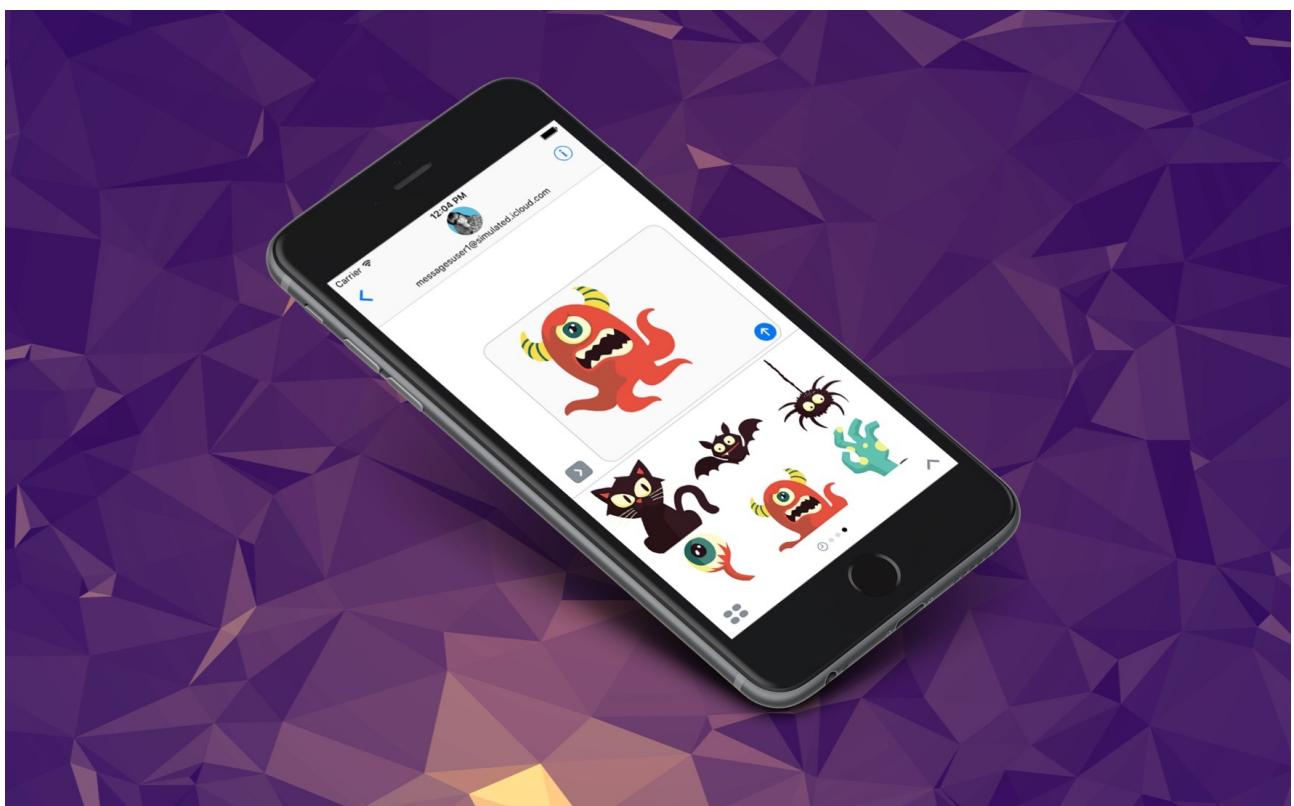


# Building a Simple Sticker App

One of the biggest announcements in WWDC 2016 is the introduction of the Message framework in iOS 10. Developers can now create app extensions for Apple's built-in Messages app. By building an app extension, you let users to interact with your app right in the Messages app. For example, you can build a message sticker extension that allows your users to send stickers while communicating with his/her friends in Messages. Or if you developed a photo editing app, for instance, you can write an extension for users to edit photos without leaving the Messages app.

The support of extension opens up a lot of opportunities for app developers. Apple even introduces App Store for iMessage, so you can sell your stickers and app extensions through the app store that is dedicated for iMessage.



To build an app extension for Messages, you will need to make use of the new Message framework. The framework supports two types of app extensions:

- Sticker packs
- iMessage apps

In this practice, we will focus on building a sticker pack.

Regardless of the type of app extension we're going to develop, we'll need to use Xcode 8, or higher.

Apple makes it very easy for everyone to build sticker packs. It doesn't need you to write a line of code. Follow this practice and learn how to create a sticker extension.

## Preparing the Sticker Images

Creating a sticker app is a two-part process:

First, we prepare the sticker images, that conforms to Apple's requirements. Secondly, we create a sticker app project using Xcode.

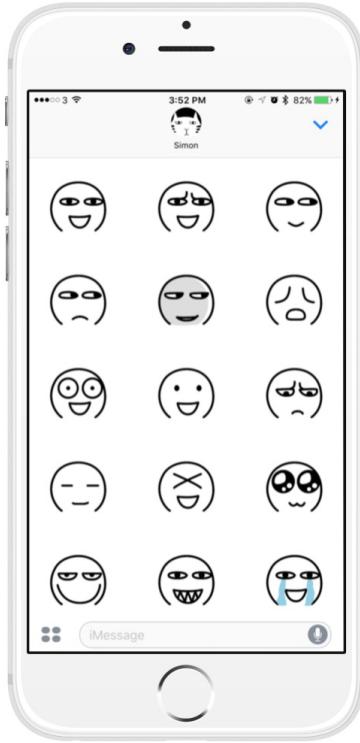
Let's start with the first part. Messages supports various sticker image formats including PNG, GIF and JPG, with a maximum size of 500KB. That said, it is recommended to use images in PNG format.

Sticker images are displayed in a grid based browser. Depending on the image size (small, regular or large), the browser presents the images in 2, 3 or 4 columns.

**Small Size**  
300px by 300px



**Regular Size**  
408px by 408px



**Large Size**  
618px by 618px



Other than size, the other thing you have to consider, while preparing your sticker images, is whether the images are static or animated. Messages supports both. For animated images, they should be either in GIF or APNG format.

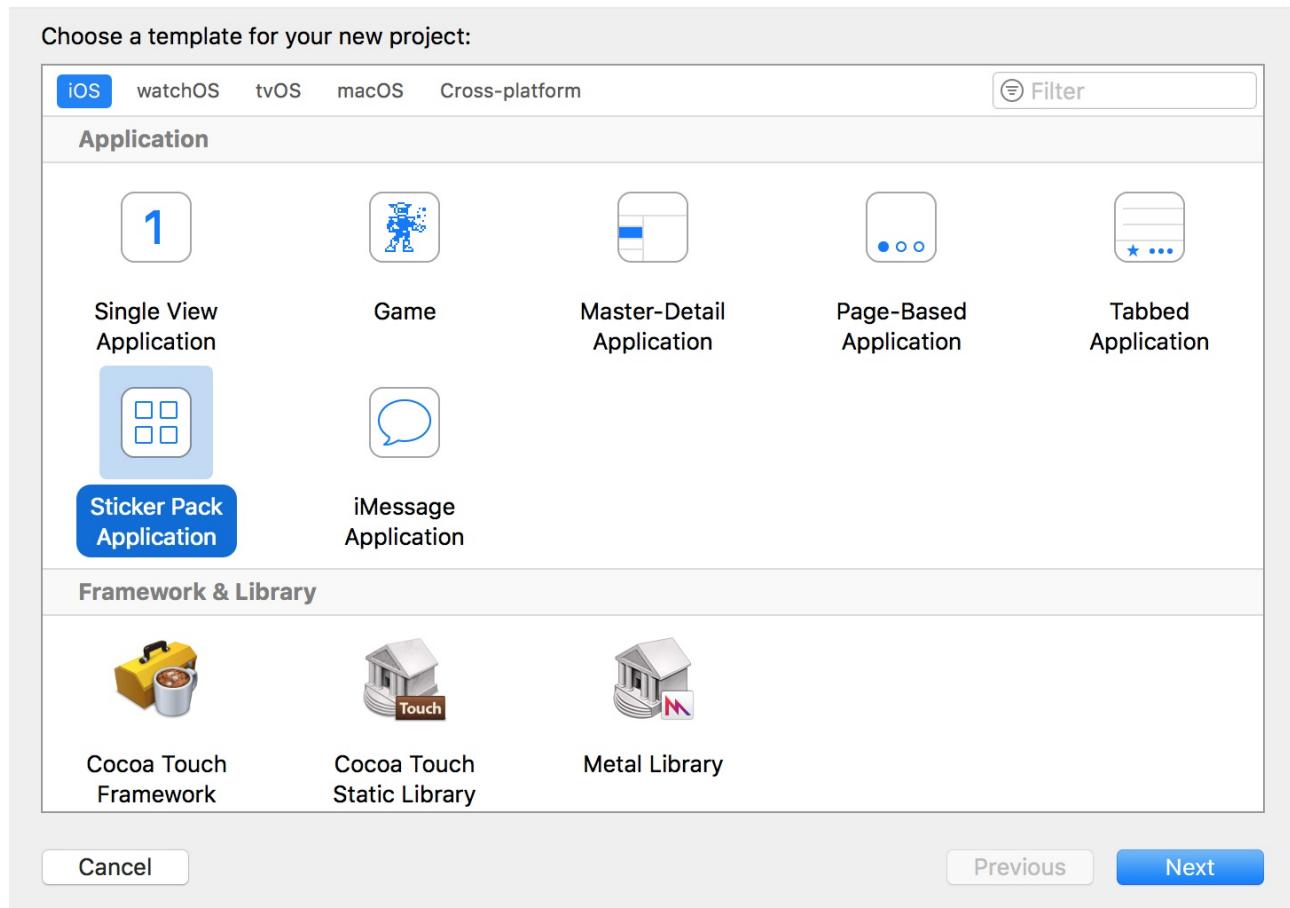
We will discuss more about animated sticker images in the later section. So let's focus on the static ones first. Now choose your own images and resize them to a size that best fits your stickers.

If you don't want to prepare your own images, you can download this image pack from:

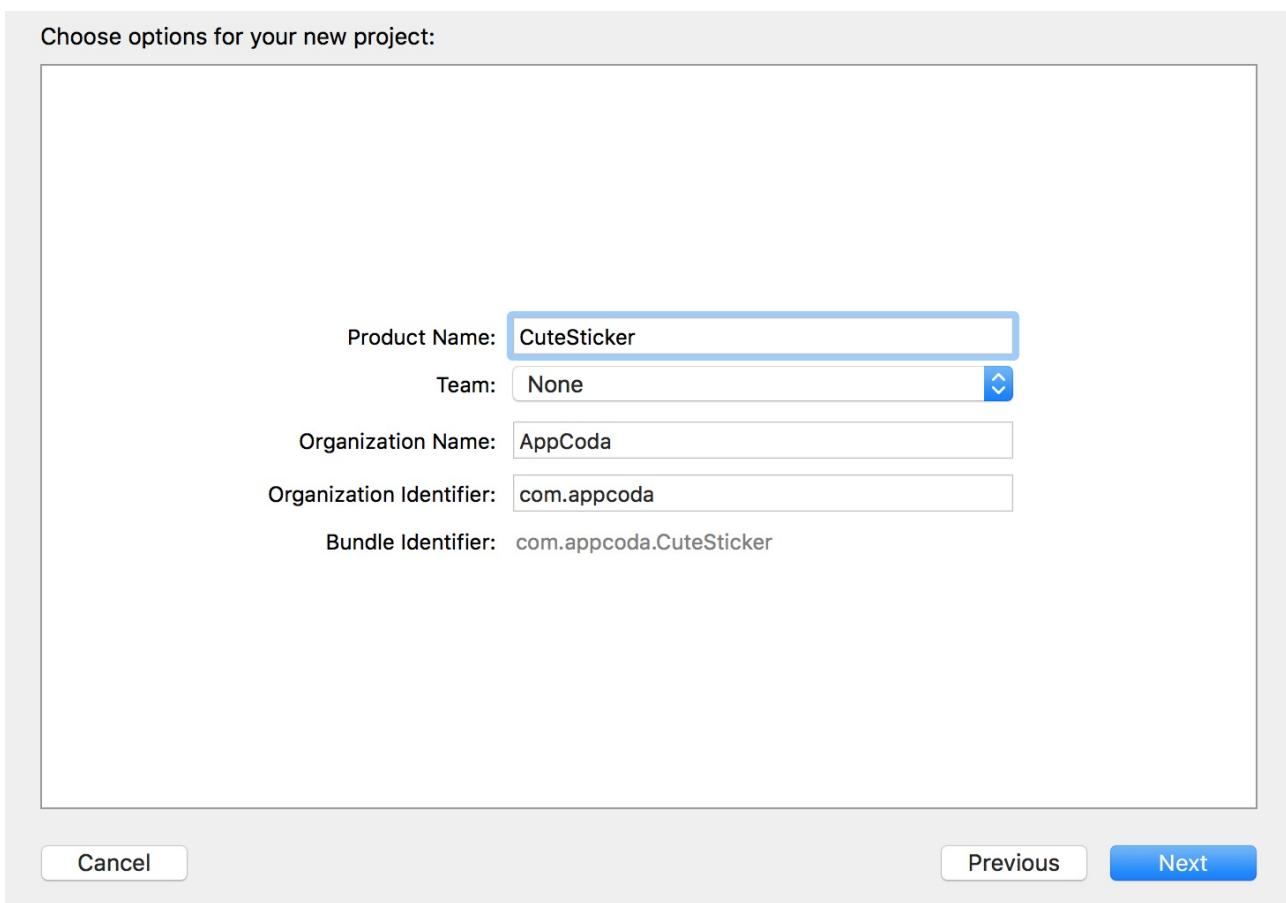
<http://www.appcoda.com/resources/swift3/StickerPack.zip>.

# Building the Sticker Package Project Using Xcode

Assuming your sticker images are ready, we're now going to build the sticker app. Fire up Xcode and create a new project. Xcode 8.\* has several project templates for building Messages extensions. For building sticker packs, please choose iOS > Application and then select Sticker Pack Application.



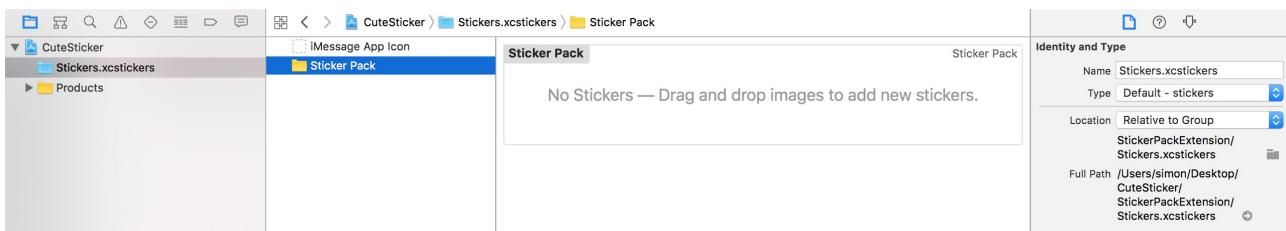
Next, fill in the project name. For this demo, we use the name CuteSticker but you can choose whatever name you prefer.



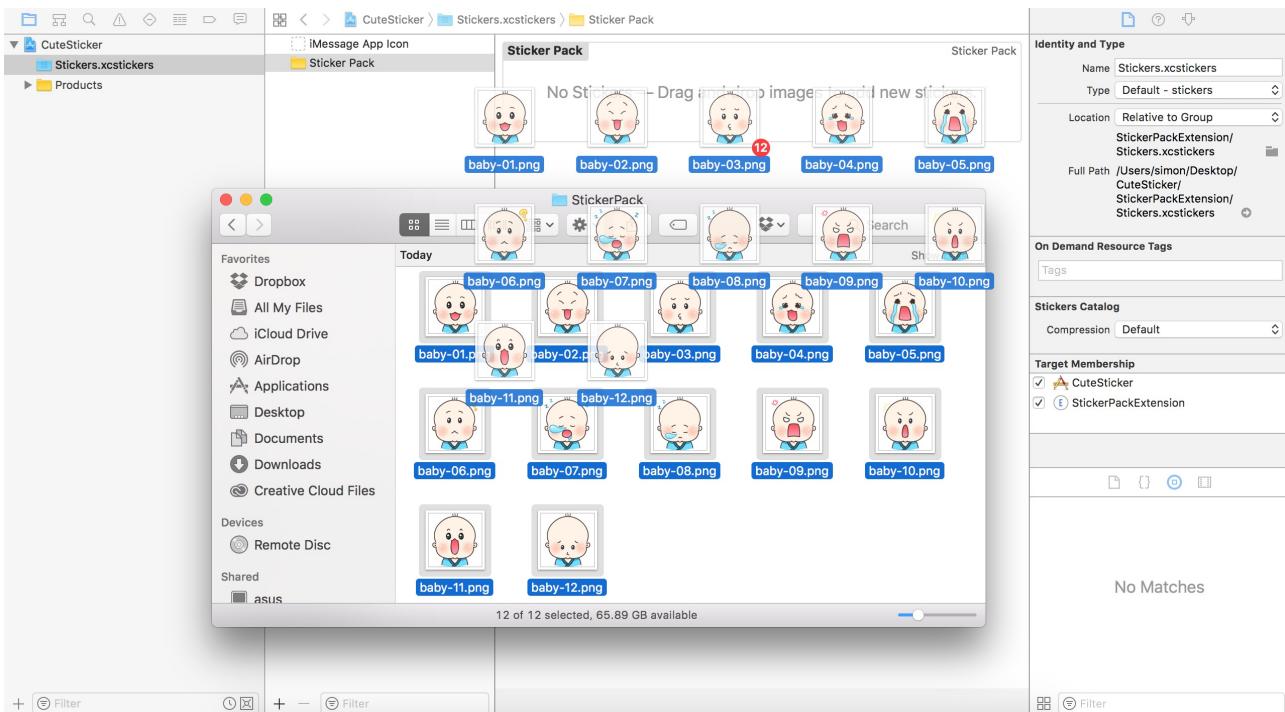
Click Next to continue and choose a folder to save your project. Xcode then generates a sticker app project for you.

## Adding Images to the Sticker Pack

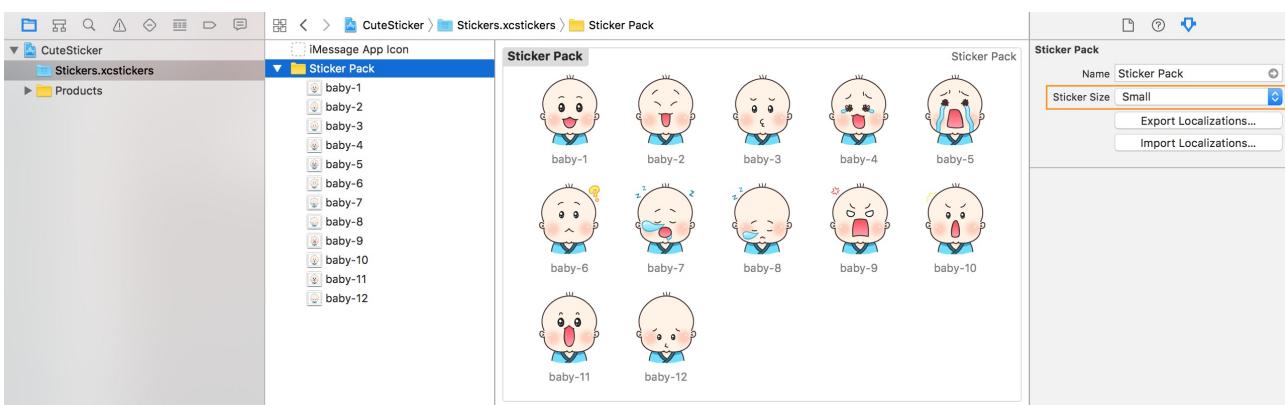
Once your Xcode project is created, you'll see two folders in the project navigator. Click `Stickers.xcstickers` and then select the `Sticker Pack` folder. This is where you put your image files.



Assuming you've downloaded the image pack, unzip it in Finder. Then select all the images and drag them into the Sticker Pack folder.



It's almost done. Now select the Sticker Pack folder, and then choose the Attributes inspector. By default, the sticker size is set to Medium. For the demo images, the size is 300px by 300px. So the best choice is to set the sticker size to Small, though you can still keep it to Medium if you want.



## Adding App Icons

Lastly, your sticker pack must have an app icon. You can download the app icon from:

<http://www.appcoda.com/resources/swift3/StickerAppIcon.zip>.

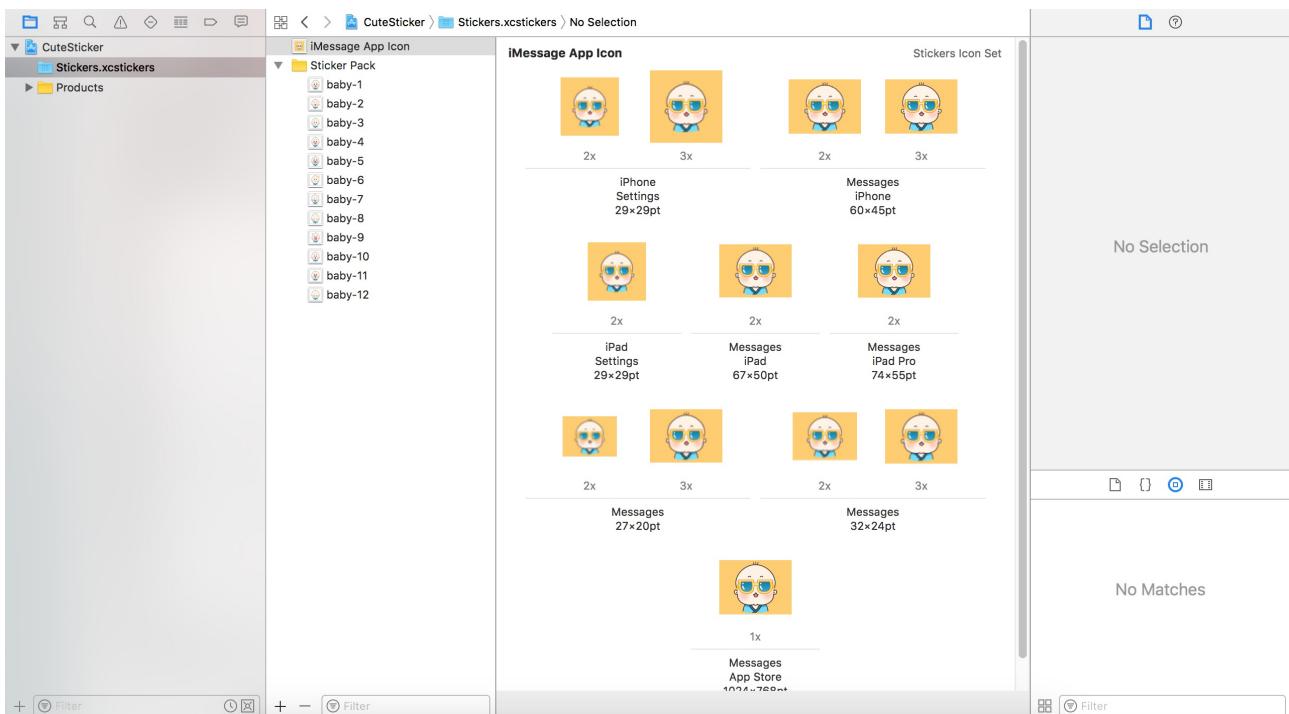
If you prefer to create your own icon, make sure you prepare the icon with different sizes:

- 1024×768 points (@1x) for Messages App Store
- 27×20 points (@1x, @2x, @3x) for Messages
- 32×24 points (@1x, @2x, @3x) for Messages
- 29×29 points (@1x, @2x, @3x) for iPhone/iPad Settings
- 60×45 points (@2x, @3x) for Messages (iPhone)
- 67×50 points (@1x, @2x) for Messages (iPad)
- 74×55 points (@2x) for Message (iPad Pro)

To simplify the icon preparation, you can download iMessage App Icon template:

(<https://developer.apple.com/ios/human-interface-guidelines/resources/>) from Apple.

After you download our demo app icon pack, unzip the file and drag all the app icon files to iMessage App Icon.



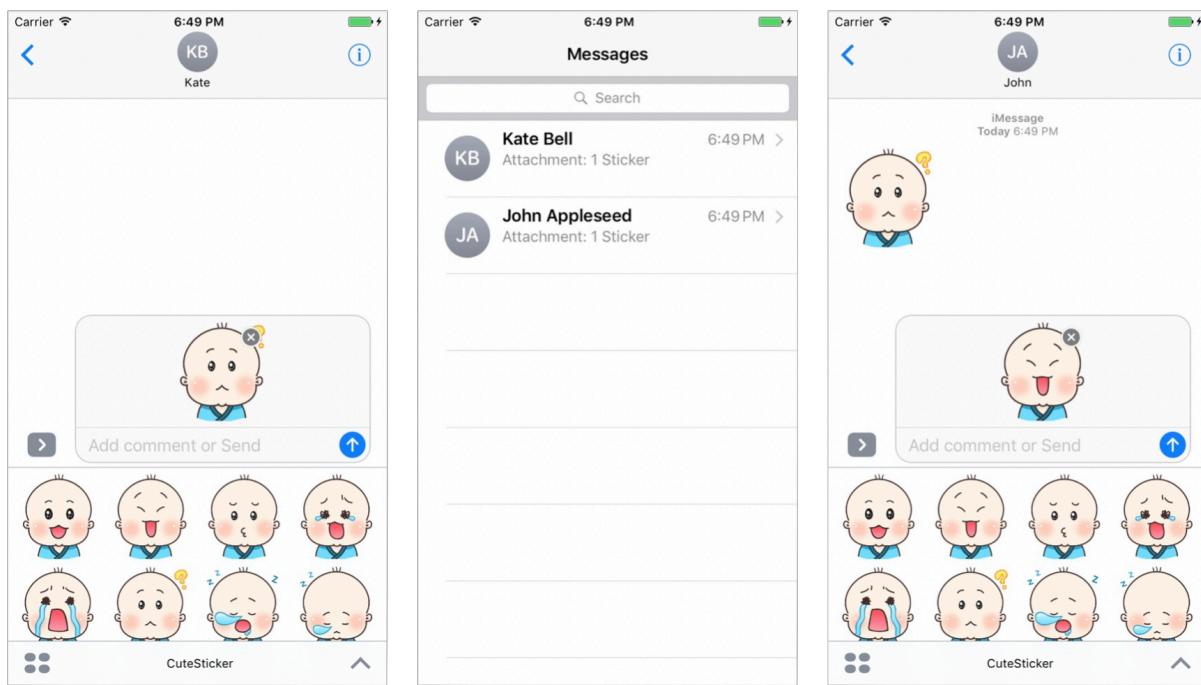
## Testing the Sticker Pack

Now that you've created a sticker pack for Messages, it's time to test it. You do not need an iOS 10 device to run the test. Xcode 8 has a built-in simulator to test any iMessage app extension. Choose a simulated device (e.g. iPhone 7) and hit the Run button to start the testing.

Since the sticker pack is an app extension, you can't execute it as a standalone application. When you run the sticker pack, Xcode loads the sticker pack into the Messages app and automatically launches it on the simulator. In case if you don't see the sticker pack, click the lower left button (i.e. App Shelf button) to reveal the stickers.



The Messages app in the simulator has come with two simulated users: Kate Bell and John Appleseed. The default user is set to Kate. To send a sticker to John, choose a sticker from the message browser and press return key to send it. Then go to John. You should find the stickers you've just sent.

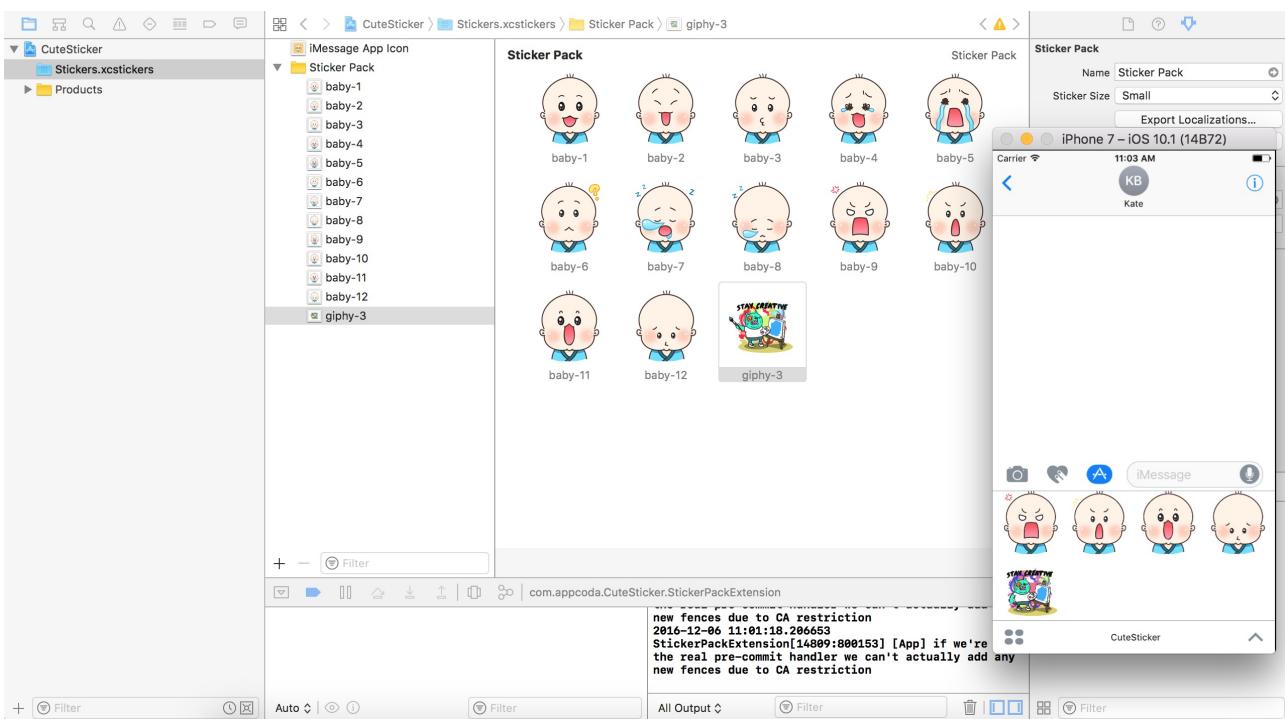


## Enhancing Your Sticker Pack with Animated Images

As mentioned at the beginning, the Messages app supports animated sticker images. If you already have some animated GIFs or APNGs, simply add the image to the stick pack. Xcode will recognize it and display the animation. As an example, you can download this free image:

(<http://giphy.com/gifs/cartoon-painting-creative-3oEduXdm2gnrsJBOo>),

and add it to the sticker pack to have a test.

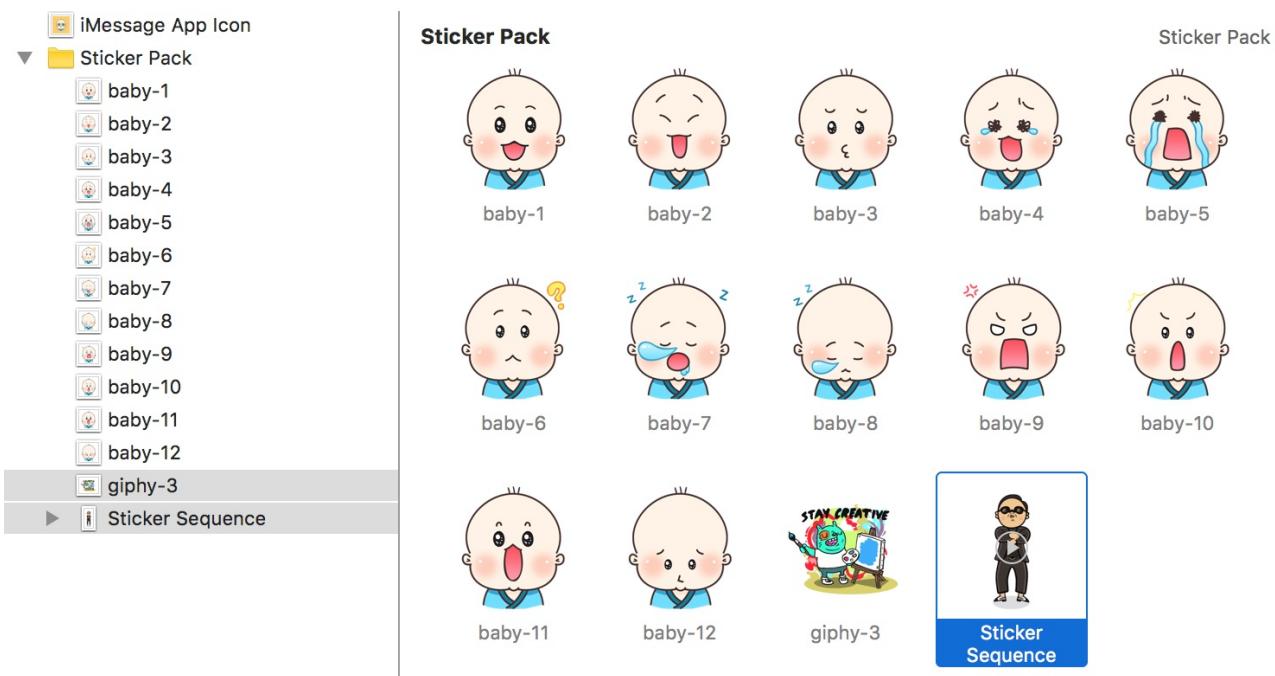


An alternative approach for creating animated images is to create a sticker sequence. Go back to your sticker pack. Right click on any blank area to bring up the option menu. Choose Add Assets > New Sticker Sequence. This creates a sticker sequence for you to add a sequence of images.

As an example, you can download this image pack:

(<http://www.appcoda.com/resources/swift3/StickerAnimatedImages.zip>)

to try it out. Unzip the pack, and drag all the images to the sticker sequence. Xcode allows you to preview the resulting animation right in the sticker pack. Place the cursor over the sticker sequence and click the Play button to preview the animation.

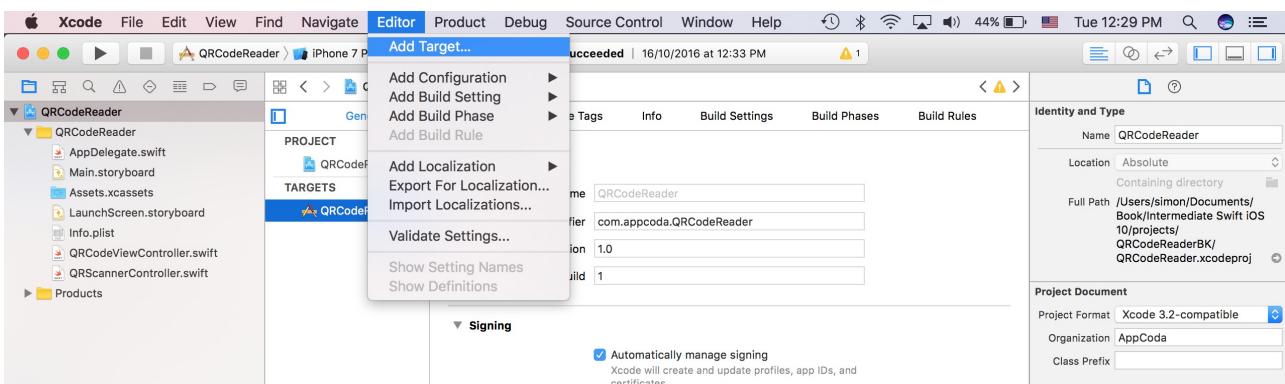


You can run the sticker pack in the simulator again. The Messages app will display both images as an animation.

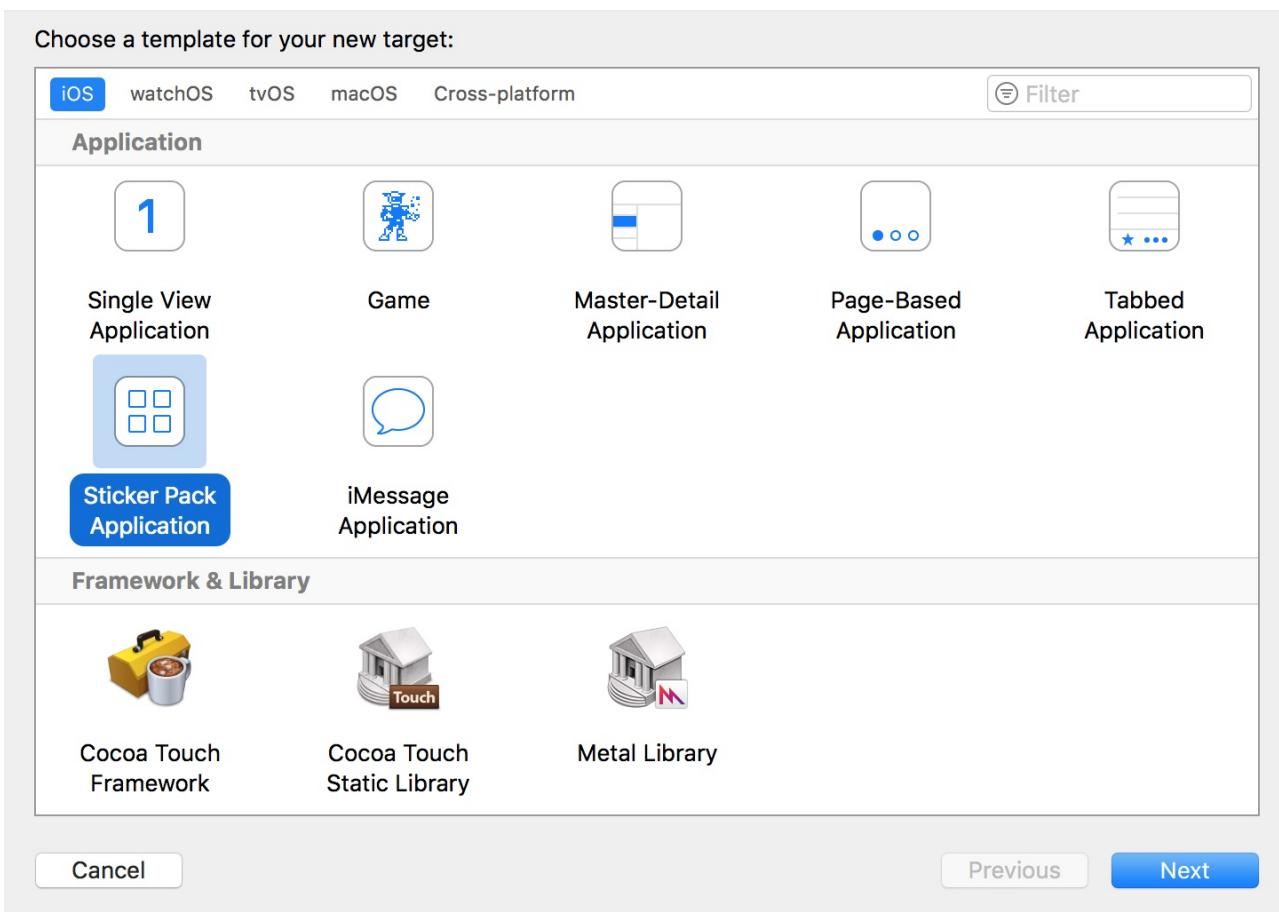
## Adding a Sticker Pack for an Existing App

Up till now, we've learned how to create an independent sticker app. What if we already have an existing app and want to bundle a sticker pack? How can we do that?

Xcode 8.\* lets us build a sticker extension for any existing apps. Assuming you've opened an existing project (CarouselApp) in Xcode, you can first select your project in the project navigator, and then go up to the menu. Select Editor > Add Target....

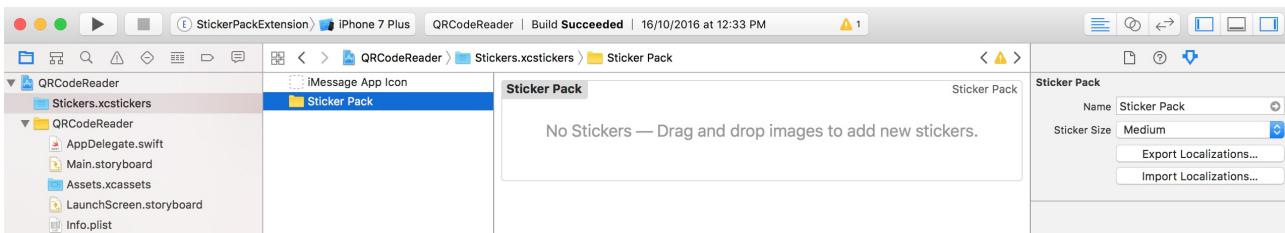


You'll then be prompted to choose a project template. Again, pick the Sticker Pack Application template to proceed.



Next, follow on-screen instructions to give your product a name. This is the name of your sticker pack that will be shown in

Messages. Finally, hit the Activate button when Xcode prompts you to activate the new scheme. Xcode will add the Stickers.xcstickers folder in your existing project. All you need to do is drag your sticker images into the sticker pack.



To test the sticker app, you can choose the StickerPackExtension scheme and then run the app in any of the simulators.

## Summary

We have just learned how to create an app extension for the Messages app in Xcode 8.\*. As you see, we don't even need to write a line of code to create a sticker pack. All we need is prepare our own images (animated or static) and we're ready to build a sticker pack.

At the time of writing this practice tutorial, the Message App Store has been launched for several months. But it is still a good time to start building your own sticker packs, especially you have an existing apps or some iconic characters for your brand. Having a sticker pack on the Message App Store will definitely give your app more exposure.

Sticker pack is just one type of the iMessage app extensions.

\*\*\*