

浙江工业大学

移动应用开发实验：轨迹球实验

2023/2024(2)



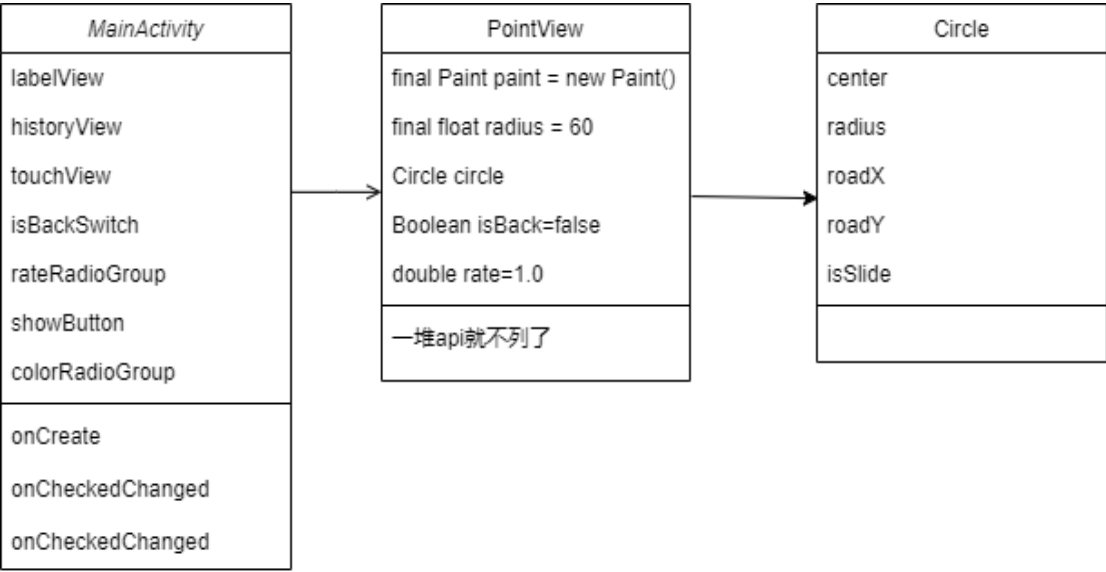
学生姓名	李飞飞
学生学号	202105710309
学生班级	软工 2206
任课教师	邱杰凡
实验时间	2024.4.7

轨迹球实验

一、 列出主要用到的组件和控件：

1. MainActivity (Activity 组件)
 - labelView TextView 控件
事件参数区
 - historyView TextView 控件
移动历史
 - touchView PointView 控件
小球活动面板
 - isBackSwitch Switch 控件
是否为反向回放
 - rateRadioGroup RadioGroup 控件
小球速率控制器
 - showButton Button 控件
启动回放
 - colorRadioGroup RadioGroup 控件
小球颜色控制器
2. LoginActivity (Activity 组件)
 - editTextUsername EditText 控件
用于输入账号
 - editTextPassword EditText 控件
用于输入密码
 - buttonLogin Button 控件
用于确认登录
3. PointView 组件 extends View
小球组件，用于实现小球动画效果
4. Circle
工具类，用于存放小球信息

二、 请写出为了实现主要逻辑， 需要使用的控件之间的逻辑。



图表 1 类图

MainActivity 为主要页面，展示所有组件，其中 labelView, historyView 分别展示当前运动参数与运动历史，touchView 组件用于控制小球，其余为控制小球参数的组件，每个具体对应功能见一

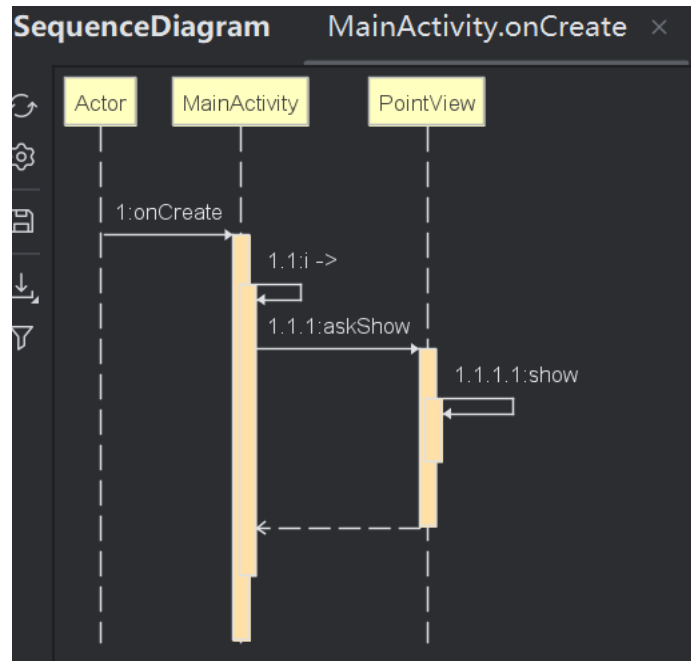
LoginActivity 为登录页面，当账号为”LiFeifei”密码 “202105710309” 时可以跳转到 MainActivity

PointView 为小球类，提供以下接口：

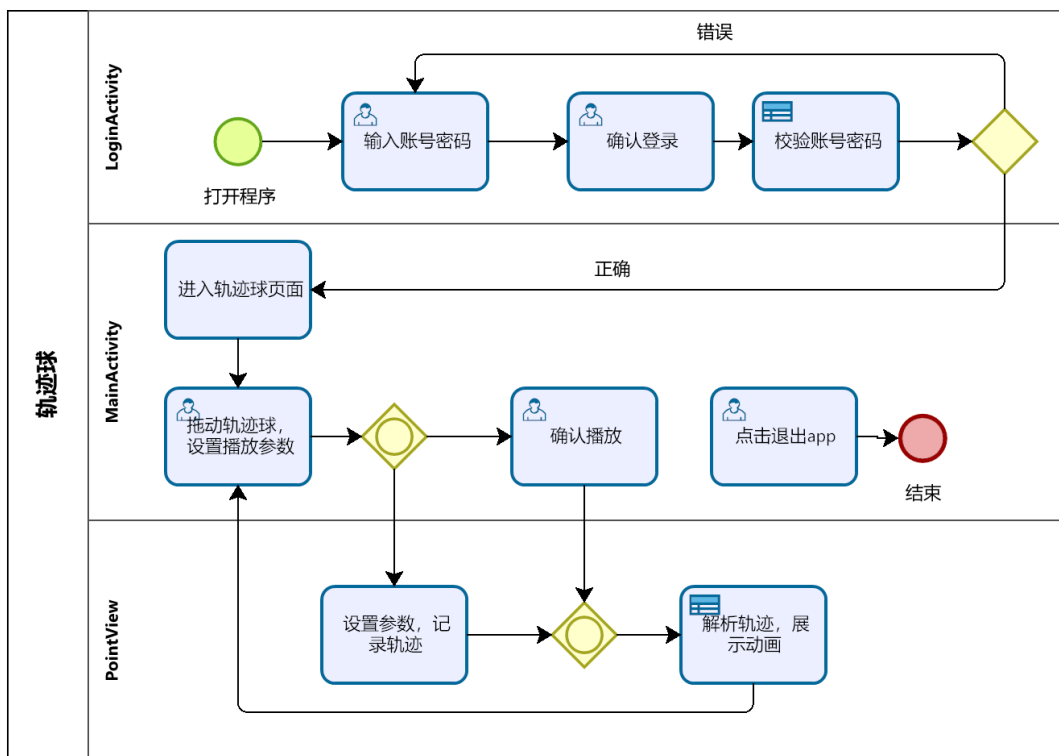
- askShow 开始播放回放
- updateIsBack 修改播放回放方式（正序，倒叙）
- setRate 设置回放速率（0~+inf）
- resetColor 设置小球颜色，将在下次移动时生效

在 MainActivity 的其它组件中操纵这些接口以达到不同的展示效果

Circle 类是封装的小球对象有半径，坐标等信息



图表 2 小球动画展示时序图



Powered by
bizagi
Modeler

图表 3 操作流程图

三、 简要说明如何实现移动模块（显示名字和学号）的绘制？简要描述如何实现回放，例如说明如何实现过触摸程数据记录，如何实现对回放速度的控制？

```
13 个用法
5     private Circle touchCircle;
6     @SuppressWarnings("ClickableViewAccessibility")
7     @Override
8     public boolean onTouchEvent(MotionEvent event) {
9         float slideX, slideY;
10        switch (event.getAction()) {
11            case MotionEvent.ACTION_DOWN:
12                if (isOnPoint(circle, event.getX(), event.getY())) {
13                    touchCircle = circle;
14                    touchCircle.roadX.remove(index: 0);
15                    touchCircle.roadY.remove(index: 0);
16                }
17                break;
18            case MotionEvent.ACTION_MOVE:
19                if (touchCircle != null && !touchCircle.isSlide) {
20                    slideX = event.getX();
21                    slideY = event.getY();
22                    touchCircle.roadX.add(slideX);
23                    touchCircle.roadY.add(slideY);
24                    touchCircle.center[0] = slideX;
25                    touchCircle.center[1] = slideY;
26                    invalidate();
27                }
28                break;
29            case MotionEvent.ACTION_UP:
30                if (touchCircle != null && !touchCircle.isSlide) {
31                    // show();
32                }
33                break;
34        }
35        return true;
36    }
```

图表 4 小球捕获操作

PointView 会捕获鼠标操作事件，在鼠标按下时开始记录，鼠标移动时保存记录移动信息，当点击“开始播放”时会解析移动信息并开辟一个线程进行动画播放。由于要支持延时展示所以必须时存储信息并开辟线程展示，因此倒放与速率就非常好实现了。倒放只需要把存的信息反转以下就行了，倍速只要让线程 sleep 的时间除对应倍速即可。其它诸如移动信息的展示、颜色切换就更简单了。

下图是实动画播放的代码,已经体现了倒放与倍速

```
23     private void show() {
24         new Thread(new Runnable() {
25             @Override
26             public void run() {
27                 circle.isSlide = true;
28                 if(isBack){
29                     Collections.reverse(circle.roadX);
30                     Collections.reverse(circle.roadY);
31                 }
32                 for (int i=0;i<circle.roadX.size();i++) {
33                     circle.center[0] = circle.roadX.get(i);
34                     circle.center[1] = circle.roadY.get(i);
35                     try {
36                         Thread.sleep((long) (10/rate));
37                         handler.sendEmptyMessage(what: 0);
38                     } catch (InterruptedException e) {
39                         e.printStackTrace();
40                     }
41                 }
42                 float startX,startY;
43                 if(isBack){
44                     startX=circle.roadX.get(circle.roadX.size()-1);
45                     startY=circle.roadY.get(circle.roadX.size()-1);
46                 }
47                 else {
48                     startX = circle.roadX.get(0);
49                     startY = circle.roadY.get(0);
50                 }
51                 circle.roadX.clear();
52                 circle.roadY.clear();
53                 circle.roadX.add(startX);
54                 circle.roadY.add(startY);
55                 circle.isSlide = false;
56             }
57         }).start();
58     }
59
60     1个用法
61     @SuppressWarnings("HandlerLeak")
62     private Handler handler = new Handler() {
63         if (msg.what == 0) {
64             invalidate();
65         }
66     };
67     >
```

倒放

倍速

图表 5 动画播放