

2019年2月18日

株式会社インプレスR&D

<https://nextpublishing.jp/>

“ちょい足し”実装レシピでUIをもっとリッチに！

## 『iOS アプリ開発 UI 実装であると嬉しいレシピブック』発行

技術書典シリーズ、2月の新刊

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『iOS アプリ開発 UI 実装であると嬉しいレシピブック』(著者:酒井 文也)を発行いたしました。

『技術書典シリーズ』とは、今もっとも注目すべき、エンジニアによるアウトプットの間である技術同人誌イベント「技術書典」で、頒布された同人誌を底本として、商業書籍として刊行する書籍シリーズです。

### 『iOSアプリ開発 UI実装であると嬉しいレシピブック』

<https://nextpublishing.jp/isbn/9784844398455>



著者:酒井 文也

小売希望価格:電子書籍版 1600円(税別)／印刷書籍版 2000円(税別)

電子書籍版フォーマット:EPUB3／Kindle Format8

印刷書籍版仕様:B5判／カラー／本文146ページ

ISBN:978-4-8443-9845-5

発行:インプレス R&D

### <<発行主旨・内容紹介>>

【少しの工夫とアイデアでできるUI表現集！】

本書は、筆者がこれまでサンプル開発や実務の中で培ったノウハウ等から、UI実装を「サイドナビゲーション型のUI」、「写真を拡大する画面遷移UI」など、いくつかのまとまったサンプル実装を例に、UI構築をする上で重要な実装ポイントやアイデアを紹介しています。

(本書の対象読者)

- これからiOS アプリを本格的に開発していこうと考えている方
  - UI 実装や表現に関する部分にさらなる磨きをかけていきたい方
- (本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。)

## 各章で解説するサンプル実装の画面とその難易度を冒頭で紹介

### 第2章

・サンプル実装の難易度: ★★★★★★☆☆

—サンプルプロジェクト名: InteractiveUIExample

<第2章: カスタムトランジションの実装サンプル>

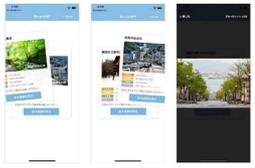


### 第3章

・サンプル実装の難易度: ★★★★★★☆☆

—サンプルプロジェクト名: LikeTinderExample

<第3章: Tinder型のカード選択実装サンプル>



### 第4章

・サンプル実装の難易度: ★★★★★★☆☆

—サンプルプロジェクト名: ReservationFormExample

<第4章: 情報入力フォーム実装サンプル>



### 免責事項

本書に記載された内容は、情報の提供のみを目的としています。したがって、本書を用いた開発、製作、運用は、必ずご自身の責任と判断によって行ってください。これらの情報による開発、製作、運用の結果について、著者はいかなる責任も負いません。

### 表記関係について

本書に記載されている会社名、製品名などは、一般に各社の登録商標または商標、商品名です。会社名、製品名については、本文中では©、®、™マークなどは表示していません。

### 底本について

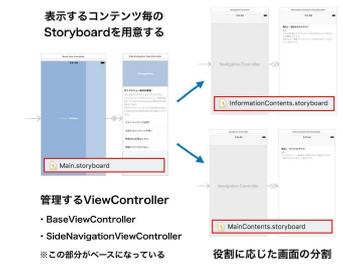
本書籍は、技術系同人誌研究会「技術書典5」で頒布されたものを底本としています。

6 | はじめにはじめに | 7

## 豊富な図解で UI 実装の考え方を解説

### 図 1.3: サンプルのStoryboard 概観

表示するコンテンツ毎のStoryboardを用意する



管理するViewController

- BaseViewController
- SideNavigationViewController

※この部分がベースになっている

役割に応じた画面の分割

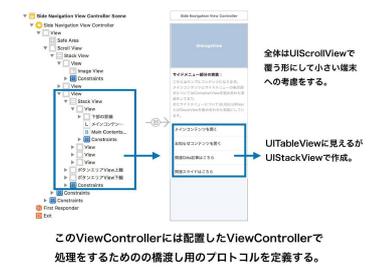
UIで実装する場合には、ContainerViewを配置している部分はお互いの関係をわかりやすいように整理しておくことや、Identifierの設定忘れ等には十分に配慮してください。また、切り替える対象のコンテンツも、画面のまとまりを意識して分割する方針をとり、ひとつのStoryboardが割断にならないようにしましょう。

このようなContainerViewの組み合わせを活用した事例は、ナビゲーションと画面切り替えの制御だけではなく、ひとつの画面の中に属性や振る舞いが異なる要素を多く配置する必要がある場合にも有効です。

#### 1.4.1 サイドナビゲーション部分の実装

まずはSideNavigationViewControllerの実装です。

### 図 1.4: SideNavigationViewController 図解



全体はUIScrollViewで覆う形にして小さい端末への考慮をする。

UITableViewに見えるがUIScrollViewで作成。

このViewControllerには配置したViewControllerで処理するための橋渡し用のプロトコルを定義する。

今回の実装では特に難しいことは行っていません。ただし、ボタン部分のデザインに関しては、項目があらかじめ決まっている形なので、UIScrollViewとUIButtonをうまく組み合わせることで実現しています。この部分については、画面の仕様やデザインによってはUITableView等で行うてもよいでしょう。

また、BaseViewController側で表示する内容を切り替えるための処理の橋渡しを行うプロトコルも定義します。

これらの点をまとめた上で、SideNavigationViewControllerの実装はリスト1.5のような形になります。

#### リスト 1.5: SideNavigationViewController の全体図

```
1: // サイドナビゲーションの押されたボタン種類
2: enum SideNavigationButtonType: Int {
3:     case mainContents = 0
4:     case informationContents = 1
5:     case pageTitle = 2
6:     case sideShareWebPage = 3
7: }
8:
```

## 図とコード実装をあわせて読むことでより豊かな UI 実装を学習

という構造にして、「表示したい1項目に対してヘッダーとセルがひとつずつ存在する」形を取る様にしています。また、UITableViewのヘッダーをそのまま配置しただけではUITableViewのスクロールに合わせて付いてくる形になってしまいます。これを防ぐため、UITableViewは、Interface Builder上で「StyleをGrouped」に設定をしておきます。実装の前段階における準備に関する図解は図4.5の形になります。

図 4.5 アコーディオン型の関係する実装するための画面やViewに関する前準備



ここからは、ヘッダー部分のViewのタップした際にUITableViewCellの表示・非表示が切り替わる処理の実装を説明します。コンテンツの閉閉処理が可能な形にするため、ヘッダー部分のViewに対してタップ検知を行うためにUITapGestureRecognizerを付与します。次にUITableViewに表示するデータについて、セクションごとの閉閉情報は変数「sectionEventLists」(extended: Bool, event: EventEntity)に持たせるようにし、UITapGestureRecognizerが発動したタイミングで閉閉のフラグを更新すると同時にセクションの更新処理を行います。そしてこのタイミングで、セクションごとの表示情報を前述の変数に定義したextendedの値に応じてセクション内に表示するセル数を変更し、この表現を実現しています。具体的なコードはリスト4.8のようになります。

リスト 4.8 アコーディオン型の閉閉処理に関する実装

```
class MainViewController: UIViewController {
    // セクションごとに分けられたイベントデータを格納する変数
    private var sectionEventLists: [(extended: Bool, event: EventEntity)] = []

    @IBOutlet weak private var eventTableView: UITableView!

    ... (対象の実装以外の部分は省略) ...

    override func viewDidLoad() {
        super.viewDidLoad()

        ... (対象の実装以外の部分は省略) ...

        setupEventTableView()
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
    }

    // MARK: - Private Function

    // UITapGestureRecognizerが発動した際に実行されるアクション
    @objc private func eventHeaderViewTapped(sender: UITapGestureRecognizer) {
        guard let headerView = sender.view as? EventHeaderView else {
            return
        }

        // 該当セクションの値をタグから取得する
        let section = Int(headerView.tag)

        // 該当セクションの閉閉状態を更新する
        let changedStatus = !sectionEventLists[section].extended
        sectionEventLists[section].extended = changedStatus

        // 該当セクション番号のUITableViewを更新する
        eventTableView.reloadSections(
            NSIndexSet(index: section) as IndexSet, with: .automatic
        )
    }
}
```

124 | 第4章 入力フォームの実装例

第4章 入力フォームの実装例 | 125

## <<目次>>

### 第1章 サイドナビゲーション型のUI

- 1.1 ContainerView についての確認
- 1.2 ContainerView の活用ポイント
- 1.3 スライドするメニュー表示の概要と実装
- 1.4 StoryBoard 構成とコードの解説
- 1.5 サイドナビゲーション実装における別解

### 第2章 写真を拡大する画面遷移UI

- 2.1 View 実装に関する Tips 集
- 2.2 使用したライブラリーのご紹介
- 2.3 カスタムトランジションの基本実装
- 2.4 画面遷移前の一覧画面の実装
- 2.5 画面遷移後の詳細画面の実装
- 2.6 本サンプルにおける画面遷移表現のまとめ

### 第3章 Tinder 風のUI

- 3.1 実装する上でのポイント
- 3.2 処理の橋渡しを行うプロトコル実装
- 3.3 画面に追加した際の演出
- 3.4 カード状のViewとUIPanGestureRecognizer
- 3.5 UIViewController との連携部分の実装
- 3.6 UIScrollView を利用した画像表示の実装

### 第4章 入力フォームの実装例

- 4.1 入力に関するView部品の実装
- 4.2 使用したライブラリーのご紹介
- 4.3 UITableView を利用した表現 Tips の紹介

#### 4.4 入力フォームの部分に関する画面実装

##### << 著者紹介 >>

酒井 文也(さかい ふみや)

アプリの UI 実装が好きな元デザイナーからジョブチェンジをしたエンジニア。Qiita や Github などでも UI 実装に関するサンプルや解説記事を投稿したり、Swift 愛好会をはじめとする勉強会等でもたまに登壇しています。

アイデアを練ったり、設計のためのメモや図解を作る時はもっぱら手書き派です。見た目と違いお酒はまったく飲みませんが、お酒の席に参加するのは好きです。

Twitter: @fumiyaac

Github : <https://github.com/fumiyaac>

Qiita : <http://qiita.com/fumiyaac@github>

Slideshare: <https://www.slideshare.net/fumiyaakai37>

##### << 販売ストア >>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple Books、紀伊國屋書店 Kinoppy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

※ 各ストアでの販売は準備が整いしたい開始されます。

※ 全国の一般書店からもご注文いただけます。

##### 【株式会社インプレス R&D】 <https://nextpublishing.jp/>

株式会社インプレス R&D (本社：東京都千代田区、代表取締役社長：井芹昌信) は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishing を使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知の流通を目指しています。

##### 【インプレスグループ】 <https://www.impressholdings.com/>

株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:唐島夏生、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「旅・鉄道」「学術・理工学」を主要テーマに専門性の高いメディア&サービスおよびソリューション事業を展開しています。さらに、コンテンツビジネスのプラットフォーム開発・運営も手がけています。

##### 【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

TEL 03-6837-4820

電子メール: [np-info@impress.co.jp](mailto:np-info@impress.co.jp)