

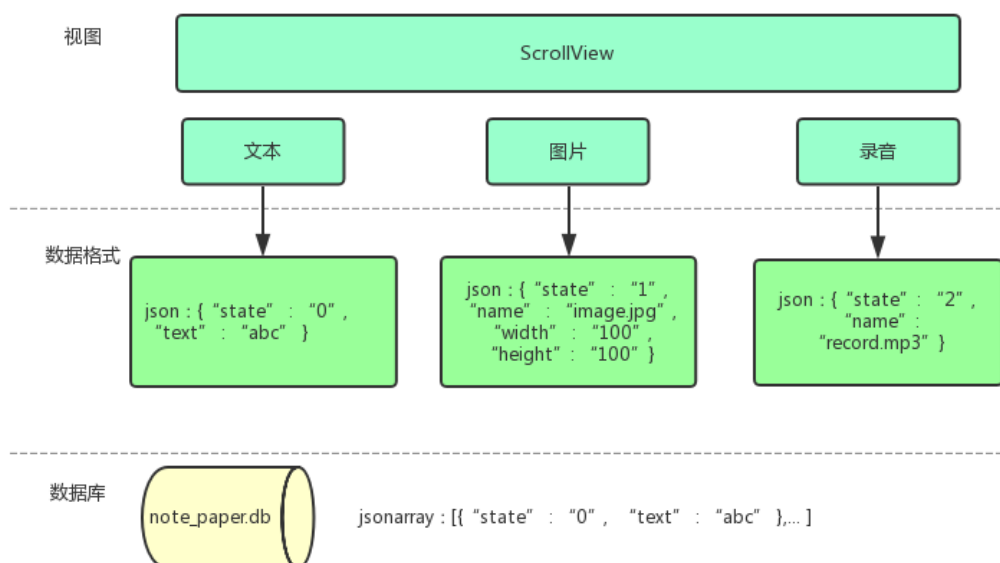
便签设计文档

一、项目结构

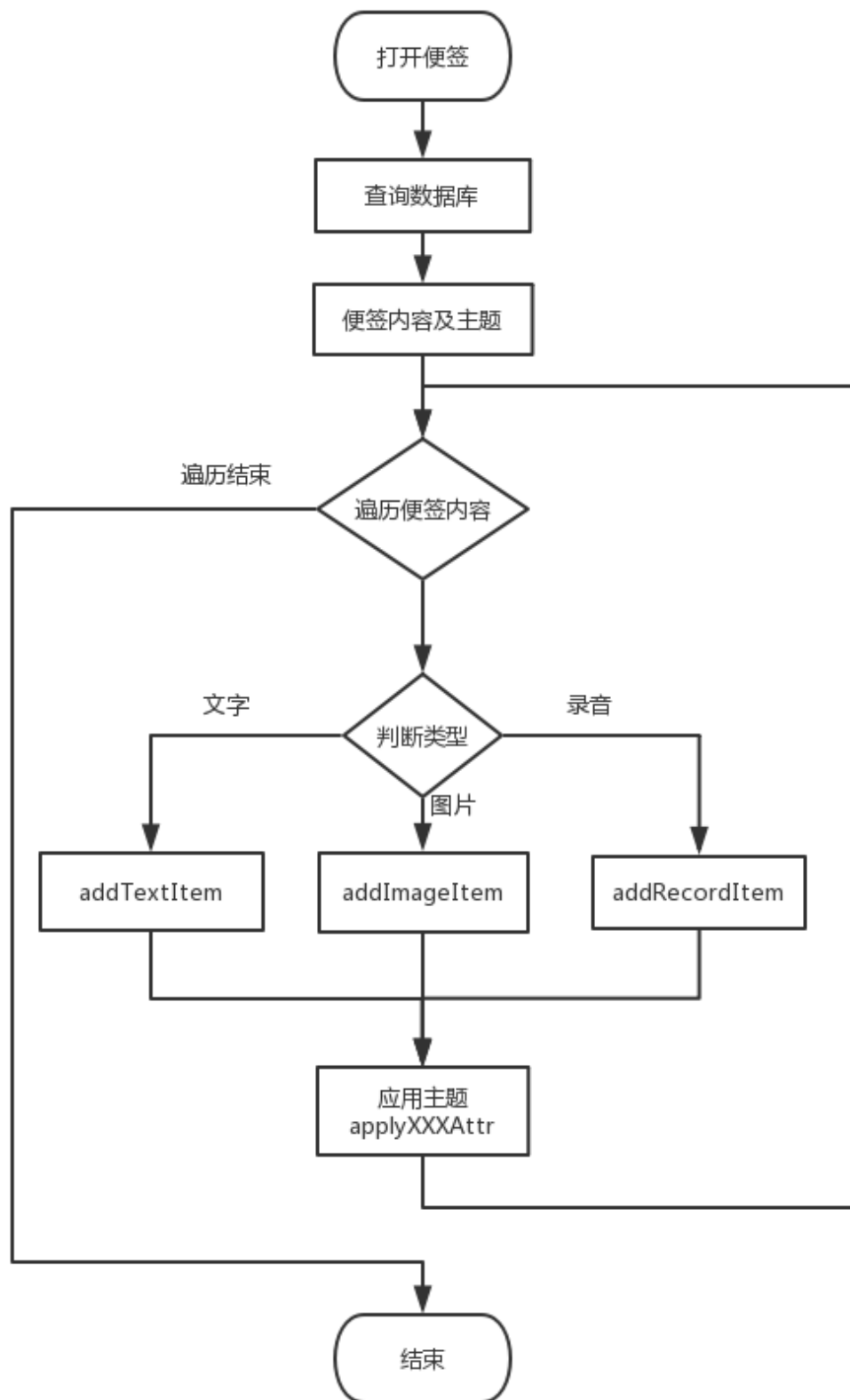
1. 模块划分：NotePaper、DownloadCommon（文件下载）、EditCommon（涂鸦）
2. NotePaper 目录结构：/app（Activity 存放目录）、/database（数据库）、/restore（撤销还原）、/template（主题模板）、/widget（view 定制）、/photoviewer（图片预览）、/flavors/publish/accountsync（账号与同步）

二、编辑界面

1. 界面组成：ScrollView 作为父容器，添加子 View：文本（TextView）、图片（ImageView）、录音（自定义 View）
2. 数据保存：遍历 ScrollView，按顺序保存文本的内容、图片的文件路径、录音的文件路径
 - 2.1 文本 json 保存格式：{"state": "0", "text": "abc" }
 - 2.2 图片 json 保存格式：{"state": "1", "name": "image.jpg", "width": "100", "height": "100"}
 - 2.3 录音 json 保存格式：{"state": "2", "name": "record.mp3"}
 - 2.4 以上数据按顺序组成 jsonarray 保存到 db



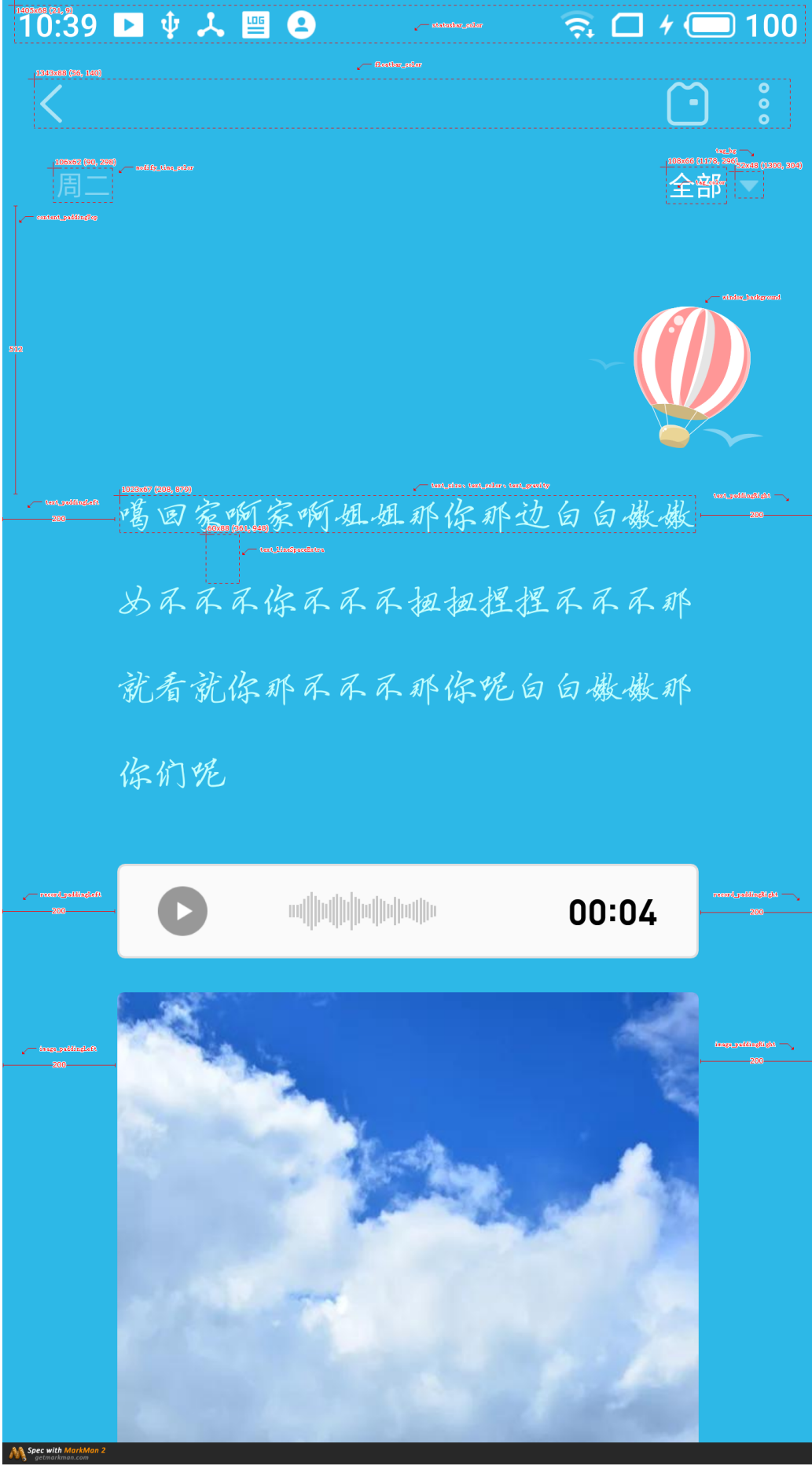
3. 打开已保存便签流程



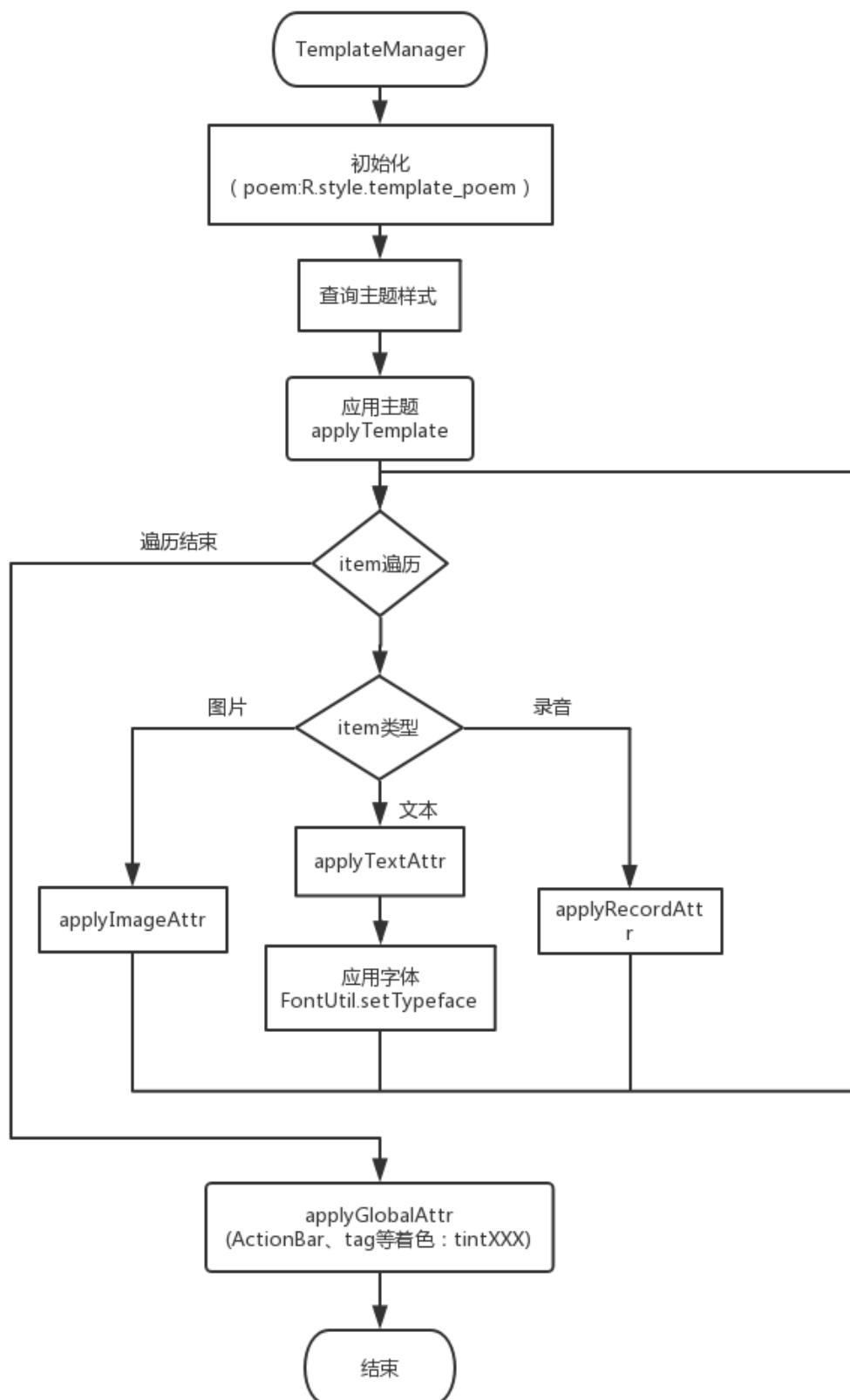
4. 主题设计：以主题名称作为 key，数据库中便签的主题字段保存该 key
- 4.1 参考系统主题在 style 中定义主题样式，在 template.xml 定义主题详情：

```
<declare-styleable name="template">
    <attr name="statusbar_color">
        <enum name="white" value="1" />
        <enum name="black" value="0" />
    </attr>
    <attr name="window_background" format="color|reference" />
    <attr name="actionbar_color" format="color|reference" />
    <attr name="tag_color" format="color|reference" />
    <attr name="tag_bg" format="color|reference" />
    <attr name="remind_color" format="color|reference" />
    <attr name="modify_time_color" format="color|reference" />
    <attr name="floatbar_color" format="color|reference" />
    <attr name="content_background" format="color|reference" />
    <attr name="content_paddingLeft" format="dimension" />
    <attr name="content_paddingRight" format="dimension" />
    <attr name="content_paddingTop" format="dimension" />
    <attr name="content_paddingBottom" format="dimension" />
    <attr name="text_paddingLeft" format="dimension" />
    <attr name="text_paddingRight" format="dimension" />
    <attr name="record_paddingLeft" format="dimension" />
    <attr name="record_paddingRight" format="dimension" />
    <attr name="image_paddingLeft" format="dimension" />
    <attr name="image_paddingRight" format="dimension" />
    <attr name="text_size" format="dimension" />
    <attr name="text_lineSpaceExtra" format="dimension" />
    <attr name="text_color" format="color|reference" />
    <attr name="text_gravity">
        <enum name="left" value="0" />
        <enum name="center" value="1" />
    </attr>
    <attr name="text_paraSpaceExtra" format="dimension" />
    <attr name="record_mode">
        <enum name="normal" value="0" />
        <enum name="simple" value="1" />
    </attr>
    <attr name="share_layout" format="reference" />
</declare-styleable>
```

详见标注（原文件：主题标注.png）



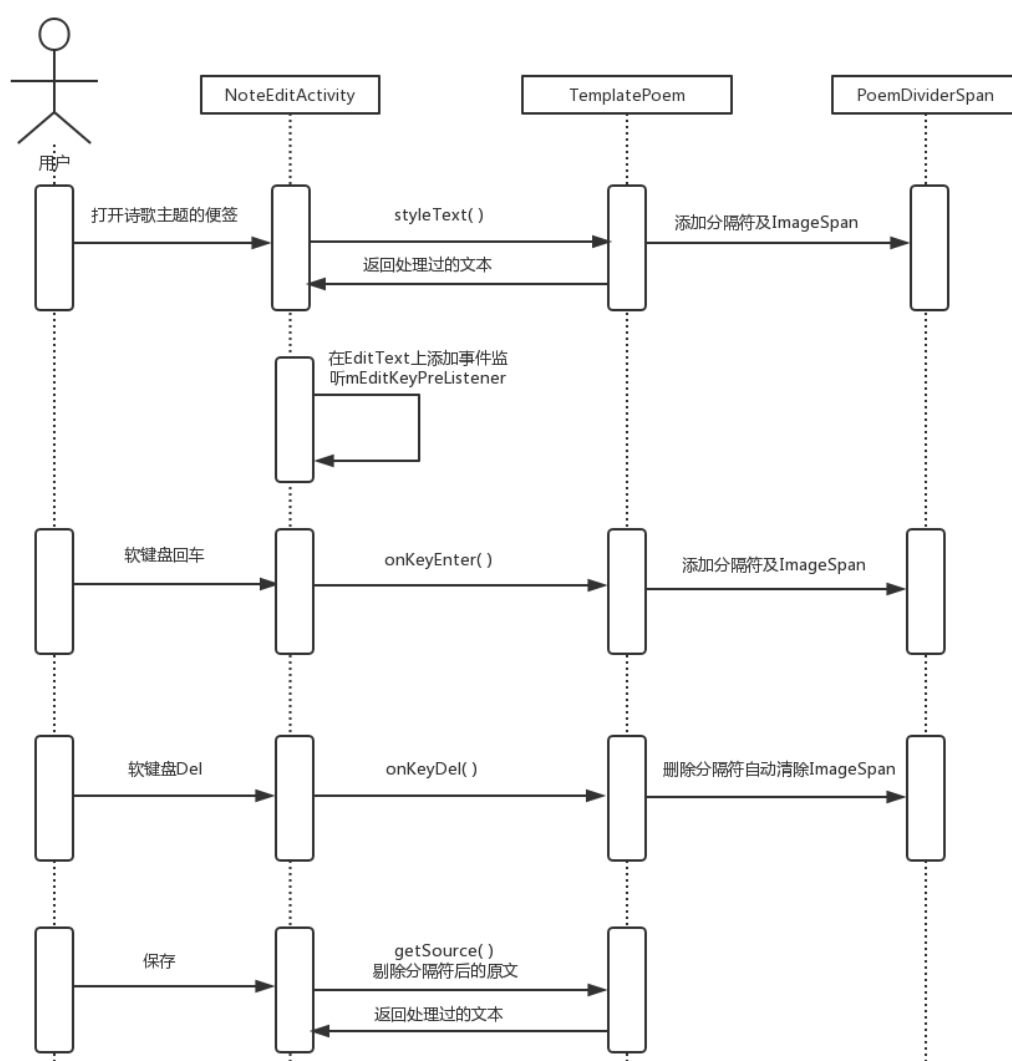
4.2 主题解析流程



4.3 主题字体：在线下载字体文件，存放在本地

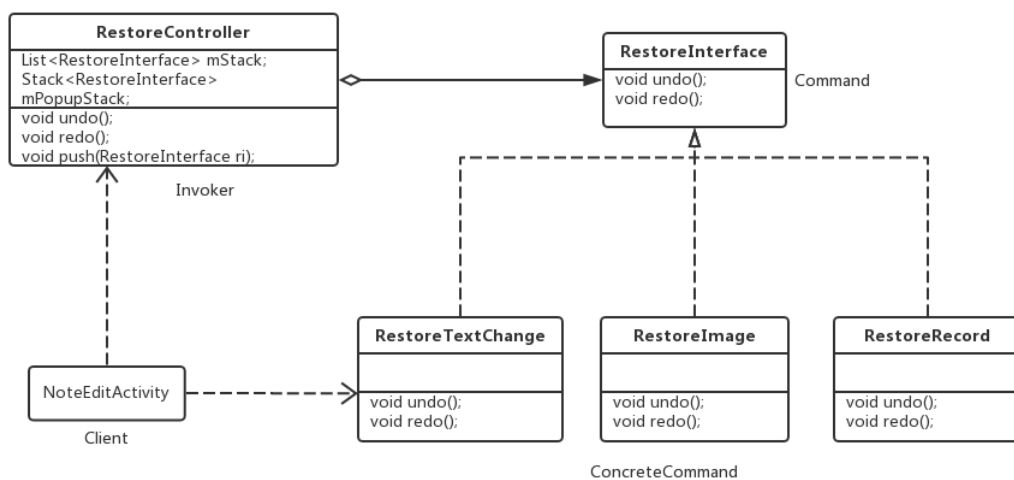
(/sdcard/Android/data/com.meizu.notepaper/files/.downloads)，调用系统 API: Typeface.createFromFile 加载字体文件返回 typeface，然后通过 TextView.setTypeFace 使用。其中 typeface 会占用比较大的 native heap，所以需要缓存使用

4.4 诗歌、樱花主题定制换行分隔符：1. 特殊分隔符是通过 ImageSpan 来展示的，定制 ImageSpan 即 PoemDividerSpan 重写 draw() 方法，在分隔符上输入内容后隐藏分隔符的显示 2. 保存数据时剔除文本中的特殊分隔符，得到的原始数据兼容其他主题。展示时预处理原始数据，在换行处添加特殊分隔符

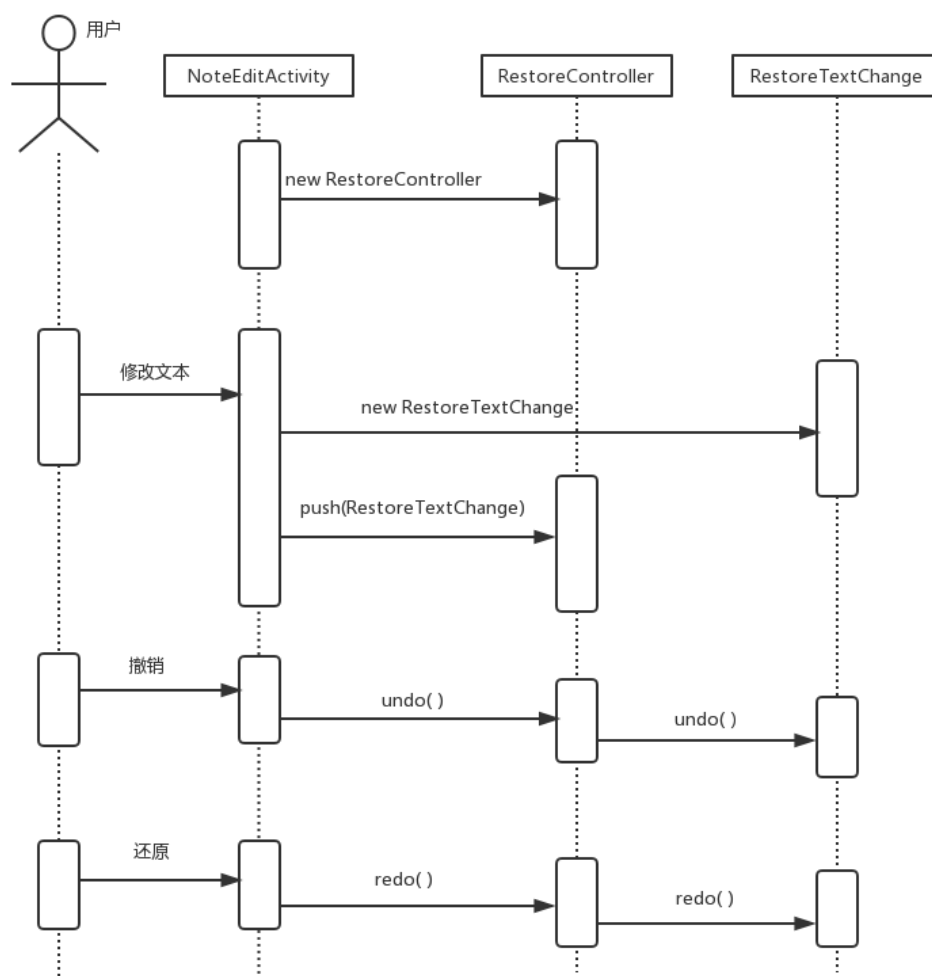


5. 撤销还原

5.1 设计上类似命令模式

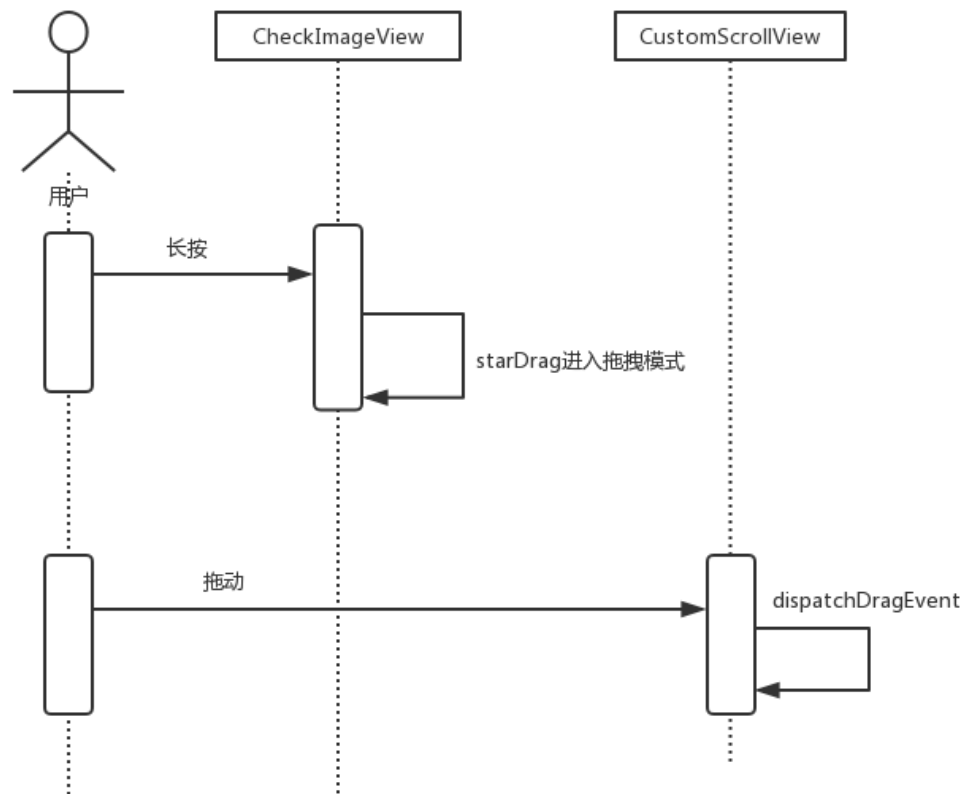


5.2 **RestoreController** 是撤销还原功能的核心，负责记录用户的操作（文字、图片、录音的修改）和撤销还原的调度



6. 定制 ScrollView 即 CustomScrollView 中实现清单拖拽、标题栏隐藏和阻尼运动、回收站便签不可编辑等功能

6.1 清单拖拽：长按清单按钮后，通过 `View.startDrag` 进入拖拽模式，然后在 `CustomScrollView` 中响应拖拽事件



6.2 标题栏隐藏和阻尼运动：在 `CustomScrollView` 的 `onLayout` 中动态计算出标题栏高度，然后滚动相应的距离来隐藏标题。重写 `onTouchEvent` 模拟阻尼运动，这里我们的实现比较简单，主要是将 `move` 事件产生的偏移进行缩小，例如实际偏移量原本是 `deltaY`，但最后只滚动 `deltaY / 2`

6.3 回收站便签不可编辑：重写 `onInterceptTouchEvent`，如果便签位于回收站则截获事件

7. 富文本功能详见同目录下的设计文档《富文本.md》