

OPPO Watch App 开发指南

OPPO Watch 操作系统是基于 Android 8.1.0 深度定制的智能手表操作系统。开发者可基于 Android 系统，结合 OPPO 提供的系统 UI 规范、控件库、SDK 等内容进行定制化开发。

本次开发包括但不限于以下分类：运动健康、效率（例如日程管理、笔记、存储、办公、邮箱等）、生活（例如时间、天气、快递、美食、购物等）、内容（例如教育学习、娱乐、资讯、报刊杂志等）、出行（例如导航、交通、旅游等）、社交（例如聊天、社区等）、财务（例如记账、投资理财、支付等）、休闲小游戏。

1. 开发环境准备

1.1 硬件环境准备

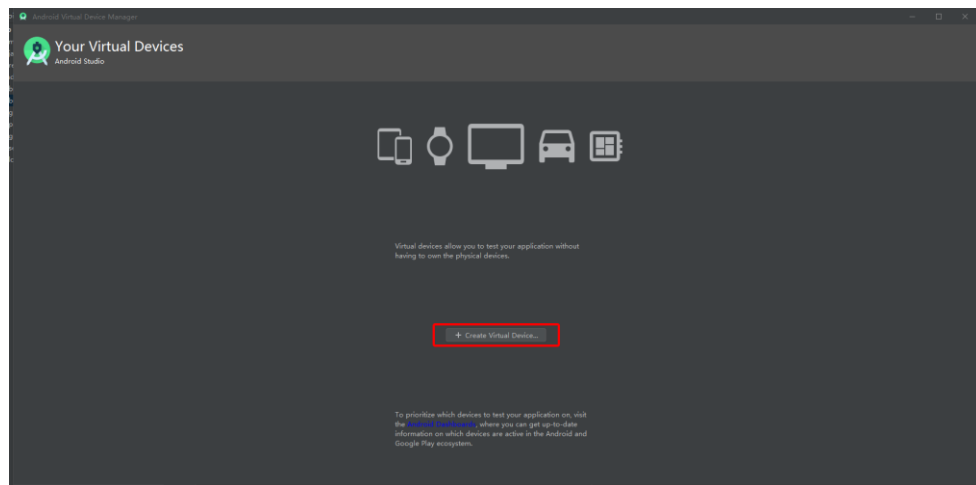
- 1) 纯 android 应用程序，只需要准备一台 PC 即可，可使用虚拟机
- 2) 依赖于手表特性 api 的应用程序，如心率接口等运动接口，则需要准备 oppo 手表，无法使用虚拟机
- 3) 依赖于手表与手机互联的应用程序，则需要多准备一部 OPPO 手机

1.2 软件环境准备

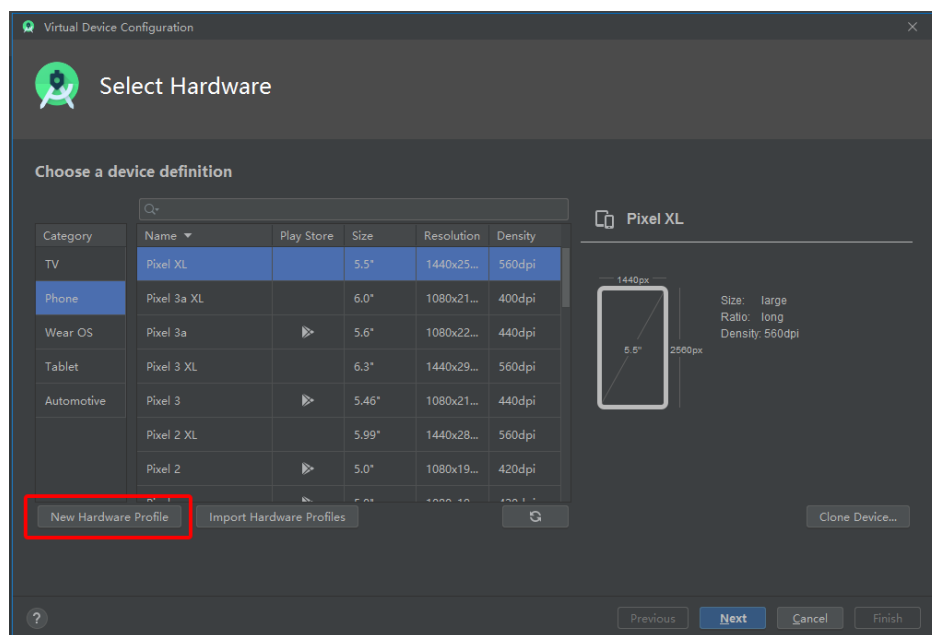
- 1) 在 PC 上安装 Android Studio version 3.0 或以上版本，并安装 Android SDK Platform version 8.1.0 (API level 27)。
- 2) 下载 OPPO 手表公共组件
- 3) 下载 OPPO 手表互联 SDK（仅支持手表与手机互联的应用程序）

2. 虚拟机创建

1) 在 Android Studio 选择创建虚拟机

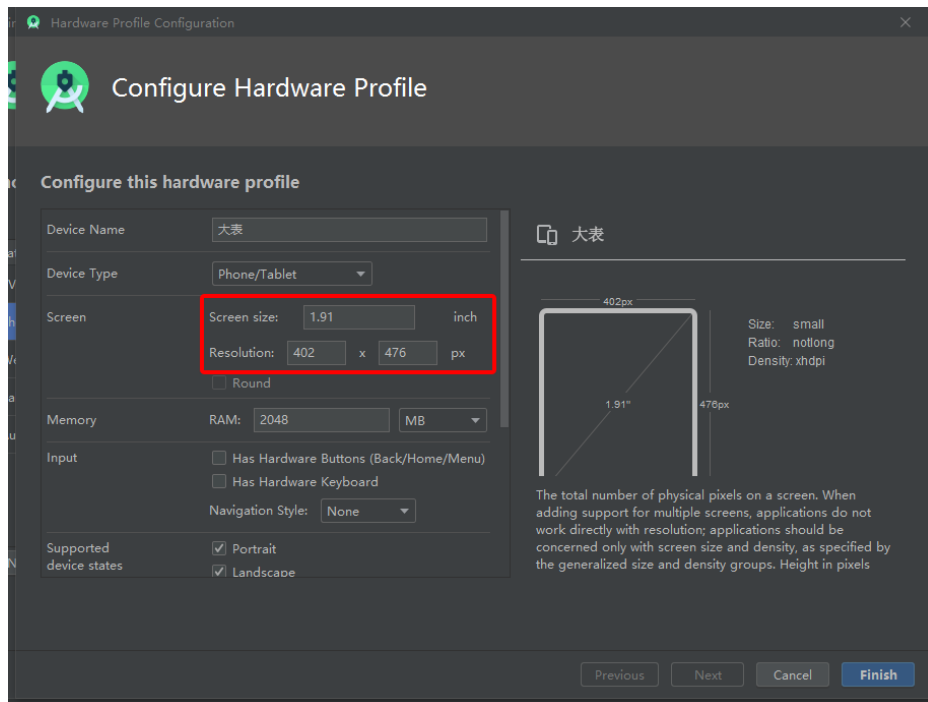


2) 创建一个新的硬件配置

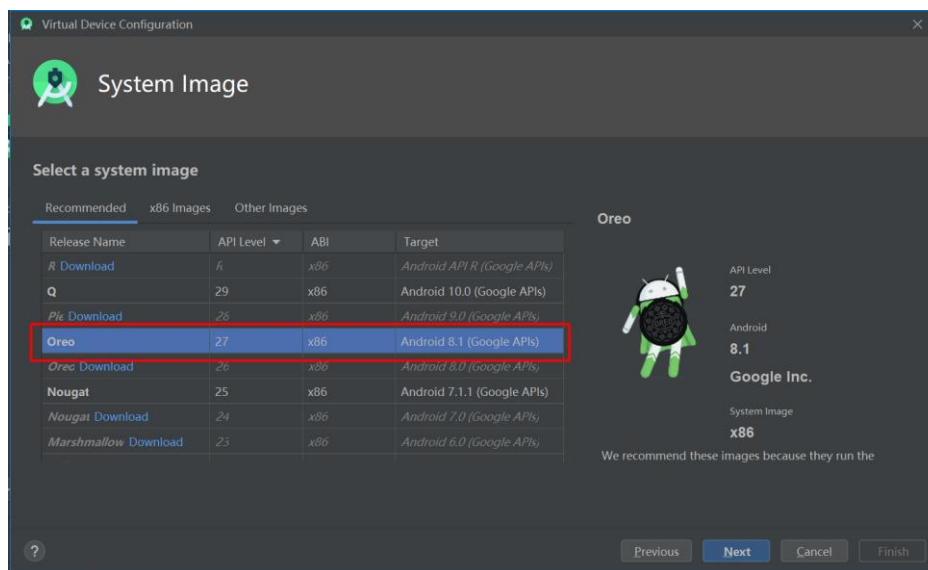


3) 填写手表的屏幕参数

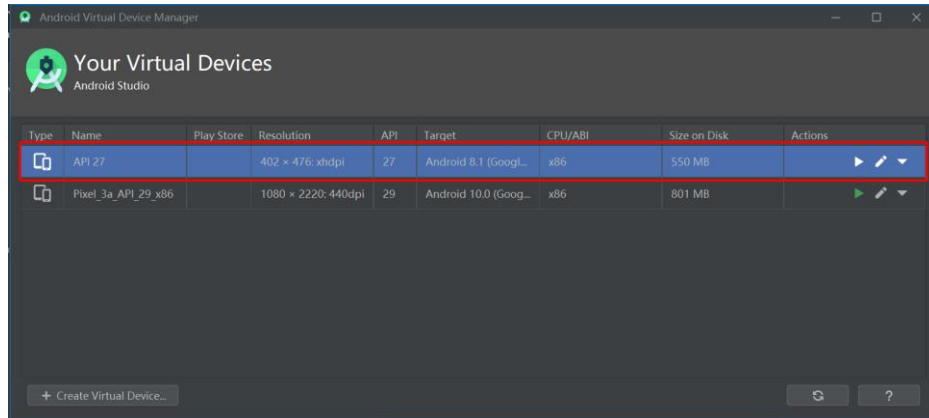
大表 1.91inch , 402x476 , 小表 1.6inch , 320x360



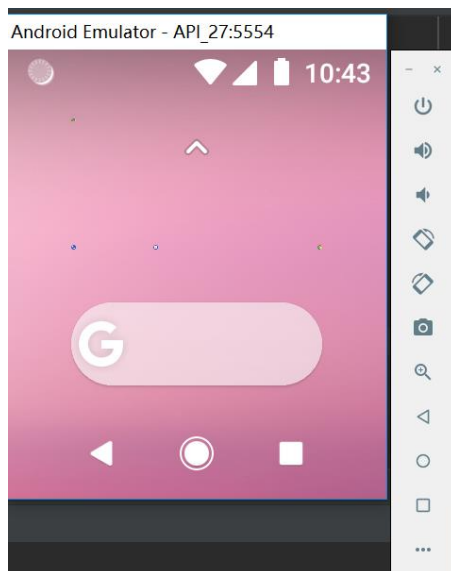
4) 选择对应的 android 版本，Android 8.1.0



5) 完成创建



6) 运行虚拟机



3.应用开发常见问题

3.1 Manifest 修改

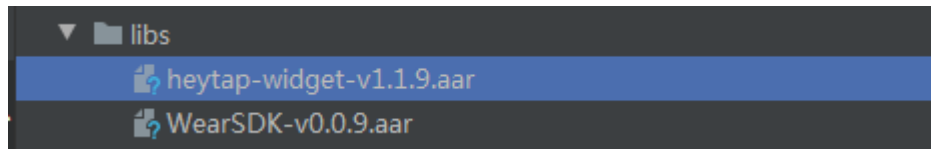
修改应用的 Manifest 文件，以表明应用为手表应用，在 Manifest 上添加

```
<uses-feature android:name="android.hardware.type.watch"/>
```

3.2 oppo 手表公共控件导入

引入公共控件库，使应用程序的设风格与手表保持一致，如果需要和手机互联，还需要引入

手表互联 SDK，目录如下：



在 gradle 文件添加

```
dependencies {  
    implementation fileTree(includes: ['*.aar'], dir: 'libs')  
}
```

3.3 启动右滑功能

手表限制于屏幕过小的原因,无法很好地支持物理键返回或手势返回,因此,实现右滑返回,能提高手表应用的交互体验,启用右滑功能可以通过以下两种方式实现:

1) 调用函数方式(Activity 中):

`getWindow().requestFeature(Window.FEATURE_SWIPE_TO_DISMISS);`, 必须在 `setContentView()`之前进行设置

2) 通过主题配置,在主题样式中加入:

```
<item name="android:windowSwipeToDismiss">true</item>
```

3.4 主题修改

由于使用了右滑返回功能,滑动时会显示低层的 activity,如果低层的 activity 状态不正确,处于 `onStop` 状态,则会导致右滑时出现界面重绘、闪烁的问题,因此,我们还要使用正确的主题,使底层的 activity 处于 `onPause` 状态,不会出现重绘问题,实现此功能的主题有两个:

1) 使用 `DeviceDefault` 主题

2) 使用 `AppCompat` 主题

3.5 关于 oppo 手表联网说明

oppo 手表除了支持 wifi 和移动网络联网之外，还支持蓝牙联网，即通过蓝牙与手机连接，使用手机的网络功能，但限制于蓝牙的传输性能，蓝牙联网的速度与稳定性也受到限制，如果速度只能维持在 80~120K 左右，对于强网络类型应用，需要在蓝牙网络下做优化，如何判断当前是否使用蓝牙网络，可使用下面方法

1) 通过设置获取

```
Settings.Global.getInt(ContextProxy.get().getContentResolver(), "bluetooth_net_proxy_on", def: 0) == 1
```

2) 通过 oppo 互联 SDK 获取，具体查看相关 SDK。