**注意事项：**

**1.项目编码格式统一采用UTF-8编码**

**2.代码中除注释外禁止出现中文**

**3.文件禁止出现警告提示 例如：**

**4.项目文件禁止出现警告提示 例如：**

**5.在项目中创建libs文件夹，统一放置引用的外部jar文件。引用第三方库的时候要谨慎，避免使用大容量的第三方库，而导致客户端非常大**

**6.在项目中，res文件夹中创建animation文件夹，统一放置动画xml文件**

**7.在项目中，res文件夹中创建drawable文件夹，统一放置selector.xml，shape.xml等配置文件**

**8.服务端可以实现的，就不要放在客户端来实现**

**9.strings.xml中使用%1$s实现字符串的通配**

**10.如果多个Activity中包含共同的UI处理，那么可以提炼一个CommonActivity，把通用部分交由它来处理，其他activity只要继承它即可**

**11.使用button+activitgroup实现tab效果时，使用Button.setSelected(true)，确保按钮处于选择状态，并使activitygroup的当前activity与该button对应**

**12.数据一定要效验，例如**

**字符型转数字型，如果转换失败一定要有缺省值；**

**服务端响应数据是否有效判断**

**13.图片的.9处理，图片要压缩**

**14.不要重用父类的handler，对应一个类的handler也不应该让其子类用到，否则会导致message.what冲突**

**15.资源文件只能以小写字母和下划线做首字母，随后的名字中只能出现 [a-z0-9\_.] ,否**

**则会导致R文件无法自动生成**

**16.onCreate中完成控件的初始化，onResume中完成数据的准备**

**17.每个activity都有自己独立的消息处理，其中包括正确的消息处理，错误处理，异常捕捉**

**18.任何主线程不需要做过于复杂的处理，并且主线程需要提供一个Handler给子线程去返回完成信息**

**19.每个项目开始前应进行分析和命名文档的编写**

**缩写：**

**bg----------background**

**txt----------textview**

**img----------imageview**

**btn----------button**

**txt----------textview**

**pic----------picture**

**lv----------listview**

**vp----------viewpager**

**gv----------gridview**

**注释：**

方法的注释：

/\*\*

\*注释

\*/

或

/\*\*注释\*/

配置文件的注释：

<!--注释-->

1.项目命名规则：

     项目功能或名称描述的英文组合或拼音组合（如果拼音过长，可采用首字母缩写组合）

2.项目包命名规则：

     项目应包含activity、adapter、data、model、utils六个模块

     目录结构：

/xunfeng

      /ruoshui

            /activity（包含启动画面Activity，主页面activity，其他页面相关activity）

            /adapter（适配器类）

            /database（有关数据库的操作）

            /model（实体封装类）

            /utils（工具和帮助类）

3.类命名规则：

①Activity

模块名+具有实际意义的英文单词或英文单词组合+Activity（每个单词首字母大写）

  例如：主页面的Activity  MainActivity

②Model

模块名+具有实际意义的英文单词或英文单词组合+Model（每个单词首字母大写）

  例如：UserInfoModel

③Adapter

模块名+具有实际意义的英文单词或英文单词组合+类型+Adapter（每个单词首字母大写）

  例如：UserInfoLVAdapter

④固定使用操作数据库的类 MyDatabaseHelper.java MyDatabaseOperation.java

⑤固定使用常量类和图片处理类和数据访问处理类 Constant.java DataManager.java ImageUtils.java

⑥检查网络连接类 CheckNetwork.java

4.变量命名规则：

①static final常量的命名：

  具有实际意义的引文单词，单词全部大写，多个英文单词，应以下划线\_分割开

②普通变量的命名：（全局变量必须用private进行修饰）

  采用具有标识变量而且具有实际意义的英文单词或英文单词的组合，单词首字母除第一个单词外全部大写。如：String userName=null；

5.方法命名规则：（方法如果有可能发生异常，应进行异常抛出，而且每个方法应进行javadoc注释说明）

具有实际意义的英文单词或单词组合，第一个单词首字母小写，其他英文单词首字母必须大写

例如：

/\*\*

\*注释

\*/

6.方法参数命名：

参数的命名规范和方法的命名规范相同，而且为了避免阅读程序时造成迷惑，请在尽量保证参数名称为一个单词的情况下使参数的命名尽可能明确。

7.图片资源命名：必须具有实际意义的英文单词或英文单词组合，如果图片为共享性图片资源，命名在前者的基础上加上share\_(命名应与代码<布局文件>中标识变量名称的英文单词<id>相一致）

8.Log使用：(系统名称 模块名称 接口名称，详细描述）

9.values中配置文件的命名规范：

  values应包含：colors,strings,styles,arrays

目录结构：

        /values

              /arrays（展示信息）

              /colors（颜色代码）

              /strings（信息描述）

              /styles（属性风格）

10.布局文件的命名规则：

   模块名+具有实际意义的英文单词或英文单词组合（每个英文单词用下划线\_分割）

   例如：main\_login.xml

11.布局文件中id的命名规则：

   控件属性+实际activity的英文标识单词+实际变量的英文表示单词 （用下划线\_隔开）

   例如：img\_main\_back