



FREE eBook

LEARNING

tvOS

Free unaffiliated eBook created from
Stack Overflow contributors.

#tvOS

Table of Contents

About	1
Chapter 1: Getting started with tvos	2
Remarks.....	2
Examples.....	2
Example for TVML app.....	2
Credits	6

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [tvos](#)

It is an unofficial and free tvos ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official tvos.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with tvos

Remarks

This section provides an overview of what tvos is, and why a developer might want to use it.

It should also mention any large subjects within tvos, and link out to the related topics. Since the Documentation for tvos is new, you may need to create initial versions of those related topics.

Examples

Example for TVML app

For starting tvOS I assume must have knowledge of swift and iOS app. One can start reading article from [apple reference document](#).

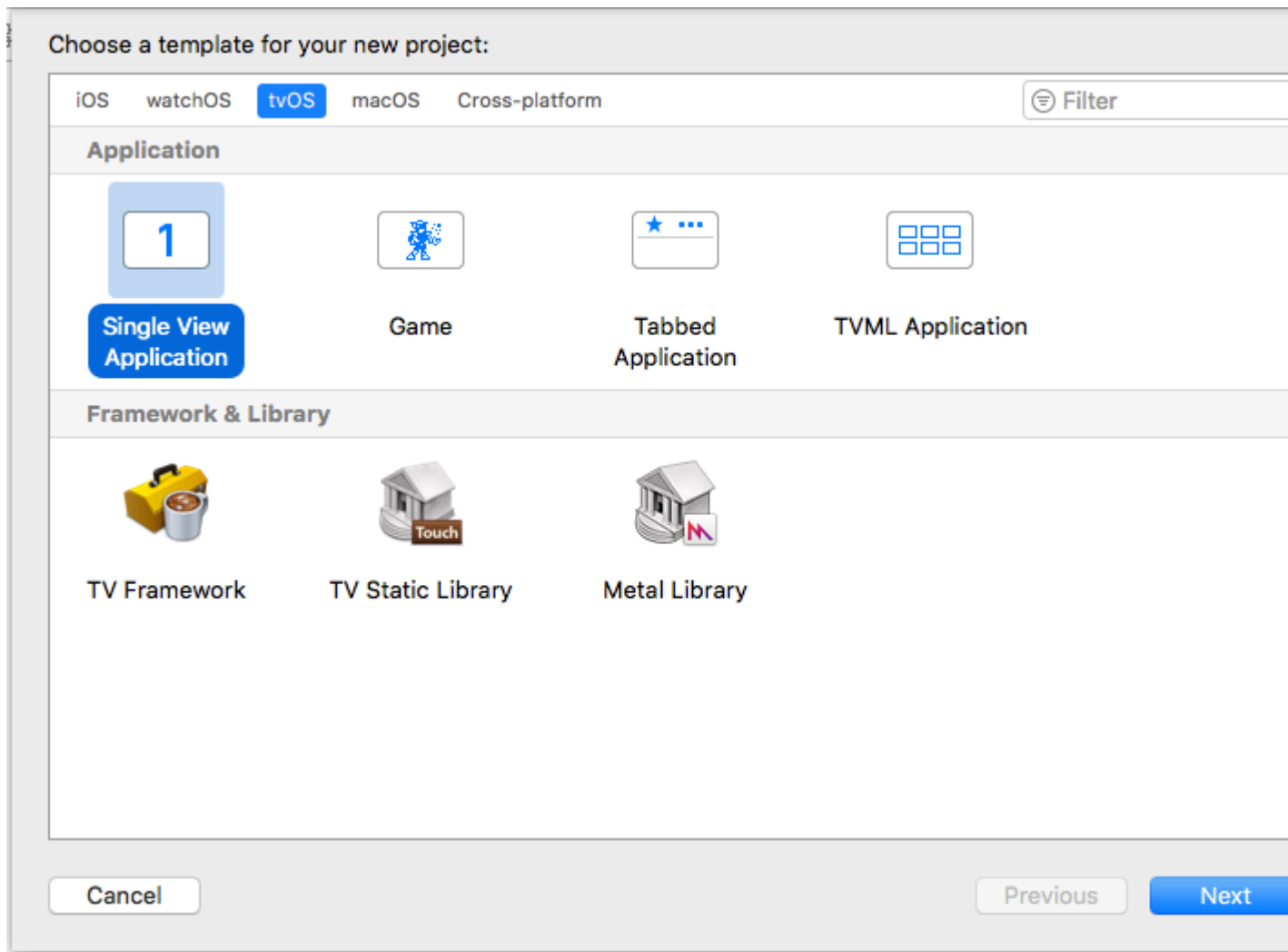
There are 2 type of Tv app we can create one is traditional app, i.e created in xcode code would be similar as we do in iOS app.

Here we take example of other type of tv app it's called Client server app

[![enter image description here][2]][2]

To build a client-server app:

1. Open Xcode and create a new project.
2. Select the Single View Application template from tvOS.



3. Remove the view controller file and the main storyboard for the app.
4. Open the info.plist file and remove the Main storyboard file base name entry.

NOTE To set up your network security, see [NSAppTransportSecurity][3].

5. Make these changes to the AppDelegate.swift file:
 - Add import TVMLKit.
 - Change the class declaration to be class AppDelegate: UIResponder, UIApplicationDelegate, TVApplicationControllerDelegate {Add the following global variable to your class: var appController: TVApplicationController?
 - Modify application:didfinishLaunchingWithOptions: according to the code found in the listing below:

```
func application(application: UIApplication, didFinishLaunchingWithOptions launchOptions:
[NSObject: AnyObject]?) -> Bool {
    self.window = UIWindow(frame: UIScreen.mainScreen().bounds)

    let appControllerContext = TVApplicationControllerContext()

    let javascriptURL = NSURL(string: "Enter path to your JavaScript file here")

    appControllerContext.javascriptApplicationURL = javascriptURL
    appControllerContext.launchOptions["BASEURL"] = TVBaseURL
```

```

    if let options = launchOptions {
        for (kind, value) in options {
            if let kindStr = kind as? String {
                appControllerContext.launchOptions[kindStr] = value
            }
        }
    }

    self.appController = TVApplicationController(context: appControllerContext, window:
self.window, delegate: self)

    return true
}

```

The code in the above example loads a JavaScript file which then loads a TVML page and displays it in the simulator or on a television screen if a new Apple TV is connected to your computer. For more information about JavaScript classes, see [Apple TV JavaScript Framework Reference](#).

The JavaScript in Listing 2-1 loads a TVML page (Listing 2-2) that displays an alert asking if the user wants to upgrade to the premium version of your app. After the page is loaded, it is pushed onto the navigation stack. The operating system then displays it to the user. For more information on available TVML templates and elements, see [Apple TV Markup Language Reference](#).

Listing 2-1 Pushing a TVML page onto the navigation stack

```

function getDocument(url) {
    var templateXHR = new XMLHttpRequest();
    templateXHR.responseType = "document";
    templateXHR.addEventListener("load", function() {pushDoc(templateXHR.responseXML);},
false);
    templateXHR.open("GET", url, true);
    templateXHR.send();
    return templateXHR;
}

function pushDoc(document) {
    navigationDocument.pushDocument(document);
}

App.onLaunch = function(options) {
    var templateURL = 'Enter path to your server here/alertTemplate.tvml';
    getDocument(templateURL);
}

App.onExit = function() {
    console.log('App finished');
}

```

Listing 2-2A TVML page to display an alert

```

<document>
  <alertTemplate>
    <title>Update to premium</title>
    <description>Go ad free by updating to the premium version</description>
    <button>

```

```
        <text>Update Now</text>
    </button>
    <button>
        <text>Cancel</text>
    </button>
</alertView>
</alertTemplate>
</document>
```

Read **Getting started with tvos** online: <https://riptutorial.com/tvos/topic/8138/getting-started-with-tvos>

Credits

S. No	Chapters	Contributors
1	Getting started with tvos	Community , duskwuff , sschunara