

知りたい! を刺激する総合専門サイト

【コラム】

## 実践! iPhoneアプリ開発

### 1 カメラアプリの作り方(1) - イメージピッカーのためのアクションシートを作る

木下誠 2009/01/26

#### App Storeに登録できるiPhoneアプリを作るために

iPhoneが登場して半年が経った。iPhoneの魅力は、デバイスとしての機能だけでなく、同時に登場したApp Storeにもあるだろう。日々新しいiPhone向けアプリケーションが登場し、その数は1万5000本を突破した。

iPhone向けアプリを作成してApp Storeに登録してみたい、と考える開発者は多くいるだろう。だが、開発に関する情報は、特に日本語のものは、絶望的に少ないのが現状だ。アプリ開発の最初の一步の解説を行う書籍は、ようやく登場し始めた。だが、そこから先の、実際にユーザに使ってもらうためのアプリを作ろうとなると、もっと実践的な情報が欲しくなる。

そこで、具体的なiPhoneアプリの開発を解説する連載を始めたいと思う。この連載では、カメラアプリや、RSSリーダアプリといった、本当に使い物になるアプリを作りながら、必要となる知識を解説していく。ここで紹介するアプリをベースにして機能を拡張すれば、App Storeに登録するものを作り上げる事もできるだろう。

対象となる読者としては、すでにiPhoneアプリ開発の基礎を理解している人を想定している。プログラミング言語であるObjective-Cや、開発環境であるXcode、基礎的なフレームワークであるFoundationの開発などは、省略する。これらは、書籍などを使って学んでほしい<sup>※1</sup>。また、ここで公開するソースコードの著作権は筆者に属するが、それを明記すればいかに再活用しても構わない。

※1 本誌特集『iPhoneアプリケーション開発入門』や、連載『ダイナミックObjective-C』でもiPhoneアプリの開発方法やObjective-Cについて解説している。参考にしてほしい。

では、早速始めよう。最初に取り上げるのは、カメラアプリの作り方だ。

#### カメラアプリを作る

カメラアプリは、iPhoneのカメラを使って写真を取り、その画像に様々なエフェクトを加えたりするアプリだ。iPhoneアプリでも人気分野の1つで、App Storeに多くの種類のカメラアプリが登録されている。

アプリの仕様を考えてみよう。まずアプリにツールバーを配置して、カメラボタンを用意し

ておく。このボタンを押して写真を撮ればいいのだが、カメラを使って目の前の写真を撮りたい事もあるだろう。そこで、カメラボタンを押したらアクションシートを表示し、カメラを使うか、画像を読み込むかを選択できるようにしよう。

カメラを選択した場合は、写真を撮るモードになる。画像読み込みの場合は、リストから画像を選択する。そうやって画像が取得できたら、それをアプリで表示する。さらに、エフェクトをかけて、標準のカメラでは得られない画像を作り出す。

このようなアプリを作ってみよう。

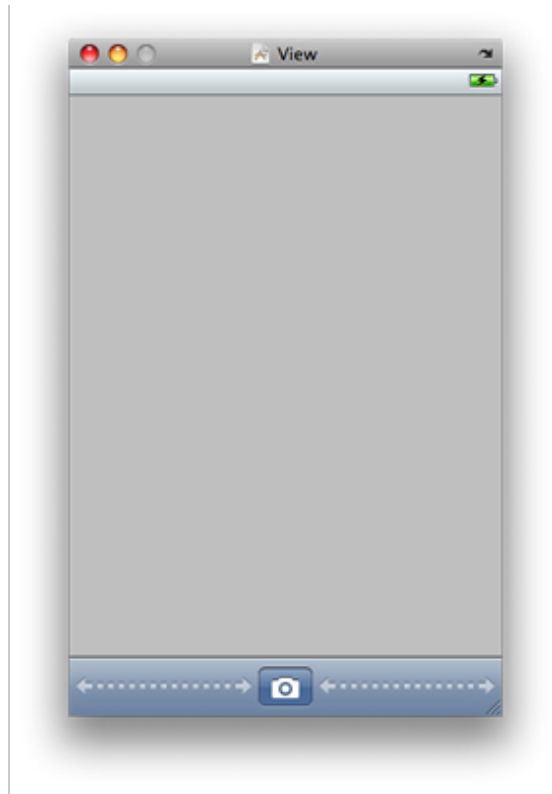
## アクションシートの表示

まず、Xcodeを起動して、新規プロジェクトを作ろう。プロジェクトのテンプレートは、今回はView-Based Applicationを選択した。プロジェクト名は「Camera」としておく。アプリケーションコントローラと、ビューコントローラが1つずつ作られる事になる。主にいじるのは、ビューコントローラの方だ。

アプリにツールバーを追加しよう。CameraViewController.xibファイルを開き、Interface Builderに移動する。このビューに、ライブラリパレットからツールバーをドラッグ&ドロップで追加する。

このバーに、カメラボタンを作ろう。バーボタンアイテムを追加して、インスペクタパネルを使い、Identifierとして「Camera」を指定する。これで、カメラのアイコンが表示されるはずだ。さらに、このボタンを真ん中に配置しておこう。両脇に、フレキシブルな空白を挿入する事で、真ん中に持っていく事ができる。

---



次に、このボタンを押したときに呼び出される、アクションを追加する。カメラで撮影するか、画像を取り込むかを選択するアクションだ。この選択は、アクションシートを使って行うようにしよう。アクションシートはUIActionSheetというクラスを使うのだが、これを呼び出すクラスはUIActionSheetDelegateプロトコルに準拠しておくとう便利だ。これも対応しておこう。

CameraViewController.hを開き、次のように編集する。

## CameraViewController.h

```
@interface CameraViewController : UIViewController <UIActionSheetDelegat
{
}

// アクション
- (IBAction)showCameraSheet;

@end
```

ポイントは、2箇所だ。まず、クラスの定義のところで、UIActionSheetDelegateプロトコルに準拠している。これにより、アクションシートを閉じたときに処理を行う事ができる。

そして、アクションシートを表示するためのメソッドを用意している。`showCameraSheet`という名前にしてみた。これを、ツールバーのカメラボタンから呼び出せばいい。  
Interface Builderに戻って、カメラボタンからの接続を行っておいてほしい。

続いて、`CaemeraViewController.m`で実装を行おう。次のメソッドを追加する。

## CameraViewController.m

```
- (IBAction)showCameraSheet
{
    // アクションシートを作る
    UIActionSheet* sheet;
    sheet = [[UIActionSheet alloc]
              initWithTitle:@"Select Sorce Type"
              delegate:self
              cancelButtonTitle:@"Cancel"
              destructiveButtonTitle:nil
              otherButtonTitles:@"Photo Library", @"Camera", @"Saved Photo",
                                [sheet autorelease];

    // アクションシートを表示する
    [sheet showInView:self.view];
}

- (void)actionSheet:(UIActionSheet*)actionSheet clickedButtonAtIndex:(NSInteger)buttonIndex
{
}
```

`showCameraSheet`では、アクションシートの表示を行っている。まず、`UIActionSheet`クラスのインスタンスを作る。初期化メソッドには多くの引数が必要だが、順に、シートのタイトル、デリゲート、キャンセルボタンのタイトル、削除などを行うボタンのタイトル(色が変わる)、他のボタンのタイトル、を指定する。最後の、他のボタンのタイトルは、可変長引数になっているので、カンマで区切りながら必要な数だけ渡す。最後に`nil`を付けるのを忘れないでほしい。

もう1つ実装している`actionSheet:clickedButtonAtIndex:`メソッドは、`UIActionSheetDelegate`で定義されているメソッドだ。このメソッドは、アクションシートでボタンをクリックしたときに呼び出される。このメソッドの中身は、次回に取り上げる。

ここまでのアプリを動作させて、アクションシートを表示させると、次のようになる。



次回は、カメラを使って撮影するためのコードを紹介しよう。

ここまでのソースコード: [Camera-1.zip](#)



毎日コミュニケーションズ

Copyright Mainichi Communications Inc.

