****

**Vstream**

**PRODUCT DOCUMENTATION**

***Version 1.0***

**5 March 2018**

|  |  |  |
| --- | --- | --- |
| ***Version Number*** | ***Date*** | ***Version Notes*** |
| *1.0* | 11 June 2018 | *First Stable Build* |

***Vstream iOS App***

Vstream is a fully customizable video streaming application that allows you to stream video content on iOS mobile devices anywhere and anytime! It helps you to create an iOS application for your own video streaming station and thus saves you a lot of time and money. This application was developed with Xcode IDE in such a way as to allow you to fully customize it to your own taste/rebrand it for whatever purposes you’d like to use it for. It comes pre-built with all basic mobile with custom video player volume control and accepts various streaming formats.

***CURRENT FEATURES***

* Full Native iOS Application
* Supports iOS 10.0 & Up
* Full Customization of App Name, Logo, Splash Screen Logo
* Play/Stop Live Video Stream for Single Station
* Supports interruptions
* Supports Single Video Streaming Stations
* Display Current Program, Next Program, and Timings
* Option To Show Weekly Schedule Of The Video Station From API
* Option To Show Upcoming Programs Of The Day
* Social Connection (Facebook, Youtube, Twitter, Instagram)
* Website page with WebView
* Easy volume control
* Support: m3u8
* High-quality user interface
* Check Network Availability
* Easy customization
* Well Documentation

SYSTEM REQUIREMENTS

**INSTALLATION**

VStream is an iOS application developed using Xcode. To build and run this project you have to install the latest version of Xcode on your Mac. To accomplish this, follow the below steps.

Step 1:

Download Xcode latest version from official iTunes website and install it.

https: //itunes.apple.com/in/app/xcode/id497799835?mt=12

Step 2:

Now it's the time to build and run the Vstream project you got as compressed zip file VideoStreaming.zip

Step 3:

Extract the VideoStreaming.zip file and copy it to a folder in your mac.

Step 4:

Now it’s the time to import Video Streaming iOS project to Xcode. To run Video Streaming project on Xcode you have to import that project from your folder location to Xcode.

Importing projects to Xcode

Step 1: Start Xcode

Step 2: From the File menu click File >Open

Step 3: Click Open

Step 4: Xcode imports the project

Step 5: Select a device from the list of devices. If u want to run in a real device, connect the device to your mac and select the device from the list of devices.

Step 6: Build and run the project

CUSTOMIZATION

You can change product name as your own product name.

You can change the app name from Video Streaming to the app name of your choice. You can do this by the following steps.

Step 1: Go to Xcode

Step 2: Go to the target

Step 3: Go to info in the target CUSTOM IOS TARGET PROPERTIES

Step 4: Change the Bundle Name to your App name.

**CUSTOMIZE COLOR**

1. To change the default color of the app, goto Constants.swift

2. Change the hex color value for all the values

Eg: dayTextColor = "#363630"

Eg: daySelectionColor = "#272727"

Eg: unselectedDayTextColor = "#FCFFFF"

Eg: unselectedDaySelectionColor = "#1F1F1F"

Eg: tabSelectionColor = "#272727"

Eg: tableTextColor = "#FFFFFF"

Eg: programcurrentdateColor = "#FCE76C"

**APP LOGO**

You can also change the app logo of your own icon by the following steps.

1. You can use makeappicon.com to create app icons of all sizes for your app.

Goto, https://makeappicon.com/

2. Click choose file and select your icon.

3. Enter your email id and press download button.

4. A zip file containing app icons of all sizes will be sent to your mail. Download the attachment.

5. In the Xcode project, click on Images.xcassets in the project navigator.

6. Unzip the attachment that you had downloaded earlier.

7. Open iOS folder.

8 . Open AppIcon.appiconset

9 . In Xcode, place the app icons of corresponding sizes in Images.xcassets

**App splash screen**

1. Drag your Splash screen to the Assets.xcassets folder in the project bundle to images named splashlogo.

**Video Stream URL**

1. Go to Constants.swift

**1.1.** If your Base streaming URL is coming from an API, Then set the URL :

Edit videoStreamURLfromApi to your custom URL.

And put videoStreamURLfromApi\_endpoint to your custom URL.

The API response is in the form

{  
"Channel\_url": ""  
   
}

**1.2.** If your Video streaming URL is not coming from an API, then ignore step **2.1**. And set the

Radio Streaming Url videoStreamURLfromApi\_endpoint directly. to the correct URL string.

***\*\*Note: You have to set either step 1.1 or step 1.2, Or streaming won’t work.\*\****

**Program schedule URL**

**1.3**. If you want to show the schedule of your programs, and your program Schedule is coming from an API, Then set the Url: Edit API\_BASE\_URL to your Base URL inside the

quotes

Edit scheduleUrl to the endpoint of API\_BASE\_URL

if schedule list is not there put scheduleUrl as “”

The API response should be in the form :

{

"schedule": [

{

"day": "Sunday",

"events": [

{

"show\_time\_start": "11:30 PM",

"show\_time\_end": "12:00 AM",

"show\_title": "abc",

"duration": "30 min",

"show\_description": ""

}

]

},

{

"day": "Monday",

"events": [

{

"show\_time\_start": "12:00 AM",

"show\_time\_end": "12:30 AM",

"show\_title": "xyz",

"duration": "30 min",

"show\_description": ""

}

]

},

{

"day": "Tuesday",

"events":

{

"show\_time\_start": "12:00 AM",

"show\_time\_end": "12:30 AM",

"show\_title": "2D",

"duration": "30 min",

"show\_description": ""

}

]

},

{

"day": "Wednesday",

"events": [

{

"show\_time\_start": "12:00 AM",

"show\_time\_end": "12:30 AM",

"show\_title": "abc",

"duration": "30 min",

"show\_description": ""

}

]

},

{

"day": "Thursday",

"events": [

{

"show\_time\_start": "12:00 AM",

"show\_time\_end": "12:30 AM",

"show\_title": "xyz",

"duration": "30 min",

"show\_description": ""

}

]

},

{

"day": "Friday",

"events": [

{

"show\_time\_start": "12:00 AM",

"show\_time\_end": "12:30 AM",

"show\_title": "xyz",

"duration": "30 min",

"show\_description": ""

}

]

},

{

"day": "Saturday",

"events": [

{

"show\_time\_start": "12:00 AM",

"show\_time\_end": "12:30 AM",

"show\_title": "xyz",

"duration": "30 min",

"show\_description": ""

}

]

}

]

}

**Facebook URL**

**1.4**. If you want to customize the Facebook URL, goto Constants.swift file and customize

facebookLink = ""

**Website URL**

**1.5**. If you want to customize the Website link URL, goto Constants.swift file and customize

websiteLink = “”

**Youtube URL**

**1.6**. If you want to customize the Youtube URL, goto Constants.swift file and customize

youtubeLink = “”

**Twitter URL**

**1.7**. If you want to customize the Twitter URL, goto Constants.swift file and customize

twitterLink = “”

**Instagram URL**

**1.7**. If you want to customize the Instagram URL, goto Constants.swift file and customize

instagramLink = “”

***Submitting to App Store***

You can modify text, icons and color themes in the RadioLite project itself. Our Vstream documentation contains assistance for Product Customization. Since iTunes accepts only unique bundle identifier for each app, before submitting your app, should change your bundle identifier in Xcode.

**Change your bundle identifier in Xcode**

1. In your Project Navigator, click on the project name

2. Click the project name under Targets

3. Change the bundle identifier from “webncrafts.VideoStreaming” to your bundle identifier

Now you have completed the full customization of your project. Finally, Save, Build and Run your customized project.

4. Click archive under product menu

5. Once the archive is created, click ‘Upload to app store’

6. Please refer the following links for steps to follow in order to submit the app to iTunes connect

https://clearbridgemobile.com/how-to-submit-an-app-to-the-app-store/

https://www.youtube.com/watch?v=6uX7B8ZfMiw

***Note: Make sure the resources you used for your own customized version includes only .png files for App Logo, App icon and App splash screen with their corresponding name, size, and dimensions of the RadioLite project resource files.***