

Android企业级 项目实战教程

黑马程序员 / 编著



添加QQ或微信号208695827，获取教学答案、源码，抢“助学金红包”。

本书涵盖多个技术热点，并且提供四个小项目，以及一个“黑马头条”企业级项目。

提供免费教学资源，包括案例源代码、长达25小时的教学视频等。

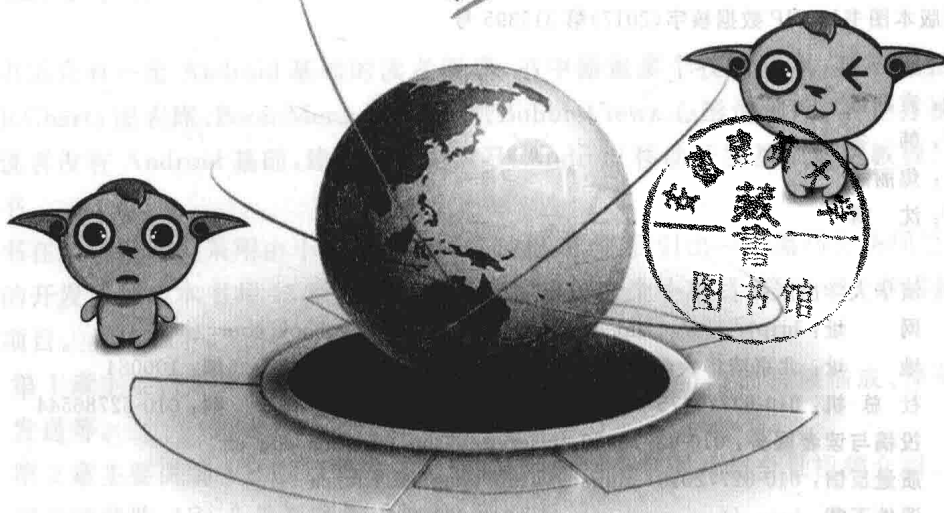


清华大学出版社

Android企业级 项目实战教程

黑马程序员 / 编著

有问题，就找黑马程序员问答精灵！



清华大学出版社
北 京

内 容 简 介

本书适合有一定 Android 基础的读者阅读,书中涵盖多个技术热点,其中包括下拉刷新、HelloCharts 图表库、BoomMenu 圆形菜单、BubbleViews 心形泡泡库、第三方视频播放等。本书在结构设计上采用由小项目逐渐深入的形式,然后引出一个黑马头条项目,讲解企业项目的开发流程。本书共 13 章,第 1~4 章每章分别讲解一个小项目,第 5~13 章每章分别讲解黑马头条项目的一个模块,包括从项目分析、效果展示到项目开发、打包发布的全过程。

本书既可作为高等院校本、专科计算机相关专业的教材,也可作为社会培训教材,是一本适合广大编程爱好者参考和学习的书籍。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Android 企业级项目实战教程/黑马程序员编著. —北京:清华大学出版社,2018

ISBN 978-7-302-49120-0

I. ①A… II. ①黑… III. ①移动终端—应用程序—程序设计—教材 IV. ①TP929.53

中国版本图书馆 CIP 数据核字(2017)第 315395 号

责任编辑:袁勤勇

封面设计:韩 冬

责任校对:焦丽丽

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京泽宇印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:21.5

字 数:525 千字

版 次:2018 年 2 月第 1 版

印 次:2018 年 2 月第 1 次印刷

印 数:1~1500

定 价:49.00 元

产品编号:077936-01



野蜂网资源站

前言

野蜂网资源站

本书共分13章，第1~4章分别讲解一个小项目，第5~13章讲解一个黑

马头条项目的各个模块，其中包括项目综述、欢迎模块、首页

模块、统计模块、视频模块、“我”模块、设置模块和项目上线。

前言

为什么要学习 Android

Android 是 Google 公司开发的基于 Linux 的开源操作系统，主要应用于智能手机、平板电脑等移动设备，还可用于穿戴设备、智能家居等领域。经过短短几年的发展，Android 系统在全球得到了大规模推广。据不完全统计，Android 系统已经用于全球 80% 以上的智能手机，中国市场的占有率更是高达 90% 以上，因此越来越多的人开始学习 Android 技术，以适应市场需并寻求更广阔的发展空间。

如何使用本书

本书适合有一定 Android 基础的读者阅读，书中涵盖多个技术热点，其中包括下拉刷新、HelloCharts 图表库、BoomMenu 圆形菜单、BubbleViews 心形泡泡库、第三方视频播放等。若读者没有 Android 基础，建议读者先学习《Android 移动开发基础案例教程》，然后再学习本书。

本书在结构设计上采用由小项目逐渐深入的形式，然后引出一个黑马头条项目，讲解企业项目的开发流程。本书共 13 章，第 1~4 章分别讲解一个小项目，第 5~13 章讲解一个黑马头条项目。具体如下。

- 第 1 章主要讲解弹幕项目的实现过程，其中包括弹幕下方的视频播放、弹幕文本的发送等。
- 第 2 章主要讲解 VR 项目的实现过程，其中包括全景图片与全景视频介绍、VR 全景图片的预览、VR 全景视频的播放等。
- 第 3 章主要讲解 HelloCharts 图表库的使用，其中以饼状图、柱状图、线形图为例进行演示。
- 第 4 章主要讲解 3D 相册的实现过程，其中包括图片展示与滑动效果等。
- 第 5~13 章分别讲解黑马头条项目的各个模块，其中包括项目综述、欢迎模块、首页模块、统计模块、视频模块、“我”模块、设置模块和项目上线。

读者在阅读本书的过程中难免会遇到一些问题，如果是对某个知识点不熟悉，则可以先行查阅，然后再进行项目开发。黑马头条项目是一个完整的项目，建议读者先厘清思路，多思考、多分析、多实践，逐步完成项目的开发。

如何领取课程

1. 扫描封面底部或右侧二维码，注册博学谷账号；
2. 点击底部导航栏“个人”，选择“我的优惠券”；
3. 输入封底刮刮卡中的优惠码，即可兑换相应的课程。



扫码领取课程

致谢

本书的编写和整理工作由传智播客教育科技有限公司完成，主要参与人员有吕春林、陈欢、柴永菲、张泽华、李印东、马伟奇、刘峰、金兴、邱本超、殷凯等，全体参编人员在将近一年的编写过程中付出了很多辛勤的汗水，在此一并表示衷心的感谢。

意见反馈

尽管我们尽了最大的努力，但本书中难免存在不妥之处，欢迎各界专家和读者朋友们来信来函给予宝贵意见，我们将不胜感激。读者在阅读本书时，如发现任何问题或有不认同之处，可以通过发送电子邮件与我们联系。

请发送电子邮件至 itcast_book@vip.sina.com。

黑马程序员

2017年9月于北京

目 录

第 1 章 弹幕	1
1.1 弹幕程序	1
任务综述	1
【任务 1-1】 弹幕界面	1
【任务 1-2】 弹幕界面逻辑代码	3
1.2 本章小结	7
第 2 章 VR	8
2.1 全景图片与全景视频介绍	8
2.2 VR 主界面	9
任务综述	9
【任务 2-1】 VR 主界面	9
【任务 2-2】 VR 主界面逻辑代码	10
2.3 VR 全景图片	11
任务综述	11
【任务 2-3】 VR 全景图片界面	11
【任务 2-4】 VR 全景图片界面逻辑代码	13
2.4 VR 全景视频	14
任务综述	14
【任务 2-5】 VR 全景视频界面	15
【任务 2-6】 VR 全景视频界面逻辑代码	16
2.5 本章小结	19
第 3 章 图表库	20
3.1 线形图	20
任务综述	20
【任务 3-1】 线形图界面	20
【任务 3-2】 创建 ViewPagerAdapter	22
【任务 3-3】 线形图界面逻辑代码	23

3.2 饼状图.....	26
任务综述.....	26
【任务 3-4】 饼状图界面	26
【任务 3-5】 饼状图界面逻辑代码	27
3.3 柱状图.....	30
任务综述.....	30
【任务 3-6】 柱状图界面	30
【任务 3-7】 柱状图界面逻辑代码	31
3.4 本章小结.....	33
第 4 章 3D 相册	34
4.1 相册.....	34
任务综述.....	34
【任务 4-1】 “相册”界面	34
【任务 4-2】 “相册”界面 Item	36
【任务 4-3】 创建 AlbumBean	38
【任务 4-4】 “相册”界面 Adapter	38
【任务 4-5】 “相册”界面逻辑代码	39
4.2 本章小结.....	43
第 5 章 项目综述	44
5.1 项目分析.....	44
5.1.1 项目名称	44
5.1.2 项目概述	44
5.1.3 开发环境	44
5.1.4 模块说明	45
5.2 效果展示.....	45
5.2.1 欢迎界面与主界面	45
5.2.2 “新闻详情”界面与“Python 学科”界面	45
5.2.3 “统计详情”界面	46
5.2.4 “视频详情”界面	48
5.2.5 “我”界面	48
5.2.6 “个人资料”界面	51
5.3 本章小结.....	53
第 6 章 欢迎模块	54
6.1 欢迎界面.....	54
任务综述.....	54
【任务 6-1】 欢迎界面	54

【任务 6-2】 欢迎界面逻辑代码	55
6.2 导航栏	56
任务综述	56
【任务 6-3】 标题栏	56
【任务 6-4】 底部导航栏	57
【任务 6-5】 底部导航栏逻辑代码	60
6.3 本章小结	63
第 7 章 首页模块	64
7.1 搭建服务器	64
任务综述	64
【任务 7-1】 首页广告栏数据	64
【任务 7-2】 首页新闻列表数据	66
7.2 工具类	67
任务综述	67
【任务 7-3】 创建 Constant 类	67
【任务 7-4】 创建 JsonParse 类	68
【任务 7-5】 创建 UtilsHelper 类	68
7.3 首页	69
任务综述	69
【任务 7-6】 水平滑动广告栏界面	69
【任务 7-7】 首页界面	72
【任务 7-8】 自定义控件 WrapRecyclerView	76
【任务 7-9】 首页界面 Item	80
【任务 7-10】 创建 NewsBean	83
【任务 7-11】 创建 AdBannerFragment	85
【任务 7-12】 创建 AdBannerAdapter	86
【任务 7-13】 首页界面 Adapter	88
【任务 7-14】 首页界面逻辑代码	90
7.4 新闻详情	96
任务综述	96
【任务 7-15】 “新闻详情”界面	97
【任务 7-16】 “新闻详情”界面逻辑代码	99
7.5 Python 学科	106
任务综述	106
【任务 7-17】 “Python 学科”界面	106
【任务 7-18】 “Python 学科”界面 Item	107
【任务 7-19】 创建 PythonBean	108
【任务 7-20】 “Python 学科”界面 Adapter	109

【任务 7-21】 “Python 学科”界面逻辑代码	110
7.6 本章小结	113
第 8 章 统计模块	115
8.1 统计	115
任务综述	115
【任务 8-1】 “统计”界面	115
【任务 8-2】 “统计”界面逻辑代码	117
8.2 统计详情	120
任务综述	120
【任务 8-3】 “Android 统计”详情界面	120
【任务 8-4】 “Android 统计”详情界面逻辑代码	122
【任务 8-5】 “Java 统计”详情界面	124
【任务 8-6】 “Java 统计”详情界面逻辑代码	126
8.3 本章小结	130
第 9 章 视频模块	131
9.1 视频列表	131
任务综述	131
【任务 9-1】 “视频列表”界面	131
【任务 9-2】 “视频列表”界面 Item	132
【任务 9-3】 创建 VideoBean	133
【任务 9-4】 “视频列表”界面 Adapter	135
【任务 9-5】 “视频列表”界面数据	136
【任务 9-6】 “视频列表”界面逻辑代码	138
9.2 视频详情	140
任务综述	140
【任务 9-7】 “视频详情”界面	141
【任务 9-8】 “视频目录”列表 Item	148
【任务 9-9】 画面尺寸菜单	149
【任务 9-10】 “视频目录”列表 Adapter	151
【任务 9-11】 创建 TopLineApplication	154
【任务 9-12】 创建 VideoDetailPagerAdapter	155
【任务 9-13】 创建 ParamsUtils	156
【任务 9-14】 视频播放进度条	157
【任务 9-15】 画面尺寸菜单逻辑代码	159
【任务 9-16】 视频清晰度菜单逻辑代码	160
【任务 9-17】 “视频详情”界面逻辑代码	162
9.3 本章小结	164

第 10 章	“我”模块(一)	166
10.1	创建数据库	166
	任务综述	166
	【任务 10-1】 创建 SQLite 数据库	166
	【任务 10-2】 创建 DBUtils 类	167
	【任务 10-3】 创建 UserBean	168
10.2	“我”	169
	任务综述	169
	【任务 10-4】 “我”界面	169
	【任务 10-5】 广播接收者	180
	【任务 10-6】 “我”界面逻辑代码	181
10.3	注册	185
	任务综述	185
	【任务 10-7】 “注册”界面	186
	【任务 10-8】 MD5 加密算法	192
	【任务 10-9】 “注册”界面逻辑代码	193
10.4	登录	196
	任务综述	196
	【任务 10-10】 “登录”界面	196
	【任务 10-11】 “登录”界面逻辑代码	199
10.5	个人资料	203
	任务综述	203
	【任务 10-12】 “个人资料”界面	203
	【任务 10-13】 “个人资料”界面逻辑代码	207
10.6	个人资料修改	214
	任务综述	214
	【任务 10-14】 个人资料修改界面	215
	【任务 10-15】 个人资料修改界面逻辑代码	216
10.7	本章小结	221
第 11 章	“我”模块(二)	222
11.1	日历	222
	任务综述	222
	【任务 11-1】 “日历”界面	222
	【任务 11-2】 “日历”界面逻辑代码	224
11.2	星座	226
	任务综述	226
	【任务 11-3】 “星座”界面	226

【任务 11-4】	创建 ConstellationBean	234
【任务 11-5】	“星座”界面数据	236
【任务 11-6】	“星座”界面逻辑代码	240
11.3	星座选择.....	244
	任务综述.....	244
【任务 11-7】	“星座选择”界面	244
【任务 11-8】	“星座选择”界面 Item	246
【任务 11-9】	“星座选择”界面 Adapter	247
【任务 11-10】	“星座选择”界面数据	249
【任务 11-11】	“星座选择”界面逻辑代码	250
11.4	涂鸦.....	252
	任务综述.....	252
【任务 11-12】	“涂鸦”界面	252
【任务 11-13】	涂鸦颜色选择界面	264
【任务 11-14】	创建 ColorsBean	271
【任务 11-15】	创建 BigSizeBean	272
【任务 11-16】	“涂鸦”界面逻辑代码	273
11.5	地图.....	280
	任务综述.....	280
【任务 11-17】	“地图”界面	280
【任务 11-18】	“地图”界面逻辑代码	281
11.6	本章小结.....	286
第 12 章	设置模块	287
12.1	收藏.....	287
	任务综述.....	287
【任务 12-1】	“收藏”界面	287
【任务 12-2】	“收藏”界面 Item	289
【任务 12-3】	“收藏”界面 Adapter	294
【任务 12-4】	收藏新闻信息表	297
【任务 12-5】	“收藏”界面逻辑代码	300
12.2	设置.....	302
	任务综述.....	302
【任务 12-6】	“设置”界面	303
【任务 12-7】	“设置”界面逻辑代码	305
12.3	修改密码.....	308
	任务综述.....	308
【任务 12-8】	“修改密码”界面	308
【任务 12-9】	“修改密码”界面逻辑代码	310

12.4	设置密保.....	313
	任务综述.....	313
	【任务 12-10】 “设置密保”界面	314
	【任务 12-11】 “设置密保”界面逻辑代码	316
12.5	本章小结.....	320
第 13 章 项目上线		321
13.1	代码混淆.....	321
	13.1.1 修改 build.gradle 文件	321
	13.1.2 编写 proguard-rules.pro 文件	321
13.2	项目打包.....	323
13.3	项目加固.....	326
13.4	项目发布.....	330
13.5	本章小结.....	332

1.1 功能描述

任务描述

本任务的功能是完成应用的安全加固，主要包含两个方面：一是应用代码的混淆，二是应用代码的加固。在开发过程中，为了提高应用的安全性，防止应用被逆向工程，需要对应用代码进行混淆和加固。本任务将详细介绍如何实现这两个功能。

【任务 1-1】 设置界面

【任务分析】

本任务的目标是完成应用的安全加固，主要包含两个方面：一是应用代码的混淆，二是应用代码的加固。在开发过程中，为了提高应用的安全性，防止应用被逆向工程，需要对应用代码进行混淆和加固。本任务将详细介绍如何实现这两个功能。

【任务实现】

1. 设置界面：在开发过程中，为了提高应用的安全性，防止应用被逆向工程，需要对应用代码进行混淆和加固。本任务将详细介绍如何实现这两个功能。

2. 设置界面：在开发过程中，为了提高应用的安全性，防止应用被逆向工程，需要对应用代码进行混淆和加固。本任务将详细介绍如何实现这两个功能。

3. 设置界面：在开发过程中，为了提高应用的安全性，防止应用被逆向工程，需要对应用代码进行混淆和加固。本任务将详细介绍如何实现这两个功能。

第 1 章

弹 幕

学习目标

- 掌握弹幕界面的开发,能够独立实现弹幕程序

在程序开发中,弹幕用得最多的地方要数直播平台了,例如常见的斗鱼。在观看视频直播时,通常在屏幕底部会有一个输入框,专门用于输入弹幕。弹幕其实很像人们发送的即时消息,只不过是显示在直播界面上。本章让我们一起实现自己的弹幕程序吧!

1.1 弹幕程序

任务综述

弹幕界面主要显示视频播放、弹幕文本信息、弹幕输入框以及弹幕的“发送”按钮,当点击弹幕界面上的任意地方时,界面底部会弹出一个输入框和一个“发送”按钮,此时可以输入文字并发送。当再次点击弹幕界面上的任意地方时,界面底部的输入框和“发送”按钮便会消失。为了与他人发送的弹幕有所区分,本机发送的弹幕会添加一个蓝色的边框。

【任务 1-1】 弹幕界面

【任务分析】

弹幕界面主要显示视频播放、弹幕文本信息、弹幕输入框以及弹幕的“发送”按钮,界面效果如图 1-1 所示。

【任务实施】

(1) 创建项目。首先创建一个项目,将其命名为 Barrage,指定包名为 com. itheima. barrage。将项目的 icon 图标 barrage_icon.png 导入 mipmap-hdpi 文件夹。

(2) 导入视频文件。在程序的 res 目录中创建一个 raw 目录,将视频文件 sun.mp4 导入该目录。

(3) 添加 DanmakuFlameMaster 库。由于弹幕界面中用到了 DanmakuFlameMaster 库中的 DanmakuView 控件,因此需要将 DanmakuFlameMaster 库添加到本项目中。在 Android Studio 中,选中项目,右键选择 Open Module Settings 选项后选择 Dependencies 选项卡,单击右上角的绿色加号,选择 Library dependency 选项,把 com. github. ctiao: DanmakuFlameMaster:0. 5. 3 库加入项目。

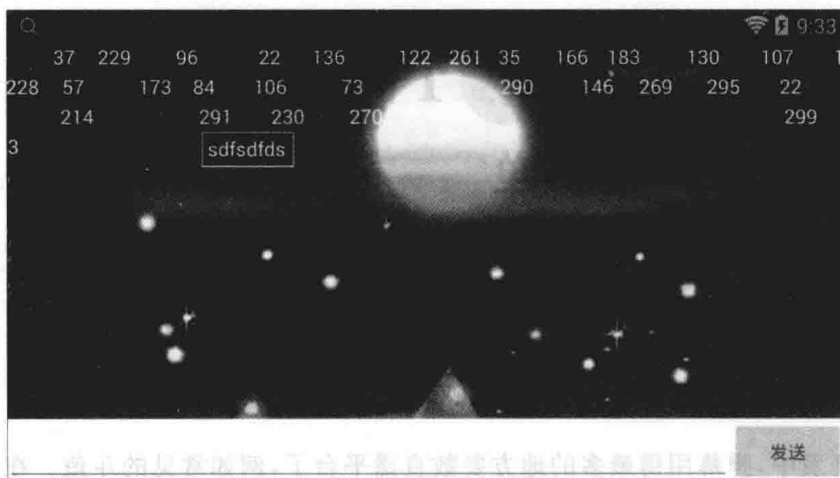


图 1-1 弹幕界面

注意：如果在 Library dependency 选项中找不到该库，则可以直接在该项目的 build.gradle 文件中添加如下代码。

```
compile 'com.github.ctiao:DanmakuFlameMaster:0.5.3'
```

(4) 放置界面控件。在 activity_main.xml 文件中放置一个 VideoView 控件用于播放视频，一个 DanmakuView 控件用于显示弹幕，一个 EditText 控件用于输入弹幕文本，一个 Button 控件用于显示发送弹幕的按钮，具体代码如文件 1-1 所示。

【文件 1-1】 activity_main.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:id="@+id/activity_main"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:background="#000">
7      <VideoView
8          android:id="@+id/videoview"
9          android:layout_width="match_parent"
10         android:layout_height="wrap_content"
11         android:layout_centerInParent="true" />
12     <master.flame.danmaku.ui.widget.DanmakuView
13         android:id="@+id/danmaku"
14         android:layout_width="match_parent"
15         android:layout_height="match_parent" />
16     <LinearLayout
17         android:id="@+id/ly_send"
18         android:layout_width="match_parent"
19         android:layout_height="50dp"
20         android:layout_alignParentBottom="true"
21         android:background="#fff"
22         android:visibility="gone">
```



```
23     <EditText
24         android:id="@+id/et_text"
25         android:layout_width="0dp"
26         android:layout_height="match_parent"
27         android:layout_weight="1" />
28     <Button
29         android:id="@+id/btn_send"
30         android:layout_width="wrap_content"
31         android:layout_height="match_parent"
32         android:text="发送" />
33 </LinearLayout>
34 </RelativeLayout>
```

【任务 1-2】 弹幕界面逻辑代码

【任务分析】

当点击弹幕界面时,界面底部会弹出一个输入框和一个“发送”按钮,当输入文字并点击“发送”按钮后,弹幕会出现一个蓝色框,框中显示刚刚发送的文字;当再次点击弹幕界面时,界面底部的输入框和“发送”按钮消失。

【任务实施】

(1) 获取界面控件。在 MainActivity 中创建一个 initView() 方法,用于获取弹幕界面上所要用到的控件。

(2) 播放视频。在 MainActivity 中创建一个 playVideo() 方法,在该方法中获取 res/raw 文件夹中的 sun.mp4 视频文件的 uri,并调用 setVideoURI() 方法设置 uri,然后调用 start() 方法播放视频。

(3) 初始化弹幕。在 MainActivity 中创建一个 initDanmaku() 方法,用于初始化弹幕并调用弹幕的随机生成与添加方法。

(4) 随机生成与添加弹幕。在 MainActivity 中创建 generateDanmakus() 与 addDanmaku() 方法,分别用于随机生成弹幕文字与添加弹幕到屏幕上,具体代码如文件 1-2 所示。

【文件 1-2】 MainActivity.java

```
1 package com.itheima.barrage;
2 public class MainActivity extends AppCompatActivity {
3     private boolean showDanmaku;
4     private DanmakuView danmakuView;
5     private DanmakuContext danmakuContext;
6     private Button sendButton;
7     private LinearLayout sendLayout;
8     private EditText editText;
9     private VideoView videoView;
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
```



```

13     setContentView(R.layout.activity_main);
14     initView();
15     playVideo();
16     initDanmaku();
17 }
18 /**
19  * 初始化界面控件
20  */
21 private void initView() {
22     videoView=(VideoView) findViewById(R.id.videoview);
23     sendLayout=(LinearLayout) findViewById(R.id.ly_send);
24     sendButton=(Button) findViewById(R.id.btn_send);
25     editText=(EditText) findViewById(R.id.et_text);
26     danmakuView=(DanmakuView) findViewById(R.id.danmaku);
27 }
28 /**
29  * 播放视频
30  */
31 private void playVideo() {
32     String uri="android.resource://" +getPackageName()+ "/" +R.raw.sun;
33     if(uri !=null) {
34         videoView.setVideoURI(Uri.parse(uri));
35         videoView.start();
36     } else {
37         videoView.getBackground().setAlpha(0);           //将背景设为透明
38     }
39 }
40 /**
41  * 创建弹幕解析器
42  */
43 private BaseDanmakuParser parser=new BaseDanmakuParser() {
44     @Override
45     protected IDanmakus parse() {
46         return new Danmakus();
47     }
48 };
49 /**
50  * 初始化弹幕
51  */
52 private void initDanmaku() {
53     danmakuView.setCallback(new DrawHandler.Callback() { //设置回调函数
54         @Override
55         public void prepared() {
56             showDanmaku=true;
57             danmakuView.start(); //开始弹幕
58             generateDanmakus(); //调用随机生成弹幕方法
59         }
60         @Override
61         public void updateTimer(DanmakuTimer timer) {
62

```

```
63         @Override
64         public void drawingFinished() {
65         }
66     });
67     danmakuContext=DanmakuContext.create();
68     danmakuView.enableDanmakuDrawingCache(true);           //提升屏幕绘制效率
69     danmakuView.prepare(parser, danmakuContext);           //进行弹幕准备
70     //为 danmakuView 设置点击事件
71     danmakuView.setOnClickListener(new View.OnClickListener() {
72         @Override
73         public void onClick(View view) {
74             if(sendLayout.getVisibility()==View.GONE) {
75                 sendLayout.setVisibility(View.VISIBLE);       //显示布局
76             } else {
77                 sendLayout.setVisibility(View.GONE);           //隐藏布局
78             }
79         }
80     });
81     //为"发送"按钮设置点击事件
82     sendButton.setOnClickListener(new View.OnClickListener() {
83         @Override
84         public void onClick(View view) {
85             String content=editText.getText().toString();
86             if(!TextUtils.isEmpty(content)) {
87                 addDanmaku(content, true);                     //添加一条弹幕
88                 editText.setText("");
89             }
90         }
91     });
92 }
93 /**
94  * 添加一条弹幕
95  * @param content    弹幕的具体内容
96  * @param border 弹幕是否有边框
97  */
98 private void addDanmaku(String content, boolean border) {
99     //创建弹幕对象,TYPE_SCROLL_RIGHT表示从右向左滚动的弹幕
100     BaseDanmaku danmaku=danmakuContext.mDanmakuFactory.createDanmaku(
101         BaseDanmaku.TYPE_SCROLL_RIGHT);
102     danmaku.text=content;
103     danmaku.padding=6;
104     danmaku.textSize=30;
105     danmaku.textColor=Color.WHITE;           //弹幕文字的颜色
106     danmaku.setTime(danmakuView.getCurrentTime());
107     if(border) {
108         danmaku.borderColor=Color.BLUE;       //弹幕文字边框的颜色
109     }
110     danmakuView.addDanmaku(danmaku);           //添加一条弹幕
111 }
```