动态信息通过FirstP的getList方法来获取主页动态信息，通过DynamicAdapter适配器来进行信息填充。通过DynamicAdapter的setOnItemClickListener来监听子布局点击事件，跳转至DynamicInfoActivity消息详情界面，通过DynamicAdapter的setOnItemChildClickListener来监听子布局单个控件的点击事件，实现点赞，点击头像进入FriendDetailsUI好友详情页，点击图片进入ImageActivity图片查看页，点击视频进入VideoPlayActivity播放视频页。

1. secondFragment的页面布局内为FragmentLayout容器，通过loadRootFragment方法来加载根布局ViewPaperFragment，根布局也是一个容器，布局内通过TabLayout控件和viewpaper控件，使用IMPagerAdapter适配器的getItem方法来实现消息、好友、群聊三个fragment的切换。

消息界面ConversationFM信息的获取通过ConversationPresenter的getConversation方法来获取消息集合

好友列表界面FriendFM信息的获取通过FriendP的requestFriendtList方法来获取好友集合

群组列表界面GroupFM信息的获取通过GroupP的requestGroupList方法来获取群组集合

以上三个获取信息的方式都是通过MVP架构，通过Model层来具体实现网络请求方法，Presenter层来实现调用Model层的方法，在View层调用Presenter层的方法来加载数据。Model层均是通过OKGo网络请求框架，传入RequestAddress类内不同的url地址来获取相应的信息。

1. thirdFragment的页面通过LinearLayout垂直排列各个TextView控件，显示我的积分、我的动态等，头像部分通过自定义RoundedImageView来可以实现加载圆形头像，二维码部分的点击事件跳转至CodeUI，通过DESUtil的encryptDES方法来获取用户信息，通过bingoogolapple框架的syncEncodeQRCode方法来创建二维码图片

三、网络请求

1、采用OKGo框架

四、基础类及方法汇总

1、

五、工具类及其内部方法

1、

六、短信登录

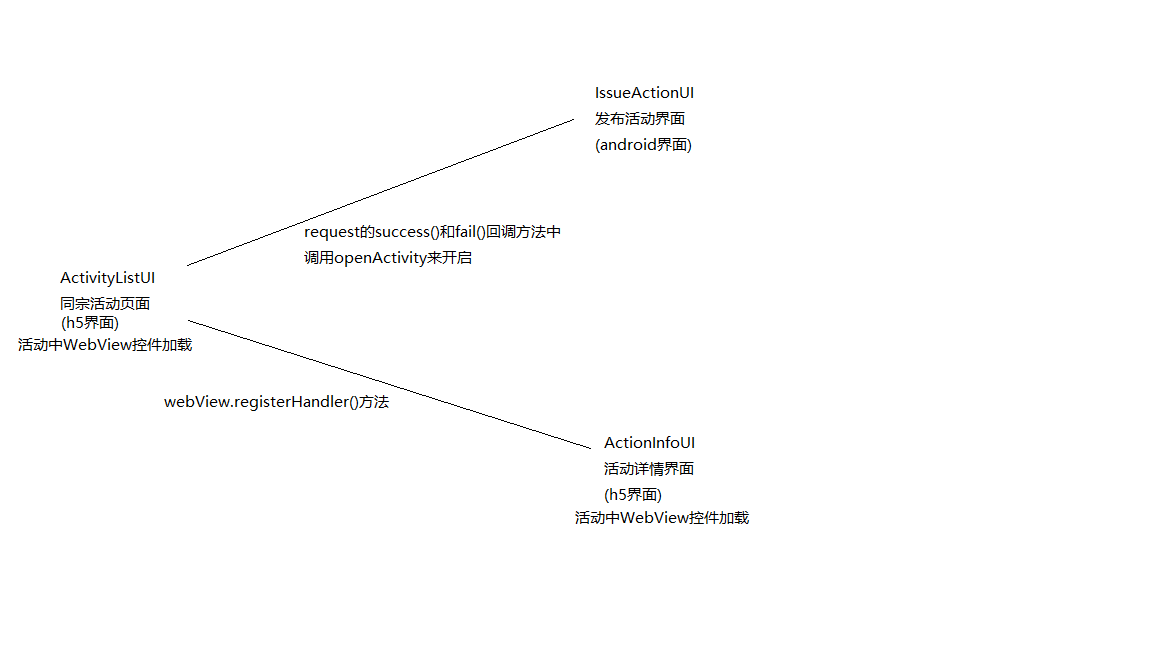
在LoginActivity中点击tv\_code按钮后通过创建的GetCodeRequest对象的getCode方法来获取验证码，获取后点击tv\_login按钮后通过创建的LoginRequest对象的login方法来进行用户名验证码的判断，在success回调方法里传回的response字符串Gson解析（LoginBean.class），若isIsOK为true且Status字段不等于303或304，即登录成功，跳转至HomeUI界面

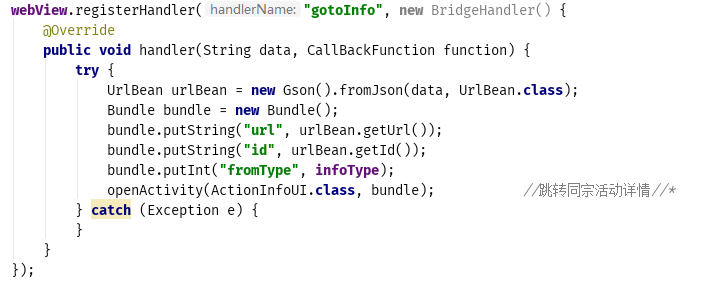
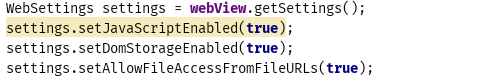
七、QQ和微信的初始化

1、在LoginActivity活动中需要初始化操作，因为可能会第三方登录

2、在分享时，在ShareDialog中调用ShareUtil工具类中的regToWX和reToQQ方法来进行微信和QQ终端的注册

八、上下拉刷新操作

八、同宗活动逻辑



将webview设置后，通过registerHandler方法写一个叫gotoInfo的方法，这样JavaScript就能通过调用gotoInfo方法，返回一个data，通过解析h5返回的信息，进行下一步操作

九、发布活动

十、二维码扫描技术

十一、发布动态

1、拍照或摄像功能

IssueDynamicActivity继承自ProjectBaseActivity，当点击拍摄rel\_camera按钮,会调用父类ProjectBaseActivity中的takePhoto()方法，在takePhoto()方法中，当通过PermissionUtil工具检查开启读写和拍照录音权限后，跳转至VideoRecordActivity进行拍照或摄像。

VideoRecordActivity的jcameraview控件的监听事件中，当captureSuccess(Bitmap)时，拍照成功，将bitmap通过FileUtil工具类的writeFile()写入手机，并setResult()将地址传出，在IssueDynamicActivity的onActivityResult中接受照片的地址；当recordSuccess时，接受视频的地址 。

在IssueDynamicActivity中，通过videoDispose(videoUrl)方法上传视频。该方法中，通过BitmapUtil的getVideoImg()方法来获取视频第一帧图片，获取失败则无法上传，成功则通过BitmaoUtil的saveBitmap方法将其存储，再通过File的getAbsolutePath()方法来获取其绝对路径，path里存储图片地址和视频地址，然后调用loadAccessoryRequest对象的load(paths)方法，进行上传…

布局文件：activity\_video\_secord.xml包含一个JCameraView即可

2、选择图片的操作

ImageActivity查看图片的操作

十二、

常用的BaseBean:

**一、登录时用户相关的**

1. LoginBean 验证短信验证码时的返回类型包含字段

**private boolean isError**;   
**private** UserBean **data**; 用户信息  
**private boolean isOk**; true  
**private** String **message**; “登录成功”  
**private** String **status**; “200”

1. UserBean为LoginBean中的data数据类型

包含字段：

**private** String **address**; “山西省，太原市，小店区”  
**private** String **headimg**;

**private** String **gender**;   
**private** AreadressBean **areadress**; ？啥地址

**private** String **token**; 登陆成功后会返回用户token值, app需要保存到本地, 以后请求用户相关信息都用token值获取

**private** String **realname**; 名 磊

**private int teamid**; 群组名？

**private** String **nickname**; 李磊全名  
**private** ClubBean **club**; 姓氏数据

**private** String **id**; ？

**private** String **job**; 工作  
**private** String **username**; 电话

**private** List<ThirdUsersBean> **thirdUsers**; 第三方登录  
------------------------------下面的接口文档未写？  
**private** String **usersig**;  
**private** String **openid**;  
**private int authtype**;  
**private int integral**;  
**private int currency**;  
**private int msg**;  
**private boolean isTourist**;

1. AreadressBean 为UserBean中的areadress数据类型

包含字段：

**private int areaid**; 140105地区id  
**private** String **created**; 创建时间  
**private** String **id**; ？  
**private int cityid**; 城市id  
**private** String **userid**; ？貌似和外层LoginBean中的id是一样的  
**private int provid**; 省id  
**private** String **updated**; 更新时间

1. ClubBean 为UserBean中的club数据类型

包含字段：

**private** String **club**; 姓 李  
**private** String **id**; ？  
**private** String **club\_qp**; ？  
**private** String **club\_sx**; ？  
**private** String **updated**; ？  
**private** String **created**; ？

1. ThirdUsersBean为LoginBean中的thirdUsers数据类型

包含字段：

**private** String **openid**;   
**private int usertype**; 2代表QQ，3代表微信

1. **HomeUI相关的**

1、StartBean 启动页获取数据的类型

包含字段：

**private boolean isError**; false  
**private** CheckVersionBean **data**; 版本信息相关  
**private boolean isOk**; true  
**private** String **message**; “启动成功”  
**private** String **status**; “200”

2、CheckVersionBean 为StartBean中data的数据类型

包含字段：

**private** String **appShowInfo**; 不同启动信息对应展示信息不同

**private int startFlag**; 0更新1不更新-1强制更新  
**private** String **startImgUrl**; 启动页图片

**private** Object **startImgGoUrl**; 启动页跳转路径

**private** String **newsid**; 消息ID

**private** String **downloadPath**; apk下载地址

**private int appType**; 1为IOS，0为Android

**private** String **created**;

**private** String **serverIp**;  
**private** String **modified**;

**private int id**;

**private** String **version**; ？？  
**private** String **resversion**; 资源版本号  
**private int resupdateFlag**; 资源是否需要更新0否1是  
**private int msg**; ？？积分