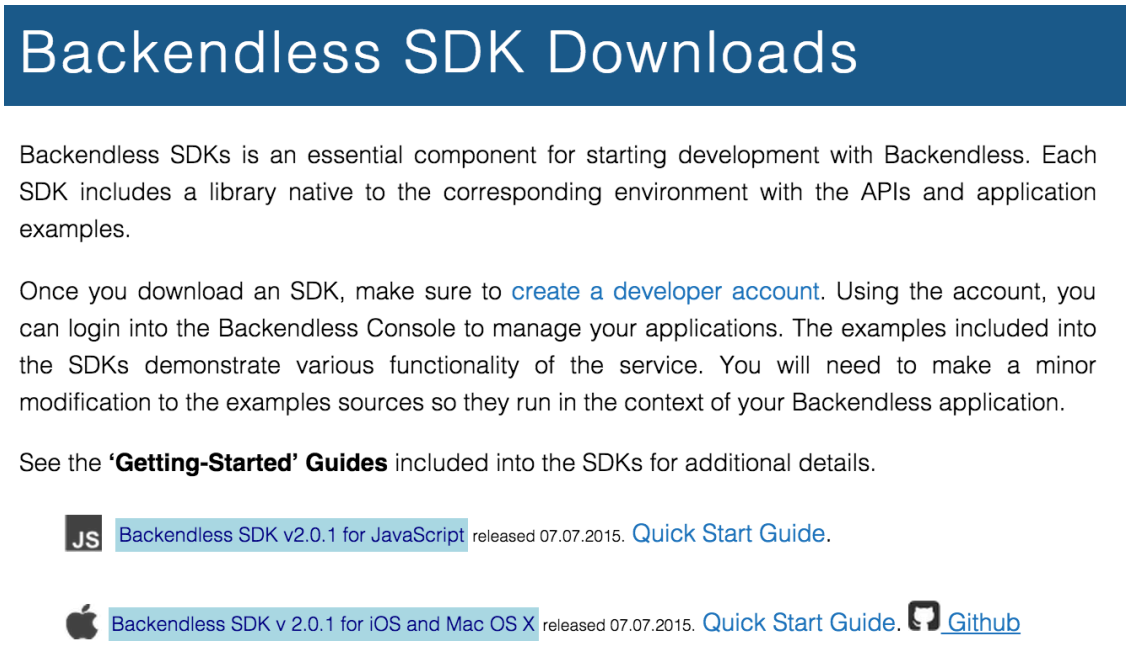
Getting Backendless SDK

1. Login to your Backendless account or register to create a new one.
2. Download Backendless SDK for iOS from the Backendless SDK Downloads page and unzip it.



Getting Started

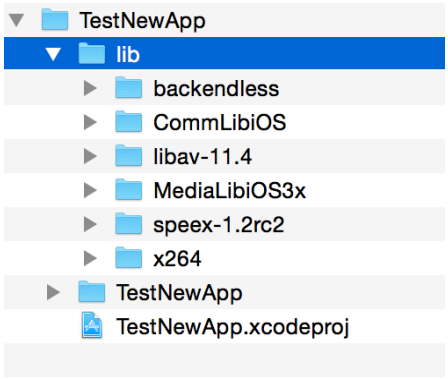
1. Start up Xcode and go to File -> New -> Project. Select iOS -> Application->Single View Application", and click Next.
2. Enter your application name for the Product Name, set the Language to Objective-C or Swift, and click Next.
3. Choose a directory to save your project, and click Create.

Let's see what Xcode has built for you. In the upper left corner of Xcode, select the iPhone 6 Simulator and click Play to test your app. You should see a blank white screen appear. Xcode has created a single blank screen in your app.

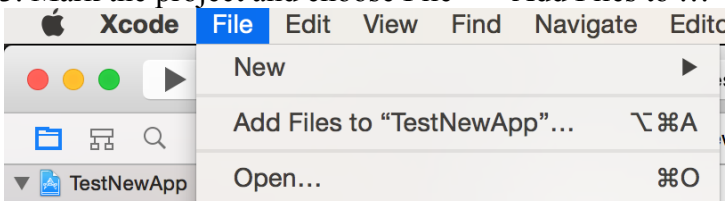
Add the Libraries and Frameworks to the project

1. Choose the project target, go to Build Phases->Link Binary
With Libraries, push "+", check the following iOS frameworks and libraries:
- *SystemConfiguration.framework*
- *libsqlite3.tbd*
- *libz.tbd*
Push "Add" button.

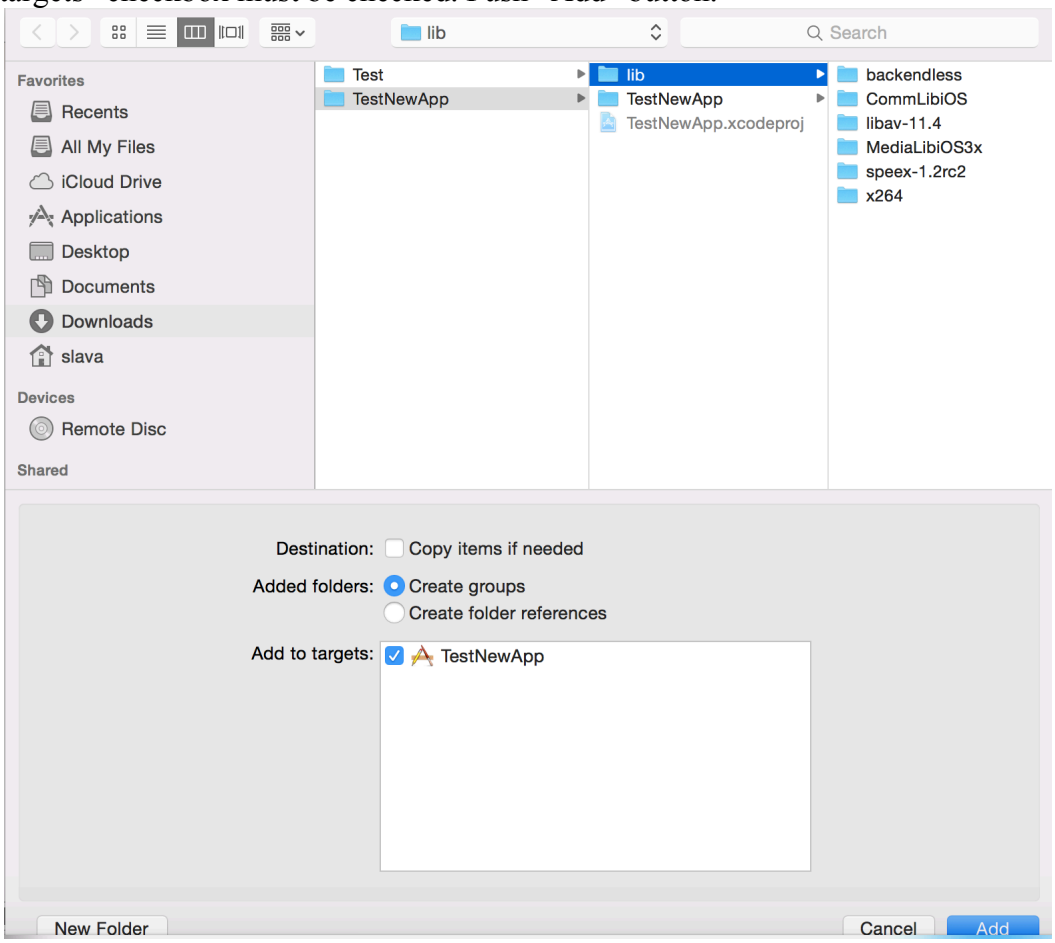
2. Add the folder “lib” from Backendless SDK kit to the your project folder:



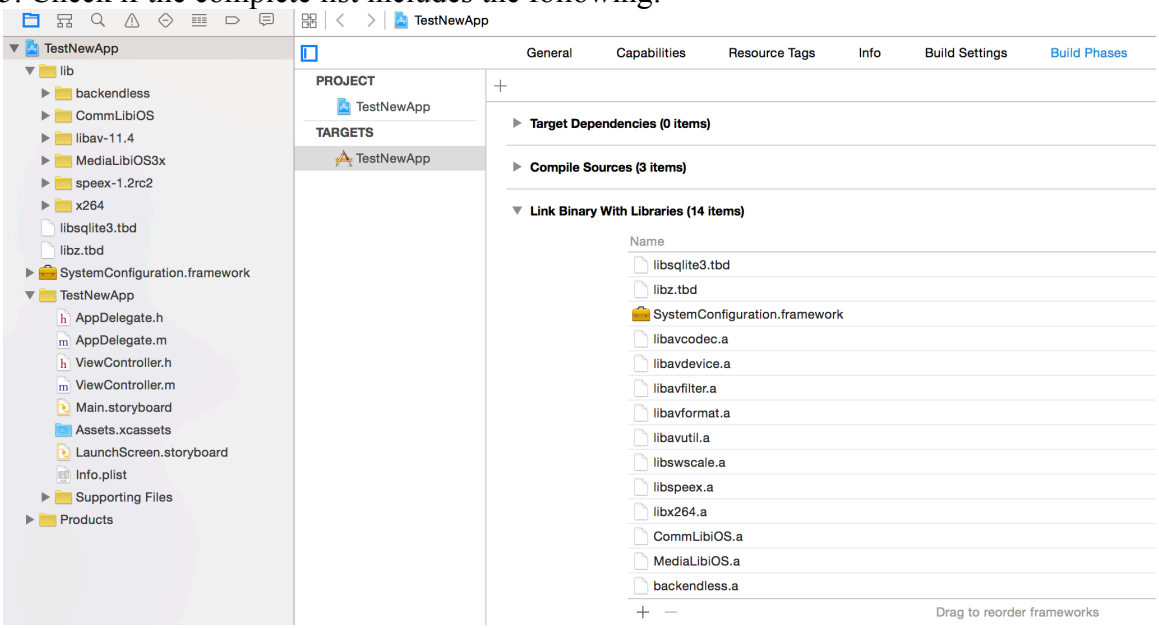
3. Mark the project and choose File- > ”Add Files to …” menu item:



4. In window choose the “lib” folder in the project folder. Make sure that the “Add to targets” checkbox must be checked. Push “Add” button.



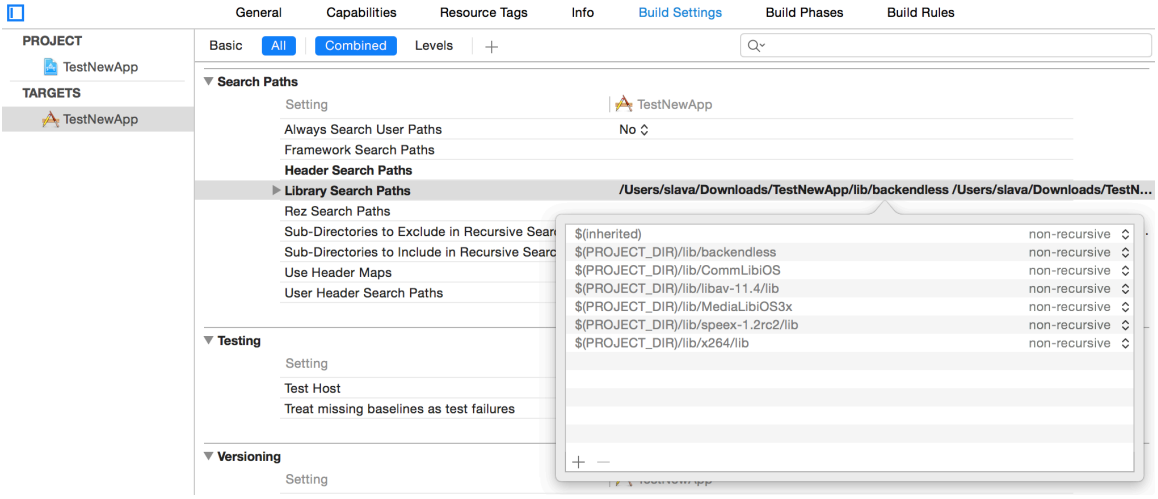
5. Check if the complete list includes the following:



6. Add the following option to the Build Settings -> Search Paths -> Library Search Paths line:

$(inherited) $(PROJECT_DIR)/lib/backendless $(PROJECT_DIR)/lib/CommLibiOS $(PROJECT_DIR)/lib/libav-11.4/lib $(PROJECT_DIR)/lib/MediaLibiOS3x $(PROJECT_DIR)/lib/speex-1.2rc2/lib $(PROJECT_DIR)/lib/x264/lib

Make a double click on this line and check it:



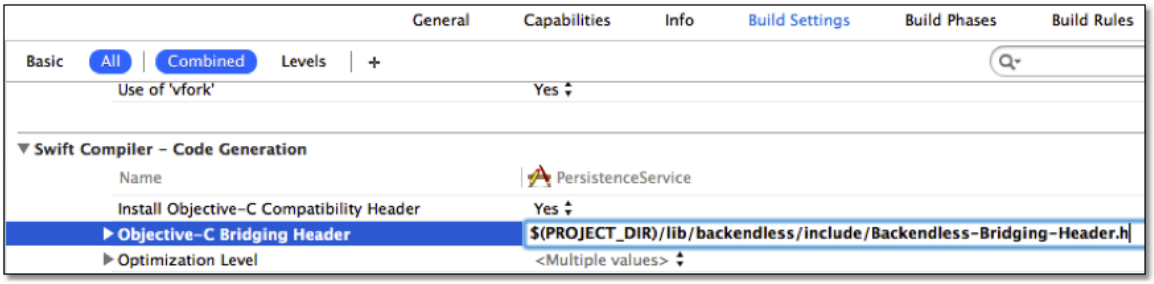
7. If the project uses Objective-C, add the following import statement to the application's delegate `AppDelegate.m` file:

```
#import "Backendless.h"
```

If you are going to use Backendless MediaService, also add:

```
#import "MediaService.h"
```

8. If the project uses Swift, add the following option to the Build Settings:



If you are going to use Backendless MediaService, you have to set this options with: $(PROJECT_DIR)/lib/backendless/include/Backendless-With-Media-Bridging-Header.h

Use CocoaPods

Another way to add frameworks and libraries to your Backendless application is CocoaPods service.

CocoaPods manages library dependencies for your Xcode projects.

The dependencies for your projects are specified in a single text file called a Podfile. CocoaPods will resolve dependencies between libraries, fetch the resulting source code, then link it together in an Xcode workspace to build your project.

Make sure you have the Cocoapods ruby gem installed your system. If you don't please follow the directions at CocoaPods Getting Started , or just fire up a Terminal window and run $ sudo gem install cocoapods .

Creating a new Xcode project with CocoaPods

To create a new project with CocoaPods, follow these simple steps:

1. Create a new project in Xcode as you would normally, then close this project.
2. Open a Terminal window, and $ cd into your project directory.
3. Create a Podfile. This can be done by running $ touch Podfile.
4. Open your Podfile using your favorite text editor (or Xcode), and add a text that looks like this:

```
platform :ios, '9.0'
pod 'Backendless-ios-SDK', '~>2.0.0'
```

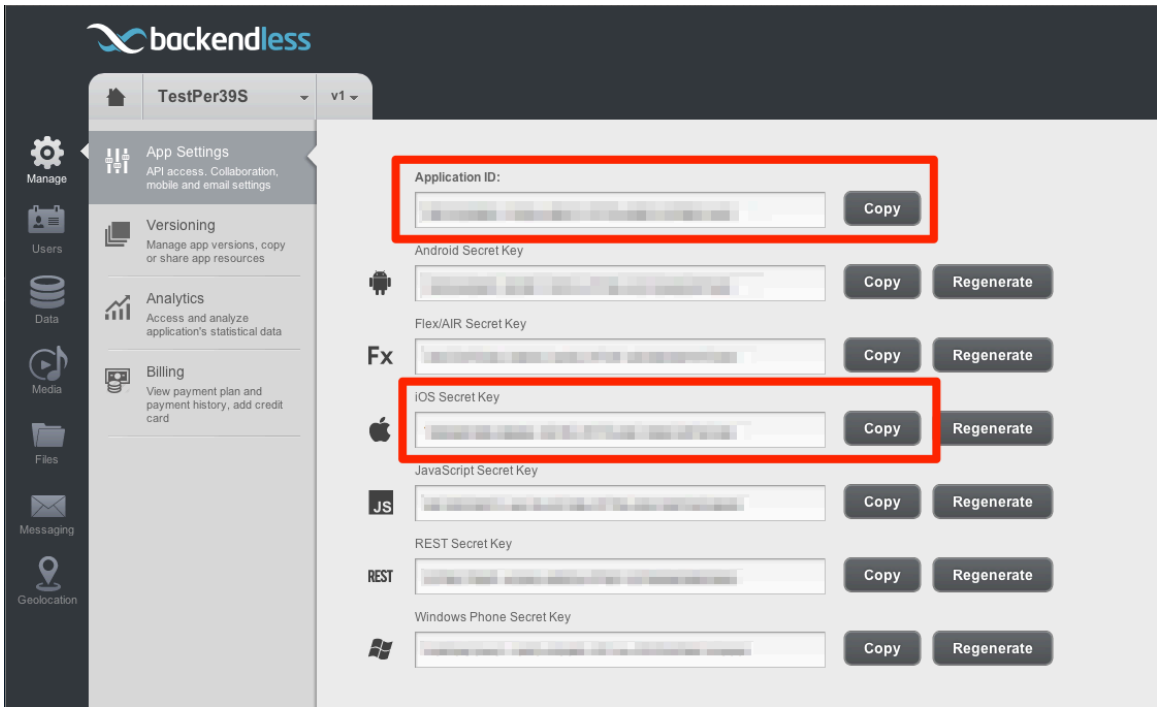
The first line specifies the platform and supported version, the second line specifies the name of Backendless folder in CocoaPods Specs repository and SDK version which you need.

5. Save Podfile, return to Terminal window and run $ pod install. Once all of the pod data is downloaded, Xcode project workspace file will be created. This should be the file you use everyday to create your app.

6. Open *.xcworkspace* file to launch your project, and build it using scheme for iOS device.

Add Backendless Application Id & Secret Key

1. Get your Backendless application and secret keys for iOS from the Backendless Console. The keys can be found on the Manage -> App Settings section:



2. Add Backendless application initialization code block in `AppDelegate.m`:

```
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

    [backendless initApp:APP_ID secret:SECRET_KEY version:VERSION_NUM];
    // if you don't need the MediaService - you must remove the next line
    backendless.mediaService = [MediaService new];

    return YES;
}
```

or in AppDelegate.swift:

```
@UIApplicationMain
class AppDelegate: UIResponder, UIApplicationDelegate {

       let APP_ID = "-YOUR-APPLICATION-ID-”
       let SECRET_KEY = "-YOUR-APPLICATION-IOS-SECRET-KEY-"
       let VERSION_NUM = "v1"

       var backendless = Backendless.sharedInstance()

       var window: UIWindow?

func application(application: UIApplication, didFinishLaunchingWithOptions
launchOptions: [NSObject: AnyObject]?) -> Bool {

       backendless.initApp(APP_ID, secret:SECRET_KEY, version:VERSION_NUM)
       // if you don’t need the MediaService – you must remove the next line
       backendless.mediaService = MediaService()

       return true
       }
. . .
}
```