

Android点播UI SDK 2.0

集成文档

changeLog

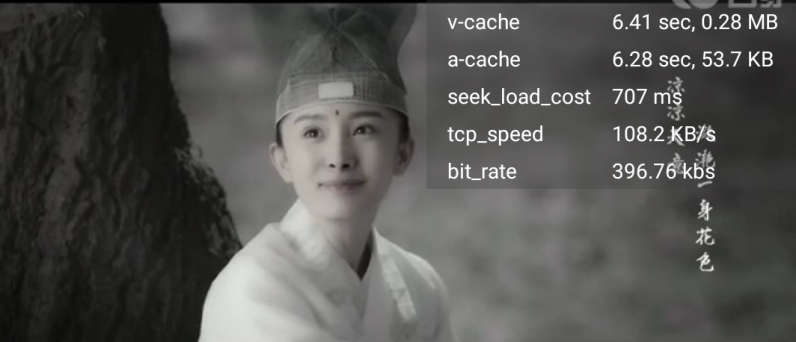
简介

带UI的点播SDK基于点播Core SDK，提供标准的UI实现，方便用户快速集成投入使用。此SDK代码开源，开发者可以自己建立分支进行开发，也欢迎给我们提issue

源码地址：<http://git2.baijiashilian.com/open-android/VideoplayerUI2.0Demo>(master分支)

项目截图

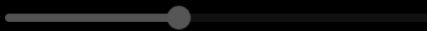
← 凉凉



vdec	avcodec
fps	22.86 / 22.86
v-cache	6.41 sec, 0.28 MB
a-cache	6.28 sec, 53.7 KB
seek_load_cost	707 ms
tcp_speed	108.2 KB/s
bit_rate	396.76 kbps



01:23



03:24





集成SDK

远程依赖

1. implementation
`'com.baijia.player:VideoplayerUI:3.19.3'`

源码依赖

1) IDE

推荐使用最新版Android studio集成SDK[点击下载](#)（需科学上网）

2) clone源码

确保电脑配置git命令行工具，执行如下指令，将源码clone到您的文件目录中

1. git clone -b master
`http://git2.baijiashilian.com/open-android/VideoplayerUI2.0Demo.git`

3) 配置项目settings.gradle文件

```
1. include ':videoplayer-ui'
2. ...其他module
3.
4. project(':videoplayer-ui').projectDir = new
   File('xxx(您的源码路径)/VideoPlayerUI/videoplayer-
   ui')
```

4) 添加依赖

在需要的module的build.gradle文件中添加源码依赖:

```
1. compile project(path: ':videoplayer-ui')
```

快速集成

1) 在Application初始化sdk

```
1. public class App extends Application {
2.     @Override
3.     public void onCreate() {
4.         super.onCreate();
5.         new BJYPlayerSDK.Builder(this)
6.             .setDevelopMode(true)
7.             //如果没有个性域名请注释
8.             .setCustomDomain("demo123")
9.             .setEncrypt(true)
10.            .build();
11.     }
12. }
```

通过BJYPlayerSDK.Builder配置参数, 可配置的参数有

- 设置开发者模式。开发者模式开启后会打印关键日志、显示播放器调试模板, 方便开发, **正式发版建议关掉**

```
1. /**
2.  * 设置开发者模式
3.  * @param isDevelopMode true 设置为开发者模式
4.  * @return
5.  */
6. public Builder setDevelopMode(boolean
    isDevelopMode)
```

- 设置专属域名。专属域名从百家云账号中心获取，传入规则：例如专属域名为 `demo123.at.baijiayun.com`，则前缀为 `demo123`，参考 [专属域名说明](#)。

```
1. /**
2.  * 设置专属域名
3.  *
4.  * @param domain 专属域名
5.  */
6. public Builder setCustomDomain(String
    domain)
```

- 设置是否加密，对在线播放和下载均有效

```
1. /**
2.  * 设置加密
3.  * @param isEncrypt true 加密，对在线和下载均有
    效
4.  * @return
5.  */
6. public Builder setEncrypt(boolean isEncrypt)
```

2) [请先阅读点播core2.0 sdk快速集成文档](#)，[电梯直达](#)

点播

VideoPlayActivity为标准的点播播放界面，该布局为全屏居中播放，用户可以参考实现任意组合下的点播播放。

- 在线播放

```
1. Intent intent = new
   Intent(VideoLauncherActivity.this,
   VideoPlayActivity.class);
2. intent.putExtra("videoid",
   Long.parseLong(videoid));
3. intent.putExtra("token", videoToken);
4. intent.putExtra("isOffline", false);
5. startActivity(intent);
```

- 离线播放

```
1. Intent intent = new
   Intent(VideoLauncherActivity.this,
   VideoPlayActivity.class);
2. intent.putExtra("videoPath", videoPath);
3. intent.putExtra("isOffline", true);
4. startActivity(intent);
```

功能介绍

包结构



自定义点播UI

component包下提供一系列标准播放器UI组件，用户可以在ComponentManager管理类中配置需要显示的组件。

```
1. /**
```

```

2.  * 默认组合的组件
3.  */
4.  public void generateDefaultComponentList(){
5.      componentMap.clear();
6.      componentChangeListeners.clear();
7.
8.      addComponent(UIEventKey.KEY_LOADING_COMPONENT,
9.          new LoadingComponent(context));
10.
11.      addComponent(UIEventKey.KEY_GESTURE_COMPONENT,
12.          new GestureComponent(context));
13.      //controller 需在gesture布局上方，否则会有事件冲突
14.
15.      addComponent(UIEventKey.KEY_CONTROLLER_COMPONENT,
16.          new ControllerComponent(context));
17.
18.      addComponent(UIEventKey.KEY_ERROR_COMPONENT,
19.          new ErrorComponent(context));
20.
21.      addComponent(UIEventKey.KEY_MENU_COMPONENT,
22.          new MenuComponent(context));
23.      if(BJYPlayerSDK.IS_DEVELOP_MODE){
24.
25.          addComponent(UIEventKey.KEY_VIDEO_INFO_COMPONENT,
26.              new MediaPlayerDebugInfoComponent(context));
27.      }
28.  }
29.  }

```

自定义组件

1) ComponentContainer

ComponentContainer extends FrameLayout，持有
ComponentManager初始化各个播放器组件，实现手势监听，

并且控制自定义事件和触摸事件的分发。

2) BaseVideoView

BaseVideoView extends FrameLayout，持有

IBJYVideoPlayer引用获取播放器回调，持有

ComponentContainer通知各个component组件更新状态，内部实现网络监听逻辑。

3) BJYVideoView

BJYVideoView extends BaseVideoView,初始化

BJYPlayerView和各个component,监听播放器回调通知各个component。

评论

1. 点播是否支持评论由 `enable_video_comment` 控制，可通过调用 `IBJYVideoPlayer.isEnableVideoComment()` 方法确认是否启用了评论功能。
2. 必须获取token参数，参考[点播评论获取token接口](#)自定义网络请求获取。使用VideoPlayActivity则只需将token赋给传入的VideoPlayerConfig的**loginToken**字段即可，内部会调用isEnableVideoComment判断是否启用评论。
3. token传递给CommentPresenter并结合CommentFragment即可使用评论功能。
CommentPresenter支持传递一个自己的头像url参数，发送评论时会自动携带该参数。获取评论时若头像字段为空则使用本地默认头像。
4. presenter的接口如下：

1. `// 刷新评论列表`
2. `void refreshComments();`
3. `// 加载更多评论，每次加载10条`

```
4. void loadMoreComments();
5. // 是否有更多评论
6. boolean hasMore();
7. // 获取已获得的评论列表
8. List<LPCommentDataModel.LPCommentItemModel>
   getComments();
9.
10. List<LPExpressionModel> getExpressions();
11. // 发送评论
12. void sendComment(String comment);
13. // 点赞评论
14. void likeComment(String commentId, boolean
   isLike, int position);
15. // 删除评论
16. void deleteComment(String commentId, int
   position);
```

也可参考Core SDK的评论网络接口，从而实现自己的评论功能方法。



下载为pdf格式