KP模板开发文档

编写人：齐圣

编写时间：2017.9.29

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本号** | **修改人** | **修改内容** | **日期** | **备注** |
| V1.0 | 齐圣 | 首次编写 | 2017.9.29 |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 引言

* 1. 编写目的

让开发人员能够加快对KP模板的理解，加快开发速度，开发进程，提升工作效率。

* 1. 背景

***项目背景：***

由李鹏辉提出并主导该项目的开发。针对我司卡瓦格博的产品KP所开发的KP模板，由于用户对KP的定制化需求较高，且考虑硬件功能及软件效率，我们决定开发出模板。之后的KP项目均基于此模板二次开发，加快开发效率，且一定程度提升性能，保证稳定性。

***开发环境：***

需要基于KP硬件的开发，固定的Android系统，现在系统版本为4.2.2，高度定制，作为KP内的唯一APP应用。

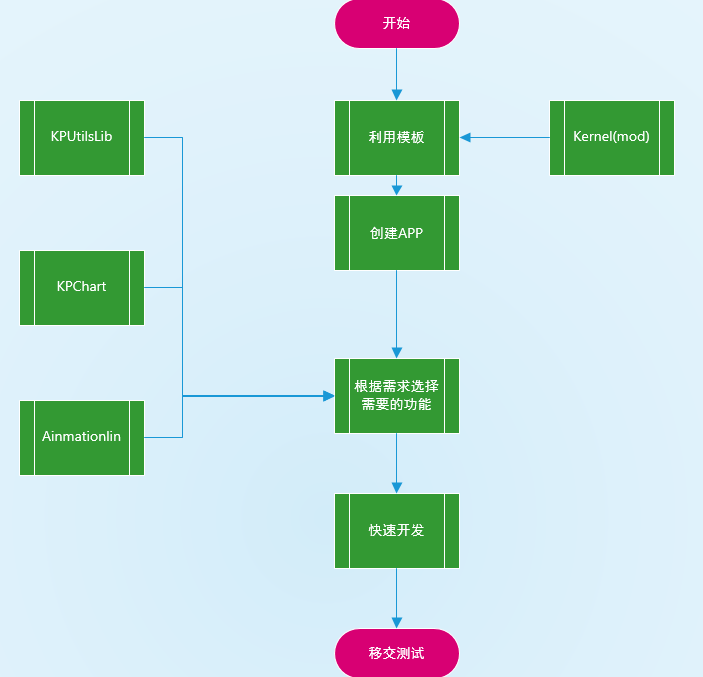
* 1. 参考资料

MPAndroidCharts: <https://github.com/PhilJay/MPAndroidChart>

Kotlin: http://kotlinlang.org/

# 模板的系统结构

快速开发流程图：



# 模板的用法

* 1. ***新建侧边栏：***
  2. ***新建子类fragment：***
  3. ***工具类库kputilslib':***

使用前先依赖: compile project(':kputilslib')

工具类库中有多个类

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 功能 | 使用方法 | 备注 |
| GuidesView | 操作引导 | 新建GuideView对象，对其Builder进行set需要的属性，最后调用.create,如果需要多个连接使用可以在click方法中调用hide与show |  |
| ClickAnimation | 点击特效 | 直接调用onScaleAnimation方法即可 |  |
| DateUtils | 时间戳工具类 | 直接调用对应方法可以获取具体的系统时间 |  |
| FileCacheUtil | 数据储存为文件 | 使用前先定义储存文件的名称，方便以后调用管理,可以储存任意数据，返回为json |  |
| NetUtils | 判断网络情况 | 可以直接判断出KP处于那种网络状态中，以便后续操作 |  |
| NoDoubleClickUtils | 防连点误触 | 直接调用，可以自行控制两次点击的时间 |  |
| NumberFormat | 数据转换 | 数据转换，只显示两位小数 |  |
| RegularUtils | 验证手机号邮箱 | 利用正则表达式验证手机号邮箱等，后续可以灵活添加修改 |  |
| SPUtils | SharedPreferences 的使用 | SharedPreferences 的使用,使用时可以直接调用put方法保存，get方法获取，remove移除对应key，clear清除所有。 |  |
| MyToast | Toast的工具类 | 封装为多种形式，分为normal，warning，info，success，error，custom，可以自定义ui显示，灵活性高 |  |

* 1. ***图表库kpchartlibrary'：***使用前先添加依赖

compile project(':kpchartlibrary')

图表库是依托于开源框架 MPandroidChart，git地址为：<https://github.com/PhilJay/MPAndroidChart>。我们基于MPandroidChart，将一些可以固定的属性进行简单的封装，调用时直接调用KPLineChart则可以展示折线图，具体属性太多不在这里赘述，可以在代码的注释里查看，目前只封装了折线图，后期依据所需情况添加。

* 1. ***动画库animationlib:***

使用前添加依赖compile project(': animationlib)

该动画库皆为基础动画，大致分为平移，晃动，渐变式，旋转，缩放等，具体内容参照下表（对应lib中的Techniques）

|  |  |
| --- | --- |
| 名称 | 简介 |
| SlideInLeft | 从左滑入中间位置 |
| SlideInRight | 从右滑入中间位置 |
| SlideInUp | 从下滑入 |
| SlideInDown | 从上滑入 |
| Flash | 闪动 |
| Pulse | 慢慢变大变小 |
| Wave | 波动 |
| Bounce | 上下晃动 |
| Shake | 左右晃动，类似撞墙 |
| FadeIn | 逐渐出现 |
| RotateIn | 旋转 向内 |

注：此处只是简单介绍大致效果，具体内容参照Techniques

# 4未解决的问题：

1. 流程动画的组合还需要想出更高效更可靠的组合办法。
2. 尚未兼容3D的快速配置。

# 5 lib使用方法详细介绍

## AnimationLib:

1：主体方法

使用时先创建Play.YoYoString，之后执行Play方法

private Play.YoYoString rope;

rope = Play.with(amin).duration(1200).playOn(view);

其中amin为播放的动画种类，view为对应的控件。

这是最基础的动画播放，还有一些其他方法：

|  |  |  |
| --- | --- | --- |
| 方法名 | 用途 | 参数 |
| With | 播放动画的类型 | 动画的类型 |
| Duration | 时间 | 毫秒 |
| pivot | 中心 | XY轴的值 |
| PivotX | X轴中心 | X轴的值 |
| pivotY | Y轴中心 | Y中的值 |
| repeat | 重播 | 时间 |
| withListener | 带监听，可以监听开始、完成、重播、取消状态 | 监听器 |
| playOn | 开始播放 | 无 |

使用本方法时，请加入判断rope是否为空，不为空时调用stop方法停止。

1. 封装的动画

|  |  |
| --- | --- |
| 名称 | 简介 |
| SlideInLeft | 从左滑入中间位置 |
| SlideInRight | 从右滑入中间位置 |
| SlideInUp | 从下滑入 |
| SlideInDown | 从上滑入 |
| Flash | 闪动 |
| Pulse | 慢慢变大变小 |
| Wave | 波动 |
| Bounce | 上下晃动 |
| Shake | 左右晃动，类似撞墙 |
| FadeIn | 逐渐出现 |
| RotateIn | 旋转 向内 |
| SlideOutLeft | 向左滑出 |
| SlideOutRight | 向右滑出 |
| SlideOutUp | 向上滑出 |
| SlideOutDown | 向下滑出 |
| RubberBand | 大小伸缩弹性 |
| Swing | 中心点不动左右晃，类似跷跷板 |
| Wobble | 左右晃动，幅度大 |
| Tada | tada 的效果 |
| RollIn | 从左滑动进 |
| RollOut | 从右滑动出 |
| FadeIn\up\down\left\right | 逐渐出现、上下左右 |
| FadeOut\in\up\down\right\left | 逐渐消失、上下左右 |
| RotateIn\down\_l&r\up\_l&r | 向内旋转出现 l&r意为左右 |
| RotateOut\down\_l&r\up\_l&r | 向外旋转出现 l&r意为左右 |
| ZoomIn\down\left\right\up | 缩放 |
| ZoomOut\down\left\right\up |  |

以上为已经封装在lib中的动画，可以依据需要增加动画。

## KPChart：

1:KPLineChart:

使用时直接新建KPLineChart对象，对该对象的属性进行设置就可以正常展示、使用了。

这是kp折线图，为了方便使用，封装了如下

|  |  |  |
| --- | --- | --- |
| 方法名 | 功能 | 默认为 |
| setNoDataText | 无数据时显示的文字 | 暂时尚无数据 |
| setTouchEnabled | 能否点击 | true |
| setDragEnabled | 可以拖拽 | true |
| setScaleEnabled | 可以滑动 | True |
| setDrawGridBackground | 背景 | False |
| setPinchZoom | 缩放 | True |

由于该控件的UI定制化极高，所以默认属性较少，下面介绍一下常用方法：

// 图表的注解(只有当数据集存在时候才生效)

Legend l = mBinding.charts1.getLegend();

// 设置图表的背景颜色 mBinding.charts1.setBackgroundColor();

// 数据显示的颜色

data.setValueTextColor(Color.BLACK);

// 可以修改图表注解部分的位置

l.setPosition(Legend.LegendPosition.LEFT\_OF\_CHART);

// 线性，也可是圆

l.setForm(Legend.LegendForm.LINE);

// 颜色

l.setTextColor(Color.WHITE);

// x坐标轴

XAxis xl = mBinding.charts1.getXAxis();

// x轴字体颜色

xl.setTextColor(Color.parseColor(color\_text));

//是否显示x轴背景线

xl.setDrawGridLines(true);

//x轴背景线的颜色

xl.setGridColor(Color.parseColor(color\_grid));

xl.setAvoidFirstLastClipping(true);

// x轴的数量

xl.setLabelCount(5);

// 如果false，那么x坐标轴将不可见

xl.setEnabled(true);

// 将X坐标轴放置在底部，默认是在顶部。

xl.setPosition(XAxis.XAxisPosition.BOTTOM);

// 图表左边的y坐标轴线

YAxis leftAxis = mBinding.charts1.getAxisLeft();

// Y轴字体颜色

leftAxis.setTextColor(Color.parseColor(color\_text));

// y轴数量

leftAxis.setLabelCount(5);

// 最大值

leftAxis.setAxisMaxValue(90f);

// 最小值

leftAxis.setAxisMinValue(40f);

// 不一定要从0开始

leftAxis.setStartAtZero(true);

//是否显示Y轴的背景线

leftAxis.setDrawGridLines(true);

//Y轴背景线的颜色

leftAxis.setGridColor(Color.parseColor(color\_grid));

// 不显示图表的右边y坐标轴线

rightAxis.setEnabled(false);

还有更多的方法参照：<https://github.com/PhilJay/MPAndroidChart>