Configuring

We can now configure our mobile application, and set it to use the Configuration(s) created in the previous step. In addition, we will set some basic information, like API keys and your apps name. Open up *AppDelegate.h* found in the 'Universal' folder in the project explorer to get started.

Set to use your Configuration

Local configuration

If you'd like to use a local configuration file (or would like to add local overview configurations), this is the time to add the Configuration JSON files to your app's project. First, delete the placeholder JSON files in the Local directory. Next, add your own config.json file (and possibly also your overview json files) to the 'Local' folder of your project using XCode as is shown below:

Online configuration

You can configure Universal to load a configuration file from the web, which has the advantage that you can update your app's content dynamically, without users having to update their app. Upload your JSON file to your web server if you haven't already, and make sure that its reachable by url. Next, open in AppDelegate.h set the CONFIG definition to point to your main configuration. Online configurations are cached locally in your app for 2 days.

For example, if your main configuration JSON can be found at blog.com/configuration.json:

```
#define CONFIG @"http://blog.com/configuration.json"
```

Your apps theme color

Be default, we'll use the light theme. You can change your apps theme color (most used as NavigationBar color) by first setting APP_THEME_LIGHT to NO and then changing the *APP_THEME_COLOR* variable to any color you like. You can manually replace the color variables (as in RGB, by default 51.0, 180,0, 227.0) or a tool like UIColor Picker (http://www.briangrinstead.com/blog/ios-uicolor-picker), just replace the generated *[UIColor ..]* with the existing *[UIColor ..]* value.

You can do something similarly change the background gradient of the menu by editing *MENU_BACKGROUND_COLOR_1* and *MENU_BACKGROUND_COLOR_2*.

Your about dialog & contact info

If you want to use the build in about dialog, simply replace the values between the quotes for NO_CONNECTION_TEXT to replace the placeholder about text (using n for line breaks) with your own about text. And replace the placeholder about url between the quotes for ABOUT_URL with a link to your

own support or contact page.

Don't want a about dialog? Just leave the value for NO_CONNECTION_TEXT empty, so it looks like this @""

Warning: Apple might require you to have contact information in your app, therefore, we do not recommend leaving it empty unless you provide contact information elsewhere in your app.

Admob

Universal supports Interstitial and Banner advertisements. If you would like to use Admob Banners to monetize your app, you need to enter an Admob Banner Unit ID. In addition, the value for BANNER_ADS_ON must be set to *true*.

Create an Admob Ad Unit ID (https://support.google.com/admob/answer/3052638?hl=en) for banners in the Admob Console (https://apps.admob.com/). After you've retrieved this ID, enter it between the quotes for ADMOB_UNIT_ID.

If you'd (also) like to show Interstitial advertisements, return to the console (https://apps.admob.com/) and create an Admob Ad Unit ID for interstitials, and enter it between the quotes for ADMOB_INTERSTITIAL_ID.
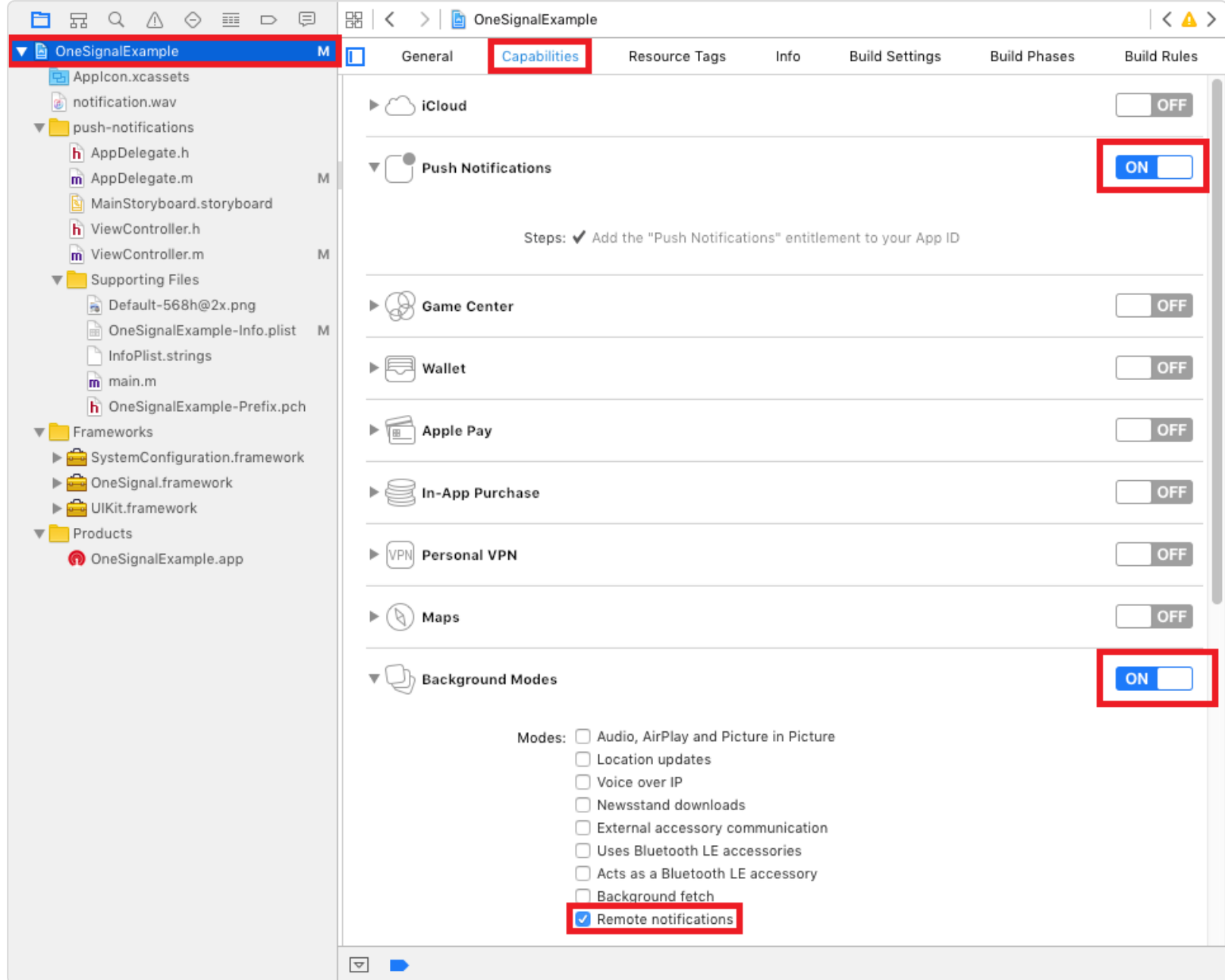
OneSignal

If you would like to enable Push Notifications with Universal, you can use OneSignal to send Push Notifications with Universal for IOS.

1. Make sure you have an account at OneSignal. Go to onesignal.com (http://onesignal.com) and sign in (or sign up if you do not have a account). And go to my apps (https://onesignal.com/apps).
2. Click 'create a new app' and enter a name Select IOS as platform and continue.
3. Now follow the OneSignal Documentation on Generating an IOS push certificate (https://documentation.onesignal.com/docs/generate-an-ios-push-certificate).
4. Return to the OneSignal.com to continue with the setup, select IOS as SDK.
5. Now, OneSignal will provide you with with an App ID.
6. Now open up AppDelegate.h again, and enter the App ID from the previous step for ONESIGNAL_APP_ID.
   If you do not enter anything between the quotes, Notifications will automatically be disabled.

Next enable remote notifications in Background Modes as described in the image below.

In-App purchases

Universal supports In-App purchases to remove Admob advertisements throughout your app and to make some menu items exclusively accessible for people that made an In-App purchase. In order to enable in-app purchases in Universal, you simply need to enter a non-consumable product ID between the quotes for IN_APP_PRODUCT. In addition, you'll need to enable the In-App purchase Capability for your target.

More information on how to create such an non-consumable product can be found in the introduction of this TutsPlus tutorial (https://code.tutsplus.com/tutorials/in-app-purchase-tutorial-with-swift-3-ios-sdk--cms-27595), you'll only need to read the introduction which describes how to create in-app products and set up a test account.

> **Important** In-App purchases require an Extended CodeCanyon license.

Api Keys

Google Maps API key

If you want to show a location in your app, using our integrated Maps content provider, you will need a Google Maps API key. *If you have no plans of integrating Google Maps, or if you are only using embedded maps (e.g. in your WordPress posts or WebView), you can continue to the next heading.*

Step 1: Enable the required APIs on the Google Developers Console

You need to activate the Google Maps SDK for iOS, in your project on the Google Developers Console.

Alternatively, you can activate the APIs yourself in the Developers Console, by doing the following:

1. Go to the Google Developers Console (https://console.developers.google.com/).
2. Select a project, or create a new one.
3. Enable the Google Maps SDK for iOS: In the sidebar on the left, expand APIs & auth. Next, click APIs. Select the Enabled APIs link in the API section to see a list of all your enabled APIs. Make sure that the API is on the list of enabled APIs. If you have not enabled it, select the API from the list of APIs, then select the Enable API button for the API.

Step 2: Get an API key

Using an API key enables you to monitor your application's API usage, and ensures that Google can contact you about your application if necessary. The key is free, you can use it with any of your applications that call the Google Maps SDK for iOS, and it supports an unlimited number of users. You obtain an API key from the Google Developers Console by providing your application's bundle identifier.

Follow these steps to create an API key from the Google Developers Console:

- In the sidebar on the left, select Credentials.
- Click Create New Key (and then optionally restrict it to iOS).
- If asked, confirm that you'd like to create the new key.

The Google Developers Console will now display your new API key. Here's an example: *AIzaSyBdVl-cTICSwYKrZ95SuvNw7dbMuDt1KG0*

Now enter the API key for MAPS_API_KEY (between the quotes).

YouTube API key

If you want to show YouTube videos in your app, using our integrated Youtube content provider, you will need a Youtube API key. *If you have no plans of integrating Youtube or if you are only using inline videos (e.g. in your WordPress posts or WebView), you can continue to the next heading.*

To get your Youtube API key, follow the following steps:

1. Go to the Google Developers Console (https://console.developers.google.com).
2. Select a project, or create a new one.
3. In the sidebar on the left, expand APIs & auth. Next, click APIs. In the list of APIs, make sure the status is ON for the YouTube Data API v3.
4. In the sidebar on the left, select Credentials.
5. Click Create New Key
6. If asked, confirm that you'd like to create the new key.

The Google Developers Console should now display a dialog with your API key. Here's an example: *AIzaSyBff7EnofVl0z6Yq9pqXQ−ZNXpuzlZrYN*

Now enter your own API key for YOUTUBE_CONTENT_KEY (between the quotes).

Twitter API keys and Access Tokens

If you want to show tweets in your app, using our integrated Twitter content provider, you will need Twitter API keys and access tokens. *If you have no plans of integrating Twitter or if you are only using embedded tweets (e.g. in your WordPress posts or WebView), you can continue to the next heading.*

Make sure to follow the following steps to obtain your Twitter API keys:

1. Go to https://dev.twitter.com/apps/new (https://dev.twitter.com/apps/new) and log in, if necessary
2. Enter your Application Name, Description and your website address. You can leave the callback URL empty.
3. Accept the TOS, and solve the CAPTCHA.
4. Submit the form by clicking the Create your Twitter Application
5. Copy the consumer key (API key) and consumer secret from the screen into your application

In addition, follow the steps described in here (https://dev.twitter.com/oauth/overview/application-owner-access-tokens) in Twitter's API Docs (https://dev.twitter.com/oauth/overview/application-owner-access-tokens) to obtain your Access Tokens.

Now that we have all the required information, enter the keys/tokens for TWITTER_API, TWITTER_API_SECRET, TWITTER_TOKEN, TWITTER_TOKEN_SECRET (between the quotes).

Facebook Access Token

If you want to show Facebook posts in your app, using our integrated Facebook content provider, you will need a Facebook access token. *If you have no plans of integrating Facebook or if you are only using embedded posts (e.g. in your WordPress posts or WebView), you can continue to the next heading.*

1. Go to developers.facebook.com and click on Log In in the top right. Log in using your personal Facebook credentials.
2. If this is your first time signing in to the Facebook Developer portal then click on Register Now. Registering is a quick an easy process which will take less than a couple of minutes. If you're already registered then you can skip ahead to step 9.
3. Accept the Facebook terms and click Continue.
4. Enter your phone number to confirm your account.
5. Facebook will send you an automated text message containing a confirmation code. Enter it in the box and click Confirm.
6. Choose to share your phone number with Only Me (unless you wish to share it with publicly or with friends).
7. If there is a step asking you to provide some information about yourself. You can skip this next step by clicking Skip.
8. Click Done.
9. Now click on Create New App.
10. If prompted, click on 'basic setup'
11. Enter your App Name. This can be anything you like. Click Continue.
12. If prompted with a security check, Fill in the Security Check and click Continue.
13. Your App should now be set up. Go to your apps dashboard and copy your App ID and App Secret. Please note that you need to click 'Show' next to the App Secret before copying.

We can use this ID en Secret, to generate an Access token. If you wish you can use this PHP script (http://stackoverflow.com/a/20514503/1683141) or this Python script (http://blog.lwolf.org/blog/2014/06/16/obtaining-never-expiring-access-token-to-post-on-facebook-page/) to obtain a valid access token over the Facebook API. You can also use the tool available at this website (https://smashballoon.com/custom-facebook-feed/access-token/)(at the bottom) or follow the tutorial here (https://www.rocketmarketinginc.com/blog/get-never-expiring-facebook-page-access-token/).

Now that we have our access token, enter your token for FACEBOOK_ACCESS_TOKEN (between the quotes).

Instagram Access Token

If you want to show Instagram posts in your app, using our integrated Instagram content provider, you will need an Instagram Access Token. *If you have no plans of integrating Instagram or if you are only using embedded posts (e.g. in your WordPress posts or WebView), you can continue to the next heading*

First, you need an Access Token, generate one as is described here. (https://sherdle.com/help/how-to-setup-the-new-instagram-api/)

Now that we have our Access Token, enter your token for INSTAGRAM_ACCESS_TOKEN (between the quotes).

Pinterest Access Token

If you want to show Pinterest posts in your app, using our integrated Pinterest content provider, you will need an Pinterest Access Token. *If you have no plans of integrating Pinterest you can continue to the next heading.*

1. If you haven't already, create a Pinterest Account.
2. Visit this page to learn how create an Access Token (https://developers.pinterest.com/docs/api/overview/?) (at least 'Read Public' is required).

Now that we have our Access Token, enter your token for PINTEREST_ACCESS_TOKEN (between the quotes).

Flickr API Key

If you want to show Flickr images in your app, using our integrated Flickr content provider, you will need an Flickr API key. *If you have no plans of integrating Flickr you can continue to the next heading.*

1. If you haven't already, create a Flickr Account.
2. Sign into your Flickr account at flickr.com.
3. Visit the API Page (https://www.flickr.com/services/api/misc.api_keys.html) and click the 'apply for your key now online' link.
4. Select 'Apply for a non-commercial key'.
5. Enter some information about your apps name and why you need Flickr in your app. Agree with the conditions and click the submit button.
6. You should now see your Flickr Key on the screen. Copy this key.

Now that we have our Key, enter your key for FLICKR_API (between the quotes).

SoundCloud ID

If you would like would like to play SoundCloud tracks in your app, using our integrated SoundCloud player, you will need an SoundCloud Client ID. If you have no plans of integrating SoundCloud you can continue to the next heading.

1. Navigate to the SoundCloud developers page (https://developers.soundcloud.com/) and login if you are not logged in already.
2. Select Your Apps and click on Register new App

3. Enter the required details and continue
4. You should now see the SoundCloud Client ID

Now that we have our SoundCloud Client ID, enter your ID for SOUNDCLOUD_CLIENT (between the quotes)
.

WooCommerce Host and API

If you would like to integrate WooCommerce in your app, using our WooCommerce content provider, you will need to enter your WooCommerce host and API keys. *If you have no plans of integrating WooCommerce you can continue to the next heading.*

1. Follow the WooCommerce documentation (https://docs.woocommerce.com/document/woocommerce-rest-api/) on enabling your API and generating keys. You only need Read access, we recommend to use an Admin user.
2. In AppDelegate.h enter your Consumer Key between the quotes for WOOCOMMERCE_KEY and your Consumer Secret Key between the quotes for WOOCOMMERCE_SECRET*.*
3. Next, enter the url to your WordPress Installation with WooCommerce for WOOCOMMERCE_HOST (without ending slash)*.*

👍 THIS ARTICLE WAS HELPFUL

🔥 0 people found this article useful