Getting Started with the iOS Sample Application

Product Version 3.3.0.14

Published: 06-Nov-2013 19:20

Table of Contents

Introduction

The Mobile Client Software Development Kit (SDK) provides a sample application that demonstrates Gracenote features:

- **GN_Music_SDK_iOS**

The SDK also provides a development project that is an example of how to incorporate the Mobile Client into your iPhone application. This document describes how to integrate the Sample Application project into your development environment.

Song Metadata Cache

Gracenote provides a local cache of fingerprints and metadata of some example songs. The SDK uses this data to attempt a local ID prior to attempting an online lookup. The SDK provides song samples in the `sample_music` folder that you can use to test local lookup identification. For additional samples, contact your Gracenote Professional Services representative.

Preparing Your iPhone Development Environment

iOS development requires a specific environment. For details on setting up this environment, refer to the instructions on the Apple iOS Dev Center: http://developer.apple.com/devcenter/ios/
You also must register as an Apple Developer to download the software required to build and run the Gracenote iOS Mobile Client sample application. To register, go to http://developer.apple.com/programs/register/

Requirements

The following are required to build and run the sample application:

General Requirements:

- Apple Developer Provisioning Certificate
- Apple App ID
- Apple Provisioning File

For Macintosh OS X Version 10.7 (Lion):

- Xcode Version 4.x
- iOS SDK Version 4.x

For Macintosh OS X Version 10.6 (Snow Leopard):

- Xcode Version 3.x and higher (Does **NOT** apply to the GN_Mobile_Client sample app, which requires Xcode 4.x or higher)
- iOS SDK Version 4.x

Installing the iOS SDK

Mobile Client Sample Application uses the iOS v4.2.3 and XCode version 4.3. Ensure your development environment supports these versions.

Follow the steps below to install the appropriate iOS SDK:

1. Go to the iOS Dev Center: http://developer.apple.com/devcenter/ios/
2. Download the iOS SDK compatible for your development environment OS X version (see *Requirements (see page 3)*). The download will also include the required version of Xcode.
3. Ensure Xcode and iPhone simulator are closed.
4. Run the downloaded iOS package (**Xcode** and **iOS SDK.mpkg**). This launches the iOS SDK installer. Once the installer has completed, tthe iOS SDK will be available as an Active SDK in Xcode.

Running the Sample Applications

To run either of the sample applications:

1. Unpack the Gracenote iOS Mobile Client Package: **GN_Music_SDK_3_x_x_x.zip**
2. Launch Xcode.
3. Close the **Welcome to Xcode** launch screen.
4. Choose **File** > **Open**.
5. Navigate to **GN_Music_SDK_iOS**.
6. To run the sample application, add the ClientID-Client Tag string provided by Gracenote Professional Services to the project.
    1. For GN_Music_SDK_iOS, edit the file **GN_Music_SDK_iOS_Prefix.pch** file under **Other Sources**.

       ```
       // For GN_Music_SDK_iOS, edit GN_Music_SDK_iOS_Prefix.pch

       #define CLIENTID @"12345678-ABCDEFGHIJKLMNOPQRSTUVWXYZ012345"
       ```

7. Select the **Overview** drop down box at the top left corner of the application window.
8. In the first grouping, select **Simulator**.
9. Leave all other settings as is and click outside of the drop down box to exit it.
10. In the top middle of the window, click the **Build and Run** icon.

11. Look for **Succeeded** in the the bottom right of the window. If the build fails, see *Troubleshooting* (see page 6).

12. The Simulator will open and the Gracenote iOS Sample Application will launch.

Testing the Sample Applications

To test the **GN_Music_SDK_iOS** sample application:

1. Tap on the buttons to perform the available queries. When testing the application, note that:
    - The **Recognize From > FPX** and **Recognize From > PCM** queries use inputs from the **GN_Music_SDK_iOS** parent directory.
    - The **Text Search** and **Lyric Fragment Search** queries require you to manually input search terms.

2. Tap the **ID Now** button on the device to identify a song. You should see a screen similar to the following:
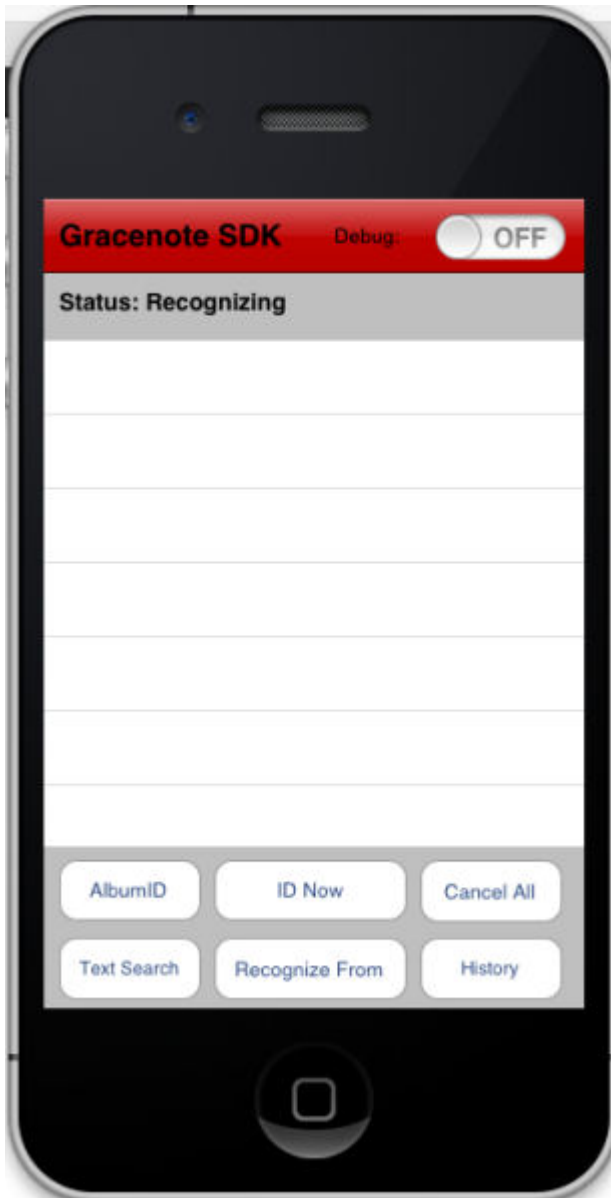
> ⚠️ In the Sample Application, if debugging is enabled through the UI, logging will take place for everything except idNow operations. However, you will not see this issue if you enable debugging programmatically, See *Using Mobile Client Debug Logging* in the *Mobile Client iOS Implementation Guide* for more information.

Troubleshooting

There are a few common reasons that the sample application build may fail:

Error Message: No architectures to compile for (ARCHS-i386 ppc, VALID_ARCHS=arm6 arm7)

Issue: Your development environment does not have the iOS SDK specified by the GN_Music_SDK_iOS project.

Potential Fix: The Gracenote iOS Sample Application project was created using iOS 4.2 SDK. If you have a newer version, change the Base SDK for the GN_Music_SDK_iOS target:

1. Click on the blue **Info** button at the top middle of the screen near **Build and Run** icon.
2. Select the **Build** tab, if it is not already selected.
3. In the Architectures section, select the Base SDK of the 4.x version that is installed on your system.

Error Message: Base SDK Missing in the Overview Dropdown.

Issue: This error normally occurs after a new SDK version and Xcode are installed. A base SDK needs to be designated.

Potential Fix: Follow the steps above to specify the Base SDK version.

Error Message: CodeSign error: code signing is required for product type 'Application' in SDK 'Device iOS x.x'

Issue: Xcode is attempting to deploy the application to an iPhone device unsuccessfully.

Potential Fix 1: If you are trying to use the simulator, change the project's Active SDK by selecting **Simulator** from the first grouping in the **Overview** drop down box in the top left hand corner.

Potential Fix 2: If you are trying to use a device, follow the steps listed in the **Devices** section of the iOS Provisioning Portal to ensure your device is correctly provisioned.