

BJPlayerManagerUI 点播UI SDK

概述

BJVideoPlayerCore(百家云 iOS Core SDK, 以下简称"Core SDK"), 是一款支持iOS **点播/回放**视频播放/下载的SDK, 不带UI, UI需要开发者自主集成. 适用于iOS8.0+。

BJPlayerManagerUI (百家云 iOS 点播 UI SDK, 以下简称"点播UI SDK"), 是基于BJVideoPlayerCore 封装的提供通用**点播**播放UI的SDK. 适用于iOS8.0+。

Demo

Demo是集成了点播UI SDK的项目, 以源码的形式详细展示了其中播放器的使用, 点播和回放视频的下载等功能. 开发者下载或者Git Clone本项目, 经过简单的修改即可运行**Demo**.

下面列表展示了**Demo**项目的项目结构:

1. BJPlayManagerUI
2. |—— Podfile
3. |—— Podfile.lock
4. |—— Pods
5. |—— Tests
6. |—— BJPlayerManagerUI.xcworkspace
7. |—— BJPlayerManagerUI
8. |—— PUAppdelegate.h
9. |—— PUAppdelegate.m
10. |—— PUMainViewController.h
11. |—— PUMainViewController.m (Demo 首页入口)
12. |—— PUPlayViewController.h
13. |—— PUPlayViewController.m (点播视频播放页面)
14. |—— playback (回放视频模块)
15. |—— BJPViewController.h (回放视频播放控制器)
16. |—— BJPViewController.m
17. |—— localVideo (本地视频播放模块)
18. |—— PMLocalViewViewController.h (本地视频播放控制器)
19. |—— PMLocalViewViewController.m
20. |—— download (下载模块)
21. |—— BJLDownload
22. |—— BJLDownloadViewController.h(视频下载控制器)
23. |—— BJLDownloadViewController.m
24. |—— setting (播放设置模块)
25. |—— PUVideoOptionsViewController.h (播放设置控制器)
26. |—— PUVideoOptionsViewController.m
27. |—— PUPlayOptionsManager.h
28. |—— PUPlayOptionsManager.m (配置Manager)
29. |—— extension (扩展类)
30. |—— UIAlertView+bjp.h
31. |—— UIAlertView+bjp.m
32. |—— NSString+md5.h
33. |—— NSString+md5.m
34. |—— UIWindow+PUMotion.h
35. |—— UIWindow+PUMotion.m
36. |—— UIViewController+PUUtil.h
37. |—— UIViewController+PUUtil.m
38. |—— BJPlayerManagerUI-Prefix.pch
39. |—— BJPlayerManagerUI-Info.plist
40. |—— Base.lproj
41. |—— LaunchScreen.storyboard
42. |—— Main.storyboard
43. |—— Images.xcassets (图片资源)
44. |—— main.m
- 45.

Demo运行

- 在 Example 的工程目录下执行 **pod install**
- 使用 Xcode 打开 Example 文件夹下的 BJPlayerManagerUI.xcworkspace 文件
- 将demo中的证书和bundleID更改为开发者自己的数据, 即可运行.

温馨提示

如果在Demo中发现了bug或者发现了SDK的bug，在告知我公司开发人员时，请将系统版本，硬件版本，产生bug的操作流程一并告知我公司开发人员，必要时需要提供日志。

BJPlayerManagerUI SDK

changelog

点播UI SDK是在Core SDK的基础上对通用点播播放功能的集成, 如果用户有更丰富的需求, 请参考点播UI SDK中接口的使用方式, 在理解点播SDK机制的前提下, 根据自己的需求进行自定义开发.适用于iOS8.0+。

通用功能简介

- 支持横竖屏切换
- 视频播放/暂停/快进/快退, 改变清晰度和倍速播放
- 横屏播放支持锁屏功能
- 视频播放界面左侧支持滑动调节亮度, 右侧支持滑动调节音量

点播UISDK源码已开源, 文件结构如下:

1. BJPlayManagerUI
2. |—— BJPlayManagerUI.h
3. |—— BJPlayManagerUI.m
4. |—— Resources
5. |—— Theme
6. |—— BJPUTheme
7. |—— Util
8. |—— BJPUVideoOptions (播放配置)
9. |—— BJPUAppearance
10. |—— MBProgressHUD+bjp
11. |—— View
12. |—— BJPUMediaControlView
13. |—— BJPUMediaSettingCell
14. |—— BJPUMediaSettingView
15. |—— BJPUProgressView
16. |—— BJPUReloadView
17. |—— BJPUSliderView
18. |—— ViewController
19. |—— BJPUViewController
20. |—— BJPUViewController+media
21. |—— BJPUViewController+observe
22. |—— BJPUViewController+protected
23. |—— BJPUViewController+reachability
24. |—— BJPUViewController+ui

SDK集成

Podfile配置

- Podfile中设置source

1. source 'https://github.com/CocoaPods/Specs.git'
2. source 'http://git.baijiashilian.com/open-ios/specs.git'
- 3.

- Podfile 中引入BJPlayerManagerUI

```
1. target 'yourProjectName' do
2.   pod 'BJPlayerManagerUI'
3. end
```

全局设置

- 配置ATS.
 - 需要在info.plist里面增加 `NSAllowsArbitraryLoads = true`
- 1.4.5 之后的版本支持设置专属域名前缀
 - 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. //注意:需要头文件中引入
2. #import <BJVideoPlayerCore/BJVAppConfig.h>
3.
4. [[BJVAppConfig sharedInstance] setPrivateDomainPrefix:privateDomainPrefix];
```

- 注意:**BJVideoPlayerCore 和 BJLRequestTokenDelegate

在开始使用点播/回放的播放和下载之前, 首先**必须**要为BJVideoPlayerCore设置tokenDelegate, 下面为tokenDelegate需要实现的两个方法:

```
1. //用于在播放/下载中, 客户需要调用completion提供某一videoID对应的视频播放token
2. - (void)requestTokenWithVideoID:(NSString *)videoID
3. completion:(void (^)(NSString * _Nullable token, NSError * _Nullable error))completion;
4.
5. //用于在播放/下载中, 客户需要调用completion提供某一回放课程对应的视频播放token
6. - (void)requestTokenWithClassID:(NSString *)classID
7. sessionID:(nullable NSString *)sessionID
8. completion:(void (^)(NSString * _Nullable token, NSError * _Nullable error))completion;
```

可以参考Demo的PUAppDelegate.m文件中BJLRequestTokenDelegate的使用方法

播放器使用

- 使用BJPUVideoOptions初始化BJPUViewController
- 根据需要设置block
- 添加播放器view到对应的页面上

```

1. self.playerUIVC = [[BJPUViewController alloc] initWithVideoOptions:[PUPlayOptionsManager
   getVideoOptions]];
2. [self addChildViewController:self.playerUIVC];
3. [self.playerUIVC didMoveToParentViewController:self];
4.
5. __weak typeof(self) weakSelf = self;
6. // 退出回调
7. [self.playerUIVC setCancelCallback:^(
8.     __strong __typeof__(weakSelf) strongSelf = weakSelf;
9.     [strongSelf.navigationController popViewControllerAnimated:YES];
10.  )];
11.
12. // 锁屏回调
13. [self.playerUIVC setScreenLockCallback:^(BOOL locked) {
14.     PUAppDelegate *appDelegate = (PUAppDelegate *)[UIApplication sharedApplication].delegate;
15.     appDelegate.allowRotation = !locked;
16.  }];
17.
18. [self.view addSubview:self.playerUIVC.view];

```

- 播放视频

```

1. //播放在线视频
2. [self.playerUIVC playWithVid:_vid token:_token];
3.
4. //播放本地视频BJVDownloadItem
5. [self.playerUIVC playWithDownloadItem:downloadItem];

```

定制UI元素(BJPUTheme.h)

```
1. //设置颜色
2. //默认状态下文本颜色
3. + (void)setDefaultTextColor:(UIColor *)color;
4. //选中状态的文本颜色
5. + (void)setHighlightTextColor:(UIColor *)color;
6. //设置topBar/bottomBar/definitionView/RateView的背景
7. + (void)setBrandColor:(UIColor *)color;
8.
9. //small image
10. //logo
11. + (void)setLogoImage:(UIImage *)image;
12. //backButtonImage
13. + (void)setBackButtonImage:(UIImage *)image;
14. //PlayButton
15. + (void)setPlayButtonImage:(UIImage *)image;
16. //pauseButton
17. + (void)setPauseButtonImage:(UIImage *)image;
18. //stopButton
19. + (void)setStopButtonImage:(UIImage *)image;
20. //nextButton
21. + (void)setNextButtonImage:(UIImage *)image;
22. //ProgressSlider
23. + (void)setProgressSliderImage:(UIImage *)image;
24. //ScaleButton
25. + (void)setScaleButtonImage:(UIImage *)image;
26. //ForwardImage
27. + (void)setForwardImage:(UIImage *)image;
28. //BackwardImage
29. + (void)setBackwardImage:(UIImage *)image;
```

代码示例

参见[Demo](#), Demo只是示范SDK中接口的使用方式,并不保证使用方式最符合用户的需求,请用户在充分理解SDK机制的前提下,根据自己的情况处理数据.对照抄SDKDemo代码引起的性能问题和其他非SDK bug,不负任何责任.

常见问题

- 为什么接入了BJPUViewController之后,点击全屏,无法全屏
 - 首先,看APP项目的设置,检查'General'-'Device Orientation'中是否打开了Landscape Left,Landscape Right.
 - 其次,确认在代码中, window、root view controller、navigation controller、tabbar controller 等VC中有没有特殊设置,是否支持旋转和支持的方向
 - 最后,如果APP其他界面代码不能支持横屏,可以仅在使用BJPUViewController时允许横屏,其他时候限制使用竖屏
- UI SDK的功能和界面无法满足现有需求
 - UI SDK是开源的,支持在源码基础上自定义修改,参考[点播 UI SDK自定义修改文档](#)