

Readit

User Guide

Thanks for purchasing **Readit**, Swift iOS Universal Social News App Template s from CodeCanyon, we really appreciate your support and we're sure you'll have so much fun with this template!

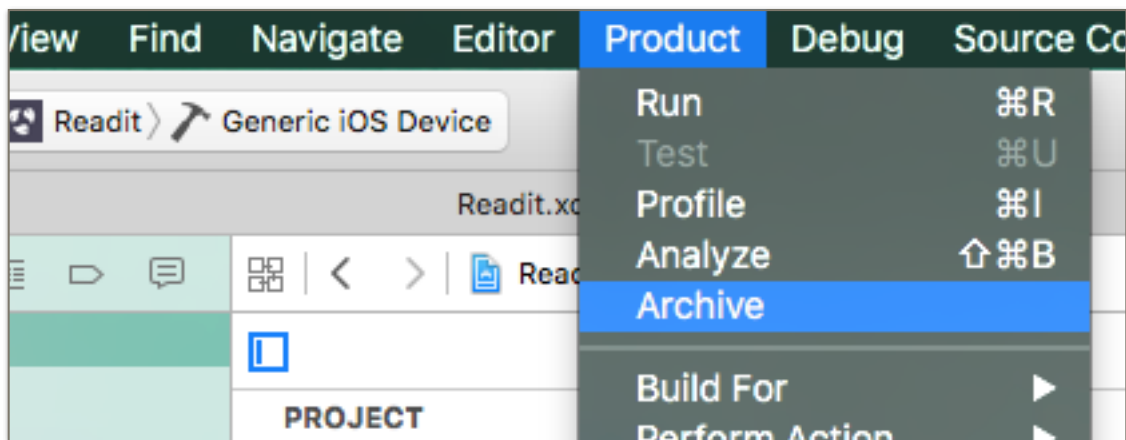
- First Setup -

- **Bundle Identifier & App name:** If you want to use the demo project included into the package, you can do so. Once you have generated a new **App ID** and **Distribution Provisioning** file from the **Apple Developer portal** (in order for you to be able to upload the binary of your app), you must change the **Bundle Identifier** accordingly to the one you generated in your provisioning profile. Also, you must change the App's name under **TARGETS** by double clicking on Readit and renaming it as you want, so you'll see your own app's name underneath the app's icon (see below):



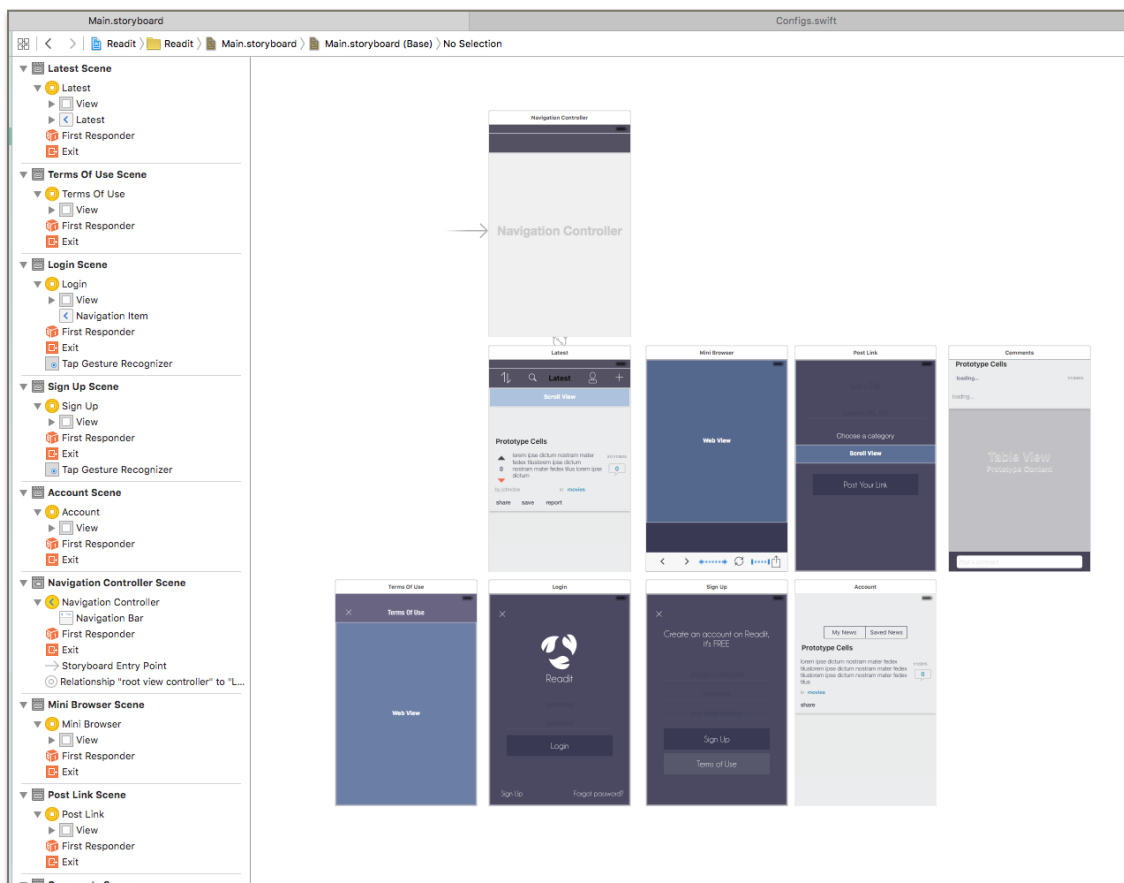
NOTE: *do not change the Readit folder's name or Project's name, or else XCode won't recognize the app and you won't be able to test it on the iOS Simulator nor Archive it for AppStore publishing.*

- **Archive the App:** To archive the app and be able to upload it to the **iTunes AppStore**, select **iOS Device** on the device list and then click on **Product -> Archive** to launch the **Organizer** and archive the app (see below):



- Customization -

- **User Interface:** If you want to edit the User Interface of Readit, you can do so by the Storyboards file called **Main.storyboard**. You can edit buttons, labels and image views via Inspector panel on the right side of the XCode window, and select them by the the Document Outline panel (see below)



- **Images:** You can easily change App icons, logo, buttons and other images by accessing **Assets.xcassets** folder and editing the *.png* images in there, just do not rename them so XCode will still recognize them. Make your reskin and save *.png* files with the same names as the existing ones in the project.

- Configuration -

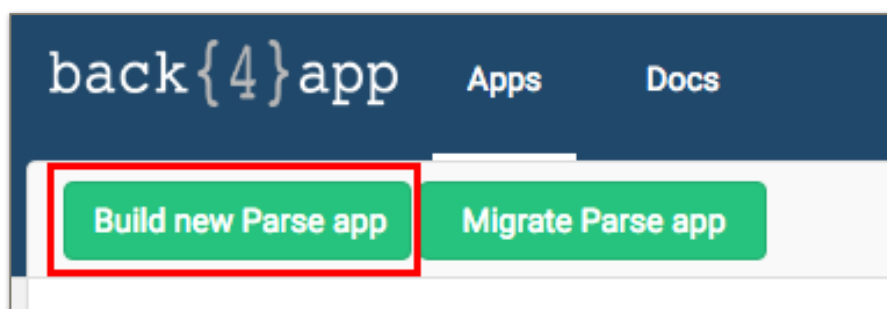
Head over **Configs.swift** file and check it out. It contains most of the keys/strings you have to set in order to customize your app and get connected to your own Parse App's database. Check out the **comments** in the swift file which will help you editing them.

Please also check **tou.html** file into the XCode project, that's the **Terms Of Use agreement** you can check within the app in the Sign Up screen. You have to edit it as you wish, especially when you'll change name to your app, since *tou.html* is just a sample terms of use agreement and it contains "**Readit**" name here and there.

- Parse database setup with back{4}app -

Before using the app, you must create your own App on <https://dashboard.back4app.com/apps> and a Bundle Identifier as mentioned in the beginning of this guide. The link we've provided above points already to the back4app Dashboard, but in case you don't have an account on back4app.com yet you'll be redirected to the sign up page.

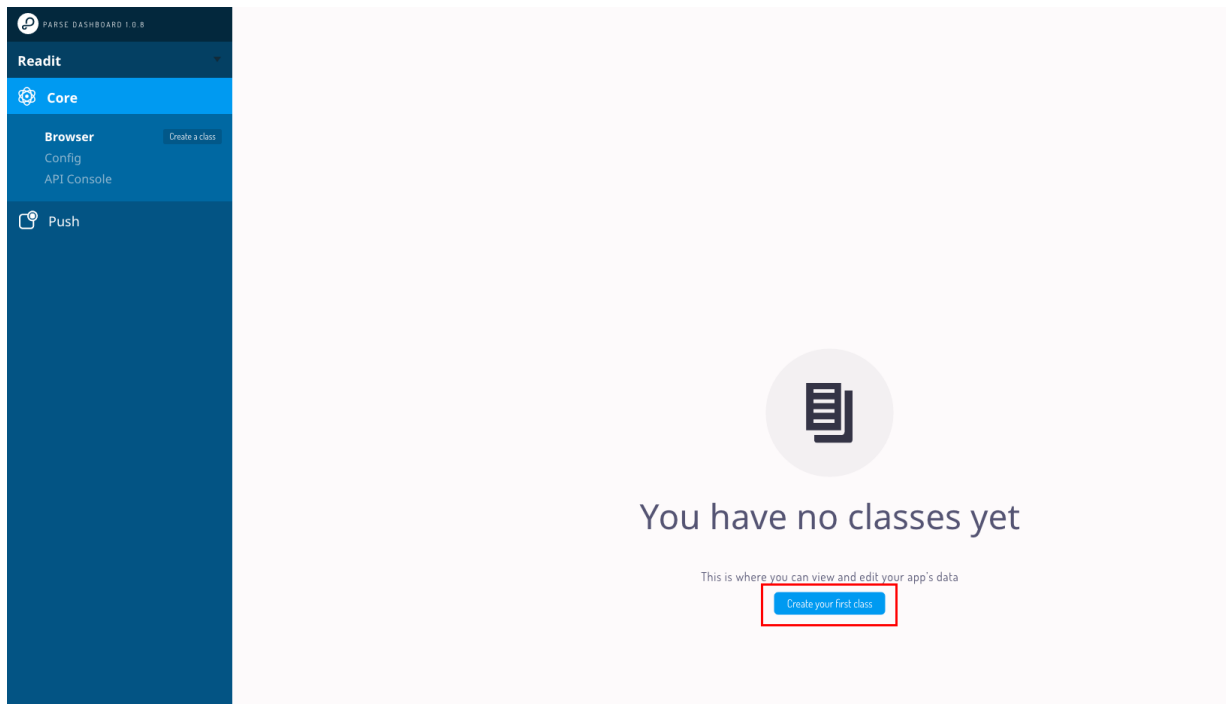
Once you'll be signed in in back4app, simply click on the **Build new Parse app** button, give your app a name and a brief description (optional), click **Next** and you'll be redirected to the setting page where you'll be able to copy the **App Key** and **Client Key** to replace the existing ones into **Configs.swift**



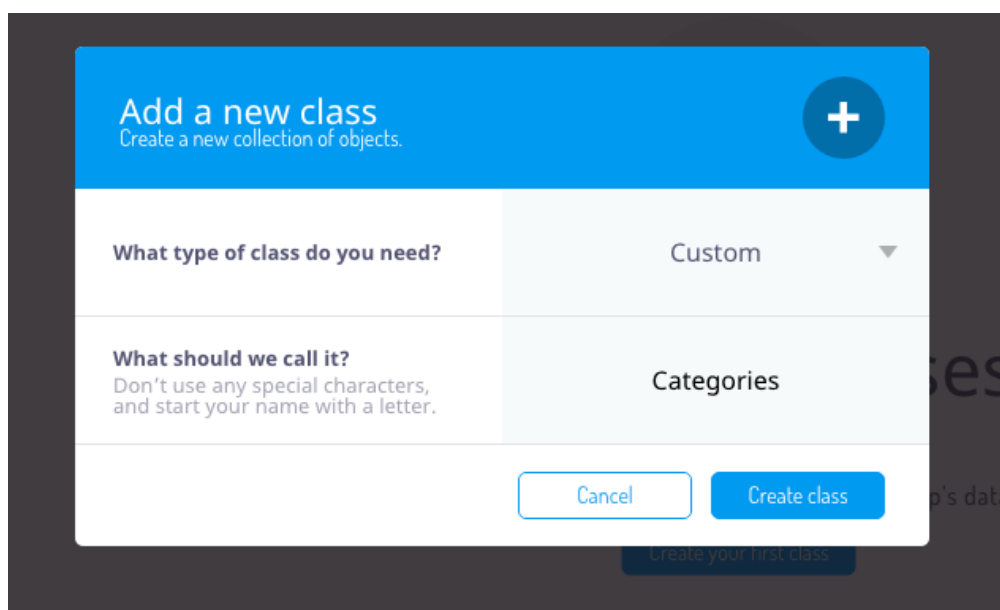
```
var PARSE_APP_KEY = "H27cDp92twGdYjJGW0SoCSHBVjJIboi5fCdgvaxEx"  
var PARSE_CLIENT_KEY = "OGBS6Ls36gQyf4oMQanBEyE45mGPn397IYDdirAK"
```

Then, you must create the **Categories** class on your Parse Dashboard and put some rows (records) into it, so the app will load them and work fine. You'll be able to add/edit those records later as you wish, but the important thing is that you insert **at least 1 record**.

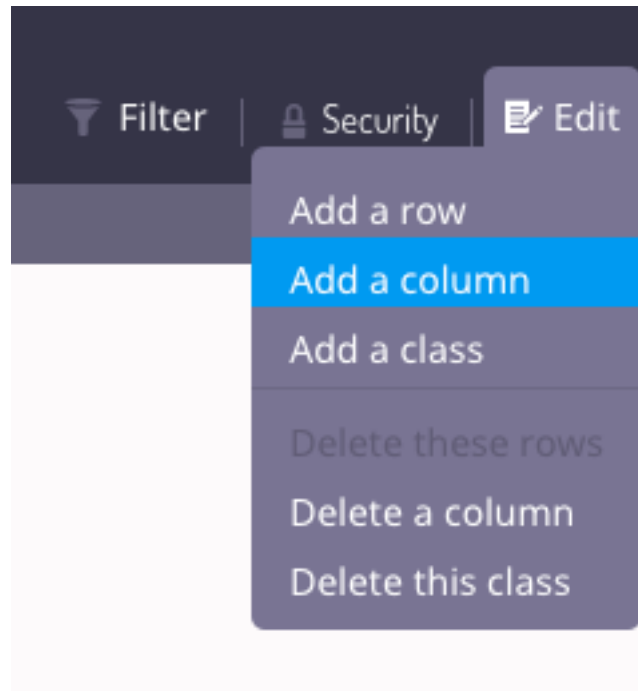
Follow these steps to create the Category Class and add rows:



1. Go into your Parse Dashboard from your app's panel in back4app
2. Click on the blue button to create a new Class



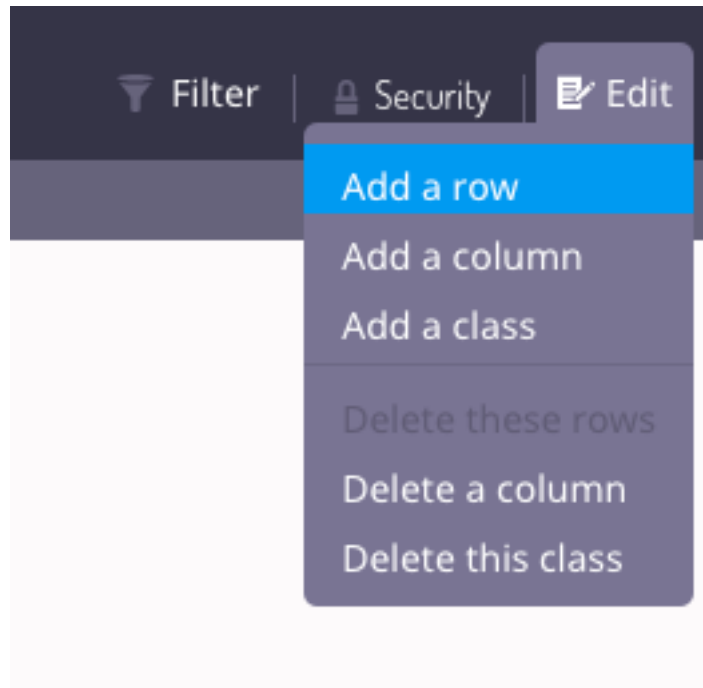
3. name it **Categories** (of type *Custom*)



4. Click **Edit** -> **Add a column** on the top-right corner of the screen

A screenshot of a 'Add a new column' dialog box. The dialog has a blue header with the title 'Add a new column' and a subtitle 'Store another type of data in this class.' in the top-left corner, and a three-dot menu icon in the top-right corner. The main content area is divided into two sections. The first section is titled 'What type of data do you want to store?' and has a dropdown menu showing 'String'. The second section is titled 'What should we call it?' with a subtitle 'Don't use any special characters, and start your name with a letter.' and has a text input field containing the word 'category'. At the bottom of the dialog, there are two buttons: 'Never mind, don't.' and 'Add column'.

5. name it **category** (of type *String*)



6. Click **Edit** -> **Add a Row** on the top-right corner of the screen

PARSE DASHBOARD 1.0.9

Readit

Core

Browser

Categories

Config

API Console

Push

CLASS Categories 7 objects

<input type="checkbox"/>	objectId String	updatedAt Date	createdAt Date	ACL ACL	category String	Add a new column
<input type="checkbox"/>	UFK04T8eYX	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	sport	
<input type="checkbox"/>	CJ5LnCYNrt	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	music	
<input type="checkbox"/>	1bLNA8hLds	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	books	
<input type="checkbox"/>	Lhua5FFUca	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	lifestyle	
<input type="checkbox"/>	53eKohszNe	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	food	
<input type="checkbox"/>	S7bWdUR9uQ	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	finance	
<input type="checkbox"/>	06UnHMKh5t	4 May 2016 at 07:...	4 May 2016 at 07:...	Public Read + Write	gaming	

7. Now type the category's name (lowercase) into the **category** field, and repeat step 6 for each category you want to add in your database.

Once you're done, you can run the app via XCode and test it out.

- Ad banners setup -

■ **AdMob Unit ID:** You must create a Banner Unit ID on www.apps.admob.com in order to get its Unit ID and replace the red string in **Configs.swift** with your own in order to make your AdMob banner work in the app.

```
let ADMOB_BANNER_UNIT_ID = "ca-app-pub-9733347540588953/7805958028"
```

■ **Apple Review process:** Sometimes it may happen that an app gets rejected once by Apple reviewers because of ads (iAd or AdMob doesn't matter), that's an issue we encountered too and we can't still understand why since we then re-submitted an app without any change and it got approved.

Anyway, check out the screenshot below, you'll get it after clicking on Submit For Review button on iTunes Connect, check the options shown below and everything should go smoothly.

Also, we suggest you to write something in the **Notes** field of your app's page on iTC telling the reviewer that "this app shows AdMob banners, test it for a while to see them showing up".

Advertising Identifier

Does this app use the Advertising Identifier (IDFA)? ☒ Yes ☐ No

The [Advertising Identifier \(IDFA\)](#) is a unique ID for each iOS device and is the only way to offer targeted ads. Users can choose to limit ad targeting on their iOS device.

If your app is using the Advertising Identifier, check your code—including any third-party code—before you submit it to make sure that your app uses the Advertising Identifier only for the purposes listed below and respects the Limit Ad Tracking setting. If you include third-party code in your app, you are responsible for the behavior of such code, so be sure to check with your third-party provider to confirm compliance with the usage limitations of the Advertising Identifier and the Limit Ad Tracking setting.

This app uses the Advertising Identifier to (select all that apply):

☒ Serve advertisements within the app

☐ Attribute this app installation to a previously served advertisement

☐ Attribute this app installation to a previously served advertisement

If you think you have another acceptable use for the Advertising Identifier, [contact us](#).

Limit Ad Tracking setting in iOS

☒ I, Anne Johnson, confirm that this app, and any third party that interfaces with this app, uses the Advertising Identifier checks and honors a user's Limit Ad Tracking setting in iOS and, when it is enabled by a user, this app does not use Advertising Identifier, and any information obtained through the use of the Advertising Identifier, in any way other than for "Limited Advertising Purposes" as defined in the [iOS Developer Program License Agreement](#).

Good luck with **Readit**, and don't forget to rate it on your Downloads page!

